Applied Soft Computing Journal 103 (2021) 107166

Contents lists available at ScienceDirect

Applied Soft Computing Journal

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

A Dynamic Ensemble Learning Algorithm based on K-means for ICU mortality prediction

Chonghui Guo *, Mucan Liu, Menglin Lu

Institute of Systems Engineering, Dalian University of Technology, Dalian 116024, China

ARTICLE INFO

Article history:
Received 10 September 2020
Received in revised form 18 December 2020
Accepted 28 January 2021
Available online 5 February 2021

Keywords:
Intensive care units
Patient mortality prediction
Ensemble learning
K-means

ABSTRACT

This research proposes a Dynamic Ensemble Learning Algorithm based on K-means (DELAK) for intensive care unit (ICU) mortality prediction. Nowadays, the widely applied traditional scoring systems, which predict the mortality risk with some scores reflecting the severity of disease and physiological states of patients in ICU, have shown insufficient predictive performance when faced with large volume of data. Although taking advantage of large volume of data, single machine learning model and ensemble learning methods show an inadequate ability to make personalized predictions for each new patient. Dynamic ensemble selection (DES) methods which are widely studied in pattern recognition field can make personalized prediction but they only select a single classifier instead of ensemble of the results from a pool of classifiers. To overcome the limitations mentioned above, this research proposes an ensemble learning algorithm based on K-means sampling and distance-based dynamic ensemble. K-means sampling helps to achieve the diversity of base classifiers and the distance-based dynamic ensemble is a flexible fusion method which creates a personalized combination of results from base classifiers for each new test sample. To evaluate the performance for mortality prediction of the proposed algorithm, comprehensive experiments are conducted on MIMIC-III dataset. The experimental results show that DELAK achieves outstanding performance in term of AUROC and AUPRC compared with other six fusion strategies, traditional scoring systems, classical ensemble models, i.e. AdaBoost, Bagging and random forest, and dynamic ensemble selection methods in most mortality prediction tasks.

© 2021 Elsevier B.V. All rights reserved.

1. Introduction

Intensive care unit (ICU) is an isolation space for the critically ill patients in a hospital, for whom it provides adequate manpower, material resources and comprehensive monitoring. Patients in ICU are faced with great mortality risk. Patient mortality is one of outcomes that reflects the physiological state of a patient, which is critical important for early diseases prevention and timely intervention [1]. Quantifying the patient's health and predicting the mortality risk of patient are focus in the field of critical care research [2]. The accurate prediction of mortality task and patient deterioration in ICU is urgently needed [3].

There are many technologies in ICU to estimate the mortality risk of patients. Some scores provide some ways of predicting mortality risk [2]. Traditional scoring systems, such as MODS [4], SAPS II [5] and SOFA [6], qualify the severity of patients according to their characteristics. However, with the larger volume and higher complexity of data, traditional scores become less competent for the prediction task. The fast developing machine learning technology can take advantage of large volume of data and achieve great prediction performance [7,8]. The prediction of mortality risk of ICU patients based on machine learning helps doctors to learn more about patient physiological state, thus assisting them in medical decision making. Traditional machine learning algorithm like naive Bayes, support vector machines, logistic regression and so on are widely used in mortality risk prediction tasks. And some early researches achieved effective mortality prediction with the help of neural network models [9,10]. However, the single machine learning classifier may have its own limitation over different classification tasks [11]. Ensemble learning, in which a series of base classifiers are built and combined in some way, is a creative and effective idea in machine learning. It can generally achieve better performance than single classifier in accuracy and robustness. But traditional ensemble learning methods lack the ability of making personalized prediction for each new test sample. Dynamic ensemble selection, which is widely studied in the field of pattern recognition, can dynamically select single proper classifier for each new sample in a personalized way, but it does not take fully advantages of ensemble learning methods by integrating the prediction results from a pool of classifiers.

* Corresponding author.
E-mail address: dlutguo@dlut.edu.cn (C. Guo).

https://doi.org/10.1016/j.asoc.2021.107166
1568-4946/© 2021 Elsevier B.V. All rights reserved.

Table 1
Comparison of different methods.

Advantages	Methods				
	Scoring systems	Single classifier	Traditional ensemble methods	Dynamic ensemble selection	DELAK
Ability to deal with large volume of data		✓	✓	✓	✓
Ensemble a pool of classifiers			✓		✓
Personalized prediction for each new test sample				✓	✓

To overcome limitations mentioned above, a Dynamic Ensemble Learning Algorithm based on K-means (DELAK) is proposed. K-means clustering algorithm is used to partition the dataset and get several cluster centroids, which are used to generate several sub datasets for base model training. To measure the predicting ability of each base classifier on each new sample, a distance measurement based method is introduced to determine the weight of each base classifier dynamically. The comparison of advantages of various kinds of methods and our DELAK is presented in Table 1. This proposed method is validated and evaluated on the mortality risk prediction task. Our experiments are based on medical information mart for intensive care III (MIMIC-III) [12] freely-available dataset. This paper is organized as follows: Section 2 introduces some related works about mortality risk prediction and ensemble learning methodologies. Section 3 provides a detailed introduction about DELAK. The experimental design and result analysis are presented in Section 4. Limitations and future directions are discussed in Section 5.

2. Literature review

In this section, scoring systems, such as MODS, SAPS and SOFA, are first introduced. Then the common machine learning technologies for mortality risk prediction are presented. Finally, ensemble learning and dynamic ensemble selection methods are further reviewed.

2.1. Scoring systems for mortality prediction

Scoring systems, such as the multiple organ dysfunction syndrome (MODS), simplified acute physiology score II (SAPS II) and sepsis-related organ failure assessment (SOFA), are widely used in clinical practice. MODS and SAPS II, which reflect the severity of illness based on logistic regression, are both designed to predict the mortality risk. SOFA score describes a sequence of complications in the critically ill, the purpose of which is to complements the existing severity indexes [6]. Based on the hypothesis that the severity of acute disease can be measured by quantifying the degree of abnormality of multiple physiologic variables, the acute physiology and chronic health evaluation (APACHE) is a classification system for measuring severity of illness in groups of critically ill patients [13]. APACHE II is a revised version of original APACHE [14]. These scoring systems assess the severity of illness of patients according to specific illness, vital signs and physiological parameters, to predict the outcome of patients. Most of these scoring systems are based on large-scale data collected from many countries, in which experts play an important role in selecting the predictive indicators closely related to the outcome of patients. Traditional scoring systems do not take fully use of large volume of patient data and can only achieve limited prediction performance.

2.2. Common machine learning technologies for mortality risk prediction

There are many studies choosing various machine learning models to predict the mortality risk to make up for the weak prediction performance of scoring systems. Dybowski et al. [15] evaluate the death during hospital admission. They select 258 patients from an intensive care database of 4484 patients at a London teaching hospital, discovering that the prediction performance of artificial neural network is better than standard logistic regression. However, some researches [16,17] reveal that artificial neural network offers no obvious advantages over the APACHE II model and that logistic regression and artificial neural network have similar performance when the sample size is adequate. Ribas et al. [18] use a variant of support vector machine which achieves better performance than APACHE II score and logistic regression. Kim et al. [19] find that decision tree algorithm slightly outperforms other data mining techniques, followed by the artificial neural network and support vector machine in terms of AUROC score. Krishnan et al. [20] use an extreme learning machine to predict patient-specific ICU mortality and achieve good prediction performance.

Researches mentioned above reveal that no single machine learning model can always outperform the others. The prediction performance of specific single model is highly related to the dataset used, the variables measured and the outcome being tested [21]. By combining various models in various ways, ensemble learning can generally achieve better prediction performance than individual model and make up for the deficiency in generalization and robustness of individual model.

2.3. Ensemble learning and dynamic ensemble selection methods

Ensemble learning is widely used in machine learning field. Such methods can improve the predictive ability by combining the results from single base model. There are mainly two stages in an ensemble learning method, model training and output fusion [22]. The diversity of base models is the premise of effectiveness of ensemble learning methods. Some methods achieve diversity of base models by special sampling methods. Bagging [23] is a classical ensemble algorithm in which the base models are trained on different data subset generated by sampling the original dataset with replacement. Random subspace method [24] achieves the diversity of base models by sampling features. Random forest [25], one of the most effective ensemble algorithms, is a hybridization of Bagging and random subspace methods. Diversity can also be achieved by dividing the original dataset into several non-overlapping subsets, on which different base models are trained. Ting et al. [26] partition the features into non-overlapping subspace using their proposed feature subspace aggregating method, which is designed to improve the predictive performance for both stable and unstable learners. Rahman et al. [27] achieve the partitioning of original dataset through K-means algorithm and classifiers are built for each data subset. Jiang et al. [28] also propose the idea of clustering-based ensemble approach for air quality prediction. Meng et al. [29]

