0368-3248-01-Algorithms in Data Mining **Fall 2013**

Lecture 8: Home Assignment, Due Dec 31st

Lecturer: Edo Liberty

Warning: *This note may contain typos and other inaccuracies which are usually discussed during class. Please do not cite this note as a reliable source. If you find mistakes, please inform me.*

---

1 Randomized meta-algorithms

setup

In this question we assume the common case where we have an input $x \in X$ and we wish to approximate a function $f : X \to \mathbb{R}^+$ (i.e. $\forall x \;\; f(x) \geq 0$). For that we have a black box randomized algorithm $A : X \to \mathbb{R}^+$ such that $\mathbb{E}[A(x)] = f(x)$. The questions ask you to designing meta algorithms using A as a black box.

question

1. Show that
$$\Pr[A(x) \geq 3f(x)] \leq \frac{1}{3}$$

2. Assume that for all x we have that $\mathrm{Var}[A(x)] \leq c \cdot [f(x)]^2$. Describe an algorithm B_2 such that for any two constants $\varepsilon, \delta > 0$:
$$\Pr[|B_2(x) - f(x)| \geq \varepsilon f(x)] \leq \delta$$

3. Assume that $\Pr[|A(x) - f(x)| \leq t] \geq \frac{1}{2} + \eta$ for some fixed value $\eta > 0$. In words, the algorithm gets an additive approximation t with probability slightly better than $1/2$. (Here we do not assume anything on the variance of $A(x)$). Design and algorithm B_3 such that for any prescribed $\delta > 0$
$$\Pr[|B_3(x) - f(x)| \leq t] \geq 1 - \delta$$
That means the algorithm achieves the same additive approximation with probability arbitrary close to one.

answers

1. This is an application of Markov's inequality which is adequate since $A(x) \geq 0$

$$\Pr[A(x) > t] \leq \frac{\mathbb{E}[A(x)]}{t}$$

Setting $t = 3f(x) = 3\mathbb{E}[A(x)]$ completes the claim.

2. Let B_2 be defined as

$$B_2(x) = \frac{1}{s}\sum_{i=1}^{s} A_i(x) \quad \text{for} \quad s = \frac{c}{\varepsilon^2 \delta}$$

where A_i are independent executions of A. Since $A_i(x)$ are independent we have that $\mathbb{E}[B_2(x)] = f(x)$ and $\mathrm{Var}[B_2(x)] = \mathrm{Var}[A(x)]/s$. Substituting this into Chebyshev's inequality we get

$$\Pr[|B_2(x) - f(x)| \geq \varepsilon f(x)] \leq \frac{\mathrm{Var}[A(x)]/s}{\varepsilon^2 f^2(x)} \leq \frac{c/s}{\varepsilon^2} \leq \delta$$

3. Here we can define B_3 as

$$B_3(x) = \mathrm{median}\{A_i(x) | i \in [s]\} \quad \text{for} \quad s \geq \frac{2\log(1/\delta)}{\eta^2}$$

where A_i are again independent executions of A. Let us define $z_i = 1$ if $|A_i(x) - f(x)| \leq t$ and zero else. It is easy to see that if $\sum_{i=1}^{s} z_i \geq s/2$ then the algorithms succeeds. This is because there are at least $s/2$ values $A_i(x)$ in the interval $(f(x) - t, f(x) + t)$. Therefore, the median must be one of those values and must also lie in $(f(x) - t, f(x) + t)$. We now use Chernoff's inequality for $\sum_{i=1}^{s} z_i$ and use the fact that $\mu = \mathbb{E}[\sum_{i=1}^{s} z_i] \geq s(1/2 + \eta)$

$$\Pr[\sum_{i=1}^{s} z_i < s/2] = \Pr[\sum_{i=1}^{s} z_i - s(1/2 + \eta) < -s\eta] \leq \Pr[\sum_{i=1}^{s} z_i - \mu < -s\eta] \leq e^{-\frac{s^2\eta^2}{4\mu}} \leq e^{-\frac{s\eta^2}{2}} \leq \delta$$

We conclude that since $\Pr[\sum_{i=1}^{s} z_i < s/2] \leq \delta$ then $\Pr[\sum_{i=1}^{s} z_i \geq s/2] \geq 1 - \delta$ which is the algorithm's success probability.

