5 Quadratically Constrained Quadratic Optimization

A class of optimization problems that has frequently arisen in applications is that of quadratically constrained quadratic optimization problems (QCQPs); i.e., problems of the form

$$\begin{array}{ll} \text{minimize} & x^T Q x \\ \text{subject to} & x^T A_i x \ge b_i \quad \text{for } i = 1, \dots, m, \end{array} \tag{10}$$

where $Q, A_1, \dots, A_m \in \mathcal{S}^n$ are given. In general, due to the non–convexity of the objective function and constraints, problem (10) is intractable. Nevertheless, it can be tackled by the so–called **semidefinite relaxation** technique. To introduce this technique, we first observe that for any $Q \in \mathcal{S}^n$,

$$x^T Q x = \text{tr}(x^T Q x) = \text{tr}(Q x x^T).$$

Hence, problem (10) is equivalent to

$$\begin{array}{ll} \text{minimize} & \text{tr}(Q x x^T) \\ \text{subject to} & \text{tr}(A_i x x^T) \ge b_i \quad \text{for } i = 1, \dots, m. \end{array}$$

Now, using the spectral theorem for symmetric matrices (see Section 3.1 of Handout B), one can verify that

$$X = x x^T \quad \Longleftrightarrow \quad X \succeq \mathbf{0},\ \text{rank}(X) \le 1.$$

It follows that problem (10) is equivalent to the following rank–cosntrained SDP problem:

$$\begin{array}{lll} \text{minimize} & \text{tr}(QX) & \\ \text{subject to} & \text{tr}(A_i X) \ge b_i & \text{for } i = 1, \dots, m, \\ & X \succeq \mathbf{0},\ \text{rank}(X) \le 1. & \end{array} \tag{11}$$

The advantage of the formulation in (11) over that in (10) is that it reveals where the difficulty of the problem lies; namely, in the non–convex constraint $\text{rank}(X) \le 1$. By dropping this constraint, we obtain the following semidefinite relaxation of problem (10):

$$\begin{array}{ll} \text{minimize} & \text{tr}(QX) \\ \text{subject to} & \text{tr}(A_i X) \ge b_i \quad \text{for } i = 1, \dots, m, \\ & X \succeq \mathbf{0}. \end{array} \tag{12}$$

Note that problem (12) is an SDP, and hence can be efficiently solved. Of course, an optimal solution X^* to problem (12) may not even be feasible for problem (10), since we need not have $\text{rank}(X^*) \le 1$. Moreover, it is generally impossible to convert X^* into an optimal solution x^* to problem (10). However, as it turns out, it is often possible to extract a near–optimal solution to problem (10) from an optimal solution to problem (12). In the next section, we will illustrate how this can be done for a well–known combinatorial problem—the Maximum Cut Problem (Max–Cut). For a survey of the theory and the many applications of the semidefinite relaxation technique, see [18].

5.1 An Approximation Algorithm for Maximum Cut in Graphs

Suppose that we are given a simple undirected graph $G = (V, E)$ and a function $w : E \to \mathbb{R}_+$ that assigns to each edge $e \in E$ a non–negative weight w_e. The Maximum Cut Problem (MAX–CUT) is that of finding a set $S \subset V$ of vertices such that the total weight of the edges in the cut $(S, V\backslash S)$; i.e., sum of the weights of the edges with one endpoint in S and the other in $V\backslash S$, is maximized. By setting $w_{ij} = 0$ if $(i, j) \notin E$, we may denote the weight of a cut $(S, V\backslash S)$ by

$$w(S, V\backslash S) = \sum_{i\in S, j\in V\backslash S} w_{ij}, \tag{13}$$

and our goal is to choose a set $S \subset V$ such that the quantity in (13) is maximized. The MAX–CUT problem is one of the fundamental computational problems on graphs and has been extensively studied by many researchers. It has been shown that the MAX–CUT problem is unlikely to have a polynomial–time algorithm (see, e.g, [12]). On the other hand, in a seminal work, Goemans and Williamson [13] showed how SDP can be used to design a 0.878–approximation algorithm for the MAX–CUT problem; i.e., given an instance (G, w) of the MAX–CUT problem, the algorithm will find a cut $(S, V\backslash S)$ whose value $w(S, V\backslash S)$ is at least 0.878 times the optimal value. In this section, we will describe the algorithm of Goemans and Williamson and prove its approximation guarantee.

