LEARNING SOIL CLASSIFICATION USING CFP

Hakime Gülay Ünsal and **H. Altay Güvenir**
Department of Computer Engineering and Information Science,
Bilkent University, Bilkent, 06533 Ankara
{unsal,guvenir}@cs.bilkent.edu.tr

Abstract- The paper presents an application of the Classification by Feature Partitioning (CFP) algorithm to the problem of soil classification. CFP is an exemplar based, incremental and supervised learning algorithm. Learning in CFP is accomplished by storing the objects separately in each feature dimension as disjoint partitions of values. Application of the CFP algorithm to soil classification, one of the important problems of soil engineering and civil engineering, is described. The classification helps the soil engineer by giving general guidance. In many soil engineering problems there are no rational expression available for the analysis for the solution.

1. INTRODUCTION

The paper presents an application of the Classification by Feature Partitioning (CFP) algorithm to the problem of soil classification. CFP is an exemplar based, incremental and supervised learning algorithm. In exemplar-based learning examples are stored in memory verbatim. The CFP technique makes several significant improvements over other exemplar-based learning algorithms. The CFP method stores the instances as factored out by their feature values. The CFP partitions each feature into segments corresponding to concepts. Therefore, the concept description learned by the CFP is a collection of feature partitions. In other words, the CFP learns a projection of the concept on each feature dimensions.

Each feature contributes the classification process by its local knowledge. Final classification is based on a voting among the predictions of the features. The CFP algorithm significantly reduces the classification complexity, over other exemplar-based techniques. The strength of the contribution of a feature in the voting process is determined by the weight of that feature.

Classification of soil plays an important role in many civil engineering problems. Relative ability of expert systems and numerical classification to improve soil classification systems are discussed in [2]. Dale et. al. conclude that numerical classification has a potentially useful part to play in establishing soil classes and generating rules for assignment in expert systems. Here we show that CFP can be used to learn an appropriate representation for soil classification in the form of feature partitions.

The rest of the paper describes the details of the CFP algorithm, the process of soil classification, and the application of CFP to a particular soil data set.

2. THE CFP ALGORITHM

An instance is defined as a vector of feature values plus a label that represents its class. The CFP algorithm learns partitions of the set of possible values for each feature. For each partition, lower and upper bounds of the feature values, its associated class and the number of instances it represents are maintained. Initially, a partition is a point (lower and upper limits are equal) on the line representing the feature dimension. A partition can be extended through generalization with other neighboring points in the same feature dimension. Classification is carried out according to a voting scheme where each feature contributes a weighted vote.

The training process of the CFP algorithm has two steps: learning feature weights and learning feature partitions. For each training example, the prediction of each feature is compared with the actual class of the example. If the prediction of a feature is correct, the weight of that feature is incremented by Δ (global feature weight adjustment rate); otherwise, it is decremented by the same amount.

The second step in the training process is to update the partitioning of each feature-space using the given training example. If the feature value of a training example falls in a partition with the same class, then simply its representativeness value (number of instances represented by that partition) is incremented. If the new feature value falls in a range partition with a different class than that of the example, the CFP algorithm specializes the existing partition by dividing it into two subpartitions and inserting a point partition (corresponding to the new feature value) in between them. On the other hand, if the example falls in an undetermined partition, the CFP algorithm tries to generalize a near partition with the feature value. If one of the nearest partitions to the left and the right of the new example is in D_f (generalization limit) distance and of the same class as the example, then it is generalized to cover the new feature value. Otherwise, a new point partition that corresponds to the new feature value, is inserted [7].

A version of CFP called GACFP has been implemented to learn these parameters of the CFP using a genetic algorithm [4].

No similarity and distance metric is used for prediction in CFP. Prediction process is performed according to local knowledge of each feature. The classification process of the CFP is based on a voting taken among the predictions made by each feature separately. For a given instance e, the prediction based on a feature f is determined by the value of e_f. If e_f falls properly within a partition with a determined class then the prediction is the class of that partition. If e_f falls into the border of more than one partitions, then among all the partitions at this point the one with the highest representativeness value is chosen. If e_f falls in a partition with no known class value, then no prediction for that feature is made. The effect of the prediction of a feature in the voting is proportional with the weight of that feature. All feature weights are initialized to one, before the training process begins. The predicted class of a given instance is the one which receives the highest amount of votes among all predictions. Figure 1 illustrates

