Testing sortedness: recap

Let $G_S = (A, S)$ be a graph such that every pair of vertices in A are connected by a path of length at most 2. There is such a graph with $|S| = n \log n$ (draw). (middle vertex connected to all n-1 vertices; recurse on first half and second half)

repeat $O(\log n)$ times:
    pick an edge ij from S uniformly at random
    check whether $A[i]$ and $A[j]$ are in the correct order
if all the checks are successful
    output "The array is nearly sorted"

analysis

- If A is sorted, the output is correct
- If A is nearly sorted, with fewer than ϵn out of order items, then we are allowed to say sorted or unsorted and we are right
- If A is far from sorted (at least ϵn entries are out of place), then:
  claim: at least $\frac{\epsilon}{4}n$ edges of S would fail
  will give us:
  $P(\text{success in a single guess}) = \frac{\epsilon n/4}{n \log n} = \frac{\epsilon}{4 \log n}$ and
  $P(\text{fail after k guesses}) = (1 - \frac{\epsilon}{4 \log n})^k < .01$ if $k = \frac{20}{\epsilon} \log n$

proof of claim that $\geq \frac{\epsilon}{4}n$ edges of S would fail

Let $G_W = (A, E_w)$, E_w are *unsorted* edges
we need to show $|S \cap E_W| \geq \epsilon n/4$

- let O be the set of out of place elements

proof of claim that $\geq \frac{\epsilon}{4}n$ edges of S would fail

Let $G_W = (A, E_w)$, E_w are *unsorted* edges
we need to show $|S \cap E_W| \geq \epsilon n/4$

- let O be the set of out of place elements
- every edge in E_W has an endpoint in O

proof of claim that $\geq \frac{\epsilon}{4}n$ edges of S would fail

Let $G_W = (A, E_w)$, E_w are *unsorted* edges
we need to show $|S \cap E_W| \geq \epsilon n/4$

- let O be the set of out of place elements
- every edge in E_W has an endpoint in O
- let M be a maximal matching of G_W
  every $e \in M \subseteq E_W$ has an endpoint in O so $|O| \leq 2|M|$

proof of claim that $\geq \frac{\epsilon}{4}n$ edges of S would fail

Let $G_W = (A, E_w)$, E_w are *unsorted* edges
we need to show $|S \cap E_W| \geq \epsilon n/4$

- let O be the set of out of place elements
- every edge in E_W has an endpoint in O
- let M be a maximal matching of G_W
  every $e \in M \subseteq E_W$ has an endpoint in O so $|O| \leq 2|M|$
- *mark* the endpoints V_m of $S \cap E_W$: $|S \cap E_W| \geq |V_m|/2$ (since every edge is counted twice by the vertices)

proof of claim that $\geq \frac{\epsilon}{4}n$ edges of S would fail

Let $G_W = (A, E_w)$, E_w are *unsorted* edges
we need to show $|S \cap E_W| \geq \epsilon n/4$

- let O be the set of out of place elements
- every edge in E_W has an endpoint in O
- let M be a maximal matching of G_W
  every $e \in M \subseteq E_W$ has an endpoint in O so $|O| \leq 2|M|$
- *mark* the endpoints V_m of $S \cap E_W$: $|S \cap E_W| \geq |V_m|/2$ (since every edge is counted twice by the vertices)
- for any $(i, j) \in M$, either $(i, j) \in S$ or $\exists k \; s.t. (i, k), (k, j) \in S$, since every pair of vertices are at most 2 hops away in G_S.

proof of claim that $\geq \frac{\epsilon}{4}n$ edges of S would fail

Let $G_W = (A, E_w)$, E_w are *unsorted* edges
we need to show $|S \cap E_W| \geq \epsilon n/4$