2 Set intersections

setup

We have a universe of N items $A = \{a_1, \ldots, a_N\}$ and m subsets $S_i \subset A$, $i \in \{1, \ldots, m\}$. We assume that given a set S_i we can iterate over its elements one by one. The exercise will deal with approximating the size of different unions of these sets. Here you are tasked with designing an algorithm. Your algorithm is allowed to preprocess the sets S in and produce data structures ($preprocess(S)$) It should then be able to take as input a set of indexed $I \subset \{1, \ldots, m\}$ and produce an ε approximation to $|\cup_{i \in I} S_i|$ with probability at least $1 - \delta$ ($estimateUnionSize(I)$). The aim is to create an algorithm which runs in time $o(\sum_{i \in I} |S_i|)$ and requires $o(|\cup_{i \in I} S_i|)$ space. That means that simply iterating through the lists and keeping items in a hash lookup table is not an adequate solution.

1. Describe $preprocess(S)$ which is the preparatory stage of the algorithm and results in our choice of data structures.
2. Describe $estimateUnionSize(I)$ which return an ε approximation to $|\cup_{i \in I} S_i|$ with probability $1 - \delta$.
3. Prove your algorithm's correctness.
4. What is the space usage of your data structures?
5. What is the runtime complexity of $estimateUnionSize(I)$?

answers

1. We first choose $s \geq 8/\varepsilon^2\delta$ hash functions $h_i : a \to [0,1]$ uniformly. For each set S_i of the m sets we compute for each hash function h_j its minimal value over the elements of S_i. Storing these concludes the preprocessing step which requires $O(s\sum_{i=1}^{m}|S_i|)$ hash evaluations and $O(sm)$ storage. Note that here we assume that the number of elements in the universe n is such that $log(n)$ is small enough to be treated as a constant. Otherwise, the hash functions must contain $\Omega(\log(n))$ bits which would give an $O(s\log(n)\sum_{i=1}^{m}|S_i|)$ running time and $O(sm\log(n))$ storage.

2. Once I is received, we compute the s minimal values over the sets S_i s.t. $i \in I$ for each hash function. This is done simply by taking the minimal values from the ones already computed in the preprocessing step. Denoting by x_j this minimal value (for hash function h_j) we return $\frac{1}{\frac{1}{s}\sum_{i=1}^{s}x_i}$.

3. The proof is identical to a proof given in class (and the class notes) so I will not repeat it here. The main statement is that the reciprocal to the mean of $s \geq 8/\varepsilon^2\delta$ minimal hash value over a set of n' objects is an ε approximation to n' with probability at least $1-\delta$. The algorithm clearly computes these minimal values for the set $\cup_{i\in I}S_i$ which completes the proof.

4. The amount of space is as stated before $O(sm) = O(8m/\varepsilon^2\delta)$ or $O(8m\log(n)/\varepsilon^2\delta)$ depending on the computational model.

5. Given that all sm minimal hash values are given in an array with $O(1)$ access time, the amount of time to compute the approximated size of $\cup_{i\in I}S_i$ is $O(s|I|)$.

