Taking the fuzz out of fuzzy logic

Warren D. Smith
NEC Research Institute Princeton NJ 08540

January 4, 1998

Abstract

This paper gives an algorithm which, given any boolean function F whose n arguments are known *probabilities*, will deduce the tightest possible upper and lower bounds (assuming nothing is known about the correlations among the probabilities) on the value of F. This may be accomplished by solving a 2^n-dimensional linear program. (It is not suprising that the linear program has exponential dimensionality, since if subexponential dimensionality were possible, then SAT would be soluble in subexponential time.) If something is known about the correlations this knowledge may also be incorporated into the linear program. With little additional trouble one may also find the max-entropy estimate of F.

"Fuzzy logic" attempts to generalize logic to concern "fuzzy truth values" – real numbers in the interval $[0,1]$. We now can have simultaneously: (1) a probabilistic interpretation of fuzzy truth values, and (2) total rigor. This comes at the cost of an exponential amount of computation, but: (1) we gain more understanding of the (polytopal) structure of fuzz-land, and (2) Arguably: only small values of n arise in practice, or else the boolean functions encountered in practice are "modularized" with each "module" having a small complexity, hence this exponential complexity is acceptable.

Fuzzy Wuzzy was a bear
Fuzzy Wuzzy lost his hair
Fuzzy Wuzzy wasn't fuzzy, was he?
No, he was a bare bear! – Unattributable.

1 Introduction

There have been several schemes suggested for trying to generalize "logic" (which are operations on Boolean $\{0,1\}$ variables) to work for "fuzzy truth values", i.e. real variables in the real interval $I = [0,1]$.

For example, one might think of logical variables as "independent" probabilities. This suggests the following reinterpretation

logical op.	independent probabilities model op.
u and v	uv
not u	$1-u$
u or v	$u+v-uv$
u xor v	$u+v-2uv$

The advantage of this scheme is it is easy to understand and the assumptions underlying it are too. The problem, of course, is nonindependence – the expression "u and u" is treated rather poorly by this scheme! (One might also observe that the present expression for "u xor v" arises by viewing it as $u\bar{v} \vee v\bar{u}$, but if instead we use the logically equivalent expression $(u \vee v) \wedge (\bar{u} \vee \bar{v})$, a different formula is obtained!)

Alternatively, one may use a scheme of the type suggested by L.A. Zadeh [6]; here is a variation due to Karvel Thornber [3]:

logical op.	Thornber fuzzy op.
u and v	$\min(u, v)$
not u	$1-u$
u or v	$\max(u, v)$
u xor v	$\max(\min(u, 1-v), \min(v, 1-u))$
	$= \min(\max(u, v), \max(1-u, 1-v))$
v when $\{\{w =$"u implies v"$\}$ and $u\}$	if $w > 1-u$ then w; otherwise $0 \leq v < w$ but v is otherwise undefined.

This seems to "lose less" somehow than the independent probabilities model. Both models preserve certain properties of the logical operands, i.e. commutativity, distributivity, and associativity of "and," "or;" Thornber's choice in the last row of the table above also preserves "symmetry" of modus ponens under $v \leftrightarrow \text{not} u$. Thornber discusses the meaning of: "reductio ad absurdam," proof by "induction," and "logical cuts" in this model.

This model will get the value of "u and u" right, unlike the probabilities model, but not "u and (not u)!" Unlike in the probability model, one has trouble understanding exactly what these choices *mean* (Are the fuzzy truth values probabilities? What are they?) and it is no longer evident just when the application of the rules above is "valid."

All this is bad. What we would like is something that both preserves the ability to interpret fuzzy truth values as probabilities, and also preserves rigor – no unjustified assumptions about independence, etc. are to be made and it will never get a value "wrong" as in the examples above – and finally also preserves the "less loss" quality of fuzzy logic. This will now be accomplished.

2 The Problem and the Algorithm

Suppose we have a boolean-valued function F of n boolean-valued arguments $x_1, \ldots, x_n$. Such a function is specified by a "truth table:" for each of the 2^n possible inputs, we specify the value (0 or 1) of F. We wish to reinterpret this function in such a way that both the inputs and the output become fuzzy. In fact, if the x_i are probabilities of certain boolean-valued events E_i (and nothing is known about the correlations among these probabilities) then we will now describe a way to compute tightest possible upper and lower bounds upon the probability that $F(E_1, \ldots, E_n) = 1$. Call these bounds U and L respectively.

Let b be an n-bit binary number, and let "b_i" denote the i^{th} bit of b. Define the 2^n variables T_b, so that T_b is the probability that $E_i = b_i$ for each $i = 1 \ldots n$. The T_b may be regarded as a "fuzzy truth table."

