8. Matroids

A matroid is a structure of the form (S, J)

$$\begin{aligned} S &\leftarrow \text{Finite ground set} \\ J &\leftarrow \text{Collection of subsets of } S \end{aligned}$$

A matroid has to satisfy 2 properties

1. **Non-emptiness -** The empty set is in J, hence J cannot be empty.
2. **Hereditary -** If some $A \subseteq S, A \in J$, then all subsets of A are also in J.
3. **Exchange -** If $A, B \in J$, and $|A| < |B|$, then $\exists x \in B \backslash A, A \cup \{x\} \in J$.
4. **Extended Exchange argument -** If $A, B \in J$, and $|A| \leq |B|$, then $\forall x \in A - B, \exists y \in B - A$ such that $B - \{y\} \cup \{x\} \in J$.

An example of a Matroid

$$(S, J) = (\{1, 2, 3\}, \{\{\}, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{2, 3\}, \{1, 3\}\})$$

We can verify that all the properties are satisfied in this tuple.

Definitions

- **Independent Sets -** Every set in J is called an independent set, any set $\notin J$ is dependent.
- **Maximal Independent Set (Basis) -** Any independent set that is not a proper subset of any other independent set.
- **Rank of Matroid** (S, J) **-** Defined as the cardinality of a basis i.e. $\max_{A \in J} |A|$.

Claim - All basis sets have the same cardinality
Proof -
Proof by contradiction

Assume there exists basis sets $A, B \in J$, $|A| < |B|$.
By the exchange property, we know that $\exists x \in B \backslash A$.
Hence, $A' = A \cup \{x\} \in J, A \subset A'$

Which goes against the definition of A being a basis set, hence there's a contradiction.

Examples of Matroids

Graphic Matroid

$M(G)$, where G is an undirected connected graph.

- $S = E$
- J: all subsets of E that form an acyclic sub graph of G. Hence the bases are the spanning trees of G.

Uniform Matroid

$U_{n,k}$
$S = \{1, 2, 3, \ldots, n\}$
J: all subsets of S of size at most k.

Matching Matroid

$S = V$
J: sets of vertices that are covered by a matching.
Matching - M is a subset of E s.t every vertex has at most 1 edge from M incident on it.

Proving that Graphic Matroid is a matroid

- $S = E$
- J: all subsets of E that form an acyclic sub graph of G. Hence the bases are the spanning trees of G.
  Non-emptiness - Trivially satisfied, as an empty edge set creates no cycles.
  Hereditary - Trivially satisfied as well, as picking a subset of an edge set cannot create cycles.

Proving Exchange Property

[Reference]
Suppose $(V, X), (V, Y)$ be two graphs where $|X| < |Y|, X, Y \in J$

Let $(U_1, X_1), (U_2, X_2), \ldots, (U_{k_1}, X_{k_1})$ be the connected components of (V, X)
Let $(W_1, Y_1), (W_2, Y_2), \ldots, (W_{k_2}, Y_{k_2})$ be the connected components of (V, Y)
We know that X_i, Y_j are all acyclic as $X, Y \in J$.

For a tree $T = (V, E), |E| = |V| - 1$

$$\therefore |X| = \sum_{i=1}^{k_1} |X_i| = |V| - k_1$$

Similarly, $|Y| = |V| - k_2$
$|X| < |Y| \implies k_1 > k_2$

If we assumed that every connected component of Y lied in a connected component of X, it would imply that $k_2 \geq k_1$, which is false. Therefore, there exists some connected component (W_p, Y_p) such that it intersected with components (U_a, X_a) and (U_b, X_b). Hence, Y_k contains some edge $e = (u, v)$ between U_a and U_b, which we know are disconnected, hence adding e to (V, X) will not create a cycle.

Therefore, we've proven the exchange property.

Weighted matroid

Comes with (S, J, w)
$w : S \to R$
Which essentially means each element in S has been assigned some weight.

$$\forall A \in J,$$
$$w(A) = \sum_{x \in A} w(x)$$

We are going to take a look at an algorithm which attempts to find the maximum weight maximal independent set.

Algorithm

1. Sort elements of S in non-increaseing order of weights, $x_1, x_2, \ldots, x_{|S|}$
2. $A \leftarrow \{\}$
3. For i from 1 to $|S|$:
    1. If $A \cup \{x_i\} \in J$, then $A \leftarrow A \cup \{x_i\}$

4. Return A

Proving A is a maximal independent set

Reference

A is an Independent Set

Proof -
We know it's an independent set as we've always checked before adding if the resultant $\in J$.

