Occam's Razor and a Non-syntactic Measure of Decision Tree Complexity

Goutam Paul

Department of Computer Science, SUNY Albany
1400 Washington Avenue, NY 12222
Email: goutam@cs.albany.edu, Web: http://www.cs.albany.edu/~goutam

Occam's razor, attributed to the fourteenth century English philosopher William of Occam, states: "plurality should not be assumed without necessity." The machine learning interpretation of Occam's razor is that if two models have the same performance on the training set, choose the simpler. Decision tree learning widely uses Occam's razor. Popular decision tree generating algorithms are based on information gain criterion which inherently prefers shorter trees (Mitchel 1997). Furthermore, decision tree pruning is common regardless of the splitting criterion. Experiments suggest that shorter trees indeed have better generalization accuracy (*GA*), typically estimated by a validation set prediction accuracy. However, some case studies show evidence apparently against Occam's razor. Recently, Webb (1996) has built C4.5X, a version of C4.5 decision tree classifier (Quinlan 1993) with a postprocessor, which adds more nodes and branches to the tree generated by basic C4.5. He showed that though C4.5 and C4.5X have identical training set accuracies, the generalization accuracy over some datasets is better for C4.5X. But Webb's argument is based on the traditional syntactic complexity measure (number of nodes) of decision trees. In this paper, we explore a non-syntactic measure of decision tree complexity using the notion of Kolmogorov Complexity (Kolmogorov 1965) and show that in this measure the complexity of C4.5X tree is less than that of C4.5 tree on average. Hence, according to our measure of complexity, C4.5X does not violate Occam's razor.

The Kolmogorov Complexity $K_U(X)$ of an object X (typically represented by a binary string) is the shortest possible description $I_{\min}$ (represented by another binary string) of X under a fixed universal description method U. We can think of each description method as a Turing machine and each description as a program. When the Turing machine executes its input program string, it generates the object string as the output. Since the Kolmogorov Complexity of the same object in two different universal Turing machines is bounded by some constant independent of the object (Li & Vitanyi, 1997), it is a measure of intrinsic complexity of the object, unlike information theoretic complexity measures of an object's source (Shanon 1948). Hence, we can omit the subscript U and simply write $K(X)$. We can extend the definition above from a single object to two or more objects by encoding the set of objects into a single string in such a way so that the machine can parse it into a set of description strings for the individual objects (Li & Vitanyi, 1997). If the set of objects is $\{X_1, X_2, \ldots, X_k\}$, we denote the Kolmogorov Complexity of this set by $K(X_1, X_2, \ldots, X_k)$.

If there are m independent attributes, we can represent each instance as a point in an m-dimensional instance space. Each test of attribute places a hyperplane in the instance space parallel to that attribute axis. Thus a decision tree T with k leaf nodes induces a partitioning of the instance space into k blocks $B_1, B_2, \ldots, B_k$, where block B_i is the set of instances belonging to the leaf node i, $i=1,2,\ldots,k$. The hyperplanes placed by all the attribute-tests occurring along the path from the root node to the leaf node i mark the boundaries of the set of instances B_i. We define the *K-complexity* of a decision tree T for a data set as $C(T)=K(B_1, B_2, \ldots, B_k)$. Our measure is non-syntactic in the sense that it does not consider the parameters of the model, rather it considers the partitions of the instance space as induced by the model.

The Kolmogorov complexity of an object is incomputable. There are formal approximations using bounded measures such as Levin Complexity (Zvonkin & Levin 1970). In this paper, we employ the less formal approach of using the length of the compressed object after applying a fixed compression technique. Similar approach has been adopted by the Kolmogorov Complexity community for measuring similarity between strings (Cilibrasi & Vitanyi 2003). We have concatenated the blocks together with an additional newline character as a delimiter between the instances of one block and those of the next and then applied the Linux "compress" utility.

