组合最优化算法 习题解答

原生生物

* 邵嗣烘老师《组合最优化算法》课程作业。

目录

1 第一次作业

1. **描述 P 与 NP 问题。**

   基本定义：

   - 决定问题：给定一些 (某种编码下) 字符串，并输出是或否的问题。
   - 输入规模：一定编码下字符串的长度。
   - 运行时间：可考虑确定型图灵机模型下需要的步数作为时间的判定标准，一般可以理解为操作数量。
   - 规约：问题 Q1 能够规约于问题 Q2，也即 Q1 可以等价于 Q2 或其某种特殊情况，用相同方法求解。

   若某个问题存在运行时间为输入规模多项式量级的求解算法，则称为 P 问题；若可以在多项式时间内决定结果是否正确，则称为 NP 问题 (由此 P 问题一定是 NP 问题)。

   若任何一个 NP 问题都可以规约于某问题，则此问题称为 NPH 问题；若 NPH 问题是一个 NP 问题，则称为 NPC 问题 (利用规约的定义，一切 NPC 问题均等价)。

   问题：P 与 NP 是否相同？只要证明某 NPC 问题存在多项式时间算法，即可说明相同；而只要证明某 NPC 问题不存在多项式时间，即可说明不同。目前普遍认为 P 与 NP 不同。

2. **找三个组合优化问题 (NP 问题) 使用遍历法，记录规模与计算时间。**

   考虑如下的三个问题 (代码文件 1.ipynb，以 Python 自带的计时模块 time 为标准)：

   (a) 背包问题：给定物品 $I_i, i=1,\dots,n$，各有重量 s_i 与价值 c_i，背包总承重 S，求可能放入背包的最高价值。

   遍历方法：遍历所有可能的放入方法，在承重限制内的情况下计算最大值。

   输出结果：

```
N = 10, S = 100
[ 9 18  9 11 19 17 16 18  6  7]
[34 52 61 72  2 38  2 24 35 74]
best cost: 390
best choice: 1111010111
time: 0.0019998550415039062s

N = 15, S = 150
[ 9 16  5  5 17  7 13 12 16 15 12 19 13 15 16]
[93 54 15 68 56 67 76  1 28 98  1 23 21 85 97]
best cost: 758
best choice: 111111101100111
time: 0.07551312446594238s

N = 20, S = 200
[16 13 16  6 17 14 13  6  6 17 12 14  9  9 10  6  5 19 15 19]
[77 80 59 80 72 47 29 61  2 13 94 57 24  7 84 43  5 70 97  8]
best cost: 981
best choice: 11111111001111110110
time: 3.061936140060425s

N = 25, S = 250
[11  8  6  7 11  8 19 16 19 10 13  9  8  9 13 11 10 15  8 10  7
```

```
12 12 17 6]
[79 70 23 22 92 92 22 70 87 81 64 55 10 92 21 25 63 19 28 92 97
21 36 58 84]
best cost: 1371
best choice: 1111110111110111111111111
time: 121.68455338478088s
```

每个输出的第一行为物品个数 N 与承重 S 的值，接下来两部分为随机生成的 s 与 c 向量，输出最优价值、最优选择 (0 表示不选对应物品，1 代表选中对应物品)，与耗费的时间，可以发现，复杂度为 $O(N2^N)$，随规模提升巨大，在 $N=25$ 时需要 2 分钟，因此 $N=30$ 时已不可能在 30 分钟内完成。

(b) 最大割问题：给定无向图 (V,E) 与源点 s、汇点 t，求顶点划分 S,T 使得 $s\in S, t\in T$，且边 $(s_0,t_0), s_0\in S, t_0\in T$ 的总数尽可能大。

遍历方法：遍历 $V-\{s\}-\{t\}$ 的所有子集 A，并将 $s\cup A$ 设定为 S，比较得到最大割。

输出结果：

```
N = 10, M = 10
max edges: 9
best choice: 1101100100
time: 0.0010099411010742188s

N = 10, M = 45
max edges: 25
best choice: 1111100000
time: 0.0019969940185546875s

N = 15, M = 50
max edges: 35
best choice: 110011010011100
time: 0.07805824279785156s

N = 15, M = 100
max edges: 56
best choice: 110111010100000
time: 0.13410186767578125s

N = 20, M = 10
max edges: 10
best choice: 10101001100100110000
time: 0.4873371124267578s

N = 20, M = 100
max edges: 65
best choice: 11011010001101100000
time: 4.537440776824951s

N = 25, M = 30
max edges: 28
```

```
best choice: 1101000000100011010100100
time: 44.50195860862732s

N = 25, M = 300
max edges: 156
best choice: 1111111111110000000000000
time: 442.4826683998108s
```

这里 N 表示顶点数，M 表示边数，边为从完全图中随机选取 M 条不同的而生成，为方便观察，并未将全部边进行输出。接下来两部分为输出最佳的边数与对应的最优选择，规则与背包问题相同，不过人为规定第一个点为源点，最后一个点为汇点，因此第一位始终为 1，最后一位始终为 0。算法的理论复杂度为 $O(2^{N-2}M)$，实际测试结果与此复杂度基本一致。同样，在 $N=30$ 时一般已不可能在 30 分钟内完成。

(c) Cheeger 问题：给定无向图 (V,E) 与源点 s、汇点 t，求顶点划分 S,T 使得 $s\in S,t\in T$，且边 $(s_0,t_0),s_0\in S,t_0\in T$ 的总数除以 S、T 中较小的顶点度数和结果尽可能小。
遍历方法：先遍历所有边计算顶点度数向量，再遍历 $V-\{s\}-\{t\}$ 的所有子集 A，并将 $s\cup A$ 设定为 S，比较满足要求的最小割集。
输出结果：

```
N = 10, M = 10
cheeger const: 0.4
best choice: 1000110100
time: 0.00099945068359375s

N = 10, M = 45
cheeger const: 0.5555555555555556
best choice: 1111100000
time: 0.0019974708557128906s

N = 15, M = 50
cheeger const: 0.4
best choice: 111100110100000
time: 0.08765435218811035s

N = 15, M = 100
cheeger const: 0.5483870967741935
best choice: 101011110000100
time: 0.15214967727661133s

N = 20, M = 10
cheeger const: 0.1111111111111111
best choice: 10001010001000001000
time: 1.157426357269287s

N = 20, M = 100
cheeger const: 0.41414141414141414
best choice: 10110100111011010000
```

```
time: 5.332449197769165s

N = 25, M = 30
cheeger const: 0.1724137931034483
best choice: 1101100011000000101101100
time: 72.89438128471375s

N = 25, M = 300
cheeger const: 0.5416666666666666
best choice: 1111111111110000000000000
time: 474.84622955322266s
```

