Maximizing quadratic programs: extending Grothendieck's inequality

Moses Charikar[*]
Princeton University

Anthony Wirth[†]
Princeton University

Abstract

This paper considers the following type of quadratic programming problem. Given an arbitrary matrix A, whose diagonal elements are zero, find $x \in \{-1,1\}^n$ such that $x^T A x$ is maximized. Our approximation algorithm for this problem uses the canonical semidefinite relaxation and returns a solution whose ratio to the optimum is in $\Omega(1/\log n)$. This quadratic programming problem can be seen as an extension to that of maximizing $x^T A y$ (where y's components are also ± 1). Grothendieck's inequality states that the ratio of the optimum value of the latter problem to the optimum of its canonical semidefinite relaxation is bounded below by a constant.

The study of this type of quadratic program arose from a desire to approximate the maximum correlation in correlation clustering. Nothing substantive was known about this problem; we present an $\Omega(1/\log n)$ approximation, based on our quadratic programming algorithm.

We can also guarantee that our quadratic programming algorithm returns a solution to the MAXCUT *problem that has a significant advantage over a random assignment.*

1. Introduction

In this paper we describe an approximation algorithm for a fairly general type of quadratic programming problem. Given matrix A, with null diagonal entries, maximize

$$\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j, \quad \text{s.t. } x_i \in \{-1,1\} \text{ for all } i\,, \qquad (1)$$

a problem we call MAXQP. We enforce the $a_{ii} = 0$ condition because the terms $a_{ii} x_i x_i$ are equal to a_{ii} and so are just additive constants.[1] It is important to note that, for $i \neq j$, the a_{ij} terms are arbitrary real values and are not restricted to being nonnegative. The goal is to return a solution that is at least some fraction α of the optimum of (1), $\mathsf{OPT}_{\mathsf{QP}}$. Our approximation problem is well-defined, for $\mathsf{OPT}_{\mathsf{QP}}$ is strictly positive unless A is the zero matrix (see Lemma 4).

There exists a body of work that has developed approximation algorithms for maximizing generic quadratic programs. However, they use a slightly different definition of approximation algorithm [16], partly because they do not assume that the a_{ii} values are zero. The difficulty of allowing negative a_{ii} values is that the ratio between the semidefinite relaxation and the integral optimum could become arbitrarily large. Therefore, rather than trying to achieve an objective value some fraction α of the optimum, their algorithms achieve *relative accuracy* μ if the objective value that is returned, val, satisfies

$$\frac{\mathsf{val} - \mathsf{MIN}}{\mathsf{MAX} - \mathsf{MIN}} \geq \mu\,,$$

where MAX (MIN) is the maximum (minimum) value of all feasible solutions of the quadratic program. Nesterov [16] presented an SDP-based algorithm with relative accuracy $\pi/2-1$ for quadratic programming. Unfortunately these relative accuracy algorithms guarantee nothing in terms of the usual type of approximation factor. For example, a MAXQP instance corresponding to a complete unweighted MAXCUT problem has $\mathsf{MAX} \in O(n)$, but $\mathsf{MIN} \in -\Omega(n^2)$, and so a constant μ algorithm would not even guarantee returning a positive solution. We note in passing that Nesterov [16] described a $2/\pi$ approximation algorithm, in the usual sense, for instances in which A is positive semidefinite.

Our formulation of MAXQP was partly inspired by a problem in correlation clustering [3], which we detail below. Alon and Naor [2] provided further inspiration with their success in approximating the CUTNORM by trying to

* Email: `moses@cs.princeton.edu`. Supported by NSF ITR grant CCR-0205594, DOE Early Career Principal Investigator award DE-FG02-02ER25540, NSF CAREER award CCR-0237113 and an Alfred P. Sloan Fellowship.

† Email: `awirth@cs.princeton.edu`. Supported by a Gordon Wu Fellowship and NSF ITR grant CCR-0205594.

1 Actually it would not harm any of our arguments if we were to allow the a_{ii} values to be nonnegative, but the exposition is simpler if we just ignore these terms.

maximize

$$\sum_{i=1}^{m}\sum_{j=1}^{n} a_{ij}x_i y_j, \quad \text{s.t. } x_i, y_j \in \{-1,1\} \text{ for all } i,j\,. \qquad (2)$$

We can cast (2) as an instance of (1) by letting

$$z = \begin{pmatrix} x \\ y \end{pmatrix} \quad \text{and} \quad \hat{A} = \begin{bmatrix} 0 & A \\ 0 & 0 \end{bmatrix}, \qquad (3)$$

so (2) is now $z^T\hat{A}z$. Seeing that MAXQP is an extension of Alon and Naor's problem, we spend some time detailing the similarities and differences between them.