To begin, let (G, w) be a given instance of the MAX–CUT problem, with $n = |V|$. Then, we can formulate the MAX–CUT problem as an integer quadratic program, viz.

$$\begin{array}{rclll} v^* & = & \text{maximize} & \displaystyle\frac{1}{2}\sum_{(i,j)\in E} w_{ij}(1 - x_i x_j) & \\ & & \text{subject to} & x_i^2 = 1 & \text{for } i = 1, \ldots, n. \end{array} \tag{14}$$

Here, the variable x_i indicates which side of the cut vertex i belongs to. Specifically, the cut $(S, V\backslash S)$ is given by $S = \{i \in \{1, \ldots, n\} : x_i = 1\}$. Note that if vertices i and j belong to the same side of a cut, then $x_i = x_j$, and hence its contribution to the objective function in (14) is zero. Otherwise, we have $x_i \neq x_j$, and its contribution to the objective function is $w_{ij}(1-(-1))/2 = w_{ij}$.

In general, problem (14) is hard to solve. Thus, we consider relaxations of (14). One approach is to observe that both the objective function and the constraints in (14) are *linear* in $x_i x_j$, where $1 \le i, j \le n$. In particular, if we let $X = xx^T \in \mathbb{R}^{n\times n}$, then problem (14) can be written as

$$\begin{array}{rcll} v^* & = & \text{maximize} & \displaystyle\frac{1}{2}\sum_{(i,j)\in E} w_{ij}(1 - X_{ij}) \\ & & \text{subject to} & \operatorname{diag}(X) = e, \\ & & & X = xx^T. \end{array} \tag{15}$$

Since $X = xx^T$ iff $X \succeq \mathbf{0}$ and $\operatorname{rank}(X) \le 1$, from our earlier discussion, we can drop the non–convex rank constraint and arrive at the following relaxation of (15):

$$\begin{array}{rcll} v^*_{sdp} & = & \text{maximize} & \displaystyle\frac{1}{2}\sum_{(i,j)\in E} w_{ij}(1 - X_{ij}) \\ & & \text{subject to} & \operatorname{diag}(X) = e, \\ & & & X \succeq \mathbf{0}. \end{array} \tag{16}$$

Note that (16) is an SDP. Moreover, since (16) is a relaxation of (15), we have $v^*_{sdp} \geq v^*$.

Now, suppose that we have an optimal solution X^* to (16). In general, the matrix X^* need not be in the form xx^T, and hence it does not immediately yield a feasible solution to (15). However, we can extract from X^* a solution $x' \in \{-1,1\}^n$ to (15) via the following randomized rounding procedure:

1. Compute the *Cholesky factorization* $X^* = U^TU$ of X^*, where $U \in \mathbb{R}^{n\times n}$. Let $u_i \in \mathbb{R}^n$ be the i–th column of U. Note that $\|u_i\|_2^2 = 1$ for $i = 1,\ldots,n$.
2. Let $r \in \mathbb{R}^n$ be a vector uniformly distributed on the unit sphere $S^{n-1} = \{x \in \mathbb{R}^n : \|x\|_2 = 1\}$.
3. Set $x'_i = \mathrm{sgn}\left(u_i^T r\right)$ for $i = 1,\ldots,n$, where

$$\mathrm{sgn}(z) = \begin{cases} 1 & \text{if } z \geq 0, \\ -1 & \text{otherwise.} \end{cases}$$

In other words, we choose a random hyperplane through the origin (with r as its normal) and partition the vertices according to whether their corresponding vectors lie "above" or "below" the hyperplane.

Since the solution $x' \in \{-1,1\}^n$ is produced via a randomized procedure, we are interested in its expected objective value; i.e.,

$$\frac{1}{2}\mathbb{E}\left[\sum_{(i,j)\in E} w_{ij}\left(1 - x'_ix'_j\right)\right] = \frac{1}{2}\sum_{(i,j)\in E} w_{ij}\mathbb{E}\left[1 - x'_ix'_j\right] = \sum_{(i,j)\in E} w_{ij}\Pr\left[\mathrm{sgn}\left(u_i^Tr\right) \neq \mathrm{sgn}\left(u_j^Tr\right)\right]. \tag{17}$$

The following theorem provides a lower bound on the expected objective value (17) and allows us to compare it with the optimal value v^*_{sdp} of (15).

