Scientific Computing

Lecture 05: Linear Programming
Part 01

Vladimir Kulyukin
Department of Computer Science
Utah State University

Some Notes on HW01

Request 1: do not change the names of the Python module files (e.g., `cs3430_s24_hw01.py`). The unit tests import functions from these modules and will not run with changed names.

Request 2: write your name and `A#` at the beginning of the files you are submitting. This info makes it easier for me to enter your scores in Canvas.

Unit test 1.4: My unit test 1.4 was off, because it tested the linear system as if it was solvable. It is not solvable, because the determinant of the coefficient matrix `A` is 0. Thus, when testing your solutions I modified the unit test as stated on the next slide.

Multiprocessing in Determinant Computation: When we went over a multi-processing optimization to the leibnitz determinant problem in the F2F lecture on 01/24/24, some of you asked me to post that solution in Canvas. Please see Slides 18 – 21 in the Lecture 04 PDF in Canvas.

Modified Unit Test 1.4 for HW01

```
def test_hw01_prob_1_4(self):
    try:
      A, x, b = cs3430_s24_hw_01_prob_1_4()
      if np.allclose(np.linalg.det(A), 0):
         print('det(A) ==0; Ax = b is unsolvable')
         return True
         assert np.allclose(np.dot(A, x), b)
    except Exception as e:
         print(e)
         assert 1 == 0
```

Optimization Problems

An **optimization** problem is formulated as follows.

Given a set of **variables** $x_1, x_2, ..., x_n$ and some **quantity** that depends on these variables (i.e., $f(x_1, ..., x_n)$), **maximize/minimize** this quantity in such a way that the values of the variables satisfy a specific set of **constraints**.

We will focus on problems where the quantity and constraints are expressable as linear functions or line inequalities. Hence, the term **Linear Programming**.

Linear Programming

Linear Programming (LP) is a method to solve optimization problems where the quantity being maximized/minimized (i.e., $f(x_1, \ldots, x_n)$) and all constraints are **linear** functions.

LP is used in transportation, logistics, distribution, manufacturing, marketing, resource allocation.

When solving a LP problem, we need to identify the **decision variables**, the **objective function**, and the **linear constraints**.

Decision Variables, Constraints, Feasible Sets, Objective Functions

The **decision variables** identify the resources mentioned in the problem.

The **constraints** are inequalities that put limits on the amount of resources mentioned in the problem and on the ranges of the decision variables. The constraints we will work with will always be inclusive, i.e., the ineqalities expressed with $\geq$ and $\leq$.

The set of points satisfying the constraints of the problem is known as the **feasible set** of the problem.

The function to be maximized/minimized is known as the **objective function** for the problem. The objective function is a function of the decision variables and, typically, involves the computation of the cost/profit from the values of the decision variables.

LP Problem Formulation: 3 Steps

1) Define the decision variables.

2) Formulate the constraints using the decision variables.

3) Define the objective function using the decision variables.

LP Problem 1

Ted's Toys makes toy cars and trucks using plastic and steel. Each car requires 4 ounces of plastic and 3 ounces of steel, while each truck requires 3 ounces of plastic and 6 ounces of steel. Each day Ted has 30 pounds of plastic and 45 pounds of steel to use for manufacturing. Ted's profit is $5 per car and $4 dollars per truck. How many cars and trucks should Ted make to maximize his profit?

LP Problem 1

1) We can define the following decision variables: x_1 is the number of cars built per day and x_2 is the number of trucks built per day.

2) Since each car requires 4 ounces of plastic and each truck requires 3 ounces of plastic we have the following plastic constraint: $4x_1 + 3x_2 \leq 480$, where $480 = 30 \cdot 16$.

Since each truck car requires 3 ounces of steel and each truck requires 6 ounces of steel, we have the following steel constraint: $3x_1 + 6x_2 \leq 720$, where $720 = 45 \cdot 16$.

