BACK TO MAIN

Probabilistic Logic

Nilsson, *Artificial Intelligence*, '86

- Suppose we have set β of logical sentences, where sentence *i* has the probability π_i (vector Π). The first sentence is a tautology and has the probability 1. $|\beta|$ = L-1.
- We want to find the probability of sentence *S* given β and Π. This is an entailment problem.
- Given β and *S,* we use a semantic tree to find all sets of possible worlds that are possible. There are *K* sets of possible worlds. A sentence is either true or false in a given set of worlds.
- Let *V* be a *L x K* matrix – rows representing each sentence and columns representing a set of worlds. A position in a matrix is 1 if the sentence for that row is true in the world for that column, and 0 otherwise.
- By omitting the last row of *V (*the row for sentence *S)* we obtain *V'.* Using *V',* we can create a convex hull to determine bounds on the probability of *S* (i.e. the probability of S is a any point in the hull based on the probabilities assigned to the sets of worlds). Linear programming can be used for this.
- Nilsson presents two techniques to approximate the probability of the entailed sentence. One method uses the projection of the vector associated with *S* from *V* projected onto the subspace defined by the row vectors of *V'.* Another method find the probability for sentence *S* that minimizes entropy. However, these techniques are only appropriate for very small matrices.
- For larger matrices, Nilsson creates a systematic way to create a smaller matrix, *V**, that approximates *V'.* This is done by changing values in the vector Π. to either 0 or 1 and checking its consistency. If consistent, the column is added to *V**. The more columns that are added, the more *V** comes to resemble *V'*.