8 Binary quadratic optimisation

A binary quadratic optimisation is a problem of the form

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{maximise}} & x^T Q x \\ \text{subject to} & x_i \in \{-1,1\},\ i=1,\dots,n. \end{array} \tag{1}$$

The feasible set is the binary hypercube $\{-1,1\}^n$ and consists of a finite number of points, namely 2^n points. The objective function is quadratic. A well-known example of binary quadratic optimisation problem is the *maximum cut* problem that we describe now.

Maximum cut Consider an undirected weighted graph $G=(V,E)$ with vertex set V, edge set $E\subseteq\binom{V}{2}$, and weight function $w: E\to\mathbb{R}_+$. A *cut* is a partition of V into two sets (S,S^c) where $S\subseteq V$ and $S^c = V\setminus S$. The value of a cut is the total weight of edges that connect an element of S and an element of S^c:

$$\sum_{i\in S, j\in S^c} w_{ij}. \tag{2}$$

We consider that $w_{ij}=0$ if $\{i,j\}\notin E$. The maximum cut problem is the problem of finding a cut with maximum value. To formulate the maximum cut problem in the form (1) we can associate to any partition (S,S^c) a labelling of the nodes $x_i=+1$ if $i\in S$ and $x_i=-1$ if $i\in S^c$. Then the value of the cut (2) is equal to (up to factor 1/4)

$$\frac{1}{2}\sum_{i,j\in V} w_{ij}(x_i-x_j)^2. \tag{3}$$

Let L_G be the matrix that represents the quadratic form (3), i.e., so that

$$x^T L_G x = \frac{1}{2}\sum_{i,j\in V} w_{ij}(x_i-x_j)^2. \tag{4}$$

Exercise 8.1. *Show that the entries of L_G are given by*

$$(L_G)_{ii} = \begin{cases} \sum_{k\in V} w_{ik} & \text{if } i=j \\ -w_{ij} & \text{if } i\neq j. \end{cases} \tag{5}$$

Verify that L_G is diagonally dominant, i.e., that $(L_G)_{ii}\geq\sum_{j\neq i}|(L_G)_{ij}|$. The quadratic form (4) is known as the Laplacian of the graph G.

The maximum cut problem can thus be written as follows:

$$\begin{array}{ll} \underset{x\in\mathbb{R}^n}{\text{maximise}} & x^T L_G x \\ \text{subject to} & x_i \in \{-1,1\},\ i=1,\dots,n. \end{array} \tag{MC}$$

Semidefinite relaxation The maximum cut problem (MC) is hard in general. We will now formulate a *relaxation* of (MC), which as we will show later, will allow us to approximate the value of the maximum cut by solving a semidefinite program. This relaxation is given by:

$$\begin{array}{ll} \underset{X \in \mathbf{S}^n}{\text{maximise}} & \mathrm{Tr}(L_G X) \\ \text{subject to} & X \succeq 0 \\ & X_{ii} = 1 \ (i = 1, \dots, n) \end{array} \tag{SDP}$$

Let v^* be the optimal value of (MC) and p^*_{SDP} be the optimal value of (SDP). Several remarks/comments:

1. The optimisation problem (SDP) is a semidefinite program, and thus can be solved efficiently. Note that the variable in (SDP) is a symmetric matrix $X \in \mathbf{S}^n$, whereas the variable in (MC) was a vector $x \in \mathbb{R}^n$.

2. If x is feasible for (MC) then $X = xx^T$ is feasible for (SDP) and $\mathrm{Tr}(L_G X) = x^T L_G x$. This shows that $p^*_{SDP} \geq v^*$.

3. If the solution of the SDP (SDP) happens to be rank-one then we know we have solved the problem (MC) (i.e., in this case $p^*_{SDP} = v^*$). Indeed any rank-one matrix $X \succeq 0$ with $X_{ii} = 1$ must be of the form $X = xx^T$ with $x \in \{-1, 1\}^n$.

4. If we add an additional constraint "$\mathrm{rank}(X) = 1$" in (SDP), then the resulting problem has the same optimal value as (MC), i.e., v^*.

5. Rounding: In general, the solution to (SDP) will not be of rank one. The question is then: is there are a way to convert a X from (SDP) with $\mathrm{rank}(X) > 1$, to a point $x \in \{-1, 1\}^n$ on the hypercube with a value $x^T L_G x$ "close to" $\mathrm{Tr}(L_G X)$? This is called a *rounding* problem. We will discuss this question in more detail next lecture.

6. Covariance: It is useful to think of the matrix X in (SDP) as a *covariance* matrix: if x is a random vector on the hypercube $\{-1, 1\}^n$ with $\mathbb{E}[x] = 0$ then its covariance matrix $X = \mathbb{E}[xx^T]$ satisfies the conditions $X \succeq 0$ and $X_{ii} = 1$ for all $i = 1, \dots, n$. Furthermore the cost $\mathrm{Tr}(L_G X)$ is nothing but the expected cost $\mathbb{E}[x^T L_G x] = \mathrm{Tr}(L_G X)$.

