Oct & Nov:
- Spend some time on ILP algorithms, related optimization.
- Weekly homeworks (light) to illustrate.
- We’ll start to have presentations on proposals.
- Long presentations in November.

= 

Today! Bin packing, 2 bins:

\( n \) items, sizes \( a_1, \ldots, a_n > 0 \)

Say \( a_1 + \ldots + a_n = 2 \) ideally we could split
\( a_1, \ldots, a_n \) into two groups, each adding to one.

Problem: ILP relax to LP give very bad bounds...

Problem! Bin packing, or partition (2 bins) is NP-complete.
- Today! Bin Packing:
  W Fernandez de la Vega & G.S. Lueker. "Bin packing can be solved within 1+ε in linear time." Combinatorica (1) 349-355, 1981.

- Upshot: You can have Gurobi (or some other optimizer) give an approximate bin packing solution. You could even program this by hand without any LP solver.

==
We have a_1, ..., a_n, sum = 2 want divide into two sets of roughly equal size.

We had a wonderful illustration with chalk...
First: \(a_1 \geq a_2 \geq \ldots \geq a_n > 0\)

Imagine that we want to get within 10% of optimal. Say

\[
a_1 \geq \ldots \geq a_k \geq 0.1
\]

\[
a_{k+1}, \ldots, a_n < 0.1
\]

1. Look at all ways of splitting up \(a_1, \ldots, a_k\)

   (in practice, you'd Branch and Bound over \(a_1, a_2, \ldots\) in that order)

   Remark: since each \(a_i\) \(a_1, \ldots, a_k\) is \(\geq 0.1\),

   \(k \leq 20\) (since \(a_1 + \ldots + a_k \leq 2\))

   Worst case: for each \(M \subset \{1, \ldots, k\}\)

   \[
   \max \left( \sum_{m \in M} a_m, \sum_{m \notin M} a_m \right)
   \]

   \[
   \text{so } M \subset \{1, \ldots, k\}, \text{ at largest } \{1, \ldots, 20\}
   \]

   \# M's \(\leq 2^{20}\)
Two cases:

If $a_1, \ldots, a_k$ have best partition
\[
\{1, \ldots, k\} \text{ into } m, m',
\]
\[
\min_{m, m'} \max_{m} \left( \sum_{a_m} \right) = \{1.315\} \text{ or } \geq 1
\]

i.e. dividing some (the large elements) can't be done with one bin at least 1.315

Then add all $a_{k+1}, \ldots, a_n$ to smaller part and get
\[
1.315 \text{ versus } 2 - 1.315 = 0.685.
\]

Then you've found the best packing --

If
\[
\min_{m, m'} \max_{m} \left( \sum_{a_m} \right) < 1
\]

Then back and forth with $a_{k+1}, \ldots, a_n$ and get with a difference of 0.1: worst 0.95 one bin, 1.05 other bin
Best possible: 1 in one bin, 1 in other bin, so now within 0.05 of