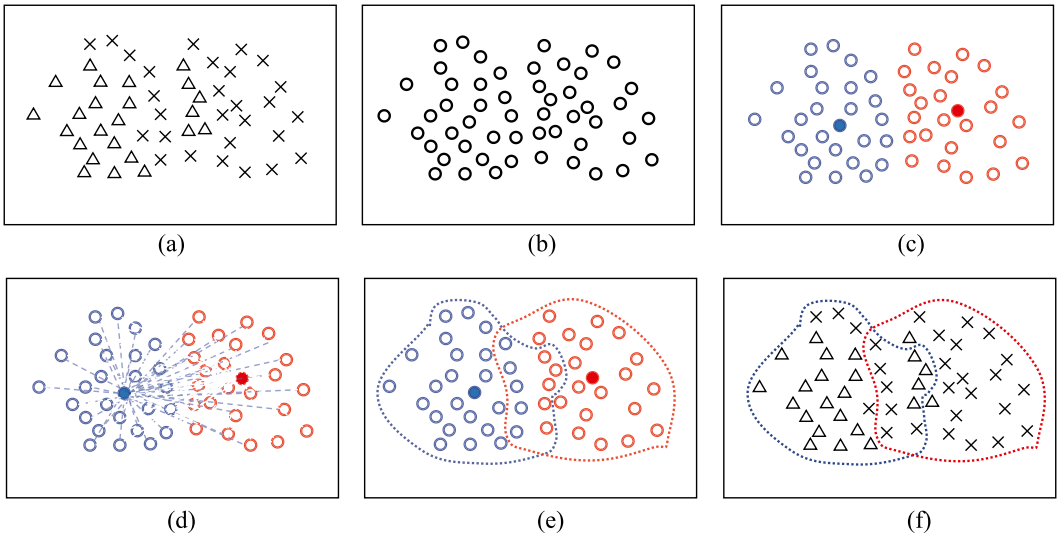


Fig. 1. Process of K-means sampling. (a) shows the given dataset with labels D. (b) shows the dataset X of which the labels are dropped. In (c), clusters and cluster centers are obtained through K-means. The distance between each instance and each cluster center are computed in (d). The closest int($m \times \phi$) instances are chosen into corresponding sub dataset in (e). The labels of instances are returned in (f).

divide features into unrelated groups by an affinity propagation clustering algorithm with a bicor correlation coefficient and construct diversed feature subsets for ensemble learning. There are also lots of researches focusing on output fusion strategy. Haque et al. [30] propose the use of Matthews correlation coefficient (MCC) to measure the weights of each base classifier. A weighted area under the receiver operating characteristic curve ensemble (WAUCE) approach is proposed by Wang et al. [31], which is proven to be reliable and robust on breast cancer diagnosis task.

Some researchers propose ensemble learning algorithm for mortality risk prediction. Pirracchio et al. [32] present a super learner algorithm, which is an ensemble of machine learning models for improving the ICU mortality risk predicting performance. Kwon et al. [33] conduct extreme gradient boosting, light gradient boosting machine, and random forest with quick sepsis-related organ failure assessment (qSOFA) variables to predict three-day mortality, which provide better accuracy than the qSOFA score in the emergency department. Mahmoud et al. [34] propose a stacking ensemble approach in which data are split into six modalities based on the expert's decisions and a separate classifier is selected for each modality based on the performance of the classifiers. Ghorbani et al. [35] propose a new ensemble classifier based on the combination of Stacking and Boosting ensemble methods to create an early mortality prediction model on a highly imbalanced dataset.

Ensemble learning methods can achieve better prediction performance in accuracy and robustness, but they lack the ability of making personalized prediction for each new test sample. In the field of pattern recognition, dynamic ensemble selection (DES) method is widely studied to dynamically make predictions for each new test sample [36–40]. The idea of DES is to select a proper model from a pool of models for each test sample according to the local predictive ability in a local region around the test sample. For example, in dynamic classifier selection based on local accuracy (DCS-LA), classifier which achieve the best predictive performance on the neighbor region of each new test sample is selected [37]. Cruz et al. [36] propose a dynamic ensemble selection framework using meta-learning, in which a meta stage is designed to dynamically select model for new test sample. But few researches apply this dynamic strategy into the fusion stage of ensemble learning.

To make full use of advantages and make up for shortcomings of methods mentioned above, we propose DELAK which innovates in both sampling method to achieve diversity of base models and output fusion method to achieve dynamic prediction. The proposed DELAK is proven to improve the predicting performance on mortality risk prediction task.

3. Methodology

In this section, the proposed DELAK is introduced in detail. There are two main parts innovative in our proposed algorithm: K-means sampling and the distance-based dynamic ensemble method. In K-means sampling, K-means algorithm is used to partition the training datasets. Taking the prediction ability of each base classifier for each new test sample into consideration, the distance-based dynamic ensemble method sets weight of each base classifier according to Euclidean distances between test sample and cluster centers. For each new test sample, a new combination of base classifier is produced, thus achieving a flexible way of ensemble.

3.1. K-means sampling

Clustering is a widely used unsupervised machine learning technology, with the help of which, data without labels can be divided into several partitions. Samples in the same cluster are in high similarity while samples in different clusters are very different. K-means is one of the most widely used cluster algorithms. In DELAK, K-means algorithm is used to generate specified number of cluster centers, then specified number of samples most similar to each cluster centers are found to generate several sub datasets for base classifiers training rather than adopting randomly sampling methods such as bootstrap. The metric of measuring dissimilarity between samples in K-means sampling is Euclidean distance. The process of K-means sampling (choosing two cluster centers as example) is presented in Fig. 1.

In this research, given training dataset $D = \{(\boldsymbol{x_i}, y_i)|\boldsymbol{x_i} \in R^n, y_i \in \{0, 1\}, i = 1, 2, \ldots, m\}$, which is shown as Fig. 1(a), there are labels namely whether a patient died or not during some time window in the dataset. In the stage of clustering, the labels are dropped to get a dataset without labels $X = \{\boldsymbol{x_i}|\boldsymbol{x_i} \in R^n, i = 1, 2, \ldots, m\}$, shown as Fig. 1(b). Given X and the cluster number K, cluster partition $C = C_1 \cup C_2 \cup \cdots \cup C_K$ and corresponding cluster centers $\boldsymbol{\mu_1}, \boldsymbol{\mu_2}, \ldots, \boldsymbol{\mu_K}$ are obtained, shown as Fig. 1(c).

Then the Euclidean distances between each instance in dataset and each cluster center are computed, shown as Fig. 1(d). For each cluster center, the most similar specified number of samples are chosen into the corresponding sub dataset and the partition $C' = C'_1 \cup C'_2 \cup \cdots \cup C'_K$ is obtained, shown as Fig. 1(e). The number of samples in each sub dataset is up to the sampling ratio ϕ, which is the ratio of selected samples to all training samples. Finally, the labels in D are returned to instances in C' to get K data subsets $D' = D'_1 \cup D'_2 \cup \cdots \cup D'_K$. The algorithm of K-means sampling is presented in Algorithm 1.

Algorithm 1 K-means sampling.

Input: Sampling ratio ϕ; Training dataset D; Training data without labels X; Cluster number K
Output: Data subsets $D' = D'_1 \cup D'_2 \cup \cdots \cup D'_K$;

procedure K-MEANS SAMPLING(ϕ, X, K)
  randomly choose K samples from X as the initial cluster centers: $\boldsymbol{\mu_1}, \boldsymbol{\mu_2}, \ldots, \boldsymbol{\mu_K}$
  set $C_i = \varnothing (1 \le i \le K)$
  repeat
    for $j = 1, 2, \ldots, m$ **do**
      compute the Euclidean distances between sample $\boldsymbol{x_j}$ and cluster center $\boldsymbol{\mu_i} (1 \le i \le K)$
      identify the cluster marker of sample $\boldsymbol{x_j}$ according to the closest cluster center: $\lambda_j = \arg\min_{i \in \{1,2,\ldots,K\}} d_{ji}$
      allocate sample $\boldsymbol{x_j}$ to the corresponding cluster: $C_{\lambda_j} = C_{\lambda_j} \cup \{\boldsymbol{x_j}\}$
    end for
    for $i = 1, 2, \ldots, K$ **do**
      compute the new cluster centers: $(\boldsymbol{\mu_i})' = \frac{1}{|C_i|} \sum_{\boldsymbol{x} \in C_i} \boldsymbol{x}$
      if $(\boldsymbol{\mu_i})' \neq \boldsymbol{\mu_i}$ **then**
        update the current center $\boldsymbol{\mu_i}$ as $(\boldsymbol{\mu_i})'$
      else
        remain cluster center as $\boldsymbol{\mu_i}$
      end if
    end for
  until all the cluster center remain constant
  for $i = 1, 2, \ldots, K$ **do**
    compute the Euclidean distances between each instance in training dataset X and cluster center $\boldsymbol{\mu_i}$
    selected int$(m \times \phi)$ instances nearest to the cluster center into partition C'_i
  end for
  return the labels in D to instances in C' to get K data subsets $D' = D'_1 \cup D'_2 \cup \cdots \cup D'_K$
end procedure

3.2. Ensemble learning based on K-means

3.2.1. Static ensemble learning based on K-means

Generally, clustering ensemble represents a framework to associate clusters produced by different clustering models according to some consensus solution, thus improving the stability and robustness of a clustering system [22]. However, we aim to deal with the supervised learning problem with the help of clustering technology.

In this section, we introduce some fusion methods that combine the results from base classifiers in a global way. To distinguish with our proposed dynamic ensemble, we deem following methods as static ensemble learning based on K-means. In research [27], the researchers propose a way of dealing with classification problems with the help of clustering technology. It trains base classifiers directly on each cluster in which the instance labels are returned. And the final result is obtained through the idea of model selection rather than model fusion. However, algorithm mentioned above has a fatal disadvantage. Although the dataset is partitioned into different parts to keep the diversity of base classifiers, lots of information of original dataset is lost. Especially when cluster number is large, there are very few instances for each base classifier. To overcome problems mentioned above, we propose a K-means sampling method presented in Subsection 3.1 to get training subsets $D' = D'_1 \cup D'_2 \cup \cdots \cup D'_K$.

