# CSCI 5654 (Linear Programming, Fall 2013)

Lecture-1

August 26, 2013

# CSCI 5654: Linear Programming

Instructor: Sriram Sankaranarayanan.

Meeting times: Tuesday-Thursday, 12:30-1:45 p.m.
ECCS 1B12 (CAETE Classroom).

Office Hours: After class today and thursday.
Semester office hours TBA.

Web page:

http://www.cs.colorado.edu/~srirams/classes/doku.php/linear_programming_fall_2013.

# Today's Lecture

1. Course information, some ground rules.
2. Introduce Linear Programming.
3. Motivation: expressing some example problems as linear programs.

# Book

Main Book: Linear Programming by Vasek Chvátal.

On reserve at Engineering library soon.

Bookstore: out of print.

Numerous copies available on `amazon.com` and other places.

I will hand out notes and photocopies for first 3-4 weeks.

Alternate: Linear Programming: Foundations and Extensions by R.J. Vanderbei.

Available on-line through Springer.

If you are not able to download, get in touch with me.

Personally prefer Chvátal (but difference is minor).

Course webpage has other references: Dantzig's book is worth reading.

# Other Issues

Office Hours: After class today and thursday @ ECOT 624
Subsequent office hours TBA.

Weekly Assignments: 60% of your grade (70% for CAETE students).

- Collaborations allowed: discuss solution strategies, hints, brainstorming etc.. with your classmates.
- Write down the solutions by yourself. **Do not** collaborate, consult outside sources.
- **Acknowledge** collaborations clearly at the beginning of the assignment.
- Direct questions/doubts to the instructor.

# Other Issues

Exams: Three Quizzes (in class).
(CAETE students need to set up a proctor).

Course Grading Policy: If you want an $A-$ or above:

1. Do not worry about grades: try to enjoy the course!!
2. Attend class regularly or watch lectures.
3. Ask me lots of questions (in class or by email).
4. Averaged $> 85\%$ overall, guaranteed $A-$ or higher grade.

Any questions so far?

# Linear Programming Problem

$$
\begin{array}{rcll}
\text{maximize} & : & c_1x_1 + c_2x_2 + \cdots + c_nx_n + c_0 & \leftarrow \text{Objective} \\
\text{subj.to.} & : & & \downarrow \text{ Constraints} \\
 & & a_{11}x_1 + \cdots + a_{1n}x_n \leq b_1 & \\
 & & \ddots & \\
 & & a_{m1}x_1 + \cdots + a_{mn}x_n \leq b_m &
\end{array}
$$

**Note:** $x_1, \ldots, x_n$ are called the decision variables.

Decision variables are real valued: $(x_1, \ldots, x_n) \in R^n$.

**Q:** Why should we bother spending a semester studying this?

# Linear Programming

- An important optimization problem.
- Maximize (or minimize) a linear function subject to linear constraints.
- Supremely important, applications in many fields:
  - Economics (planning, scheduling, ...)
  - Finance (portfolio management)
  - Control
  - Verification
  - Statistics (machine learning)
  - and more.

# Diet Problem

To live a healthy life, XYZ needs:

- $\geq 200$ grams of carbohydrates.
- $\geq 20$ grams of proteins.
- Other nutrients (ignored for now).

Food sources at grocery store:

| Food | Carb/Unit | Prot/Unit | Cost (dollar/unit) |
|---|---|---|---|
| Carrots (c) | 50 | 3 | 5 |
| Rice (r) | 40 | 1 | 2 |
| Potatoes (p) | 40 | 2 | 1 |
| Quinoa (q) | 40 | 5 | 10 |

**Question:** How can XYZ satisfy her nutritional needs while paying the least amount of money possible?

# Diet Problem as a Linear Program

1. Set up the decision variables:

| Var | Meaning |
|---|---|
| $x_c$ | Quantity of carrots. |
| $x_r$ | Quantity of rice. |
| $x_p$ | Quantity of potatoes. |
| $x_q$ | Quantity of quinoa. |

2. Identify the Objective Function: XYZ needs to pay the least amount of money possible.

$$\text{Money Paid:} \quad \underbrace{5}_{\$/\text{unit Carrot}} * x_c + \underbrace{2}_{\$/\text{unit Rice}} * x_r + 1 * x_p + 10 * x_q$$

