600.432/632 Declarative Methods - Prof. Jason Eisner
Answers to "ILP Solvers" discussion problems

1. (a) The original problem has 3 variables and 3 constraints. But one of them is an equality constraint.

   Answer #1: First convert to canonical form by replacing == with a pair of <= and >= constraints. This gives 3 variables and 4 constraints. Now to get to standard form, introduce a slack variable for each constraint to turn it into an equality constraint. This gives a total of 7 variables and 4 constraints.

   Answer #2: Convert directly to standard form, without going through canonical form. We introduce a slack variable for each of the 2 inequality constraints to turn it into an equality constraint. This gives a total of 5 variables and 3 constraints.

   (b) Let's rewrite carefully in the form max cx subto Ax <= b, laying out the matrix in rows and columns:

```
var p,q,r >= 0;
maximize wealth: p   +     q + r;
subto c1:        p               <= 10
subto c2:     -3*p   + 4  *q     <= 0
subto c3:        p   - 0.5*q - r <= 0
subto c4:       -p   + 0.5*q + r <= 0
```

   Now we can see that the dual will be

```
var c1, c2, c3, c4 >= 0;
minimize dual: 10*c1;
subto p:          c1 - 3*c2 +     c3 -     c4 >= 1;
subto q:               4*c2 - 0.5*c3 + 0.5*c4 >= 1;
subto r:                         -c3 +     c4 >= 1;
```

   Alternatively, one can translate the equality constraint directly into an unbounded variable. This eliminates c4 from the above but allows c3 to become negative.

   (c) Constraint c2 must be tight at this solution (3p=4q). The dual variable had to become positive in order to enforce it there. If we relaxed this constraint, the primal objective would be able to increase, at a rate given by the value of the dual variable.

   If the dual variable were 0, we can't conclude anything. (Except that constraint c2 is not being actively enforced by the solver: if we relaxed it, the primal objective would not increase. This could happen either because constraint c2 is slack, or because it is coincidentally tight -- i.e., the solver is actively enforcing m other constraints, whose hyperplanes define a vertex, and the hyperplane for constraint c2 happens to also pass through that vertex.)

2. [Oops, this program was not actually feasible, but let's pretend it is!]

   (a) It will still have 4 constraints, but 6 variables. We'll need 3 new variables to take up slack in the inequalities and turn them into equalities.

   (b) There are 4 constraints (4 rows of the A matrix), so AT MOST 4 variables can be > 0 (the basic variables in a given basic feasible solution).

Here's another way to see it. The feasible polytope (for the original problem with inequalities) is 3-dimensional, with variables x,y,z. Thus, identifying a vertex of the polytope will require 3 tight constraints. Now, one of those constraints will certainly be equality constraint, which is necessarily tight in a feasible solution -- in fact, that constraint reduces the polytope to just 2 dimensions! That constraint doesn't have a variable attached. We also need 2 of the 6 other variables to be 0, which says that there's no slack in 2 of the other inequalities (e.g., perhaps x >= 0 is tight, making x=0, or perhaps 7x+5y <= 5 is tight, making the associated slack variable = 0). This leaves at most 4 variables that could be > 0.

(c) No. This refers to solutions that are not at a vertex. Such a solution is a convex combination (weighted average) of two or more optimal vertices. Each vertex has a different choice of 4 basic variables that can be nonzero. But then their average will generally have more nonzeros. For example, (1,1,1,1,0,0) and (0,0,1,1,1,1) have 4 nonzeros each, but their average has 6 nonzeros.

(d) Subtracting constraint 2 from constraint 1, we get 6x + 3y <= 4, and therefore 2x-y <= 4/3. At any solution where x and y are integers, therefore, 2x-y <= 1.

(e) max M <= max S <= max P.

(f) No. Yes, the relaxed problem will be cut back eventually to have only integer vertices. But in this case, that's not because the cost matrix is totally unimodular.

Remember the definition of total unimodularity. If the original cost matrix wasn't totally unimodular, it must have had a submatrix that wasn't integer or whose determinant was not -1, 0, or +1. But adding new rows to the cost matrix (new cutting plane constraints) will certainly not eliminate that bad submatrix!

