ARTÍCULO

A meta-learning framework for pattern classification by means of data complexity measures

J. M. Sotoca, R. A. Mollineda, J. S. Sánchez

Dept. de Llenguatges i Sistemes Informàtics, Universitat Jaume I
Av. Sos Baynat s/n, 12071 Castelló de la Plana, Spain
{sotoca,mollined,sanchez}@uji.es

Abstract

It is widely accepted that the empirical behavior of classifiers strongly depends on available data. For a given problem, it is rather difficult to guess which classifier will provide the best performance or to set a proper expectation on classification performance. Traditional experimental studies consist of presenting accuracy of a set of classifiers on a small number of problems, without analyzing why a classifier outperforms other classification algorithms. Recently, some researchers have tried to characterize data complexity and relate it to classifier performance. In this paper, we present a general meta-learning framework based on a number of data complexity measures. We also discuss the applicability of this method to several problems in pattern analysis.

Key words: Data Complexity, Meta-learning, Classification, Prototype Selection, Feature Selection.

1 Introduction

Pattern classification is a growing field with applications in very different areas such as speech and handwriting recognition, computer vision, image analysis, marketing, data mining, medical science, and information retrieval, to name a few.

In brief, pattern classification constitutes a subdiscipline of Pattern Recognition devoted to extracting relevant information from data by identifying meaningful patterns. A pattern can be represented by an ordered set of n variables as the single vector $x = \{x_1, x_2, \ldots, x_n\}$. Each pattern belongs to one of C possible classes or categories, denoted as $y_k c$. Thus we have $x \in X \subseteq R^n$ as the input pattern space and $y \in Y = \{y_1, y_2, \ldots, y_c\}$ as the output class space. Therefore, pattern classification can be regarded as a function $d : X \rightarrow Y$, which assigns an output class label $y_k c$ to each input pattern $x \in X$.

The main problem of pattern classification refers to the capability of the learned classifier to generalize, that is, correctly classify unseen patterns. This problem is very hard to put in a theoretical setting and most common approaches are to a large extent heuristic in nature.

Typically, classification rules are established from randomly selected training instances from each class and are applied to test samples to evaluate their classification accuracy. In such a situation, performance of each classifier is closely related to the characteristics of the data. Consequently, an analysis of data characteristics appears to be an essential tool for selecting the appropriate classification algorithm in a particular problem.

Only few works relating the performance of classifiers to data characteristics have been carried

out up to now [1, 7, 9, 13]. The general idea consists of predicting the applicability and performance of a classifier based upon certain data characteristics. To this end, one could employ a set of data complexity measures, usually concerning statistical, geometrical and information theoretic descriptions.

In this paper, we review a number of data measures existing in the literature and discuss how they can be utilized as a meta-learning method in the domain of pattern classification. Meta-learning can be loosely defined as learning of meta-knowledge about learned knowledge. In our work, we concentrate on learning from data complexity measures and how to employ them in various pattern classification problems.

We conclude that the meta-analysis of data characteristics could become especially useful when working with very large databases (for instance, in data and web mining applications). In this context, one could estimate the utility of a classifier for a particular problem by simply computing a number of complexity measures on the training data, instead of experimenting with it.

From now on, the rest of the present paper is organized as follows. Section 2 described several measures of data complexity. Section 3 discusses a number of practical problems where the meta-learning method can be employed. Finally, Section 4 provides the main conclusions and future directions of research.

2 Data complexity measures

As already mentioned, the behavior of classifiers is strongly dependent on data complexity. Usual theoretical analysis consists of searching accuracy bounds, most of them supported by impractical conditions. Meanwhile, empirical analysis is commonly based on weak comparisons of classifier accuracies on a small number of unexplored data sets.

Such studies usually ignore the particular statistical and geometrical descriptions of class distributions to explain classification results. Various recent papers [1, 6, 11, 13] have introduced the use of measures to characterize the data complexity and to relate such descriptions to classifier performance.

Most of the data measures discussed in this paper are defined only for two-class discrimination, although in many cases it is possible to generalize them for the C-class problem. Next sections describe a number of measures selected from various papers.

A natural measure of a problem difficulty (or complexity) is the error rate associated to a given classifier. However, it can result important to employ other measures that are less dependent on the classifier chosen. Moreover, these alternative measures could be useful as a guide to select a particular classifier for a given problem.

2.1 Measures of overlap

These measures mainly focus on the effectiveness of a single feature dimension in separating the classes. They examine the range and spread of values in the data set with respect to each feature, and check for overlaps among different classes.