这里 N 表示顶点数，M 表示边数，边为从完全图中随机选取 M 条不同的而生成，为方便观察，并未将全部边进行输出。接下来两部分为输出 Cheeger 常数 (即所求最小值) 与对应的最优选择，规则与背包问题相同，不过人为规定第一个点为源点，最后一个点为汇点，因此第一位始终为 1，最后一位始终为 0。算法的理论复杂度为 $O(2^{N-2}(M+N))$，实际测试结果与此复杂度基本一致。同样，在 $N=30$ 时一般已不可能在 30 分钟内完成。

3. 证明背包问题动态规划算法结果的最优性。

只需证明其正确构造了所有 $c(i,j)$，即可通过寻 n 找使 $c(n,j)$ 不为 nil 的最大 j 得到最优结果，而这又只需验证初始化和递推的正确性即可。

初始化：A 在 $\{1\}$ 中时，只有可能为空集，对应价值 0；或为 $\{1\}$，对应价值 c_1，由此 $i=1$ 时情况完全确定。

递推：若 $i=k-1$ 的情况已知，考虑 $c(k,j)$。若其非空，对应的选择里或选择了物品 k，或没有选择物品 k。对前者，$c(k-1,j-c_k)$ 不能为空，且体积必须不超过 $S-s_k$，否则将无法放入，而这时由于 $c(k-1,j-c_k)$ 已是其他物品总体积最低的选择，添加物品 k 后仍为最低；对后者，即为 $c(k-1,j)$ 的结果。由此，两者都存在时需要对 $S_{c(k-1,j-c_k)}+s_k$ 与 $S_{c(k-1,j)}$ 进行比较，按其中总体积最小的作为最终方案，这即是算法的过程：只要 $c(k-1,j-c_k)$ 为空或体积超过 $S-s_k$，无论如何不可能放入第 k 件物品，于是 $c(k,j)=c(k-1,j)$，无论其是否为空；否则若 $c(k-1,j)$ 为空，无需比较，直接放入第 k 件即可；若两者均存在，则必须比较后得到最终结果。

4. 计算松弛后的连续线性规划问题的最优解。

仍按照编号越小价容比越大，且 $S-s_{k+1}<s_1+\cdots+s_k\le S$ 假设进行计算。记 $y_i=s_ix_i$，$t_i=c_i/s_i$，则问题变为求

$$\max t(y)=\sum_i t_iy_i$$

使得

$$\sum_i y_i\le S,\quad y_i\in[0,s_i]$$

将最优解记作 y^*，若 $\sum_i y_i^*<S$，任意增大 y^* 某不为 s_i 的分量可使结果变大，矛盾。

假设 $S>s_1$，若 $y_1^*<s_1$，可知 $\sum_{i>1}y_i>S-s_1>0$，任取某非零 y_j，考虑 $\hat{y}$ 满足

$$\hat{y}_1=y_1^*+\min(y_j,s_1-y_1^*),\hat{y}_j=y_j^*-\min(y_j,s_1-y_1^*)$$

且其余分量与 y^* 相同，则 $t(\hat{y})-t(y^*)=(t_1-t_j)\min(y_j,s_1-y_1^*)\ge 0$。若大于号成立，与 y^* 最优矛盾，因此必成立等于号，可如此降低 y_k 增加 y_1。若 min 中取右侧，则此时 $\hat{y}_1=s_1$ 已经成立，若否，则 y_j 变

为 0。重复此过程，最多执行 $n-1$ 次，即可使 $\hat{y}_1$ 变为 s_1，由此，必然存在 $y_1^* = s_1$ 的最优解 y^*，这时问题即化为求

$$\max t^{(2)}(y) = \sum_{i>1} t_i y_i$$

使得

$$\sum_{i>1} y_i \leq S - s_1, \quad y_i \in [0, s_i]$$

与之前同理可知，只要 $S - s_1 > s_2$，仍可取出 $y_2^* = s_2$ 的最优解，以此重复可得到能取出最优解 y^* 使得 $y_k^* = s_k$ 对所有 k 成立，问题最终化为求

$$\max t^{(k+1)}(y) = \sum_{i>k} t_i y_i$$

使得

$$\sum_{i>k} y_i \leq S - \sum_{i=1}^{k} s_i, \quad y_i \in [0, s_i]$$

注意到，即使 $S \leq s_1$，也可直接化为上式，这时 $k = 0$，s_i 的求和项不存在。而此时直接有

$$t^{(k+1)}(y) \leq t_{k+1} \sum_{i>k} y_i \leq t_{k+1} \left(S - \sum_{i=1}^{k} s_i \right)$$

且取

$$y_{k+1} = S - \sum_{i=1}^{k} s_i, \quad y_t = 0, \quad t > k+1$$

可取到最大值，由此这就是问题的最优解，利用 $x_i = y_i / s_i$ 即得原问题的一个最优解：

$$x_j = \begin{cases} 1 & j = 1, \dots, k \\ \frac{1}{s_{k+1}} \left(S - \sum_{i=1}^{k} s_i \right) & j = k+1 \\ 0 & j > k+1 \end{cases}$$

