k Nearest Neighbors,
unbalanced data,
incomplete data

Tomasz ORCZYK[1], Piotr PORWIK[1], Marcin LEWANDOWSKI[1], Marcin CHOLEWA[1]

INSTANCE BASED KNN MODIFICATION FOR CLASSIFICATION OF MEDICAL DATA

Paper describes a novel modification to a well known kNN algorithm, which enables using it for medical data, which often is a class-imbalanced data with randomly missing values. Paper presents the modified algorithm details, experiment setup, results obtained on a cross validated classification of a benchmark database with randomly removed values (missing data) and records (class imbalance), and their comparison with results of the state of the art classification algorithms.

1. INTRODUCTION

Classification algorithms are crucial part of decision support systems. Nowadays such systems gain a share on a field of medicine [3], [8] which brings specific problems to be solved. Due to overload of medical units triage-like systems are more accepted. These are systems that help to pre-qualify patient for a specific (and often expensive) medical diagnostic routines using only generic tests and interviews. Such tests alone may be non-specific, but sometimes it is possible to use many non-specific tests and examinations to induce a specific and quite accurate diagnosis.

Before such system may be used in real life it must prove its usefulness and accuracy. This has to be done on an archive data, which brings many issues. Typical ones are: different number of patients with different medical conditions or a condition stage, different sets of data for different patients, a high amount of measured parameters, examinations cannot be repeated to acquire missing data. Medical diagnosis relies on a similarity to known, traditionally diagnosed cases, so a choice of a kNN classifier seems to be quite natural. Unfortunately the basic implementation of the kNN classifier [1] does not perform well on data that has many features. Proposed classifier treats each feature as a separate dimension of a feature space. When a measure such as an Euclidean distance, which is used to determine the nearest neighbors in the kNN algorithm, is defined using many coordinates, the more dimensional space it is, distances between different pairs of samples are becoming more uniform and value of a single parameter has less effect on a distance - this phenomenon is called the curse of dimensionality [10].

Another problem is the fact that different patients might have done a different set of examinations, resulting in different features being available for different reference (training) and test samples. Conventional methods of dealing with missing values [6] cannot be applied here, as either re-acquiring data nor reducing the dimension which is not determined for all samples is

[1]University of Silesia, e-mail:{tomasz.orczyk, piotr.porwik, marcin.lewandowski, marcin.cholewa}@us.edu.pl

not possible. The best option would be using all available features for each sample.
Last problem addressed in this paper is unequal class distribution which, in extreme cases, may lead to the situation in which classifier never chooses the minority class.

2. CLASSIFIER ARCHITECTURE

Proposed algorithm may be understand as an ensemble of instance based kNN classifiers, each utilizing only a single feature. This architecture is also know as a feature projection kNN [2]. This approach has several advantages over a regular kNN, as each feature is analyzed independently:

- there is no need for data normalization,
- problem of missing values doesn't exist (only available values for each feature are utilized),
- there is no need for a complex, high-dimensional distance calculation,
- effects of the curse of dimensionality are minimized,
- feature weights are easy to implement.

Instead of returning a single winning class (like regular kNN does), each classifier, operating on a single feature, returns support values for all classes. These values, coming from all partial classifiers, are summarized in order to select a winning class. Support value is defined as an a posteriori probability that analyzed instance belongs to a given class. For the kNN the support value may be simply defined as a count of instances belonging to a given class among all nearest neighbors (divided by the number of nearest neighbors if it is about to be treated as a probability value):

$$v^f(c) = \frac{{k'^f}_c}{k'^f}, \qquad f = 1, \ldots, m, \;\; c = 1, \ldots, n, \tag{1}$$

where:
$v^f(c)$ is a support value for class c coming from the feature f,
k'^f is a true number of analyzed neighbors within a feature f,
${k'^f}_c$ is a number of instances that belong to class c within k' analyzed neighbors for feature f.

