CS 354: Unfulfilled Algorithmic Fantasies May 6th, 2019

Lecture 11: Average-Case Complexity - Hardness of SPCA

Lecturer: Aviad Rubinstein *Scribe: Yunsung Kim*

1 Introduction

The topic of this lecture is average-case complexity. The term *average-case* is among the most overloaded terms in complexity theory, so it is important to be able to tell which specific notion of average-case complexity is being used. In particular, following are some contexts in which the term "average-case" is used.

1. **There exists a *natural* distribution of hard instances.** Some examples include the Planted Clique problem (which we will see today), random 3-SAT, and the stochastic block model.
2. **There exists an *efficiently samplable* distribution of hard instances.** In particula for cryptographic applications, it is not enough to know that there exists a hard instance (e.g. a hard-to-decrypt-without-key cyphertext) — we need an efficient algortihm for constructing hard instances.
3. **Finding a function f is (approximately) equal to some f^* on most input.** One example is PAC learning, which we will see on Wednesday. Also in circuit complexity, we would like to come up with a function that small circuits would fail to evaluate on most inputs.
4. **Semirandom models.** These are models that consider instances generated by a mix of adversarial and random processes. A noteable example is *smoothed analysis*.

Another interesting question that is unfortunately very difficult to model in theory, is what is the complexity of a certain problem on the "average" instances encountered in practice.

Today we will see our first average-case complexity result for a problem called Sparse PCA, which is the problem of analyzing the most significant direction (principal component) in a data set. In particular we will demonstrate a distribution over instances for which finding a *sparse* principal component is hard given a fixed number of samples, by reducing to Sparse PCA from another problem called Planted Clique, which is conjectured to be hard in the average-case.

2 Planted Clique Problem

Before discussing Sparse PCA, we will first define the Planted Clique Problem.

Given an Erdös-Renyi graph $G \sim G(n, 1/2)$, the Planted Clique problem concerns with finding a clique of size k. It is easy to find a $k = \log_2(n)$-size clique, but a 40-year-old conjecture by Karp asserts that finding a clique of size $k = (1+\epsilon)\log_2(n)$ is hard[1]:

[1]Notice that this is not a total search problem (i.e., not in TFNP), unlike classes PPAD or PPP. While such a

Conjecture 2.1 ([6]). *Given $G \sim G(n, 1/2)$, finding a clique of size $(1+\epsilon)\log_2(n)$ is hard.*

It is, however, quite easily verifiable that the graph does have a bigger clique of size $k \geq 2\log_2(n)$ with high probability, so there is a gap between the guaranteed size of a clique and an algorithm for finding them.

When we refer to the *Planted Clique problem*, we will refer to the following decision version of the aforementioned problem:

Definition 2.2 (The Planted Clique Problem). The Planted Clique Problem is defined as follows:

- **Input:** A graph G with n vertices sampled from one of the following two distributions:
  - $\mathcal{G}_0 : G \sim G(n, 1/2)$
  - $\mathcal{G}_1 : G \sim G(n, 1/2)$ + a random k-clique
- **Goal:** Distinguish whether G is sampled from $\mathcal{G}_0$ or from $\mathcal{G}_1$.

The distributions $\mathcal{G}_0$ and $\mathcal{G}_1$ are statistically distinguishable for any non-negligible k, and there are also known polynomial time algorithms for the case $k = C\sqrt{n}$. However, no polynomial-time algorithm is known for $k \ll \sqrt{n}$, and in some restricted models of computation, such algorithm provably doesn't exist [1, 4]. This is the average-case hardness assumption that we will use to prove the hardness of SPCA:

Assumption 2.3 (Planted Clique is Hard). *No polynomial time algorithm can solve Planted Clique for $k = O(n^c)$ for any fixed positive $c < 1/2$.*

3 Sparse PCA and its LP relaxation

Having discussed the Planted Clique Problem and its hardness, we will now define Sparse PCA.

Assume we have samples $X_1, ..., X_t \in \mathbb{R}^d$ drawn i.i.d. from some distribution $\mathcal{D}$ (assume that d is very large), and we'd like to analyze these samples. If these samples are 2-dimensional one can easily plot them, but since d is large not all directions can be represented. The idea of PCA is to find the "most important" direction(s) and analyze the samples along these directions at least.