Algorithm 2 The distance-based dynamic ensemble.

Input: Training dataset $D' = D'_1 \cup D'_2 \cup \cdots \cup D'_K$; Test example $\boldsymbol{e_i} = (e_i^1, e_i^2, \ldots, e_i^n)^T$; Cluster centers $\mu = \{\boldsymbol{\mu_i} | i = 1, 2, \ldots, K\}$;
Output: Predicted class label y_{pred}; Predicted probability of positive class y_{prob}.

procedure BASE CLASSIFIERS TRAINING(D')
  for $\eta = 1, 2, \ldots, K$ **do**
    train base model $h_\eta(\cdot)$ on training data subset D'_η
  end for
  Return $h = \{h_\eta(\cdot) | \eta = 1, 2, \ldots, K\}$
end procedure
procedure ENSEMBLE($\boldsymbol{e_i}$,μ,h)
  use each base classifier to predict the label of $\boldsymbol{e_i}$ to get a result vector $\boldsymbol{h(e_i)} = (h_1(\boldsymbol{e_i}), h_2(\boldsymbol{e_i}), \ldots, h_K(\boldsymbol{e_i}))^T$
  compute the Euclidean distance between test example e_i and each cluster centers to get the distance vector $\boldsymbol{d_i} = (d_{i1}, d_{i2}, \ldots, d_{iK})^T$
  compute the weight vector for test example $\boldsymbol{e_i}$: $\boldsymbol{w_i} = (\frac{1/d_{i1}}{\sum_{j=1}^K 1/d_{ij}}, \frac{1/d_{i2}}{\sum_{j=1}^K 1/d_{ij}}, \ldots, \frac{1/d_{iK}}{\sum_{j=1}^K 1/d_{ij}})^T$
  compute the probability of $\boldsymbol{e_i}$ belonging to positive class: $H(\boldsymbol{e_i}) = \boldsymbol{w_i}^T \cdot \boldsymbol{h(e_i)}$, namely y_{prob}.
  if $y_{prob} \le 0.5$ **then**
    $y_{pred} = 0$
  else
    $y_{pred} = 1$
  end if
end procedure

The base classifier training process is presented as the first stage in Algorithm 2. In each sub dataset, a base classifier is trained to learn the decision boundary. Then a series of classifiers corresponding to each cluster center are built. In the stage of predicting, for each test sample $\boldsymbol{e_i} = (e_i^1, e_i^2, \ldots, e_i^n)^T$, a result vector $\boldsymbol{h}(\boldsymbol{e_i}) = (h_1(\boldsymbol{e_i}), h_2(\boldsymbol{e_i}), \ldots, h_K(\boldsymbol{e_i}))^T$ can be obtained through each base classifier $h_1(\cdot), h_2(\cdot), \ldots, h_K(\cdot)$. What to do next is to combine the result vector $\boldsymbol{h}(\boldsymbol{e_i})$ for each test instances into a final result value $H(\boldsymbol{e_i})$, which is the probability of belonging to positive class. For the test dataset $E = \{\boldsymbol{e_i} | \boldsymbol{e_i} \in R^n, i = 1, 2, \ldots, p\}$, a final result vector $\boldsymbol{H(e)} = (H(e_1), H(e_2), \ldots, H(e_p))^T$ is obtained.

We implement several fusion strategies according to [31] to combine the results of the base classifiers, which aims to compare with the proposed DELAK. Following fusion strategies combine the result vector from each base classifier in different ways.

- Average ensemble based on K-means (AvgE)
  AvgE means calculating the average of $\boldsymbol{h(e)}$ as the final predicting probabilities. The strategy can be expressed as

$$\boldsymbol{H(e)} = (\sum_{\eta=1}^{K} \boldsymbol{h}_\eta(\boldsymbol{e}))/K \tag{1}$$

- Maximum ensemble based on K-means (MaxE)

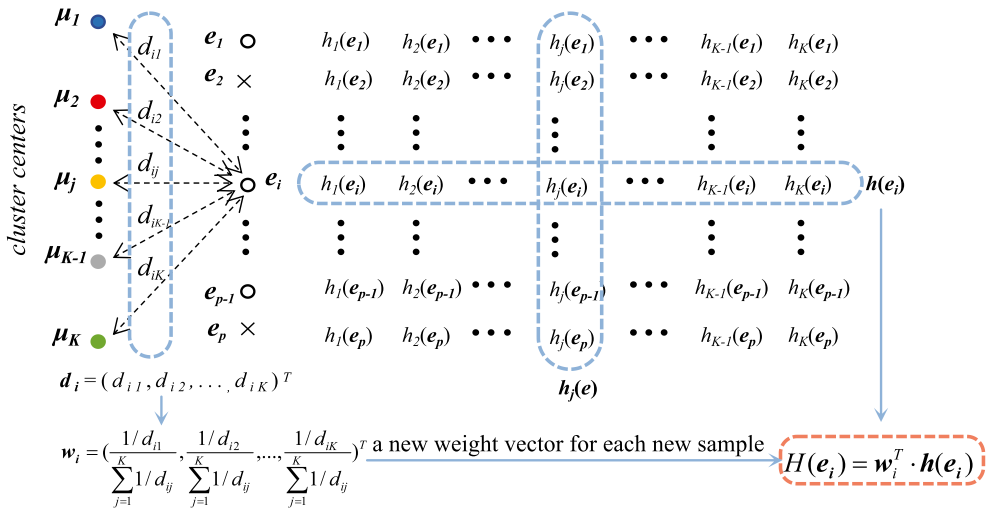


Fig. 2. Presentation of the distance-based dynamic ensemble.

MaxE always chooses the maximum predicting probability in $\{\boldsymbol{h_1}(\boldsymbol{e}), \boldsymbol{h_1}(\boldsymbol{e}), \ldots, \boldsymbol{h_K}(\boldsymbol{e})\}$ as the final result, which tends to predict samples to be positive. MaxE can be expressed as

$$\boldsymbol{H}(\boldsymbol{e}) = \max\{\boldsymbol{h}_\eta(\boldsymbol{e}) | \eta = 1, 2, \ldots, K\} \tag{2}$$

- Minimum ensemble based on K-means (MinE)
  On the contrary, MinE always chooses the minimum predicting probability in $\{\boldsymbol{h_1}(\boldsymbol{e}), \boldsymbol{h_1}(\boldsymbol{e}), \ldots, \boldsymbol{h_K}(\boldsymbol{e})\}$ as the final result, which tends to predict samples to be negative. MinE can be expressed as

$$\boldsymbol{H}(\boldsymbol{e}) = \min\{\boldsymbol{h}_\eta(\boldsymbol{e}) | \eta = 1, 2, \ldots, K\} \tag{3}$$

- Weighted area under the ROC curve ensemble based on K-means (WAUCE)
  This fusion method is based on the creative idea from research [31]. The AUROC performance is used to weight each predicting probability from each base classifier, which can be expressed as

$$\boldsymbol{H}(\boldsymbol{e}) = \sum_{\eta=1}^{K} \frac{AUROC_\eta}{\sum_{j=1}^{K} AUROC_j} \boldsymbol{h}_\eta(\boldsymbol{e}) \tag{4}$$

- Single best (SB)
  In addition, SB, which chooses the base classifier of the best performance to finish the prediction is used as a comparison, which can be expressed as

$$\boldsymbol{H}(\boldsymbol{e}) = \boldsymbol{h}_{\boldsymbol{\Delta}}(\boldsymbol{e}),\ \Delta = \arg\max_{\eta} \{AUROC_\eta | \eta = 1, 2, \ldots, K\} \tag{5}$$

Fusion methods introduced in this section adopt a global way of combining the results obtained from different base classifiers built on different data subset. All the predicting results of a new sample are combined in the same weight vector, which we deem as static fusion method. In the next section, we will introduce a distance-based dynamic fusion method.

3.2.2. Distance-based dynamic ensemble based on K-means

In addition to introducing K-means sampling, we also propose a distance based dynamic model fusion method in DELAK. The first stage of DELAK is K-means sampling like methods mentioned above. However, a dynamic weighted ensemble strategy is adopted in the model fusion stage. Shown as Fig. 2, for each new sample $\boldsymbol{e_i} = (e_i^1, e_i^2, \ldots, e_i^n)^T$, a result vector $\boldsymbol{h}(\boldsymbol{e_i}) = (h_1(\boldsymbol{e_i}), h_2(\boldsymbol{e_i}), \ldots, h_K(\boldsymbol{e_i}))^T$ can be obtained through each base classifier $h_1(\cdot), h_2(\cdot), \ldots, h_K(\cdot)$. Then the Euclidean distance between test example $\boldsymbol{e_i}$ and each cluster centers is computed as a vector $\boldsymbol{d_i} = (d_{i1}, d_{i2}, \ldots, d_{iK})^T$. This distance measures how much a new sample data belongs to a certain cluster, and also measures the ability of the classifier trained on that cluster to predict the new sample $\boldsymbol{e_i}$. The closer a new sample is to a certain cluster center, the better predictive effect may be obtained using the corresponding base classifier. Larger distance means relatively weaker predictive ability of a base classifier to the new sample $\boldsymbol{e_i}$. For a new sample, the weight vector is the normalized reciprocal of the Euclidean distance, i.e. $\boldsymbol{w_i} = (\frac{1/d_{i1}}{\sum_{j=1}^{K} 1/d_{ij}}, \frac{1/d_{i2}}{\sum_{j=1}^{K} 1/d_{ij}}, \ldots, \frac{1/d_{iK}}{\sum_{j=1}^{K} 1/d_{ij}})^T$. And the final result can be calculated as $H(e_i) = \boldsymbol{w_i}^T \cdot \boldsymbol{h}(\boldsymbol{e_i})$. The algorithm of predicting and ensemble is presented as Algorithm 2.

