k-Nearest Neighbor Classification Using Dissimilarity Increments

Helena Aidos and Ana Fred

Instituto de Telecomunicações, Instituto Superior Técnico, Lisboa, Portugal
{haidos,afred}@lx.it.pt

Abstract. In this paper we propose a classification method that generalizes the k-nearest neighbor (k-NN) rule in a maximum *a posteriori* (MAP) approach, using an additional characterization of the datasets. That characterization consists of a high order dissimilarity called dissimilarity increment; this dissimilarity measure uses information from three points at a time, unlike typical distances which are pairwise measures. In practice, in this model, the likelihood of a point not only depends of its direct k neighbors, but also of the nearest neighbor of each one of its k neighbors. Experimental results show that the proposed classifier outperforms more traditional and simple classifiers like Naive Bayes and k-nearest neighbor classifiers. This improved performance is especially noticeable relative to k-NN when k is poorly chosen.

Keywords: classification, dissimilarity increments, maximum *a posteriori*, k-nearest neighbor.

1 Introduction

Pattern recognition is an important area of engineering, with applications in fields such as biology, marketing, computer vision, and remote sensing. It is a broad field with fuzzy boundaries; generally, one can say that it involves the automatic detection of interesting structures in sets of data, with little or no intervention from human experts.

In classification, one has access to a set of objects which have already been labeled (by human experts or by some other means) and the goal is to use that knowledge to label new data objects. Formally, the goal is to assign an object from the test set, $\mathbf{x}$, to one of M classes $c_1, \ldots, c_M$ [2,6]. There are numerous ways to do this assignment, either assuming probabilistic models for the data, or based on dissimilarities computed between target objects and a representative training set. The k nearest neighbors algorithm (k-NN) is a very popular algorithm in the latter class, which assigns the new object to a class determined by the most frequent class among its k closest objects within the training set. Other popular approaches include Naive Bayes, neural networks, support vector machines (SVMs) and Parzen windows, among many others [2,5,7].

In a Bayesian approach, the classification consists in computing conditional probabilities, the *a posteriori* probabilities, $p(c_i|\mathbf{x})$, $i = 1, \ldots, M$, for an unknown pattern, $\mathbf{x}$, and assign that pattern to the class with the highest conditional probability value. This decision rule is known as Maximum A Posteriori (MAP) [2,5,7].

A. Campilho and M. Kamel (Eds.): ICIAR 2012, Part I, LNCS 7324, pp. 27–33, 2012.

In this paper we focus on the MAP approach and the k-NN density estimation. We develop an algorithm that takes into account the k nearest neighbors of a pattern and also a high-order dissimilarity measure, called *dissimilarity increment*, which uses information from three points at a time. The use of this measure is motivated by the fact that a pattern may be misclassified by k-NN if the classes are close or overlapped.

This paper is organized as follows: Section 2 starts with a brief presentation of the dissimilarity increments distribution (DID) and formulates the proposed algorithm, MAP-kDID, which is a maximum *a posteriori* approach that uses k-NN and the DID to obtain the likelihood probability. We present, in Section 3, the performance of the proposed algorithm for real datasets from UCI Machine Learning Repository, in comparison with other traditional classification algorithms. Conclusions are drawn in Section 4.

2 The Algorithm MAP-*k*DID

2.1 Dissimilarity Increments Distribution

Let X be a set of patterns, and $(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k)$ a triplet of nearest neighbors belonging to X, where $\mathbf{x}_j$ is the nearest neighbor of $\mathbf{x}_i$ and $\mathbf{x}_k$ is the nearest neighbor of $\mathbf{x}_j$, different from $\mathbf{x}_i$. The *dissimilarity increment* (DI) [4] between these patterns is defined as

$$d_{inc}(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k) = |d(\mathbf{x}_i, \mathbf{x}_j) - d(\mathbf{x}_j, \mathbf{x}_k)| \,. \tag{1}$$

Here, $d(\cdot, \cdot)$ is some pairwise dissimilarity measure or distance. The measure d_{inc} contains information different from a distance: the latter is a pairwise measure, while the former is a measure for a triplet of points; it is thus a measure of higher-order dissimilarity of the data.