To make more precise the definition of the "most important" direction, consider the unnormalized empirical covariance matrix $\hat{A} = \sum_i X_i X_i^T$ along with the following program:

$$\begin{aligned} \max_{v \in \mathbb{R}^d} \quad & v^T \hat{A} v \\ \text{s.t.} \quad & \|v\|_2 = 1. \end{aligned}$$

The quantity $v^T \hat{A} v$ is called the **explained variance** to denote how much variance in data is explained in a single direction, and the 2-norm constraint guarantees that v indicates a direction.

Notice that this program is very easy - simple linear algebra suffices to find its exact solution. When d is huge, however, this principal component has some practical issues of overfitting and

clique does exist with high probablity, the existence of a clique cannot be verified efficiently: we don't have a way of verifying that the graph was sampled from the Erdös-Renyi distribution, and even if it was there is a small probability that it doesn't have a clique of that size.

uninterpretability. To avoid such problems we can look for a *sparse* vector that tells us something about the data, so we will add an extra m-sparsity constraint to the PCA program:

$$\begin{aligned} \max_{v \in \mathbb{R}^d} \quad & v^T \hat{A} v \\ \text{s.t.} \quad & \|v\|_2 = 1 \\ & \|v\|_0 = m. \end{aligned}$$

This is the program for SPCA. This program, however, is nonconvex and computationally intractable. One algorithmic approach for SPCA is to *relax* the SPCA program.

LP Relaxation of SPCA Consider the following is LP relaxation of SPCA. (It helps to think of $z^+_{i,j} - z^+_{i,j}$ as representing the i,j-th entry of $Z := vv^T$.)

$$\begin{array}{lll} \max & \sum_{i,j} & A_{i,j}(z^+_{i,j} - z^-_{i,j}) \\ \text{s.t.} & & z^+_{i,j}, z^-_{i,j} \quad \geq 0 \\ & \sum_i & (z^+_{i,i} - z^-_{i,i}) \quad = 1 \\ & \sum_{i,j} & (z^+_{i,j} + z^-_{i,j}) \quad \leq m \end{array} \tag{1}$$

The second constraint is equivalent to the unit l_2-norm constraint, and the last constraint is designed to *roughly* control the sparsity of v.

Indeed this LP, although fasible, is only a rough represention of SPCA[2] and it is natural to ask how "good" this relaxation actually is in solving SPCA. For example, would SDP relaxations be "better"?

We first need a way to quantify "goodness" of an algorithm for SPCA. Informally speaking, we can say that this LP relaxation of SPCA is "good" if for some representative distribution of instances, its "capacity" to find the sparse principal component is close to the optimal achievable by any computationally feasible algorithm. In the sections that follow, we will make the notion of such "capacity" more concrete and prove that this LP relaxation is in fact computationally optimal under that notion.

4 The Sparse Component Detection Challenge

Following is one possible statistical framework for understanding the power of algorithms for SPCA:

Definition 4.1 (Sparse Component Detection Challenge)**.** The Sparse Component Detection challenge is defined as follows:

- **Input:** $X_1, ..., X_t \in \mathbb{R}^d$ sampled i.i.d. from one of the following two distributions:
  - $\mathcal{D}_0 : X_i \sim \mathcal{N}(0, I_d)$
  - $\mathcal{D}_1 : X_i \sim \mathcal{N}(0, I_d + \theta vv^T)$

[2]Technically the last constraint isn't even an l_0 constraint at all, although the l_0 constraint does imply this constraint.

- **Goal:** Distinguish[3] whether $X_{1,\ldots,t}$ are sampled from $\mathcal{D}_0$ or from $\mathcal{D}_1$.

Here, v is the special direction in which the variables are correlated. Parameter θ controls how much correlation is present relative to noise, and is called the **signal strength** or **signal-to-noise ratio (SNR)**. If θ is large, a small number of samples will be enough to reveal a large correlation, whereas if θ is small, it requires many samples to detect the presence of a signal. (For this lecture, we will assume for simplicity that a subgaussian distribution is used instead of the actual normal distribution $\mathcal{N}$. It is possible to fix this, but the fix is not trivial[4].)

