DP1: Supervised and Unsupervised Clustering

Joel D. Martin

Dept. of Computer Science, University of Pittsburgh, Pittsburgh, PA, 15260, USA

Abstract. This paper presents DP1, an incremental clustering algorithm that accepts a description of the expected performance task — the goal of learning — and uses that description to alter its learning bias. With different goals DP1 addresses a wide range of empirical learning tasks from supervised to unsupervised learning. At one extreme, DP1 performs the same task as does ID3, and at the other, it performs the same task as does COBWEB.

There is a traditional contrast between supervised and unsupervised learning (Duda & Hart, 1973). The supervised learner is told one target variable that will be important at performance. On the other hand, unsupervised learners have no such guidance and learn all predictive structure in the domain.

We present an integrated algorithm that smoothly varies its learning bias depending upon a description of the anticipated performance task. DP1 can be made to address the same task as ID3 (Quinlan, 1986), COBWEB (Fisher, 1987), or Anderson and Matessa's (1992) method (BC).

1 Expected distribution of prediction tests

A performance task is supervised if a particular variable has a special status. In such tasks, the probability that the learner will know the value of a non-target variable at prediction time is 1.0 and 0.0 for the target variable. These probabilities are the *availability probabilities* (Table 1). Conversely, the probability that the learner will have to guess the value of a non-target variable is 0.0 and is 1.0 for the target variable. These probabilities are called *goal probabilities* (Table 1).

Alternatively, a learning task is unsupervised if no variable has a special status. In other words, the probability that the learner will know a variable value at prediction time is some uniform value. Similarly, the probability that the learner will have to guess the value of a certain variable is uniform across the values.

2 The Model: DP1

The DP1[1] algorithm is an incremental concept formation method that is a descendant of COBWEB (Fisher, 1987) and BC (Anderson & Matessa, 1992). It

[1] Directed Partitioning, Version 1.

performs a beam search (with a beam size of two) to find a single set of mutually exclusive partitions that is expected to lead to highly accurate predictions. Each partition is a subset of the encountered instances. As each instance arrives, it is added to one of the existing concepts or a new concept is created. This is done for each of the two hypotheses in the beam. DP1 assigns the instance to a concept if by doing so, the measure of predictive accuracy is improved more than by assigning the instance to any other concept.

There are several novel characteristics of DP1 as compared with other incremental partitioning algorithms. Many of these result from adding supervision to a partitioning algorithm. First, instead of performing a simple hill-climbing search, DP1 uses a beam search with a beam size of two. That is, it keeps track of two alternative partitions of the instances at any time. This is less expensive than the merging and splitting used by COBWEB (Fisher, 1987). DP1 only maintains two partitionings whereas COBWEB maintains several.

Second, DP1 evaluates a partition based on the current instance and a set of test instances. The test instances provide a domain dependent measure of when the current instance is different enough to belong to a new concept. A small set of previous instances is retained to correctly reflect the dependencies between variables. In DP1 every concept keeps track of the instances that are the most and least probable examples of that concept.

DP1's algorithm is as follows:

1. The model is initialized to contain no concepts.
2. Given a model that partitions $n-1$ instances, for each concept calculate estimate of predictive accuracy when the n-th instance is classified to the concept.
3. Assign the n-th instance to the concept with the maximum estimate.
4. When predicting the value of variable i, do the following,
   (a) Classify the instance to the most probable concept given F,
   (b) Choose the most probable value given the concept.

2.1 The calculations

The accuracy score used in DP1 captures the two intuitions, a) that the probability of being correct is higher if the goal variables have highly probable values for the concept; and b) the probability of being correct is higher if the available variables have distinctive values for the concept.

The score used is equal to:

$$\sum_h \sum_k \left(\sum_i P(goal_{A_i}) P(V_{ij}|C_k)\right) \sum_l P(available_{F_{hl}}) \frac{P(C_k)P(F_{hl}|C_k)}{\sum_m P(C_m)P(F_{hl}|C_m)} \tag{1}$$

The quantities referred to are calculated as follows.

$$P(C_k|F_{hl}) = \frac{P(C_k)P(F_{hl}|C_k)}{\sum_m P(C_m)P(F_{hl}|C_m)} \tag{2}$$

Table 1. Predictive accuracy. Standard dev. in parentheses.