In [1] the DIs distribution (DID) was derived for a vectorial feature space, using the Euclidean distance as the pairwise dissimilarity measure, and under the hypothesis of Gaussian distribution of the data. This distribution was written as a function of the mean value of the DIs, λ. The mathematical expression of the DID is given by

$$\begin{aligned} p_{d_{inc}}(w;\lambda) = \frac{\pi\beta^2}{4\lambda^2} w \exp\left(-\frac{\pi\beta^2}{4\lambda^2}w^2\right) + \frac{\pi^2\beta^3}{8\sqrt{2}\lambda^3}\left(\frac{4\lambda^2}{\pi\beta^2} - w^2\right) \times \\ \times \exp\left(-\frac{\pi\beta^2}{8\lambda^2}w^2\right) \operatorname{erfc}\left(\frac{\sqrt{\pi}\beta}{2\sqrt{2}\lambda}w\right), \end{aligned} \tag{2}$$

where $\operatorname{erfc}(\cdot)$ is the complementary error function, and $\beta = 2 - \sqrt{2}$. We present in Figure 1 an example of the fit of the DID to a histogram of increments. Those increments were computed using the Euclidean distance in a Gaussian dataset with 1000 samples in 5 dimensions.

In this paper, we will use feature vector representations (all the experiments performed here use this representation) and the notation from the previous paragraphs.

2.2 MAP-*k*DID

Let $\{\mathbf{x}_i, c_i, S_i\}_{i=1}^N$ denote the labeled dataset, where $\mathbf{x}_i$ is a feature vector in $\mathbb{R}^D$ representing an object, c_i is the corresponding class label, and $S_i = \{w_{i_1}, \ldots, w_{i_{M_r}}\}$ is

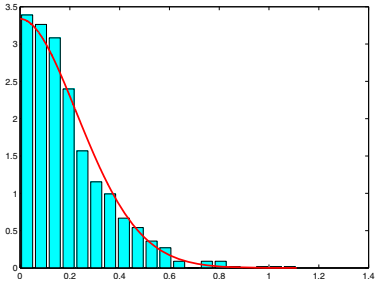

Fig. 1. Scaled histogram of increments and fit dissimilarity increments distribution for a Gaussian dataset

the set of increments yielded by all the triplets of nearest neighbors points containing $\mathbf{x}_i$. We assume that each class c_i has a single statistical model for the increments, with an associated parameter λ_i. We present an illustrative example of this labeled dataset in Figure 2.

We design a maximum a posteriori (MAP) classifier that combines the k-nearest neighbor (k-NN) density estimator and the information given by the increments, assuming that $\mathbf{x}_i$ and S_i are conditionally independent given c_j. Therefore, $p(\mathbf{x}_i, S_i|c_j) = p(\mathbf{x}_i|c_j)p(S_i|c_j)$.

Thus, the class-conditional density of the vector $\mathbf{x}_i$, according to the k-NN density estimator is given by

$$p(\mathbf{x}_i|c_j) = \frac{N_{c_j}}{kV}, \tag{3}$$

with k the number of neighbors, N_{c_j} the number of points of class c_j among the k nearest neighbors of $\mathbf{x}_i$, and V is the volume of a neighborhood of $\mathbf{x}_i$ containing its k nearest neighbors [7]. The class-conditional density of the set of increments associated with $\mathbf{x}_i$ is given by

$$p(S_i|c_j) = \frac{1}{M_r}\sum_{n=1}^{M_r} p(w_{i_n}|c_j), \tag{4}$$

where M_r is the number of increments of the set S_i, w_{i_n} is the n-th increment of that set, and $p(w_{i_n}|c_j) = p(w_{i_n}|\lambda_j)$ is the DID given by equation (2).

