Comparing High-Order Boolean Features

Adam Drake[1] and Dan Ventura[2]

Computer Science Department, Brigham Young University
[1] acd2@cs.byu.edu
[2] ventura@cs.byu.edu

Abstract

Many learning algorithms attempt, either explicitly or implicitly, to discover useful high-order features. When considering all possible functions that could be encountered, no particular type of high-order feature should be more useful than any other. However, this paper presents arguments and empirical results that suggest that for the learning problems typically encountered in practice, some high-order features may be more useful than others.

Keywords: high-order correlations, feature selection, learning theory.

1. Introduction

Searching for useful high-order relationships (relationships between two or more of the original input features of a learning problem) is a fundamental task of many learning algorithms. Typically, the search for useful high-order features and the types of high-order features learned are implicit in the algorithms. High-order features allow algorithms to more accurately and/or more efficiently model phenomena for which the original, first-order features may be insufficient.

There are many high-order features that could be considered by a learning algorithm—typically far more than can be considered in a feasible amount of time. Therefore, algorithms must limit themselves to searching for one or a few types of high-order relationships. This paper explores the question of whether certain types of high-order relationships are more likely than others to be found in the data of real-world learning problems, and by extension, whether certain types of relationships are more useful to examine than others.

1.1. Motivation

Learning algorithms based on the discrete Fourier transform of Boolean functions (functions of the form f: $\{0,1\}^n \rightarrow \{1,-1\}$) have been used with great success in the field of computational learning theory to prove various learnability results [1][2][3]. However, the potential benefit of applying Fourier-based techniques to real-world problems is not well studied. One real-world application has been presented [4], but it requires the use of a membership oracle, limiting its applicability. The question of whether Fourier-based algorithms can effectively solve more general real-world problems, for which oracle queries may not be possible, remains open.

The study of practical Fourier-based learning leads to a question about the utility of Fourier representations. Fourier-based algorithms represent functions as a linear combination of Fourier basis functions. Let f be an arbitrary function of n Boolean inputs, x_1 through x_n. The Fourier transform of f gives the coefficients, $\hat{f}(\alpha)$, that allow f to be represented as a linear combination of the basis functions χ_α:

$$f(x_1,...,x_n) = \sum_{\alpha \in \{0,1\}^n} \hat{f}(\alpha)\chi_\alpha(x_1,...,x_n)$$

The basis functions are defined as follows:

$$\chi_\alpha(x_1,...,x_n) = \begin{cases} +1 & \text{if } \sum_{i=0}^{n-1} \alpha_i x_i \text{ is even} \\ -1 & \text{if } \sum_{i=0}^{n-1} \alpha_i x_i \text{ is odd} \end{cases}$$

and the Fourier coefficients are computed as shown here:

$$\hat{f}(\alpha) = \frac{1}{2^n} \sum_{x \in \{0,1\}^n} f(x)\chi_\alpha(x)$$

Note that the Fourier transform used here, also known as a Walsh transform, is a simplified Fourier transform for functions of Boolean inputs.

The Fourier basis functions are parity functions, each computing the parity (or the logical XOR) of all inputs x_i such that $\alpha_i = 1$. Thus, the high-order features considered by Fourier-based learning algorithms are high-order XOR functions. The Fourier basis is capable of representing any Boolean function; however, the fact that the representation is based on XOR relationships suggests that a Fourier-based approach would be especially beneficial when useful high-order XOR relationships exist in the data. Similarly, it would seem less beneficial when such correlations do not exist.

This observation begs the following question: Are high-order XOR relationships likely to be found in the

data of real-world problems? And more generally, are some high-order relationships more likely to be found than others?

This paper presents an argument that because feature selection is done by humans, and is therefore biased towards human reasoning, the high-order relationships that exist in real-world data will tend to be biased towards relationships that are indicative of the way humans correlate data. This hypothesis is tested by examining the prevalence of high-order XOR relationships, which are relatively non-intuitive, to more intuitive high-order AND and OR relationships. Tests on several real-world problems suggest that AND and OR relationships are more likely to be found in the data of real-world problems.