2.1.1 Fisher's discriminant ratio (F1)

The plain version of this well-known measure computes how separated are two classes according to a specific feature.

$$F1 = \frac{(m_1 - m_2)^2}{\sigma_1^2 + \sigma_2^2} \tag{1}$$

where m_1, m_2, σ_1^2, and σ_1^2 are the means of the two classes and their variances, respectively.

A possible generalization for C classes, which also considers all feature dimensions, can be stated as follows [10]:

$$F1_{gen} = \frac{\sum_{i=1}^{C} n_i \cdot \delta(m, m_i)}{\sum_{i=1}^{C} \sum_{j=1}^{n_i} \delta(x_j^i, m_i)} \tag{2}$$

where n_i denotes the number of samples in class i, δ is a metric (generally, the Euclidean distance), m is the overall mean, m_i is the mean of class i, and x_j^i represents the sample j belonging to class i.

2.1.2 Volume of overlap region (F2)

This measure computes, for each feature f_k, the length of the overlap range normalized by the length of the total range in which all values of both classes are distributed. Then the volume of the overlap region for two classes is obtained as the product of normalized lengths of overlapping ranges for all features.

$$F2 = \prod_k \frac{\min\max_k - \max\min_k}{\max\max_k - \min\min_k} \qquad (3)$$

where $k = 1, \ldots, d$ for a d-dimensional problem, and

$$\begin{aligned}
\min\max_k &= \min\{\max(f_k, c_1), \max(f_k, c_2)\} \\
\max\min_k &= \max\{\min(f_k, c_1), \min(f_k, c_2)\} \\
\max\max_k &= \max\{\max(f_k, c_1), \max(f_k, c_2)\} \\
\min\min_k &= \min\{\min(f_k, c_1), \min(f_k, c_2)\}
\end{aligned}$$

A very simple generalization of F2 for the C-class problem can be obtained by summing the plain measure for all possible pairs of classes [10]:

$$F2_{gen} = \sum_{(c_i, c_j)} \prod_k \frac{\min\max_k - \max\min_k}{\max\max_k - \min\min_k} \qquad (4)$$

where (c_i, c_j) goes through all pairs of classes, $k = 1, \ldots, d$, and

$$\begin{aligned}
\min\max_k &= \min\{\max(f_k, c_i), \max(f_k, c_j)\} \\
\max\min_k &= \max\{\min(f_k, c_i), \min(f_k, c_j)\} \\
\max\max_k &= \max\{\max(f_k, c_i), \max(f_k, c_j)\} \\
\min\min_k &= \min\{\min(f_k, c_i), \min(f_k, c_j)\}
\end{aligned}$$

2.1.3 Feature efficiency (F3)

In high dimensional problems, it is important to know how the discriminatory information is distributed across the features. In this context, it has to be used a measure of efficiency of individual features that describes how much each feature contributes to the separation of the two classes [4].

We can use a procedure that progressively removes unambiguous points falling outside the overlapping region in each dimension. The efficiency of a feature is defined as the fraction of all remaining points that can be separated by that feature. For a two-class problem, the maximum feature efficiency (that is, the largest fraction of points distinguishable by using only one feature) is taken as a measure of overlap.

The generalization for C classes can be defined as the overall fraction of points in some overlap range of any feature for any pair of classes. Obviously, points in more than one range are counted once.

2.2 Measures of class separability

These measures evaluate to what extent two classes are separable by examining the existence and shape of the class boundary.

2.2.1 Probabilistic distance measures

The *Bayes* error is supposed to be theoretically the best estimate to describe class separability. However, it is difficult to use in practice because of its computational complexity and it is often empirically rather than analytically derived. In these situations, a number of statistical probability distances such as *Bhattacharya, Chernoff, Mahalanobis, Matusita*, etc. provide upper and lower bounds for the error as a special case for a two-class problem [11].

2.2.2 Linear separability (L1, L2)

The linear separability is the maximum probability of correct classification when discriminating the pattern distribution with hyperplanes. In two-class problems, it represents the probability of overlapping if each class is distributed in a convex region.

