Computing The Boolean Rank Exactly

Reka A Kovacs

June 13, 2019

Let $X \in \{0,1\}^{n\times m}$ be a binary matrix considered over the Boolean semiring of two elements $\{0,1\}$, with multiplication defined as $\wedge$, addition as $\vee$ and matrix mulitplication defined as $X = A \circ B \iff x_{i,j} = \bigvee_\ell a_{i,\ell} \wedge b_{\ell,j}$. The smallest integer r for which there exist matrices $A \in \{0,1\}^{n\times r}$ and $B \in \{0,1\}^{r\times m}$ such that $X = A \circ B$ is the Boolean rank of X [1].

For a given binary matrix $X \in \{0,1\}^{n\times m}$, [2] focuses on computing a Boolean rank-k approximation. Here, we describe an integer program to compute an exact factorisation of X, i.e. computing two binary matrices $A \in \{0,1\}^{n\times r}$ and $B \in \{0,1\}^{r\times m}$, for which $X = A \circ B$ and r is minimal. The integer program described here is a close relative to the one appeared in [2].

To avoid the appearance of quadratic terms in our integer program, we use McCormick envelopes [3] for the product of $a, b \in [0,1]$ and denote it by $MC(a,b)$,

$$\begin{aligned} MC(a,b) := \{y \in \mathbb{R} : 0 &\le y, && (1)\\ y &\le a, && (2)\\ y &\le b, && (3)\\ a + b - 1 &\le y\}. && (4) \end{aligned}$$

Note that when $a, b \in \{0,1\}$, then $MC(a,b)$ only contains the point $ab \in \{0,1\}$, allowing us to express the non-linear relationship $y = ab$ in terms of linear constraints only.

The Boolean rank of a given $X \in \{0,1\}^{n\times m}$ can now be computed by solving the following integer program (IP),

$$
\begin{aligned}
\text{(IP)} \quad & \min_{a,b,y,d} \sum_{\ell=1}^{k} d_\ell & & \\
\text{s.t. } & y_{i,\ell,j} \in MC(a_{i,\ell}, b_{\ell,j}) & i \in [n], j \in [m], \ell \in [k] & \quad (5)\\
& \sum_{\ell=1}^{k} y_{i,\ell,j} \geq 1 & (i,j) \in S_1 & \quad (6)\\
& y_{i,\ell,j} = 0 & \ell \in [k], (i,j) \in S_0 & \quad (7)\\
& a_{i,\ell} \leq d_\ell & i \in [n], \ell \in [k] & \quad (8)\\
& b_{\ell,j} \leq d_\ell & \ell \in [k], j \in [m] & \quad (9)\\
& a_{i,\ell}, b_{\ell,j}, y_{i,\ell,j}, d_\ell \in \{0,1\} & i \in [n], j \in [m], \ell \in [k], & \quad (10)
\end{aligned}
$$

where $k := \min(n,m)$, $[n] := \{1, 2.., n\}$ and

$$S_1 := \{(i,j) : x_{i,j} = 1\}, \qquad S_0 := \{(i,j) : x_{i,j} = 0\} \tag{11}$$

denote the index set of 1 and 0 valued entries of X respectively.

The optimal solution of (IP), $A = [a_{i,\ell}] \in \{0,1\}^{n\times k}$, $B = [b_{\ell,j}] \in \{0,1\}^{k\times m}$ provides an exact factorisation of $X = A \circ B$, because the McCormick envelopes (5) imply $y_{i,\ell,j} = a_{i,\ell} b_{\ell,j} \in \{0,1\}$ for all i, ℓ, j, while constraints (6),(7) ensure that $x_{i,j} = \bigvee_{\ell}^{k} y_{i,\ell,j}$ for all (i,j) in S_1 and S_0. The objective value of (IP) gives the Boolean rank of X, $r = \sum_{\ell=1}^{k} d_\ell$. Constraints (8) and (9) imply that at most r columns of A are non-zero, and at most r rows of B are non-zero, therefore $A \circ B$ is of Boolean rank at most r.

In computations, removing some redundant constraints provides a speed-up but does not alter the optimal solution. In addition, introducing the partial symmetry breaking constraints $d_{\ell-1} \geq d_\ell$ helps. The following model is used in my Python code:

$$
\begin{aligned}
\text{(IP)} \quad & \min_{a,b,y,d} \sum_{\ell=1}^{k} d_\ell \\
& \text{s.t.} \ \sum_{\ell=1}^{k} y_{i,\ell,j} \geq 1 && (i,j) \in S_1 && (12)\\
& \quad y_{i,\ell,j} \leq a_{i,\ell} && \ell \in [k], (i,j) \in S_1 && (13)\\
& \quad y_{i,\ell,j} \leq b_{\ell,j} && \ell \in [k], (i,j) \in S_1 && (14)\\
& \quad y_{i,\ell,j} = 0 && \ell \in [k], (i,j) \in S_0 && (15)\\
& \quad a_{i,\ell} + b_{\ell,j} \leq 1 && \ell \in [k], (i,j) \in S_0 && (16)\\
& \quad a_{i,\ell} \leq d_\ell && \ell \in [k], i \in [n] && (17)\\
& \quad b_{\ell,j} \leq d_\ell && \ell \in [k], j \in [m] && (18)\\
& \quad d_\ell \leq d_{\ell-1} && \ell \in [2,k] && (19)\\
& \quad y_{i,\ell,j} \in [0,1] && i \in [n], j \in [m], \ell \in [k], && (20)\\
& \quad a_{i,\ell}, b_{\ell,j}, d_\ell \in \{0,1\} && i \in [n], j \in [m], \ell \in [k]. && (21)
\end{aligned}
$$

References

[1] K.H. Kim. *Boolean matrix theory and applications.* Monographs and textbooks in pure and applied mathematics. Dekker, 1982.

[2] Reka A. Kovacs, Oktay Gunluk, and Raphael A. Hauser. Low-rank boolean matrix approximation by integer programming. In *http://opt-ml.org/papers/OPT2017_paper_34.pdf*, NIPS Optimization for Machine Learning Workshop, Dec 2017.

[3] Garth P. Mccormick. Computability of global solutions to factorable nonconvex programs: Part i – convex underestimating problems. *Math. Program.*, 10(1):147–175, December 1976.