We have the following common sense constraints: $x_1 \geq 0$ and $x_2 \geq 0$.

3) The objective function is $f(x_1, x_2) = 5x_1 + 4x_2$. This is the function that we must maximize.

LP Problem 1: Summary

Let x_1 be the number of toy cars to make every day. Let x_2 is the number of toy trucks to make each day. x_1 and x_2 are the **decision variables**.

The **objective function** is $f(x_1, x_2) = 5x_1 + 4x_2$.

The **constraints** are

1. $4x_1 + 3x_2 \leq 480$ (plastic constraint);
2. $3x_1 + 6x_2 \leq 720$ (steel constraint);
3. $x_1 \geq 0$ (number of cars must be non-negative);
4. $x_2 \geq 0$ (number of trucks must be non-negative).

LP Problem 2

Let us quickly consider another LP problem.

```
A hiker plans her trail foor for a long hike,
which is to include a mix of peanuts and raisins.
Each day she needs 600 calories and 90 grams of
carbohydrates from the mix. Each gram of raisins
contains 0.8 grams of carbohydrates and 3 calories
and costs 4 cents. Each gram of peanuts contains
0.2 grams of carbohydrates and 6 calories and costs
5 cents. Find the number of grams of each food
which will meet the hiker's constraints at the
smallest cost per day.
```

LP Problem 2

1) We can define the following decision variables: x is the number of grams of raisins and y is the number of grams peanuts.

2) Since each gram of raisins contains 0.8 grams of carbs and each gram of peanuts contains 0.2 grams of carbs, we have the following carb constraint: $0.8x + 0.2y \geq 90$.

Since each gram of raisins contains 3 calories and each gram of peanuts contains 6 calories, we have the following caloric constraint: $3x + 6y \geq 600$.

3) The objective function is $f(x, y) = 4x + 5y$. This is the function that we must minimize.

Two Half Planes in 2D

Let us review lines in 2D. This work will help us transition to from 2D to multi-dimensional LP problems later.

Each linear inequality in 2D of the form $Ax + By \leq C$ or $Ax + By \geq C$ corresponds to the following *boundary* line $Ax + By = C$.

The boundary line $Ax + By = C$ devides the plane into 2 *half planes*.

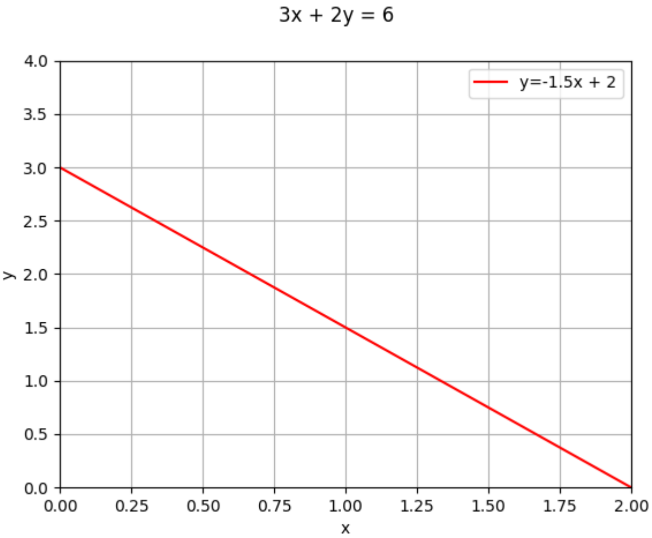
We can use any point off the specified line $Ax + By = C$ to determine which half plane satisfies the inequality $Ax + By \leq C$ or $Ax + By \geq C$.