With different combination of base classifier, number of clusters and sampling ratio, a model space is generated. When dealing with different task, the target to achieve varies. The DELAK is formulated as an optimization problem. What to do is to find the model which can achieve best performance on specify target from the model space. The objective function is presented as

$$\max_{\phi, K, f} E(y_i, F(\boldsymbol{x_i} | \phi, K, f) | i = 1, 2, \ldots, m) \tag{6}$$

where E is the performance evaluation measurement, $\boldsymbol{x_i}$ is the feature vector, y_i is the true label of example in validation dataset, F is the function determined by Dynamic Ensemble Learning Algorithm based on K-means, $\phi \in (0, 1)$ is the sampling ratio, $K \in Z$ is the number of clusters and f is the base classifier. In our research, the performance evaluation measurement is AUROC score. When base classifier is selected, the cluster number and sampling ratio can be determined with grid search strategy.

4. Experimental results and analysis

To evaluate the performance of the proposed DELAK, several experiments are conducted on publicly available medical information mart for intensive care III (MIMIC-III) dataset, which is the basis of many recently public researches in the field of medical data mining [41–45]. To obtain the data used for the mortality prediction task, we conduct data extraction process and data preprocessing.

Table 2
Predictors used in the experiments.

Feature	Item ID	Item name	Table
Glasgow Coma Scale	184	GCSEyes	CHARTEVENTS
	220739	Eye Opening	CHARTEVENTS
	454	GCSMotor	CHARTEVENTS
	223901	Motor Response	CHARTEVENTS
	723	GCSVerbal	CHARTEVENTS
	223900	Verbal Response	CHARTEVENTS
White Blood Cells Count	51301	White Blood Cells	LABEVENTS
	51300	WBC Count	LABEVENTS
Potassium Level	50971	Potassium	LABEVENTS
	50822	Potassium, whole Blood	LABEVENTS
Po2	50821	pO2	LABEVENTS
Serum Bicarbonate Level	50882	Bicarbonate	LABEVENTS
Sodium Level	50983	Sodium	LABEVENTS
Urea Nitrogen (Bun)	51006	Urea Nitrogen	LABEVENTS
Bilirubin, Total	50885	Bilirubin, Total	LABEVENTS
Temperature	678	Temperature F	CHARTEVENTS
	223761	Temperature Fahrenheit	CHARTEVENTS
	676	Temperature C	CHARTEVENTS
	223762	Temperature Celsius	CHARTEVENTS
Urine Output	40055	Urine Out Foley	OUTPUTEVENTS
FiO2	223835	Inspired O2 Fraction	CHARTEVENTS
	190	FiO2 Set	CHARTEVENTS
Heart Rate (HR)	211	Heart Rate	CHARTEVENTS
	220045	Heart Rate	CHARTEVENTS
Systolic Blood Pressure (SBP)	220179	Noninvasive Systolic Blood Pressure	CHARTEVENTS
	455	NBP[Systolic]	CHARTEVENTS
Age	–	–	PATIENTS
Admission Type	–	–	ADMISSIONS

Table 3
Ratio of died patients in different time window.

Time window	Ratio (%)
72 hours	4.93
in-hospital	18.67
30 days	18.96
1 year	33.32

4.1. Dataset and preprocessing

MIMIC-III dataset collects exhaustive clinical data of 58,976 admissions admitted to an ICU at the Beth Israel Deaconess Medical Center from 2001 to 2012. The data preprocessing is similar as our previous research [46]. To achieve a global view of mortality prediction, we do not focus on patients with some specified diseases but all the patients. After removing records duplicate, we obtain totally 42,145 admission of them; patients of age less than 15 are excluded. To keep the personal information from leakaging and ensure similar experimental settings with related works, we only use the first ICU admissions of each patient. In the MIMIC-III database, bedside monitoring data, laboratory test data, input events and output events all consist of time series with time tags. The data of selected predictors in our research are obtained from three tables: chartevents, labevents and outputevents. Referring to relevant researches [2,32], we choose predictors used in SAPS II, as shown in Table 2. We use the separate raw value of each predictor. For example, we treat GCSVerbal, GCSMotor, and GCSEyes from the glasgow coma scale (GCS) score as separate features. For the time series data, we choose the maximum, minimum, mean and standard deviation value as the input features. We only use the data of the first 24 h after the patient was admitted to the ICU.

Data preprocessing mainly includes handling missing values, noisy values and duplicated values. We remove patients with more than 30% missing data. Predictors with more than 40% missing values are also removed. Then the remaining missing values are dealt with interpolation. Abnormal values of each predictor which are significantly out of the clinical normal range are processed. For example, to protect information of surviving patients older than 90, the ages of them are recorded as 300. Here, we replace it with median value. In addition, duplicated records are removed and units are transformed into consistent form. Interval values are represented by the median values of them. Ultimately, 6855 admissions are obtained after preprocessing. Fig. 3 shows the patient cohort selection inclusion criteria and the data extraction process. The median age of the patients in our dataset is 65 and 58.9% are male patients. The ratio of died patients to all patients in different time window is represented in Table 3. The 72 hours mortality prediction task is faced with a relatively large data imbalance problem.

4.2. Design of experiments

To comprehensively figure out how DELAK performs, we design mainly three parts of experiments. In the first part, we use several common machine learning models, including K-nearest neighbor (KNN), neural network (NN), support vector machine (SVM), logistic regression (LR) and decision tree (DT), as base classifiers to select a best one. These models are set as default parameters in scikit-learn software [47]. We explore how AUROC score varies with different number of clusters and different sampling ratio. In the experiments of varying number of clusters, we set the sampling ratio as 0.5 and the cluster number ranging from 1 to 15 with step size of 1. In the experiment of varying sampling ratio, we set the cluster number as 10 and the sampling ratio ranging from 0.1 to 0.9 with step size of 0.1. In the second part, we choose several static fusion methods introduced in Section 3.2.1 to prove the effectiveness of the distance-based dynamic ensemble. The number of clusters is set as 10 and the sampling ratio is set as 0.5. Finally, we also compare DELAK with traditional scoring systems, classical ensemble methods including Bagging, random forest and AdaBoost [48] and several outstanding dynamic ensemble selection methods including META-DES proposed in [36], KNORAE proposed in [38], KNOP proposed in [39], and DES-MI proposed in [40]. In this part, we conduct grid search to select a combination of cluster number and sampling ratio for each task, the objective is to maximize the AUROC score

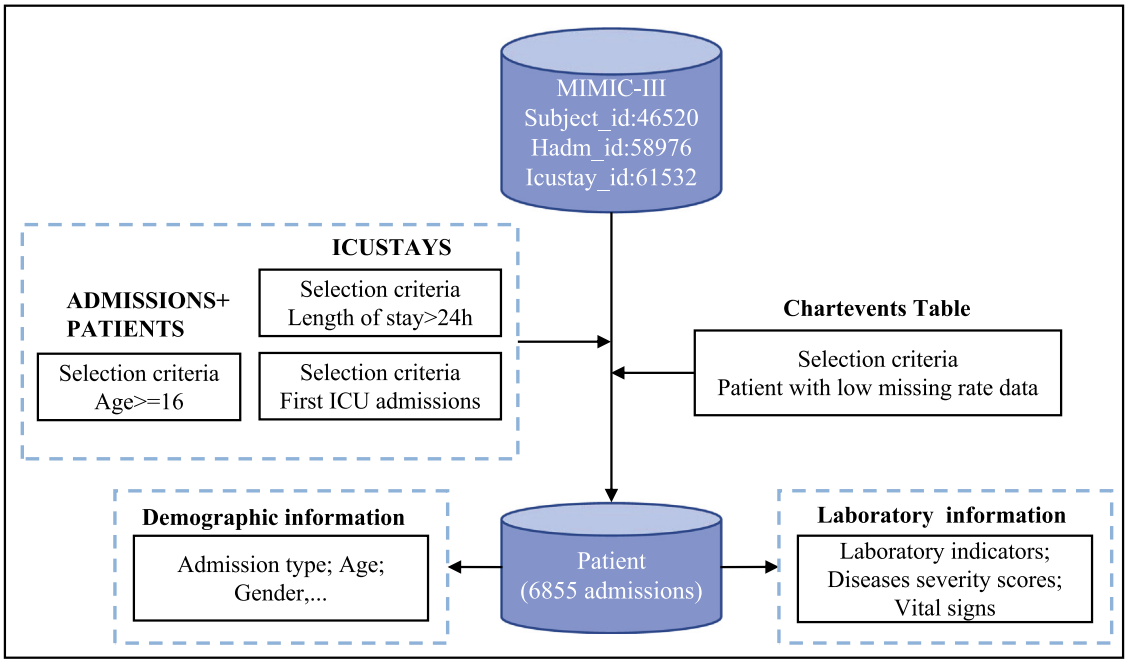


Fig. 3. Patient cohort selection inclusion criteria and data extraction process.

on validation dataset. According to the experimental results mentioned above, we choose NN as base classifier. The sampling ratio is searched in the range of 0.8–0.95 with step size of 0.05 and the number of clusters is searched in the range of 11–15 with the step size of 1. The number of base classifiers of Bagging, random forest and AdaBoost is set as 20, 100 and 200. For dynamic ensemble selection methods, we create a base classifier pool composed of 15 NNs trained in the way of Bagging. A stratified K-fold cross-validation method is adopted to estimate the generation error for our model. To eliminate as much randomness brought by dataset partition as possible, each prediction task is repeated with five replications. There are K iterations in each K-fold cross-validation. The original dataset is divided into K partition of roughly equal size and the ratio of the two classes is the roughly the same as original dataset in each partition. With one fold of data as test data, the last K-1 folds of data are partitioned into training data and validation data. The AUROC scores on validation dataset is used to select base classifier, number of clusters and sampling ratio. In our experiment, the random seeds are set when conducting the K-fold cross-validation to keep the data partition for each model the same, thus the performance differences would entirely come from models rather than the randomness.