2. Kth-Order Boolean Features

The high-order features considered in this paper are patterned after the basis functions of the Fourier transform. Each high-order feature is a function over a subset of the original Boolean input features. The three function types considered here are conjunction (AND), disjunction (OR), and parity (XOR) functions.

Let n be the number of input features of a particular problem, let x_i be the value of the i^{th} feature, and let $S \subseteq \{1,\ldots,n\}$ be the subset of features over which a particular Boolean function is defined. Then the AND, OR, and XOR functions can be defined as follows (note that the XOR functions defined below are functionally equivalent to the Fourier basis functions described previously, but are now defined in terms of the subset S):

$$AND_S(x_1,\ldots,x_n) = \begin{cases} +1 & \text{if } \forall i \in S,\ x_i = 1 \\ -1 & \text{if } \exists i \in S,\ x_i = 0 \end{cases}$$

$$OR_S(x_1,\ldots,x_n) = \begin{cases} +1 & \text{if } \exists i \in S,\ x_i = 1 \\ -1 & \text{if } \forall i \in S,\ x_i = 0 \end{cases}$$

$$XOR_S(x_1,\ldots,x_n) = \begin{cases} +1 & \text{if } \sum_{i \in S} x_i \text{ is even} \\ -1 & \text{if } \sum_{i \in S} x_i \text{ is odd} \end{cases}$$

The AND, OR, and XOR functions compute the logical AND, OR, and XOR, respectively, of the input features specified in S. For example, the function $AND_{\{1,3,4\}}$ computes the logical AND of the first, third, and fourth features. It is equivalent to the expression $x_1 \wedge x_3 \wedge x_4$. The order of a function, k, is the number of elements in S. Thus, $AND_{\{1,3,4\}}$ is a third-order feature. In this paper, high-order features are defined as those for which $k \geq 2$.

Given a data set with n input features, there are 2^n possible subsets, and therefore 2^n possible functions, for each type of relationship. One of these subsets is the empty set, which for each relationship type gives a constant function. In addition, there are n subsets containing only one feature. These n first-order functions are also equivalent for each type of relationship. The remaining 2^n-n-1 functions are unique for each type of relationship, and compute all possible second- and higher-order AND, OR, and XOR relationships.

When computing the Fourier transform of a function f, a negative coefficient indicates that f is negatively correlated with some XOR function, XOR_S, and therefore positively correlated with $XNOR_S$. Similarly, if f is negatively correlated with an AND or OR function, it is positively correlated with the corresponding NAND or NOR function, respectively. However, for simplicity, the inversion is ignored in the following discussion, and a strong correlation could refer to either a strong positive or a strong negative correlation. Thus, for example, an AND correlation could refer to either an AND or a NAND correlation. (The grouping of AND with NAND and OR with NOR is natural when patterning the AND and OR functions after the Fourier basis functions. However, by DeMorgan's law, $NAND_S$ is equivalent to $OR_{S'}$, and NOR_S is equivalent to $AND_{S'}$, where S' signifies that the inputs in S are inverted. Consequently, AND could be logically grouped with NOR, and OR with NAND. However, the choice of grouping does not significantly alter the results presented in this paper, nor does it affect the conclusions.)

3. An Argument for Intuitive High-Order Features

A "no free lunch" [5] argument would suggest that no high-order relationships will be better on average than any others. When considering two possible high-order relationships, there will be just as many functions for which the first is better as there will be for the second. However, there are reasons why some correlations might be more likely to be useful in practice.

Data sets encountered in the real world are not randomly generated. In general, data sets are gathered by people who select the features that they think will be most useful in analyzing a particular problem. Because people are selecting the features, the data sets of real-world problems will be biased towards whatever reasoning humans use to select features. Thus, the question of whether certain high-order relationships are more likely to appear in data than others can be reposed as a question of whether the

features selected by humans are more likely to exhibit some high-order relationships than others.