A is maximal

Proof -
Proof by contradiction, suppose A is not maximal.
This implies that $\exists C \in J$ s.t $A \subsetneq C \implies \exists x_j \in C - A$.
Now, let's take a look at the elements that we had picked before index j in the algorithm.

$$B = \{x_i \in A | i < j\} \subset A$$

We'll now define A_k as the set A after the k^{th} iteration. Therefore, $B = A_{j-1}$.
Now, in the j^{th} iteration, we would've tested if

$$A_{j-1} \cup \{x_j\} \in J$$

Since we know that $x_j \in C \backslash A$, $x_j \notin A$. This must mean that we found that

$$A_{j-1} \cup \{x_j\} \notin J$$

However, we know that

$$A_{j-1} \subset C, x_j \in C \implies A_{j-1} \cup \{x_j\} \subseteq C$$

And since C is an independent set, it implies all subsets, including $A_{j-1} \cup \{x_j\} \in J$ are independent sets (hereditary property), hence we arrive at a contradicton.

Therefore, A cannot be the proper subset of any other independent set, proving the fact that A is a maximal independent set.

Preliminary Proof

Claim - Let B be another maximal independent set, let a_s be the smallest index in $A - B$, let a_t be the smallest index in $B - A$, then $s < t$

Proof -

Proof by contradiction, assume $s > t$ ($s = t$ is not possible).

We then create a set

$$C = \{a_i \in A | i < t\}$$

By definition, $C = A_{t-1}, C \subset B$.

Hence, for $x_t \notin A$, we must have found in the t^{th} iteration that

$$A_{t-1} \cup \{x_t\} \notin J$$

But, we know that

$$C \subset B, x_t \in B \implies C \cup \{x_t\} \subseteq B$$

Which implies that $C \cup \{x_t\} \in J$ by the hereditary property, which is a contradiction. Hence, $s < t$ is proved.

Proving A is a max weight maximal independent set

Proof -

Proof by contradiction, assume that A is not a max weight set.

Suppose $B_1, B_2, \ldots, B_r$ are all the max weight maximal independent sets.

We then define $s_1, s_2, \ldots, s_r$ as well as $t_1, t_2, \ldots, t_r$. WLOG, we assume that s_1 is the largest index among all, and work with B_1

Now, we'll make use of the *extended exchange argument* property of the matroid, by which we know that $\exists x_j \in B_1 \backslash A$, such that

$$B' = (B_1 \cup \{x_{s_1}\}) \backslash \{x_j\} \in J$$

Now, we know that $j \geq t_1$, as t_1 is the smallest index in $B_1 \backslash A$, and we also know that $t_1 > s_1$.

$$\therefore j \geq t_1 > s_1$$

Since we've ordered the sets in non-increasing order,

$$x_j \leq x_{s_1}$$

Since we are getting a net gain weight ≥ 0 by switching to set B',

$$\mathrm{wt}(B') \geq \mathrm{wt}(B_1)$$

But B_1 is a max weight independent set, hence

$$\mathrm{wt}(B') = \mathrm{wt}(B_1)$$

therefore, B' is also a max weight maximal independent set, hence corresponding to some index p in $B_1, B_2, \ldots, B_r$.

Since

$$A - B' = A - B - \{x_{s_1}\}$$

It implies that the smallest index in $A - B'$ must be greater than s_1, as we added that to B to get B'. This implies that $s_p > s_1$, which contradicts our assumption that s_1 was the largest index among all s_i.
Hence, we've proven that A must be a maximum weight maximal independent set.

Solving Minimum Weight Maximal Independent Set

This can be done by replacing the weight function w with w',

$$w'(a) = -w(a) \forall a \in S$$

Time Complexity of Algorithm

Suppose it takes f time to determine membership if J (assuming it's independent of size you're checking), then the time complexity is

$$O(n \log n + n \cdot f)$$

If f was dependent upon r (the size of set being tested) , then we'd define the time complexity as

$$O(n \log n + \sum_{r=1}^{n} f(r))$$

Problem - Given an edge weighted graph (V, E, w). The weight of any subgraph (V', E') is

$$\sum_{e \in E'} w(e)$$

Then we would like to compute a spanning tree if the given graph such that its weight is minimum. This can by solved by the matroid algorithm given above.