MST = minimum spanning tree

Given an undirected $G=(V,E)$
with positive edge weights $w(e)>0$

Find the min weight connected subgraph,

For $S \subset E$, $w(S) = \sum_{e \in S} w(e)$

Min weight connected subgraph is a tree.
Hence, MST problem.

Basic graph theory:

1) Tree on n vertices has $n-1$ edges.
2) Any connected $G=(V,E)$ where $|E|=|V|-1$ is a tree
3) In a tree, there is exactly one path between every pair of vertices.

Forest = acyclic graph.
A tree is an example of a forest.
But also a disconnected set of trees is a forest.

First we'll look at a structural lemma then we'll use the lemma to get Prim's & Kruskal's MST algorithms.

Cut Property:

Consider undirected $G=(V,E)$.

Fix $X \subset E$ where $X \subset T$ for a MST T.

(so X is part of a MST)

Take a subset $S \subset V$ where no edge of X crosses $S \Leftrightarrow \bar{S}$

(so $(S,\bar{S})$ is a cut of X)

Look at <u>all</u> edges <u>of G</u> that cross $S \Leftrightarrow \bar{S}$.

Let e^* be the edge of min weight that cross $S \Leftrightarrow \bar{S}$.

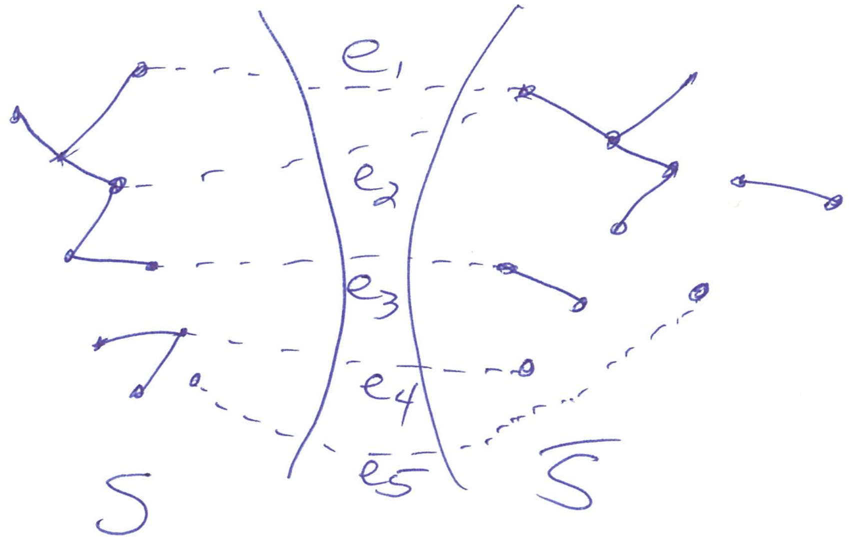
Then:

$X \cup e^* \subset T'$ where T' is a MST

(so $X+e^*$ is part of a MST).

③

Example: $X \subset E$ marked by solid lines



$e_1, \ldots, e_5$ are edges of G crossing $S \Leftrightarrow \bar{S}$

Suppose e_4 is min weight of these.

Then $X \cup e_4$ is part of a MST T')

(assuming X was part of some MST T)

How do we use it?

Idea of Dijkstra's algorithm:

R = explored vertices.

During the course of the algorithm,

for $w \in R$, $dist(w)$ = length of shortest path from s to w

for $w \notin R$, $dist(w)$ = length of shortest s to w path only using vertices in R

Next vertex to explore: $w \notin R$ with min $dist(w)$.

Prim's MST algorithm:

$S = R =$ explored vertices

Have X which is part of a MST
& X connects S.

~~For w~~

Add edge $e^* = (v, w)$ where $v \in S$, $w \in \bar{S}$
of min weight

The cut property says that
$X \cup e^*$ is part of a MST.

Then we add w to S and repeat.

