6.S091: Problem Set 3

Suyeol Yun
syyun@mit.edu

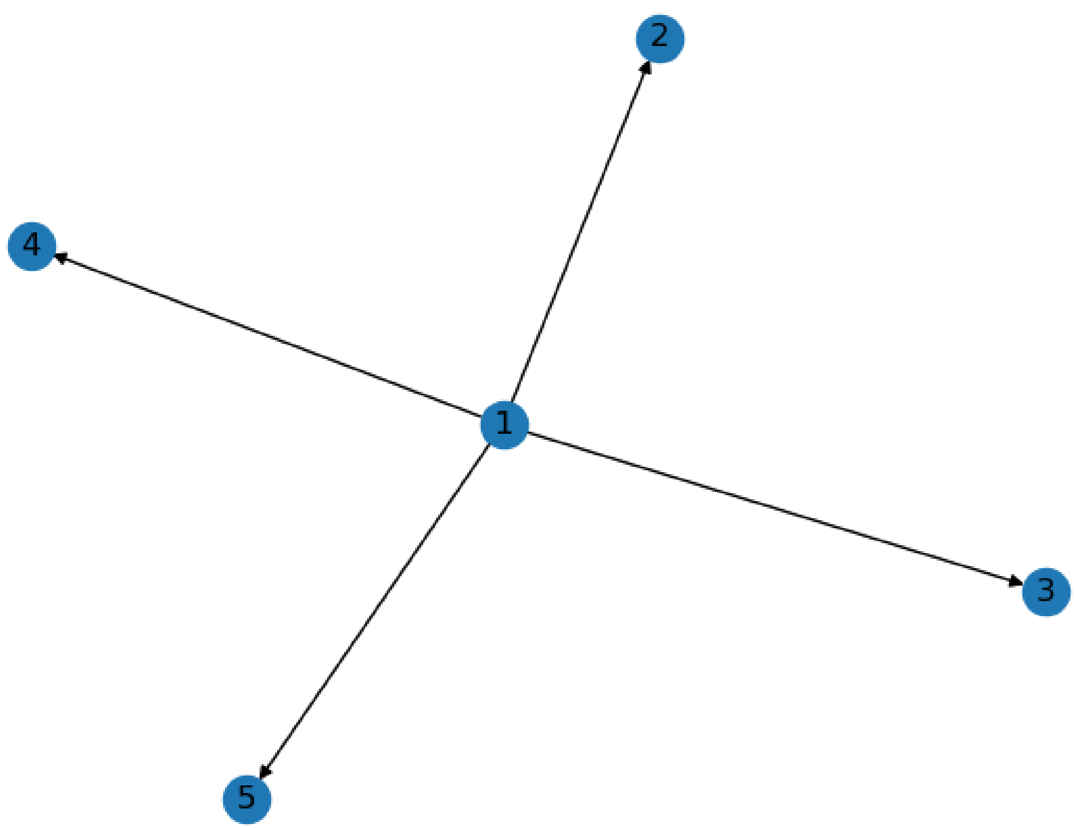
February 1, 2023

Problem 1: Constructing Minimal I-MAPs [5 points]