The process of electing a winner is now a case of summarizing support values for all classes from all features (2), and choosing the class with the greatest value (3):

$$V(c) = \sum_{f=1}^{m} v^f(c), \qquad c = 1, \ldots, n, \tag{2}$$

$$N = \underset{c}{\operatorname{argmax}}\{V(c) : c = 1, \ldots, n\}, \tag{3}$$

where:
$v^f(c)$ is a support value of class c coming from the feature f,
$V(c)$ is a summarized support value of class c,
N is an identifier of the winning class (aggregated decision).

To improve classifier accuracy on imbalanced datasets a class imbalance compensation was introduced in a form of additional instance class weights. The class score based on the number of instances belonging to that class within the given set of nearest neighbors is additionally weighted according to the formula:

$$w_c = 1 - \frac{n(T_c)}{n(T)}, \tag{4}$$

where:
w_c is a weight factor for a class c,
$n(T)$ is a cardinality of the training set,
$n(T_c)$ is a number of instances that belong to the class c in the whole training set.

This can be understand as a complement of a priori probability of choosing an instance belonging to a given class from the training set.

Following figures illustrate the difference in the winner election process between a regular kNN (Fig. 1) and the proposed algorithm (Fig. 2):

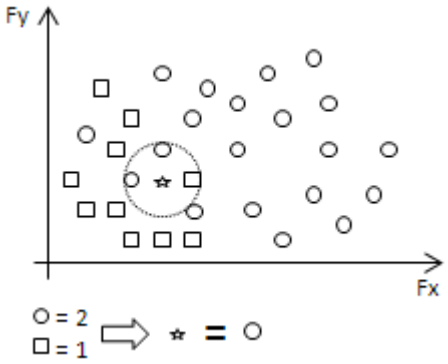

Fig. 1. Regular kNN and and its winner election process.

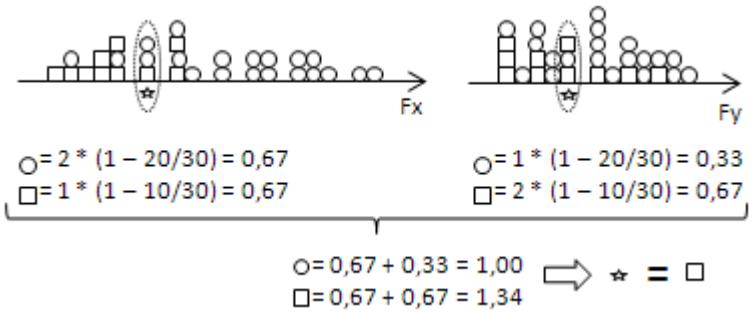

Fig. 2. Feature projection kNN with instance class weighting and its winner election process.

where:
★ is a classified sample,
○ is an instance of class ○ ($n(○) = 20$),
□ is an instance of class □ ($n(□) = 10$).

In the kNN algorithm there are two kinds of possible ties:

- same number of neighbors from different classes among analyzed nearest neighbors,
- same distance from the classified instance to two or more training instances.

With low and odd number of analyzed nearest neighbors (i.e. $k = 3$) and the fact that winner is chosen after summarizing all support values from all features, chances of ties of the first type are low and additionally instance class weighting further minimizes it. This kind of ties is solved by the rule that the class which is higher on a class list wins (ie. alphabetical order may decide).

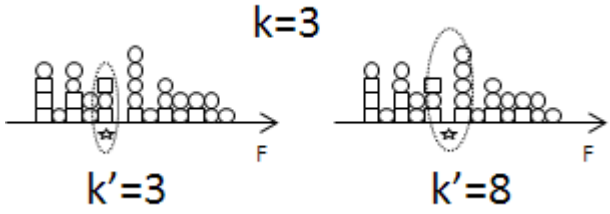

Fig. 3. Left: no ties, right: equal distance ties.

Second kind of ties (Fig. 3) is solved by including all neighboring instances that are equally distant from the classified instance. So k defines minimum number of analyzed neighbors, and the real number of analyzed neighbors varies from case to case (so $k' \geq k$). For example, lets assume that the attribute is an integer value, $k = 3$, the training set consists of 6 instances, and there are 2 neighbors in distance = 1, 2 neighbors in distance = 2, and 2 neighbors in distance = 3. The resulting numbers of analyzed neighbors will be $k' = 4$: 2 neighbors in distance = 1 gives $k' = 2$, $2 \ngeq 3$ so the next closest distance is considered, and there are 2 neighbors in distance = 2, so now we have $k' = 4$ neighbors and $4 \geq 3$ what terminates the nearest neighbor search routine.