To find e^* quickly, we maintain:

for $w \notin S$:

$\text{cost}(w)$ = weight of min weight edge from w to some $v \in S$.

$\text{prev}(w)$ = this neighbor v

To find e^*, look for w with min $\text{cost}(w)$

& then add $(w, \text{prev}(w))$ to X.

Prim(G,w):

input: undirected, connected $G=(V,E)$ with edge weights $w(e)>0$ for all $e\in E$

output: MST defined by array prev(w).

```
for all y∈V, set [ cost(y) = ∞
                 [ prev(y) = NULL
Choose a start vertex s.
Set cost(s) = 0.
H = ∅
for all y∈V, Insert(H, y, cost(y))
While H ≠ ∅:
      y = DeleteMin(H)
      for all (y,z) ∈ E
            if cost(z) > w(y,z)
            then [ cost(z) = w(y,z)
                 [ prev(z) = y
                 [ DecreaseKey(H, z, cost(z))
```

Running time: Same as Dijkstra's $O((n+m)\log n)$.

Proof of cut property:

Fix G, X, S, T, & e^* satisfying the hypotheses.

We know $X \subset T$ & T is a MST.

no edge of X crosses $S \longleftrightarrow \bar{S}$.

for all $e = (y,z) \in E$ if $y \in S$, $z \in \bar{S}$

then $w(e^*) \leq w(e)$

We need to show a tree T'

where T' is a MST

& $X \cup e^* \subset T'$.

To show that T' is a MST, we'll show:

1) T' is a tree

& 2) $w(T') \leq w(T)$

2 cases:

$e^* \in T$ or $e^* \notin T$.

if $e^* \in T$ then we know

$X \subset T$ & $e^* \in T$

thus $X \cup e^* \subset T$

& we know T is a MST

So we're done.

Suppose $e^* \notin T$.

Let $e^* = (a,b)$. Say $a \in S$ & $b \in \bar{S}$.

Add e^* to T.

It will have a cycle.

There is 1 path between a & b in T.

Let P denote this path.

Then $P \cup e$ is a cycle in $T \cup e^*$

Since $a \in S$ & $b \in \bar{S}$, then

P must cross $S \Leftrightarrow \bar{S}$

at least once (maybe more).

Take one edge e' of P that crosses $S \Leftrightarrow \bar{S}$.

(If more than one choose any.)

Set $T' = T \cup e^* - e'$.

<u>Claim:</u> (i) $X \cup e^* \subset T'$

(ii) T' is a tree

(iii) $w(T') \leq w(T)$

This proves the cut property lemma.

Just need to prove the claim.

We know $X \subset T$ so $X \cup e^* \subset T \cup e^*$

What about e'?

We know no edge of X crosses $S \leftrightarrow \bar{S}$.

So $e' \notin X$.

Thus, $X \cup e^* \subset \underbrace{T \cup e^* - e'}_{\text{this is } T'!}$ That proves (i)

For (iii):

We assumed e^* is the min weight edge across $S \leftrightarrow \bar{S}$

Since e' ~~is~~ crosses $S \leftrightarrow \bar{S}$

then $w(e^*) \leq w(e')$

$$w(T') = w(T) + w(e^*) - w(e')$$

$$\leq w(T) \quad \text{Since } w(e^*) - w(e') \leq 0$$

For (ii):

Recall if T' is connected & has $n-1$ edges then it's a tree.

We know it has $n-1$ edges.

To see it's connected, consider a pair of vertices y, z.

y & z are connected in T by a path P'.

If P' does not contain e' then P' also appears in T'.

If P' contains e' then

Consider $P \cup e^* - e'$ call this $\mathcal{E}$.

If $e' = (c, d)$ then $\mathcal{E}$ is a path between c & d.

So we can replace edge e' in P' by the path $\mathcal{E}$ and this gives a new path between y & z.

This new path appears in T'.

Thus y & z are connected in T'.

$\blacksquare$