A quadratic program may be used to model a graph optimization problem in which a_{ij} represents some property of the edge between i and j. One immediate observation from (3) is that if (2) represents the maximization of a function on a *bipartite* graph, then (1) represents the maximization of the same function on a *complete* graph.

Alon and Naor's problem has two key properties that ours does not. Firstly, different rounding techniques can be used for the $\{x_i\}$ and the $\{y_j\}$ in (2), and secondly, (2) has $\mathsf{MAX} = -\mathsf{MIN}$ (simply replace y with $-y$ in one of the extreme solutions). It turns out that these facts are crucial to their analysis.

Both (1) and (2) have canonical semidefinite relaxations, respectively,

$$\begin{array}{ll} \max & \sum_{i,j} a_{ij}\, v_i \cdot v_j \\ \text{s.t.} & v_i \cdot v_i = 1 \quad \text{for all } i \\ & v_i \in \mathbb{R}^n\,, \end{array} \qquad (4)$$

and

$$\begin{array}{ll} \max & \sum_{i,j} a_{ij}\, u_i \cdot v_j \\ \text{s.t.} & u_i \cdot u_i = v_j \cdot v_j = 1 \quad \text{for all } i,j \\ & u_i, v_j \in \mathbb{R}^n\,. \end{array} \qquad (5)$$

Unfortunately the term-by-term analysis that Goemans and Williamson [10] used in their 0.878 SDP-based approximation algorithm for MAXCUT—which is, in fact, a shifted special case of MAXQP—fails for both (4) and (5) because some of the terms might be negative. Nevertheless, Megretski [15] showed that the ratio of $\mathsf{OPT}_{\mathsf{QP}}$ to the optimum of the SDP relaxation (4) is in $\Omega(1/\log n)$.[2] Grothendieck's inequality [11], a key result in functional analysis, states that the integrality gap between (5) and (2) is in fact a constant. The exact value of this constant is not known, but Alon and Naor converted proofs of the existence of the constant bound [17, 14]—which are also bounds on its size—into approximation algorithms for (2).

Alon and Naor's first algorithm is deterministic: an explicit set of fourwise independent vectors in $\{-1,1\}^n$ is constructed. They showed that there exists one such vector whose projections onto the optimal SDP solution vectors, truncated to lie in $[-1,1]$, give a fractional solution no less than $1/27$ of optimum (see the beginning of Section 2 to see why a good fractional solution is sufficient). The second algorithm is simply the random hyperplane split of Goemans and Williamson, but with a more involved analysis. This technique treats the x and y variables identically, but the analysis cannot be applied to our problem because it relies on the ratio of MAX to $-\mathsf{MIN}$ being at least some constant. The third algorithm shows that there exist two new families of vectors, $\{u'_i\}$ and $\{v'_j\}$, so that the expected value of every $x_i y_j$ term obtained by splitting the u' and v' vectors with a random hyperplane is some common constant multiple of the corresponding $u_i \cdot v_j$ value. These u' and v' vectors are found by maximizing a semidefinite program. None of these techniques seems to apply directly to MAXQP, though we adapt some of the ideas.

Correlation clustering [3], which we also referred to as *clustering with qualitative information* in earlier work [6], is a relatively new problem. We are provided with pairwise judgments of the similarity of n data items. In the simplest version of the problem there are three possible inputs for each pair: similar (aka *positive*), dissimilar (aka *negative*), or no judgment. We refer to an instance in which a judgment is given for every pair as *complete*, otherwise it is a *general* instance.

In the original paper of Bansal, Blum and Chawla [3], the aim was to partition the items into clusters so that the number of pairs in agreement with the judgments would be maximized (MAXAGREE). A pair is called an *agreement* if it is a positive pair within one cluster or a negative pair across two distinct clusters. In like manner, we might want to minimize the number of disagreements (MINDISAGREE), where a *disagreement* is a positive pair across two different clusters or a negative pair in one cluster. These two problems are equivalent from an optimization point of view, but rather different from an approximation point of view. For general instances, there exist constant factor algorithms [6, 18] for MAXAGREE, but MINDISAGREE is as hard as the minimum multicut problem and (consequently) only $O(\log n)$ algorithms are known [6, 7, 8].

Bansal, Blum and Chawla [3] propose the open problem of maximizing the *correlation* in correlation clustering (MAXCORR). The correlation is the difference between the number of agreements and the number of disagreements. Their only observation was that the optimal value of this quantity is always in $\Omega(n)$ for a complete instance; no nontrivial approximation guarantees were known for MAXCORR. Based on our approximation algorithm for MAXQP, we obtain an algorithm for MAXCORR.