3 Weak random projections

setup

In this question we will construct a simple and weak proof of the Johnson-Lindenstrauss lemma. Given two vectors $x, y \in \mathbb{R}^d$ we will find two new vectors $x', y' \in \mathbb{R}^k$ such that from x' and y' we could approximate the value of $||x - y||$. The idea is to define k vectors $r_i \in \mathbb{R}^d$ such that each $r_i(j)$ takes a value in $\{+1, -1\}$ uniformly at random. Setting $x'(i) = r_i^T x$ and $y'(i) = r_i^T y$ the questions will lead you through arguing that $\frac{1}{k}||x' - y'||_2^2 \approx ||x - y||_2^2$.

questions

1. Let $z = x - y$, and $z' = x' - y'$. Show that $z'(\ell) = r_\ell^T z$ for any index $\ell \in [1, \ldots, k]$.
2. Show that $E[\frac{1}{k}||z'||_2^2] = E[(z'(\ell))^2] = ||z||_2^2$.
3. Show that
$$\mathrm{Var}[(z'(\ell))^2] \leq 4||z||_2^4.$$
Hint: for any vector w we have $||w||_4 \leq ||w||_2$.
4. From 3 (even if you did not manage to show it) claim that
$$\mathrm{Var}[\frac{1}{k}||z'||_2^2] \leq 4||z||_2^4/k.$$
5. Use 3 and Chebyshev's inequality do obtain a value for k for which:
$$(1 - \varepsilon)||x - y||_2^2 \leq \frac{1}{k}||x' - y'||_2^2 \leq (1 + \varepsilon)||x - y||_2^2$$
with probability at least $1 - \delta$.

answers

1. This is a consequence of the linearity of the operator.

$$z'(\ell) = x'(\ell) - y'(\ell) = r_\ell^T x - r_\ell^T y = r_\ell^T (x - y) = r_\ell^T z$$

2. Since $||z'||_2^2 = \sum_{i=1}^k z'(i)^2$ and since $z'(i)$ are identically distributed we have that $\mathbb{E}[\frac{1}{k}||z'||_2^2] = \mathbb{E}[\frac{1}{k}\sum_{i=1}^k z'(i)^2] = \mathbb{E}[(z'(\ell))^2]$. Now we compute $\mathbb{E}[(z'(\ell))^2]$.

$$\begin{aligned}
\mathbb{E}[(z'(\ell))^2] &= \mathbb{E}[(\sum_{i=1}^d r_\ell(i)z(i))(\sum_{j=1}^d r_\ell(j)z(j))] &(1)\\
&= \mathbb{E}[\sum_{i=1}^d \sum_{j=1}^d r_\ell(i)r_\ell(j)z(i)z(j)] &(2)\\
&= \sum_{i=1}^d \sum_{j=1}^d \mathbb{E}[r_\ell(i)r_\ell(j)]z(i)z(j) &(3)\\
&= \sum_{i=1}^d z(i)^2 = \|z\|^2 &(4)
\end{aligned}$$

The double summation was reduced to a single sum since $\mathbb{E}[r_\ell(i)r_\ell(j)] = 0$ if $i \neq j$. Also, if $i = j$ we have that $\mathbb{E}[r_\ell(i)r_\ell(j)]z(i)z(j) = z(i)^2$

3. To compute $\mathrm{Var}[(z'(\ell))^2]$ we start with computing $\mathbb{E}[(z'(\ell))^4]$.

$$\begin{aligned}
\mathbb{E}[(z'(\ell))^4] &= \mathbb{E}[(\sum_{i=1}^d r_\ell(i)z(i))(\sum_{j=1}^d r_\ell(j)z(j))(\sum_{k=1}^d r_\ell(k)z(k))(\sum_{m=1}^d r_\ell(m)z(m))]\\
&= \mathbb{E}[\sum_{i=1}^d \sum_{j=1}^d \sum_{k=1}^d \sum_{m=1}^d r_\ell(i)r_\ell(j)r_\ell(k)r_\ell(m)z(i)z(j)z(k)z(m)]\\
&= \sum_{i=1}^d \sum_{j=1}^d \sum_{k=1}^d \sum_{m=1}^d \mathbb{E}[r_\ell(i)r_\ell(j)r_\ell(k)r_\ell]z(i)z(j)z(k)z(m)\\
&= \sum_{i=1}^d x(i)^4 + \binom{4}{2} \sum_{i<j} z(i)^2 z(j)^2
\end{aligned}$$

