ESKİŞEHİR TECHNICAL UNIVERSITY JOURNAL OF SCIENCE AND TECHNOLOGY
A- APPLIED SCIENCES AND ENGINEERING

2022, 23(1), pp. 87-108, DOI:10.18038/estubtda.906920

RESEARCH ARTICLE

COMPARATIVE OF SUCCESS OF KNN WITH NEW PROPOSED K-SPLIT METHOD AND STRATIFIED CROSS VALIDATION ON REMOTE HOMOLOGUE PROTEIN DETECTION

Fahriye GEMCI [1 *], Turgay IBRIKCI [2], Ulus CEVIK [3]

[1] Department of Computer Engineering, Faculty of Engineering&Architecture, Kahramanmaras Sutcu Imam University, Kahramanmaraş, Turkey
[2] Department of Software Engineering, Faculty of Engineering, Adana Alparslan Turkes Science and Technology University, Adana, Turkey
[3] Department of Electrical-Electronics Engineering, Faculty of Engineering, Cukurova Unversity, Adana, Turkey

ABSTRACT

In this study, a remote homologous protein detection problem, which is a problem related to the field of bioinformatics and has made a great contribution in the field of medicine, is discussed. Protein sequences taken from the SCOP database, which is an important and widely used database for proteins, were tested for remote homologue protein detection in this study. Feature vectors were obtained from the protein sequences using the bag-of-words model. These obtained feature vectors were classified using the k-nearest Neighbor classifier algorithm. In this classification, the different distances used were Bray Curtis, Euclidean, Minkowski, Dice, Jaccard, Chebyshev, Cosine, SokalSneath, correlation, matching coefficient, RogersTanimoto, SokalMichener, Canbera, Hamming, Kulczynski, and RussellRao on the k-nearest Neighbor classifier for remote homologue protein detection. Two different new methods is proposed for preventing the imbalanced data problem. The first of these is special k-fold value and the other is novel k-split method. It is observed that the k-nearest Neighbor algorithm with the Bray Curtis distance and cross validation with special k-fold value and novel k-split method show the most successful performance, with 98.9% and 83.8% accuracy and 77% and 92% ROC score, respectively.

Keywords: Remote Homologue Protein, k-nearest Neighbor, Bag-of-words model, Distances, k-fold Stratified Cross Validation

1. INTRODUCTION

Biomedicine is academic research topic that uses biology and engineering methods for solving problems in medicine [1, 2]. Bioinformatics is a promising field to analyze and extract valuable knowledge from biomedical information. Protein remote homology detection, studied in order to discover new undiscovered protein structures, is an important topic in bioinformatics, because the discovery of the similarity and relationship of unknown proteins with each other will facilitate the discovery of unknown protein structures [3]. The discovery of new protein structures contributes to the diagnosis of diseases and the discovery of new drugs in medicine. Hence, bioinformatics and biomedicine are regarded as two areas of study that support each other.

The bioinformatics problem is still one of the difficult problems to solve. The first known methods for protein remote homology detection are methods based on pairwise sequence comparison such as Smith–Waterman local alignment algorithm. However, sequence alignment-based methods show low success due to low sequence similarity of remote homologs. Subsequently, generative models testing on protein families were constructed such as hidden markow model [4].

After generative methods, the discriminative methods have been proposed to consider protein family differences such as Support Vector Machine (SVM) for protein remote homology. SVM methods detect remote homology by generating kernels based on the properties obtained from protein sequences [4]. SVM-Ngram, SVM with Top Ngram, SVM-Ngram-p1 and SVM-Ngram-KTA performed remote

*Corresponding Author: fahriyegemci@ksu.edu.tr
Received: 12.07.2021 Published: 30.03.2022

homologous protein detection using SCOP 1.53 Dataset with 81,2 % ROC score, 71,72 % ROC score, 88,7 % ROC score and 89,2 % ROC score, respectively [5]. More than 90% roc score was obtained in this study, including Soft bag-of-words (Soft BoW) and Soft PLSA [5], for the remote homology problem. In these studies, it has been observed that the n-gram method is a useful feature extraction method on remote homologous protein detection.

The studies of remote homologue protein detection involve several important challenges. The most critical of them is to determine different lengths of amino acids from protein sequences. Hence an important step in remote homology problems is to convert amino acid sequences of proteins into fixed-length datasets. Hence, the bag-of-words (BoW) model is used to convert protein sequences into fixed-length feature vectors.

One area of vital importance in protein classification is sequence mining [5]. Since the sequential information of the data is lost when sequential data are converted to non-sequential data, the traditional classification fails for classification of the converted non-sequential data, although the traditional classification is normally successful for classification of non-sequential data [5]. It was observed whether there was any effect on the classification success of vitally important information of subsequences [5]. Because of the importance of subsequence information in that study [6], n-grams of protein sequences were used as a feature vector in the present study.

In the present study, first of all, the SCOP database from which the protein sequences were obtained, the BoW model used to obtain the feature vectors, four different distances to measure the distance between protein samples, and the k-nearest Neighbor (kNN) classification method used to determine remote homologue proteins are introduced. Then the effectiveness of kNN with 16 different distances, namely Bray Curtis, Euclidean, Minkowski, Dice, Jaccard, Chebyshev, Cosine, SokalSneath, Correlation, Matching coefficient, RogersTanimoto, SokalMichener, Canbera, Hamming, Kulczynski, and RussellRao distances, for remote homologue protein detection is compared in the present study.

In [4], the kNN method based on pairwise sequence similarity scores was used for remote homology detection. The kNN-pairwise results obtained when the k value were 3 and the Euclidean distance is taken lower success than the results obtained with the SVM algorithms [4]. Contrary to [4], in this study for protein remote homology, successful results were obtained with the bray-curtis distance measurement and kNN method by using some methods, such as n-gram to extract features and stratified cross validation with novel k-split formula and novel k-split method to solve imbalanced problem.

2. MATERIALS AND METHODS

2.1. Protein Classification

One of the important problems in bioinformatics is to classify proteins according to their amino acid sequences. The protein structure is more conserved than the sequence. Hence, discovering protein sequence similarities is helpful for predicting protein functions. Remote homologues can also be defined as thin sequence similarities [6]. Commonly used methods for the remote homologue problem can be classified into 3 types: similarity-search, structure-based alignment, and supervised classification methods [7].

2.2. Dataset

The SCOP database is known as the gold standard protein database [8]. The SCOP 1.53 protein dataset is used frequently in remote homologue protein detection [8]. Hence, protein sequences from SCOP 1.53 are used to test the method. The SCOP 1.53 dataset is composed of 54 families and 23 superfamilies.

2.3. BoW Model

BoW model is a method frequently used to convert feature vectors of the same length on texts of different lengths consisting of the same alphabet in natural language processing. The text in the model

can be a sentence or a document. When they are documents, n-grams consist of words; n-grams are made up of characters when they are sentences. In the protein similarity study, protein sequences are sentences, so n-grams consist of amino acid characters. For texts consisting of sentences, a bag consisting of the characters of all sentence samples is created. A feature vector for each sample is obtained depending on the frequency of occurrence of the characters in the bag passed in each sentence sample [9].

BoW model has been utilized in many protein studies in the field of bioinformatics [10, 11]. BoW for proteins is usually created from character level tokens on protein sequences. Normally proteins are based on 20 alphabets. But the feature extraction with the BoW model of proteins can also be built on the alphabet for the purpose of reduced size reduction, such as 2-state and 3-state.

2.4. kNN Algorithm

Success of the kNN algorithm for two-class classification problems has been observed in previous studies [6,12]. Although kNN is simple, it is one of the most widely used classifiers because it rivals the most complex classifiers in performance [13]. kNN was introduced by Fix, E. and Hodges, J.L. in 1951 [14]. The kNN algorithm is widely used in many areas such as data mining and pattern classification problems, which are still being developed today [13].

kNN is based on a lazy learning algorithm that is a non-parametric classification algorithm. kNN accepts that each instance matches a point in an n-dimensional space. The kNN algorithm depends on feature similarity. The kNN algorithm calculates the distances of a new instance to all instances. In the kNN algorithm, neighbors are instances with k closest distances of a new instance. The new instance is assigned to a majority class of classes of its neighbors. kNN calculates distances using a distance function such as Euclidean, Cosine, Jaccard, and Minkowski distances [14]. In the present study, a remote homologous protein was used for detection. These 16 different methods are explained and introduced with their formulas in the following section, 2.4.1.