3. EXPERIMENT DESCRIPTION

3.1. WINE DATASET

Authors couldn't find any publicly available medical database suitable for this comparison. The wine dataset [5] contains results of chemical analysis of wines and thus it can be used as a substitute of a medical dataset. It has 3 classes, which were balanced by removing random rows from majority classes (leaving 48 instances in each class), and 13 attributes without any missing values. In these aspects is a perfect candidate for a benchmark database.

For the purpose of this experiment a number of degraded databases has been derived. These databases have randomly removed a given number of values and records to introduce missing values and class imbalance. Datasets used in the experiments contained 0%, 5%, 15%, 25%, and 35% of missing values (due to method of elimination random values these amounts may not be perfectly matched). For each experiment rows were removed incrementally and in each test case the same rows were removed (random seed was static). Results of proposed algorithm (ICFP-kNN) were compared with a feature projection kNN classifier without additional weighting (FP-kNN), a regular kNN classifier (kNN) and with a C4.5 based PART classifier (C4.5) [4].

As a measure of classifier accuracy an overall accuracy has been used. Overall accuracy for multiclass classification is defined as follows:

$$Overall\ Accuracy = \frac{Number\ of\ correctly\ classified\ instances}{Total\ number\ of\ instances} \times 100\% \qquad (5)$$

First experiment has been done on a balanced dataset and results of this experiment are presented in a form of a graph on Fig. 4.

For a minority class endurance test three cases were tested, where the cardinality of one class was reduced respectively to 50%, 25%, and 10% of the initial number of instances in that class. Results of these experiments are presented in a form of a graph on Fig. 5.

Lastly, for a class imbalance endurance test, also three cases were tested, in which cardinality of two out of three classes was reduced respectively to 75% and 50%, 50% and 25%, and 25%

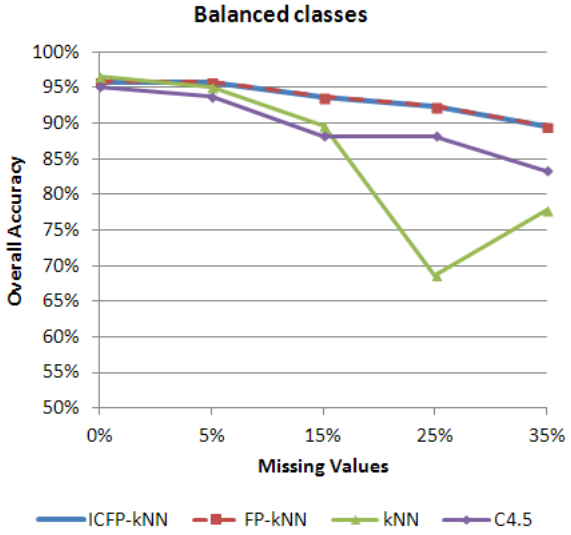

Fig. 4. Classification overall accuracy on a balanced dataset.