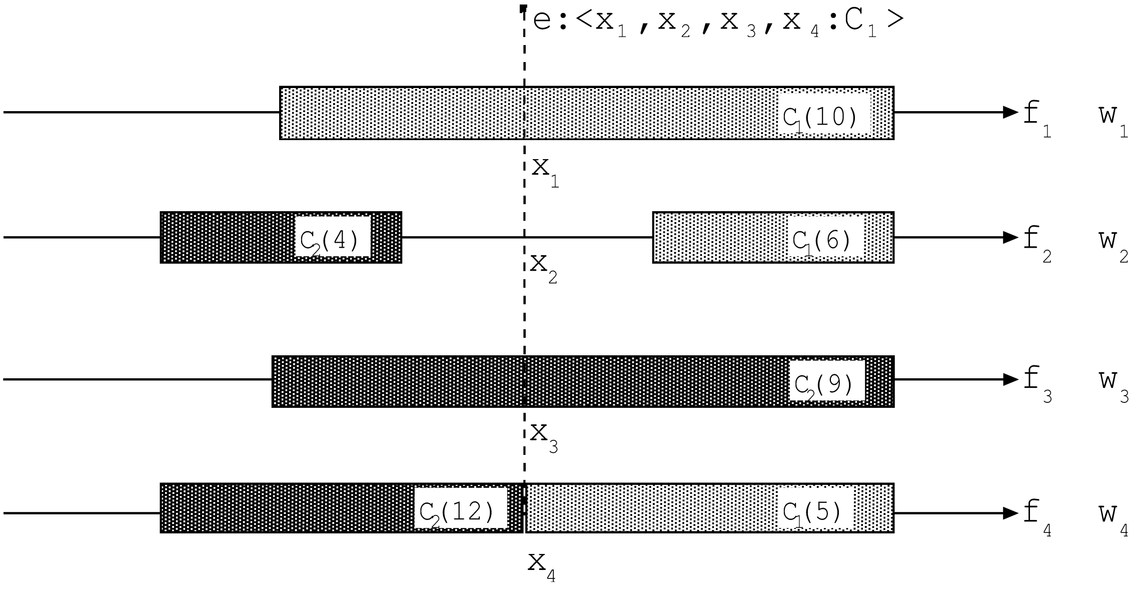

Figure 1. Voting in the classification.

the voting mechanism used in the classification through an example. Consider a test example e of class C_1 with feature values x_1, x_2, x_3, and x_4. The prediction of the first feature is C_1. The second feature does not predict any class value (*undetermined*). The prediction of the third feature is C_2. The forth feature value x_4 of e falls into the border of two partitions. In this case the representativeness values are used to determine the class value. Since the partition of class C_2 has a greater representativeness value than that of C_1 partition, the prediction of the forth feature is C_2. Final prediction of the CFP depends on the values of the feature weights (w_f's). If $w_1 > (w_3 + w_4)$ then CFP will classify e as a member of C_1; otherwise the prediction would be C_2.

The *sample complexity* and *training complexity* analysis of the CFP algorithm with respect to PAC-learning theory (Probably Approximately Correct learning) shows that, it requires small number of examples and a small amount of memory to learn a given concept, compared to many other similar algorithms [3]. Another outcome of this analysis is that, the CFP has a low learning complexity. Empirical evaluation of the CFP algorithm on various real-world data is given in [8]. Here we add a new domain (soil classification) to test the performance of the CFP.

Most real-world data sets contain missing attribute values. Previous learning systems usually overcame this problem by either filling in missing attribute values, or looking at the probability distribution of values of attributes. Most common approaches are compared in [6], leading to a general conclusion that no one approach is uniformly superior to others. In contrast, CFP solves this problem very naturally. Since CFP treats each attribute value separately, in the case of an unknown attribute value, it simply leaves the partitioning of that feature intact.

There are several types of noise that may exist in a data set. One possible type of noise is the *classification noise*. Here the attribute values of an instance represent a valid point in the instance space, however the associated classification is incorrect. In order to cope with this type of noise one must be able to differentiate misclassified instances from correctly classified ones. The CFP algorithm can remove partitions that are believed to be introduced by noisy instances. A new parameter, called *confidence threshold (or level)* (*CT*), is introduced to control the process of removing the partitions from the concept description. If the new training example falls in a partition with a different class than that of the example in a feature, the CFP algorithm specializes the existing partition by dividing it into two subpartitions and inserting a point partition, corresponding to the new example, in between and distributes the representativeness value of the old partition to the newly formed partitions. If the representativeness values of any of the resulting subpartitions drop below the confidence threshold times the observed frequency of its class, then that subpartition is removed from partition list of the feature; see [7] for details.