The overall performance of single replication is obtained by average the results of each iteration, i.e.

$$\bar{P} = \frac{1}{K}\sum_{k=1}^{K} P_K \tag{7}$$

where P_K is the performance metric, which in our study is mainly area under the receiver operating characteristic curve (AUROC) score; K is the number of iterations in each replication.

In the last two parts of experiments, we also present the area under the precision–recall curve (AUPRC) of each prediction task. AUROC and AUPRC are adopted to evaluate the discrimination ability of the model in the condition that data are imbalanced labeled. The higher the AUROC and the AUPRC are, the better the model is. For the purpose of comparing the robustness of different models, standard deviation of the experimental results of 5 replications is computed.

Our experiments are conducted with Python 3.7.3. The base classifiers and classical ensemble learning methods are implemented with scikit-learn 0.21 packages.

4.3. Experimental results and analysis

In this section, the results and analysis of the three parts of experiments mentioned above are presented.

4.3.1. Comparison of base models

The experiments of varying cluster number and sampling ratio are separately conducted in this section. The AUROC scores on validation dataset of the proposed algorithm with different base classifiers are presented in Figs. 4–7. It should be noted that AUROC scores with base model SVM, LR and NN are presented with the left axis while the scores of DT and KNN are presented with the right axis, which aims to reflect the trend of results of all base classifiers clearly. In the results of studying the effects of cluster numbers, it can be observed that as the number of cluster increases, the AUROC scores show an upward trend. For relatively weaker classifiers DT and KNN, the increase of AUROC scores is much larger than SVM, LR and NN. When the number of clusters varies from 1 to 3 or 4, there is a relatively large leap for all base classifiers in all tasks, the reason of which is mainly that with small cluster numbers, the training dataset is underutilized. When the number of clusters reaches around 10, the promotion becomes slow. However, for DT, the almost unchanged trend is not as apparent as other base classifiers, which means we can still increase the performance by increasing the number of clusters. For all the mortality predicting tasks, NN achieves the highest score when cluster number is in the range of 3–15. In the study of effect of sampling ratio, the results are more interesting. The trend of AUROC score differs with different base classifiers while the trends are similar when choosing the same base classifier for different tasks. For SVM and NN, the AUROC scores increase as the sampling ratio increases. One exception is SVM for 72 hour mortality prediction task, the reason of which is that SVM with default parameters can hardly cope with the imbalance data and the degree of imbalance increases with larger sampling ratio. For

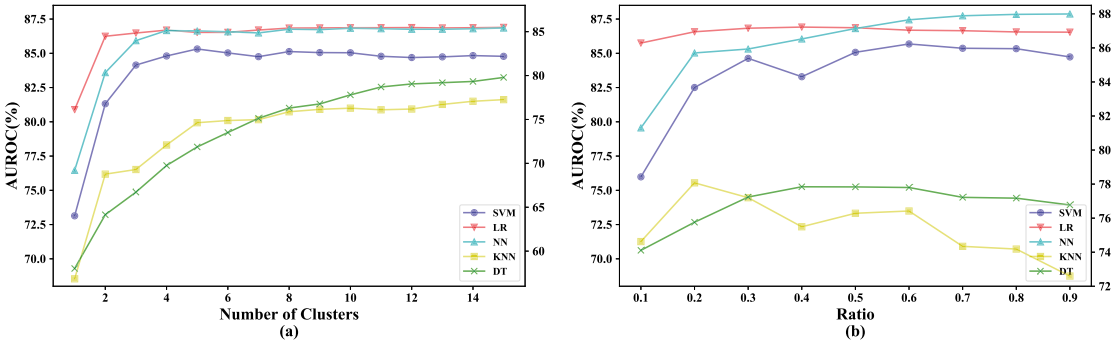


Fig. 4. 72 hours mortality prediction.

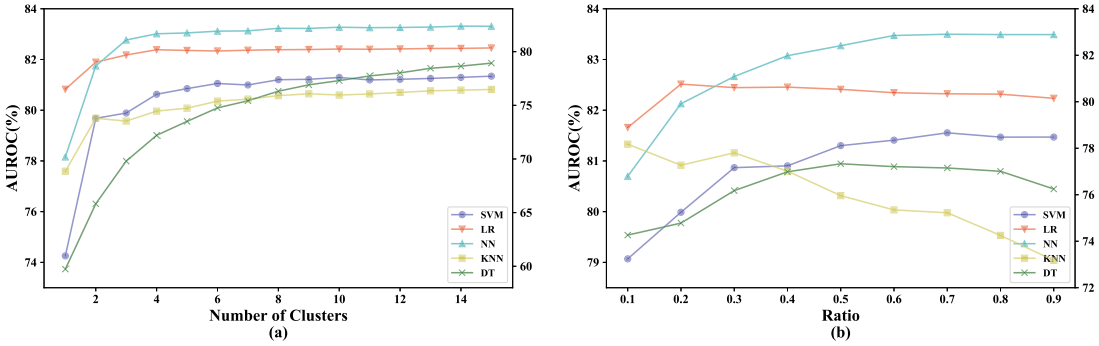


Fig. 5. In-hospital mortality prediction.

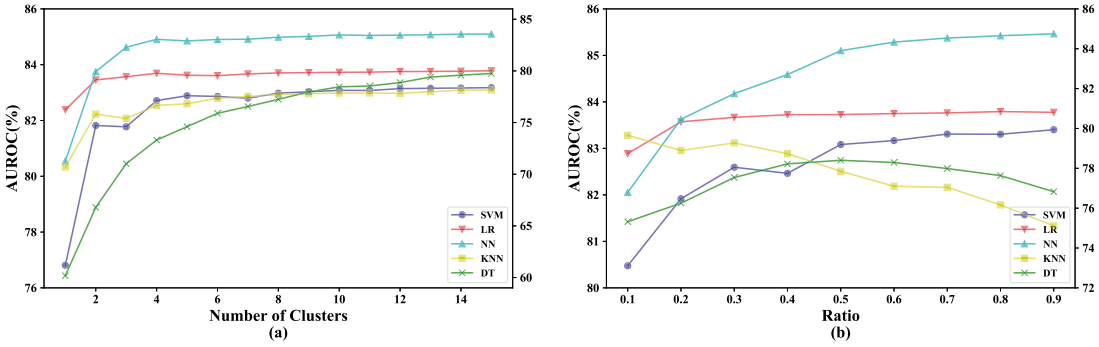


Fig. 6. 30 days mortality prediction.

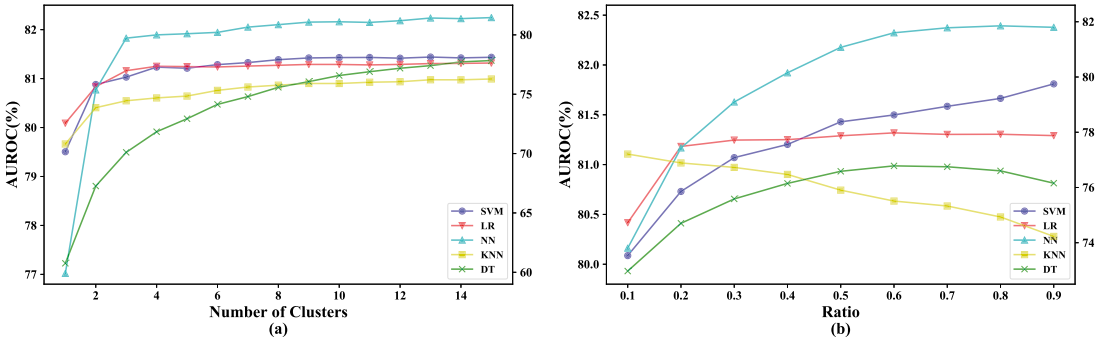


Fig. 7. 1 year mortality prediction.

Table 4
Comparison of 72 hour mortality prediction with different fusion methods.

	AUROC		AUPRC	
	Mean (%)	Std (%)	Mean (%)	Std (%)
DELAK	**86.54**	**0.38**	**31.22**	1.09
WAUCE	85.87	0.49	28.69	1.04
AvgE	86.03	**0.38**	29.38	**0.95**
MaxE	84.07	0.45	19.18	1.72
MinE	74.29	0.96	19.90	**0.97**
CBE	**86.43**	0.66	**35.40**	1.46
SB	77.05	3.30	20.13	3.69

Table 5
Comparison of in-hospital mortality prediction with different fusion methods.