2 Our approximation algorithm in Section 2 is an algorithmic version of Megretski's proof.

1.1. Our results

Our algorithm for MAXQP is the first approximation algorithm, in the usual sense, for maximizing a generic quadratic program. The algorithm uses a standard semidefinite relaxation [10], bears some similarity to the first algorithm of Alon and Naor [2], and has approximation factor in $\Omega(1/\log n)$. Our rounding procedure does not simply assign ± 1 values based on a random hyperplane cut of the vectors, but takes into account the sizes of the projections of a random vector onto the solution vectors. If necessary, these projections are truncated to lie in the $[-1, 1]$ interval; from these truncated values an integral solution is obtained using randomized rounding.

A random assignment of items in a MAXCUT instance will on average result in a cut of $1/2$ of the total weight of edges. We say that a cut has *gain* δ if a $1/2 + \delta$ fraction of the total weight of edges lie across the cut. The gain is analogous to the idea of the advantage over a random assignment [13]. Our MAXQP algorithm provides an $\Omega(1/\log(1/\delta))$ approximation for the optimal MAXCUT gain δ. Since MAXCUT is a *shifted* special case of MAXQP, it is not difficult to show that it is NP-hard to obtain an approximation algorithm for MAXQP with factor higher than $11/13$.

The MAXCORR problem restricted to two clusters is a special case of MAXQP, in which a_{ij} is 1 for each positive pair ij and a_{ij} is -1 for each negative pair ij. We show that taking the better of the singleton-clusters solution and the best two-cluster solution provides a $1/3$ approximation for the general MAXCORR problem. Therefore, we obtain an $\Omega(1/\log n)$ approximation for MAXCORR, the first (nontrivial) approximation algorithm known.

1.2. Organization

We present our approximation algorithm for MAXQP in Section 2. Then in Section 3 we explore the relationship between MAXCUT and quadratic programming, and provide some further intuition for our approximation algorithm. In Section 4 we show how our algorithm can be applied to the MAXCORR problem. Finally, we present some open problems in Section 5.

2. Maximizing quadratic programs

We start by showing that the optimum value of MAXQP, modified so that the variables are allowed to take any value in the range $[-1, 1]$, is no larger than that of the original problem (1). Consider the following randomized rounding technique for some fractional solution $y \in [-1, 1]^n$:

$$x_i = \begin{cases} -1, & \text{with probability } \frac{1-y_i}{2} \\ +1, & \text{with probability } \frac{1+y_i}{2} \end{cases} , \qquad (6)$$

where each x_i is rounded independently of the others. A simple calculation shows that, for $i \neq j$, $\mathbf{E}[x_i x_j] = y_i y_j$; since the a_{ii} terms are zero, the expected objective value of the *integral* solution equals that of the fractional solution.

So, to the approximation algorithm. We first solve the semidefinite relaxation (4) of the quadratic program in polynomial time, up to arbitrary precision. We will then round the SDP *vector* solution $\{v_i\}$ to a fractional solution y. It may be that some y_i values fall outside the range $[-1, 1]$; if so, we will truncate them to ± 1. We show that this happens so rarely that the truncation does not alter the expected value of the solution significantly. Finally, we will use the rounding technique (6) in the previous paragraph to obtain a $\{-1, 1\}$ solution x.

ApproxMaxQP

1. Obtain an optimal solution $\{v_i\}$ to the SDP (4).
2. Create vector r in which the r_i are drawn independently from the unit Normal distribution.
3. Let $z_i = v_i \cdot r/T$, where $T > 0$ will be specified later.
4. If $|z_i| > 1$, then $y_i = z_i/|z_i|$, otherwise $y_i = z_i$.
5. Obtain x_i from y_i using rounding procedure (6).

The solution x is clearly in $\{-1, 1\}^n$, and it was obtained in polynomial time, so the remainder of this section will show:

Theorem 1 *The ratio of the expected value of the solution x returned by* ApproxMaxQP *to the maximum value of the quadratic program* (1) *is in* $\Omega(1/\log n)$, *if* $T = \sqrt{4 \log n}$.

Lemma 1 *The expected value of* $z_i z_j$ *is* $v_i \cdot v_j / T^2$.

Proof: Since the distribution of the r vector is *spherically symmetric*, we can assume that $v_i = e_1$ and $v_j = ae_1 + be_2$. Therefore $Tz_i = v_i \cdot r = r_1$ and $Tz_j = v_j \cdot r = ar_1 + br_2$. Hence

$$\begin{aligned} T^2 \mathbf{E}[z_i z_j] &= a\mathbf{E}[r_1^2] + b\mathbf{E}[r_1 r_2] \\ &= a\mathbf{Var}[r_1] + b\mathbf{E}[r_1]\mathbf{E}[r_2] \\ &= a = v_i \cdot v_j . \end{aligned}$$

□

If we were lucky and every $|z_i|$ were at most 1, then we would have a $1/T^2$ approximation, since the optimum value of (4) is at least the optimum of (1). Since this might not happen, we need to analyze the truncated solution y. We show that the expected value of $\Delta_{ij} = z_i z_j - y_i y_j$ is small in magnitude.