Depending on the noise level of the data set and the number of the irrelevant attributes, the value of the confidence threshold changes between 0 (do not remove any partition) and 1 (remove a partition if its representativeness value drops below the observed frequency of the its class).

The main subject of this paper is the application of CFP algorithm to the problem of learning to classify soil.

3. SOIL CLASSIFICATION

The characteristics of soil in a construction site plays a very important role in the solution of many civil engineering problems. It is essential that a standard language should exist for the description of soils. The description should include the characteristics of both the soil material and the in-situ soil mass. The principal material characteristics are particle size distribution (or grading) and plasticity. These properties can be determined either by standard laboratory tests or by simple visual and manual procedures. Secondary material characteristics are the color of the soil, and the shape, texture and composition of the particles [1].

Measuring the fundamental soil properties such as permeability, compressibility and strength can be difficult, time consuming, and expensive. In many soil engineering problems, such as pavement design, there are no rational expressions available for the analysis for the solution [5]. For these reasons, sorting soils into groups showing similar behavior may be very helpful Such sorting is *soil classification.* Soil classification permits us to solve many types of simple soil problems and guide the test program if the difficulty and importance of the problem dictate further investigation.

The Corps of Engineers has developed a frost of susceptibility classification in which, on the basis of particle size, one can classify soil in categories of similar frost behavior. The Bureau of Public Roads of USA developed a classification for soils in highway construction. The Corps of Engineers and FAA each developed a classification for airfield construction. In 1952 the Bureau of Reclamation of USA and the Corps of Engineers developed a “unified system” intended for use in all engineering problems involving soils. In 1987 Türk Standartları Enstitüsü published TS1500 “Classification of Soils for Civil Engineering Purposes,” which is based on the unified system [9]. Unified soil classification is shown in Table 1.

The classification of of soil into classes ML, CL, OL, MH, CH and OH is done according to the plasticity chart given in Table 1. Plasticity index I_p is computed as

$$I_p = LL - PL$$

where LL is liquid limit and PL is plasticity limit.

4. APPLICATION OF CFP TO SOIL DATA

The data set used in this experiment was compiled from the records of the Geotechnics Laboratory of Department of Civil Engineering of Middle East Technical University. The data set contains 96 instances (cases) of 7 different classes (CH, CL, MH, OH, SC, SP and SW). In the classification the following features were used:

$P200$	Cumulative percentage passing $80\mu m$ sieve
$P4$	Cumulative percentage passing $4mm$ sieve
LL	Liquid limit
PL	Plasticity limit
D_{10}	The soil diameter at which 10% of the soil weight is finer
D_{30}	The soil diameter at which 30% of the soil weight is finer
D_{60}	The soil diameter at which 60% of the soil weight is finer
$DC\&O$	Existence of dark color and odor

All of these features have continuous values except the last one which is boolean valued. Depending on the values of the first two attributes, technicians performed either LL and PL tests, or diameter tests. Also depending on the value of PL and LL, in some cases the color and odor of the soil is checked. The data set contains 50% unknown feature values. The distribution of instances to classes is given in Table 2. The values in parentheses indicate the numbers used to represent the classes by CFP.

In the experiment we used 75% of the data in training and remaining 25% in testing. The CFP achieved 92% (22 out of 24 test instances) accuracy on the soil data with the following parameter values:

Weight adjustment rate:	0.08
Generalization limits:	30 (P200), 8 (P4), 5 (LL), 13 (PL), 0.1 (D_{10}), 0.5 (D_{30}), 1.2 (D_{60}), 0 (DC&O)
Confidence level:	0.01

The final picture of the CFP containing the partitions formed at the end of the test is given in Figure 2. The minimum and maximum values found each feature are shown on both sides of the feature partitions. For example, the minimum and maximum values of the second feature ($P4$) in the training data are found to be 74.01 and 100, respectively. The feature values of the last test example are given under the heading of "value." This instance has unknown values for the last four features. Individual predictions of each feature are shown under the heading of "Predict." The result of the voting is presented on top as the "PREDICTION."

