On data clustering with artificial ants

N. Monmarché
Laboratoire d'Informatique de l'Université de Tours,
Ecole d’Ingénieurs en Informatique pour l'industrie (E3i),
64 av. Jean Portalis, 37200 Tours, FRANCE
monmarche@univ-tours.fr
Phone: +33-2-47-36-14-14
Fax: +33-2-47-36-14-22

Abstract

We present in this paper a new ant based approach named AntClass for data clustering. This algorithm uses the stochastic principles of an ant colony in conjunction with the deterministic principles of the Kmeans algorithm. It first creates an initial partition using an improved ant-based approach, which does not require any information on the input data (such as the number of classes, or an initial partition). Then it uses the Kmeans to speed up the convergence of the stochastic approach. In a second phase, AntClass uses hierarchical clustering where ants may cluster together heaps of objects and not just objects. We also use an heterogeneous population of ants in order to avoid complex parameter settings. We show on typical benchmark databases that AntClass is competitive with other approaches such as the Kmeans or ISODATA.

Introduction

We consider in this paper the problem of unsupervised data classification, where clusters must be found in a numerical data set. Many standard approaches like the Kmeans or ISODATA are limited because they generally require the a priori knowledge of a probable number of classes and an initial partition. Furthermore, they also use heuristic principles which are often locally optimal.

Among the approaches that can be used to improve those standard methods, one may for instance mention Bayesian classification with AutoClass (Cheeseman and Stutz 1996), genetic-based approaches (Jones and Beltrano 1991) (Cucchiara 1993) and ant-based approaches (Deneubourg et al. 1990) (Lumer and Faieta 1994) (Kuntz and Snyers 1994) (Kuntz et al. 1997). In AutoClass, the most probable classifications are searched by the domain expert needs to perform complex parameter settings. In GA-based approaches, an individual represents a possible partition of the data set. The GA thus manages an explicit partition, and evaluates this partition with a global evaluation function. The representation of the partition can be done with permutations as in (Jones and Beltrano 1991), or with a direct binary encoding of the objects membership as in (Cucchiara 1993). GAs can also be used to provide an initial partition to the Kmeans algorithm (Babu and Murty 1993). As will be seen in the following, the ant based approach is quite different because the evolved partition is not represented as a centralized individual but is rather distributed. There is no global evaluation function as in GAs to which each ant would have access. Using artificial ants for clustering, instead of GAs for instance, is quite sensible because this is one of the problems that real ants have to solve. Real ants naturally cluster together eggs or dead bodies into heaps in a distributed way. So in some way, the artificial ant model for clustering is maybe “closer” to the clustering problem than the genetic model, and we are thus expecting it to work better and faster because it may use additional local heuristics. However, we do not provide in this paper a comparison of AntClass with a genetic approach but rather with the Kmeans and ISODATA algorithms.

In ant-based approaches, several papers have highlighted the efficiency of stochastic approaches to problems similar to data clustering. One of the first studies related to this domain is due to (Deneubourg et al. 1991), where a population of ant-like agents randomly moving onto a 2D grid are allowed to move basic objects so as to classify them. This method has been further developed by (Lumer and Faieta 1994), with simple objects that represent records in a numerical data set, and then by (Kuntz and Snyers 1994) (Kuntz et al. 1997) where a real clustering problem is studied in order to efficiently resolve an optimization problem.

Based on this existing work, we contribute in this paper to the study of clustering ants from the knowledge discovery point of view, with the aim of solving real world problems. We improve the work of Lumer and Faieta in several ways, such as introducing more robust ant-like heuristics, dealing with “unassigned objects”, speeding up convergence with the Kmeans algorithm, using hierarchical clustering on heaps of objects, testing the resulting algorithm on several real world data sets and providing a successful comparison with the Kmeans and ISODATA algorithms. Due to limited space, we will only present here the main principles of AntClass, but the interested reader may refer to (Monmarché et
algorithm but on heaps of objects rather than single
contains too many but very homogenous clusters.
it with too many clusters, the obtained partition still
Kmeans is locally optimal only and since we provide
removes really the classification errors, but since the
still carried by the ants when the algorithm stops. This
convergence which would be too long to obtain.
are quite homogenous) and with some obvious classi-
is however compound of too many clusters (but which
does not require any information such as the number
database because it has the following advantage: it
consists of too many objects, a heap being a group of
where the examples are generated according to
gaussian laws (in the same way as Lumer and Faieta).
properties where the examples are generated according to
gaussian laws (in the same way as Lumer and Faieta).
were objects. We use the same ant-based algorithms as
previously mentioned, but adapted for heaps. For in-
stance, one can define a distance between too heaps as
the distance between their center of mass. This part
of AntClass can be seen as hierarchical clustering: the
ants first work on the objects, constituting small but
very homogenous heaps. Then, working directly on
these heaps as if they were objects, they will hierar-
chically build more important classes. At the end of
this step, the real number of classes is very well ap-
proximated, but as mentioned previously, there are still
some heaps which are not assigned. Therefore we use
once more the Kmeans algorithm to obtain the final
partition. But this time, since the input partition given
to the Kmeans is very close to the "optimal" one, the
output is of high quality.
So AntClass consists mainly in four steps: (1) ant-
based algorithm for clustering objects, followed by (2)
the Kmeans algorithm using the initial partition pro-
vided by the ants, and then (3) ant-based clustering but
on heaps previously found, and finally (4) the Kmeans
algorithm on objects once more.