Lemma 2 $|\mathbf{E}[\Delta_{ij}]|$ *is less than* $8e^{-T^2/2}$.

Proof: Let us consider the expected value of $|\Delta_{ij}|$ on various regions (of r). We assume that n is sufficiently large that $T \geq 1$. On the region $S = \{r : |z_i| \leq 1, |z_j| \leq 1\}$, $y_i = z_i$ and $y_j = z_j$, so $\mathbf{E}_S[\Delta_{ij}] = 0$.

Now, due to rotational symmetry, we may again assume that $v_i = e_1$ and $v_j = ae_1 + be_2$. So the probability that

r lies in the region $B = \{r : z_i > 1\}$ is $\mathbf{Pr}[r_1 > T] = 1 - \Phi(T)$, where Φ is the cdf for the Normal distribution. Therefore,

$$\mathbf{E}_B[|y_i y_j|] \le \mathbf{E}_B[1] = \mathbf{Pr}[r \in B] = 1 - \Phi(T)\,. \quad (7)$$

Furthermore,

$$\mathbf{E}_B[T^2|z_i z_j|] = \int_{-\infty}^{+\infty}\int_{T}^{+\infty} |s(as+bt)|\frac{1}{2\pi}e^{-s^2/2}e^{-t^2/2}ds\,dt\,. \quad (8)$$

Let us consider each term of (8) one at a time,

$$\int_T^\infty s^2 e^{-s^2/2}ds = -se^{-s^2/2}\Big|_T^\infty + \int_T^\infty e^{-s^2/2}ds = Te^{-T^2/2} + \sqrt{2\pi}(1-\Phi(T))\,.$$

Also,

$$\int_T^\infty |s|e^{-s^2/2}ds = e^{-T^2/2} \quad \text{and} \quad \int_{-\infty}^{+\infty} |t|e^{-t^2/2}dt = 2\,.$$

Putting it all together, we see that

$$\begin{aligned}\mathbf{E}_B[T^2|z_i z_j|] &\le \left(\frac{|a|T}{\sqrt{2\pi}} + \frac{|b|}{\pi}\right)e^{-T^2/2} \\ &\quad + |a|(1-\Phi(T)) \\ &< Te^{-T^2/2} + (1-\Phi(T)),\end{aligned} \quad (9)$$

as $|a|, |b| \le 1$. Since $T \ge 1$, combining (7) and (9), we have

$$\mathbf{E}_B[|\Delta_{ij}|] \le \mathbf{E}_B[|z_i z_j| + |y_i y_j|] < \frac{e^{-T^2/2}}{T} + 2(1-\Phi(T))\,,$$

where the first inequality is merely the triangle inequality.

By symmetry, we will have the same result on the region $\{r : z_i < -1\}$. As there was nothing special about i, the same bound also applies for the regions $\{r : z_j > 1\}$ and $\{r : z_j < -1\}$. The union of these four regions is the complement of the set S. Since the function $|\Delta_{ij}|$ is nonnegative, its expectation on $\overline{S}$ is less than

$$\frac{4}{T}e^{-T^2/2} + 8(1-\Phi(T))\,. \quad (10)$$

But $\mathbf{E}_S[\Delta_{ij}]$ is 0, so (10) is a bound on $\mathbf{E}[|\Delta_{ij}|]$, and hence on $|\mathbf{E}[\Delta_{ij}]|$. We can bound the second term of (10) by

$$4\int_T^\infty te^{-t^2/2}dt = 4e^{-T^2/2}\,,$$

as $T \ge 1$. Therefore, $|\mathbf{E}[\Delta_{ij}]| < 8e^{-T^2/2}$. □

We now show that we are close to obtaining an $\Omega(1/\log n)$ approximation. Let $\mathsf{OPT}_{\mathsf{SDP}}$ stand for the optimum value of (4).