Within this framework, it is now easy formalize the notion of "capacity" posed earlier. The capacity of an algorithm for SPCA can be measured by the smallest signal strength θ for which it can detect the signal given a fixed number t of samples. If this signal strength threshold θ is small, then the algorithm can detect the signal even in the presence of relatively large noise, so it is a 'good' SPCA algorithm.

This brings up the natural complexity theoretic problem of finding the best possible threshold θ achievable by a polynomial-time algorithm. As a step towards answering this, we will try to make a better sense of how small these θ's are by analyzing (very roughly) the information-theoretic threshold θ_{OPT} for which SPCA is possible, and comparing it with the threshold θ_{LP} of the LP relaxation.

5 Information Theoretically Optimal θ (Rough Analysis)

We will analyze the threshold of the signal strength, below which $\mathcal{D}_0$ and $\mathcal{D}_1$ become information-theroetically indistinguishable in the SPCA program. For the two cases to be distinguishable, we need the maximum of the SPCA program objective $\hat{A} = \sum_i X_i X_i^T$ to be well separated for the two distributions with high probability. Following is a very rough magnitude analysis on the values of the explained variance $v^T \hat{A} v$ when the samples are drawn from $\mathcal{D}_0$ and $\mathcal{D}_1$.

- **Samples are drawn from $\mathcal{D}_0$:** The product is 1 in expectation, and combining the noise from T samples we obtain:

$$v^T \hat{A} v \lesssim 1 + \sqrt{\frac{\log_2 \binom{d}{m}}{t}} \approx 1 + \sqrt{\frac{m \log_2(d)}{t}}$$

  with high probability[5]

- **Samples are drawn from $\mathcal{D}_1$:** In the presence of signal the product is $1+\theta$ in expectation. Considering the noise from T samples, we have:

$$v^T \hat{A} v \gtrsim (1+\theta) - \sqrt{\frac{1}{t}},$$

  again with high probability.

[3] Ideally we would like to recover v in its entirity. Yet, the starting point for reasoning about computational complexity is to consider the decision problem of identifying the presence of a signal.

[4] See [5, 3] for instance.

[5] The $\binom{d}{m}$ factor arises as a result of applying a union bound over the choices of sparse v.

Given this observation, the information-theoretically optimal threshold for θ is rouhgly $\theta_{OPT} \approx O\left(\sqrt{\frac{m \log d}{t}}\right)$.

6 LP Signal Strength (Rough Analysis)

Now let's look at the threshold for the LP relaxation. Since it's a relaxation, at least we know that the threshold θ_{LP} cannot be smaller than θ_{OPT}. Let's look again at the lower and upper bounds of the LP.

- **Samples are drawn from $\mathcal{D}_0$:** Since this is a relaxation of SPCA, the same lower bound from above can be applied[6], namely

$$v^T \hat{A} v \gtrsim (1+\theta) - \sqrt{\frac{1}{t}}.$$

- **Samples are drawn from $\mathcal{D}_1$:** Roughly analyzing the partial sums:

$$\begin{aligned}\sum_{i,j} \hat{A}_{i,j}(z^+_{i,j} - z^-_{i,j}) &= \underbrace{\sum_{i=j} \hat{A}_{i,j}(z^+_{i,j} - z^-_{i,j})}_{\leq \max_i \hat{A}_{i,i}} + \underbrace{\sum_{i \neq j} \hat{A}_{i,j}(z^+_{i,j} - z^-_{i,j})}_{\leq m \cdot \max_{i \neq j} \hat{A}_{i,j}} \\ &\lesssim \left(1 + \sqrt{\frac{\log d}{t}}\right) + \left(m\sqrt{\frac{\log d}{t}}\right) \\ &\approx O\left(m\sqrt{\frac{\log d}{t}}\right)\end{aligned}$$

  The first bound follows since the true variance along the i-th coordinate is 1, and the second follows since the true correlation between coordinates i and j is 0, so we expect to see roughly $\sqrt{1/t}$ with d entries in each of the m signal dimensions.

So the threshold θ_{LP} for the LP relaxation is rouhgly $\theta_{LP} \approx O\left(m\sqrt{\frac{\log d}{t}}\right) = O_d\left(\frac{m}{\sqrt{t}}\right)$, which differs from θ_{OPT} by a factor of $\sqrt{m}$.