A consideration of these issues leads to the following reasoning. It is very natural for people to think in terms of conjunctions (AND) and disjunctions (OR). These logical operators are very intuitive. Because people tend to think in terms of AND and OR, we hypothesize that humans are more likely to pick features that combine well in useful high-order AND and OR relationships than in other less intuitive relationships.

The XOR relationship, although fairly intuitive when involving only two variables, is less intuitive when more variables are involved. It seems less likely that people will select features that exhibit useful high-order XOR relationships.

The generalization of these ideas would be that in general, high-order relationships that are intuitive and representative of the way people think are more likely to be useful features in human-biased data sets. Although significant testing would be required to verify this claim, the results of this paper provide some early supporting evidence.

A final consideration is not only whether certain high-order features exist, but whether they are useful. Even if it is true that intuitive features are more useful, it may still be possible to find other high-order correlations. Although these coincidental relationships may exist in the data, because they do not reflect the bias introduced by human feature selection they may not generalize as well to unseen data.

4. Comparing High-Order Features

To test the prevalence of different high-order correlations, several real-world data sets were taken from the UCI machine learning repository [6]. As this work was motivated by a study of functions with Boolean inputs, all data sets considered either contain only Boolean-valued features or have had their non-Boolean features encoded as Boolean features.

For continuously-valued inputs, a reasonable threshold was chosen, and values above the threshold were assigned a 1, while values below the threshold were assigned a 0. Nominally-valued inputs were encoded into binary using the minimum number of bits required to account for each possible value. There was some concern that this choice of encoding might affect the types of high-order correlations found, but our testing suggested that it made little or no difference. If anything, encoding the original input variables would seem to increase the likelihood of non-intuitive correlations being found.

Each data set was examined in terms of AND, OR, and XOR relationships. For each type of relationship, the most highly correlated feature was determined by checking how well all 2^n functions (and their inverses) correctly classified examples in the data set. In addition, the accuracy of the most highly correlated first-order feature was computed to give some idea of the usefulness of the high-order features.

Table 1 shows the results of this experiment. The classification accuracy of the most highly correlated function of each type, along with the accuracy of the most highly correlated first-order feature, is shown for each data set. The best accuracy for each data set is highlighted in bold.

Data Set	1st	AND_S	OR_S	XOR_S
Adult	80.3	**82.1**	81.6	81.6
Chess	68.3	67.7	**81.1**	75.3
German	71.7	71.7	**73.1**	71.7
Heart	75.6	76.3	**77.0**	76.3
Pima	73.6	**75.4**	71.1	65.9
SPECT	66.3	79.4	**87.6**	70.8
Voting	**96.3**	95.9	90.1	88.1
WBC1	87.3	87.1	**96.0**	92.7
WBC2	76.8	**80.3**	78.8	77.3
WBC3	91.4	**94.4**	89.8	91.2

Table 1: Best classification accuracy of any first-order or higher-order AND, OR, or XOR feature. For each data set, the best feature is highlighted in bold.

For each of the ten data sets tested, the most highly correlated function was always either an AND or an OR function, supporting the idea that AND and OR relationships are more likely to be found in real-world data. No XOR function was ever the most highly correlated high-order feature. On the other hand, the best XOR function was sometimes not far behind in accuracy, and it was not always the worst of the three.

For one of the data sets, the Voting set, none of the high-order features were better than the best first-order feature. However, of the three feature types, the best feature for that set was a high-order AND feature.

Table 2 shows the orders of the most highly correlated high-order AND, OR, and XOR features. In several cases, multiple features of a single type were equally well correlated. In these cases, a range of orders is reported, indicating that the orders of the best features fell within that range. The number of features in each data set is shown in parentheses.