3. Identify the constraints.

# Constraints

1. Need to consume at least $200g$ of carbs.

$$C_1: \underbrace{50}_{\text{Carb/unit Carrot}} * x_c + 40 * x_r + 40 * x_p + 40 * x_q \geq 200\,.$$

2. Need to consume at least $20g$ of proteins.

$$C_2:\ 3x_c + x_r + 2x_p + 5x_q \geq 20\,.$$

3. Cannot consume negative quantities of stuff!

$$\begin{aligned} C_3 &: \quad x_c \geq 0 \\ C_4 &: \quad x_r \geq 0 \\ C_5 &: \quad x_p \geq 0 \\ C_6 &: \quad x_q \geq 0 \end{aligned}$$

# AMPL File

```
var xc; # carrot consumption
var xr; # rice consumption
var xp; # potato consumption
var xq; # quinoa consumption

# objective function: cost/unit  of carrots * amt. of carrots consumed +
#                        ... rice + ... potato + .. quinoa.
minimize objVal: 5 * xc + 2 * xr + xp + 10 * xq;


c1: 50 * xc + 40 * xr + 40 * xp + 40 * xq >= 200; # carbohydrate constraint
c2: 3 * xc + xr + 2 * xp + 5 * xq >= 20; # protein constraint
c3: xc >= 0; # pos. amt. constraints
c4: xr >= 0;
c5: xp >= 0;
x6: xq >= 0;

solve; # directive to solve
display xc,xr,xp,xq,objVal; #display values
end;
```

# Solving using GLPK

```
unix prompt# glpsol --math diet.ampl

.... SNIP ...
.... SNIP ...

Constructing initial basis...
Size of triangular part = 2
      0: obj =   0.000000000e+00  infeas =  8.819e+00 (0)
*     2: obj =   3.333333333e+01  infeas =  0.000e+00 (0)
*     3: obj =   1.000000000e+01  infeas =  0.000e+00 (0)
OPTIMAL SOLUTION FOUND
Time used:   0.0 secs
Memory used: 0.1 Mb (115105 bytes)
Display statement at line 20
xc.val = 0
xr.val = 0
xp.val = 10
xq.val = 0
objVal.val = 10
Model has been successfully processed
```

# Diet Problem

To live a healthy life, XYZ needs:

- $\geq 200$ grams of carbohydrates.
- $\geq 20$ grams of proteins.
- $\geq 10$ grams of vitamins.

Food sources at grocery store:

| Food | Carb/Unit | Prot/Unit | Vit./Unit | Cost ($/unit) |
|---|---|---|---|---|
| Carrots (c) | 20 | 3 | 10 | 5 |
| Rice (r) | 40 | 1 | 4 | 2 |
| Potatoes (p) | 40 | 2 | 1 | 1 |
| Quinoa (q) | 20 | 5 | 6 | 10 |

# Solution

(in class)

# Spacecraft Maneuver

Spacecraft is docked at co-ordinates $(0, 0, 0)$.
We would like to move to position $(3, 3, 0)$.

Limited set of maneuvers:

| maneuver | $v_x$ | $v_y$ | $v_z$ | fuel/second |
|---|---|---|---|---|
| $M_1$ | .5 | .5 | .2 | 40 |
| $M_2$ | 1 | 0 | $-.5$ | 10 |
| $M_3$ | $-1$ | 2 | 1 | 10 |
| $M_4$ | 0 | 0 | $-1$ | 5 |

**Example Schedule:**
$M_1$ for 6 seconds, $M_4$ for 1.2 seconds.
Time taken is 7.2 seconds, Fuel consumed: $40 \times 6 + 5 \times 1.2 = 246$ units.

# Spacecraft Maneuvre (cont)

**Goal:** Find a schedule of maneuvers that minimizes the amount of fuel?

Let us do maneuvre $M_1$ for $t_1$ seconds, $M_2$ for $t_2$ seconds, and so on.
$t_1, t_2, t_3, t_4$ are called **decision variables**.

**Q1:** Find an expression for the final position?
**Q2:** Find an expression for fuel consumption.
**Q3:** Express the problem as an optimization problem.