7 Average-Case Complexity of SPCA: Reduction from Planted Clique

So now we know the information-theoretic optimal signal strengh threshold θ_{OPT} for SPCA, the corrresponding threshold θ_{LP} for *one* algorithm (the LP relaxation) that solves its relaxed version, and also the fact that they differ by a factor of $\sqrt{m}$. Since the LP relaxation is just one of the many possible relaxation algorithms, maybe there is a better algorithm that achieves a better signal strength threshold?

We will now prove that, under the Planted Clique assumption, no feasible algorithm can beat the signal strength threshold of θ_{LP}.

[6] This is because the optimal v for SPCA is in the feasible region of the LP

Theorem 7.1 ([2]). *Assuming Planted Clique is hard for $k = o(\sqrt{n})$, θ_{LP} is optimal.*

Quite expectably, we will prove this by reducing Planted Clique to SPCA. The reduction from Planted Clique will be similar to the reductions we've seen so far, but rather than mapping YES instances to YES instances and NO instance to NO instances, we will be mapping a *distribution* over the graphs to a distribution over the samples as in Figure 1. It is helpful to think of the reduction as taking an element (a graph) in the support of the original distribution and transforming it into a set of random vectors which, if the original graph is drawn from $\mathcal{G}_0$ (resp. $\mathcal{G}_1$), will be distributed according to $\mathcal{D}_0$ (resp. $\mathcal{D}_1$.)

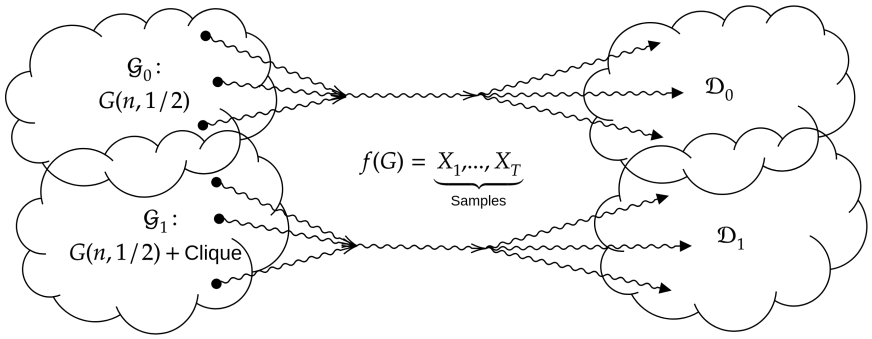


Figure 1: Reduction from Planted Clique to SPCA. We are mapping a pair of distributions over graphs to a pair of distributions over samples $X_1, ..., X_t$. (Notice in the figure that the supports of $\mathcal{G}_0$ and $\mathcal{G}_1$ intersect. In fact, unlike the figure, the support of $\mathcal{G}_0$ contains that of $\mathcal{G}_1$.)

Reduction from Planted Clique Following is the polynomial-time reduction from a graph instance $G = (E, V)$ ($|V| = n$) to a set of random ± 1 vector samples. Let $A \in \{0, 1\}^{n \times n}$ be the adjacency matrix of the random graph G, and let $M = 2A - 1 \in \{\pm 1\}^{n \times n}$. Let $S \in V$ be a random subset of the indices, and define $X = M_{S, V \setminus S}$[7] to be the restriction of M to the rows and columns corresponding to S and $V \setminus S$. We will take the rows of X as our $|S| (\approx t)$ samples. (Refer to Figure 2 for a pictoral illustration of the reduction.)

Analyzing the Reduction We would like to know how the threshold on the clique size k translates to the threshold on the signal strength θ. To do this, we will again compare the optimal values of the objective function $v^T \hat{A} v$ for the two distributions. (Henceforth we will make implicit that the mentioned results hold with high probability.)

Consider the (normalized) empirical variance matrix $\tilde{A} = \frac{1}{|S|} \sum_{l \in S} X_l X_l^T$. When the underlying graph is a plain Erdös-Renyi graph ($G \sim \mathcal{G}_0$), $\tilde{A}$ will have all 1's in the diagonal since the true variance is 1 along each coordinate, and ≈ 0's in the off-diagonal since the true correlation between different coordinates is zero. In this case, it is easy to see that $v^T \tilde{A} v \approx 1$ for any choice of any unit v.