Then the T_b obey the linear equalities and inequalities

$$0 \leq T_b, \quad \sum_b T_b = 1, \tag{1}$$

$$\sum_{b,\ b_i=1} T_b = x_i, \text{ for } i = 1 \ldots n, \tag{2}$$

and furthermore, any set of T_b obeying these constraints indeed does represent a valid fuzzy truth table.

Therefore, the tightest possible bounds L and U on F, arise from minimizing (resp. maximizing) the values of the *objective function*

$$C = \sum_{b,\ F(b)=1} T_b \tag{3}$$

subject to the inequalities (1-2) above. This is a 2^n-dimensional *linear program*, as promised. By means of Vaidya's linear programming algorithm [5] it may be solved in a number of steps of order $2^{2.5n}N$, where each step involves an arithmetic operation on an integer with $\leq N$ bits, and here $N \leq n \log n$.

3 Examples

As an example, it is rather easily seen from the formalism above that

logical op. F	Tightest possible lower, upper bounds on F
x_1 and x_2 and .. and x_n	$\max(0, x_1 + x_2 + \ldots + x_n + 1 - n)$, $\min(x_1, x_2, \ldots, x_n)$
not u	$1 - u$, $1 - u$
x_1 or x_2 or .. or x_n	$\max(x_1, x_2, \ldots, x_n)$, $\min(1, x_1 + x_2 + \ldots + x_n)$
u xor v	$\lvert u - v \rvert$, $\min(u + v, 2 - u - v)$.

Is there a simple expression for the best possible upper and lower bounds for the fuzzy generalization of parity$(x_1 + x_2 + \ldots + x_n)$?

4 Incorporating more knowledge

If certain conditional probabilities are known, then it is possible to incorporate this extra knowledge into the linear program (1)-(3) to obtain tighter bounds.

This may be accomplished by augmenting the linear program with additional equality constraints. Namely if some probability that some boolean function G of a subset Q of the arguments is true, is known to be p, then we include an additional equality constraint (the left hand side is a sum with $\leq 2^n$ terms)

$$\sum_{b,\, b \cap Q \text{ satisfies } G} T_b = p. \tag{4}$$

Obvious schemes for collecting statistics based on previous observations of the E_i, in order to make predictions of future values of F, suggest themselves.

5 The Point of Maximum Entropy

The *mean* value $\langle C \rangle$ of C (defined in (3)) within the polytope defined by (1-2) would seem to be of some interest, perhaps of more interest than the upper and lower bounds U and L. It is possible to compute this value after triangulating the polytope, but it seems *doubly* exponentially difficult to do so, since the number of simplices in the triangulation could be doubly exponentially large.

As an alternative, I suggest the value C_{maxent} corresponding to the *point of maximum entropy* within the polytope. Namely, define the entropy H of a point in the polytope by the following 2^n-term sum:

$$H = \sum_b T_b \log \frac{1}{T_b}. \tag{5}$$

Observe that this function is a concave-$\cap$ function within the polytope, and this polytope is (of course) a convex region. The problem of maximizing concave-$\cap$ functions within convex regions is well-known to be easily solvable via the Ellipsoid method [2] or the algorithm of [4]. By the use of Vaidya's algorithm [4], we may find this point, accurate to D places of decimals, in a number of steps bounded by $2^{3.38n} N D$. We may then compute C_{maxent} according to (3).

This time bound sounds more impressive when one realizes that this single max-entropy point we have computed, will allow one to find the max-entropy estimate of the value of *any* fuzzified boolean function of the x_i.

(Incidentally, the concave-$\cap$-ness of the entropy function combined with Vaidya's algorithm will yield related results. For example, the information capacity of a discrete memoryless channel [1] with given $n \times n$ transition probability matrix specified using rationals having a total of N bits, may be found in $D \times$polynomial(n, N) steps accurate to D decimals.)

6 References

1. R. G. Gallager: *Information theory and reliable communication,* Wiley 1968.
2. M. Grötschel, L. Lovasz, A. Schrivjer: *Geometric algorithms and combinatorial optimization,* Springer 1988.
3. K.K. Thornber: *A key to Fuzzy Inference*, NECI Technical report 91-049-2-0014-2 (1991); probably will be submitted to J. Approx. Reasoning.
4. P.M. Vaidya: *A new algorithm for minimizing convex functions over convex sets,* 30th IEEE FOCS, (Oct. 30 – Nov. 1 1989, Research Triangle NC) 338-343.
5. P.M. Vaidya: *Speeding up linear programming using fast matrix multiplication*, 30th IEEE FOCS, (Oct. 30 – Nov. 1 1989, Research Triangle NC) 332-337.
6. L.A. Zadeh: *Fuzzy sets*, Information and Control 8 (1965) 338-353.