2.4.1. Distance/similarity measures on kNN

The distance between classes/clusters is also an important parameter to measure the similarity rate of this pair dataset. Numerous algorithms are available to calculate distances between pairs data. Thus, the greater the distance between the data pairs, the lower the similarity between the two data pairs. The smaller the distance between two data pairs, the greater the similarity between the two data pairs.

In the present study, the selected distance algorithms were used by kNN. In the distance measure algorithms used, it is generally supposed that X and Y are two vectors in n-dimensional space for the calculation.

It is supposed that A is a binary value of where both samples have the value 1. It is supposed that B is a binary value of where the first sample has the value 1 and the other has the value 0. It is supposed that C is a binary value of where the first sample has the value 0 and the other has the value 1. It is supposed that D is a binary value of where both samples have the value 0. Based on these assumptions of A, B, C, D; Matching coefficient, RogersTanimoto distance, RussellRao distance, SokalMichener distance, SokalSneath distance formulas are given in the relevant section.

2.4.1.1. *Bray Curtis distance*

Bray and Curtis created the Bray Curtis distance, which is also called the Sorensen distance, using the city-block metric in 1957. [16]. The Bray Curtis distance takes a value between 0 and 1. As this similarity approaches 0, it shows two more similar samples; when closer to 1, it shows two less similar samples. The Bray Curtis distance is

$$Bra\_d_{=} \sum_{i=1}^{n} \frac{|x_i - y_i|}{(x_i + y_i)} \quad (1)$$

2.4.1.2. *Chebyshev distance*

The Chebyshev distance is a distance measurement, also recognized as a maximum metric, developed from the Minkowski distance [16]. The Chebyshev distance is

$$Che\_D = \max_i(x_i - y_i) \quad (2)$$

2.4.1.3. *Cosine similarity*

Cosine similarity is a vector-based similarity measure popular used in natural language processing problems [6,18,19]. Cosine similarity calculates the similarity by measuring the cosine angle between two vectors. The cosine angle is the difference between two vectors directions, irrespective of the vector's size. Cosine similarity is

$$C\_S = \frac{X.Y}{\sqrt[2]{|X||Y|}} = \frac{\sum_{i=1}^{n} X_i Y_i}{\sqrt[2]{\sum_{i=1}^{n} X_i^2 \sum_{i=1}^{n} Y_i^2}} \quad (3)$$

2.4.1.4. *Dice distance*

The dice distance is derived from dice similarity. It is calculated by looking at common occurrences rather than incompatibility, while comparing by looking at whether the samples are the same for all sizes [20]. The dice distance is

$$D\_D = 1 - \frac{2\sum_{i=1}^{n} X_i Y_i}{\sum_{i=1}^{n} X_i^2 + \sum_{i=1}^{n} Y_i^2} \quad (4)$$

2.4.1.5. *Euclidean distance*

The Euclidean distance, which is a method frequently used in vector spaces, calculates the vector distance by taking the square root of the sum of the values in their compatible dimensions of two vectors, as given in Equation 5 [6].

$$E\_D = \sqrt{\sum_{i=1}^{n} (X_i - Y_i)^2} \quad (5)$$

2.4.1.6. *Hamming distance*

Richard Hamming developed the Hamming distance in 1950 [21]. It is the number of distinct elements of two samples of the same length. The Hamming distance is

$$H\_D = \sum_{i=1}^{n} |X_i - Y_i| \quad (6)$$

2.4.1.7. *Jaccard distance*

Jaccard distance between two vectors X and Y is given in Equation 7. Jaccard distance is calculated by subtracting the Jaccard index from 100% to find how similar two vectors are [22].

$$J\_D = \frac{\sum_{i=1}^{n} (X_i - Y_i)^2}{\sum_{i=1}^{n} X_i^2 + \sum_{i=1}^{n} Y_i^2 - \sum_{i=1}^{n} X_i Y_i} \quad (7)$$

2.4.1.8. *Kulczynski distance*

The Kulczynski distance is also called quantitative symmetric dissimilarity (QSK). QSK gives different results from standardized distances. However, after such relativization it gives the same result as Sorensen and city-block distances. The Kulczynski distance is [23]

$$K\_D = \frac{\sum_{i=1}^{n} |X_i - Y_i|}{\sum_{i=1}^{n} \min(X_i, Y_i)} \quad (8)$$

2.4.1.9. *Matching coefficient*

While the lower boundary of the distance is 0, its upper boundary is not. [24]. According to these suppositions, the matching coefficient between two samples is

$$M\_D = \frac{A+D}{A+B+C+D} \tag{9}$$

2.4.1.10. *Minkowski distance*

The Minkowski distance between two vectors X and Y is given in Equation 10. The Minkowski distance can be thought of as the generalized version of the Euclidean distance. Because of the Minkowski distance given in Equation 9, the formula for $p = 2$ gives the Euclidean distance [17].

$$Mink\_D = \sqrt[p]{\sum_{i=1}^{n} |X_i - Y_i|^p} \tag{10}$$

2.4.1.11. *RogersTanimoto distance*

Rogers and Tanimoto developed the Rogers and Tanimoto distance in 1960 [24]. According to these suppositions, the RogersTanimoto distance between two samples is

$$RT\_D = \frac{A+D}{A+2*(B+C)+D} \tag{11}$$

2.4.1.12. *RussellRao distance*

Russell and Rao developed RussellRao distance in 1940. The distance gives results between 0 and 1 [25]. According to these suppositions, the RussellRao distance between two samples is

$$RR\_D = \frac{A}{A+B+C+D} \tag{12}$$

2.4.1.13. *SokalMichener distance*

Sokal and Michener developed the SokalMichener distance, which is a weighted mean pair method, in 1958. The SokalMichener distance is calculated using product moment correlations between sample pairs [26]. According to these suppositions, the SokalMichener distance is

$$SM\_D = \frac{A+D}{A+B+C+D} \tag{13}$$

2.4.1.14. *Canberra distance*

A frequently used Euclidean distance, while giving importance to the features where the differences in the samples are high, the Canberra distance is designed to use the distance feature as a measure of difference. The Canberra distance between two vectors X and Y is given in Equation 14 [27].

$$Canb\_d = \sum_{i=1}^{n} \frac{|x_i - y_i|}{|x_i| + |y_i|} \tag{14}$$

2.4.1.15. *SokalSneath distance*

Sokal and Sneath created the SokalSneath distance in 1963. According to these suppositions, the SokalSneath distance between two samples is given in Equation 15 [28].

$$SS\_D = \frac{A+D}{(B+C)} \tag{15}$$

2.4.1.16. *Correlation similarity*

Correlation similarity takes a value between -1 and 1. Correlation similarity between two vectors X and Y is given in Equation 16 [29].

$$Corr\_S = \frac{x^T y}{||X||| Y||} \quad (16)$$

2.4.2. K-Fold Cross validation

Cross validation (CV) is a widely used method performed by separating the test and train datasets in order to increase the performance of the machine learning model on the samples. CV is used to preclude overfitting.

k-fold cross validation (k-fold CV) method divides the data into k subsets of approximately the same length. A k-fold is thus created from sub-datasets of approximately the same length. k-fold CV performs the method by using the data in each fold as separate test data [30]. The average of k performance measurements in k-fold gives the performance of the method with k-fold CV.

2.4.3. K-Fold Stratified Cross Validation

Stratified cross validation is an extension of CV to be selected so that the average response value is the same in all folds [31]. In k-fold stratified cross validation (k-fold stratified CV), each class is distributed evenly across the k-fold. In other words; a dataset is not randomly distributed into k-fold, but in a way that does not disturb the sample distribution ratios in the classes in k-fold stratified cross-validation. In this method, balanced partitioning is performed in each fold, while in the normal CV method, class distribution rates are not taken into account. Therefore, k-fold stratified CV provides a more reliable estimate of accuracy than the normal k-fold CV method [32].

