Expressing \textsc{MaxCut} as a Semidefinite Program

1 The MaxCut Problem

Given a graph $G = (V, E)$, a cut partitions $V$ into two sets $V_1$ and $V_2$. An edge $e = (u, v)$ is in the cut if exactly one of its endpoints is in $V_1$. We want to choose a partition so as to maximize the weight of the edges with exactly one endpoint in each set.

In the SDP approach, we formulate the problem using variables for the vertices, and construct a quadratic integer formulation. Relaxing the integrality requirement leads to a semidefinite program.

2 Example

\begin{center}
\begin{tikzpicture}[scale=0.7]
  \node (1) at (0,0) {1};
  \node (2) at (2,0) {2};
  \node (3) at (1,1) {3};
  \node (4) at (3,0) {4};
  \node (5) at (4,1) {5};
  \node (6) at (5,2) {6};
  \node (7) at (4,0) {7};
  \draw (1) -- (2) node[midway, above] {2};
  \draw (1) -- (3) node[midway, right] {5};
  \draw (1) -- (5) node[midway, left] {7};
  \draw (2) -- (3) node[midway, above] {4};
  \draw (2) -- (7) node[midway, left] {3};
  \draw (3) -- (4) node[midway, above] {2};
  \draw (3) -- (5) node[midway, right] {6};
  \draw (4) -- (5) node[midway, above] {2};
  \draw (4) -- (6) node[midway, above] {5};
  \draw (5) -- (6) node[midway, left] {4};
  \draw (5) -- (7) node[midway, left] {6};
  \draw (6) -- (7) node[midway, left] {3};
\end{tikzpicture}
\end{center}

Taking $V_1 = \{4, 5, 6\}$ and $V_2 = \{1, 2, 3, 7\}$ gives a cut with value $7 + 2 + 6 + 5 + 5 = 25$. With $V_1 = \{1, 4, 5, 7\}$ and $V_2 = \{2, 3, 6\}$, the cut has value $8 + 4 + 5 + 2 + 6 + 6 + 3 + 5 + 2 = 41$.

3 Reformulating as an SDP

Let $G = (V, E)$ be a graph with edge weights $w_e$. Let $n = |V|$. For any subset $U \subseteq V$, let $\delta(U)$ denote the edges in $E$ with exactly one endpoint in $U$ and let $E(U)$ denote the set of edges with both endpoints in $U$. Given a partition of $V$ into $V_1$ and $V_2$, the value of the corresponding cut can be expressed in two ways as

$$z(V_1, V_2) = \sum_{e \in \delta(V_1)} w_e = \sum_{e \in E} w_e - \sum_{e \in E(V_1)} w_e - \sum_{e \in E(V_2)} w_e.$$
For notational convenience, for any missing edge \((u, v) \in (V \times V) \setminus E\), we define \(w_{uv} = 0\). Combining the two formulations for MAXCut, we also have

\[
z(V_1, V_2) = 0.5 \left( \sum_{e \in E} w_e - \sum_{e \in E(V_1)} w_e - \sum_{e \in E(V_2)} w_e + \sum_{e \in \delta(V_1)} w_e \right)
\]

\[
= 0.25 \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} - 0.25 \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij}x_i x_j
\]

with \(x_i = 1\) if \(i \in V_1\) and \(x_i = -1\) if \(i \in V_2\).

This can be expressed more concisely in terms of the Laplacian matrix of the weighted graph:

\[
L_G = D_G - W_G,
\]

where the entries of \(W_G\) are the edge weights \(w_e\) and where \(D_G\) is a diagonal matrix with

\[
D_G(i, i) = \sum_{j \in V} w_{ij}.
\]

The Laplacian matrix for the example graph is

\[
L_G = \begin{bmatrix}
14 & -5 & -2 & -7 & 0 & 0 & 0 \\
-5 & 12 & -4 & 0 & 0 & 0 & -3 \\
-2 & -4 & 25 & -2 & -6 & -5 & -6 \\
-7 & 0 & -2 & 20 & -3 & -8 & 0 \\
0 & 0 & -6 & -3 & 13 & -4 & 0 \\
0 & 0 & -5 & -8 & -4 & 22 & -5 \\
0 & -3 & -6 & 0 & 0 & -5 & 14
\end{bmatrix}.
\]

We then have

\[
z(V_1, V_2) = 0.25 x^T L_G x
\]

with \(x_i = 1\) if \(i \in V_1\) and \(x_i = -1\) if \(i \in V_2\). Hence, the MAXCut problem can be written as the quadratic binary problem

\[
\begin{align*}
\max_x & \quad 0.25 x^T L_G x \\
\text{subject to} & \quad x_i = \pm 1 \quad \forall i \in V.
\end{align*}
\]
This problem can then be relaxed to a semidefinite program. First, note from properties of the trace function that

$$x^T L_G x = \text{trace}(x^T L_G x) = \text{trace}(L_G(xx^T))$$

Now, introduce an $n \times n$ matrix $X$. We can express problem (1) equivalently as

$$\max_{x, X} \quad 0.25\text{trace}(L_G X)$$
subject to
$$x_i = \pm 1 \quad \forall i \in V$$
$$X = xx^T$$

which in turn is equivalent to the problem

$$\max_{x, X} \quad 0.25\text{trace}(L_G X)$$
subject to
$$X_{ii} = 1 \quad \text{for } i = 1, \ldots, n$$
$$X \succeq 0,$$

(2)

Since $X = xx^T$, we must have that $X$ is symmetric and positive semidefinite. Furthermore, it must have rank equal to 1. Relaxing the restriction on the rank, we get the following semidefinite programming relaxation of MaxCut:

$$\max_X \quad 0.25\text{trace}(L_G X)$$
subject to
$$X_{ii} = 1 \quad \text{for } i = 1, \ldots, n$$
$$X \succeq 0,$$

(2)

where the notation $X \succeq 0$ is equivalent to the requirement that $X$ be symmetric and positive semidefinite.

For the example problem, the optimal value of the SDP relaxation is 47. The optimal solution $X$ does have rank equal to one, so factorizing $X$ gives the optimal solution to the MAXCUT instance:

$$V_1 = \{1, 3, 6\}, \quad V_2 = \{2, 4, 5, 7\}$$
The SDP relaxation doesn’t always give the optimal solution. For example, for a cycle of length 5 with every edge length equal to one, the MAXCut has value 4, but the value of the SDP relaxation is 4.5224 and the optimal matrix has rank 2. The ratio $4/4.5524 = 0.8845$. 

\begin{center}
\begin{tikzpicture}
  \node [circle, draw] (1) at (0,1.732) {$v_1$};
  \node [circle, draw] (2) at (1,0) {$v_2$};
  \node [circle, draw] (3) at (1,-1.732) {$v_3$};
  \node [circle, draw] (4) at (-1.732,0) {$v_4$};
  \node [circle, draw] (5) at (0,-1.732) {$v_5$};

  \draw[very thick,red] (1) -- (2);
  \draw[very thick,red] (2) -- (3);
  \draw[very thick,red] (3) -- (4);
  \draw[very thick,red] (4) -- (5);
  \draw[very thick,red] (5) -- (1);
\end{tikzpicture}
\end{center}