	AUROC		AUPRC	
	Mean (%)	Std (%)	Mean (%)	Std (%)
DELAK	**83.91**	**0.19**	**53.94**	**0.25**
WAUCE	**83.69**	0.28	**53.30**	**0.29**
AvgE	83.66	**0.19**	53.17	0.31
MaxE	80.68	0.55	42.02	1.09
MinE	80.36	0.36	49.50	0.67
CBE	83.24	0.26	52.88	0.44
SB	80.09	0.79	47.13	0.97

Table 6
Comparison of 30 day mortality prediction with different fusion methods.

	AUROC		AUPRC	
	Mean (%)	Std (%)	Mean (%)	Std (%)
DELAK	**85.69**	0.15	**58.00**	**0.33**
WAUCE	**85.44**	**0.11**	57.03	**0.34**
AvgE	85.37	0.15	57.06	0.39
MaxE	82.47	0.24	44.63	1.34
MinE	81.99	0.53	52.65	0.67
CBE	85.13	**0.11**	**57.96**	0.72
SB	82.53	0.48	51.04	1.10

Table 7
Comparison of 1 year mortality prediction with different fusion methods.

	AUROC		AUPRC	
	Mean (%)	Std (%)	Mean (%)	Std (%)
DELAK	**82.62**	**0.11**	**68.49**	0.29
WAUCE	82.39	**0.12**	**68.24**	**0.25**
AvgE	**82.40**	0.14	68.20	**0.19**
MaxE	78.35	0.46	56.85	0.41
MinE	79.86	0.53	65.26	0.45
CBE	81.92	0.19	67.29	0.72
SB	79.72	0.40	64.43	0.42

KNN and DT, the scores increase first and then decrease as the sampling ratio increases. And the best scores are obtained when the sampling ratio is around 0.2 or 0.3 for KNN. When the ratio is too large, the training data can be too complex for KNN with default parameters. The effect of sampling ratio is relatively small for logistic regression. According to results presented above, we choose NN as base classifier in the following experiments.

4.3.2. Comparison of different fusion strategies

In this part, the proposed DELAK is compared with the other five fusion methods. The method of cluster-based ensemble of classifiers proposed in [27] is also implemented as comparison. In this part, we directly identify the number of clusters and sampling ratio, so the AUROC and AUPRC scores on test dataset are presented. To compare the effectiveness improvement and model reliability, average value and standard deviation of AUROC and AUPRC scores are collected and presented in Tables 4–7 and Figs. 8–11. The highest two average values of scores and the lowest two standard deviation values are highlighted in bold. Our method achieves the highest AUROC and AUPRC on all tasks except the AUPRC of the 72 hour mortality prediction, which shows our proposed method has great advantages over other fusion methods in the term of AUROC and AUPRC. Fusion methods combining the results from each base classifier generally outperform those adopted the results from a single base classifier. The SB represents the best performance a single base classifier can achieve. It can be observed that effectiveness and reliability of fusion methods greatly outperform single best base classifier.

4.3.3. Comparison with scoring systems, classical ensemble learning and dynamic ensemble selection methods

In this part, the results of proposed DELAK, traditional scoring systems, three most common ensemble learning methods and several outstanding dynamic ensemble selection methods are presented in Table 8. From the table, we can find that AUROC and AUPRC scores of scoring systems including SOFA, MODS, SAPS and NEW SAPS II are obviously lower than other methods based on machine learning techniques. As for dynamic ensemble selection methods, META-DES gets highest AUPRC in in-hospital and 1 year mortality prediction tasks. DES-MI performs well in the in-hospital mortality prediction task. But all dynamic ensemble selection methods are of obviously poor performance in the task of 30 days mortality. Among classical ensemble learning methods, random forest achieves top two AUROC and AUPRC in several tasks and it performs more stable than dynamic ensemble selection methods in which predictions are made by dynamically selecting only one classifier. And our proposed DELAK which takes advantages of both ensemble learning and dynamic ensemble selection achieves better performance than best ensemble learning method random forest and dynamic ensemble selection methods in ICU patient mortality prediction.

4.4. Discussion

By conducting a series of experiments, we explore the effect of base classifiers, cluster numbers and sampling ratio in DELAK and compared DELAK with several different ensemble fusion methods, scoring systems, classical ensemble methods and dynamic selection methods. In the part of base model comparison, we present a changing relationship between AUROC score and cluster numbers and sampling ratio. It shows that for weaker classifiers like KNN and DT, which can greatly outperform single base classifier with DELAK, may achieve highest AUROC score with over 20 cluster numbers. For stronger classifiers like SVM, LR and NN, the increase of score becomes slower when the number of clusters reaches 10. According to results presented above, we can set proper parameter range for different base classifier. Besides, different model corresponds to different optimal sampling ratio, of which the suggested ranges are presented in Section 4.3.1. For a specific task, we can choose a best base classifier for it. In our study NN is selected as the best base classifier, for example. In the part of fusion methods comparison, it is observed that our proposed DELAK outperforms all the other fusion methods, especially single best classifier, in terms of the mean value and standard deviation of AUROC and AUPRC, which means our proposed DELAK has apparent advantages of effectiveness and reliability over other fusion methods. Finally, we compare traditional scoring systems, classical ensemble learning methods and several dynamic selection methods with our proposed DELAK. From the experimental results, we can find that methods based on machine learning can perform better than traditional scoring systems. Our proposed DELAK which takes advantages of both ensemble learning and dynamic ensemble selection can achieve better prediction performance than ensemble learning and dynamic ensemble selection methods in our study.

Table 8
Comparison of scoring systems, classical ensemble methods and dynamic ensemble selection methods.

	72 hours mortality		In-hospital mortality		30 days mortality		1 year mortality	
	AUROC (%)	AUPRC (%)	AUROC (%)	AUPRC (%)	AUROC (%)	AUPRC (%)	AUROC (%)	AUPRC (%)
SOFA	72.51±0.46	17.88±0.50	68.22±0.07	38.89±0.11	60.66±0.15	39.97±0.31	64.82±0.10	61.59±0.07
MODS	73.18±0.22	15.57±0.71	69.33±0.05	35.20±0.09	60.58±0.48	38.81±0.17	64.89±0.14	60.69±0.14
SAPS	74.78±0.23	17.37±0.52	74.24±0.02	39.59±0.69	66.31±0.17	42.80±0.22	71.69±0.10	66.12±0.16
NEW SAPSII	76.65±0.12	20.24±0.64	76.05±0.04	43.87±0.23	70.63±0.56	48.15±0.04	76.17±0.02	72.13±0.07
AdaBoost-20	85.08±0.64	26.59±0.88	81.68±0.05	49.49±0.73	82.89±0.23	51.68±0.45	80.43±0.18	65.06±0.52
AdaBoost-100	84.57±0.35	27.02±1.40	82.01±0.18	50.62±0.54	83.74±0.33	53.31±0.27	81.01±0.27	65.77±0.41
AdaBoost-200	81.82±1.18	26.80±1.10	80.92±0.32	49.20±0.42	82.67±0.50	52.00±0.40	79.97±0.20	64.47±0.36
Bagging-20	80.71±0.70	26.44±0.74	80.54±0.33	47.03±0.66	81.48±0.46	48.78±0.54	79.90±0.48	62.37±0.92
Bagging-100	77.01±1.39	22.26±1.64	78.40±0.34	42.76±0.40	79.50±0.24	45.13±0.86	78.32±0.31	59.56±0.66
Bagging-200	85.21±0.35	32.85±0.74	82.80±0.11	52.33±0.47	83.44±0.21	54.06±0.19	81.98±0.23	66.67±0.37
RadndomForest-20	81.42±0.70	27.68±1.19	81.03±0.14	47.74±0.47	81.72±0.15	49.73±0.55	79.71±0.47	62.24±0.79
RadndomForest-100	86.01±0.17	34.64±0.61	83.34±0.11	53.49±0.51	84.16±0.15	55.23±0.74	82.02±0.28	66.79±0.86
RadndomForest-200	**86.59±0.30**	**35.26±0.45**	83.65±0.06	54.07±0.28	**84.34±0.22**	**56.01±0.41**	**82.26±0.25**	67.72±0.73
META-DES Cruz et al. [36]	85.49±0.24	27.62±1.22	84.20±0.22	**55.97±0.65**	73.07±0.49	50.14±0.63	80.19±0.24	**76.37±0.16**
KNOP Ko et al. [38]	83.36±0.77	25.82±1.37	82.62±0.20	54.19±0.88	71.37±0.63	49.21±0.76	79.34±0.26	74.64±0.15
KNORAE Cavalin et al. [39]	85.36±0.17	26.79±0.47	83.88±0.28	55.34±0.63	72.28±0.41	49.09±0.65	79.67±0.29	75.69±0.29
DES-MI García et al. [40]	85.59±0.21	28.03±1.01	**84.31±0.11**	**55.96±0.69**	73.04±0.55	50.09±1.07	80.28±0.23	**76.36±0.35**
DELAK	**87.09±0.51**	**35.24±1.21**	**84.20±0.20**	54.40±0.38	**86.05±0.06**	**59.06±1.00**	**82.94±0.11**	69.07±0.30