5. Conclusions and Future Work

Given that we had no prior reason to believe that a particular correlation type would be more or less prevalent in the data sets tested, the fact that no XOR relationship was ever the best individual high-order

Data Set	AND_S	OR_S	XOR_S
Adult (34)	3	2	2
Chess (37)	2	5-6	6
German (24)	5	4-7	2
Heart (16)	2	2-3	2
Pima (8)	2	2	2
SPECT (22)	5-22	8-12	2
Voting (16)	2	2	3
WBC1 (36)	2	5-9	2
WBC2 (33)	4-20	2-6	3-4
WBC3 (30)	4	2	2

Table 2: The orders of the most highly correlated AND, OR, or XOR features for each data set. The number of input features in each data set is shown in parentheses.

feature is significant. These results seem to suggest that algorithms that implicitly or explicitly search for high-order AND and OR correlations will tend to be more successful than those based on XOR.

An interesting observation is that although the best high-order features were always either AND or OR relationships, neither the AND nor the OR features were universally better. For example, the Chess data set exhibited strong high-order OR correlations, but only weak AND correlations. On the other hand, the Voting data set contained significantly stronger AND correlations than OR correlations. This suggests that an algorithm capable of effectively learning both types of correlations should be more successful over a broader range of learning problems.

A potentially interesting implication of this research regards the importance of being able to learn XOR relationships. For example, the perceptron learning algorithm has received criticism for its inability to learn XOR relationships [7]. However, the results presented here suggest that this may not be a significant weakness when working with typical real-world problems.

Another interesting observation regards the orders of the best high-order features shown in Table 2. In general, the orders of the most useful relationships tended to be fairly low relative to the total number of input features. This observation also seems to support the idea that data sets are biased towards human reasoning, as humans are not likely to consider very high-order relationships. (The unusually high-order correlations found in the SPECT and WBC2 data sets are primarily a consequence of the high ratios of positive to negative examples found in those sets.)

It is important to note that the results of this paper test individual high-order features, and not the learning potential of combinations of high-order features. An interesting test for future work will be to determine if the same patterns exist when combinations of high-order features are used. For example, do combinations of either AND or OR features always outperform combinations of XOR features. Another interesting question is whether an algorithm that can use each type of high-order feature benefits from the use of high-order XOR features or if they tend to not be useful even in combination with other features.

It may be true that high-order features that are more intuitive to humans are more likely to be useful in solving real-world learning problems. Although the research presented here is supportive, more research will be required to validate this claim. Future work would include testing over a broader range of learning problems and testing more types of correlations. For example, this research tested correlations of Boolean-valued attributes on classification problems, but there are other correlations and problems to consider. For example, which high-order correlations are most useful when dealing with real-valued features or when performing regression.

Another important area of future work will be in comparing the generalization capabilities of each type of high-order feature. Although many high-order relationships may exist in data, some may not generalize as well as others.

6. References

[1] N. Linial, Y. Mansour, and N. Nisan, "Constant Depth Circuits, Fourier Transform, and Learnability," *Journal of the Association for Computing Machinery*, vol. 40, n. 3, pp. 607-620, 1993.

[2] E. Kushilevitz and Y. Mansour, "Learning Decision Trees using the Fourier Spectrum," *SIAM Journal on Computing*, vol. 22, n. 6, pp. 1331-1348, 1993.

[3] J. Jackson, "An Efficient Membership-Query Algorithm for Learning DNF with Respect to the Uniform Distribution," *Journal of Computer and System Sciences*, vol. 55, pp. 414-440, 1997.

[4] Y. Mansour & S. Sahar, "Implementation Issues in the Fourier Transform Algorithm," *Machine Learning*, vol. 14, pp. 5-33, 2000.

[5] D. Wolpert and W. Macready, "No Free Lunch Theorems for Optimization," *IEEE Transactions on Evolutionary Computing*, vol. 1, n. 1, pp. 67-82, 1997.

[6] C. L. Blake and C. J. Merz, UCI Repository of Machine Learning Databases [http://www.ics.uci.edu/~mlearn/MLRepository.html], Irvine, CA: University of California, Department of Information and Computer Science, 1998.

[7] M. Minsky and S. Papert, *Perceptrons*, MIT Press: Cambridge, MA, 1969.