Graphing in 2D with Matplotlib

We can use `matplotlib` to plot a line. Here's an example of a function that plots $f(x) = -1.5x + 3$. The source code is in `graphs2D.py`.

```
import numpy as np
import matplotlib.pyplot as plt
def lec_05_01():
    def f1(x): return -1.5*x + 3.0
    xvals = np.linspace(-2, 2, 10000)
    yvals1 = np.array([f1(x) for x in xvals])
    fig1 = plt.figure(1)
    fig1.suptitle('3x + 2y = 6')
    plt.xlabel('x')
    plt.ylabel('y')
    plt.ylim([0, 4])
    plt.xlim([0, 2])
    plt.grid()
    plt.plot(xvals, yvals1, label='y=-1.5x + 3', c='r')
    plt.legend(loc='best')
    plt.show()
```

Graphing in 2D with Matplotlib



LP Problem 3

Let us consider another LP problem.

A hiker believes that on a long hike she'll need snacks with at least 600 calories. She plans to take chocolate and raisins. The chocolate has 150 calories per ounce and raisins have 80 calories per ounce. Find and graph the system of inequalities that satisfy the hiker's constraints.

LP Problem 3

1. We have the following decision vars:

    1. x – number of ounces of chocolate;
    2. y – number of ounces of raisins.

2. Since the hiker needs at least 600 calories, we have the constraint $150 + 80y \geq 600$. In total, we have 3 constraints for this problem:

    1. $150x + 80y \geq 600$;
    2. $x \geq 0$;
    3. $y \geq 0$.

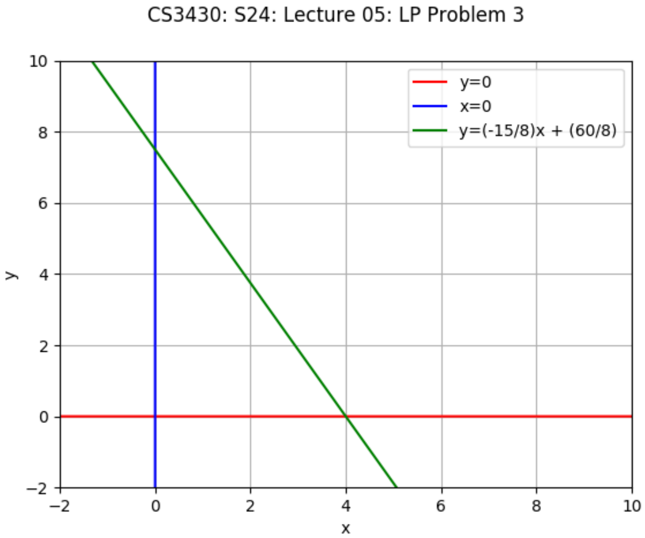
3. The feasible set for this problem is the intersection of the 3 sets: $\{(x, y)|x \geq 0\}$, $\{(x, y)|150x + 80y \geq 600\}$, $\{(x, y)|y \geq 0\}$.

Graphing LP Problem 3

We can write the following function (source code in graphs2D.py) to graph the feasible set for this problem. Note that we turn $150x + 80y = 600$ into $y(x) = \frac{-15}{8}x + \frac{60}{8}$.

```
def lec_05_lp_problem_03():
    def y(x): return (-15/8)*x + (60/8)
    xvals = np.linspace(-2, 10, 10000)
    yvals1 = np.array([y(x) for x in xvals])
    yvals = np.linspace(-2, 10, 10000)
    zero_x = np.linspace(0, 0, 10000)
    zero_y = np.linspace(0, 0, 10000)
    fig1 = plt.figure(1)
    fig1.suptitle('CS3430: S24: Lecture 05: LP Problem 03')
    plt.xlabel('x')
    plt.ylabel('y')
    plt.ylim([-2, 10])
    plt.xlim([-2, 10])
    plt.grid()
    plt.plot(xvals, zero_y, label='y=0', c='r')
    plt.plot(zero_x, yvals, label='x=0', c='b')
    plt.plot(xvals, yvals1, label='y=(-15/8)x + (60/8)', c='g')
    plt.legend(loc='best')
    plt.show()
```

Graphing in 2D with Matplotlib



References

1. D.P. Maki, M. Thompson. *Finite Mathematics*.