# Spacecraft Maneuver Linear Program

# Solution

**Decision Variables:** $t_1, \ldots, t_4$, where $t_i$ is time for maneuver $M_i$.
**Objective:** Fuel consumption (to minimize):

$$40t_1 + 10t_2 + 10t_3 + 5t_4\,.$$

**Constraints:** We need to reach the final position:

$$\begin{array}{rl} & t_1, t_2, t_3, t_4 \geq 0 \\ x \rightarrow & .5t_1 + t_2 - t_3 = 3 \\ y \rightarrow & .5t_1 + 2t_3 = 3 \\ z \rightarrow & .2t_1 - .5t_2 + t_3 - t_4 = 0 \end{array}$$

**Optimal Solution:** $t_1 = \frac{30}{13}, t_2 = \frac{36}{13}, t_3 = \frac{12}{13}, t_4 = 0.$

# Visualizing Linear Programs

Consider the LP:

$$\begin{array}{rcl} \text{max.} & : & -x + 2y \\ \text{s.t.} & & x \leq 3 \\ & & x \geq -1 \\ & & 4y + x \leq 7 \\ & & 4y + x \geq -5 \end{array}$$

**Feasible Region:** Set of $(x, y)$ values that satisfy constraint.
**Observation:** Feasible region is a polyhedron.

# Outcomes for Linear Program

Three possible outcomes:

Infeasible: The constraints have no solution in the first place.

Unbounded: It is possible to get large values of the optimum.

Feasible & Bounded: There is an optimal solution.

# Example of Infeasible LP

Example #1:

$$
\begin{array}{rlll}
\text{max.} & : & -x + 2y & \\
\text{s.t.} & & x & \leq & 3 \\
 & & 4y + x & \leq & 7 \\
 & & x & \geq & 4 \\
 & & 4y + x & \geq & -5
\end{array}
$$

Example #2:

$$
\begin{array}{rlll}
\text{max.} & : & -x + 2y & \\
\text{s.t.} & & x + y & \geq & 10 \\
 & & x & \leq & 8 \\
 & & y & \leq & 1
\end{array}
$$

# Example of Unbounded LP

$$\begin{array}{rlrl} \text{max.} & : & -x+2y & \\ \text{s.t.} & & x & \leq & 3 \\ & & y & \geq & 7 \\ & & -x+2y & \geq & -5 \end{array}$$

Why unbounded?

| Solution | Obj Val |
|---|---|
| $x = 3, y = 7$ | 11 |
| $x = -3, y = 10$ | 23 |
| $x = -3 \times 10^{10}, y = 5 \times 10^{10}$ | $13 \times 10^{10}$ |
| $x \to -\infty, y \to \infty$ | $\infty$ |

# Solving Linear Programs

Three categories of methods:

- Simplex (George Dantzig, 1940s): Hill climbing search on vertices.
  **Complexity:** Exponential in # Vars + # Constraints.

  Polynomial time on almost all instances.

- Ellipsoid Method (Leonid Kachiyan, 1970s): Approximate feasible region through a sequence of ellipsoids of decreasing volume.
  **Complexity:** Polynomial time (not a practical method).

# Solving LPs (cont)

- Interior Point Methods (Narendra Karmarkar, 1982): Practical polynomial time algorithm.
  Recent work: Path following methods, primal-dual methods, predictor-corrector methods,...

Other approaches for special cases:

- Two variables per constraint case.
- Transshipment problem, problems over graphs: Network Simplex Method.
- ..

# History of Linear Programming

- G.B. Dantzig (1948)
  - The Simplex Algorithm.
  - Motivated by planning/scheduling problems during World War II.
  - Early publication (RAND Corporation tech. report in 1948).
  - Linear Programming & Extensions book.
- Dantzig's ideas used extensively in game theory.
  (Kuhn, Motzkin, ..)
- Ellipsoid method (Kachiyan, 1979).
  - First polynomial time method.
  - Theoretical interest.
- Interior point methods ( Karmarkar 1984)
  - First practical polynomial time technique.
  - Analysis and extended to general convex optimization.
    (Nestorov & Nemirovsky, 80s, 90s).
- Integer Linear Programming.
  - NP-completeness theory (Cook'71, Karp'72).
  - Very general, easy reductions from a bunch of other problems.
  - Recent progress in randomized and approximation algorithms.

# Thursday's Lecture

Formal start of class. We will cover:

1. Linear programming examples
2. Standard form of LP
3. Slack variables and
4. Run through the Simplex algorithm informally.

Would be nice if you go over Chvátal Ch.1, 2.