To compare with Webb's result, we have used the same experimental set-up as described in his paper (Webb 1996). The result of our experiments on four datasets is

		Pima	Iris	New Thyroid	Glass
Average Training Set Error (%)	**R**	15.12	1.7	1.486	0
	X	15.12	1.7	1.486	0
Average Test Set Error (%)	**R**	26.812	6.464	7.688	3.057
	X	26.478	5.564	7.432	3.483
Average Tree Size	**R**	49.9	8.2	14.2	11
(Number of Nodes)	**X**	72.14	14.34	22.12	15.58
Average K-complexity	**R**	2706.04	502.21	617.28	1280
	X	2705.98	503.17	616.13	1278.81
% times R and X partitioning differs		98	92	84	96
% times Occam's Razor Holds		72	76	83	80

Table 1: Comparison of Regular C4.5 and Webb's C4.5X: "R" means regular C4.5 and "X" means Webb's extended C4.5

summarized in Table 1. For all of these data sets, the more syntactically complicated C4.5X tree returns better predictive accuracy than the C4.5 tree. Except for Iris, the Kolmogorov Complexity of the test set partitioning averaged over 100 experiments (random divisions of the data into training and test set) is less for C4.5X than for regular C4.5. We define Occam's razor as holding, if and only if for two classifiers T_i and $T_{j,}$ either $C(T_i) \leq C(T_j)$ and $GA(T_i) \geq GA(T_j)$, or $C(T_i) \geq C(T_j)$ and $GA(T_i) \leq GA(T_j)$, otherwise we say Occam's razor fails. As the table shows, out of 100 experiments, more than 70% support Occam's razor in all the datasets including Iris.

We have measured the complexity of partitioning on the test set, rather than on the training set. This is so because C4.5 and C4.5X produce the same partitioning on the training set and both have the same training set accuracies. However, our complexity measure is generic. We can also measure the complexity based on partitioning on the training set, when we are considering two different decision tree generating algorithms having different training set partitioning.

Our *K*-complexity measure of decision trees has two potential applications. First, we can choose between two or more decision trees based on the minimum complexity criterion. Secondly, we can perform prediction on an unknown test instance x as follows. Let $\{B_1, B_2, \ldots, B_k\}$ be the partitioning of the training set. Let $E_i = B_i \cup \{x\}$ and $j = \mathrm{argmin}_{i=1,2,\ldots,k}\{K(E_i) - K(B_i)\}$. We can set the class label of x to be the most occurring class label amongst the instances in block B_j, i.e. the block having the least increase in complexity. We plan to further explore these applications in our future work.

We also intend to compare MML/MDL approach with our work. Traditional message length formulations of decision tree learning were purely syntactic (Quinlan & Rivest 1989, Wallace & Patrick 1992). However, recently people have shown close relationship between MML/MDL and Kolmogorov Complexity in general (Wallace & Dowe 1999, Vitanyi & Li 2000).

References

Mitchell T. M., *Machine Learning*, McGraw Hill, 1997

Webb G. I., "Further Experimental Evidence against the Utility of Occam's Razor," *Journal of Artificial Intelligence Research,* 4 (1996) 397-417

Quinlan J. R., *C4.5: programs for Machine Learni*ng, Morgan Kaufman (1993)

Kolmogorov A. N., "Three approaches to the quantitative definition of information," *Problemi Peredachi Informatsii* (Problems of Information Transmission) 1 (1965) 3-11.

Li M., Vitanyi P., *An Introduction to Kolmogorov Complexity and Its Applications*, Springer (1997)

Shannon C. E., "The Mathematical Theory of Communication," U. of Illinois Press, Urbana, IL (1948)

Zvonkin A. K., Levin L. A., "The Complexity of finite objects and the Algorithmic Concepts of Information and Randomness," *Russian Math. Surveys*, 25:6 (1970), 83-124

Cilibrasi R., Vitanyi P., "Clustering by compression," CWI manuscript (2003), Available from *http://homepages.cwi.nl/~paulv/kolmcompl.html*

Quinlan J. R., Rivest R., "Inferring Decision Trees Using the Minimum Description Length Principle," *Information & Computation* 80 (1989), 227-248

Wallace C. S., Patrick J. D., "Coding Decision Trees," *Technical Report 91/153* (1992), Monash U., Australia

Wallace C., Dowe D., "Minimum Message Length and Kolmogorov complexity," *Computer Journal*, 42:4 (1999), 270-283

Vitanyi P., Li M., "Minimum Description Length Induction, Bayesianism, and Kolmogorov Complexity," *IEEE Transactions on Information Theory*, IT-46:2(2000), 446-464