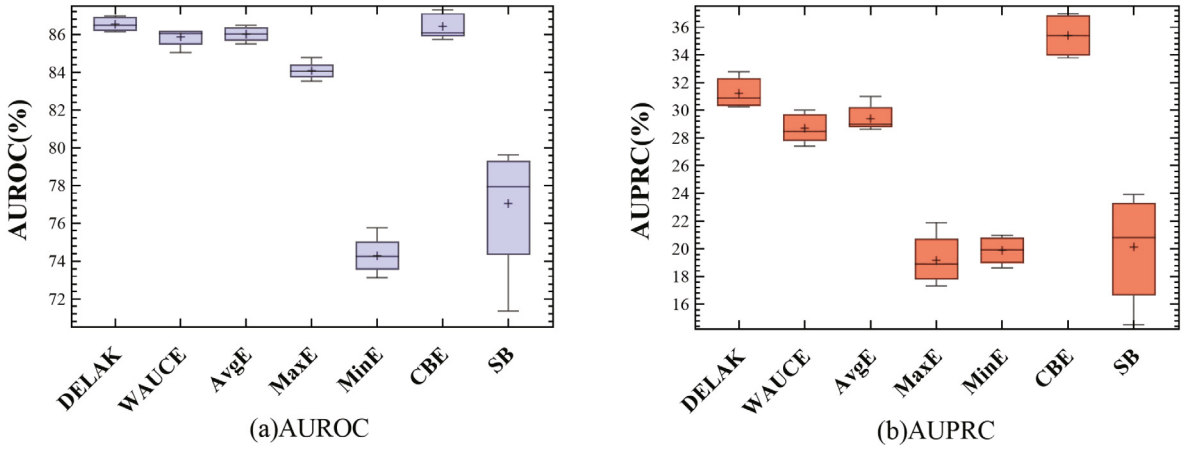


Fig. 8. 72 hours mortality prediction.

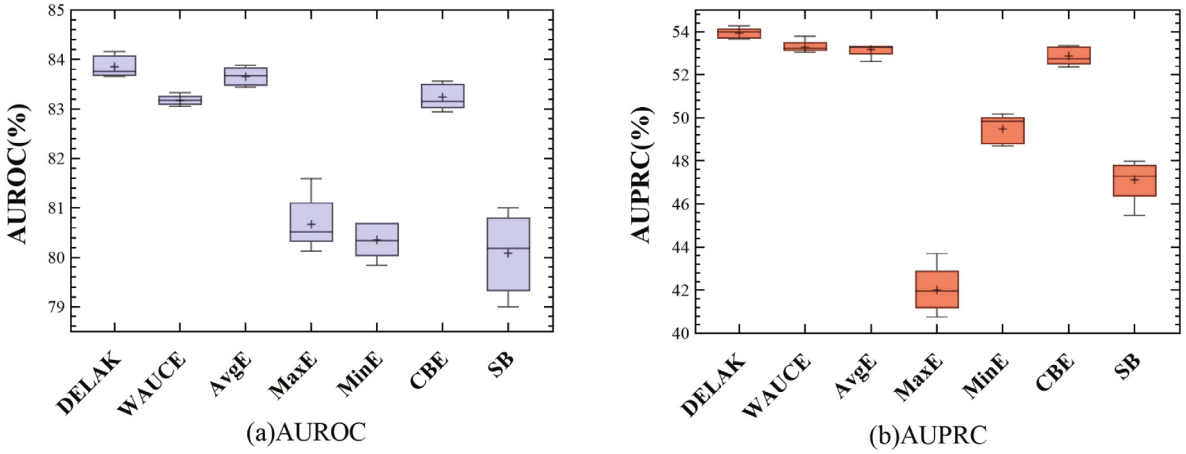


Fig. 9. In-hospital mortality prediction.

5. Conclusion, limitations and future work

In this research, a novel effective Dynamic Ensemble Learning Algorithm based on K-means (DELAK) is proposed for mortality prediction. The innovations are mainly reflected in two aspects, K-means sampling and the distance based dynamic ensemble. The former one generates several data subsets through K-means, on which base classifiers can learn the classification boundary more focused. The distance based dynamic ensemble method generates personalized combination for each new instance instead of in a global way. By this means, different base classifiers are combined in a way that works best for each new instance. The selection of optional base classifiers, cluster numbers and sampling ratio is explored. The comparison of several fusion methods reveals

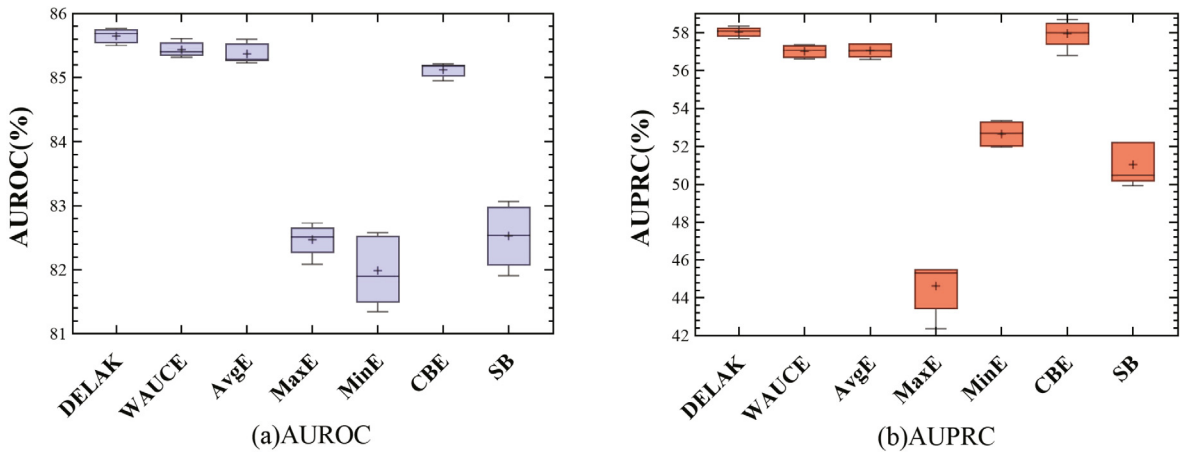


Fig. 10. 30 days mortality prediction.

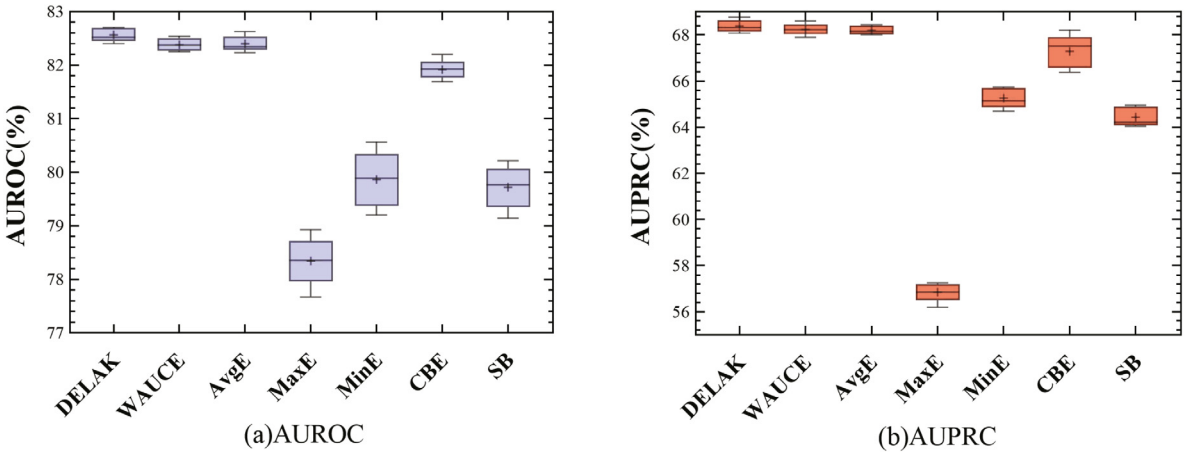


Fig. 11. 1 year mortality prediction.

methods taking each base classifier's prediction results into consideration outperform those adopt single classifier's decision, in which the DELAK has the most outstanding performance. DELAK is also competitive with scoring systems, classical ensemble learning methods and dynamic ensemble selection methods in ICU patient mortality prediction tasks.

However, our method also has some limitations. Firstly, there are some essential flaws in K-means algorithm, which may get into local optimization. Then the search for base classifier may be a relatively time-consuming process. In the future, there are many aspects in which can be carried out research. Firstly, the DELAK can be used for other classification tasks. Then, the clustering method like Gaussian mixture model (GMM) can be used to finish the first sampling stage, and other similarity measurement like cosine similarity can be also taken into consideration. Also, researches on searching optimal sampling ratio and cluster numbers can be further refined.

CRediT authorship contribution statement

Chonghui Guo: Supervision, Writing - review & editing , Funding acquisition. **Mucan Liu:** Conceptualization, Methodology, Software, Validation, Visualization, Writing - original draft. **Menglin Lu:** Conceptualization, Methodology, Validation, Visualization, Writing - review & 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.

Acknowledgments

This research is supported by the National Natural Science Foundation of China (No. 71771034), the Science and Technology Program of Jieyang, China (No. 2017xm041) and the Scientific and Technological Innovation Foundation of Dalian, China (No. 2018J11CY009). The authors would also like to thank the MIT Laboratory for Computational Physiology and collaborating research groups for providing the freely available database (MIMIC-III).

References

[1] L. Zimmerman, P. Reyfman, A. Smith, Z. Zeng, A. Kho, L. Sanchez-Pinto, Y. Luo, Early prediction of acute kidney injury following ICU admission using a multivariate panel of physiological measurements, BMC Med. Inform. Decis. Mak. 19 (Suppl 1) (2019) 16.