2 第二次作业

1. **证明权衡算法的性能比结论。**

   设改变权值后对应的最优解为 C_m，原问题最优解对应的新权值和为 C_o。

   若不加向下取整，新权值对应的最优解应为 $\frac{n(1+h)}{M}\mathrm{opt}$，而由于此至多比 C_o 增加了每个物品 1，必有

   $$\frac{n(1+h)}{M} \mathrm{opt} < C_o + n \leq C_m + n$$

   另一方面，根据定义，由于 c_{GGG} 中每个权值被放大的倍数不超过 $n(1+h)/M$，有

   $$\frac{n(1+h)}{M} c_{GGG} \geq C_m$$

   于是

   $$c_{GGG} + \frac{M}{1+h} > \mathrm{opt}$$

   而利用 $M \leq \mathrm{opt}$，移项可得结论。

2. **证明背包问题存在有效算法，当且仅当其判定形式存在有效算法。**

   左推右：有解时可直接求出解 opt，从而进行 K 与 opt 的比较而判定。

   右推左：假设判定问题的复杂度为 $p(n)$。利用二分法，以 $\sum_i c_i$ 为上界、0 为下界寻找最好的 K，则至多进行 $\log(Mn)$ 次，由此可得到复杂度为 $O(p(n)\log(nM))$，而输入数据所需存储空间为 $O(n\log(M))$，由此最终仍为 $n\log M$ 的多项式量级。

3. 验证图形式的等价性结论。

对 $(0,0)$ 通往 $(n+1,S)$ 的每一条路径，根据点连接的情况，其每条边必然从 $(i-1,j)$ 连向了 (i,k)。将这条路径中所有满足 $k>j, i\in[1,n]$ 的 i 集合记为 I，可发现由于 k 始终不超过 S，根据定义有 $\sum_{i\in I} s_i \le S$，而这条路径的长度即为 $-\sum_{i\in I} c_i$。

反之，对于背包问题的每一种选法 I，也可如此构造路径，当且仅当 $i\in I$ 时连接 $(i-1,j)$ 与 $(i,j+s_i)$，否则连接 $(i-1,j)$ 与 (i,j)，并最终连接 $(\sum_i s_i, n)$ 与 $(S,n+1)$，可发现路径长度为 $-\sum_{i\in I} c_i$。

由此，背包问题的每一种选择与图上的每一条 $(0,0)$ 到 $(n+1,S)$ 的路径意义对应，且背包问题的解与路径长度为相反数，由此即得最大价值与最短路径为相反数。

4. 证明基础版本最短路问题的最优值等于不相交 **s-t** 割的最大个数。

考虑 s 到 t 的最短路径，设其长度为 m，则其上第 i 个点 (设 s 为第 0 个、t 为第 m 个)v_i 应属于 V_i，否则可找到更短的路径推出矛盾。由此构造 $U_i=\bigcup_{k=0}^{i} V_i, i=0,\dots,m-1$，根据定义可知 $\delta^{out}(U_i)$ 为 V_i 到 V_{i+1} 的所有可能路径，于是它们对应不相交的 s-t 割。这证明了不相交 s-t 割的个数大于等于最短路的长度。

另一面，考虑一系列不相交的割，对每个 U_i，考虑 i 为使得 v_0 到 v_i 都属于 U_i，而 v_{i+1} 不属于 U_i 的最小 i (由 v_0 属于，v_m 不属于，i 一定存在)，则 (v_i,v_{i+1}) 在第 i 个割中。又由这些割不交，每个割至少包含一条 (v_i,v_{i+1})，即知最多取 m 个，从而得证。

5. 利用合适的数据结构构造 $O((|E|+|V|)\log|V|)$ 复杂度的正权最短路径算法。

除了数组 $f(u)$ 外，额外构造布尔数组 $b(u)$ 表示到 u 的最短路径是否已确定，然后利用最小优先队列储存 $(u,f(u))$ 对，以 $f(u)$ 最小作为排序标准，具体操作为：

(1) 初始优先队列中只有 $(s,0)$，$f(u)$ 只有 s 为 0、其余为 ∞，$b(u)$ 全部为 0；

(2) 从优先队列中取出队顶元素 u 并弹出队列；

(3) 若其最短路径已经确定，回到 (2)，否则进入下一步；

(4) 对每条边 $a=(u,v)\in A$，若 $f(v)>f(u)+l(a)$，则 $f(v)=f(u)+l(a)$，并将 $(v,f(v))$ 压入队列。

(5) 置 $b(u)=1$，若队列未空则回到 (2)，否则结束算法。

注意到，虽然一个顶点可能会多次压入队列，但每次新压入一定比之前压入的长度更短，因此队顶一定为实际上当前 $f(u)$ 里最小的，算法正确性得以保证。

压入队列的操作至多对每条边进行两次，而压入队列的复杂度为队长度的 log 量级，由于队中顶点个数不可能超过 $2|E|$，这部分的时间复杂度为 $|E|\log|V|$。

取出队顶元素的复杂度为 1，而根据之前的讨论，队中顶点个数不会超过 $2|E|$，于是 (2)(3) 的循环也不会超过 $2|E|$ 次，考虑到每个结点至少被遍历一次，总复杂度即成为 $O((|V|+|E|)\log|V|)$。

6. 构造无负环最短路的 **Bellman-Ford** 算法并证明其正确性。

其基本算法为：

(1) 初始化 $f(u)$ 与 Dijkstra 算法相同。

(2) 对每条边 $a=(u,v)\in A$ 循环 $|V|-1$ 次：若 $f(v)>f(u)+l(a)$，则 $f(v)=f(u)+l(a)$。

(3) 输出结果。

我们归纳证明，若 s 到 v 的一条最短路径途径边数为 m，则在第 m 次循环后 v 的最短路径长度必然被确定：

- 当 $m=1$ 时，由于第一次循环即将所有与 s 相连的结点置为了 s 到其的路径长度，最小长度已经获得。

- 若 $m=k-1$ 时成立，考虑某点 v 与 s 到其的实际最短路径，途径结点为

$$s=v_0,v_1,v_2,\ldots,v_{k-1},v=v_k$$

由于第 $k-1$ 次循环后，s 到 v_{k-1} 的最小长度已经确定，第 k 次循环进行后有 $f(v_k)\le f(v_{k-1})+l(v_{k-1},v_k)$，根据假设此即为最小长度，从而得证。