Lemma 3

$$\mathbf{E}[\sum_{i,j} a_{ij}y_i y_j] > \frac{\mathsf{OPT}_{\mathsf{SDP}}}{T^2} - 8e^{-T^2/2}\sum_{i,j}|a_{ij}|$$

Proof:

$$\mathbf{E}[\sum_{i,j} a_{ij}y_i y_j] = \mathbf{E}[\sum_{i,j} a_{ij}z_i z_j] + \mathbf{E}[\sum_{i,j} a_{ij}(-\Delta_{ij})]$$

From Lemma 1, we know that the first term of the right hand side is $\mathsf{OPT}_{\mathsf{SDP}}/T^2$. The second term is

$$\begin{aligned}-\mathbf{E}[\sum_{i,j} a_{ij}\Delta_{ij}] &= -\sum_{i,j} a_{ij}\mathbf{E}[\Delta_{ij}] \\ &\ge -\left|\sum_{i,j} a_{ij}\mathbf{E}[\Delta_{ij}]\right| \\ &\ge -\sum_{i,j}|a_{ij}||\mathbf{E}[\Delta_{ij}]|\,,\end{aligned}$$

which Lemma 2 proves is greater than $-8e^{-T^2/2}\sum_{i,j}|a_{ij}|$. □

The obvious next step is to show that this *error* term is insignificant. Recall that $\mathsf{OPT}_{\mathsf{QP}}$ stands for the optimum value of (1).

Lemma 4

$$\mathsf{OPT}_{\mathsf{QP}} \ge \frac{1}{n}\cdot\sum_{i,j}|a_{ij}|$$

Proof: Consider constructing a random matching on an n-vertex complete graph in the following way. Select an edge uniformly at random, remove the endpoints from the graph, and repeat. It is easy to show that the probability of an edge being included in the matching is $1/(n-1)$ if n is even and $1/n$ if n is odd. Now, if we assign to each edge the weight $|a_{ij}|$, then there exists some matching on the n vertices of total weight at least $\sum_{i,j}|a_{ij}|/n$ (the expected value under this random construction).

Given a matching, we randomly construct a vector $x \in \{-1,1\}^n$ whose expected QP objective value is the same as the matching weight. For each edge ij in the matching, we independently set x_i to ± 1 uniformly at random; we let x_j equal x_i if and only if a_{ij} is nonnegative. The unmatched vertex, if n is odd, is assigned a value independently. For each matched pair ij we score $|a_{ij}|$, for every other pair the expected score is 0, hence the same total as the matching. Therefore $\mathsf{OPT}_{\mathsf{QP}}$ is at least $\sum_{i,j}|a_{ij}|/n$.

□

We can now prove Theorem 1.

Proof: Substituting Lemma 4 into the statement of Lemma 3, with $T = \sqrt{4\log n}$, we see that

$$\mathbf{E}[\sum_{i,j} a_{ij}y_i y_j] > \mathsf{OPT}_{\mathsf{QP}}[\frac{1}{4\log n} - 8n^{-2}\cdot n]\,,$$

which is in $\Omega(\mathsf{OPT_{QP}}/\log n)$. Finally, note that

$$\mathbf{E}[\sum_{i,j} a_{ij} x_i x_j] = \mathbf{E}[\mathbf{E}[\sum_{i,j} a_{ij} x_i x_j \mid y]] = \mathbf{E}[\sum_{i,j} a_{ij} y_i y_j].$$

□

3. Observations

3.1. The relationship with MAXCUT

Maximizing the gain in the MAXCUT problem can easily be formulated as a quadratic program. Let w stand for the total weight of edges in a MAXCUT instance and δ stand for the gain of the optimal cut—recall that a cut of size $w(1/2+\delta)$ has gain δ. The Goemans-Williamson algorithm guarantees only that its solution will have a cut of at least $w(0.439 + 0.878\delta)$, which may have no gain at all. Zwick's outward rotations method [20] or Feige and Langberg's RPR^2 algorithm for LIGHTMAXCUT [9] might approximate the gain well. In fact, our algorithm is an instantiation of Feige and Langberg's rounding scheme, using an s-linear function. These authors do not, however, present any theoretical analysis that applies to our setting.

We can express MAXCUT as a special case of quadratic programming, by setting A as follows. For all pairs $i < j$ for which there exists an edge of weight w_{ij}, let $a_{ij} = -w_{ij}$; for all other values of i and j, let $a_{ij} = 0$. For a given solution $x \in \{-1,1\}^n$, the value of the quadratic program $q(x)$, the value of the cut $k(x)$ and the value of its gain $g(x)$, satisfy $q(x) = 2k(x) - w = 2w\, g(x)$.