Theorem 1 *Let $u, v \in S^{n-1}$, and let r be a vector uniformly distributed on S^{n-1}. Then, we have*

$$\Pr\left[sgn\left(u^Tr\right) \neq sgn\left(v^Tr\right)\right] = \frac{1}{\pi}\arccos\left(u^Tv\right). \tag{18}$$

Furthermore, for any $z \in [-1,1]$, we have

$$\frac{1}{\pi}\arccos(z) \geq \alpha \cdot \frac{1}{2}(1-z) > 0.878 \cdot \frac{1}{2}(1-z), \tag{19}$$

where

$$\alpha = \min_{0\leq\theta\leq\pi} \frac{2\theta}{\pi(1-\cos\theta)}.$$

Proof To establish (18), observe that by symmetry, we have

$$\Pr\left[\mathrm{sgn}\left(u^Tr\right) \neq \mathrm{sgn}\left(v^Tr\right)\right] = 2\Pr\left(u^Tr \geq 0,\ v^Tr < 0\right).$$

Now, by projecting r onto the plane containing u and v, we see that $u^Tr \geq 0$ and $v^Tr < 0$ iff the projection lies in the wedge formed by u and v. Since r is chosen from a spherically symmetric distribution, its projection will be a random direction on the plane containing u and v. Hence, we have

$$\Pr\left(u^Tr \geq 0,\ v^Tr < 0\right) = \frac{\arccos\left(u^Tv\right)}{2\pi},$$

as desired.

To establish the first inequality in (19), consider the change of variable $z = \cos\theta$. Since $z \in [-1, 1]$, we have $\theta \in [0, \pi]$. Thus, it follows that

$$\frac{1}{\pi}\arccos(z) = \frac{\theta}{\pi} = \frac{2\theta}{\pi(1-\cos\theta)} \cdot \frac{1}{2}(1-\cos\theta) \geq \alpha \cdot \frac{1}{2}(1-z),$$

as desired. The second inequality in (19) can be established using calculus and we leave the proof to the readers. □

Corollary 1 *Given an instance (G, w) of the* Max–Cut *problem and an optimal solution to (16), the randomized rounding procedure above will produce a cut $(S', V\backslash S')$ whose expected objective value satisfies $w(S', V\backslash S') \geq 0.878v^*$.*

Proof Let x' be the solution obtained from the randomized rounding procedure, and let S' be the corresponding cut. By (17) and Theorem 1, we have

$$\begin{aligned}
\mathbb{E}\left[w(S', V\backslash S')\right] &= \frac{1}{\pi}\sum_{(i,j)\in E} w_{ij} \cdot \arccos\left(u_i^T u_j\right) \\
&\geq 0.878 \cdot \frac{1}{2}\sum_{(i,j)\in E} w_{ij}\left(1 - u_i^T u_j\right) \\
&= 0.878 v^*_{sdp} \\
&\geq 0.878 v^*,
\end{aligned}$$

as desired. □

6 Sparse Principal Component Analysis

Principal Component Analysis (PCA) (see, e.g, [23]) is a very important tool in data analysis. It provides a way to reduce the dimension of a given data set, thus revealing the sometimes hidden underlying structure of and facilitating further analysis on the data set. To motivate the problem of finding principal components, consider the following scenario. Suppose that we are interested in some attributes $X_1, \ldots, X_n$ of a population. In order to estimate the values of these attributes, one may sample from the population. Specifically, let X_{ij} be the value of the j–th attribute of the i–th individual, where $i = 1, \ldots, m$ and $j = 1, \ldots, n$. For $j, k = 1, \ldots, n$, define

$$\bar{X}_j = \frac{1}{m}\sum_{i=1}^{m} X_{ij}, \quad \sigma_{jk} = \frac{1}{m}\sum_{i=1}^{m}(X_{ij} - \bar{X}_j)(X_{ik} - \bar{X}_k)$$

to be the sample mean of X_j and the sample covariance between X_j and X_k, respectively. Define $\Sigma = [\sigma_{jk}]_{j,k}$ to be the sample covariance matrix. The goal is then to find the **principal components** $u_1, \ldots, u_n$ such that the linear combination $\sum_{j=1}^{n} u_j X_j$ has *maximum* sample variance. In other words, we would like to solve the following problem:

$$\max_{\|u\|_2 \leq 1} u^T \Sigma u, \tag{20}$$