由于途径边数至多为 $|V|-1$，进行 $|V|-1$ 次循环后必然能确定所有最短路径。

3 第三次作业

1. **将最长有向哈密顿路问题转化为最大独立子集问题，并对应定义贪婪算法，计算性能比。**

定义有向完全图的边集为 E，其子集族 $\mathcal{I}$ 满足 $I\in\mathcal{I}$ 当且仅当 I 为一些不相交的路，则由定义其构成独立系统。

由于原本的算法仍然适用，计算性能比只需估算 ρ，下面证明 $\rho\le 3$，也即要证对 E 子集 F 的任何极大独立集 I,J 有 $|J|\le 3|I|$。

先考虑 I,J 中无公共边的情况。

与之前类似，已知 I 为极大独立集后，F 中任何一条不在 I 中的边或与 I 中某条边共起点、或与 I 中某条边共终点、或连接 I 中某条路的起点与另一条路的终点。

将 J 中满足三种性质的边集合分别记为 J_1、J_2 与 J_3，由于 J 为极大独立集，I 中任何边至多与 J 中一条边共起点/终点 (否则 J 中会有共起点/终点的边)，且 I 中的某条路至多被 J 中一条边将其起点与另一条路终点连接 (否则 J 中会有共起点的边)。进一步利用 I 中路的条数不超过边数即可得到

$$|J|=|J_1|+|J_2|+|J_3|\le 3|I|$$

若 I,J 有公共边，设有 n 条，将这 n 条边从 F,I,J 中去掉后，可发现剩余的 I,J 均构成剩余的 F 的极大独立集，且无公共边，因此

$$|J|-n\le 3(|I|-n)$$

从而

$$|J|\le 3|I|-2n\le 3|I|$$

2. **设 $(E,\mathcal{I})$ 是一个独立系统，且假设 E 的所有极大独立集都含有 k 个元素。考虑非负权函数 c，仍类似前文定义 ρ，并考虑权和最小的极大独立子集问题，设真实最优解 I'，对应的贪婪算法选出的集合为 I_G，证明**

$$c(I')\le c(I_G)\le\frac{1}{\rho}c(I')+\frac{\rho-1}{\rho}kM,\quad M=\max_e c(e)$$

根据 I' 为最优解可知 $c(I')\le c(I_G)$。另一方面，考虑将所有边按权值从小到大排列，利用上课证明过程仍然有

$$c(I_G)=\sum_{i=1}^{n-1}|E_i\cap I_G|(c(e_i)-c(e_{i+1}))+|E_n\cap I^*|c(e_n)=\sum_{i=1}^{n-1}|E_i\cap I_G|(c(e_i)-c(e_{i+1}))+kM$$

$$c(I')=\sum_{i=1}^{n-1}|E_i\cap I'|(c(e_i)-c(e_{i+1}))+|E_n\cap I'|c(e_n)=\sum_{i=1}^{n-1}|E_i\cap I'|(c(e_i)-c(e_{i+1}))+kM$$

利用此时贪婪算法的构造过程，可知 $I_G\cap E_i$ 仍为 E_i 的极大独立子集，从而可知 (注意 $c(e_i)-c(e_{i+1})$ 非正)

$$c(I_G)\le\sum_{i=1}^{n-1}u(E_i)(c(e_i)-c(e_{i+1}))+kM$$

$$c(I') \geq \sum_{i=1}^{n-1} v(E_i)(c(e_i) - c(e_{i+1})) + kM$$

从而利用 $v(E_i) \leq \rho u(E_i)$ 可知

$$c(I') \geq \sum_{i=1}^{n-1} \rho u(E_i)(c(e_i) - c(e_{i+1})) + kM$$

于是

$$\rho c(I_G) - c(I') \geq (\rho - 1)kM$$

变形即得结论。

3. **验证图拟阵构成拟阵。**

先考虑无向图的情况。对于 E 的任何子集 F，考虑其连通分支 $F_1, \dots, F_k$，且顶点个数分别为 $V_1, \dots, V_k$。由于在某连通分支中取边不会影响其他连通分支中是否有环，可分每个连通分支讨论，而单个连通分支中即为最小生成树，边数为 $|V_i| - 1$，由此总边数固定为 $\sum_i |V_i| - k$。

对于有向图，设 $V_1, \dots, V_k$ 为 F 的强连通分支，每个连通分支内的讨论如前，而取不同连通分支间的有向边一定不会成环 (否则两侧的点应在同一强连通分支)，由此总边数固定为 $\sum_i |V_i| - k$ 加上所有不同 V_i 间的边。

4. **验证证明中构造的 $\mathcal{G}_i$ 为独立系统，且交为 $\mathcal{G}$。**

若 $c_i \not\subset F$，则 c_i 不可能包含于 F 的任何子集，因此 $\mathcal{G}_i$ 为独立系统。

由于 c_i 均为 $\mathcal{G}$ 的极小相关集，$\mathcal{G}$ 中任何独立子集不可能包含任何 c_i，从而

$$\mathcal{G} \subset \bigcap_{i=1}^{n} \mathcal{G}_i$$

另一方面，若 $\bigcap_{i=1}^{n} \mathcal{G}_i$ 中有一个元素 F，根据定义可知 $c_i \not\subset F$ 对任何 i 成立。若 $F \notin \mathcal{G}$，考虑从 F 中逐步去掉元素得到的极小相关集，其与任何 c_i 不同，这与 c_i 取遍 $\mathcal{G}$ 的极小相关集矛盾。

5. **证明 $r(A)$ 是次模函数。**

由于 E 为拟阵，$r(A)$ 即为其中任何极大独立子集的大小。

设 $A \cap B$ 的某极大独立子集为 E，不断添加使得其成为 A 的某极大独立子集 $E \cup E_A$，$E_A \cap E = \varnothing$，再不断添加使得其成为 $A \cup B$ 的极大独立子集 $E \cup E_A \cup E_B$，这里 $E_A \cap E_B = E \cap E_B = \varnothing$。

