RANDOMIZED ALGORITHMS IN LINEAR ALGEBRA

A matrix. Can be read from external memory, but too large to be stored in RAM; too large to do Linear Algebra on A.

What can be say about the whole of A from a random sample of rows / columns / entries of A ?

Uniform Sampling

Each entry is equally likely to be picked. One advantage is that “coins” can be tossed “blindly” - before A is read and the picked rows / columns / entries drawn in one “pass” through A.

Many recent results on what we can do with Uni. Sampling.

Interestingly, many of these results have been proved independently by two approaches - Combinatorial and Linear Algebraic.

Example - Maximum Cuts in graphs : Given $n \times n$ symmetric A, with $|A_{ij}| \leq 1$, find $n-$vector attaining (the maximum of a quadratic form) :

$$f(A) = \text{Max}_{x : x_i \in \{-1,1\}} \quad (\underline{1} + x)^T \, A \, (\underline{1} - x).$$

THEOREM If B is a $q \times q$ matrix formed by picking uniformly at random q rows (and same columns) of A, then with probability at least 99/100, we have

$$\left| f(A) - \frac{n^2}{q^2} f(B) \right| \leq \frac{100 \log q}{q^{1/4}} n^2.$$

$$\text{Max}_{x,y : x_i, y_i \in \{-1,1\}} \quad x^T \, A \, y \quad = ||A||_{\infty \to 1}.$$

THEOREM A, B as above. With probability 99/100,

$$\left| ||A||_{\infty \to 1} - \frac{n^2}{q^2} ||B||_{\infty \to 1} \right| \leq \frac{100 \log q}{q^{1/4}} n^2.$$

Easy : $||B|| \geq (***)||A||$. Harder : $||B|| \leq (***)||A||$.

Literature

Max Cut is a special case of a "Constraint Satisfaction Problem" (CSP) with "arity" equal to 2.

Arora, Karger, Karpinski (1995) : In poly time, we can approximate CSP of arity r to **additive error** ϵn^r.

Goldreich, Goldwasser, Ron (1996) Detailed analysis of many arity 2 problems using combinatorial approach. Area has come to be known as Property Testing.

Frieze, Kannan (1996) Poly time algorithms for all CSP problems (any fixed arity r) achieving additive error ϵn^r using Linear Algebra methods to approximate resulting matrices A by low-rank matrices.

For arity $r > 2$, we get $r-$dimensional arrays which are approximated by the sum of a small number of outer-products of vectors (analog to low-rank) to obtain solutions.

The two theorems above are special cases of Theorems (for arbitrary arity) proved in : Alon, de la Vega, Kannan, Karpinski (2002).

The second theorem has been strengthened by Rudelson and Vershynin using some recent techniques of Bourgain and Tzafriri.

A purely combinatorial approach has independently achieved the same algorithmic ends (for CSP 's) as our paper : Andersson and Engebretsen (2002).

Use of non-uniform Sampling

Given an $m \times n$ matrix A, can we find an approximation A' to A (also an $m \times n$ matrix) so that

(i) A' is storable in much less space than A.

(ii) A' can be found from A in few passes of A from external memory.

Yes; indeed enough to store $O(1)$ randomly picked columns and $O(1)$ randomly picked rows of A, but picked according to a certain probability distribution. Also, the columns and rows can be picked in two passes through A.

– Drineas, Kannan, building on results of Frieze, K., Vempala.

A any $m \times n$ matrix. C an $m \times s$ matrix of s random columns of A picked in i.i.d. trials; in each trial,

$$\text{Prob of picking column } j = \frac{\sum_i A_{ij}^2}{\sum_{k,l} A_{kl}^2}.$$

[Probabilities are proportional to length squared of the columns.] Similarly R - $s \times n$ from s random rows of A.

THEOREM From C, R, we can compute a $s \times s$ matrix U so that

$$E\left(||(A - CUR)||_{2\to 2}^2\right) \leq \frac{4}{\sqrt{s}}||A||_F^2$$

$$E\left(||(A - CUR)||_F^2\right) \leq \frac{4}{s^{1/4}}||A||_F^2 +$$

Best we can do with a rank $\sqrt{s}$ matrix.

$$\begin{pmatrix} & & \\ & A & \\ & & \end{pmatrix}$$

$$\approx \begin{pmatrix} \\ C \\ \\ \end{pmatrix} \cdot \begin{pmatrix} U \end{pmatrix} \cdot \begin{pmatrix} & R & \end{pmatrix}$$

Recall SVD yields approximations of the form

$$A \approx SDW^T,$$

where S, W have (small number of) orthonormal columns and D is diagonal. But it takes a long time to compute S, D, W !

Achlioptas and McSherry's algorithm :

p probability. Independently for each entry A_{ij} of matrix, replace it with A_{ij}/p with probability (w.p) p and 0 with probability $1-p$. So, number of non-zero entries reduced by a factor of p.

$$\hat{A}_{ij} = \begin{cases} 0 & \text{w.p.} 1-p \\ A_{ij}/p & \text{w.p.} p. \end{cases}$$

$$\begin{pmatrix} 5 & 3 & 3 & -2 & -7 & 8 & 9 \\ 1 & 2 & 2 & -17 & 1 & -8 & 9 \\ 21 & 41 & 22 & -2 & 0 & 0 & 0 \end{pmatrix} \rightarrow$$

$$\begin{pmatrix} 10 & 6 & 0 & 0 & -14 & 16 & 0 \\ 2 & 4 & 0 & 0 & 0 & -16 & 18 \\ 0 & 0 & 44 & 0 & 0 & 0 & 0 \end{pmatrix}$$

If $|A_{ij}| \leq 1$, WHP, $||A - \hat{A}||_2$ is small.