3. EXPERIMENTAL RESULTS

This study was carried out using Python version 3.8. The Biopython tool was used to access the protein sequence and information as desired. Firstly, in the study, protein sequences were obtained from the SCOP 1.53 database. In this study, samples from the same family in the SCOP 1.53 dataset are separated as positive test samples. Samples from the same superfamily and different families in the dataset are separated as positive training samples. Negative samples are taken from different folds with positive samples. Negative samples are divided into train and test sets at the same rate as positive samples. Then the BoW model was created for feature extraction of protein sequences. Words in the BoW model are n-grams of protein sequences, when the bag was composed of the n-grams. N-grams were between 2 and 4 grams of protein sequences. The maximum number of features in the study was limited to 20000 features. The highest frequency 20000 features (n-grams) were selected and used in the study.

After that obtaining feature matrix, the kNN classification algorithm was used to detect remote homologue proteins. The k neighbors number of kNN was taken as values from 2 to 5. As the number of k increases from 3 to 5, the success of the algorithm decreases, so higher k values was not tried. Since the most successful results were obtained when the k value was 3, the results with the k value of 3 have given in the present study. kNN with 16 different distances were tested for the problem.

Three different tests in the present study were performed for remote homologue detection using kNN. The first of these was the test process without using CV; the second one was the field testing process using CV. In the second, the k-fold stratified CV method was used to solve the imbalanced class problem brought about by the remote homology problem in the protein dataset. k-fold stratified CV method was implemented by selecting a special number k. This k number was calculated one by one with a special formula for each family. The special formula for the binary class problem was given in Equation 17. classLow represents the class with the lowest number of samples. The purpose of this formula is to provide convenience by automating the varying k-numbers in k-fold stratified CV method.

$$k\ number = samples\ number\ of\ the\ classLow \quad (17)$$

Third one was the novel k-split method. This new method was proposed to solve the imbalanced class problem for remote homology problem. The k-split method is that designed based on the class with low sample number. Samples were taken from the high sample number class and tested as much as the sample number in the low sample class. This process was repeated until there were no untested samples in the high sample class. The average results of these tests were also calculated as the success of this test.

$$test\ number\ = \frac{sample\ number\ on\ high\ sample\ number\ class}{sample\ number\ on\ low\ sample\ number\ class} + 1 \quad (18)$$

The accuracy results of kNN with 16 distances for remote homology were given in Table 1. The mean accuracy results were calculated by averaging the accuracy of all families in the SCOP 1.53 dataset. Lowest and highest accuracy show the lowest accuracy and highest accuracy among all 54 family accuracy values, respectively. While Table 1 gave accuracy results without k-fold stratified CV, Table 2 gave accuracy results with stratified cross validation with special k value fold. Table 3 gave accuracy results with novel k-split method.

While the accuracy results with 16 different distances without CV are between 94% and 99%, the accuracy results with CV are between 95% and 100%. The accuracy results with novel k-split method are between 61% and 83%. The increase in success was only between 1% and 2% according to the accuracy results in Table 1 and Table 2. Although Table 3 accuracy results gave lower results than the previous two tests, success cannot be measured by accuracy alone.

Table 1. Accuracy results of kNN with four distances for remote homology without CV

Distance/Similarity Methods	Lowest Accuracy	Highest Accuracy	Mean Accuracy
Bray Curtis	0.95464	0.99586	0.98758
Euclidean	0.95959	0.99572	0.98715
Minkowski	0.95959	0.99572	0.98715
Dice	0.95364	0.99609	0.98736
Jaccard	0.95364	0.99609	0.98736
Chebyshev	0.95687	0.99655	0.99655
Cosine	0.94224	0.99609	0.98714
SokalSneath	0.95364	0.99609	0.98736
Correlation	0.94199	0.99609	0.98712
Matching	0.95945	0.98729	0.99609
Rogers Tanimoto	0.95945	0.99609	0.98729
Sokal Michener	0.95945	0.99609	0.98729
Canberra	0.95918	0.99609	0.98718
Hamming	0.94851	0.99609	0.98708
Kulczynski	0.95860	0.99609	0.98843
RussellRao	0.96530	0.99609	0.98867

Table 2. Accuracy results of kNN with four distances for remote homology with StratifiedKFold cross validation

Distance/Similarity Methods	Lowest Accuracy	Highest Accuracy	Mean Accuracy
Bray Curtis	0.97078	0.99901	0.98968
Euclidean	0.96327	0.99951	0.98866
Minkowski	0.96327	0.99951	0.98866
Dice	0.97186	1.0	0.99039
Jaccard	0.97186	1.0	0.99039
Chebyshev	0.96735	0.99901	0.98825
Cosine	0.96717	1.0	0.98882
SokalSneath	0.97186	1.0	0.99039
Correlation	0.96717	1.0	0.98882
Matching	0.97173	1.0	0.98935
Rogers Tanimoto	0.97173	1.0	0.98935
Sokal Michener	0.97173	1.0	0.98935
Canberra	0.97000	1.0	0.98893
Hamming	0.94127	0.99803	0.98875
Kulczynski	0.96789	0.99951	0.99030
RussellRao	0.97173	0.99951	0.99025

Table 3. Accuracy results of kNN with four distances for remote homology with k-split method

Distance/Similarity Methods	Lowest Accuracy	Highest Accuracy	Mean Accuracy
Bray Curtis	0.72522	0.96145	0.83880
Euclidean	0.69421	0.93852	0.83183
Minkowski	0.71280	0.93852	0.83183
Dice	0.57604	0.95946	0,78301
Jaccard	0.57604	0.95946	0,78301
Chebyshev	0.72117	0.90193	0.81497
Cosine	0.58537	0.96844	0.77055
SokalSneath	0.57604	0.95946	0,78301
Correlation	0.59826	0.97072	0.77398
Matching	0.52541	0.94213	0.70663
Rogers Tanimoto	0.52541	0.89203	0.70511
Sokal Michener	0.52541	0.89203	0.70511
Canberra	0.51736	0.90710	0,68537
Hamming	0.48890	0.74993	0.57411
Kulczynski	0.51915	0.97162	0.63731
RussellRao	0.52071	0.96757	0.61409

Although everything looks normal without cross validation in Table 1, it was observed that the class with a low number of samples was mostly misclassified in the results of the confusion matrix in Table 4. It was understood that the reason for the misclassification here is imbalanced data. To solve this problem, stratified cross validation method with a special k value fold was proposed. While in Table 4, confusion matrix results in different families obtained without cross validation were given, confusion matrix results in different families obtained with the newly proposed method were given in Table 5. From Table 5, it was observed that the classification using the Bray Curtis distance showed the most successful classification. Then it was observed that the Euclidean distance gave the second most successful performance.

According to the 2.1.1.2 protein family result given in Table 5, using the Matching, Rogers Tanimoto, Hamming, Sokal Michener, Canberra, RussellRao, and Kulczynski distances, 22 samples, which were the samples of the remote homologue class in the 2.1.1.2 protein family which is the 2nd class, were misclassified. It was seen that the remote homology detection performed on the other hand, using the Bray Curtis, Euclidean, Minkowski, Dice, Jaccard, Chebyshev, Cosine and SokalSneath distance, gives better results in the class with a small number of samples, in Table 5. Additionally, it was observed that, using the Bray Curtis distance, 21 of 22 samples of the family were classified correctly and only 1 sample was misclassified, resulting in the best performance. However, it was observed that 758 proteins belonging to the 1st class, which is a non-remote homologous class in the 2.1.1.2 protein family, perform with over 95% accuracy in all different distances. Table 6, Table 7, Table8, Table 9 and Table 10 have shown Confusion matrixes of kNN with distances for remote homology with k-split method for 1.4.1.1 family, 2.1.1.2 family, 2.28.1.1 family, 3.42.1.5 family and 7.3.10.1 family, respectively.

