THEORY OF COMPUTING, Volume 9 (17), 2013, pp. 587–592
www.theoryofcomputing.org

NOTE

SPECIAL ISSUE: ANALYSIS OF BOOLEAN FUNCTIONS

A Monotone Function Given By a Low-Depth Decision Tree That Is Not an Approximate Junta

Daniel M. Kane *

Received April 23, 2012; Revised September 18, 2012; Published June 10, 2013

Abstract: We present a family a monotone functions $f_d : \{0,1\}^n \to \{0,1\}$ so that f_d can be computed as a depth-d decision tree and so that f_d disagrees with any k-junta on a constant fraction of inputs for any $k = \exp(o(\sqrt{d}))$. This gives a negative answer to a problem circulated independently by Elad Verbin and by Rocco Servedio and Li-Yang Tan.

ACM Classification: F.2.3

AMS Classification: 06E30

Key words and phrases: Boolean functions, monotone functions, decision tree

1 Introduction

In [3], O'Donnell and Servedio show that any monotone function given by a depth-d decision tree can be learned to constant accuracy from random samples in $\text{poly}(n, 2^d)$ time. The impact of this result is somewhat lessened by an apparent lack of interesting monotone functions given by low-depth decision trees. In particular, Elad Verbin as well as Rocco Servedio and Li-Yang Tan independently suggested in 2010 that all such functions may essentially depend on few variables [1, page 10].

*Supported by an NSF postdoctoral fellowship

 DOI: 10.4086/toc.2013.v009a017

Conjecture 1.1. *For every $\varepsilon > 0$ and every monotone function $f : \{0,1\}^n \to \{0,1\}$ given by a depth-d decision tree, there is a k-junta, g, for $k = poly_\varepsilon(d)$ so that f and g agree on all but an ε-fraction of inputs.*

In this note, we disprove the above conjecture, and in particular provide an example of a monotone low-degree function that is not well approximated by any small junta. In particular we prove:

Theorem 1.2. *There exists a constant $\varepsilon > 0$ so that for every positive integer d, there exists a $k = \exp(\Omega(\sqrt{d}))$ and a monotone function $f : \{0,1\}^n \to \{0,1\}$ given by a depth-d decision tree, so that for every k-junta g, f and g disagree on at least an ε-fraction of inputs.*

In fact it is known that the bound on k in Theorem 1.2 is tight up to the constant in the exponent. In particular, it is shown in [3] that any monotone function given by a depth-d decision tree has total influence $I(f) = O(\sqrt{d})$. We combine this with the main result of [2], which says that any Boolean function f can be ε-approximated by a k-junta for $k = \exp\big(O(I(f)/\varepsilon)\big)$. Combining these results we find that:

Observation 1.3. If f is a monotone function given by a depth-d decision tree, and if $\varepsilon > 0$, then there is a k-junta g that agrees with f on all but an ε fraction of inputs for $k = \exp\big(O(\sqrt{d}/\varepsilon)\big)$.

The function we construct to show Theorem 1.2 will combine ideas from two previous constructions, the monotone addressing function and Talagrand's function.

The monotone addressing function is defined by

$$f(x_1,\dots,x_{d-1},y_0,\dots,y_{2^{d-1}-1}) = \begin{cases} 1 & \text{if } \sum x_i > \lfloor (d-1)/2 \rfloor, \\ y_{x_0\dots x_{d-1}} & \text{if } \sum x_i = \lfloor (d-1)/2 \rfloor, \\ 0 & \text{if } \sum x_i < \lfloor (d-1)/2 \rfloor. \end{cases}$$

This is an example of a monotone function given by a depth-d decision tree that depends on exponentially many variables, and thus provides us with a good starting point. The monotone addressing function fails to provide a counter-example to Conjecture 1.1 though since it agrees with the majority function except on a set of measure $O(1/\sqrt{d})$.

Given the bound on the total sensitivity of a low-depth monotone function, we know that any f satisfying the conditions of Theorem 1.2 must not only have near the maximum possible total influence for a low-depth monotone function, but also must not be approximable by a function with much lower total influence. Because of this restriction, our construction will look somewhat similar to a construction of Talagrand in [4]. In particular, Talagrand constructs a monotone function f on $\{0,1\}^d$ so that on a constant fraction of inputs, f has sensitivity (i. e., the number of coordinates such that changing the input at that coordinate would change the output of f) $\Omega(\sqrt{d})$. Since, as is easily seen, the average sensitivity over all inputs is equal to the total influence, this is as large as possible. On the other hand, this condition tells us that f retains large average sensitivity even after ignoring any ε-fraction of inputs for sufficiently small constant ε. Talagrand's function fails to provide a counter-example to Conjecture 1.1 on its own, because it is already a d-junta.