Lemma 5 *If δ^* is the optimum gain of a* MAXCUT *instance,* ApproxMaxQP *returns a solution whose gain is in*

$$\Omega\left(\frac{\delta^*}{\log(1/\delta^*)}\right).$$

Proof: Lemma 3 says that *ApproxMaxQP* will return a solution x for which $q(x)$ is at least

$$\frac{\mathsf{OPT_{QP}}}{T^2} - 8e^{-T^2/2} w\,, \tag{11}$$

as w is the sum of the $|a_{ij}|$ terms. By definition, $\mathsf{OPT_{QP}} = 2w\delta^*$, so if we set $T = \sqrt{32\log(1/\delta^*)}$ then it is not hard to show that (11) is greater than

$$\frac{\mathsf{OPT_{QP}}}{64\log(1/\delta^*)}, \quad \text{for all } \delta^* \le 1/2\,.$$

Since the gain, $g(x)$, is a constant multiple of $q(x)$, the result follows. □

Just as we have a reduction to obtain an approximation algorithm for maximizing the gain of a cut, we can also obtain a hardness result for MAXQP from one for MAXCUT.

Lemma 6 *It is NP-hard to approximate* MAXQP *within factor $11/13 + \varepsilon$.*

Proof: Let α now stand for the hardness factor for MAXCUT. Since $\mathsf{OPT_{QP}} = 2\mathsf{OPT_{CUT}} - w$, distinguishing between the cases $\mathsf{OPT_{CUT}} \ge k$ and $\mathsf{OPT_{CUT}} < \alpha k$ is equivalent to distinguishing between $\mathsf{OPT_{QP}} \ge 2k - w$ and $\mathsf{OPT_{QP}} \le 2\alpha k - w$. The ratio of these two bounds on $\mathsf{OPT_{QP}}$ is

$$\frac{2\alpha k - w}{2k - w} = \alpha + w\frac{\alpha - 1}{2k - w}\,.$$

The lemma follows from the $16/17 + \varepsilon$ hardness result for MAXCUT [12, 19], which holds for $k = 17w/21$. □

3.2. Correlated random variables and distributions on cuts

We make some general observations about rounding solutions to the MAXQP SDP relaxation and point out some interesting connections to generating $\{-1,1\}$ random variables with given correlations. The value of the SDP solution is $\sum_{i,j} a_{ij} v_i \cdot v_j$. Suppose we could generate $\{-1,1\}$ random variables X_i such that $\mathbf{E}[X_i X_j] = C v_i \cdot v_j$ for all i, j; this would immediately lead to a C-approximation algorithm. In fact, Alon and Naor's third algorithm (based on Krivine's proof) is exactly of this form: they transform the vectors u_i, v_j to obtain new vectors u'_i, v'_j and apply random hyperplane rounding to these new vectors to get a distribution on $\{-1,1\}$ random variables with the desired property. We reiterate that, in doing this, they crucially use the fact that they can apply one transformation for the $\{u_i\}$ and a another one for the $\{v_j\}$; hence this technique does not apply to rounding MAXQP.

One might wonder whether the existence of an appropriate distribution on $\{-1,1\}$ random variables is a lucky coincidence. In fact we demonstrate (the possibly surprising fact) that such a distribution always exists.

In order to do this, we write an LP formulation for the maximum gap of the SDP for a *fixed* vector solution. Given a set of vectors v_i produced by an optimal SDP solution, consider the problem of finding the A matrix that maximizes the gap between $\mathsf{OPT_{SDP}}$ and $\mathsf{OPT_{QP}}$. The problem can be formulated as an LP as follows (note that the a_{ij} are variables here).

$$\begin{array}{lll} \min & c & \\ \text{s.t.} & \sum_{ij} a_{ij}\, v_i \cdot v_j = 1 & \\ & \sum_{ij} a_{ij}\, x_i x_j \le c & \text{for all } x \in \{-1,1\}^n \end{array}$$

The accompanying dual program, below, has one p_x variable for every possible setting of $x \in \{-1,1\}^n$.

$$\begin{array}{lll} \max & b & \\ \text{s.t.} & \sum_{x} p_x = 1 & \\ & \sum_{x} p_x\, x_i x_j = b\, v_i \cdot v_j & \text{for all } i, j \end{array}$$

These p_x values specify a probability distribution on the x_is. We know that the primal LP has optimal value no lower than the gap of the SDP for MAXQP. By duality, there exists a distribution on $\{-1, 1\}$ random variables such that $\mathbf{E}[x_i x_j] = b\, v_i \cdot v_j$, where b, which could be a function of n, is at least the worst case gap of the SDP for MAXQP.

Our analysis in Section 2 shows that the gap of the MAXQP SDP is in $\Omega(1/\log n)$, which has the following interesting interpretation. Given a positive semidefinite matrix K, it is well known that there exist correlated normal random variables X_i such that $\mathbf{E}[X_i X_j] = k_{ij}$. What if you wanted $\{-1, 1\}$ random variables instead? Our results show that if we scale the (off-diagonal) entries of the correlation matrix by $C \in O(1/\log n)$, then we can guarantee the existence of $\{-1, 1\}$ random variables with the appropriate correlations.