According to depending on all families, it can be said that 99% of the samples belonging to the non-remote homologue class, which was the class with the highest number of data, were classified correctly, while a success between 94% and 100% was observed for the remote homologue class for the Bray Curtis distance. Among these 16 distance methods, they shared the 2nd most successful distance method in remote homology problem, Euclidean and Minkowski distance together. Precision values, Recall values and ROC scores for kNN with distances for remote homology with stratified k-fold cross validation with 16 different distances were given in Table 11, Table 13 and Table 15, respectively. Precision values, Recall values and ROC scores for kNN with distances for remote homology with k-split method with 16 different distances were given in Table 12, Table 14 and Table 16, respectively. Table 15 and Table 16 including ROC scores supports the success of Bray Curtis, Euclidean and Minkowski distances in remote homology.

Table 17 and Table 18 were given mean ROC scores for two proposed method for remote homology. According to Table 2 accuracy results and Table 4 confusion matrix values, it is observed that kNN with StratifiedKFold cross validation with special k-fold value is quite successful and sufficient. However, when the table 17 average ROC results were obtained, it was observed that the roc values needed to be

improved. For this reason, a new method, the k-split method, was proposed. In his last, the average ROC results given in Table 18 are quite promising. It is expected that it will be useful not only for the remote homology problem, but also for other imbalanced data problems. It has been observed that it is a very easy and successful method to implement. According to the accuracy results, kNN homology with StratifiedKFold cross validation with new formula is more successful, while looking at the ROC results, the novel k-plit method is observed to be more successful. It can be said that both methods are useful tools for the protein remote homolog problem.

Table 4. Confusion matrixes of kNN with distances for remote homology without StratifiedKFold cross validation

Distance/Similarity Methods	**1.4.1.1 family**	**2.1.1.2 family**	**2.28.1.1 family**	**3.42.1.5 family**	**7.3.10.1 Family**
Bray Curtis	[1994 0] [22 1]	[753 5] [14 8]	[3039 5] [44 0]	[1436 1] [13 0]	[3625 28] [44 51]
Euclidean	[1990 4] [10 13]	[750 8] [14 8]	[3040 4] [44 0]	[1435 2] [13 0]	[3618 35] [90 5]
Minkowski	[1990 4] [10 13]	[753 5] [5 17]	[3040 4] [44 0]	[1435 2] [13 0]	[3618 35] [90 5]
Dice	[1993 1] [23 0]	[757 1] [22 0]	[3041 3] [44 0]	[1437 0] [13 0]	[3641 12] [86 9]
Jaccard	[1993 1] [23 0]	[757 1] [22 0]	[3041 3] [44 0]	[1437 0] [13 0]	[3641 12] [86 9]
Chebyshev	[1990 4] [11 12]	[751 7] [18 4]	[3037 7] [44 0]	[1433 4] [13 0]	[3626 27] [87 8]
Cosine	[1994 0] [23 0]	[757 1] [22 0]	[3034 10] [44 0]	[1437 0] [13 0]	[3644 9] [67 28]
SokalSneath	[1993 1] [23 0]	[757 1] [22 0]	[3041 3] [44 0]	[1437 0] [13 0]	[3641 12] [86 9]
Correlation	[1994 0] [23 0]	[756 2] [22 0]	[3034 10] [44 0]	[1437 0] [13 0]	[3642 11] [67 28]
Matching	[1993 1] [23 0]	[757 1] [22 0]	[3043 1] [44 0]	[1436 1] [13 0]	[3590 63] [89 6]
Rogers Tanimoto	[1993 1] [23 0]	[758 0] [22 0]	[3043 1] [44 0]	[1436 1] [13 0]	[[3590 63] [89 6]
Sokal Michener	[1993 1] [23 0]	[757 1] [22 0]	[3043 1] [44 0]	[1436 1] [13 0]	[3590 63] [89 6]
Canberra	[1992 2] [23 0]	[757 1] [22 0]	[3043 1] [44 0]	[1436 1] [13 0]	[3589 64] [89 6]
Hamming	[1991 3] [23 0]	[758 0] [22 0]	[3044 0] [44 0]	[1437 0] [13 0]	[3552 101] [92 3]
Kulczynski	[1994 0] [23 0]	[758 0] [22 0]	[3043 1] [44 0]	[1437 0] [13 0]	[3653 0] [95 0]
RussellRao	[1994 0] [23 0]	[758 0] [22 0]	[3044 0] [44 0]	[1437 0] [13 0]	[3653 0] [95 0]

Table 5. Confusion matrixes of kNN with distances for remote homology with StratifiedKFold cross validation

Distance/Similarity Methods	**1.4.1.1 family**	**2.1.1.2 family**	**2.28.1.1 family**	**3.42.1.5 family**	**7.3.10.1 family**
Bray Curtis	[1981 13] [0 23]	[753 5] [1 21]	[3033 11] [2 42]	[1429 8] [2 11]	[3623 30] [5 90]
Euclidean	[1979 15] [1 22]	[753 5] [5 17]	[3028 16] [4 40]	[1425 12] [1 12]	[3610 43] [8 87]
Minkowski	[1979 15] [1 22]	[753 5] [5 17]	[3028 16] [4 40]	[1425 12] [1 12]	[3610 43] [8 87]
Dice	[1991 3] [3 20]	[755 3] [14 8]	[3043 1] [5 39]	[1437 0] [6 7]	[3631 22] [6 89]
Jaccard	[1991 3] [3 20]	[755 3] [14 8]	[3043 1] [5 39]	[1437 0] [6 7]	[3631 22] [6 89]
Chebyshev	[1985 9] [5 18]	[750 8] [13 9]	[3029 15] [5 39]	[1426 11] [6 7]	[3590 63] [11 84]
Cosine	[1993 1] [4 19]	[757 1] [17 5]	[3042 2] [4 40]	[1437 0] [13 0]	[3645 8] [17 78]
SokalSneath	[1991 3] [3 20]	[755 3] [14 8]	[3043 1] [5 39]	[1437 0] [6 7]	[3631 22] [6 89]
Correlation	[1993 1] [3 20]	[757 1] [17 5]	[3042 2] [4 40]	[1437 0] [13 0]	[3643 10] [17 78]
Matching	[1993 1] [6 17]	[758 0] [22 0]	[3024 20] [3 41]	[1436 1] [13 0]	[3634 19] [19 76]
Rogers Tanimoto	[1993 1] [6 17]	[758 0] [22 0]	[3024 20] [3 41]	[1436 1] [11 2]	[[3634 19] [19 76]
Sokal Michener	[1993 1] [6 17]	[758 0] [22 0]	[3024 20] [3 41]	[1436 1] [11 2]	[3634 19] [19 76]
Canberra	[1989 5] [6 17]	[758 0] [22 0]	[3025 19] [4 40]	[1435 2] [10 3]	[3637 16] [20 75]
Hamming	[1994 0] [10 13]	[758 0] [22 0]	[3030 14] [5 39]	[1436 1] [13 0]	[3639 14] [20 75]
Kulczynski	[1994 0] [14 9]	[758 0] [22 0]	[3044 0] [5 39]	[1437 0] [13 0]	[3653 0] [21 74]
RussellRao	[1994 0] [18 5]	[758 0] [22 0]	[3044 0] [7 37]	[1437 0] [13 0]	[3653 0] [22 73]

Table 6. Confusion matrixes of kNN with distances for remote homology on 1.4.1.1 family with k-split method