Dataset - Task	DP1	DP1 US	COBWEB/CLASSIT	ID3
Hepatitis	0.83 (0.06)	0.79 (0.05)	0.77 (0.05)	0.78 (0.05)
XOR-3	0.96 (0.08)	0.44 (0.02)	0.51 (0.28)	0.69 (0.36)

$$P(F_{hl}|C_k) = \prod_i P(V_{ij}|C_k) \tag{3}$$

For discrete variables:

$$P(V_{ij}|C_k) = \frac{n_{ij} + \alpha_i}{n_i + \alpha_0} \tag{4}$$

For continuous or ordered variables:

$$P(V_{ij}|C_k) = Normal(V_{ij}, \mu_{C_k}, \sigma_{C_k}) \tag{5}$$

In these equations, V_{ij} refers to the jth value of the ith variable. F_l refers to a partial instance. C_k refers to the classification of an instance to the kth concept. The quantities, α_i, are parameters of a Dirichlet distribution and α_0 is the sum over those parameters. We set all the α_i's to be 1.0 and never vary them.

3 Application to machine learning datasets

DP1 was applied to one natural domain, hepatitis, and one artificial domain called the XOR-3 domain that has three independent XOR relationships.

We ran each of five learning systems, DP1 in supervised mode, DP1 in unsupervised mode, COBWEB/CLASSIT (Gennari, Langley, & Fisher, 1990), and ID3 (Quinlan, 1986). When unsupervised, DP1 had availability probabilities set uniformly to 1.0 and goal probabilities set uniformly to 1/16. In supervised mode, DP1 had availability probabilities set to 1.0 for all variables but the 'lives' variable and goal probabilities set to 0.0 except for 'lives' variable which was 1.0.

For the hepatitis dataset, in each of ten trials, each of the systems was taught a randomly selected set of 116 (75%) instances and then tested on a separate set of 39 (25%) instances. In all the empirical studies below, testing a system on an instance meant removing a variable value from the instance, allowing the system to predict the value of that variable, and then checking if the prediction was correct. Average prediction performance for whether a patient lives or dies is shown in Table 1.

For this dataset, performance was significantly better (ANOVA, $P < .05$) for the supervised DP1 than for the unsupervised. Further, the supervised performance was significantly better than that for the other systems.

Finally, all systems were applied to the XOR-3 dataset in which there are nine

Table 2. Accuracy for all variables, XOR-3. Standard dev. in parentheses.

Dataset - Task	DP1	DP1 US
XOR-3	0.441 (0.09)	0.635 (0.05)

variables. These variables are partitioned into three sets of three variables. The values for each set of three variables are interrelated by the XOR relationship.

In each of ten trials, each system was taught a randomly selected set of 50 instances and then tested on a separate set of 14 instances. The average prediction performance for predicting the value of the first variable is shown in Table 1.

For this dataset, performance was significantly better (ANOVA, $P < .05$) for the supervised DP1 than for the unsupervised and significantly better than for any other system.

In addition, the extra work done by DP1 in unsupervised mode translates to better average prediction for all attributes. To test this, we ran DP1 in supervised and unsupervised modes, testing its ability to predict the value of any of the last six (of nine) variables. In each of ten trials, the system received 50 training instances and 14 test instances (means in Table 2).

4 Discussion

This paper presents DP1 that can perform both supervised and unsupervised learning and introduces a notation to describe the expected distribution of prediction tests. DP1 performs best when it expects the performance task that it later receives.

Bibliography

Anderson, J. R. & Matessa, M. (1992). Explorations of an incremental, Bayesian algorithm for categorization. *Machine Learning, 9*, 275–308.

Duda, R. O. & Hart, P. E. (1973). *Pattern classification and scene analysis.* New York: Wiley & Sons.

Fisher, D. H. (1987). Knowledge acquisition via incremental conceptual clustering. *Machine Learning, 2*, 139–172.

Gennari, J., Langley, P., and Fisher, D. (1990). Models of incremental concept formation. *Artificial Intelligence, 40*, 11–61.

Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning, 1*, 81–106.