We classify new patterns according to the maximum a posteriori (MAP) rule, defined by

$$\mathbf{x}_i \in c_j : j = \arg\max_l \, p(c_l|\mathbf{x}_i, S_i). \tag{5}$$

According to the Bayes rule we can write

$$p(c_j|\mathbf{x}_i, S_i) = \frac{p(\mathbf{x}_i, S_i|c_j)p(c_j)}{\sum_m p(\mathbf{x}_i, S_i|c_m)p(c_m)} = \frac{N_{c_j}p(S_i|c_j)p(c_j)}{\sum_m N_{c_m}p(S_i|c_m)p(c_m)}. \tag{6}$$

Note that this expression does not depend on k or V. We used as a prior class probability, $p(c_j)$, the percentage of points of the training set belonging to class c_j. Each pattern is assigned the label which maximizes the posterior probability.

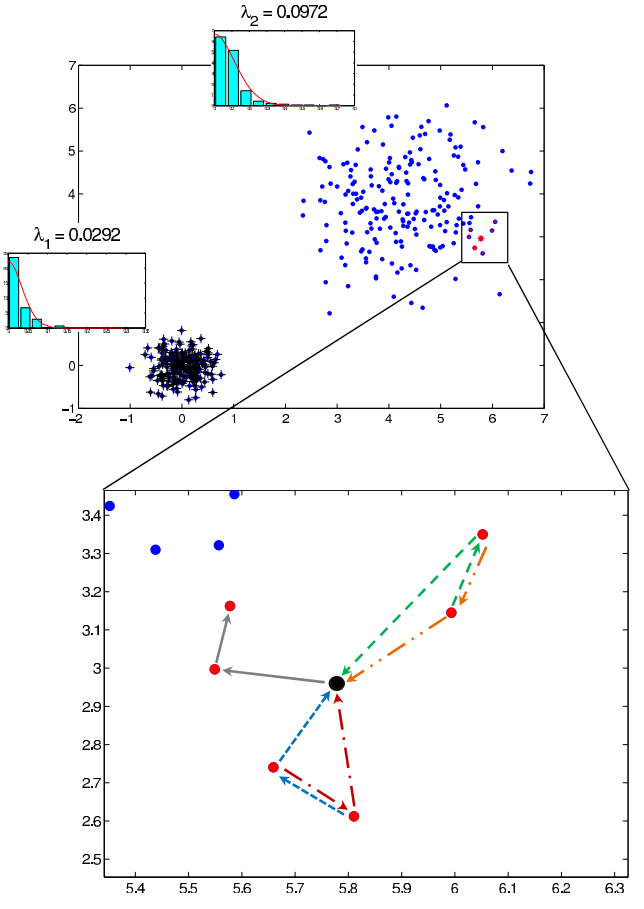

Fig. 2. Illustrative example of a labeled dataset, $\{\mathbf{x}_i, c_i, S_i\}_{i=1}^{N}$, exploring dissimilarity increments. This dataset is composed by two classes, with a statistical model λ_i for each class represented in the histograms. Also, we present an example of a dissimilarity increments set, S_i, in the zoomed area in the bottom. The zoom-in is centered around a point $\mathbf{x}_i$ from the test data. In this example, there are 5 increments containing $\mathbf{x}_i$, each represented with two arrows of the same color. The direction of the arrow indicates the result of the search of triplets of nearest neighbors points. S_i thus has $M_r = 5$ increments. The red points are the training points involved in the computation of $\mathbf{x}_i$'s increments.

3 Experimental Results and Discussion

To test the performance of the proposed method we used 12 real-world datasets from the UCI Machine Learning Repository[1]. See Table 1 for a summarized description of each dataset.

[1] `http://archive.ics.uci.edu/ml`

Table 1. Real-world datasets with the corresponding number of samples, number of features and number of classes

Data	# samples	# features	# classes
Breast-cancer	683	9	2
Crabs	200	5	2
Iris	150	4	3
Pima	768	8	2
Wdbc	569	14	2
Ionosphere	351	10	2
Austra	690	15	2
German	1000	24	2
Heart	270	9	2
Liver	345	6	2
Auto-mpg	398	6	2
Uci-Segmentation	2310	10	7

We compared the proposed algorithm (MAP-kDID) with k-NN, 1-NN, Naive Bayes and maximum *a posteriori* classifier assuming a Gaussian Mixture Model (MAP-GMM). We generated 100 versions of each dataset by randomly permuting its elements. On each of these permuted datasets, the first 10% of samples are kept for testing, with the remaining used to train the classifiers. Figures 3 and 4 present the average error rate on those 100 permutations along with standard deviation.