1.4.1.1 family Distance/Similarity Methods	**1. split**	**2. split**	**3. split**	**…**	**k. split**
Bray Curtis	[24 1] [9 16]	[12 13] [6 19]	[20 5] [11 14]	…	[11 14] [3 22]
Euclidean	[23 2] [4 21]	[7 18] [3 22]	[18 7] [4 21]	…	[9 16] [4 21]
Minkowski	[23 2] [4 21]	[7 18] [3 22]	[18 7] [4 21]	…	[9 16] [4 21]
Dice	[25 0] [4 21]	[25 0] [5 20]	[22 3] [13 12]	…	[12 13] [3 22]
Jaccard	[22 3] [13 12]	[25 0] [5 20]	[22 3] [13 12]	…	[12 13] [3 22]
Chebyshev	[19 6] [3 22]	[5 20] [3 22]	[14 11] [3 22]	…	[7 18] [1 24]
Cosine	[24 1] [5 20]	[21 4] [21 4]	[22 3] [13 12]	…	[24 1] [4 21]
SokalSneath	[25 0] [4 21]	[25 0] [5 20]	[22 3] [13 12]	…	[12 13] [3 22]
Correlation	[24 1] [5 20]	[21 4] [19 6]	[20 5] [12 13]	…	[24 1] [3 22]
Matching	[11 14] [0 25]	[16 9] [0 25]	[19 6] [12 13]	…	[24 1] [16 9]
Rogers Tanimoto	[11 14] [0 25]	[16 9] [0 25]	[19 6] [12 13]	…	[24 1] [16 9]
Sokal Michener	[11 14] [0 25]	[16 9] [0 25]	[19 6] [12 13]	…	[24 1] [16 9]
Canberra	[21 4] [15 10]	[4 21] [4 21]	[9 16] [12 13]	…	[25 0] [14 11]
Hamming	[22 3] [16 9]	[2 23] [3 22]	[10 15] [15 10]	…	[25 0] [21 4]
Kulczynski	[25 0] [25 0]	[24 1] [25 0]	[25 0] [25 0]	…	[25 0] [16 9]
RussellRao	[25 0] [25 0]	[25 0] [25 0]	[25 0] [25 0]	…	[25 0] [23 2]

Table 7. Confusion matrixes of kNN with distances for remote homology on 2.1.1.2 family with k-split method

2.1.1.2 family Distance/Similarity Methods	**1. split**	**2. split**	**3. split**	**…**	**k. split**
Bray Curtis	[29 32] [10 51]	[50 11] [12 49]	[44 17] [10 51]	...	[0 56] [0 61]
Euclidean	[42 19] [13 48]	[55 6] [6 55]	[58 3] [11 50]	...	[0 56] [0 61]
Minkowski	[42 19] [13 48]	[55 6] [6 55]	[58 3] [11 50]	...	[0 56] [0 61]
Dice	[42 19] [13 48]	[55 6] [6 55]	[58 3] [11 50]	…	[0 56] [0 61]
Jaccard	[31 30] [11 50]	[57 4] [23 38]	[[46 15] [15 46]	…	[0 56] [0 61]
Chebyshev	[23 38] [5 56]	[32 29] [3 58]	[50 11] [4 57]	…	[0 56] [0 61]
Cosine	[26 35] [12 49]	[59 2] [26 35]	[34 27] [11 50]	...	[0 56] [0 61]
SokalSneath	[31 30] [11 50]	[57 4] [23 38]	[46 15] [15 46]	…	[0 56] [0 61]
Correlation	[27 34] [12 49]	[59 2] [24 37]	[34 27] [13 48]	…	[0 56] [0 61]
Matching	[56 5] [45 16]	[61 0] [35 26]	[60 1] [39 22]	…	[0 56] [0 61]
Rogers Tanimoto	[56 5] [45 16]	[61 0] [35 26]	[60 1] [39 22]	…	[0 56] [0 61]
Sokal Michener	[56 5] [45 16]	[61 0] [35 26]	[60 1] [39 22]	…	[0 56] [0 61]
Canberra	[57 4] [49 12]	[61 0] [48 13]	[60 1] [46 15]	…	[0 56] [0 61]
Hamming	[58 3] [58 3]	[61 0] [61 0]	[61 0] [60 1]	…	[0 56] [0 61]
Kulczynski	[9 52] [1 60]	[61 0] [61 0]	[40 21] [30 31]	…	[0 56] [0 61]
RussellRao	[2 59] [0 61]	[61 0] [61 0]	[25 36] [19 42]	…	[0 56] [0 61]

Table8. Confusion matrixes of kNN with distances for remote homology on 2.28.1.1 family with k-split method

2.28.1.1family Distance/Similarity Methods	**1. split**	**2. split**	**3. split**	**…**	**k. split**
Bray Curtis	[27 4] [4 27]	[27 4] [6 25]	[15 16] [12 19]	…	[0 12] [0 31]
Euclidean	[31 0] [8 23]	[31 0] [11 20]	[11 20] [14 17]	…	[0 12] [0 31]
Minkowski	[31 0] [8 23]	[31 0] [11 20]	[11 20] [14 17]	…	[0 12] [0 31]
Dice	[26 5] [1 30]	[25 6] [2 29]	[29 2] [16 15]	…	[0 12] [0 31]
Jaccard	[26 5] [1 30]	[25 6] [2 29]	[29 2] [16 15]	…	[0 12] [0 31]
Chebyshev	[30 1] [14 17]	[31 0] [17 14]	[31 0] [14 17]	…	[0 12] [0 31]
Cosine	[30 1] [6 25]	[29 2] [3 28]	[31 0] [18 13]	…	[0 12] [0 31]
SokalSneath	[26 5] [1 30]	[25 6] [2 29]	[29 2] [16 15]	…	[0 12] [0 31]
Correlation	[30 1] [5 26]	[29 2] [3 28]	[31 0] [18 13]	…	[0 12] [0 31]
Matching	[21 10] [11 20]	[29 2] [10 21]	[8 23] [10 21]	…	[0 12] [0 31]
Rogers Tanimoto	[21 10] [11 20]	[29 2] [10 21]	[8 23] [10 21]	…	[0 12] [0 31]
Sokal Michener	[21 10] [11 20]	[29 2] [10 21]	[8 23] [10 21]	…	[0 12] [0 31]
Canberra	[21 10] [11 20]	[29 2] [11 20]	[6 25] [8 23]	…	[0 12] [0 31]
Hamming	[17 14] [12 19]	[30 1] [16 15]	[0 31] [4 27]	…	[0 12] [0 31]
Kulczynski	[12 19] [1 30]	[22 9] [1 30]	[31 0] [20 11]	…	[0 12] [0 31]
RussellRao	[8 23] [0 31]	[16 15] [1 30]	[31 0] [26 5]	…	[0 12] [0 31]

Table 9. Confusion matrixes of kNN with distances for remote homology on 3.42.1.5 family with k-split method

3.42.1.5 family Distance/Similarity Methods	**1. split**	**2. split**	**3. split**	**…**	**k. split**
Bray Curtis	[17 3] [7 13]	[12 8] [3 17]	[16 4] [3 17]	…	[2 18] [0 20]
Euclidean	[8 12] [1 19]	[15 5] [4 16]	[5 15] [4 16]	…	[0 20] [0 20]
Minkowski	[8 12] [1 19]	[15 5] [4 16]	[5 15] [4 16]	…	[0 20] [0 20]
Dice	[9 11] [4 16]	[7 13] [1 19]	[6 14] [0 20]	…	[3 17] [0 20]
Jaccard	[9 11] [4 16]	[7 13] [1 19]	[6 14] [0 20]	…	[3 17] [0 20]
Chebyshev	[18 2] [7 13]	[9 11] [4 16]	[13 7] [6 14]	…	[5 15] [1 19]
Cosine	[14 6] [5 15]	[13 7] [0 20]	[10 10] [6 14]	…	[3 17] [0 20]
SokalSneath	[9 11] [4 16]	[7 13] [1 19]	[6 14] [0 20]	…	[3 17] [0 20]
Correlation	[14 6] [4 16]	[13 7] [0 20]	[10 10] [7 13]	…	[3 17] [0 20]
Matching	[13 7] [16 4]	[14 6] [14 6]	[15 5] [18 2]	…	[0 20] [0 20]
Rogers Tanimoto	[9 11] [12 8]	[13 7] [16 4]	[14 6] [14 6]	…	[0 20] [0 20]
Sokal Michener	[19 1] [0 20]	[9 11] [12 8]	[13 7] [16 4]	…	[0 20] [0 20]
Canberra	[11 9] [15 5]	[8 12] [0 20]	[4 16] [3 17]	…	[0 20] [0 20]
Hamming	[10 10] [4 16]	[0 20] [0 20]	[19 1] [20 0]	…	[0 20] [0 20]
Kulczynski	[16 4] [10 10]	[3 17] [1 19]	[19 1] [16 4]	…	[0 20] [0 20]
RussellRao	[20 0] [19 1]	[20 0] [20 0]	[20 0] [20 0]	…	[0 20] [0 20]