2 The construction

In order to define the function f with the properties specified by Theorem 1.2, we first introduce some background notation. We let d,t and m be integers with $t = \Theta(\sqrt{d})$ and $m = \Theta(2^t)$. We furthermore assume that $2^{-t}m$ is sufficiently small given the value of $t/\sqrt{d}$. We let $\mathcal{S} = (S_1, \ldots, S_m)$ be a random sequence of sets, where the S_i are chosen independently and uniformly from the set of subsets of $\{1,2,\ldots,d-1\}$ of size exactly t. Given this $\mathcal{S}$, we define the function $T_{\mathcal{S}}$ on $\{0,1\}^{d-1}$ as follows:

$$T_{\mathcal{S}}(x_1,\ldots,x_{d-1}) = \{1 \le i \le m : x_j = 1 \text{ for all } j \in S_i\}.$$

We will hereafter abbreviate T by suppressing the explicit dependence on $\mathcal{S}$, and abbreviate $(x_1,\ldots,x_{d-1})$ by x.

We finally define f as

$$f_{\mathcal{S}}(x_1,\ldots,x_{d-1},y_1,\ldots,y_m) = \begin{cases} 1 & \text{if } |T(x)| \ge 2, \\ 0 & \text{if } |T(x)| = 0, \\ y_i & \text{if } T(x) = \{i\}. \end{cases}$$

Again, we will often suppress the dependence of f on $\mathcal{S}$. It is clear that f is monotone. Furthermore, f is given by a depth-d decision tree, since after fixing the values of the x_i, the value of f depends on at most one more coordinate. In the next section, we show that f cannot be approximated by any k-junta for small k.

Note that Talagrand's function is given (for appropriately chosen $\mathcal{S}$) by

$$G(x_1,\ldots,x_{d-1}) = \begin{cases} 1 & \text{if } |T(x)| \ge 1, \\ 0 & \text{if } |T(x)| = 0. \end{cases}$$

3 Approximation bounds

Theorem 1.2 will follow from the following proposition:

Proposition 3.1. *There exists an $\varepsilon > 0$ so that for $f_{\mathcal{S}}$ defined as above, with constant probability over the choice of $\mathcal{S}$, f is not ε-approximated by any k-junta for $k = o(2^t)$.*

A key step in our proof will be to show that with constant probability f actually depends on one of the y_i.

Lemma 3.2. *With T as above,*

$$\Pr_{\mathcal{S},x}(|T_{\mathcal{S}}(x)| = 1) = \Omega(1).$$

Proof. We will show the further claim that

$$\mathbb{E}\left[|T_{\mathcal{S}}(x)|(2 - |T_{\mathcal{S}}(x)|)\right] = \Omega(1). \tag{1}$$

Since the term in the expectation is positive only if $|T| = 1$, this will complete our proof. We note that

$$\begin{aligned}\mathbb{E}\left[|T_{\mathbb{S}}(x)|\right] &= \sum_{i=1}^{m} \Pr(i \in T_{\mathbb{S}}(x)) \\ &= \sum_{i=1}^{m} \Pr(x_j = 1 \text{ for all } j \in S_i) \\ &= m2^{-t}.\end{aligned}$$

On the other hand, we have that

$$\begin{aligned}\mathbb{E}\big[|T_{\mathbb{S}}(x)|(|T_{\mathbb{S}}(x)|-1)\big] &= \sum_{i\neq j} \Pr(i,j \in T_{\mathbb{S}}(x)) \\ &= \sum_{i\neq j} \Pr(i \in T_{\mathbb{S}}(x)) \Pr(j \in T_{\mathbb{S}}(x) \mid i \in T_{\mathbb{S}}(x)) \\ &= \sum_{i\neq j} 2^{-t} \Pr(x_\ell = 1 \text{ for all } \ell \in S_j \mid x_\ell = 1 \text{ for all } \ell \in S_i)\,.\end{aligned}$$

To compute this conditional probability we let $S_j = \{a_1, \dots, a_t\}$ where the a_i are picked randomly from $\{1, 2, \dots, d-1\}$ without replacement. We compute it as the product

$$\prod_{k=1}^{t} \Pr(x_{a_k} = 1 \mid x_{a_1} = \dots = x_{a_{k-1}} = 1 \text{ and } x_\ell = 1 \text{ for all } \ell \in S_i)\,.$$

These probabilities are approximated by first fixing the values of S_i and $a_1, \dots, a_{k-1}$. After additionally fixing the value of a_k, the probability in question becomes 1 if $a_k \in S_i$ and $1/2$ otherwise. Thus the probability that $x_{a_r} = 1$ is

$$\frac{1+\Pr(a_r \in S_i)}{2} = \frac{1 + \frac{|S_i \setminus \{a_1,\dots,a_{r-1}\}|}{d-r}}{2} = 1/2 + O(t/d)\,.$$

