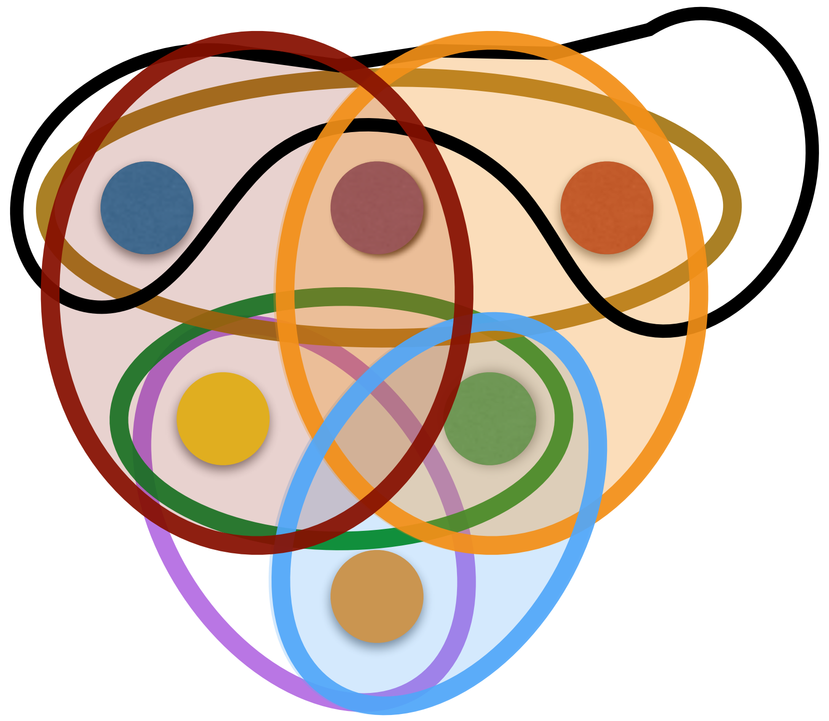
Set cover, linear programming and randomized rounding



Result so far

Iterated randomized rounding gives
collection of sets
that is a set cover with
probability 95%
and with average
cost at most (ln(n)+3) OPT.

Not guaranteed!

Not guaranteed!

Q: What if you want the output to always be a set cover?

Desired result:

algorithm that gives collection of sets that is a set cover and with average cost at most O(ln(n)) OPT.

Guaranteed!

Obvious solution:

Replace
“repeat $\ln(n)+3$ times”
by
“repeat until you have a set cover.”

Equivalent algorithms

```
Repeat
 For each S
    Put S in cover w.pr. x(S)
    (if not there yet)
Until you have a set cover
```

```
Repeat
  Choose S w.pr. $x_S / \sum_{S'} x(S')$
   Put S in cover
   (if not there yet)
Until you have a set cover
```

Sample-and-iterate algorithm

```
Repeat
  Choose S w.pr. $x_S / \sum_{S'x(S')}$
   Put S in cover
   (if not there yet)
Until you have a set cover
```

Set cover, linear programming and randomized rounding