首先，若 $E_B \cap A \neq \varnothing$，则其中至少存在一个 $x \in A$，于是 $E \cup E_A \cup \{x\}$ 为 A 的独立子集，与极大性矛盾，由此可知 $E_B \cap A = \varnothing$，于是 $(E \cup E_A \cup E_B) \cap B = E \cup E_B$ 为 B 的独立子集，即有

$$|E| + |E_B| \leq r(B)$$

而再结合

$$|E| = r(A \cap B), \quad |E| + |E_A| = r(A), \quad |E| + |E_A| + |E_B| = r(A \cup B)$$

即可得到

$$r(A \cap B) + r(A \cup B) \leq r(A) + r(B)$$

得证。

4 第四次作业

1. **给出能化为最小次模覆盖问题的实例，并结合实例解释下方性能比结论的含义。**

   考虑最小击中集问题：给定 V 与其子集族 $\mathcal{C}$，将 V 中元素视为顶点，$\mathcal{C}$ 中子集称为超边 (也即可能连接多个顶点的边) 可定义超图，点的度数为包含其的边的数量。超图的击中集 $A \subset V$ 定义为使得每条边至少包含一个 A 中顶点的集合 A，给定每个顶点上的非负权值 $c(a)$，求超图中使得总度数最小的击中集。

   问题转化：对 V 的子集 A，定义

   $$f(A) = |\mathcal{E}(A)|, \quad \mathcal{E}(A) = \{C \in \mathcal{C} \mid C \cap A \neq \varnothing\}$$

   利用定义可发现

   $$\mathcal{E}(A \cup B) = \mathcal{E}(A) \cup \mathcal{E}(B), \quad \mathcal{E}(A \cap B) \subset \mathcal{E}(A) \cap \mathcal{E}(B)$$

   于是

   $$f(A) + f(B) = |\mathcal{E}(A) \cup \mathcal{E}(B)| + |\mathcal{E}(A) \cap \mathcal{E}(B)| \geq f(A \cup B) + f(A \cap B)$$

   而直接利用定义可验证其单调增与正规，从而只需证明 Ω_f 为全部击中集。

   对任何击中集 A，根据定义可知 $f(A) = |\mathcal{C}|$ 达到最大值，从而不会再增加，而只要 $f(A) < |\mathcal{C}|$，$\mathcal{C}$ 中还有 C 与 A 无交，增添 C 中任何元素可使其更大，从而得证。

   性能比含义：此时 γ 即代表 A 中点的最大度数，也即代表性能比可被最大度数的对数控制。

2. **证明 f 为单调增次模函数时**

   $$\Omega_f = \{A \subset E \mid f(A) = f(E)\}$$

   若 $f(A) = f(E)$，利用单调增性可知对任何 B 有 $f(A) \leq f(A \cup B) \leq f(E)$，由此 $f(A \cup B) = f(A)$，从而 $f(A) \in \Omega_f$。

   若 $A \in \Omega_f$，设 $E \backslash A = \{x_1, \ldots, x_k\}$，则

   $$f(E) - f(A) = \sum_{i=1}^{k} \Delta_{x_i} f(A \cup \{x_1, \ldots, x_{i-1}\}) \leq \sum_{i=1}^{k} \Delta_{x_i} f(A) = 0$$

   再利用单调性可知 $f(E) \geq f(A)$，从而得证相等。

3. **证明半定规划的原问题与对偶问题都有内点可行解时，原问题与对偶问题必存最优解，且满足互补条件。**

   (a) **问题定义**

   设原问题为，给定 n 阶实对称方阵 $C, A_1, \ldots, A_m$，与实数 $b_1, \ldots, b_m$，求对 $i = 1, \ldots, m$ 有 $X \cdot A_i = b_i$ 且 X 半正定时 $C \cdot X$ 的最小值，这里 $\cdot$ 表示逐个元素对应乘积并求和。

   其对偶问题为，对满足

   $$\sum_{i=1}^{m} y_i A_i + Z = C$$

   的 y 与半正定阵 Z，求 $b^T y$ 的最大值。

   严格可行解定义为条件中的半正定改为正定时满足条件的 X 或 y 与 Z，互补条件指原问题与对偶问题的最优值相同。

   (b) **弱对偶性**：对任何可行解 X, y, Z 有 $C \cdot X \geq b^T y$。

   利用条件可知

   $$b^T y = \sum_i y_i (X \cdot A_i) = X \cdot \sum_i y_i A_i = X \cdot C - X \cdot Z$$

   由此左减右即为 $X \cdot Z = \mathrm{tr}(XZ)$。设 X 正交相似对角化为 $P^T D P$，则 $\mathrm{tr}(XZ) = \mathrm{tr}(DPZP^T)$，而 PZP^T 半正定，利用定义取 $e_i^T (PZP^T) e_i$ 可知对角元非负，而乘非负对角阵 D 后结果仍然非负，从而得证结果非负。

(c) **强对偶性**：原问题与对偶问题最优值相同。

利用特征值连续性可知对任何正定阵 X，存在 ε 使得对任何所有元素模长不超过 1 的对称阵 A，$X+\varepsilon A$ 正定。由此，由于原问题、对偶问题可行域均为线性流形与正定条件的交，正定解一定代表其为可行域的相对内点，

利用定义，半正定阵的正线性组合仍为半正定，由此原问题与对偶问题可行域均为凸集，再由优化目标为线性函数可知其为凸优化，而原问题严格可行解存在代表符合 Slater 条件，从而强对偶性成立，且对偶问题最优值可取到。

(d) **均可取到**

由于上方已经证明，原问题存内点可行解意味着强对偶性成立、对偶问题最优值可取到，而对偶问题的对偶是原问题，因此对偶问题存内点可行解意味着原问题最优值也可取到，这就说明了两边最优值均可取到，且强对偶性成立。

4. **设计近似比 $\geq 1/2$ 的最大割贪婪算法。**

定义 $f(A)=|E(A,A^c)|$，从 $A_0=\varnothing$ 出发，每次令新的集合 A_{n+1} 为比 A_n 多或少一个元素，且 $f(A_{n+1})-f(A_n)$ 尽量大的 A_{n+1}，直到 $f(A_{n+1})-f(A_n)$ 最大值非正，将最终集合记为 A_G。