Linear classifiers can be obtained by a linear programming formulation proposed by Smith [12] that minimizes the sum of distances of error points to the separating hyperplane (subtracting a constant margin).

$$\begin{array}{ll}
\text{minimize} & \mathbf{a^t t} \\
\text{subject to} & \mathbf{Z^t w + t \geq b} \\
& \mathbf{t \geq 0}
\end{array}$$

where $\mathbf{a}$, $\mathbf{b}$ are arbitrary constant vectors, $\mathbf{w}$ is the weight vector, $\mathbf{t}$ is an error vector, and $\mathbf{Z}$ is

a matrix where each column $\mathbf{z}$ is defined on an input vector $\mathbf{x}$ and its class c (with value c_1 or c_2) as follows:

$\mathbf{z} = +\mathbf{x}$ if $c = c_1$
$\mathbf{z} = -\mathbf{x}$ if $c = c_2$

The value of the objective function is used in [6] as a class separability measure (L1). It is zero for a linearly separable problem. It is to be noted that this measure can be heavily affected by the presence of outliers in the data set.

On the other hand, a second measure (L2) simply corresponds to the error rate of such a linear classifier (that defined for L1) on the original training set.

2.2.3 Fraction of points on boundary (N1)

Friedman and Rafsky [3] proposed a test on whether two samples are from the same distribution. It is thus useful for deciding whether the points labelled as two different classes form separable distributions. This method is based on the construction of a Minimum Spanning Tree (MST), connecting all points in the data set to their nearest neighbors. Then it counts the number of points connected to the opposite class by an edge in the MST. These points are considered to be close to the class boundary.

N1 is computed as the fraction of such points on boundary over the total number of points in the data set.

2.2.4 Non-parametric separability of classes (N2, N3)

The first measure (N2) [6] is the ratio of the average distance to intraclass nearest neighbor and the average distance to interclass nearest neighbor. It compares the intraclass dispersion with the interclass separability. Smaller values suggest more discriminant data.

Let $\mathcal{N}_1^{=}(x_i)$ and $\mathcal{N}_1^{\neq}(x_i)$ be the intra-class nearest neighbor and the inter-class nearest neighbor of a given example (x_i, ω_i), respectively. Then, N2 can be computed as follows:

$$\mathrm{N2} = \frac{\sum_{i=1}^{n} \delta(\mathcal{N}_1^{=}(x_i), x_i)}{\sum_{i=1}^{n} \delta(\mathcal{N}_1^{\neq}(x_i), x_i)} \tag{5}$$

The proximity of points in opposite classes affects the error rate of a nearest neighbor classifier. Thus, N3 simply corresponds to the estimated error rate of the nearest neighbor decision rule by the leaving-one-out method.

2.3 Measures of geometry and density

These measures are intended to describe the geometry or the shapes of the manifolds spanned by each class.

2.3.1 ϵ-Neighborhoods (T1)

This measure counts the number of balls needed to cover each class, being each ball centered at a training point and grown to the maximal size (in units of ϵ) before it reached a point from another class [6]. Redundant balls lying completely in the interior of other balls are removed. This count is then normalized by the total number of points.

This provides an interior description rather than a boundary description as given by the MST-based measures (see Section 2.2.3).

2.3.2 Average number of points per dimension (T2)

It has to be noted that this measure contributes to understand the behavior of some classification problems. Thus T2 describes the density of spatial distributions of samples by computing the number of instances in the data set over the number of feature dimensions.

$$\mathrm{T2} = \frac{n}{d} \tag{6}$$

where n is the number of points and d is the dimensionality of the feature space.

2.3.3 Density (D1)

This density measure can defined as the average number of samples per unit of volume where all points are distributed [10]. This volume is the product of the lengths of all feature ranges where values are spanned across all classes.

Note that D1 presents two extreme cases when a moderate number of points is described by a (relative high) number of features which either varies from 0 to 1, or takes values greater than 1.

2.3.4 Volume of local neighborhood (D2)

This measure represents the average volume occupied by the k nearest neighbors of each training instance. Let $\mathcal{N}_k(x_i)$ be the set of the k nearest neighbors of a given example (x_i, ω_i), then the volume of this can be defined as follows:

$$\mathcal{V}_i = \prod_{h=1}^{d} (\max(f_h, \mathcal{N}_k(x_i)) - \min(f_h, \mathcal{N}_k(x_i))) \tag{7}$$

where $\max(f_h, \mathcal{N}_k(x_i))$ and $\min(f_h, \mathcal{N}_k(x_i))$ represent the maximum and minimum values of feature f_h among the k nearest neighbors of instance x_i.

From this, the volume of local neighborhood can be expressed as the average value of $\mathcal{V}_i$ for the n training instances.

$$\text{D2} = \frac{1}{n} \sum_{i=1}^{n} \mathcal{V}_i \tag{8}$$

2.3.5 Class density in overlap region (D3)

The aim of this measure is to determine the density of each class in the overlap regions. In general, overlap regions contain the most critical cases for the classification task and accordingly give rise to most classifier errors. Taking into account this, in the present paper we propose a new measure of class density in overlap regions, namely D3, which is based on the well-known Wilson's editing [15].