When the underlying graph has a planted clique ($G \sim \mathcal{G}_1$), $|\text{clique} \cap S|$-many samples X_l will have $|\text{clique} \setminus S|$-many 1's in the coordinates corresponding to the vertices in $(\text{clique} \cap S)$. So when

[7] Here the columns are chosen to be $V \setminus S$ to guarantee that X has the right amount of intersection with the clique. This raises an issue of samples not being i.i.d.. We will briefly discuss a fix for this later.

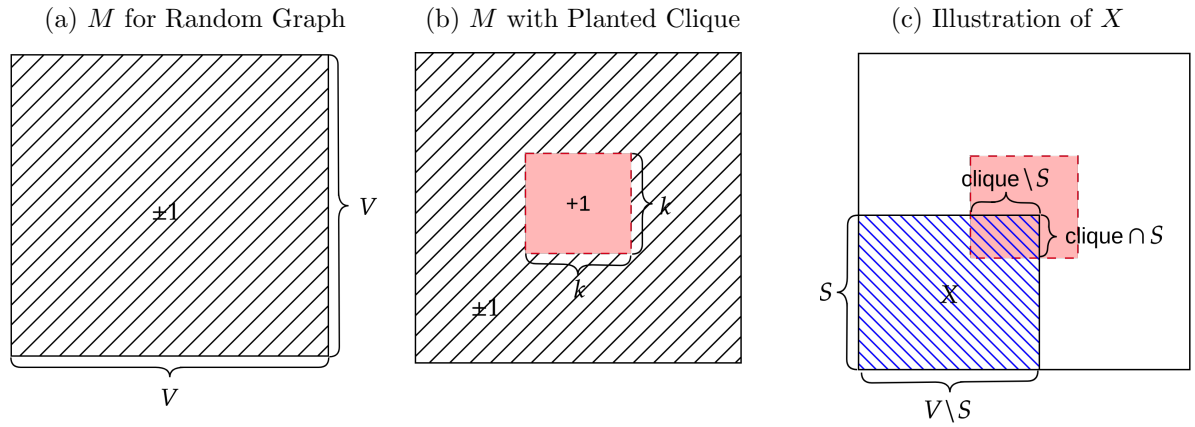


Figure 2: Pictoral illustration of the reduction. When G is a random Erdös-Renyi graph, M is a random ± 1 matrix (except for the main diagonal which is all 1's), whereas when G has a planted clique of size k, M has a block of ones of size $k \times k$ shown as the red square. (W.l.o.g. the square is positioned at the center for the convenience of illustration.) The sampling amounts to selecting a random subset S of the vertices and taking the rows of the rectangular submatrix defined by $S \times (V \setminus S)$

G is drawn from $\mathcal{G}_1$, $\tilde{A}$ will also have all 1's in the diagonal and ≈ 0 values in the off-diagonal, but in addition to that it will have a submatrix of dimension $|\text{clique} \setminus S| \times |\text{clique} \setminus S|$ whose values are $\approx \dfrac{|\text{clique} \cap S|}{|S|}$. (Refer to Figure 3 for an illustration.) In this case, $v^T \tilde{A} v$ is maximized when v^* takes values $\dfrac{1}{\sqrt{|\text{clique} \cap S|}}$ on indices corresponding to vertices in $(\text{clique} \cap S)$ and 0 elsewhere (Figure 3). So we get

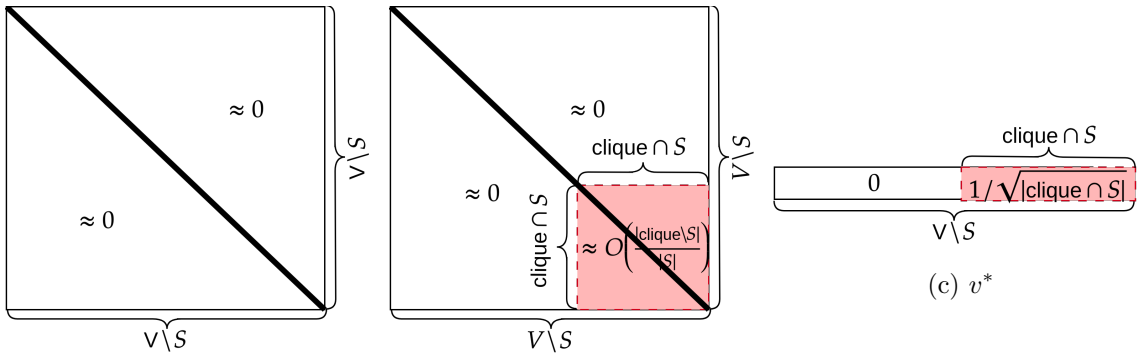
$$\begin{aligned} v^* \tilde{A} v^* &\approx 1 + \frac{|\text{clique} \setminus S|}{|S|} \cdot |\text{clique} \cap S|^2 \cdot \left(\frac{1}{\sqrt{|\text{clique} \cap S|}} \right)^2 \\ &= 1 + \frac{|\text{clique} \setminus S| \cdot |\text{clique} \cap S|}{|S|}. \end{aligned}$$

