Unique coverage and the interference problem:

The Unique coverage problem: given a universe U of n elements and given a collection S of subsets of U, we want to find a sub-collection $S' \subseteq S$ which maximizes the number of elements that are uniquely covered, i.e. appear in exactly one set S'.

The Budgeted Unique Coverage Problem: give profits for elements and costs for sub sets, given also a budget B, find a sub-collection $S' \subseteq S$ whose total cost is at most B and maximizes the total profit of elements that are uniquely covered.

The budgeted case is a special case of the coverage problem, motivated by an application at Bell-labs, in which we want to find a set of base stations and options within the total budget which maximizes the total satisfaction weighted by client densities (Satisfaction S_k for covering a region by k base stations where $S_1 \geq S_2 \geq \dots \geq 0$ where for budgeted unique coverage $S_1 = 1$ and $S_2 = S_3 = \dots = 0$).

→ Though the problem has the same flavor of maximum coverage it is much harder. The problem generalizes Max-Cut and has very close connections to the radio broadcast problem, which is studied extensively.

For simplicity we focus of the Unique coverge only though the generalization are not much harder.

The problem has application for envy-free pricing an important problem in computational economics.

Simple $O(\log n)$ algorithm:

- Partition the elements into $\log n$ classes according to their degrees i.e. the number of sets that cover an element
- Let i be the class of maximum cardinality
- Choose a set in S to be in S' with prob $\frac{1}{2^i}$

Pf: Fix an element x in group i and suppose it has been covered d times in S, where $2^i \leq d \leq 2^{i+1}-1$.

<u>We claim</u>: we uniquely cover $\frac{1}{e^2}$ fraction of elements of class i in expectation

The probability that x is covered exactly once is $\left(\frac{d}{2^i}\right)\left(1-\frac{1}{2^i}\right)^{d-1}$

(the factor d since we have d choices of which set cover x, a $\frac{1}{2^i}$ probability that this set is kept and a $1-\frac{1}{2^i}$ probability that each of the $d-1$ set is discarded by our bound on d, the probability is at least $\left(\frac{2^i}{2^i}\right)\left(1-\frac{1}{2^i}\right)^{2^{i+1}} \geq \frac{1}{e^2}$.

The expected total profit of elements covered exactly once is thus $\frac{1}{e^2} \times \frac{n}{\log n}$ which is $\frac{1}{e^2 \log n}$ opt as desired.

We can get the same way for the budgeted case $O(\log n)$ approx algorithm

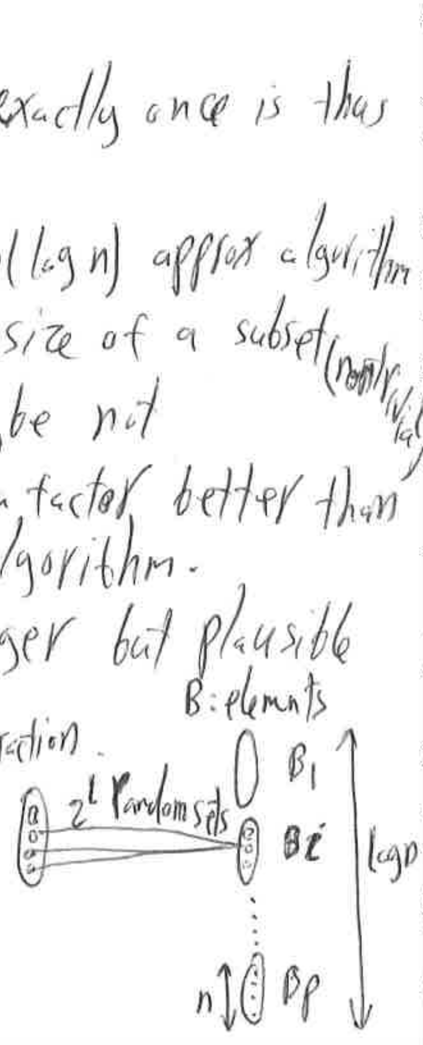
We can also get $O(\log B)$ where B is the maximum size of a subset (nontrivial)

The algorithm seems naive. Can we do better? maybe not

<u>Thm</u>: This problem is hard to approximate within a factor better than $O(\log^c n)$, $0<c<1$ unless NP has a sub-exponential algorithm.

$O(\log^{\frac{1}{3}} n)$ hard or even $O(\log n)$ hard under stronger but plausible complexity assumptions.

<u>Pf idea</u>: A bad instance that we cannot uniquely cover $\frac{1}{\log n}$ fraction.



In the graph, at most $O(n)$ elements of B can be uniquely covered by sets of A (according to $|A|$)

You can turn the example into a real <u>hardness</u>