For k-NN and MAP-kDID we need to set the value of k. Thus, we performed two types of experiments: in the first one, we fixed k to be 5 (see Figure 3); on the other experiments, we run these two algorithms with $k \in \{3, 5, 9, 11, 15, 19\}$ and the best k was chosen by the best average error rate of each algorithm (see Figure 4). Also, for the MAP-GMM algorithm we trained the parameters using the Gaussian Mixture Decomposition proposed in [3]. We performed 20 runs of this algorithm and chose the best parameters using the intrinsic criterion, which is a minimum description length.

In Figure 3, MAP-5DID is better than 5-NN in most datasets. Moreover, MAP-5DID is better than or equally good as all the other algorithms in all the datasets. In particular, MAP-5NN performs very well on the German, Ionosphere, Pima, and Breast Cancer datasets. On the other hand, its performance is similar to that of other algorithms in datasets such as Iris or UCI-Segmentation; in fact, on those datasets, all algorithms yield very similar error rates.

If k for MAP-kDID and k-NN is chosen according to the smallest error rate, as in Figure 4, we notice that MAP-kDID is slightly better than k-NN. k-NN (recall that $k \in \{3, 5, 9, 11, 15, 19\}$) has improved error rates compared to 5-NN, especially in the Breast Cancer, Pima, Ionosphere and German datasets. Also, in the Austra dataset, Naive Bayes is slightly better than all the other classifiers; finally, 1-NN is, in most of the datasets, the worst classifier. In the Iris and Auto-mpg datasets, MAP-kDID is slightly better than the other classifiers, and has improved when compared to MAP-5DID.

The main conclusion from these results is that the use of the DID significantly improves the results of k-NN when k is fixed. The use of the high-order dissimilarity helps to mitigate the loss of performance from wrong choices of k, as shown by the

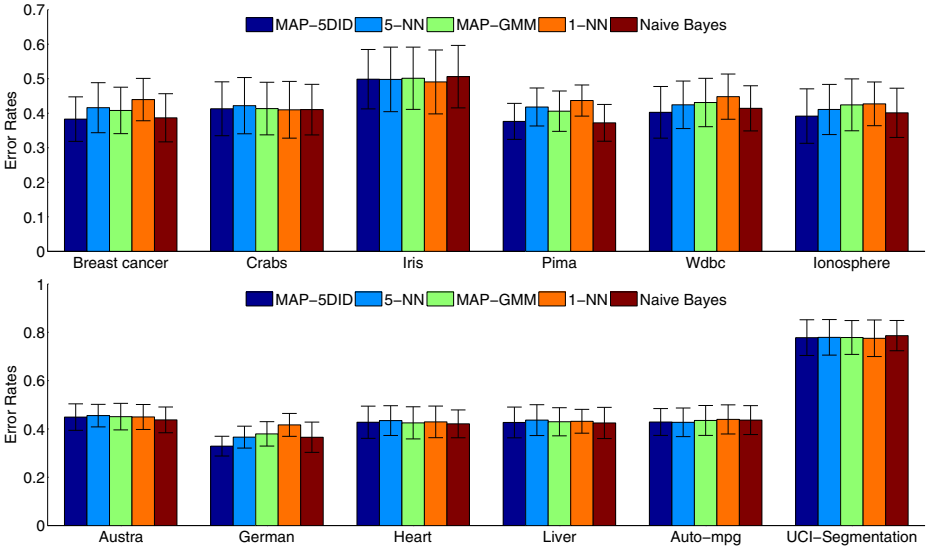

Fig. 3. Average and standard deviations of the error rate over 100 permutations of the datasets for the proposed algorithm (MAP-kDID, with $k = 5$), k-Nearest Neighbor (k-NN, with $k = 5$), Gaussian Mixture Models (MAP-GMM), 1-Nearest Neighbor (1-NN) and Naive Bayes classifier