7. Geometric picture: We can try to visualise the relaxation (SDP). To do so we are first going to reformulate (MC) so that the variable lives in the same space as that of (SDP). Namely we are going to introduce the variable X that plays the role of xx^T:

$$\begin{array}{ll} \underset{X \in \mathbf{S}^n}{\text{maximise}} & \mathrm{Tr}(L_G X) \\ \text{subject to} & X \in P_n \end{array} \tag{6}$$

   where

$$P_n = \mathrm{conv}\left\{ xx^T : x \in \{-1, 1\}^n \right\} \subset \mathbf{S}^n. \tag{7}$$

   Recall that conv denotes the convex hull. It is easy to verify that (6) is equivalent to (MC), since the optimal point of (6) is attained at a vertex of P_n.

   > To be sure let v_0^* be the optimal value of (6). If x is feasible for (MC) then $X = xx^T$ is feasible for (6). Thus $v_0^* \geq v^*$. On the other hand if X is feasible for (6) then we can write $X = \sum_j \lambda_j x_j x_j^T$ where $\lambda_j \geq 0$, $\sum_j \lambda_j = 1$ and $x_j \in \{-1, 1\}^n$ and so

$\text{Tr}(L_G X) = \sum_j \lambda_j x_j^T L_G x_j \leq \sum_j \lambda_j v^* = v^*$. Thus this shows $v_0^* \leq v^*$.

We can now compare the feasible set P_n of (6) and that of (SDP), since they live in the same space $\mathbf{S}^n$. Let E_n be the feasible set of (SDP):

$$E_n = \{X \in \mathbf{S}^n : X \succeq 0 \text{ and } X_{ii} = 1 \; \forall i = 1, \ldots, n\} \subset \mathbf{S}^n. \tag{8}$$

It is clear that $P_n \subseteq E_n$. Figure 1 depicts P_n and E_n in the case $n = 3$: in this case the convex sets E_3 and P_3 are three-dimensional and what the figure shows are the projections on the off-diagonal entries (the diagonal entries are equal to 1), namely

$$\{(X_{12}, X_{13}, X_{23}) : X \in P_3\} \subset \mathbb{R}^3 \quad \text{and} \quad \{(X_{12}, X_{13}, X_{23}) : X \in E_3\} \subset \mathbb{R}^3.$$

The set P_3 is a tetrahedron and the set E_3 looks like a "puffed" tetrahedron. We see that $P_3 \subset E_3$. Note that P_3 is polyhedral (i.e., it can be expressed as an intersection of a finite number of halfspaces) whereas E_3 is not.

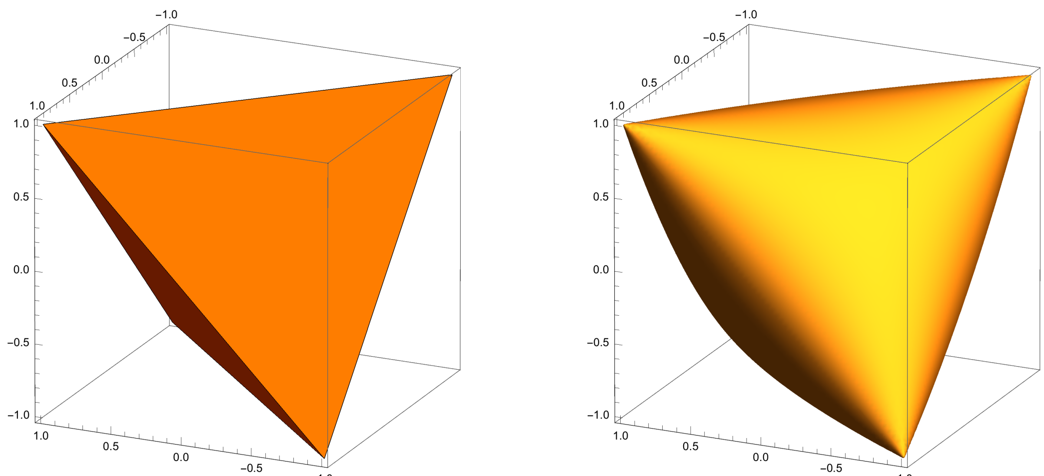

Figure 1: Comparison of P_3 (see (7)) and E_3 (see (8), feasible set of semidefinite relaxation). Left: The set P_3, projected onto off-diagonal terms. Right: The set E_3 projected onto off-diagonal terms. We see that $P_3 \subset E_3$.

Remark. *The set P_n is known as the* cut polytope. *In the case $n = 3$ we saw it is simply a tetrahedron, but for general n this polytope is much more complicated. The set E_n can be seen as an "approximation" of P_n that is more computationally tractable. The set E_n is called the* elliptope.