Table 10. Confusion matrixes of kNN with distances for remote homology on 7.3.10.1 family with k-split method

7.3.10.1 family Distance/Similarity Methods	**1. split**	**2. split**	**3. split**	**…**	**k. split**
Bray Curtis	[53 0] [0 53]	[52 1] [0 53]	[53 0] [0 53]	…	[0 48] [0 53]
Euclidean	[51 2] [0 53]	[51 2] [1 52]	[53 0] [0 53]	…	[0 48] [0 53]
Minkowski	[51 2] [0 53]	[51 2] [1 52]	[52 1] [0 53]	…	[0 48] [0 53]
Dice	[49 4] [1 52]	[50 3] [1 52]	[52 1] [0 53]	…	[0 48] [0 53]
Jaccard	[49 4] [1 52]	[50 3] [1 52]	[52 1] [0 53]	…	[0 48] [0 53]
Chebyshev	[43 10] [0 53]	[44 9] [0 53]	[50 3] [0 53]	…	[0 48] [0 53]
Cosine	[53 0] [3 50]	[53 0] [3 50]	[53 0] [2 51]	…	[0 48] [0 53]
SokalSneath	[49 4] [1 52]	[50 3] [1 52]	[52 1] [0 53]	…	[0 48] [0 53]
Correlation	[53 0] [3 50]	[53 0] [2 51]	[53 0] [2 51]	…	[0 48] [0 53]
Matching	[41 12] [0 53]	[41 12] [10 43]	[45 8] [0 53]	…	[0 48] [0 53]
Rogers Tanimoto	[41 12] [0 53]	[41 12] [10 43]	[45 8] [0 53]	…	[0 48] [0 53]
Sokal Michener	[41 12] [0 53]	[41 12] [10 43]	[45 8] [0 53]	…	[0 48] [0 53]
Canberra	[39 14] [0 53]	[44 9] [11 42]	[45 8] [0 53]	…	[0 48] [0 53]
Hamming	[37 16] [0 53]	[43 10] [14 39]	[30 23] [0 53]	…	[0 48] [0 53]
Kulczynski	[53 0] [46 7]	[53 0] [48 5]	[53 0] [46 7]	…	[0 48] [0 53]
RussellRao	[53 0] [52 1]	[53 0] [52 1]	[53 0] [52 1]	…	[0 48] [0 53]

Table 11. Precision values of kNN with distances for remote homology with StratifiedKFold cross validation

Distance/Similarity Methods	1.4.1.1 family	2.1.1.2 family	2.28.1.1 family	3.42.1.5 family	7.3.10.1 family
Bray Curtis	0.99588	0.99329	0.99640	0.99484	0.99232
Euclidean	0.99488	0.98718	0.99463	0.99482	0.98946
Minkowski	0.99488	0.98718	0.99463	0.99482	0.98946
Dice	0.99703	0.97462	0.99803	0.99588	0.99337
Jaccard	0.99703	0.97462	0.99803	0.99588	0.99337
Chebyshev	0.99372	0.97017	0.99442	0.99037	0.98616
Cosine	0.99745	0.97396	0.99803	0	0.99312
SokalSneath	0.99703	0.97462	0.99803	0	0.99337
Correlation	0.99797	0.97396	0.99803	0	0.99259
Matching	0.99640	0	0.99435	0	0.98986
Rogers Tanimoto	0.99640	0	0.99435	0.98948	0.98986
Sokal Michener	0.99640	0	0.99435	0.98948	0.98986
Canberra	0.99444	0	0.99411	0.98956	0.99021
Hamming	0.99507	0	0.99461	0.98214	0.99069
Kulczynski	0.99311	0	0.99838	0	0.99443
RussellRao	0.99116	0	0.99774	0	0.99417

Table12. Precision values of kNN with distances for remote homology with k-split method

Distance/Similarity Methods	1.4.1.1 family	2.1.1.2 family	2.28.1.1 family	3.42.1.5 family	7.3.10.1 family
Bray Curtis	0.87282	0.80508	0.83131	0.77941	0.95545
Euclidean	0.85171	0.84105	0.76905	0.79754	0.93080
Minkowski	0.85171	0.84105	0.76905	0.79754	0.93080
Dice	0.84065	0.75623	0.83367	0.77980	0.94413
Jaccard	0.84065	0.75623	0.83367	0.77980	0.94413
Chebyshev	0.85871	0.83131	0.72773	0.74615	0.92664
Cosine	0.82799	0.71722	0.84832	0.76658	0.93272
SokalSneath	0.84065	0.75623	0.83367	0.77980	0.94413
Correlation	0.83215	0.71785	0.85022	0.76822	0.93760
Matching	0.71147	0.66153	0.71530	0.69000	0.83603
Rogers Tanimoto	0.71147	0.66153	0.71530	0.69000	0.83603
Sokal Michener	0.71147	0.66153	0.71530	0.69000	0.83603
Canberra	0.71026	0.64068	0.71177	0.67424	0.83741
Hamming	0.58946	0.46985	0.66648	0.48273	0.79325
Kulczynski	0.49866	0.46784	0.81158	0.60713	0.78858
RussellRao	0.44980	0.43303	0.78770	0.51310	0.76226

Table 13. Recall values of kNN with distances for remote homology with StratifiedKFold cross validation

Distance/Similarity Methods	1.4.1.1 family	2.1.1.2 family	2.28.1.1 family	3.42.1.5 family	7.3.10.1 family
Bray Curtis	0.99356	0.99231	0.99580	0.99310	0.99066
Euclidean	0.99207	0.98718	0.99352	0.99104	0.98639
Minkowski	0.99207	0.98718	0.99352	0.99104	0.98639
Dice	0.99703	0.97821	0.99806	0.99586	0.99253
Jaccard	0.99703	0.97821	0.99806	0.99586	0.99253
Chebyshev	0.99306	0.97308	0.99352	0.98828	0.98026
Cosine	0.99753	0.97692	0.99806	0.99104	0.99333
SokalSneath	0.99703	0.97821	0.99806	0.99586	0.99253
Correlation	0.99802	0.97692	0.99806	0.99586	0.99280
Matching	0.99653	0.97180	0.99255	0.99104	0.98986
Rogers Tanimoto	0.99653	0.97180	0.99255	0.99172	0.98986
Sokal Michener	0.99640	0.97180	0.99255	0.99172	0.98986
Canberra	0.99455	0.97180	0.99255	0.99172	0.99040
Hamming	0.99507	0.97180	0.99385	0.99035	0.99093
Kulczynski	0.99306	0.97180	0.99838	0.99104	0.99440
RussellRao	0.99108	0.97180	0.99773	0.99104	0.99413

Table 14 . Recall values of kNN with distances for remote homology with k-split method

Distance/Similarity Methods	1.4.1.1 family	2.1.1.2 family	2.28.1.1 family	3.42.1.5 family	7.3.10.1 family
Bray Curtis	0.85701	0.79312	0.80961	0.75045	0.95879
Euclidean	0.83356	0.83129	0.74763	0.77162	0.93266
Minkowski	0.83356	0.83129	0.74763	0.77162	0.93266
Dice	0.81310	0.72661	0.80339	0.73739	0.94621
Jaccard	0.81310	0.72661	0.80339	0.73739	0.94621
Chebyshev	0.84368	0.79522	0.70108	0.72072	0.92468
Cosine	0.74299	0.66829	0.80201	0.71554	0.93169
SokalSneath	0.81310	0.72661	0.80339	0.73738	0.94621
Correlation	0.75149	0.67415	0.80454	0.71892	0.93750
Matching	0.63379	0.60483	0.70892	0.61329	0.79357
Rogers Tanimoto	0.63379	0.60483	0.70892	0.61329	0.79357
Sokal Michener	0.63379	0.60483	0.70892	0.61329	0.79357
Canberra	0.62851	0.57602	0.70154	0.60225	0.79357
Hamming	0.53770	0.50412	0.63565	0.51532	0.70238
Kulczynski	0.59563	0.54323	0.74325	0.60856	0.64698
RussellRao	0.58368	0.53152	0.68057	0.58221	0.57151

