Competitive Learning

Christopher D. Rosin   Richard K. Belew
crosin@cs.ucsd.edu   rik@cs.ucsd.edu
Department of Computer Science and Engineering
University of California, San Diego
La Jolla, CA 92093-0114

1 The Competitive Learning Framework

In a *competitive learning* problem, no absolute measure of performance is available to guide the search for solutions, and the only feedback to an adaptive algorithm is the outcome of competitions between potential solutions. At any point in the search, potential solutions are rated in a tournament with members of a current *test set*. The tournament structure proposed in this paper separates the test set from the search points. This is often not the case for typical competitive learning, in which the genetic algorithm is used, with its population providing the test set and potential solutions simultaneously.

Since no external performance measure is provided in competitive learning, an internally defined absolute performace measure is needed to measure the success of competitive learning. Let $G = (V, E)$ be a directed graph in which the vertices correspond to possible competitors in the search space, and an edge from vertex v_1 to vertex v_2 indicates that v_1 defeats v_2 in the competition. The "relative indegree" of a vertex v, denoted $\sigma(v)$, is defined to be the fraction of all vertices that have edges *to* v; all others have edges *from* v. Note that $\sigma(v)$ is the probability that a competitor chosen randomly from the uniform distribution defeats the competitor corresponding to v. So, define the goal of competitive learning to be a vertex v with minimal $\sigma(v)$.

2 Predicting Performance from Test Set Size

Given a test set, the search seeks a competitor that defeats all members of it. In the competition graph, this corresponds to finding a vertex with edges to every member of the test set. Performance of competitors is bootstrapped by creating a new test set of *independent* solutions to the problem of defeating the old test set, and the procedure is then repeated. It is possible to create a Markov model of such a tournament, with discrete states corresponding to states of the test set. We are currently exploring this, and results are qualitatively similar to those given by the model presented here. However, the number of states in the Markov model rises exponentially with the number of σ values in the graph.

Let $\{\sigma_i\}$ be the set of all values of $\sigma(v)$ for $v \in V$. Let P_0 be a probability distribution over σ_i, so that $P_0(\sigma)$ is the fraction of vertices in V with $\sigma(v) = \sigma$. Assume that members of $\mathcal{T}$, the test set, are chosen independently from a distribution P_1 over σ_i. Now, we wish to calculate the distribution P_2, over σ_i, of indegrees for vertices, v, that defeat all members of the test set. Denote the event of v_a defeating v_b by $v_a \rightarrow v_b$, and let $\mathcal{T} = \{v_1, v_2, \ldots, v_n\}$. The distribution P_2 is given by Bayes' rule. Assuming independence of $\Pr[v \rightarrow v_i \mid \sigma(v) = \sigma \& \sigma(v_i) = \sigma_i]$ and $\Pr[v \rightarrow v_j \mid \sigma(v) = \sigma \& \sigma(v_j) = \sigma_j]$, and defining P_{σ_a,σ_b} to be $\Pr[u \rightarrow w \mid \sigma(u) = \sigma_a \& \sigma(w) = \sigma_b]$,

$$P_2(\sigma) = \frac{P_0(\sigma)(\sum_{\sigma_j}(P_1(\sigma_j)P_{\sigma,\sigma_j}))^n}{\sum_{\sigma_i}(P_0(\sigma_i)(\sum_{\sigma_j}(P_1(\sigma_j)P_{\sigma_i,\sigma_j}))^n)}$$

The important result to be described is the qualitative performace predicted by the model. Here, two σ values are used, $\sigma_a >> \sigma_b$; the P_{σ_1,σ_2} are completely determined in this case by constraints imposed by the graph. In order for the search task to be difficult, $P_0(\sigma_b)$, the frequency of the "desired" σ value, will typically be small.

A new test set distribution, P_{t+1}, is created from the previous distribution, P_t, as described above. The general behavior seen in P_t, as t increases, is a rapid rise in $P_t(\sigma_b)$, converging to a final equilibrium value, $P_\infty(\sigma_b)$. This usually requires a small number iterations, except in rare borderline cases. There is typically a sharp transition in $P_\infty(\sigma_b)$ from near 0 to near 1, as n rises through its discrete values. Most importantly, the size, n, of the test set required to obtain a high $P_\infty(\sigma_b)$, is linear in the logarithm of $P_0(\sigma_b)$, as is shown in the graph below. This is exactly what one hopes for from competitive learning: the computational effort required rises linearly (assuming constant search time), with exponential increase in the difficulty of locating high-performance individuals.