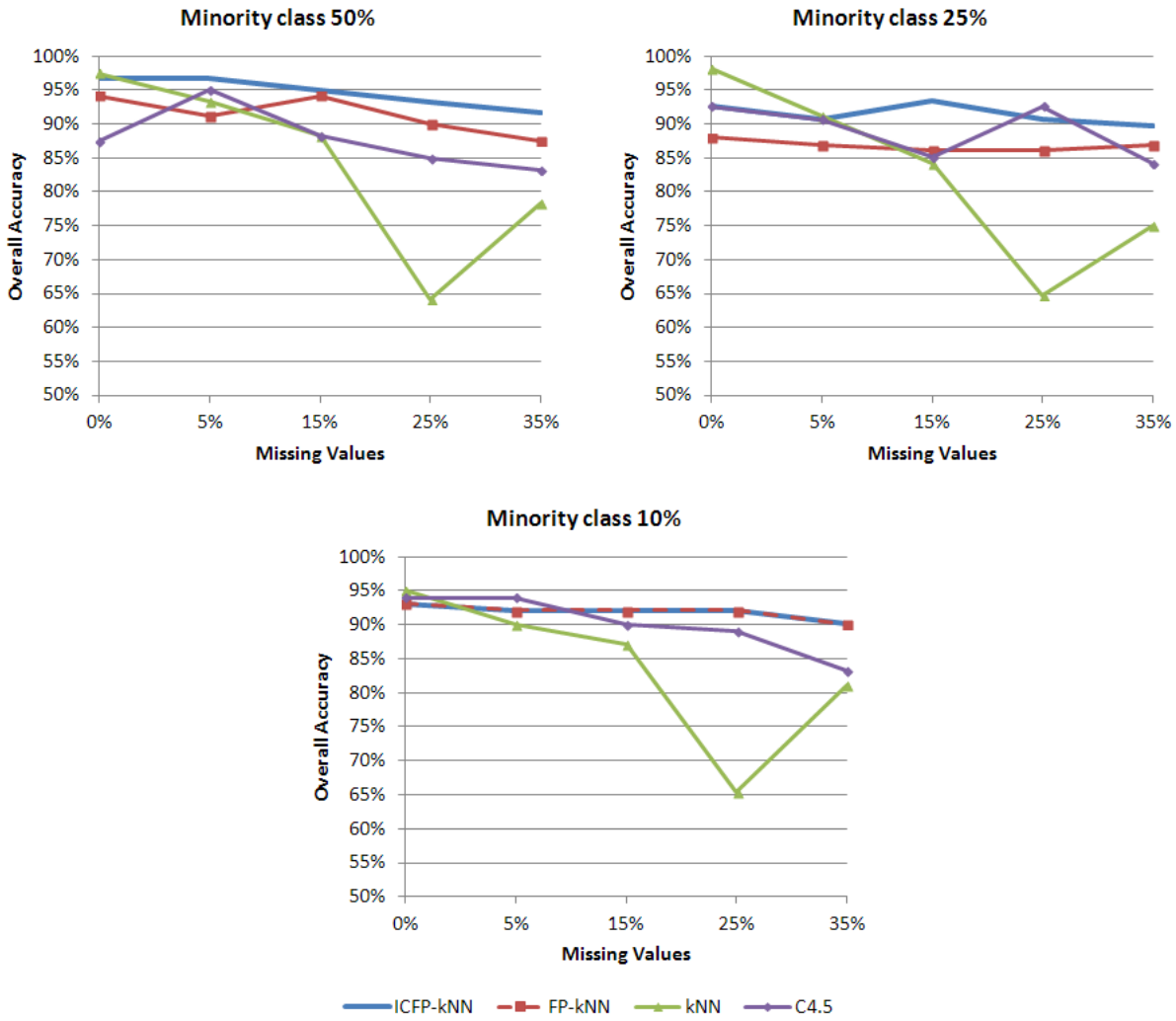

Fig. 5. Classification overall accuracy on datasets with a single minority class.

and 10% of the initial number of instances in these classes. Results of these experiments are presented in a form of a graph on Fig. 6.

