Subset Systems

Definition
A *subset system* $S = (E, \mathcal{I})$ is a finite set $E$ with a collection $\mathcal{I}$ of subsets $E$ such that:

$$\text{if } A \in \mathcal{I} \text{ and } B \subset A \text{ then } B \in \mathcal{I}$$

i.e., “$\mathcal{I}$ is closed under inclusion”

Example

1. $E = \{e_1, e_2, e_3\}$, $\mathcal{I} = \{\{e_1, e_2\}, \{e_2, e_3\}, \{e_1\}, \{e_2\}, \{e_3\}, \{\}\}$
2. $E$ is the edges of a graph and $\mathcal{I}$ is the acyclic subsets of edges
3. $E$ is the edges of a graph and $\mathcal{I}$ are the **matchings**, i.e., subsets of edges such that no two edges share a vertex
Generic Problem and Greedy Algorithms

Problem Given a subset system \( S = (E, \mathcal{I}) \) and weight function \( w : E \rightarrow \mathbb{R}^+ \), find \( A \in \mathcal{I} \) such that \( w(A) = \sum_{e \in A} w(e) \) is maximized.

Algorithm (Greedy)

1. \( A = \emptyset \)
2. Sort elements of \( E \) by non-increasing weight
3. For each \( e \in E \): If \( A + e \in \mathcal{I} \) then \( A \leftarrow A + e \)

For what subset systems does this give optimal results?
Examples

Example
Let $E = \{e_1, e_2, e_3\}$, $I = \{\{e_1, e_2\}, \{e_2, e_3\}, \{e_1\}, \{e_2\}, \{e_3\}, \{\}\}$, and $w(e_1) = 3$, $w(e_2) = 1$, and $w(e_3) = 4$. The greedy algorithm returns

\[\{e_2, e_3\}\]

and this is optimal.

Example (Maximum Weight Forest)
$E$ is the edges of a graph and $I$ is the acyclic subsets of edges. This is essentially the same as the MST and greedy does work.

Example (Maximum Weight Matching)
$E$ is the edges of a graph and $I$ are the matchings. Greedy does not work.
Matroid Definition and Theorem

Definition
Subset system \((E, \mathcal{I})\) has the exchange property if

\[
\forall A, B \in \mathcal{I} : (|A| < |B|) \implies (\exists e \in B - A \text{ such that } A + e \in \mathcal{I})
\]

Definition
A matroid is a subset system \(M = (E, \mathcal{I})\) with the exchange property

Theorem
Given a subset system \((E, \mathcal{I})\), the following statements are equivalent:
1. Greedy algorithm returns optimal solution for any weight function.
2. The subset system obeys the exchange property, i.e., it’s a matroid.
Matroid implies Greedy Algorithm is Optimal

- Proof by contradiction: Assume \((E, \mathcal{I})\) is a matroid and let
  
  - greedy solution: \(A = \{e_1, e_2, \ldots, e_k\}\)
  - optimal solution: \(B = \{f_1, f_2, \ldots, f_{k'}\}\) where \(w(B) > w(A)\)

- Can deduce \(k = k'\) by the exchange property. (Both solutions are maximal and if \(k \neq k'\) then the exchange property would imply an element from the larger set could be added to the smaller set).

- Can assume by reordering
  
  \[
  w(e_1) \geq w(e_2) \geq \ldots \geq w(e_k) \\
  w(f_1) \geq w(f_2) \geq \ldots \geq w(f_{k'})
  \]

- Consider smallest such \(s\) with \(w(f_s) > w(e_s)\).

- Consider subsets of \(A\) and \(B\):
  
  \[
  \alpha = \{e_1, e_2, \ldots, e_{s-1}\} \quad \text{and} \quad \beta = \{f_1, f_2, \ldots, f_s\}
  \]

- By the exchange property there exists \(t \in [s]\) such that:
  
  \[
  f_t \in \beta - \alpha \quad \text{with} \quad \alpha + f_t \in \mathcal{I}
  \]

- But then \(w(f_t) \geq w(f_s)\) and hence \(w(f_t) > w(e_s)\)

- Contradiction since greedy algorithm picked \(e_s\) rather than \(f_t\)
Greedy Algorithm Always Optimal implies \((E, \mathcal{I})\) is Matroid

- Sufficient to show that greedy may not work if \((E, \mathcal{I})\) isn’t a matroid
- \((E, \mathcal{I})\) not a matroid implies that
  \[
  \exists A, B \in \mathcal{I} \text{ such that } |A| < |B| \text{ and } \nexists e \in B - A \text{ with } A + e \in \mathcal{I}
  \]
- Let \(m = |A|\) and \(n = |E|\). Define weight function:
  \[
  w(e) = \begin{cases} 
  m + 2 & \text{if } e \in A \\
  m + 1 & \text{if } e \in B - A \\
  1/(2n) & \text{otherwise}
  \end{cases}
  \]
- Greedy algorithm returns \(A\) with weight at most \((m + 2)m + 1/2\) but a better solution is \(B\) with weight at least \((m + 1)^2\)