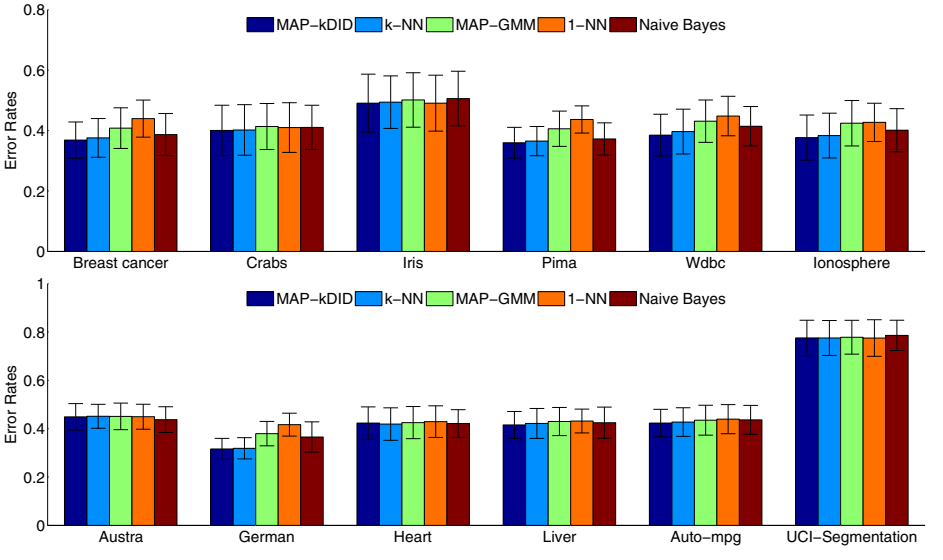

Fig. 4. Average and standard deviations of the error rate over 100 permutations of the datasets for the proposed algorithm (MAP-kDID), k-Nearest Neighbor (k-NN), Gaussian Mixture Models (MAP-GMM), 1-Nearest Neighbor (1-NN) and Naive Bayes classifier. MAP-kDID and k-NN were run for $k \in \{3, 5, 9, 11, 15, 19\}$ and the best value was chosen by the best average error rate of each method.

significant difference between MAP-5DID and 5-NN in Figure 3. If one knows the true value of k, or can estimate it as we did in Figure 4, the improvement is less noticeable.

4 Conclusions

We proposed a new Bayesian classifier, MAP-kDID, which is a maximum *a posteriori* decision rule using the k-Nearest Neighbor (k-NN) density estimation and the distribution of a high order dissimilarity, called dissimilarity increments.

Experimental results show that the use of the dissimilarity increments distribution (DID) improves the performance of k-NN, and that MAP-kDID is equally good as or better than other classifiers. In particular, the use of the DID improves the performance of k-NN in cases where k is unknown or poorly chosen.

Acknowledgments. We acknowledge financial support from the Portuguese Foundation for Science and Technology (FCT) grant PTDC/EIA-CCO/103230/2008 and partially by the scholarship number SFRH/BD/39642/2007.

References

1. Aidos, H., Fred, A.: Statistical modeling of dissimilarity increments for d-dimensional data: Application in partitional clustering. Pattern Recognition (2012), http://dx.doi.org/10.1016/j.patcog.2011.12.009
2. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. John Wiley & Sons Inc. (2001)
3. Figueiredo, M.A.T., Jain, A.K.: Unsupervised learning of finite mixture models. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(3), 381–396 (2002)
4. Fred, A., Leitão, J.: A new cluster isolation criterion based on dissimilarity increments. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(8), 944–958 (2003)
5. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd edn. Springer Series in Statistics. Springer (2009)
6. Jain, A.K., Duin, R.P.W., Mao, J.: Statistical pattern recognition: A review. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(1), 4–37 (2000)
7. Theodoridis, S., Koutroumbas, K.: Pattern Recognition, 4th edn. Elsevier Academic Press (2009)