[2] S. Purushotham, C. Meng, Z. Che, Y. Liu, Benchmark of deep learning models on large healthcare MIMIC datasets, J. Biomed. Inform. 83 (2018) 112–134.

[3] J. Finnell, B. Dixon, Clinical Informatics Study Guide: Text and Review, 2016.

[4] J. Marshall, D. Cook, N. Christou, G. Bernard, C. Sprung, W. Sibbald, Multiple organ dysfunction score: A reliable descriptor of a complex clinical outcome, Crit. Care Med. 23 (1995) 1638–1652.

[5] J.-R. Le Gall, S. Lemeshow, F. Saulnier, A new simplified acute physiology score (SAPS II) based on a European/North American multicenter study, JAMA 270 (1993) 2957–2963.
[6] J.-L. Vincent, R. Moreno, J. Takala, S. Willatts, A. Artigas, H. Bruining, C. Reinhart, P. Suter, L. Thijs, The SOFA (Sepsis-Related Organ Failure Assessment) score to describe organ dysfunction/failure, Intensive Care Med. 22 (1996) 707–710.
[7] T. Verplancke, S. Looy, D. Benoit, S. Vansteelandt, P. Depuydt, F. Turck, J. Decruyenaere, Support vector machine versus logistic regression modeling for prediction of hospital mortality in critically ill patients with haematological malignancies, BMC Med. Inform. Decis. Mak. 8 (2008) 56.
[8] S. Sakr, R. El Shawi, A. Ahmed, W. Qureshi, C. Brawner, S. Keteyian, M. Blaha, M. Al-Mallah, Comparison of machine learning techniques to predict all-cause mortality using fitness data: The Henry ford exercise testing (FIT) project, BMC Med. Inform. Decis. Mak. 17 (2018) 174.
[9] G. Doig, K. Inman, W. Sibbald, C. Martin, J. Robertson, Modeling mortality in the intensive care unit: Comparing the performance of a back-propagation, associative-learning neural network with multivariate logistic regression, in: Proceedings Annual Symposium on Computer Application in Medical Care, 1993, pp. 361–365.
[10] C. Hanson, B. Marshall, Artificial intelligence applications in the intensive care unit, Crit. Care Med. 29 (2001) 427–435.
[11] D. Wolpert, The supervised learning no-free-lunch theorems, in: Soft Computing and Industry, 2001, pp. 25–42.
[12] A. Johnson, T. Pollard, L. Shen, L.-w. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Celi, R. Mark, MIMIC-III, a freely accessible critical care database, Sci. Data 3 (2016) 160035.
[13] W.A. Knaus, J.E. Zimmerman, D.P. Wagner, E.A. Draper, D.E. Lawrence, APACHE-acute physiology and chronic health evaluation: A physiologically based classification system, Crit. Care Med. 9 (1981) 591–597.
[14] W. Knaus, E. Draper, D. Wagner, J. Zimmerman, APACHE II: A severity of disease classification system, Crit. Care Med. 13 (1985) 818–829.
[15] R. Dybowski, P. Weller, R. Chang, V. Gant, Prediction of outcome in critically ill patients using artificial neural network synthesised by genetic algorithm, Lancet 347 (1996) 1146–1150.
[16] L. Wong, J. Young, A comparison of ICU mortality prediction using the APACHE II scoring system and artificial neural network, Anaesthesia 54 (1999) 1048–1054.
[17] G. Clermont, D. Angus, S. DiRusso, M. Griffin, W. Linde-Zwirble, Predicting hospital mortality for patients in the intensive care unit: A comparison of artificial neural networks with logistic regression models, Crit. Care Med. 29 (2001) 291–296.
[18] V. Ribas, J. Caballero, A. Ruiz-Sanmartin, J. Ruiz-Rodríguez, J. Rello, A. Wojdel, A. Vellido, Severe sepsis mortality prediction with relevance vector machines, in: IEEE Engineering in Medicine and Biology Society Conference, 2011, pp. 100–103.
[19] S. Kim, W. Kim, R.W. Park, A comparison of intensive care unit mortality prediction models through the use of data mining techniques, Healthc. Inform. Res. 17 (2011) 232–243.
[20] G.S. Krishnan, S. K. S, A novel GA-ELM model for patient-specific mortality prediction over large-scale lab event data, Appl. Soft Comput. 80 (2019) 525–533.
[21] A. Awad, M. Bader-El-Den, J. McNicholas, Patient length of stay and mortality prediction: A survey, Health Serv. Manag. Res. 30 (2017) 105–120.
[22] O. Sagi, L. Rokach, Ensemble learning: A survey, Wiley Interdiscip. Rev. Data Min. Knowl. Discov. 8 (2018) e1249.
[23] L. Breiman, Bagging predictors, Mach. Learn. 24 (1996) 123–140.
[24] T. Ho, The random subspace method for constructing decision forests, IEEE Trans. Pattern Anal. Mach. Intell. 20 (1998) 832–844.
[25] L. Breiman, Random forests, Mach. Learn. 45 (2001) 5–32.
[26] K.M. Ting, J.R. Wells, S.C. Tan, S.W. Teng, G.I. Webb, Feature-subspace aggregating: Ensembles for stable and unstable learners, Mach. Learn. 82 (2011) 375–397.
[27] A. Rahman, B. Verma, Cluster-based ensemble of classifiers, Expert Syst. 30 (2012) 270–282.
[28] F. Jiang, J. He, T. Tian, A clustering-based ensemble approach with improved pigeon-inspired optimization and extreme learning machine for air quality prediction, Appl. Soft Comput. 85 (2019) 105827.
[29] J. Meng, H. Hao, Y. Luan, Classifier ensemble selection based on affinity propagation clustering, J. Biomed. Inform. 60 (2016) 234–242.
[30] M. Haque, N. Noman, R. Berretta, P. Moscato, Optimising weights for heterogeneous ensemble of classifiers with differential evolution, in: IEEE Congress on Evolutionary Computation, 2016, pp. 233–240.
[31] H. Wang, B. Zheng, S.W. Yoon, H. Ko, A support vector machine-based ensemble algorithm for breast cancer diagnosis, European J. Oper. Res. 267 (2018) 687–699.
[32] R. Pirracchio, M.L. Petersen, M. Carone, M.R. Rigon, S. Chevret, M. J, Mortality prediction in intensive care units with the super ICU learner algorithm (SICULA): A population-based study, Lancet Respir. Med. 3 (2015) 42–52.
[33] B. Kwon, Young Suk, M. Seong, Development and validation of a quick sepsis-related organ failure assessment-based machine-learning model for mortality prediction in patients with suspected infection in the emergency department, J. Clin. Med. 9 (2020) 875.
[34] N. El-Rashidy, S. El-Sappagh, T. Abuhmed, S. Abdelrazek, H.M. El-Bakry, Intensive care unit mortality prediction: An improved patient-specific stacking ensemble model, IEEE Access 8 (2020) 133541–133564.
[35] R. Ghorbani, R. Ghousi, A. Makui, A. Atashi, A new hybrid predictive model to predict the early mortality risk in intensive care units on a highly imbalanced dataset, IEEE Access 8 (2020) 141066–141079.
[36] R.M. Cruz, R. Sabourin, G.D. Cavalcanti, Tsang, META-DES: A dynamic ensemble selection framework using meta-learning, Pattern Recognit. 48 (2015) 1925–1935.
[37] K. Woods, W.K. Bowyer, Combination of multiple classifiers using local accuracy estimates, IEEE Trans. Pattern Anal. Mach. Intell. 19 (1997) 405–410.
[38] A.H. Ko, R. Sabourin, A.S. Britto Jr., From dynamic classifier selection to dynamic ensemble selection, Pattern Recognit. 41 (2008) 1718–1731.
[39] P. Cavalin, R. Sabourin, C. Suen, Dynamic selection approaches for multiple classifier systems, Neural Comput. Appl. 22 (2013) 673–688.
[40] S. García, Z.-L. Zhang, A. Altalhi, S. Alshomrani, F. Herrera, Dynamic ensemble selection for multi-class imbalanced datasets, Inform. Sci. 445–446 (2018) 22–37.
[41] A.E.W. Johnson, T.J. Pollard, R.G. Mark, Reproducibility in critical care: A mortality prediction case study, in: Proceedings of the 2nd Machine Learning for Healthcare Conference, vol. 68, 2017, pp. 361–376.
[42] K. Caballero Barajas, R. Akella, Dynamically modeling patient's health state from electronic medical records: A time series approach, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015, pp. 69–78.
[43] J. Calvert, Q. Mao, A. Rogers, C. Barton, M. Jay, T. Desautels, H. Mohamadlou, J. Jan, R. Das, A computational approach to mortality prediction of alcohol use disorder inpatients, Comput. Biol. Med. 75 (2016) 74–79.
[44] H. Harutyunyan, H. Khachatrian, D. Kale, A. Galstyan, Multitask learning and benchmarking with clinical time series data, Sci. Data 6 (2017) 96–113.
[45] G.S. Krishnan, S. Sowmya Kamath, A supervised approach for patient-specific ICU mortality prediction using feature modeling, in: Frontier Computing, 2019, pp. 287–295.
[46] C. Guo, M. Lu, J. Chen, An evaluation of time series summary statistics as features for clinical prediction tasks, BMC Med. Inform. Decis. Mak. 20 (2020) 48.
[47] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.
[48] Y. Freund, R. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, J. Comput. System Sci. 55 (1997) 119–139.