ECON 6100

Problem Set 1

Gabe Sekeres

May 7, 2025

1. Problem of shipping books.

   (a) In canonical form, defining x as the number of books shipped from Novato to San Francisco, y as the number of books shipped from Novato to Sacramento, z as the number of books shipped from Lodi to San Francisco, and w as the number of books shipped from Lodi to Sacramento, this problem is

$$\begin{aligned} v_P(b) = \max\ & -(5x + 10y + 15z + 4w) \\ \text{s.t.}\quad & -x - z \leq -600 \\ & -y - w \leq -400 \\ & x + y \leq 700 \\ & z + w \leq 800 \\ & x, y, z, w \geq 0 \end{aligned}$$

   In matrix form, we could represent this problem as

$$\begin{aligned} v_P(b) = \max\ & c \cdot x \\ \text{s.t.}\quad & Ax \leq b \\ & x \geq 0 \end{aligned}$$

   where

$$A = \begin{bmatrix} -1 & 0 & -1 & 0 \\ 0 & -1 & 0 & -1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \qquad \text{and} \qquad b = \begin{bmatrix} -600 \\ -400 \\ 700 \\ 800 \end{bmatrix} \qquad \text{and} \qquad c = \begin{bmatrix} -5 \\ -10 \\ -15 \\ -4 \end{bmatrix}^T$$

   (b) Using the same variables as in part (a), and introducing the slack variable $\lambda \in \mathbb{R}^4$, the problem in standard form is

$$\begin{aligned} v_P(b) = \max\ & -(5x + 10y + 15z + 4w) + 0 \cdot \lambda \\ \text{s.t.}\quad & -x - z + \lambda_1 = -600 \\ & -y - w + \lambda_2 = -400 \\ & x + y + \lambda_3 = 700 \\ & z + w + \lambda_4 = 800 \\ & x, y, z, w \geq 0 \\ & \lambda \geq 0 \end{aligned}$$

   In matrix form, we could represent this problem as

$$\begin{aligned} v_P(b) = \max\ & c \cdot x + 0 \cdot \lambda \\ \text{s.t.}\quad & Ax + I_4 \lambda = b \\ & x \geq 0 \\ & \lambda \geq 0 \end{aligned}$$

where, as above,

$$A = \begin{bmatrix} -1 & 0 & -1 & 0 \\ 0 & -1 & 0 & -1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \qquad \text{and} \qquad b = \begin{bmatrix} -600 \\ -400 \\ 700 \\ 800 \end{bmatrix} \qquad \text{and} \qquad c = \begin{bmatrix} -5 \\ -10 \\ -15 \\ -4 \end{bmatrix}^T$$

and I_4 is the 4×4 identity matrix.

2. Starting from a linear program

   (a) To write this problem in canonical form, we need to first deal with the free variable y. Define $y = y^+ - y^-$, and we can say that $y^+, y^- \geq 0$. Our problem in canonical form is

$$\begin{aligned} v_P(b) = \max\ & c_x x + c_y y^+ - c_y y^- \\ \text{s.t.} \quad & x \leq 1 \\ & x, y^+, y^- \geq 0 \end{aligned}$$

   With matrices, we have that the problem is

$$\begin{aligned} & \max c \cdot z \\ \text{s.t.} \quad & Az \leq b \\ & z \geq 0 \end{aligned}$$

   where

$$z = \begin{bmatrix} x \\ y^+ \\ -y^- \end{bmatrix}^T \qquad c = \begin{bmatrix} c_x \\ c_y \\ c_y \end{bmatrix} \qquad A = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad b = 1$$

   (b) Using the same variable replacement as above, and introducing the slack variable $\lambda \in \mathbb{R}$, the problem in standard form is

$$\begin{aligned} v_P(b) = \max\ & c_x x + c_y y^+ - c_y y^- + 0 \cdot \lambda \\ \text{s.t.} \quad & x + \lambda = 1 \\ & x, y^+, y^- \geq 0 \\ & \lambda \geq 0 \end{aligned}$$

   With matrices, we have that the problem is

$$\begin{aligned} v_P(b) = \max\ & c \cdot z + 0 \cdot \lambda \\ \text{s.t.} \quad & Az + I^3 \lambda = b \\ & z \geq 0 \\ & \lambda \geq 0 \end{aligned}$$

   where, as above,

$$z = \begin{bmatrix} x \\ y^+ \\ -y^- \end{bmatrix}^T \qquad c = \begin{bmatrix} c_x \\ c_y \\ c_y \end{bmatrix} \qquad A = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \qquad b = 1$$

3. The primal problem is

$$\begin{aligned} v_P(b) = \max\ & x_1 + 2x_2 \\ \text{s.t.} \quad & x_1 + x_2 \leq 4 \\ & x_1 + 3x_2 \leq b \end{aligned}$$

(a) The primal problem represented in standard form, attained by introducing variables $x_1^+, x_1^-, x_2^+, x_2^+$, is

$$\begin{aligned} v_P(b) = \max\ & x_1^+ - x_1^- + 2x_2^+ - 2x_2^- \\ \text{s.t.} \quad & x_1^+ - x_1^- + x_2^+ - x_2^- \le 4 \\ & x_1^+ - x_1^- + 3x_2^+ - 3x_2^- \le b \\ & x_1^+, x_1^-, x_2^+, x_2^- \ge 0 \end{aligned}$$

We have that the relevant matrices are

$$c = \begin{bmatrix} 1 \\ -1 \\ 2 \\ -2 \end{bmatrix}^T \quad ; \quad b = \begin{bmatrix} 4 \\ b \end{bmatrix} \quad ; \quad A = \begin{bmatrix} 1 & -1 & 1 & -1 \\ 1 & -1 & 3 & -3 \end{bmatrix}$$