- let O be the set of out of place elements
- every edge in E_W has an endpoint in O
- let M be a maximal matching of G_W
  every $e \in M \subseteq E_W$ has an endpoint in O so $|O| \leq 2|M|$
- *mark* the endpoints V_m of $S \cap E_W$: $|S \cap E_W| \geq |V_m|/2$ (since every edge is counted twice by the vertices)
- for any $(i, j) \in M$, either $(i, j) \in S$ or $\exists k\ s.t. (i, k), (k, j) \in S$, since every pair of vertices are at most 2 hops away in G_S.
- further, either $(i, k) \in E_W$ or $(k, j) \in E_W$ since (i, j) is unsorted

proof of claim that $\geq \frac{\epsilon}{4}n$ edges of S would fail

Let $G_W = (A, E_w)$, E_w are *unsorted* edges
we need to show $|S \cap E_W| \geq \epsilon n/4$

- let O be the set of out of place elements
- every edge in E_W has an endpoint in O
- let M be a maximal matching of G_W
  every $e \in M \subseteq E_W$ has an endpoint in O so $|O| \leq 2|M|$
- *mark* the endpoints V_m of $S \cap E_W$: $|S \cap E_W| \geq |V_m|/2$ (since every edge is counted twice by the vertices)
- for any $(i,j) \in M$, either $(i,j) \in S$ or $\exists k\ s.t.(i,k),(k,j) \in S$, since every pair of vertices are at most 2 hops away in G_S.
- further, either $(i,k) \in E_W$ or $(k,j) \in E_W$ since (i,j) is unsorted
- so either i or j or both are marked: $|V_m| \geq |M|$ (recall M is a matching)

proof of claim that $\geq \frac{\epsilon}{4}n$ edges of S would fail

Let $G_W = (A, E_w)$, E_w are *unsorted* edges
we need to show $|S \cap E_W| \geq \epsilon n/4$

- let O be the set of out of place elements
- every edge in E_W has an endpoint in O
- let M be a maximal matching of G_W
  every $e \in M \subseteq E_W$ has an endpoint in O so $|O| \leq 2|M|$
- *mark* the endpoints V_m of $S \cap E_W$: $|S \cap E_W| \geq |V_m|/2$ (since every edge is counted twice by the vertices)
- for any $(i, j) \in M$, either $(i, j) \in S$ or $\exists k\ s.t. (i, k), (k, j) \in S$, since every pair of vertices are at most 2 hops away in G_S.
- further, either $(i, k) \in E_W$ or $(k, j) \in E_W$ since (i, j) is unsorted
- so either i or j or both are marked: $|V_m| \geq |M|$ (recall M is a matching)

proof of claim that $\geq \frac{\epsilon}{4} n$ edges of S would fail

Let $G_W = (A, E_w)$, E_w are *unsorted* edges
we need to show $|S \cap E_W| \geq \epsilon n/4$

- let O be the set of out of place elements
- every edge in E_W has an endpoint in O
- let M be a maximal matching of G_W
  every $e \in M \subseteq E_W$ has an endpoint in O so $|O| \leq 2|M|$
- *mark* the endpoints V_m of $S \cap E_W$: $|S \cap E_W| \geq |V_m|/2$ (since every edge is counted twice by the vertices)
- for any $(i, j) \in M$, either $(i, j) \in S$ or $\exists k \; s.t. (i, k), (k, j) \in S$, since every pair of vertices are at most 2 hops away in G_S.
- further, either $(i, k) \in E_W$ or $(k, j) \in E_W$ since (i, j) is unsorted
- so either i or j or both are marked: $|V_m| \geq |M|$ (recall M is a matching)

putting it together:

$$|S \cap E_W| \geq |V_m|/2 \geq |M|/2 \geq |O|/4 \geq \epsilon n/4$$

Matroids

motivation: an abstract mathematical object that will allow us to show that many greedy algorithms are optimal

use: if you can show that your problem can be caste as a matroid (problem), then you get an optimal, greedy algorithm for free!

An example: Kruskal's algorithm for *maximum* weight spanning tree

set $T = \emptyset$
while $\exists e \notin T$ s.t. $T \cup \{e\}$ is a forest
    choose such an e with maximum weight
    replace T by $T \cup \{e\}$

An example: Kruskal's algorithm for *maximum* weight spanning tree

set $T = \emptyset$
while $\exists e \notin T$ s.t. $T \cup \{e\}$ is a forest
    choose such an e with maximum weight
    replace T by $T \cup \{e\}$

Let $\mathcal{I} = \{\mathcal{J} \subseteq \mathcal{E} : \mathcal{J} \text{ is a forest}\}$ (i.e. the set of all forests).