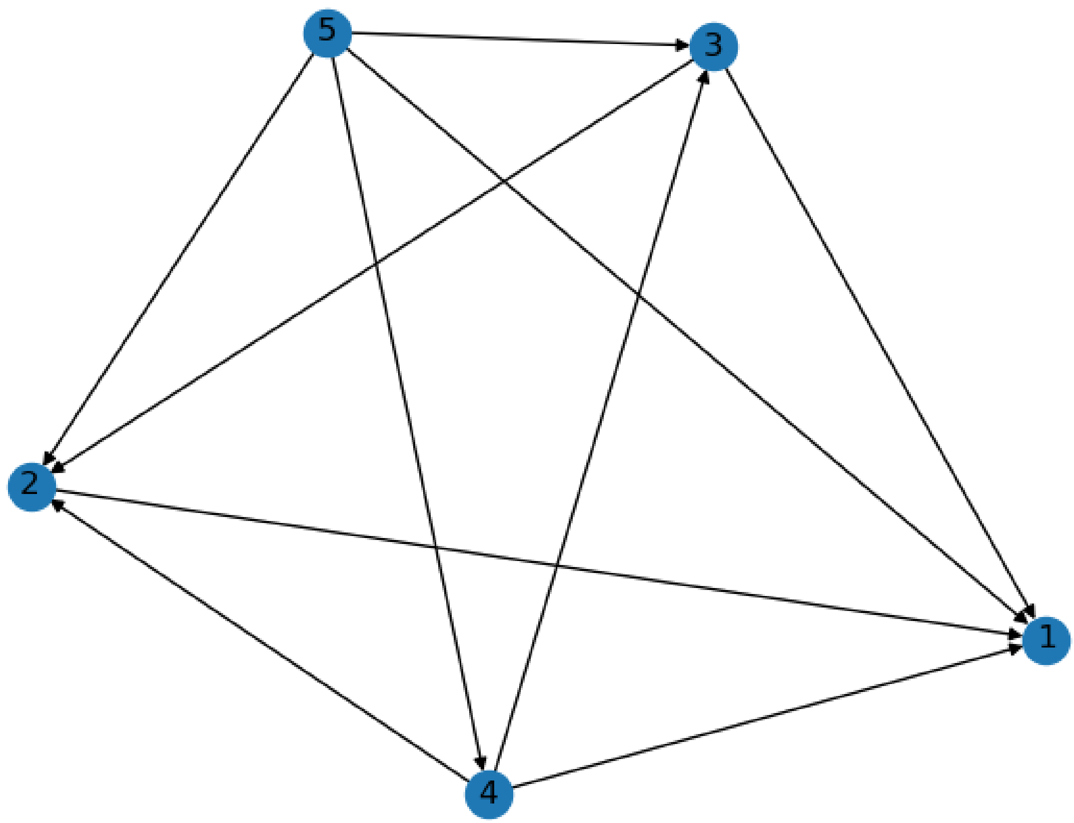
* Code is available at https://github.com/syyunn/6.S091/blob/main/pset3/pb1.py

(a)

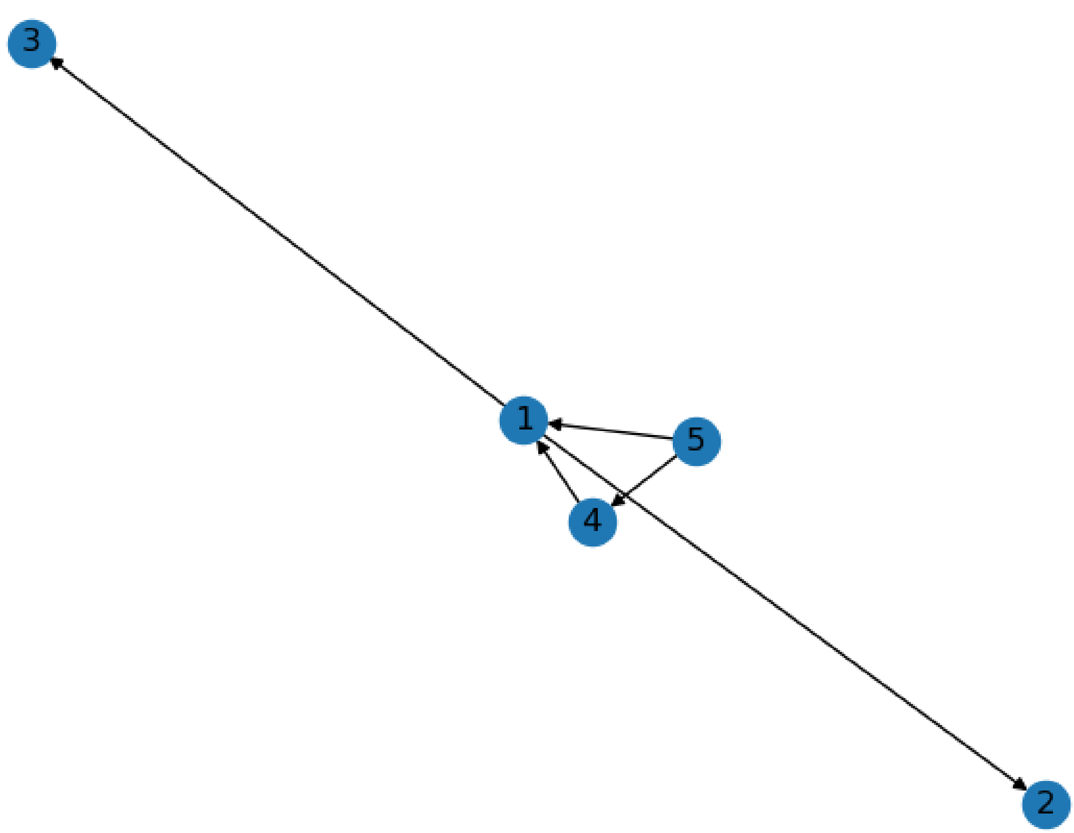
1. Draw $\mathcal{G}_{\pi_a}$



2. Draw $\mathcal{G}_{\pi_b}$



3. Draw $\mathcal{G}_{\pi_c}$



(b)

We can transform $\mathcal{G}_{\pi_a}$ into $\mathcal{G}_{\pi_c}$ by the Chickering sequence (Add $5 \to 4$, Reverse $1 \to 5$, then Reverse $1 \to 4$).

