Exploring Different Functions for Heuristics, Discretization, and Rule Quality Evaluation in Ant-Miner

Khalid M. Salama and Fernando E. B. Otero

School of Computing, University of Kent, Canterbury, UK
{kms39@kent.ac.uk,F.E.B.Otero}@kent.ac.uk

Data mining is a process that supports knowledge discovery by finding hidden patterns, associations and constructing analytical models from databases. Classification is one of the widely studied data mining tasks in which the aim is to discover, from labelled cases, a model that can be used to predict the class of unlabelled cases. Ant-Miner, proposed by Parpinelli et al. [3], is the first ACO algorithm for discovering classification rules. Ant-Miner has been shown to be competitive with well-known classification algorithms, in terms of producing comprehensible model with high predictive accuracy. Therefore, there has been an increasing interest in improving the Ant-Miner algorithm [1].

Otero et al. [2] presented *c*Ant-Miner as a variation of the original Ant-Miner algorithm, which is able to cope with continuous-valued attributes during the rule construction process through the creation of discrete intervals on-the-fly. Salama et al. recently introduced an efficient version of the algorithm, μAnt-Miner [4], based on selecting the consequent class of the rule before constructing its antecedent and utilizing multiple pheromone types, one for each permitted rule class. This idea gives the motivation of utilizing the pre-selected class in term heuristic information calculation and continuous attribute discretization using different measure functions.

In this paper, we utilize the μAnt-Miner idea of selecting the class before the rule construction to extend *c*Ant-Miner in three essential aspects. First, we use a class-based measure function to compute heuristic information for a term. Second, we use this function as criteria to carry out the dynamic discretization of the continuous attributes and select the best created interval with respect to the pre-selected class. Third, we use the same measure function used for both previous operations to evaluate the quality of the constructed rule for the sake of pheromone update.

Since we evaluate the quality of a constructed rule with a given function f_x, there is no need to select terms that maximize another function f_y. Intuitively, the selection of terms that maximize f_x should lead to construct a high quality rule with respect to f_x. Moreover, using class-based evaluation function for heuristic information and discretization leads to the selection of terms that are relevant to the prediction of a specific class, rather than selecting terms simply to reduce the entropy among the class distribution on the dataset as in the original *c*Ant-Miner. Therefore, we use a unified quality evaluation function QEF to compute the heuristic information of a term, to create intervals from continuous attributes in the discretization, with respect to the pre-selected class value, and to evaluate the quality constructed rule as well.

First, in order to compute heuristic value for $term_{ij}$ given class k, we construct a temporary rule with only $term_{ij}$ in its antecedent and labelled with class k, and we evaluate the quality of this rule using the unified QEF.

Unlike cAnt-Miner, where the threshold value is selected only to minimize the entropy among the classes, we aim to select a threshold value that generates partitions with more relevance for predicting that class by taking the advantage of the class pre-selection. In essence, we calculate the absolute difference in quality (measured in terms of QEF) between the upper and the lower intervals for each candidate value v_i. The idea is to select the threshold value v_{best} that maximizes the quality discrimination—with respect to the current selected class value—between the two intervals.

Finally, the QEF function is used to evaluate the constructed rules, where the best rule created in the colony is used to update the pheromone.

We explore the use of 10 different functions—for heuristics information calculation, continuous attributes discretization and rule quality evaluation. The set of functions is {`Certainty Factor`, `Collective Strength`, `f-Measure`, `Jaccard`, `Kappa`, `klosgen`, `m-Estimate`, `R-Cost`, `Sensitivity` × `Specificity`, `Support + Confidence`}.

Concerning the predictive accuracy, there is no algorithm that performs absolutely best. Our results show a great diversity amongst the performance of different quality evaluation functions. This suggests that combining the measures of multiple quality evaluation functions can lead to improvements in the search of the algorithm, since the use of different measures can capture different aspects of the performance of a candidate rule and provide a more robust measure of quality across multiple datasets. Moreover, different quality evaluation functions can be used for each component of the algorithm—i.e., for heuristic, dynamic discretization and rule evaluation. These ideas present research directions worth further exploration.

References

1. Martens, D., Baesens, B., Fawcett, T.: Editorial survey: swarm intelligence for data mining. Machine Learning 82(1), 1–42 (2011)
2. Otero, F., Freitas, A., Johnson, C.: cAnt-Miner: an ant colony classification algorithm to cope with continuous attributes. In: Dorigo, M., Birattari, M., Blum, C., Clerc, M., Stützle, T., Winfield, A. (eds.) Proceedings of the 6th International Conference on Swarm Intelligence (ANTS 2008), Lecture Notes in Computer Science 5217. pp. 48–59. Springer-Verlag (2008)
3. Parpinelli, R., Lopes, H., Freitas, A.: Data mining with an ant colony optimization algorithm. IEEE Transactions on Evolutionary Computation 6(4), 321–332 (2002)
4. Salama, K., Abdelbar, A., Freitas, A.: Multiple pheromone types and other extensions to the ant-miner classification rule discovery algorithm. Swarm Intelligence 5(3-4), 149–182 (2011)