Table 15. ROC scores of kNN with distances for remote homology with StratifiedKFold cross validation

Distance/Similarity Methods	1.4.1.1 family	2.1.1.2 family	2.28.1.1 family	3.42.1.5 family	7.3.10.1 family
Bray Curtis	0.99492	0.94849	0.98275	0.89444	0.97783
Euclidean	0.97985	0.88996	0.96645	0.96815	0.96504
Minkowski	0.97985	0.88996	0.96645	0.96815	0.96504
Dice	0.95635	0.75900	0.94705	0.79594	0.97579
Jaccard	0.95635	0.75900	0.94705	0.79594	0.97579
Chebyshev	0.92576	0.77339	0.94436	0.84262	0.95065
Cosine	0.94281	0.71702	0.95438	0.61539	0.94090
SokalSneath	0.95635	0.75900	0.94705	0.79594	0.97579
Correlation	0.95696	0.71702	0.95438	0.79594	0.94053
Matching	0.91448	0.64350	0.97277	0.61539	0.93194
Rogers Tanimoto	0.91448	0.64350	0.97277	0.66768	0.93194
Sokal Michener	0.91448	0.64350	0.97277	0.66768	0.93194
Canberra	0.91376	0.64350	0.95028	0.69295	0.92883
Hamming	0.85696	0.64350	0.94444	0.61538	0.92939
Kulczynski	0.79883	0.64350	0.94720	0.61539	0.92815
RussellRao	0.74018	0.64350	0.93221	0.61539	0.92473

Table 16. ROC scores of kNN with distances for remote homology with k-split method

Distance/Similarity Methods	1.4.1.1 family	2.1.1.2 family	2.28.1.1 family	3.42.1.5 family	7.3.10.1 family
Bray Curtis	0.91601	0.90255	0.91399	0.87135	0.98380
Euclidean	0.91084	0.92861	0.88427	0.89543	0.98133
Minkowski	0.91084	0.92861	0.88427	0.89543	0.98133
Dice	0.89475	0.86345	0.92864	0.88191	0.98384
Jaccard	0.89475	0.86345	0.92864	0.88191	0.98384
Chebyshev	0.93337	0.96723	0.86060	0.85833	0.98906
Cosine	0.93484	0.90664	0.94791	0.89194	0.97905
SokalSneath	0.89475	0.86345	0.92864	0.88191	0.98384
Correlation	0.93617	0.90344	0.94904	0.89113	0.98300
Matching	0.88064	0.85618	0.83185	0.83601	0.94934
Rogers Tanimoto	0.88064	0.85618	0.83185	0.83601	0.94934
Sokal Michener	0.88064	0.85618	0.83185	0.83601	0.94934
Canberra	0.88232	0.85901	0.83404	0.83953	0.94911
Hamming	0.86396	0.86550	0.86167	0.83417	0.94358
Kulczynski	0.92077	0.92306	0.94656	0.88212	0.97496
RussellRao	0.91667	0.92960	0.94941	0.88086	0.96984

Table 17. ROC scores of kNN with distances for remote homology with StratifiedKFold cross validation

Distance/Similarity Methods	Lowest ROC	Highest ROC	Mean ROC
Bray Curtis	0.52855	0.99940	0.77673
Euclidean	0.52724	1.0	0.76428
Minkowski	0.52724	1.0	0.76428
Dice	0.52874	1.0	0.72237
Jaccard	0.52874	1.0	0.72237
Chebyshev	0.52855	0.98751	0.77173
Cosine	0.52855	1.0	0.69526
SokalSneath	0.52874	1.0	0.72237
Correlation	0.52855	1.0	0.69551
Matching	0.97058	1.0	0.69007
Rogers Tanimoto	0.52855	1.0	0.69007
Sokal Michener	0.52854	1.0	0.69007
Canberra	0.52855	1.0	0.68911
Hamming	0.52855	0.99774	0.66834
Kulczynski	0.52855	0.97441	0.65961
RussellRao	0.52855	0.96517	0.64057

Table 18. ROC scores of kNN with distances for remote homology with k-split method

Distance/Similarity Methods	Lowest ROC	Highest ROC	Mean ROC
Bray Curtis	0.81516	0.98921	0,92024
Euclidean	0.81795	0.98392	0.91570
Minkowski	0.79848	0.98392	0.91570
Dice	0.74609	0.98861	0,89754
Jaccard	0.74609	0.98861	0,89754
Chebyshev	0.82378	0.98906	0.91341
Cosine	0.75831	0.97920	0.91016
SokalSneath	0.74609	0.98861	0,89754
Correlation	0.76938	0.98300	0.91087
Matching	0.72844	0.97979	0.86811
Rogers Tanimoto	0.72844	0.97979	0.86811
Sokal Michener	0.72844	0.97979	0.86811
Canberra	0.72729	0.97099	0.86359
Hamming	0.69662	0.94358	0.84520
Kulczynski	0.72462	0.97632	0.88270
RussellRao	0.72516	0.99466	0.87927

It has been claimed that SVM-based methods outperform knn-based methods for protein remote homology problem, in Ref. 4. Table 19 shows ROC score comparative for remote homology on proteins. Table 19 shows that the proposed methods with kNN Methods in the study is at least as successful as svm-based methods. In fact, the new k-split method leads to success in most cases SOFM-SMSW in Ref. 34 also uses the knn method while obtaining feature extraction and substitution score. When the kNN method is used with appropriate algorithms, it is seen that it is as successful as SVM for protein remote homology detection.

Table 19. ROC scores Comparative on the methods for remote homology

Distance/Similarity Methods	mean ROC	Ref
kNN with Bray Curtis with StratifiedKFold cross validation with n-gram	0.77673	The Proposed Study
kNN with Bray Curtis with k-split method with n-gram	0,92024	The Proposed Study
SVM-Ngram	0,81200	Ref. 5
SVM with Top Ngram	0,71720	Ref. 5
SVM-Ngram-p1	0, 88700	Ref. 5
SVM-Ngram-KTA	0, 89200	Ref. 5
VBKC	0.92400	Ref. 33
SVM (SW)	0.89600	Ref. 33
SVM (LA)	0.92500	Ref. 33
SVM (MM)	0.87200	Ref. 33
SVM (Mono)	0.91900	Ref. 33
SVM pairwise (SVM PW)	0.7329	Ref. 34
GPkernal	0.76210	Ref. 34
LSTM	0.80240	Ref. 34
SOFM-Top	0.82100	Ref. 34
SOFM-SW	0.92100	Ref. 34
SOFM-SMSW	0.94100	Ref. 34

4. DISCUSSION AND CONCLUSION

Remote homologue protein detection has been shown to be a more difficult problem to solve than homologue protein detection. Because the number of remote homologous proteins is proportionally lower, the problem of imbalanced data arises. However, when only the accuracy values are looked at, it has achieved quite good success with an average of about 98.7% accuracy in the classification with the kNN method with Bray Curtis, Euclidean, Minkowski, Dice, Jaccard, Chebyshev, Cosine, SokalSneath, correlation, matching coefficient, RogersTanimoto, SokalMichener, Canberra, Hamming, Kulczynski, and RussellRao. On the other hand, looking at the confusion matrixes, it is observed that this success is not entirely correct. The reason for this is the imbalanced data problem. Despite the imbalanced data problem, kNN with the Bray Curtis distance and stratified cross validation with novel *k* fold and novel k-split method are promising in this problem.