The dual problem is defined as

$$\begin{aligned} v_D(c) = \min\ & y \cdot b \\ \text{s.t.} \quad & yA \ge c \\ & y \ge 0 \end{aligned}$$

Which becomes

$$\begin{aligned} v_D(c) = \min\ & 4y_1 + by_2 \\ \text{s.t.} \quad & y_1 + y_2 \ge 1 \\ & -y_1 - y_2 \ge -1 \\ & y_1 + 3y_2 \ge 2 \\ & -y_1 - 3y_2 \ge -2 \\ & y_1, y_2 \ge 0 \end{aligned}$$

(b) The constraint sets for the problems are in Figure 1 (on the next page), with the primal constraint set shaded in Green and the dual constraint set in Purple. Note that the dual constraint set is a point – it is defined precisely by the two equations $y_1 + y_2 = 1$ and $y_1 + 3y_2 = 2$, the intersection of which is exactly the point $(0.5, 0.5)$. The optimal points are also represented. Since the Duality Theorem holds, the same optimal value is attained in both problems – in the primal, the value of 2.5 is attained at the point $(5.5, -1.5)$, and in the dual the value of 2.5 is attained at the point $(0.5, 0.5)$.

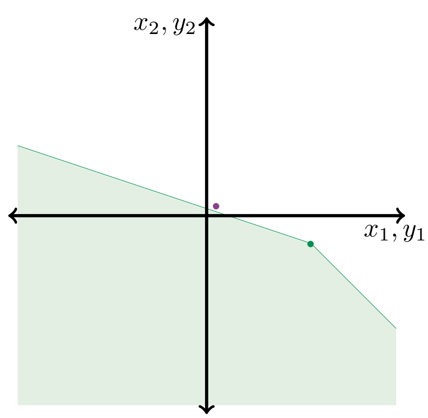

Figure 1: Constraint Sets

(c) The function $v_P(b)$ is, from the Vertex Theorem, always defined by the intersection of the functions $x_1 + x_2 = 4$ and $x_1 + 3x_2 = b$, where if $(x_1^\star, x_2^\star)$ is a solution to that system, $v_P(b) = x_1^\star + 2x_2^\star$. Specifically, since we have that $x_2 = 4 - x_1$, we have that $x_1 = 6 - \frac{b}{2}$, and $x_2 = \frac{b}{2} - 2$, so

$$v_P(b) = 6 - \frac{b}{2} + b - 4 = 2 + \frac{b}{2}$$

Thus, for $b \in [0, 14]$, $\partial v_P(b) = \frac{1}{2}$. Both results are confirmed by solving the linear program for a discretization of that space, which leads to the plot in Figure 2.

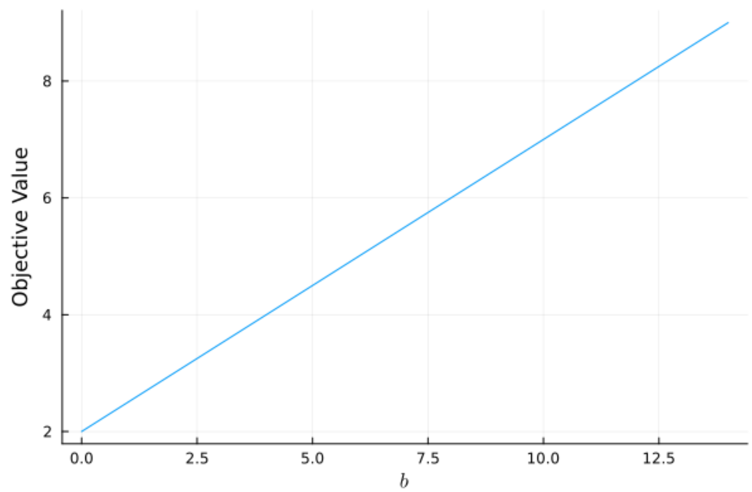

Figure 2: Objective value for a range of b, 0 to 14 with step size 0.01.

The code to create this figure, as well as to solve the linear program and the dual, is:

```
using JuMP, HiGHS, Plots, LaTeXStrings

function solve_lp(b)
    model = Model(HiGHS.Optimizer)

    @variable(model, x[1:2])

    @objective(model, Max, x[1] + 2x[2])

    @constraint(model, x[1] + x[2] <= 4)
    @constraint(model, x[1] + 3x[2] <= b)

    set_silent(model)
    optimize!(model)

    status = termination_status(model)
    if status == OPTIMAL
        return string(status), objective_value(model), value.(x)
    else
        return string(status), Float64[], Float64[]
    end
```

```julia
end

function solve_dual(b)
    model = Model(HiGHS.Optimizer)

    @variable(model, y[1:2] >= 0)

    @objective(model, Min, 4y[1] + b*y[2])

    @constraint(model, y[1] + y[2] == 1)
    @constraint(model, y[1] + 3y[2] == 2)

    set_silent(model)
    optimize!(model)

    status = termination_status(model)
    if status == OPTIMAL
        return string(status), objective_value(model), value.(y)
    else
        return string(status), Float64[], Float64[]
    end
end

b = 1
status, objective, solution = solve_lp(b)
println("Status: $status")
println("Objective value: $objective")
println("Solution: $solution")

dual_status, dual_objective, dual_solution = solve_dual(b)
println("Dual Status: $dual_status")
println("Dual Objective value: $dual_objective")
println("Dual Solution: $dual_solution")


range_of_b = 0:0.01:14
values = zeros(length(range_of_b))

for (i, b) in enumerate(range_of_b)
    values[i] = solve_lp(b)[2]
end

using Plots
plot(range_of_b, values, label="Objective Value", xlabel=L"b", ylabel=
   "Objective Value", legend=false)
savefig("ps1_objective_value.png")
```