Problem 2

(a)

Let γ_m be an arbitrary node of the d-connecting path γ. Since γ is d-connected, γ_m is unblocked over the path from $A \in \mathbf{A}$ to $B \in \mathbf{B_1}$ given $\mathbf{S} \cup \mathbf{B_2}$. Since γ_m is unblocked, γ_m is either a non-collider or a collider. First, if γ_m is a non-collider, $\gamma_m \notin \mathbf{S} \cup \mathbf{B_2}$ and it implies $\gamma_m \notin \mathbf{S}$, which implies that γ_m is unblocked over the path from $A \in \mathbf{A}$ to $B \in \mathbf{B}$ given $\mathbf{S}$ because $B \in \mathbf{B_1} \subset \mathbf{B}$. Second, if γ_m is a collider, then $\overline{\mathrm{de}}_{\mathcal{G}}(\gamma_m) \cap (\mathbf{S} \cup \mathbf{B_2}) \neq \emptyset$. Let $\zeta_m \in \overline{\mathrm{de}}_{\mathcal{G}}(\gamma_m)$ be the closest node to γ_m among the nodes in $\overline{\mathrm{de}}_{\mathcal{G}}(\gamma_m) \cap (\mathbf{S} \cup \mathbf{B_2})$. Then ζ_m is either in $\mathbf{S}$ or in $\mathbf{B_2} \setminus \mathbf{S}$. If $\zeta_m \in \mathbf{S}$, γ_m is unblocked over the path from $A \in \mathbf{A}$ to $B \in \mathbf{B}$ given $\mathbf{S}$ because $\overline{\mathrm{de}}_{\mathcal{G}}(\gamma_m) \cap \mathbf{S} \neq \emptyset$. If $\zeta_m \in \mathbf{B_2} \setminus \mathbf{S}$, any nodes over the decendent path between γ_m to ζ_m are not in $\mathbf{S}$ thus they are unblocked over the path from $A \in \mathbf{A}$ to $\zeta_m \in \mathbf{B_2} \subset \mathbf{B}$ given $\mathbf{S}$ because they are non-colliders and they are not in $\mathbf{S}$. Therefore, we can always find a d-connecting path γ' between $A \in \mathbf{A}$ and $B' \in \mathbf{B}$ given $\mathbf{S}$.

(b)

B can be either in $\mathbf{B_1}$ or $\mathbf{B_2}$. If $B \in \mathbf{B_2}$, then the path γ can be a d-connecting path γ' from $A \in \mathbf{A}$ to $B' \in \mathbf{B_2}$ given $\mathbf{S}$. Otherwise, if $B \in \mathbf{B_1}$, let γ_m be an arbitrary node of the d-connecting path γ. Then γ_m is either non-collider or collider. If γ_m is a collider, $\overline{\mathrm{de}}_{\mathcal{G}}(\gamma_m) \cap \mathbf{S} \neq \emptyset$ which implies $\overline{\mathrm{de}}_{\mathcal{G}}(\gamma_m) \cap (\mathbf{S} \cup \mathbf{B_2}) \neq \emptyset$ which implies γ_m is unblocked over the path from $A \in \mathbf{A}$ to $B \in \mathbf{B_1}$ given $\mathbf{S} \cup \mathbf{B_2}$. If γ_m is non-collider, $\gamma_m \notin \mathbf{S}$. If γ_m is also not in $\mathbf{B_2}$, then γ_m is unblocked over the path from $A \in \mathbf{A}$ to $B \in \mathbf{B_1}$ given $\mathbf{S} \cup \mathbf{B_2}$. If γ_m is not in $\mathbf{S}$ but in $\mathbf{B_2}$, we can shorten the path γ by removing the trailing nodes of γ_m from γ. This shortened path is d-connecting path γ' from $A \in \mathbf{A}$ to $B' \in \mathbf{B_2}$ given $\mathbf{S}$. Therefore, we can always find a d-connecting path of either one of the given two types.

Problem 3

* Code is available at https://github.com/syyunn/6.S091/blob/main/pset3/search_mec.py

(a)

starting_dag2 and 3 both have 4 neighbors.

(b)

1. starting_dag2 has the shortest path of length 1.
2. starting_dag3 has the shortest path of length 3.