Readme

Q1:

Introduction:

In Q7, to select a set V of k views such that Gain is maximized, we adopted the following methods: construct the tree and compare the gain for every combination of distinct nodes, choose the highest one.

The result is :

the k=3 views are: **abc ade bcde**

MaxGain is: **197.60000000000002**

Methods:

1. Construct the tree to save nodes, and for every node:
   a) Attribution: name\ value\ children (the children is a list to save all the nodes connected to it)
   b) Methods:
      i. Add: add the children nodes list to itself
      ii. Change_to_node: when choose K=3 views of nodes, we change the key to the nodes, so we can compare the value between initdict and tempdict, so we can choose the highest.
2. Chose distinct three nodes randomly and compute the gain.
3. Select the highest gain.

Q2:

get a lower bound of **1.385383.**

Proof.

Since $b_j > a_i > 1 + x + y$ the minimum spanning tree has length

$$mst = ((6 + 2x)k + 2(1 + x + y)(k - 1))\, l. \tag{7}$$

The change in length of a minimun spanning tree after selection of a star $A_i \cup N(A_i)$ is $2(1 + x + y)l$. Now we have to find the minimizing vertical component after the stars $(A_0 \cup N(A_0)) \cup \ldots \cup (A_{i-1} \cup N(A_{i-1}))$ have been selected. For a fixed h with $1 \leq h \leq l$ let $S_h = \{w_{i,h} | 1 \leq i \leq 4k\}$. For the ith step, using symmetry and monotonicity of the function f_i, it is enough to consider the subsets $S_h^{i,1} = S_h \backslash (\{w_{4j,h} | 1 \leq j \leq i\} \cup \{w_{4j-1,h} | 1 \leq j \leq i\})$ and $S_h^{i,2} = \{w_{j,h} | 4i + 1 \leq j \leq 4k\}$.

For the selection function of $N(A_i)$ we get

$$\begin{aligned} f_i(N(A_i)) &= \frac{l+1}{l} \cdot \frac{a_i(k,l,x,y)}{2(1+x+y)} \leq \min(f_i(S_h^{i,1}), f_i(S_h^{i,2})) \\ &= \min\left(\frac{(4 + 2x + y)k - (2 + x)i}{\frac{mst}{l} - (4 + 2x)i}, \right. \end{aligned}$$

$$
\left. \frac{(k-i)(4+2x+y)}{(k-i)(6+2x)+(k-i-1)(2+2x+2y)} \right)
$$
$$
= \quad f_i(S_h^{i,1}). \tag{8}
$$

Therefore we can specify the value of

$$
a_i(k,l,x,y) = \frac{l}{l+1} \cdot \frac{2(1+x+y)\left((4+2x+y)k-(2+x)i\right)}{\frac{mst}{l}-(4+2x)i}. \tag{9}
$$

Notice that a_i is monotone increasing. After all A-stars are selected the argument for selecting the stars with center B_i is similar. Only the change in length of a minimun spanning tree after selection of a star $B_i \cup N(B_i)$ has changed and is $2+2x$.

Again using monotonicity, the minimizing vertical component is

$$
S_h^{i,3} = \{w_{4j+\kappa,h} | j \le i < k-1, 1 \le \kappa \le 2\} \cup \{w_{4k,h}, w_{4k-1,h}\}
$$

after every A-star and the stars $B_j \cup N(B_j)$ with $j < i < k-1$ have been contracted. For the selection function of $N(B_i)$ we get

$$
\begin{aligned}
f_{k-i+1}(N(B_i)) &= \frac{l+1}{l} \cdot \frac{b_i(k,l,x,y)}{2+2x} \\
&\le f_{k-i+1}(S_h^{i,3}) \\
&= \frac{(k-i)(2+x+y)+2+x}{(k-i)(4+2x)+2}.
\end{aligned} \tag{10}
$$

Therefore we can specify the value of

$$
b_i(k,l,x,y) = \frac{l}{l+1} \cdot 2(1+x) \cdot \frac{(k-i)\cdot(2+x+y)+2+x}{(k-i)\cdot(4+2x)+2}. \tag{11}
$$

Notice for the last star $B_{k-1} \cup N(B_{k-1})$ the component $S_h^{k-1,3}$ consists of only four terminals and there is no edge with costs y in $SMT(S_h^{k-1,3})$. So we get

$$
b_{k-1}(k,l,x,y) = \frac{l}{l+1} \cdot 2(1+x) \cdot \frac{2+x}{3+x}. \tag{12}
$$

Notice that b_i is monotone increasing.

After every star $A_i \cup N(A_i)$ and $B_i \cup N(B_i)$ is selected the algorithm has to choose full components of size two. This can only be satisfied if $4k < \min((2+2x+y)k+1, (3+2x)k)$, because otherwise the algorithm chooses the components $\{w_{2i,h} | 1 \leq i \leq 2k\} \cup \{w_{4k,h}\}$ resp. $\{w_{4i+j,h} | 0 \leq i < k, 2 \leq j \leq 4\}$.

If every contraint is satisfied we get a lower bound for the performance ratio of

$$\frac{4kl + (l+1)\sum_{i=0}^{k-2} a_i + (l+1)\sum_{i=0}^{k-1} b_i}{(4+2x+y)kl}. \tag{13}$$

This is the case for the values $k = 5, l \geq 15, x = 0.8688$ and $y = 0.063$ and so we get a lower bound of 1.385383. □

References:

[1] Lower Bounds for the Relative Greedy Algorithm for Approximating Steiner Trees.

[2] Proof methods and greedy algorithms. Magnus Lie Hetland Lecture notes, May 5th 2008

[3] http://www.cs.cornell.edu/courses/cs482/2007su/exchange.pdf