Very crudely speaking, $|S| \approx n$, $|\text{clique} \cap S| \approx m$, and $|\text{clique} \setminus S| \approx \Theta(k)$, so we finally have

$$v^* \tilde{A} v^* \approx 1 + \Theta\left(\frac{mk}{n} \right).$$

Therefore, the signal strength threshold is roughly in the order of $\Theta\left(\dfrac{mk}{n} \right)$. Since $k = \Omega(\sqrt{n})$ by the planted clique assumption, the optimal signal strength threshold is thus $\Omega\left(\dfrac{m}{\sqrt{n}} \right)$, which mathces the upper bound $\theta_{LP} = O_d\left(\dfrac{m}{\sqrt{t}} \right)$.

Fixing A Small Cheat: Samples Are Not I.I.D. There is one issue that we still need to fix in this reduction. In SPCA the samples $X_1, ..., X_t$ are drawn i.i.d., but in this reduction the



(a) G is a random graph (b) G has planted clique (c) v^*

Figure 3: Pictoral illustration of the empirical covariance matrix $\tilde{A}$ when (a) $G \sim \mathcal{G}_0$ and (b) $G \sim \mathcal{G}_1$, and (c) the shape of v^*. The thick black diagonal indicates that the diagonal entries in both matrices are all-1's. Off-diagonal entries are close to 0 in both matrices, but when G has a planted clique, the resulting empirical covariance matrix has a block (shaded in red) whose entries are $\approx \frac{|\text{clique} \cap S|}{|S|}$

samples generated are not i.i.d. for a few reasons: (1) there is only a limited number of clique rows that can be sampled, and (2) even worse, the number of sampled rows that correspond to the clique affects the number of columns that correspond to the clique.

One fix to this issue is to choose two small disjoint random sets S and S' (which are small constant fractions - say 1% - of $|V|$) instead of S and $V \backslash S$ as the rows and columns of X. There still is some correlation between the rows and columns, but the correlation is very small, and the same analysis that we did above can follow. What we now need to show is that the distribution we get from this reduction is 'close' to the distribution we get from the reduction we mentioned, and that a low error probability in signal detection in the fix indicates a low error probability in the original reduction.

References

[1] Boaz Barak, Samuel B. Hopkins, Jonathan A. Kelner, Pravesh Kothari, Ankur Moitra, and Aaron Potechin. A nearly tight sum-of-squares lower bound for the planted clique problem. In *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 428–437, 2016.

[2] Quentin Berthet, Philippe Rigollet, et al. Optimal detection of sparse principal components in high dimension. *The Annals of Statistics*, 41(4):1780–1815, 2013.

[3] Matthew Brennan, Guy Bresler, and Wasim Huleihel. Reducibility and computational lower bounds for problems with planted sparse structure. In *Conference On Learning Theory, COLT 2018, Stockholm, Sweden, 6-9 July 2018.*, pages 48–166, 2018.

[4] Vitaly Feldman, Elena Grigorescu, Lev Reyzin, Santosh Srinivas Vempala, and Ying Xiao. Statistical algorithms and a lower bound for detecting planted cliques. *J. ACM*, 64(2):8:1–8:37, 2017.

[5] Chao Gao, Zongming Ma, and Harrison H Zhou. Sparse cca: Adaptive estimation and computational barriers. *The Annals of Statistics*, 45(5):2074–2101, 2017.

[6] Richard M. Karp. The probabilistic analysis of some combinatorial search algorithms. 1976.