Effective Parallelisation for Machine Learning

Motivation

Effective parallelisations achieve the same confidence and error bounds as the underlying base learning algorithm in much shorter time.

Efficient in the sense of Nick's Class: polylogarithmic runtime on polynomially many processing units.

Any Base Learning Algorithm is parallelised by the Radon Machine

Theoretical Analysis

Theorem: given a consistent and efficient regularised risk minimisation algorithm $L$ on hypothesis space $\mathcal{F}$ with finite Radon number $r$ (and dimension of the data is polylogarithmic in the number of examples), then the Radon Machine $R$ has polylogarithmic runtime on quasi-polynomially many processors.

Proof (sketch):

1. Choose $h \approx ld M - ld ld M$
2. $n \approx M/2^s = ld M$
3. $c = r^h \in \mathcal{O}(M^{ld r})$
4. $T_R \in \mathcal{O}(ld^a M + r^h ld M)$
5. $N \geq N^g (r, \Delta) = r^h N^g (c, \delta) \Rightarrow n = \frac{N}{ld} \leq \frac{M}{2^s}$
6. Runtime Complexity: $T_{R,R}(N) = T_L(n) + hr^d$
7. Sample Complexity: $r = d + 2$

Empirical Evaluation

Speedup over Base Learning Algorithm

Comparison with Spark and Averaging