D3 can be measured by counting, for each class, the number of points lying in the region of some different class. To this end, we first find the k nearest neighbors of each example (x_i, ω_i). Then if a majority of these k neighbors belong to a class different from ω_i, we can consider that (x_i, ω_i) lies in an overlap region. Note that the higher the value of D3 for a given class, the lower the number of examples from such a class in the overlap region.

2.3.6 Nonlinearity (L3, N4)

Hoekstra and Duin [8] proposed a measure for the *nonlinearity* of a classifier with respect to a given data set. Given a training set, this method first generates a test by linear interpolation between randomly drawn pairs of points belonging to the same class. Then, the error rate of the classifier on such a test set is measured.

In [6], both the nonlinearity of the linear classifier (L3) and that of the nearest neighbor classifier (N4) are considered.

2.4 Statistical measures

In Statlog project, several classification techniques were compared over 22 data sets. These sets were described in terms of various statistics, trying to predict the applicability of a classifier based on certain data characteristics. (Statlog is an acronym for an ESPRIT project (1990-1993) involved in comparative testing of statistical and logical machine learning algorithms).

Among others, the following descriptive and multivariate statistics were used to summarize the data sets in the Statlog project: total number of instances in the whole data set, number of training patterns, number of patterns used for test, number of features, number of binary attributes, number of classes, mean absolute correlation coefficients between two features, mean skewness of features, mean kurtosis of features, entropy of classes, average entropy of discrete features, and mutual information of class and feature.

All these and many other descriptive and statistical measures could be still applied to data characterization as a tool for predicting the most appropriate classifier on a particular problem, although the reliability of the predictions can be rather questionable.

3 Meta-learning from data complexity measures

The following sections discuss some possible applications of data complexity measures in the general framework of pattern classification. We review several recent works in which those measures have been employed with different aims.

3.1 Meta-analysis of classifiers

In the last years, several researchers [1, 2, 5, 13, 14] have attempted to perform a meta analysis of classification algorithms. The aim is that given a data set with known characteristics, one can derive a number of meta-rules for providing practical guidelines in efficient classifier selection.

For instance, Sohn [13] describes a total of 19 data characteristics and performs a regression analysis between the error rate of eleven classifiers (including statistical, machine learning and neural networks) and those data measures.

Bernardó and Ho [1] firstly define a space of nine data complexity measures and compute the complexity measures for each problem using all available data points. Then they look for regions in the complexity space where each classifier is significantly better than the others, and regions where multiple classification methods score similarly. In their empirical study, they evaluate six classifiers.

3.2 Prototype selection

Prototype selection consists of selecting an appropriate reduced subset of patterns from the original training set and applying the nearest neighbor rule using only the selected examples. Two different families of prototype selection methods exist in the literature: editing and condensing algorithms.

While editing approaches eliminate erroneous patterns from the original set and "clean" possible overlapping between regions from different classes (note that these usually leads to significant improvements in performance), condensing aims at selecting a sufficiently small set of training patterns that produces approximately the same performance than the nearest neighbor rule using the whole training set.

Singh [11] employs several data complexity measures to remove outliers from a training set and also points out that another utility would be to help reduce the data set size without compromising on the test performance of classifiers. More specifically, those patterns that are found deep inside class boundaries could be removed from the training set since they are least likely to help in classification of test samples.

On the other hand, Mollineda et al. [10] investigate on the utility of a set of complexity measures as a tool to predict whether or not the application of some prototype selection (editing and/or condensing) algorithm could result appropriate in a particular problem. They test different data complexity measures using 17 databases and derive a number of practical situations under which prototype selection is suited and in this case, which is the algorithm most appropriate.

3.3 Feature selection

Attribute or feature selection consists of picking, out of all attributes potentially available for the classification algorithm, a subset of features relevant for the target task. One important motivation for feature selection is to minimize the error rate. Indeed, the existence of many irrelevant/redundant attributes in a given data set may "confuse" the classifier, leading to a high error rate.

There are two major approaches to feature selection, namely the wrapper and the filter approaches. In the wrapper approach, the data set is divided into two subsets: the training subset and the evaluation subset. Then a heuristic search is done in the space of subsets of attributes. In this search, the quality of a subset of attributes is computed in two steps. Firstly, the classifier itself is trained on the training subset by using only the subset of attributes being evaluated. Secondly, the error rate of the discovered rules on the evaluation subset is measured and it is directly used as a measure of the quality of the feature subset being evaluated.