Hence the probability that $j \in T_{\mathbb{S}}(x)$ given that $i \in T_{\mathbb{S}}(x)$ is

$$\big(1/2 + O(t/d)\big)^t = 2^{-t} \exp(O(t^2/d))\,.$$

Therefore, we have that

$$\mathbb{E}\big[|T_{\mathbb{S}}(x)|(|T_{\mathbb{S}}(x)|-1)\big] = \sum_{i\neq j} 2^{-2t} \exp(O(t^2/d)) \leq (2^{-t}m)^2 \exp(O(t^2/d))$$

and hence

$$\begin{aligned}\mathbb{E}\left[|T_{\mathbb{S}}(x)|(2-|T_{\mathbb{S}}(x)|)\right] &= \mathbb{E}\left[|T_{\mathbb{S}}(x)|\right] - \mathbb{E}\left[|T_{\mathbb{S}}(x)|(|T_{\mathbb{S}}(x)|-1)\right] \\ &\geq (2^{-t}m) - (2^{-t}m)^2 \exp(O(t^2/d)) \\ &= (2^{-t}m)\big(1 - (2^{-t}m)\exp(O(t^2/d))\big)\,.\end{aligned}$$

As long as $2^{-t}m$ is bounded below by a constant and above by $\exp(-O(t^2/d))/2$, this is $\Omega(1)$. □

We are now ready to prove Proposition 3.1. By Lemma 3.2, we note that with constant probability over $\mathcal{S}$, that $\Pr_x(|T(x)| = 1) = \Omega(1)$. For such $\mathcal{S}$, we claim that f has the desired property. In particular we claim the following:

Lemma 3.3. *If f is as above and g is a k-junta, then*

$$\Pr\big(f(x,y) \neq g(x,y)\big) \geq \frac{\Pr_x(|T(x)| = 1) - k2^{-t}}{2}.$$

Proof. This follows from the simple observation that if, after fixing the value of x, we have that $T = \{i\}$ where g does not depend on y_i, then $\Pr_y(f(x,y) \neq g(x,y)) = 1/2$. This is because after further conditioning on the values of all y_j for $j \neq i$, g becomes a constant function (by assumption) and f takes the values 0 and 1 each with probability $1/2$. Therefore we have that

$$\begin{aligned}
\Pr(f(x,y) \neq g(x,y)) &\geq \frac{\Pr(T(x) = \{i\} \text{ for some } i, \text{ and } g \text{ does not depend on } y_i)}{2} \\
&= \frac{\Pr(|T(x)| = 1) - \Pr(T(x) = \{i\} \text{ for some } i, \text{ and } g \text{ depends on } y_i)}{2} \\
&= \frac{\Pr(|T(x)| = 1) - \sum_{i: g \text{ depends on } y_i} \Pr(T(x) = \{i\})}{2} \\
&\geq \frac{\Pr(|T(x)| = 1) - \sum_{i: g \text{ depends on } y_i} \Pr(i \in T(x))}{2} \\
&= \frac{\Pr(|T(x)| = 1) - \sum_{i: g \text{ depends on } y_i} 2^{-t}}{2} \\
&\geq \frac{\Pr_x(|T(x)| = 1) - k2^{-t}}{2}.
\end{aligned}$$

□

Proposition 3.1 and Theorem 1.2 now follow immediately.

Acknowledgements

I would like to thank Ryan O'Donnell for making me aware of this problem, and for his help with finding appropriate references for this paper. This research was done with the support of an NSF postdoctoral fellowship.

References

[1] Open Problems in Analysis of Boolean Functions, Compiled for the Simons Symposium, February 5-11, 2012. Technical report, 2012. [arXiv:1204.6447] 587

[2] EHUD FRIEDGUT: Boolean functions with low average sensitivity depend on few coordinates. *Combinatorica*, 18(1):27–35, 1998. [doi:10.1007/PL00009809] 588

[3] RYAN O'DONNELL AND ROCCO A. SERVEDIO: Learning monotone decision trees in polynomial time. *SIAM J. Comput.*, 37(3):827–844, 2007. Preliminary version in CCC'06. [doi:10.1137/060669309] 587, 588

[4] MICHEL TALAGRAND: How much are increasing sets positively correlated? *Combinatorica*, 16(2):243–258, 1996. [doi:10.1007/BF01844850] 588

AUTHOR

Daniel M. Kane
Stanford University Department of Mathematics, Stanford, CA
dankane@math.stanford.edu
http://math.stanford.edu/~dankane/

ABOUT THE AUTHOR

DANIEL KANE graduated from the mathematics department at Harvard in 2011; his advisor was Barry Mazur. His thesis work covered a number of topics mostly in the areas of theoretical computer science and analytic number theory. Although his interests are diverse, his recent work on computer science has tended to focus on polynomial threshold functions and derandomization. He is currently working as a postdoc in the mathematics department of Stanford University.

Daniel's early interest in mathematics was strongly influenced by his father, Jonathan Kane, who was himself a professor of mathematics and computer science at the University of Wisconsin-Whitewater.