4. Maximizing correlation in correlation clustering

We now turn to the correlation clustering application. A series of papers [3, 4, 7, 8, 6, 18] resolved many of the issues related to the MAXAGREE and MINDISAGREE problems. Recall that in the MAXCORR problem the aim is to maximize the difference between the number of agreements and disagreements. Until now, no nontrivial approximation algorithm was known for MAXCORR, but we demonstrate that the $\Omega(1/\log n)$ approximation for MAXQP can be used to obtain an algorithm with the same asymptotic factor for MAXCORR.

Let $\mathsf{corr}(\kappa)$ stand for the correlation of clustering κ. There is only one way of placing each item into a singleton cluster: call this clustering κ_n and let $\mathsf{OPT}_n = \mathsf{corr}(\kappa_n)$. In contrast, there are several ways of splitting the items into two clusters, but we let κ_2^{OPT} stand for one of those with maximal correlation OPT_2. Finally, κ^{OPT} is some partitioning that has maximum correlation OPT.

Lemma 7

$$\mathsf{OPT}_n + 2\mathsf{OPT}_2 \geq \mathsf{OPT}\ .$$

Proof: Let the four quantities, w_+, a_+, w_- and a_-, stand for the numbers of positive (negative) pairs within (across) clusters in our *optimal* solution κ^{OPT}. By definition,

$$\mathsf{OPT} = w_+ - a_+ - w_- + a_-\ .$$

If we split everything up into singletons, we see that

$$\mathsf{OPT}_n = -w_+ - a_+ + w_- + a_-\ .$$

Although we cannot calculate OPT_2, we can at least provide a lower bound for it. Consider constructing a partitioning by randomly assigning each *cluster* in κ^{OPT} to one of two new superclusters. The expected correlation of the result of this random procedure is a lower bound for OPT_2. All within-cluster pairs remain within-cluster pairs. With probability $1/2$ each across-cluster pair becomes a within-cluster pair; consequently its contribution to the expected correlation is zero, and so $\mathsf{OPT}_2 \geq w_+ - w_-$. We now find that

$$\mathsf{OPT}_n + 2\mathsf{OPT}_2 \geq w_+ - a_+ - w_- + a_- = \mathsf{OPT}\ .$$

□

As mentioned in the introduction, MAXCORR restricted to two clusters is a special case of MAXQP. So Lemma 7 shows that a reasonable approximation to OPT_2 will provide a reasonable approximation algorithm for MAXCORR. This suggests the following algorithm:

ApproxMaxCorr

1. Construct the matrix A thus:
   if $i < j$ and pair ij is similar then $a_{ij} = 1$,
   if $i < j$ and pair ij is dissimilar then $a_{ij} = -1$;
   otherwise $a_{ij} = 0$.
2. Execute *ApproxMaxQP* on A and obtain solution x.
3. Form partitioning κ_2 by assigning item i to cluster one if $x_i = -1$ and to cluster two if $x_i = 1$.
4. Calculate OPT_n and $\mathsf{corr}(\kappa_2)$ and return the clustering with higher correlation.

Lemma 8 ApproxMaxCorr *achieves an approximation of* $\alpha/(2+\alpha)$, *where* α *is the approximation factor of* ApproxMaxQP.

Proof: It is easy to verify that maximizing $x^T A x$ subject to $|x_i| = 1$ is equivalent to MAXCORR restricted to just two clusters, both in terms of feasible solutions and objective values. Clearly,

$$\begin{aligned} \max\{\mathsf{OPT}_n, \mathsf{corr}(\kappa_2)\} &\geq t\,\mathsf{OPT}_n + (1-t)\,\mathsf{corr}(\kappa_2) \\ &\geq t\,\mathsf{OPT}_n + (1-t)\,\alpha\,\mathsf{OPT}_2\ , \end{aligned}$$

for all $t \in [0, 1]$. If we let $t = \alpha/(2+\alpha)$, then

$$\begin{aligned} \max\{\mathsf{OPT}_n, \mathsf{corr}(\kappa_2)\} &\geq \frac{\alpha}{2+\alpha}\mathsf{OPT}_n + \frac{2\alpha}{2+\alpha}\mathsf{OPT}_2 \\ &\geq \frac{\alpha}{2+\alpha}\mathsf{OPT}\ , \end{aligned}$$

by Lemma 7. □

Theorem 1 tells us that the approximation factor of *ApproxMaxQP* is in $\Omega(1/\log n)$, so from Lemma 8 we conclude:

Theorem 2 ApproxMaxCorr *is an approximation algorithm for* MAXCORR *with factor in* $\Omega(1/\log n)$.

Like the MAXQP problem, this approximation factor for MAXCORR is a long way from the best-known hardness of approximation result.