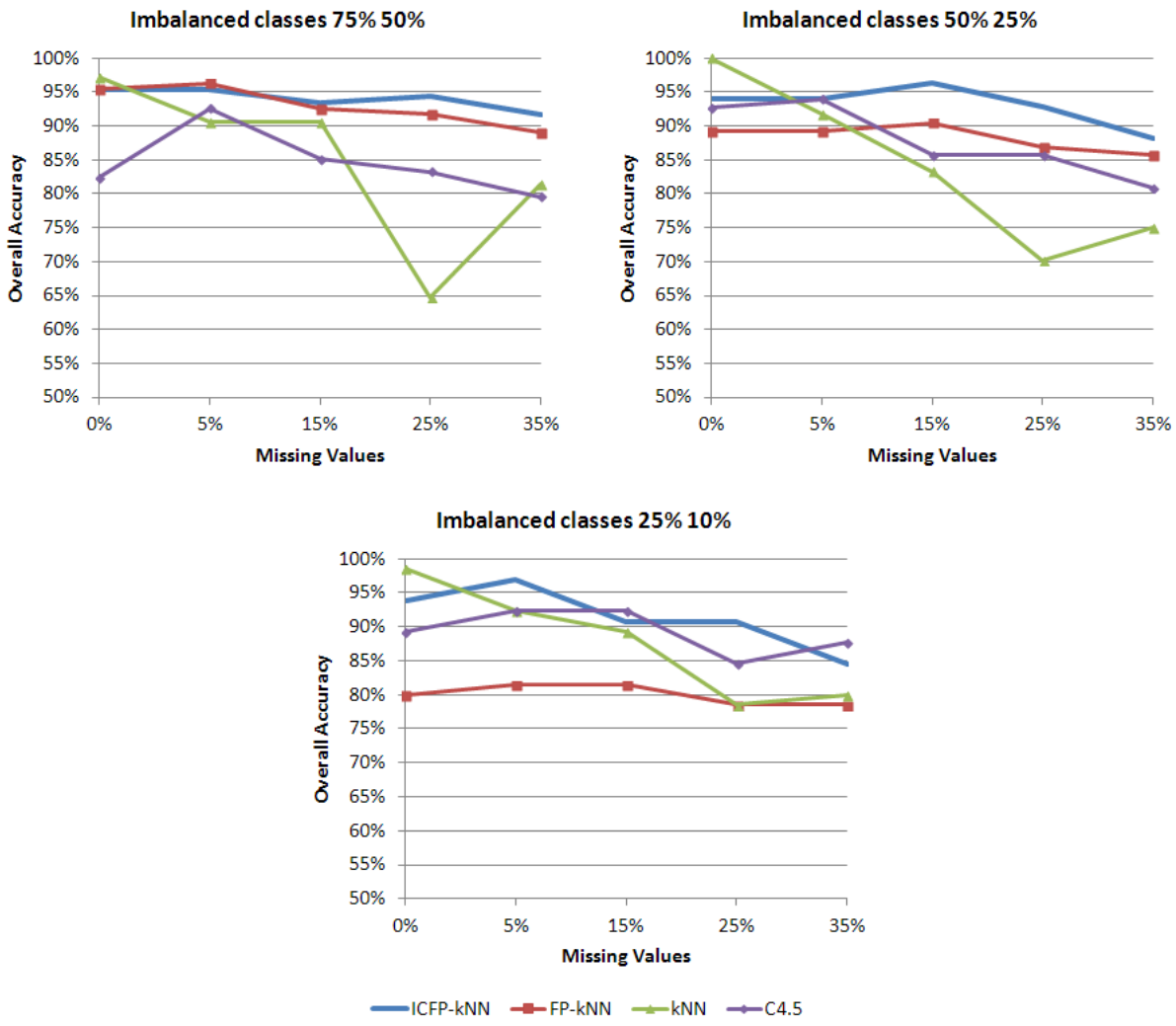

Fig. 6. Classification overall accuracy on imbalanced datasets.

3.2. HCV DATASET

This test (Fig. 7) is a comparison of accuracy of previously tested classifiers and proposed classifier on a real medical database. Data used in this experiment comes from the Gastroenterology and Hepatology Department of the Independent Public Central Hospital of the Silesian Medical University. It contains medical records of 290 patients infected with a hepatitis virus type C. These records consist of patients' age, routine blood test results (26 features) and a liver biopsy result (3 classes: L - 129 instances, M - 102 instances, H - 59 instances), 25% of values is missing. More detailed description of this database can be found in [7]. There is no reference point set for it, as the database naturally contained missing values and class imbalance.

