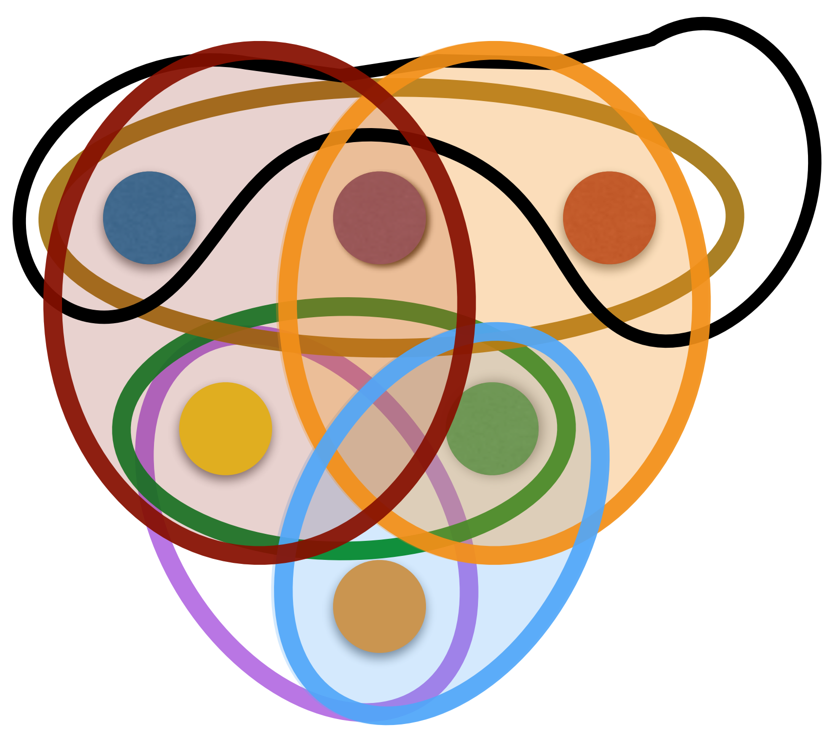
Set cover, linear programming and randomized rounding



Result

The sample-and-iterate algorithm gives a collection of sets that **is** a set cover with **average** cost at most $(1+\ln(n))$ OPT.

A more efficient algorithm

Linear programming takes
polynomial time
but
is often slower than
combinatorial algorithms

Greedy

Repeat
  Choose S maximizing $\#(\text{new elts covered})/c_S$
   Put S in cover
Until you have a set cover

Result:

Greedy also gives
a collection of sets
that **is** a set cover
and with
cost at most $(1+\ln(n))$ OPT.

Can we do better?

No:
It is NP-hard to obtain (in polynomial time) a better-than-ln(n) approximation for set cover

Uri
Feige

Vašek
Chvátal

László
Lovász

David
Johnson

Neal
Young

What have we learned?

- Famous problem: set cover
- Concept: Randomization
- Algorithmic technique: Randomized rounding
- Analysis tool: Linearity of expectation
- Laying out an analysis: slow & steady, orderly – like hiking up a mountain

Set cover, linear programming and randomized rounding