Results and conclusion
For all the databases hereafter mentioned, the results
are obtained in a about 10 seconds on a standard PC
(Pentium 166MHz) for one run. All results have been
averaged over 50 runs. We have used the following
databases (numbers in brackets indicate respectively,
the number of objects, the number of numerical at-
tributes, and the number of classes): Artif. 1 (80, 2,
4), Artif. 2 (270, 2, 9), Artif. 3 (200, 2, 4), Artif. 4
(150, 10, 3), Iris (178, 4, 3)(Fisher 1936), Wine (178,
13, 3), Glass (214, 9, 2-6), Soybean (47, 21, 4), Thyroid
( 215, 5, 3). "Artif. 1" to "Artif. 4" have been used to
evaluate AntClass on databases with known prop-
erties where the examples are generated according to
gaussian laws (in the same way as Lumer and Faieta).
The other real world databases come from the machine
learning repository. All values in the database are nor-
malized in [0, 1].
We have defined two performance measures to evalu-
ate how close is the obtained partition to the real one.
The first measure is a classification error rate. It is
computed as follows: for a given cluster \( H \) obtained
by AntClass, consider the majorative class among \( H \)
according to the "Class" attribute. All objects of \( H \)
that do not belong to this class are considered as being
missclassified. The classification error rate is simply the
ratio between the total number of misclassified objects
for all created clusters and the total number of objects
in the database. The second performance measure is
simply the number of created clusters.
The results mentioned in table 1 on the artificial
databases show the progression of AntClass towards a
relevant classification. In the first step of AntClass, the
Ant-based algorithm finds an initial partition but with
classification errors and really too many clusters. At the end of the second step of AntClass, i.e. the use of the Kmeans on the initial partition found in the previous step, the classification errors are reduced but the number of clusters is still really too high. This is due to the fact that the Kmeans algorithm is really sensitive to the initial partition. If this initial partition contains too many clusters, then the final partition is unlikely to be the optimal one. Once the third step has been performed, the ants converge very closely to the right number of classes by working on heaps of objects rather than objects themselves. One can notice that some classification errors remain. At the end of the fourth step, using the Kmeans once more decreases the classification errors on again. But this time, since the number of classes and an almost optimal partition have been well determined in the previous step, the Kmeans really finds optimal or near optimal results.

We describe now the results obtained with the Kmeans algorithm and with ISODATA (Ball and Hall 1965). Each algorithm is initialized with 10 classes, and all objects are initially assigned randomly to these classes. The Kmeans and ISODATA algorithms are used with 10 iterations. Results obtained by the three algorithms (AntClass, Kmeans and ISODATA) are represented in table 2. As can be seen, AntClass outperforms the two other algorithms, both in terms of classification errors and of correct number of classes, on most of the databases. The only exception is for Fisher's Iris database. In this database, the Setosa class is completely distinguishable from the two others. The last two classes are more difficult to separate, unless more than three clusters are created, which is what Kmeans of ISODATA do.

As a conclusion, we have presented in this paper a new hybrid and ant-based algorithm named AntClass for data clustering in a knowledge discovery context. The main features of this algorithm are the following ones. AntClass deals with numerical databases. It does not require any initial information about the future classification, such as an initial partition or an initial number of classes. AntClass introduces new heuristics for the ant colony, and also an hybridization with the Kmeans algorithm in order to improve the convergence. We have also introduced in AntClass hierarchical clustering where ants may carry heaps of objects and not just objects. Furthermore, AntClass uses an heterogeneous population of ants in order to avoid complex parameter settings to be performed by the domain expert. Finally, AntClass has been tested with success on several databases, including real world ones. We may mention also that it has been applied to a real world but confidential application and gets competitive results with SAS 6.12.

Future work consists in two ways:

- we need to define new performance measures that could take into account both the number of classes and the quality of the classification,
- other sources of inspiration from real ants are to be considered for the clustering problem. For instance, ants that meet on the board could exchange objects.

## References


Table 2: Results obtained by AntClass, Kmeans and ISODATA with artificial and real world databases.