The last transition requires an explanation. The expectation of $r_\ell(i)r_\ell(j)r_\ell(k)r_\ell$ when the power of one of the terms $r_\ell(i)$ is odd is zero. Thus, we are only left with terms of the form $x(i)^4$ and $x(i)^2x(j)^2$. The coefficient of $x(i)^4$ is 1 since there is only one what to obtain it. The coefficient of $x(i)^2x(j)^2$ is $\binom{4}{2}$ since two of the indexes should be i and the two others j. There are $\binom{4}{2} = 6$ to get it. In what comes next we use the fact that:

$$\sum_{i<j} z(i)^2 z(j)^2 = [\sum_{i=1}^d \sum_{j=1}^d z(i)^2 z(j)^2 - \sum_{i=1}^d z(i)^4]/2$$

Picking up where we left off:

$$\begin{aligned}\mathbb{E}[(z'(\ell))^4] &= \sum_{i=1}^{d} x(i)^4 + 6\sum_{i<j} z(i)^2 z(j)^2 \\ &= \sum_{i=1}^{d} x(i)^4 + 3[\sum_{i=1}^{d}\sum_{j=1}^{d} z(i)^2 z(j)^2 - \sum_{i=1}^{d} z(i)^4] \\ &= 3\|z\|_2^4 - 2\|z\|_4^2\end{aligned}$$

Finally we have that

$$\begin{aligned}\mathrm{Var}(z'(\ell)^2) &= \mathbb{E}[(z'(\ell))^4] - \mathbb{E}[(z'(\ell))^2]^2 \\ &= 3\|z\|_2^4 - 2\|z\|_4^2 - (\|z\|_2^2)^2 = 2(\|x\|_2^4 - \|x\|_4^4) \leq 2\|x\|_2^4\end{aligned}$$

4. Since $z'(\ell)$ are independent variables we have that

$$\mathrm{Var}[\frac{1}{k}\|z'\|^2] = \mathrm{Var}[\frac{1}{k}\sum_{\ell=1}^{k} z'(\ell)^2] = \frac{1}{k^2}\sum_{\ell=1}^{k}\mathrm{Var}[z'(\ell)^2] = \frac{1}{k}\mathrm{Var}[z'(\ell)^2] \leq 2\|x\|_2^4/k$$

5. From Chebishev's inequality we have that

$$\Pr[|\frac{1}{k}\|z'\|^2 - \mathbb{E}[\frac{1}{k}\|z'\|^2]| \geq t] \leq \frac{\mathrm{Var}[\frac{1}{k}\|z'\|^2]}{t^2}$$

Substituting $\mathbb{E}[\frac{1}{k}\|z'\|^2] = \|z\|^2$, $t = \varepsilon\|z\|^2$ and $\mathrm{Var}[\frac{1}{k}\|z'\|^2] \leq 2\|x\|_2^4/k$ we get:

$$\Pr[|\frac{1}{k}\|z'\|^2 - \|z\|]| \geq \varepsilon\|z\|] \leq \frac{2\|x\|_2^4/k}{\varepsilon^2\|z\|^4} = \frac{2}{k\varepsilon^2}$$

By setting $k \geq \frac{2}{\varepsilon^2\delta}$ we get that $\Pr[|\frac{1}{k}\|z'\|^2 - \|z\|]| \geq \varepsilon\|z\|] \leq \delta$ which means that $\|z\|(1-\varepsilon) \leq \frac{1}{k}\|z'\|^2 \leq \|z\|(1+\varepsilon)$ with probability at least $1-\delta$.