An example: Kruskal's algorithm for *maximum* weight spanning tree

set $T = \emptyset$
while $\exists e \notin T$ s.t. $T \cup \{e\}$ is a forest
    choose such an e with maximum weight
    replace T by $T \cup \{e\}$

Let $\mathcal{I} = \{\mathcal{J} \subseteq \mathcal{E} : \mathcal{J}$ is a forest$\}$ (i.e. the set of all forests).

set $J = \emptyset$
while $\exists e \notin J$ s.t. $J \cup \{e\} \in \mathcal{I}$
    choose such an e with maximum weight
    replace J by $J \cup \{e\}$

An example: Kruskal's algorithm for *maximum* weight spanning tree

set $T = \emptyset$
while $\exists e \notin T$ s.t. $T \cup \{e\}$ is a forest
    choose such an e with maximum weight
    replace T by $T \cup \{e\}$

Let $\mathcal{I} = \{\mathcal{J} \subseteq \mathcal{E} : \mathcal{J}$ is a forest$\}$ (i.e. the set of all forests).

set $J = \emptyset$
while $\exists e \notin J$ s.t. $J \cup \{e\} \in \mathcal{I}$
    choose such an e with maximum weight
    replace J by $J \cup \{e\}$

$\mathcal{I}$ is the **graphic matroid**: it is a family of subsets of E with some other properties that guarantee the above greedy algorithm is correct/optimal.

For what families $\mathcal{I}$ does this prototypical "greedy" algorithm work?

set $J = \emptyset$
while $\exists e \notin J$ s.t. $J \cup \{e\} \in \mathcal{I}$
    choose such an e with maximum weight
    replace J by $J \cup \{e\}$

For what families $\mathcal{I}$ does this prototypical "greedy" algorithm work?

set $J = \emptyset$
while $\exists e \notin J$ s.t. $J \cup \{e\} \in \mathcal{I}$
  choose such an e with maximum weight
  replace J by $J \cup \{e\}$

matchings? let $\mathcal{I}$ be the set of all matchings in a graph
the greedy algorithm fails to find the max-weight matching (e.g. cycle with edge weights 7,3,8,9)

Matroid: definition

for a *ground set* S and *independent set* family $\mathcal{I}$ of subsets of S, $M = (S, \mathcal{I})$ is a *matroid* if:

- **non-empty** $\emptyset \in \mathcal{I}$
- **heredity** if $J \in \mathcal{I}$ and $J' \subseteq J$, then $J' \in \mathcal{I}$
- **exchange** if $J, J' \in \mathcal{I}$ and $|J| < |J'|$ then there is an element $e \in J'$ such that $J \cup \{e\} \in \mathcal{I}$

Matroid: definition

for a *ground set* S and *independent set* family $\mathcal{I}$ of subsets of S, $M = (S, \mathcal{I})$ is a *matroid* if:

- non-empty: $\emptyset \in \mathcal{I}$
- heredity: if $J \in \mathcal{I}$ and $J' \subseteq J$, then $J' \in \mathcal{I}$
- exchange: if $J, J' \in \mathcal{I}$ and $|J| < |J'|$ then there is an element $e \in J'$ such that $J \cup \{e\} \in \mathcal{I}$

matchings do not satisfy exchange (e.g. odd-length alternating path example)

Matroid: definition

for a *ground set* S and *independent set* family $\mathcal{I}$ of subsets of S, $M = (S, \mathcal{I})$ is a *matroid* if:

- **non-empty** $\emptyset \in \mathcal{I}$
- **heredity** if $J \in \mathcal{I}$ and $J' \subseteq J$, then $J' \in \mathcal{I}$
- **exchange** if $J, J' \in \mathcal{I}$ and $|J| < |J'|$ then there is an element $e \in J'$ such that $J \cup \{e\} \in \mathcal{I}$

Matroid: definition

for a *ground set* S and *independent set* family $\mathcal{I}$ of subsets of S, $M = (S, \mathcal{I})$ is a *matroid* if:

- **non-empty** $\emptyset \in \mathcal{I}$
- **heredity** if $J \in \mathcal{I}$ and $J' \subseteq J$, then $J' \in \mathcal{I}$
- **exchange** if $J, J' \in \mathcal{I}$ and $|J| < |J'|$ then there is an element $e \in J'$ such that $J \cup \{e\} \in \mathcal{I}$

a *basis* is any independent set that is not a strict subset of any other independent subset a.k.a. a maximal set (e.g. spanning trees)

Matroid: definition

for a *ground set* S and *independent set* family $\mathcal{I}$ of subsets of S, $M = (S, \mathcal{I})$ is a *matroid* if:

- non-empty: $\emptyset \in \mathcal{I}$
- heredity: if $J \in \mathcal{I}$ and $J' \subseteq J$, then $J' \in \mathcal{I}$
- exchange: if $J, J' \in \mathcal{I}$ and $|J| < |J'|$ then there is an element $e \in J'$ such that $J \cup \{e\} \in \mathcal{I}$

a *basis* is any independent set that is not a strict subset of any other independent subset a.k.a. a maximal set (e.g. spanning trees)

theorem: the greedy algorithm finds a maximum-weight basis.

set $J = \emptyset$
while $\exists e \notin J$ s.t. $J \cup \{e\} \in \mathcal{I}$
    choose such an e with maximum weight
    replace J by $J \cup \{e\}$

Proof that the greedy algorithm finds a max-weight basis

let $G = \{g_1, g_2, \ldots, g_k\}$ be the solution found by greedy, with elements given in the order they were found. G is a basis, by the greedy nature of the algorithm

Proof that the greedy algorithm finds a max-weight basis

let $G = \{g_1, g_2, \ldots, g_k\}$ be the solution found by greedy, with elements given in the order they were found. G is a basis, by the greedy nature of the algorithm

we prove G is optimal by a greedy stays ahead argument

Proof that the greedy algorithm finds a max-weight basis

let $G = \{g_1, g_2, \ldots, g_k\}$ be the solution found by greedy, with elements given in the order they were found. G is a basis, by the greedy nature of the algorithm

we prove G is optimal by a greedy stays ahead argument

let $O = \{o_1, o_2, \ldots, o_\ell\}$ be an optimal solution with elements ordered by decreasing weight; suppose that $w(G) < w(O)$

Proof that the greedy algorithm finds a max-weight basis

let $G = \{g_1, g_2, \ldots, g_k\}$ be the solution found by greedy, with elements given in the order they were found. G is a basis, by the greedy nature of the algorithm

we prove G is optimal by a greedy stays ahead argument

let $O = \{o_1, o_2, \ldots, o_\ell\}$ be an optimal solution with elements ordered by decreasing weight; suppose that $w(G) < w(O)$

by the exchange property of matroids and the greediness of the algorithm, $k = \ell$.

Proof that the greedy algorithm finds a max-weight basis

let $G = \{g_1, g_2, \dots, g_k\}$ be the solution found by greedy, with elements given in the order they were found. G is a basis, by the greedy nature of the algorithm

we prove G is optimal by a greedy stays ahead argument

let $O = \{o_1, o_2, \dots, o_\ell\}$ be an optimal solution with elements ordered by decreasing weight; suppose that $w(G) < w(O)$

by the exchange property of matroids and the greediness of the algorithm, $k = \ell$.

let i be the first index such that $w(g_i) < w(o_i)$. by the exchange property of matroids, there is an element $e \in \{o_1, o_2, \dots, o_i\}$ that can be added (while maintaining independence) to $\{g_1, g_2, \dots, g_{i-1}\}$. by the ordering of the elements, we have that $w(e) \geq w(o_i) > w(g_i)$. but e would then contradict the choice of g_i. $\square$

disjoint path matroid

Let $G = (V, E)$ be an arbitrary directed graph, and let s be a fixed vertex. A subset $I \subseteq V$ is independent if and only if there are edge-disjoint paths from s to each vertex in I.

solves
: Given a directed graph with a special vertex s, find the largest set of edge-disjoint paths from s to other vertices.