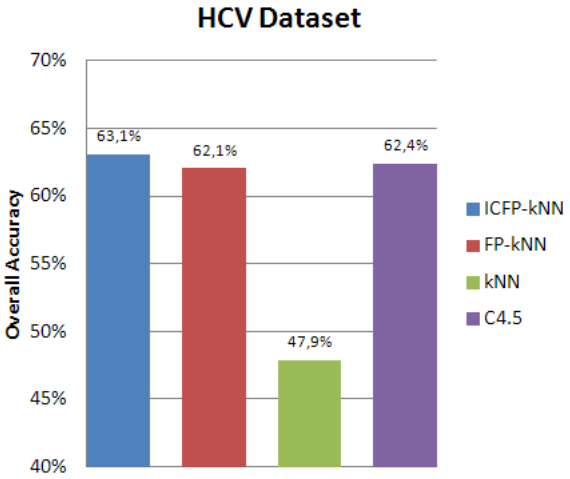

Fig. 7. Classification overall accuracy on a HCV dataset.

4. RESULT DISCUSSION

Results of experiments prove the usefulness of the proposed modification, at least on specific type of data. In the means of the overall accuracy, beginning from 5% of missing values in the dataset, proposed method has outperformed an FP-kNN, a regular kNN classifier, and in a majority of cases also a commonly used C4.5 based classifier. Algorithm maintains a satisfactory classification accuracy even in a case of strong class imbalance and on datasets with a significant number of missing values at one time. Imbalance compensation by decision weighting improved feature projection kNN accuracy on datasets with disproportion of instances from different classes of less than 1:9. In case of balanced classes and in case of cardinality of one class was 10% of cardinality of another class results were the same as for the regular feature projection kNN. On a real medical dataset, proposed algorithm, has also achieved best overall accuracy among tested methods, significantly outperforming a regular kNN.

5. CONCLUSIONS

Presented results are promising, current works show a chance of further improvement of proposed classifier accuracy, by using a wrapper type (using the same classifier) feature selection. The question how to compare classifier accuracy on an imbalanced dataset remains open. Measures commonly used in many papers to assess or compare classifier accuracy like overall accuracy, sensitivity or specificity may not be meaningful if used on a multi-class dataset with a severe class imbalance or a single minority class. For example if we consider a two class dataset containing 990 instances of class A, and 10 instances of class B, a classifier which would always return class A as a winner would achieve 99% overall accuracy, sensitivity for class A would be 100% (0% for class B), and specificity for class A would be 0% (100% for class B). So possibly it would be a better idea to use an F-measure or a G-measure in such cases [9].

BIBLIOGRAPHY

[1] AHA D. W., KIBLER D., ALBERT M. K. Instance-based learning algorithms. Machine Learning, 1991, Vol. 6. pp. 37–66.
[2] AKKUS A., GÜVENIR H. A. knearest neighbor classification on feature projections. Proceedings of the 13th International Conference on Machine Learning, 1996. Morgan Kaufmann, pp. 12–19.

[3] FOSTER K., KOPROWSKI R., SKUFCA J. Machine learning, medical diagnosis, and biomedical engineering research - commentary. BioMedical Engineering OnLine, 2014, Vol. 13. p. 94.
[4] FRANK E., WITTEN I. H. Generating accurate rule sets without global optimization. Fifteenth International Conference on Machine Learning, 1998. Morgan Kaufmann, pp. 144–151.
[5] LICHMAN M. UCI machine learning repository. 2013.
[6] LITTLE R., RUBIN D. Statistical analysis with missing data. 1987. John Wiley & Sons.
[7] ORCZYK T., PORWIK P., BERNAŚ M. Medical diagnosis support system based on the ensemble of single-parameter classifiers. Journal of Medical Informatics & Technologies, 2014, Vol. 23. pp. 173–179.
[8] PORWIK P., SOSNOWSKI M., WESOLOWSKI T., WROBEL K. A computational assessment of a blood vessel's compliance: A procedure based on computed tomography coronary angiography. Hybrid Artificial Intelligent Systems, 2011, Vol. 6678 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 428–435.
[9] POWERS D. M. W. Evaluation: From precision, recall and f-measure to roc, informedness, markedness & correlation. Journal of Machine Learning Technologies, 2011, Vol. 2. pp. 37–63.
[10] STEINLEY D. Curse of dimensionality. Encyclopedia of measurement and statistics, 2007. SAGE Publications, pp. 210–212.