由于每次比较涉及 $|V|$ 个点、计算割值变化也为 $|V|$ 量级，且每次至少增加了一条边，因此算法时间复杂度为 $O(|V|^2|E|)$，是多项式量级。

另一方面，我们证明

$$2|E(A_G,A_G^c)|\geq|E|$$

而最大割值不可能超过边数，也就证明了近似比结论。

考虑 A_G 中的每个顶点 a，应有 $|E(a,A_G^c)|\geq|E(a,A_G)|$，否则将其放入 A_G^c 中能使割边增多，两侧对 A_G 中的 a 求和可知 (注意左侧的每条边计算了一次，右侧每条边计算了两次)

$$|E(A_G,A_G^c)|\geq 2|E(A_G,A_G)|$$

而对 A_G^c 中的顶点，完全同理可知

$$|E(A_G,A_G^c)|\geq 2|E(A_G^c,A_G^c)|$$

于是

$$2|E(A_G,A_G^c)|\geq|E(A_G,A_G)|+|E(A_G^c,A_G^c)|+|E(A_G,A_G^c)|=|E|$$

从而得证。

5. **验证对偶 Cheeger 问题可以等价为三值向量形式。**

设 $A=\{i\mid y_i=1\}$，$B=\{i\mid y_i=-1\}$，则只需证明

$$\frac{2|E(A,B)|}{\mathrm{vol}(A\cup B)}=1-\frac{\sum_{\{i,j\}\in E}|y_i+y_j|}{\sum_{i\in V}d_i|y_i|}$$

即可得到三值向量与 $A,B,(A\cup B)^c$ 的划分一一对应，下记 $C=\{i\mid y_i=0\}$、$M=A\cup B$。

先计算分母，由于

$$\sum_{i\in V}d_i|y_i|=\sum_{y_i=\pm1}d_i=\sum_{i\in M}d_i=\mathrm{vol}(M)$$

左右的分母相同，从而只需证明

$$2|E(A,B)|=\mathrm{vol}(M)-\sum_{\{i,j\}\in E}|y_i+y_j|$$

分类讨论可得 (注意无向图 E 中 (i,j) 与 (j,i) 是一条边)

$$\sum_{\{i,j\}\in E}|y_i+y_j| = \sum_{y_i=0,y_j=\pm 1}1+\sum_{y_i=y_j=1}2+\sum_{y_i=y_j=-1}2$$

也即其为

$$|E(M,C)|+2|E(A,A)|+2|E(B,B)|$$

而另一方面

$$\mathrm{vol}(M)=|E(M,C)|+2|E(M,M)|=|E(M,C)|+2|E(A,A)|+2|E(B,B)|+2|E(A,B)|$$

作差即得结论。

5 第五次作业

1. **证明 L 对应的广义特征值问题 $L\vec{x}=\lambda D\vec{x}$ 属于最大特征值的特征向量 $\vec{x}$ 满足**

$$\vec{x}=\arg\max_{\vec{x}\neq\mathbf{0}}\frac{\vec{x}^TL\vec{x}}{\vec{x}^TD\vec{x}}$$

由于考虑的是度数与边集，无任何边相连的点不计入考虑，因此可不妨设 D 为正对角阵。设 $\vec{y}=\sqrt{D}\vec{x}$，最大值问题即变为 (注意此时 D 对称且可逆，$\sqrt{D}$ 代表每个对角元开平方根)

$$\max_{\vec{y}\neq\mathbf{0}}\frac{\vec{y}^T\sqrt{D}^{-1}L\sqrt{D}^{-1}\vec{y}}{\vec{y}^T\vec{y}}$$

记 $\tilde{L}=\sqrt{D}^{-1}L\sqrt{D}^{-1}$，由于左右为相合可发现其仍为对称阵，因此设正交相似对角化为 $Q^T\tilde{D}Q$，再令 $\vec{z}=Q\vec{y}$，利用 Q 正交性可得问题化为

$$\max_{\vec{z}\neq\mathbf{0}}\frac{\vec{z}^T\tilde{D}\vec{z}}{\vec{z}^T\vec{z}}$$

由于 $\tilde{D}$ 为对角阵，不妨设 $\vec{z}^T\vec{z}=1$，可发现

$$\sum_i\tilde{D}_{ii}z_i^2\le\max_i\tilde{D}_{ii}\sum_j z_j^2=\max_i\tilde{D}_{ii}$$

且当 $\vec{z}$ 只有 $\tilde{D}$ 最大对角元对应的分量为 1，其他为 0 时可以取到，从而原问题的最大值为 $\tilde{D}$ 的最大对角元。

利用相似对角化，$\tilde{D}$ 最大对角元即 $\tilde{L}$ 最大特征值，即使得

$$\sqrt{D}^{-1}L\sqrt{D}^{-1}\alpha=\lambda\alpha$$

成立的最大 λ。设 $\alpha=\sqrt{D}\beta$，则有

$$L\beta=\lambda D\beta$$

利用 $\beta=\sqrt{D}^{-1}\alpha$ 可知原式最大值结果即为广义特征值问题的最大特征值。

当 $\vec{x}$ 为广义特征值问题属于最大特征值的特征向量时，直接代入可发现

$$\frac{\vec{x}^TL\vec{x}}{\vec{x}^TD\vec{x}}=$$

即为最大特征值，从而得证。

2. **从割值的估算出发验算最终的近似比结论。**

也即要计算

$$\min_{\varepsilon\in(0,1/16)}\frac{1-4\sqrt{\varepsilon}+8\varepsilon}{1-\varepsilon}$$