Table 1: Unified Soil Classification

Coarse-Grained More than half $> 80\mu m$	*Gravel* More than half $> 4mm$	Clean gravels	$C_u \geq 4 and 1 \leq C_r \leq 3$	GW
			$C_u < 4 and/or 1 < C_r < 3$	GP
		Gravels with fines	Fines ML or MH	GM
			Fines CL or CH	GC
	Sand More than half $< 4mm$	Clean sands	$C_u \geq 6 and 1 \leq C_r \leq 3$	SW
			$C_u < 6 and/or 1 < C_r < 3$	SP
		Sands with fines	Fines ML or MH	SM
			Fines CL or CH	SC
Fine-Grained More than half $< 80\mu m$	*Silts and Clays* $LL < 50$	Inorganic	$I_p > 7$ and above A-line	CL
			$I_p < 7$ and below A-line	ML
		Organic	Dark color and odor	OL
	Silts and Clays $LL > 50$	Inorganic	I_p above A-line	CH
			I_p below A-line	MH
		Organic	Dark color and odor	OH

Table 2: Class distribution in the soil data set.

CH(1)	CL(2)	MH(3)	OH(4)	SC(5)	SP(6)	SW(7)
46	28	3	3	11	2	3

Figure 2. Result of CFP for the soil data set.

The actual class value of the instance is also shown as the "CLASS." The last test instance is correctly predicted to be class 1 (CH). The numbers above and below the partitions represent the class number of the corresponding partition. The final weights of features are displayed by the CFP; for example, the weight of the first feature (P200) is found to be 0.920. The CFP determined that the liquid limit is the most important feature with the weight value of 4.751. The least significant feature is the existence of dark color and the odor, whose weight is 0.025. These results closely agree with the human experts in the field.

5. CONCLUSION

This paper presents an application of CFP to learning to classify soil. Soil classification has proved to be a valuable tool to the soil engineer [5]. It helps the engineer by giving him general guidance through making available in an empirical manner the results of field experience. In many soil engineering problems there are no rational expression available for classification of soil.

It is shown that the CFP algorithm has successfully learned to classify soil examples. The representation of classification knowledge learned by CFP, feature partitions, are closer to that of an expert, than the decision tree given in Table 1. For example, given the values of D_{10}, D_{30}, D_{60}, an expert can easily determine the as SP, SC or SW, without considering the values of other attributes. Also if the soil has dark color and odor, an expert can directly determine that the soil is of type OH.

Feature weights are used to cope with the problem of attributes with different importance in classification. Here CFP determined that liquid limit plays the most important role in the soil classification.

Finally, we conclude that the representation of classification knowledge in the form of feature partitions with their relative weights is also applicable to the problem of soil classification.

References

[1] R.F. Craig, *Soil Mechanics*, English Language Book Society.

[2] M. B. Dale, A. B. McBratney, and J. S. Russell, "On the role of expert systems and numerical taxonomy in soil classification," *Journal of Soil Science*, **40**, pp. 223-234, 1989.

[3] H. A. Güvenir and I. Şirin, "Complexity of the CFP, A Method for Classification based on Feature Partitioning" in: *Advances in Artificial Intelligence LNAI 728*, Springer-Verlag, Berlin, pp. 202-207, 1993.

[4] H. A. Güvenir and I. Şirin, "A Genetic Algorithm for Classification by Feature Partitioning" in *the Proceedings of ICGA'93*, Morgan Kaufmann, Illinois, July 1993, pp. 543-548.

[5] T. William Lambe and Robert V. Whitman, *Soil Mechanics, SI Version*, John Wiley & Sons, New York.

[6] J. R. Quinlan, *C4.5: Programs for Machine Learning.* California: Morgan Kaufmann, 1993.

[7] I. Şirin, "Learning with Feature Partitions" M.Sc. Thesis, Bilkent University, Dept. of Computer Engineering and Info. Sci., Tech. Rep. No. BU-CEIS-9312.

[8] I. Şirin and H. A. Güvenir, "Empirical Evaluation of the CFP Algorithm" in *the Proceedings of the 6th Australian Joint Conference on Artificial Intelligence*, World Scientific, Melbourne Australia, Nov. 1993, pp. 311-315.

[9] TSE, "Classification of Soils for Civil Engineering Purposes" TS 1500/Eylül 1987.