4 SVD and the power method

setup

Here we will prove some basic facts about singular values, matrices, and the power method. For the reminder of the question we assume $A \in \mathbb{R}^{m \times n}$ is an arbitrary matrix. For convenience and w.l.o.g. assume $m \leq n$ and denote by $\sigma_1 \geq \ldots \sigma_m \geq 0$ the singular values of A. Define the numeric rank of a matrix $\rho(A)$ to be $\rho(A) = \|A\|_F^2 / \|A\|_2^2$. $\rho(A)$ is a smoothed version of the algebraic rank $rank(A)$. It is always true that $1 \leq \rho(A) \leq Rank(A) \leq \min(m, n)$. If $\rho(A) \leq 1 + \varepsilon$ for a sufficiently small ε the matrix is "close" to being of rank 1.

question

1. Let $P \in \mathbb{R}^{m \times m}$ and $Q \in \mathbb{R}^{n \times n}$ be unitary matrices. Show that $\|PAQ\|_F = \|A\|_F$. Hint, begin with the case where one of the matrices P or Q are the identity matrix.

2. Using the above show that for any matrix A we have that

$$\|A\|_F = \sqrt{\sum_{i=1}^{m} \sigma_i^2}.$$

   It might help you to show that $\|A\|_F^2 = tr(AA^T)$ where $tr(\cdot)$ stands for the matrix trace.

3. Give an expression to the numeric rank of A only in terms of its singular values σ_i.

4. Express the numerical rank of $(AA^T)^k A$ only in terms of σ_i.

5. Assume that the matrix A is such that $\sigma_2/\sigma_1 \leq \eta$ for some $\eta < 1$. Use your expressions from above to find k such that $\rho((AA^T)^k A)) \leq 1 + \varepsilon$. How does this relate to the the Power Method for computing the largest singular value and vectors of A?

answers

1. As the hint suggests, let us first show that $\|AQ\|_F = \|A\|_F$ for any matrix A and unitary matrix Q.

$$\|AQ\|_F^2 = \sum_{i=1}^{n} \|(AQ)_i\|_2^2 = \sum_{i=1}^{n} \|A_i Q\|_2^2 = \sum_{i=1}^{n} \|A_i\|_2^2 = \|A\|_F^2$$

   Here $(AQ)_i$ denoted the i'th columns of AQ and A_i denoted the i'th column of A. Repeating this argument for the left side completes the claim.

2. As we saw in class, if we compute the SVD of A we get $A = USV^T$ where U and V are unitary and S is diagonal such that $S_{i,i} = \sigma_i$. Using the above we get that $\|A\|_F = \|S\|_F = \sqrt{\sum_{i=1}^{m} \sigma_i^2}$.

3. The numeric rank of A is defined as $\|A\|_F^2 / \|A\|_2^2$. Since $\|A\|_2^2 = \sigma_1^2$ and $\|A\|_F^2 = \sum_{i=1}^{m} \sigma_i^2$ we have

$$\rho(A) = \frac{\|A\|_F^2}{\|A\|_2^2} = \sum_{i=1}^{m} (\frac{\sigma_i}{\sigma_1})^2$$

4. In term of the SVD of A we have that $(AA^T)^k A = US^{2k+1}V^T$. Therefore the singular values of $(AA^T)^k A$ are equal to σ_i^{2k+1}. Using the equation above we get that

$$\rho((AA^T)^k A) = \sum_{i=1}^{m} (\frac{\sigma_i}{\sigma_1})^{4k+2}$$

5. Rewriting the expression above we get

$$\sum_{i=1}^{m} (\frac{\sigma_i}{\sigma_1})^{4k+2} = (\frac{\sigma_1}{\sigma_1})^{4k+2} + \sum_{i=2}^{m} (\frac{\sigma_i}{\sigma_1})^{4k+2} \leq 1 + m\eta^{4k+2}$$

   Which gives us that $\rho((AA^T)^k A) \leq 1 + \varepsilon$ when $k \geq \log(m/\varepsilon)/4\log(\eta) + O(1)$. This gives another explanation for the success of the power method. The reason is that even for relatively small values of k the matrix $(AA^T)^k A$ is essentially rank 1. This means that the direction of $(AA^T)^k Ax$ is independent of x. It only depends on the left (and only) singular vector of $(AA^T)^k A$. The power method computes $(AA^T)^k Ax$ iteratively and returns the resulting vector (normalized) which is the left singular vector of A corresponding to the top singular value.