Based on these results in the article, its contributions to the literature are as follows:

a. kNN with stratified k-fold CV has been observed to be a successful method for remote homologous protein detection.
b. kNN with novel k-split method has been observed to be a successful method for remote homologous protein detection.
c. kNN with stratified k-fold CV has been successful for the imbalanced dataset. Imbalanced data set is an important problem that we encounter in many data sets.
d. In the remote homology problem, after kNN tried 16 different distances, Bray Curtis was the most successful, Euclidean and Minkowski distances were the second success. Therefore, information is given on the comparison of the performances of 16 different distance methods for both imbalanced data sets and remote homology problem in this study.
e. An automatic k value formula is suggested for the stratified k-fold CV method. Thus, the trouble of searching for k values randomly or by experimenting is avoided.
f. The new beneficial k-split method is proposed to solve imbalanced problem.

In the future work of this study, studies can be carried out on the selection of protein features that are meaningful for remote homolog detection or a different problem. New methods can be developed for protein feature extraction. Since the protein dataset is growing rapidly, big data technologies can be used to keep protein data. New methods about cross-validation or resampling can be developed to solve the imbalanced data problem, which is the most important and fundamental problem of this study.

CONFLICT OF INTEREST

The authors stated that there are no conflicts of interest regarding the publication of this article.

REFERENCES

[1] Li J, Wong L, Yang Q. Guest editors' introduction: Data Mining in Bioinformatics. IEEE Intell. Systems, 2005; 20(6):16-18.

[2] Yoo I, Alafaireet P, Marinov M, Pena-Hernandez K, Gopidi R, Chang J.-F, Hua L. Data mining in healthcare and biomedicine: a survey of the literature. Journal of medical systems, 2012; 36(4):2431-2448.

[3] Chen J, Guo M, Wang X, Liu B. A comprehensive review and comparison of different computational methods for protein remote homology detection. Briefings in Bioinformatics, 2018; 19(2): 231-244.

[4] Liao L, Noble WS. Combining pairwise sequence similarity and support vector machines for detecting remote protein evolutionary and structural relationships. Journal of computational biology, 2003; 10(6), 857-868.

[5] Lovato P, Cristani M, Bicego M. Soft Ngram representation and modeling for protein remote homology detection. IEEE/ACM transactions on computational biology and bioinformatics, 2016; 14(6), 1482-1488.

[6] Dong QW. Lin L, Wang XL, Li MH. A pattern-based SVM for protein remote homology detection. In 2005 International Conference on Machine Learning and Cybernetics, 2005; Vol.6, 3363-3368. IEEE.

[7] Beaume N, Ramstein G, Jacques Y. An expert-based approach for the identification of remote homologs. WCSB. 2008; (pp. 17-20).

[8] Murzin AG, Brenner SE, Hubbard T, Chothia C. SCOP: a structural classification of proteins database for the investigation of sequences and structures. Journal of molecular biology, 1995; 247(4), 536-540.

[9] Harris A, Jones SH. Words. In Writing for Performance. 2016; (pp. 19-35). Rotterdam, Netherlands: Sense.

[10] Ofer D, Brandes N, Linial M. The language of proteins: NLP, machine learning & protein sequences. Computational and Structural Biotechnology Journal, 2021.

[11] Sushma K. Vani KS. Protein Secondary Structure Extraction using Bag of Words Model. ICETCSE 2016 Special Issue International Journal of Computer Science and Information Security, 2016; 14,106-110.

[12] Kumar NP, Rao MV, Krishna PR, Bapi RS. Using sub-sequence information with kNN for classification of sequential data. International Conference on Distributed Computing and Internet Technology. Springer, Berlin, Heidelberg, 2005. (p. 536-546).

[13] Abu Alfeilat HA, Hassanat AB, Lasassmeh O, Tarawneh A S, Alhasanat M B, Eyal Salman H S Prasath V S. Effects of distance measure choice on k-nearest neighbor classifier performance: A review. Big data, 2019; 7(4), 221-248.

[14] Fix E, Hodges JL. Discriminatory analysis. Nonparametric discrimination; consistency properties. Technical Report 4, USAF School of Aviation Medicine, Randolph Field, TX, USA. 1951.

[15] Han Jiawei, Kamber Micheline. Data Mining, Concepts and Techniques, Morgan Kaufmann Publishers. 2001.

[16] Michie MG. Use of the Bray-Curtis similarity measure in cluster analysis of foraminiferal data. Journal of the International Association for Mathematical Geology,1982; 14(6), 661-667.

[17] Mousa A, Yusof Y. Fuzzy C-Means with Improved Chebyshev Distance for Multi-Labelled Data. Journal of Engineering and Applied Sciences, 2018;13(2), 353-360.

[18] Qian G, Sural S, Gu Y, Pramanik S. Similarity between Euclidean and cosine angle distance for nearest neighbor queries. In Proceedings of the 2004 ACM symposium on Applied computing 2004, March; (pp. 1232-1237).

[19] Li B, Han L. Distance weighted cosine similarity measure for text classification. In International conference on intelligent data engineering and automated learning. 2013, October; (pp. 611-618). Springer, Berlin, Heidelberg.

[20] Prasath VB, Alfeilat HAA, Hassanat A, Lasassmeh O, Tarawneh AS, Alhasanat MB, Salman H SE. Distance and Similarity Measures Effect on the Performance of K-Nearest Neighbor Classifier--A Review, 2017; arXiv preprint arXiv:1708.04321.

[21] Stabili D, Marchetti M, Colajanni M. Detecting attacks to internal vehicle networks through Hamming distance. In 2017 AEIT International Annual Conference, 2017 September; (pp. 1-6). IEEE.

[22] Cha SH. Comprehensive survey on distance/similarity measures between probability density functions. International J. Mathematical Models and Methods in Applied Science, 2007; 1(4), 300-307.

[23] Kocher M. Savoy J. Distance measures in author profiling. Information processing & management, 2017; 53(5), 1103-1119.

[24] Boyce RL, Ellison PC. Choosing the best similarity index when performing fuzzy set ordination on binary data. Journal of Vegetation Science, 2001; 12(5), 711-720.

[25] Chay ZE, Lee CH, Lee KC. Oon JS, Ling M. Russel and Rao coefficient is a suitable substitute for Dice coefficient in studying restriction mapped genetic distances of Escherichia coli. Computational and Mathematical Biology, 2010;1(1), 1-9.

[26] Stacey B. A Standardized Treatment of Binary Similarity Measures with an Introduction to k-Vector Percentage Normalized Similarity, 2016.

[27] Putra RE. Suciati N. Wijaya AY. Implementing content based image retrieval for Batik using Rotated Wavelet Transform and Canberra distance. Image, 2011; 2(3), 4-5.

[28] Choi SS. Cha SH. Tappert CC. A survey of binary similarity and distance measures. Journal of systemics, cybernetics and informatics, 2010; 8(1), 43-48.

[29] Yan H, Zhou X, Ge Y, Neighborhood repulsed correlation metric learning for kinship verification. 2015 Visual Communications and Image Processing (VCIP), December 2015; IEEE, 1-4.

[30] Berrar D. Cross-validation. Encyclopedia of bioinformatics and computational biology, 2019; 1:542-545.

[31] Breiman L. Friedman JH. Olshen RA. Stone CJ. Classification and Regression Trees (Wadswort h International Group). 1984.

[32] Zeng X. Martinez TR. Distribution-balanced stratified cross-validation for accuracy estimation. Journal of Experimental & Theoretical Artificial Intelligence, 2000; 12(1), 1-12.

[33] Damoulas T, Girolami MA. Probabilistic multi-class multi-kernel learning: on protein fold recognition and remote homology detection. Bioinformatics,. 2008; 24(10), 1264-1270.

[34] Nakshathram S. Duraisamy R. Pandurangan M. Sequence-Order Frequency Matrix-Sampling and Machine learning with Smith-Waterman (SOFM-SMSW) for Protein Remote Homology Detection, 2021