Lemma 9 *It is NP-hard to approximate* MAXCORR *within a factor of* $43/44 + \varepsilon$.

Proof: An immediate consequence of our proof of the hardness of approximating MAXAGREE (Theorem 9 in [6]). □

5. Open problems

Although Bellare and Rogaway [5] prove various strong inapproximability results for quadratic and polynomial programming, they do not apply to our formulation of MAXQP. The most interesting open problem suggested by our results is whether a constant factor approximation for MAXQP is possible. Certainly, it will not involve the SDP (4), as Alon *et al.* [1] have recently shown that the integrality gap is in $O(1/\log n)$. In proving this fact, they work with a variant of the dual characterization discussed in Section 3.2.

It would be interesting to prove hardness results for the problem of maximizing the gain of MAXCUT (the advantage over a random assignment); the hardness results of Håstad and Venkatesh [13] do not apply to this problem. In light of our $\Omega(\log(1/\delta))$ approximation, obtaining an $o(1)$ hardness of approximation result would be quite challenging.

Finally, Grothendieck's inequality is a fundamental tool in functional analysis, with several applications. It would be interesting to investigate whether our extension has any functional analysis interpretations and consequences.

Acknowledgements

The authors would like to thank Noga Alon and Assaf Naor for the reference to Megretski's work and other feedback, Venkatesan Guruswami for an improvement to the hardness of approximation factor for MAXQP, and Amit Chakrabarti, Michael Dinitz, Subhash Khot, Konstantin Makarychev, and Yury Makarychev for several stimulating discussions.

References

[1] N. Alon, K. Makarychev, Y. Makarychev, and A. Naor. Quadratic forms on graphs. Preprint.

[2] N. Alon and A. Naor. Approximating the cut-norm via Grothendieck's inequality. In *Proc. of 36th STOC*, pages 72–80, 2004.

[3] N. Bansal, A. Blum, and S. Chawla. Correlation clustering. In *Proceedings of 43rd FOCS*, pages 238–47, 2002.

[4] N. Bansal, A. Blum, and S. Chawla. Correlation clustering. *Machine Learning*, 56:89–113, 2004.

[5] M. Bellare and P. Rogaway. The complexity of approximating a nonlinear program. *Journal of Mathematical Programming B*, 69:429–41, 1995.

[6] M. Charikar, V. Guruswami, and A. Wirth. Clustering with qualitative information. In *Proc. of 44th FOCS*, pages 524–33, 2003.

[7] E. Demaine and N. Immorlica. Correlation clustering with partial information. In *Proc. of 6th APPROX*, pages 1–13, 2003.

[8] D. Emanuel and A. Fiat. Correlation clustering—minimizing disagreements on arbitrary weighted graphs. In *Proc. of 11th ESA*, pages 208–20, 2003.

[9] U. Feige and M. Langberg. The RPR^2 rounding technique for semidefinite programs. In *Proc. of 28th ICALP*, pages 213–24, 2001.

[10] M. Goemans and D. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite progamming. *JACM*, 42:1115–45, 1995.

[11] A. Grothendieck. Résumé de la théorie métrique des produits tensoriels topologiques. *Bol. Soc. Mat. Sao Paulo*, 8:1–79, 1953.

[12] J. Håstad. Some optimal inapproximability results. *JACM*, 48:798–859, 2001.

[13] J. Håstad and S. Venkatesh. On the advantage over a random assignment. In *Proc. of 34th STOC*, pages 43–52, 2002.

[14] J. Krivine. Sur la constante de Grothendieck. *C. R. Acad. Sci. Paris Ser. A-B*, 284:445–6, 1977.

[15] A. Megretski. Relaxation of quadratic programs in operator theory and system analysis. In *Systems, Approximation, Singular Integral Operators, and Related Topics (Bordeaux, 2000)*, pages 365–92. Birkhäuser, Basel, 2001.

[16] Y. Nesterov. Semidefinite relaxation and nonconvex quadratic optimization. *Optimization Methods and Software*, 9:141–60, 1998.

[17] E. Rietz. A proof of the Grothendieck inequality. *Israel J. Math.*, 19:271–6, 1974.

[18] C. Swamy. Correlation Clustering: Maximizing agreements via semidefinite programming. In *Proc. of 15th SODA*, pages 519–20, 2004.

[19] L. Trevisan, G. Sorkin, M. Sudan, and D. Williamson. Gadgets, approximation, and linear programming. *SIAM J. Comp.*, 29:2074–97, 2000.

[20] U. Zwick. Outward rotations: A tool for rounding solutions of semidefinite programming relaxations, with applications to max cut and other problems. In *Proc. of 31st STOC*, pages 679–87, 1999.