Machine Learning, 13, 131-134 (1993)
© 1993 Kluwer Academic Publishers, Boston. Manufactured in The Netherlands.

Research Note on Decision Lists

RON KOHAVI RONNYK@CS.STANFORD.EDU
SCOTT BENSON SBENSON@CS.STANFORD.EDU
Computer Science Dept., Stanford University, Stanford, CA 94305

Editor: David Haussler

Abstract. In his article "Learning Decision Lists," Rivest proves that (k-DNF $\cup$ k-CNF) is a proper subset of k-DL. The proof is based on the following incorrect claim:

> . . . if a function f has a prime implicant of size t, then f has no k-DNF representation if $k < t$.

In this note, we show a counterexample to the claim and then prove a stronger theorem, from which Rivest's theorem follows as a corollary.

1. A counterexample

In the article "Learning Decision Lists" (Rivest, 1987) Rivest proves that (k-DNF $\cup$ k-CNF) is a proper subset of k-DL. The proof is based on the following incorrect claim:

> . . . if a function f has a prime implicant of size t, then f has no k-DNF representation if $k < t$.

The following counterexample shows that it is possible for a function f with a prime implicant of size four to have a 3-DNF representation. The function f shown below is in 3-DNF, yet the term $w\bar{x}y\bar{z}$ is a prime implicant of the function.

$$f(v, w, x, y, z) = vw\bar{x} \vee \bar{v}y\bar{z}$$

Figure 1 shows the function using a Karnaugh map of five variables with the prime implicant containing four literals shaded. (For a description of Karnaugh maps, see, for example, Kohavi (1978) or Friedman (1986), although readers not familiar with them may easily check that the given term is indeed a prime implicant.)

2. The expressive power of decision lists

Let n be the number of variables in our language.

Definition 1 (Prime implicant). *A prime implicant for a function f is a product term α that implies f, but that does not imply f if any literal in α is deleted.*

vw \ xyz	000	001	011	010	110	111	101	100
00	0	0	0	1	1	0	0	0
01	0	0	0	1	1	0	0	0
11	1	1	1	1	0	0	0	0
10	0	0	0	0	0	0	0	0

Figure 1. A Karnaugh map that refutes the claim.

Definition 2 (Essential prime implicant). *An essential prime implicant α of f is a prime implicant such that there exists an $x \in \{0, 1\}^n$ with $\alpha(x) = 1$, yet for no prime implicant $\beta \neq \alpha$ does $\beta(x) = 1$.*

Lemma 1. *If a function f has an* essential prime implicant *of size t, then f has no k-DNF(n) representation if $k < t$.*

Proof: The essential prime implicant must appear in any DNF(n) representation that uses only prime implicants. Any k-DNF(n) representation has an equivalent k-DNF(n) representation using only prime implicants; therefore, there cannot exist a k-DNF(n) representation of f with $k < t$. □

Note that this lemma only defines a sufficient condition for not having a k-DNF(n) representation. There are functions that have no essential prime implicants at all.

Lemma 2. *A prime implicant α of size n is an essential prime implicant.*

Proof: Let $x \in \{0, 1\}^n$ be the unique vector such that $\alpha(x) = 1$. If there exists a prime implicant $\beta \neq \alpha$ for which $\beta(x) = 1$, then α and β cannot disagree on any literal (or else $\beta(x) \neq 1$). Since all variables appear in α, the prime implicant β must contain only a subset of the literals in α, contradicting the fact that α is a prime implicant. □

Theorem 3. *For $1 < k < n$ and $n > 2$, there are functions representable in k-DL(n) but not in (j-CNF(n) $\cup$ j-DNF(n)) for any $j < n$.*

Proof: We prove a stronger result, namely, that 2-DL(n) contains functions not representable in (j-CNF(n) $\cup$ j-DNF(n)) for any $j < n$, and $n > 2$.

Let f be the function represented by the following 2-DL(n):

$(\overline{x_1}\,\overline{x_2}, 0)$, $(\overline{x_1}\,\overline{x_3}, 0)$, . . . , $(\overline{x_1}\,\overline{x_n}, 0)$, $(\overline{x_1}, 1)$, $(x_1\,\overline{x_2}, 1)$, $(x_1\,\overline{x_3}, 1)$, . . . , $(x_1\,\overline{x_n}, 1)$, (true, 0)

$x_1x_2x_3$ / x_4x_5	000	001	011	010	110	111	101	100
00	0	0	0	0	1	1	1	1
01	0	0	0	0	1	1	1	1
11	0	0	1	0	1	0	1	1
10	0	0	0	0	1	1	1	1

Figure 2. A Karnaugh map showing the funciton in 2-DL(n) for $n = 5$.

Note that the last term could be replaced by $(x_1, 0)$, but the definition of a decision list requires the last term to contain the constant function **true**. Figure 2 shows a Karnaugh map of the function for $n = 5$.

Let α be the term $\overline{x_1}x_2x_3 \ldots x_n$ and let α' be a term derived from α with one literal l_i deleted. α' implies $\alpha'\overline{l_i}$, but for any $\vec{x} \in \{0, 1\}^n$ such that $\alpha\overline{l_i}$ is true, $f(\vec{x})$ is 0, and thus α is a prime implicant of f. By lemma 2, α is an essential prime implicant, and by lemma 1, f has no j-DNF(n) representation for $j < n$.

Similarly, the term $x_1x_2x_3 \ldots x_n$ is an essential prime implicant of $\bar{f}$, and thus the function $\bar{f}$ cannot be represented in j-DNF(n) for $j < n$. Since the complement of every j-CNF(n) formula is a j-DNF(n) formula, there is no j-DNF(n) representation for f, and hence f cannot be represented in j-DNF(n) $\cup$ j-CNF(n) for $j < n$.

Corollary 4 (Rivest). *For* $0 < k < n$ *and* $n > 2$, *(k-CNF(n)* $\cup$ *k-DNF(n)) is a proper subset of k-DL(n).*

Proof: The original article (Rivest, 1987) correctly proved that any k-CNF(n) formula and any k-DNF(n) formula can be written in k-DL(n). By theorem 3, there are functions in k-DL(n) not in (k-CNF(n) $\cup$ k-DNF(n)) for $k > 1$, so only the case $k = 1$ remains to be proved.

If $k = 1$, then the following decision list from 1-DL(n) represents a function f that is not in 1-CNF(n) $\cup$ 1-DNF(n):

$$(x_1, 0), (x_2, 1), (x_3, 1), (\text{true}, 0)$$

The only prime implicants of the function f are $\overline{x_1}x_2$ and $\overline{x_1}x_3$. Both are essential, so f does not have a 1-DNF(n) representation. Similarly, the function $\bar{f}$ has x_1 and $\overline{x_2}\,\overline{x_3}$ as the only prime implicants and again both are essential, so f does not have a 1-DNF(n) $\cup$ 1-CNF(n) representation.

Acknowledgments

We thank the anonymous referee for comments on making the proof shorter. We thank Nils Nilsson, Ron Rivest, and George John for their comments on a previous version of this article, and Shai Halevi for coming up with the counterexample, which is smaller than our original one.

References

Friedman, Arthur D. (1986). *Fundamentals of logic design and switching*. Computer Science Press.
Kohavi, Zvi. (1978). *Switching and finite automata theory*, 2nd edition. New York: McGraw-Hill.
Rivest, Ronald L. (1987). Learning decision lists. *Machine Learning, 2, 229#246*.

Received July 30, 1992
Accepted October 7, 1992
Final Manuscript February 18, 1993