换元即

$$\min_{x\in(0,1/4)} \frac{1-4x+8x^2}{1-x^2}$$

其在 0 处为 1，$\frac{1}{4}$ 处为 $\frac{8}{15}$，而中间的最小值必须满足导数为 0，也即

$$(1-4x+8x^2)(-2x) = (1-x^2)(-4+16x)$$

整理得

$$2x^2-9x+2=0$$

其在范围内的解为

$$x = \frac{9-\sqrt{65}}{4}$$

代入即得

$$\frac{1-4x+8x^2}{1-x^2} = \frac{\sqrt{65}-7}{2} < \frac{8}{15}$$

从而这就是最小值，约为 0.531。

3. **证明上述图的基本量间的关系。**

(1) $\alpha(G)=\omega(\bar{G})$

由于原图中有边等价于补图中无边，原图中两两相邻 (即团) 等价于补图中两两不相邻 (即独立集)，从而原图最大团与补图最大独立集对应的顶点集合一致，得证。

(2) $\bar{\chi}(G)=\chi(\bar{G})$

对图的一个 k 染色，假设划分出的顶点集合为 $C_1,\dots,C_k$，由条件可知 C_i 内部不存在边，于是补图中 C_i 为团，由此，$C_1,\dots,C_k$ 是补图中的一个团覆盖。同理，对补图中的团覆盖 $C_1,\dots,C_k$，其在原图中对应一个 k 染色。由此，原图的染色与补图的团覆盖一一对应，且染色色数等于覆盖所用的团数，由此二者最小值相等。

(3) $\omega(G)\le\chi(G)$

由于存在一个大小为 $\omega(G)$ 的完全子图，对 G 的任何染色，其中所有点颜色不同，从而得证。

(4) $\alpha(G)\le\bar{\chi}(G)$

利用 (1)(3) 有 $\alpha(G)=\omega(\bar{G})\le\chi(\bar{G})=\bar{\chi}(G)$。

(5) $\tau(G)=|V|-\alpha(G)$

对 G 任何独立集 H，先证明 H^c 是一个顶点覆盖。若否，存在一条边与 H^c 无交，也即连接了 H 中两点，矛盾。反之，同理可得 H 是顶点覆盖时 H^c 为独立集，从而独立集与顶点覆盖一一对应，且顶点数和为 $|V|$，由此最大独立集大小与最小顶点覆盖大小和为 $|V|$。

4. **设 $c\in\mathbb{R}^n$、$b\in\mathbb{R}^m$、$A\in\mathbb{R}^{m\times n}$，定义原始问题与对偶问题分别为**

$$\min_{\Omega} c^Tx,\quad \Omega=\{x\in\mathbb{R}^n \mid Ax\ge b,\ x\ge 0\}$$

$$\min_{\Omega'} b^Tw,\quad \Omega'=\{w\in\mathbb{R}^m \mid w^TA\ge c^T,\ w\ge 0\}$$

考察它们的解的性质。

- **弱对偶性：**

  (1) $c^Tx\ge b^Tw$

  (2) $c^Tx\ge b^Tw^*, b^Tw\le c^Tx^*$

  (3) $c^Tx=b^Tw\Rightarrow c^Tx=c^Tx^*, b^Tw=b^Tw^*$

  (4) $\min c^Tx=-\infty\Rightarrow\Omega'=\varnothing$
  $\max b^Tw=+\infty\Rightarrow\Omega=\varnothing$

当 $a \geq 0$ 时，若 $s \geq t$，由于每个分量对应成立大于等于，作正组合仍有 $s^T a \geq t^T a$。由此，$c^T x \geq (A^T w)^T x = w^T A x = (Ax)^T w \geq b^T w$，第一个式子得证。由于对任何可行解都成立，当 x 或 w 取到一边最优解时仍成立，第二个式子得证。于是，当 $c^T x = b^T w$ 时，假设 $b^T w$ 取到的不是最大值，则会有 $b^T w' > c^T x$，矛盾，类似可知 $c^T x$ 取到的一定是最小值，第三个式子得证。最后，若 $c^T x$ 最小值无界，则任何 w 都无法满足 $c^T x \geq b^T w$ 对任何 x 成立，只能不存在可行解 w，第四个式子的另一边同理。

- **最优解存在性**：若原问题与对偶问题都有可行解，则它们都有最优解。

  分别记为 x_0 与 w_0，由弱对偶定理，任何可行解 x 满足 $c^T x \geq b^T w_0$，于是函数 $c^T x$ 有下界。这就排除了原问题问题可行域为空与无界的情况，从而必须有最优解，对对偶问题同理。

- **强对偶性**: 若原问题与对偶问题一方有最优解，则另一方亦有最优解，且最优值一致。

  记 $y = Ax - b$，则可写出问题原问题的标准形式，其中等式约束为 $\begin{pmatrix} A & -I \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = b$。记 a_j 为 $\begin{pmatrix} A & -I \end{pmatrix}$ 的第 j 列，c_j 在原有列后均应扩充 0。设原问题最优解 x^*, y^*，对应可行基分解为 B, N，根据线性规划的性质有 $c_B^T B^{-1} a_j - c_j \leq 0$。

  记 $w = c_B^T B^{-1}$，则上方条件在 $1 \leq j \leq n$ 时为 $A^T w \leq c$，在 $j > n$ 时为 $-w \leq 0$，于是 w 是对偶问题可行解，而计算得 $w^T b = c_B^T B^{-1} b$ 为原问题最优解，因此由弱对偶定理可知 w 为对偶问题最优解，最优值一致。

  由于对偶问题的对偶与原问题等价，若对偶问题有最优解，则同样得到原问题也有最优解，最优值一致。

6 第六次作业

1. **给出一个不满足非退化假设的线性规划问题，使其存在基础可行解到可行基的双射。**

   考虑标准形式的线性规划问题 $\min x$，约束条件为 $x = 0$、$x \geq 0$，其只有一组可行基 $\{1\}$，也只有一个基础可行解 (0)，虽然可行解是退化的，但仍然对应唯一的可行基。

2. **用管道舍入给出最大可满足性问题的 $\frac{\mathrm{e}}{\mathrm{e}-1}$ 近似算法。**

   最大可满足性问题为，给定合取式 F，其子句 $C_1, \ldots, C_m$ 包含布尔变量 $x_1, \ldots, x_n$，求满足的子句数量最大值。

   其整数规划形式为，求 $z_1 + \cdots + z_m$ 的最大值，使得

   $$\sum_{x_i \in C_j} y_i + \sum_{\bar{x}_i \in C_j} (1 - y_i) \geq z_j$$

   $$y_i \in \{0, 1\}, \quad z_j \in \{0, 1\}$$

   而由于最优解一定当且仅当求和为 0 时 $z_j = 0$，上式可以写为，求 $y_i \in \{0, 1\}$ 时

   $$\sum_{j=1}^{n} \min \left\{ 1, \sum_{x_i \in C_j} y_i + \sum_{\bar{x}_i \in C_j} (1 - y_i) \right\}$$