In contrast, in the filter approach the quality of a given subset of attributes is evaluated by some method that does not employ the target classification algorithm. Typical functions used in the filter approach include distance measures,

consistency measures, dependency measures, and information-theoretic measures.

Most of the measures defined in Section 2 (especially those related to class separability) can be employed in a filter approach as an effective optimization criterion to obtain the best feature subset. For example, Singh [11] conducts an empirical study to demonstrate that the employment of a neighborhood separability measure constitutes a suitable criterion for optimization in feature selection for a face recognition problem.

4 Concluding remarks

This paper provides a general meta-learning framework for pattern classification problems, and it is based on the employment of a number of data complexity measures found in the literature and discusses three important application fields belonging to the domains of pattern classification and learning.

The measures here described correspond to four categories: measures of overlap, measures of class separability, measures of geometry and density, and statistical measures. However, other measures could be still applied to characterize data complexity.

We point out three application areas where those measures have already been successfully employed: meta analysis of classifiers, prototype selection, and feature selection. Alternative utilities can be devised in other domains. In this sense, data complexity measures could be useful for classifier selection in a context of classifier fusion. In fact, this constitutes one of the most important direction for our future research.

In the context of pattern classification and learning, we think that data complexity measures can be especially relevant for applying to very large data sets (for example, in web and data mining problems). Suppose we have a very large training data set and a number of classifiers. We have to choose the most appropriate classifier for the problem in hand. To this end, one could test all the classifiers on the (large) training data set and then select the one with the highest accuracy. However a better alternative (in the sense of less computing time and independence of the particular problem) would consist of describing the problem in terms of data complexity and then pick up the most suitable classifier according to the particular characteristics of the problem.

Acknowledgements

This work has been supported in part by grants TIC2003-08496 from the Spanish CICYT (Dept. of Science and Technology), GV04A-705 from Generalitat Valenciana, and P1-1B2004-08 from Fundació Caixa Castelló-Bancaixa. We would like to thank the anonymous Reviewers for their valuable and constructive remarks.

References

[1] E. Bernardó and T.-K. Ho. On classifier domain of competence, In: *Proc. 17th. Int. Conference on Pattern Recognition*, Cambridge, UK, pages 136–139, 2004.

[2] P.-K. Chan and S.J. Stolfo. On the accuracy of meta-learning for scalable data mining, *Journal of Intelligent Information Systems*, 8:5–28, 1997.

[3] J.H. Friedman and L.C. Rafsky Multivariate generalizations of the Wald-Wolfowitz and Smirnov two-sample tests, *Tha Annals of Statistics*, 7:697–717, 1979.

[4] T.-K. Ho and H.S. Baird. Pattern classification with compact distribution maps, *Computer Vision and Image Understanding*, 70:101–110, 1998.

[5] T.-K. Ho. Data complexity analysis for classifier combination, In: *Proc. 2nd. Int. Workshop on Multiple Classifier Systems*, Cambridge, UK, pages 53–67, 2001.

[6] T.-K. Ho and M. Basu. Complexity measures of supervised classification problems, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24:289–300, 2002.

[7] T.-K. Ho. A data complexity analysis of comparative advantages of decision forest constructors, *Pattern Analysis and Applications*, 5:102–112, 2002.

[8] A. Hoekstra and R.P.W. Duin. On the nonlinearity of pattern classifiers, In: *Proc. 13th. Int. Conference on Pattern Recognition*, Vienna, Austria, pages 271–275, 1996.

[9] Y. Lee and K.-W. Hwang. Selecting good speech features for recognition, *ETRI Journal*, 18:29–41, 1996.

[10] R.A. Mollineda, J.S. Sánchez, and J.M. Sotoca. Data characterization for effective prototype selection, In: *Proc. 2nd. Iberian Conference on Pattern Recognition and Image Analysis*, Estoril, Portugal, pages 27–34, 2005.

[11] S. Singh. PRISM — A novel framework for pattern recognition, *Pattern Analysis and Applications*, 6:134–149, 2003.

[12] F.W. Smith. Pattern classifier design by linear programming, *IEEE Trans. on Computers*, 17:367–372, 1968.

[13] S.-Y. Sohn. Meta analysis of classification algorithms for pattern recognition, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 21:1137–1144, 1999.

[14] L. Todorovski and S. Dzeroski. Combining classifiers with meta decision trees, Machine Learning, 50:223–249, 2003.

[15] D.L. Wilson. Asymptotic properties of nearest neighbor rules using edited data sets, *IEEE Trans. on Systems, Man and Cybernetics*, 2:408–421, 1972.