(g) Solving the constraints for z, we have lower bounds on z

```
   L1: (1/500)cost - (1/5)x + (4/5)y <= z
   L2:                        -y + 5 <= z
   L3:                             0 <= z
```

and upper bounds on z

```
   U1:                                  z <= (1/500)cost - (1/5)x + (4/5)y
   U2:                                  z <= -2x + 7
```

Joining these constraints to eliminate z, we get new linear constraints saying that all the lower bounds are <= all the upper bounds. These hold of an assignment to the other variables iff it can be extended into a solution to z that satisfies L1,L2,L3,U1,U2.

```
   L1+U1:  (trivial constraint)
   L1+U2: (1/500)cost - (1/5)x + (4/5)y <= -2x + 7
   L2+U1: -y + 5 <= (1/500)cost - (1/5)x + (4/5)y
   L2+U2: -y + 5 <= -2x + 7
   L3+U1: 0 <= (1/500)cost - (1/5)x + (4/5)y
   L3+U2: 0 <= -2x + 7
```

(h) It may help to write out the original in a form that makes the matrix clearer:

```
      max 100x - 400y + 500z
    subto   7x +   5y +   0z <= 5
            1x +   2y +   0z  = 1
            0x +   1y +   1z >= 5
            2x +   0y +   1z <= 7
            x,y,z >= 0
```

The new problem is

```
      min 5a + 1b + 5c + 7d  (coefficients taken from the "b" column)
    subto 7a + 1b + 0c + 2d >=  100  (constraint taken from "x" coeffs)
          5a + 2b + 1c + 0d >= -400  (constraint taken from "y" coeffs)
          0a + 0b + 1c + 1d >=  500  (constraint taken from "z" coeffs)

          a >= 0  (since first constraint row of primal is <=)
          b unconstrained  (since second constraint row is =)
          c <= 0  (since third constraint row of primal is >=)
          d >= 0  (since fourth constraint row of primal is <=)
```

(i) The first dual constraint (associated with primal var x) is tight: 7a+b+2d = 100.

(j) The dual var a (associated with the first primal constraint) is 0.

(k) You can't conclude anything. You still have a >= 0, but it's not required to be > 0.

(l) The primal optimum is <= k.

4. (a) Suppose that distributing $1000 is not optimal, so there is a solution that distributes only $900 (for example) that is better than any way to distribute $1000. That implies that giving the extra $100 to any of the clubs would actually hurt society. So none of the functions f_i are monotonic -- for every club, there is some amount of funding for which it hurts to give them more.

Some of you thought that some of the f_i must be negative, but that is not necessary. The point is that ADDITIONAL funding would reduce the social benefit, but the social benefit might still be positive.

(b) This is a question about how, as x increases, the derivative of f_i(x) changes:

```
convex: derivative only ever increases (function looks like u)
concave: derivative only ever decreases (function looks like n)
both: derivative remains constant (function is linear)
        (That is the only way for the whole function
        to be convex and for the whole function also
        to be concave!)
neither: derivative sometimes increases and sometimes decreases
```

It's useful to think of the derivative as "How much would it help to have another $1?" Different answers were okay if appropriately explained.

"Concave" is common for resource allocation problems. If you give someone an extra $1, it doesn't help as much if they're already rich. They're already buying the most important things, and the extra money will now go to less important things.

"Neither" is also a good answer, because an extra $1 may also not help if they're really poor -- maybe they need at least $100 before they can get anything useful done. Then an extra $1 is most helpful for clubs that have $99 already and need that last dollar.

(c) maximize sum_i f_i(x_i) - L1*(variance(x) - 100)
                            - L2*(sum_i x_i - 1000)
    subject to L1, L2 <= 0.

(d) If the answer to (b) is "concave," then the maximization problem will be concave and can therefore be solved by hill-climbing (gradient ascent).

That's because the Lagrangian objective is a sum of concave functions: namely,
* the concave functions f_i,
* a negative multiple of the variance function
  (which the start of the problem mentions is convex),
* the function L2*(sum_i x_i - 1000), which since it's linear
  is both concave and convex.

You may find it surprising that the variance constraint doesn't make optimization harder. It actually makes it easier! If L2 is large enough, then the problem will mainly just try to minimize variance, and there's a single solution to that -- distribute all the money equally.

If the answer to 11b. is "neither," then all bets are off and the optimization problem is likely to be difficult. (You may be able to use integer switching variables to split it into a bunch of concave problems, but then you have to do some kind of backtracking search over the integer variables.)