   设

   $$L(y) = \sum_{j=1}^{n} \min \left\{ 1, \sum_{x_i \in C_j} y_i + \sum_{\bar{x}_i \in C_j} (1 - y_i) \right\}$$

   $$F(y) = \sum_{j=1}^{n} \left(1 - \prod_{x_i \in C_j} (1 - y_i) \prod_{\bar{x}_i \in C_j} y_i \right)$$

   与课上完全相同 (由于 x_i 与 $\bar{x}_i$ 不可能同时出现在 C_j 中，这里所有项事实上独立取值) 可证明

   $$\forall i, y_i \in \{0, 1\} \Longrightarrow F(y) = L(y)$$

$$\forall i, y_i \in [0,1] \Longrightarrow F(y) \geq \left(1 - \frac{1}{\mathrm{e}}\right) L(y)$$

由此，对整数规划形式求解 y_i，并利用 F 进行管道舍入：对 y_i 的每个不在 $\{0,1\}$ 中的分量，比较其变为 0 与变为 1 后 F 的大小，并选取较大的作为结果 (由于此处无需保证总和相同，对每个分量进行即可)。由于 F 关于 y_i 的每个分量是线性的，舍入过程 F 不减，这就得到了近似比结论。

3. **证明对弱超模函数 f 与 G 的子图 F，$f(S) - |\delta_F(S)|$ 仍为弱超模函数。**

先证明

$$|\delta_F(A)| + |\delta_F(B)| \geq |\delta_F(A \cap B)| + |\delta_F(A \cup B)|$$

将 $A \cap B$、$A - B$、$B - A$、$(A \cup B)^c$ 分别记为顶点集合 1、2、3、4，并将顶点集 i 指向顶点集 j 且在 F 中的边数记为 e_{ij}，则左侧为

$$(e_{13} + e_{14} + e_{23} + e_{24}) + (e_{12} + e_{14} + e_{32} + e_{34})$$

右侧为

$$(e_{12} + e_{13} + e_{14}) + (e_{14} + e_{24} + e_{34})$$

由此左减右为 $e_{23} + e_{32}$，从而得证。

再证明

$$|\delta_F(A)| + |\delta_F(B)| \geq |\delta_F(A - B)| + |\delta_F(B - A)|$$

与之前相同定义 e_{ij}，此时右侧为

$$(e_{21} + e_{23} + e_{24}) + (e_{31} + e_{32} + e_{34})$$

由无向性 $e_{ij} = e_{ji}$，由此左减右为 $2e_{14}$，从而得证。

综合以上两式，只要

$$f(A) + f(B) \leq f(A - B) + f(B - A)$$

即有

$$(f(A) - |\delta_F(A)|) + (f(B) - |\delta_F(B)|) \leq ((f(A - B) - |\delta_F(A - B)|)) + (f(B - A) - |\delta_F(B - A)|)$$

只要

$$f(A) + f(B) \leq f(A \cap B) + f(A \cup B)$$

即有

$$(f(A) - |\delta_F(A)|) + (f(B) - |\delta_F(B)|) \leq ((f(A \cap B) - |\delta_F(A \cap B)|)) + (f(A \cup B) - |\delta_F(A \cup B)|)$$

从而成立。

4. **证明迭代舍入算法给出了原问题的一个 3-近似解。**

假设迭代进行次数为 t 次，对应得到的 F 为 $F_1, \dots, F_t$，F_t 即为最终的 F。由于算法已经保证了解的可行性，只需证明近似性质。

最后一次迭代前已经设置了 F_{t-1} 中的 x_e 为 1，而可行性要求为 $x_e \in [0,1]$ 且

$$\sum_{\delta_{G - F_{t-1}}(S)} x_e \geq f(S) - |\delta_{F_{t-1}}(S)|$$

由于 $\delta_G(S) = \delta_{G - F_{t-1}}(S) \cup \delta_{F_{t-1}}(S)$，且此并无交，利用舍入方式可知最后一次迭代的解满足

$$\sum_e c_e x_e = \sum_{e \in F} c_e \leq \sum_{e \in F_{t-1}} c_e + 3 \sum_{e \in F_t - F_{t-1}} c_e x_e^{(t)} \leq \sum_{e \in F_{t-1}} c_e + 3 \sum_{e \in G - F_{t-1}} c_e x_e^{(t)}$$

这里 $x_e^{(t)}$ 表示第 t 次迭代中未舍入的最优解。另一方面，由于第 t 次迭代比第 $t-1$ 次迭代固定了更多 x_e 为 1，其最优值不应超过第 $t-1$ 次迭代的最优值，也即

$$\sum_{e\in F_{t-1}-F_{t-2}} c_e + \sum_{e\in G-F_{t-1}} c_e x_e^{(t)} \le \sum_{e\in G-F_{t-2}} c_e x_e^{(t-1)}$$

再由 $x_e^{(t-1)} \le 1$ 可知

$$\sum_{e\in G-F_{t-1}} c_e x_e^{(t)} \le \sum_{e\in G-F_{t-1}} c_e x_e^{(t-1)}$$

于是可得到

$$\sum_e c_e x_e \le \sum_{e\in F_{t-1}} c_e + 3\sum_{e\in G-F_{t-1}} c_e x_e^{(t-1)}$$

重复上述过程即可最终得到

$$\sum_e c_e x_e \le \sum_{e\in F_0} c_e + 3\sum_{e\in G-F_0} c_e x_e^{(1)}$$

而由 $F_0=\varnothing$，此式即

$$\sum_e c_e x_e \le 3\sum_{e\in G} c_e x_e^{(1)}$$

由于右侧求和为第一次迭代后的最优值，也即原问题松弛后的最优值，其必然小于等于 opt，即得证近似解不超过 $3\,\mathrm{opt}$。