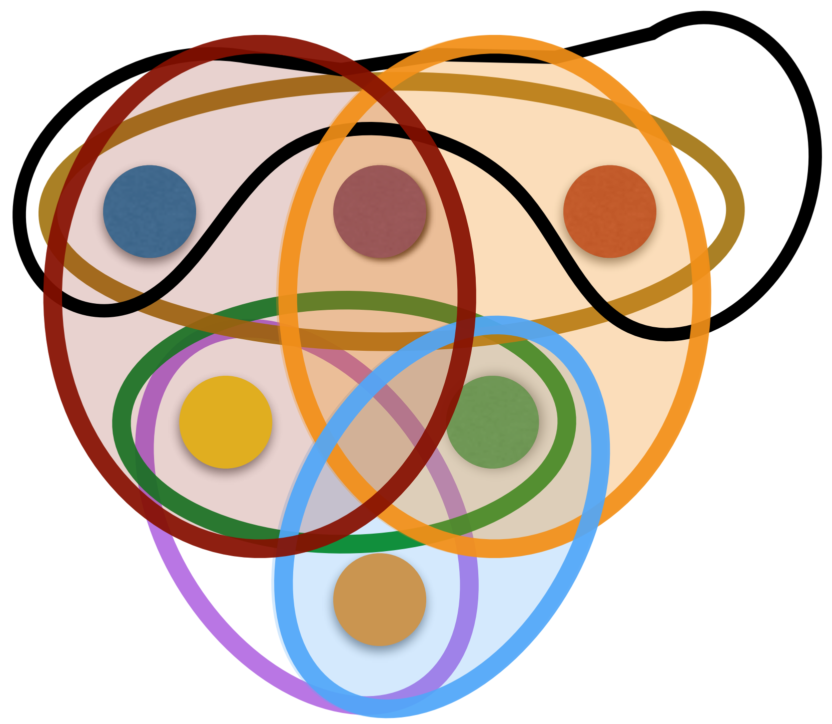
Set cover, linear programming and randomized rounding



Getting a set cover

Idea: repeat!

Randomized rounding algorithm

n=#elements

Repeat ln(n)+3 times

  For each S

    Put S in cover w.pr. x(S)

    (if not there already)

Note: $e^3 = 20.0\ldots$

Cost

In expectation:
at most
$(\ln(n) + 3)OPT$

Correctness

$$\mathbf{Pr[cover] = 1 - Pr[not\ cover]}$$

$$\mathbf{Pr[not\ cover] =}$$
$$\mathrm{Pr}[\exists\ \mathbf{element\ not\ covered}] \leq$$
$$\sum_{\mathbf{e}} \mathrm{Pr}[\mathbf{e\ not\ covered}]$$

For one element e
and for one iteration

$$\mathbf{Pr}[\mathbf{e}\ \text{not covered}] < \mathbf{1/e}$$

For one element e
and for all iterations together

$$\mathbf{Pr[e\ not\ covered]} < \mathbf{(1/e)^{\ln(n)+3}} = \frac{\mathbf{1}}{\mathbf{e^3 n}}$$

$$\sum_{\mathbf{e}} \mathbf{Pr[e\ not\ covered]} < \mathbf{n\frac{1}{e^3 n} = \frac{1}{e^3} < 0.05}$$

So:

$$\mathbf{Pr[cover] > 0.95}$$

Result

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.

Set cover, linear programming and randomized rounding