Homework #3

Due July 11 @ 11:59pm

Submission requirements

Upload a **single PDF or HTML file** of your IJulia notebook for this entire assigment. Clearly denote which question each section of your file corresponds to.

Problem 1 -- Patty cake, patty cake

The bakery from Homework 2 is expanding operations to include more than just cakes, but they need your help! The bakery wants to make a variety of desserts, each classified according to one of four types: {Cake, Cookie, Pastry, Candy}. Each dessert belongs to exactly one of the four types. There is a set of 11 different features the bakery has identified by reading extensive reviews of desserts, and the bakery wants to ensure that they offer at least one dessert that has each of the features. To cut down on cost and prep, the bakery wants to make no more than three desserts of each possible type.

(a) Write a **maximum flow problem** that will tell the bakery whether or not there is a feasible set of desserts to make that will meet the bakery's requirements. *Hint:* Draw the network first. *Hint2.0:* There is some code below that will help you get started building the nodes and arcs in Julia. There is more than one way to build this network; you do not have to follow the structure I have given below.

(b) Without doing any additional calculations, what is the smallest upper bound you can give on the maximum flow through this network (i.e., the minimum cut)? What set of arcs gives this minimum cut? Use your answer to (a) and duality theory to prove such a cut exists (*Hint:* You can answer this directly with a single statement from the duality module).

(c) (Ungraded) To get more familiar with random number generation and manipulating Julia code, try altering some of the input data (e.g., the seed or the density parameter). Note what happens. How low can you make the density before no solution can be found? What would change if you required **exactly** 3 desserts from each group, rather than 3 being an upper bound?

In [3]:

```julia
# types of desserts
types = [:cake, :pastry, :cookie, :candy]

# desserts that are cakes
cakes = [:carrot, :chocolate, :vanilla, :angelfood, :blackforest, :funfetti, :germanchoc, :oliveoil, :fruitcake

# desserts that are pastries
pastries = [:eclair, :creampuff, :danish, :croissant, :baklava, :handpie, :galette]

# desserts that are cookies
cookies = [:chocolatechip, :peanutbutter, :snickerdoodle, :cowboy, :molasses, :gingersnap,
    :thumbprint, :shortbread, :teacake, :sugar, :oatmealraisin, :biscotti]

# desserts that are candy
candy = [:truffle, :rockcandy, :fudge, :cakepop, :toffee, :caramel, :taffy]

# all the possible desserts
desserts = vcat(cakes,pastries,cookies,candy)

# dessert features, using words from reviews of the flavors
features = [:chewy, :sweet, :salty, :crunchy, :crumbly, :complex, :flakey, :buttery, :airy, :heavy, :fluffy]

using Random
# set a seed so we get the same random numbers every time
seed = 345899
Random.seed!(seed)
# randomly assign features to  desserts
dessert_feature = Dict()
density = 0.15 # between 0 and 1. lower density means features less likely to be assigned
for d in desserts
    for f in features
        # if a (pseudo)randomly generated uniform(0,1) number is less than density
        if Random.rand() < density
            # the dessert has that feature
            dessert_feature[(f,d)] = 1
        end
    end
end

# create the list of nodes in the graph
nodes = vcat([:source], features, desserts, types, [:sink]);

# create all arcs needed in the network
arcs_1 = [(:source,i) for i in features] # arcs from source to features
arcs_2 = [n[1] for n in dessert_feature] # arcs from features to desserts with that feature
arcs_3 = [(i,:cake) for i in desserts if i in cakes] # arcs from desserts to cake type
arcs_4 = [(i,:pastry) for i in desserts if i in pastries] # arcs from desserts to pastry type
arcs_5 = [(i,:cookie) for i in desserts if i in cookies] # arcs from desserts to cookie type
arcs_6 = [(i,:candy) for i in desserts if i in candy] # arcs from desserts to candy type

# concatenate all the arcs and add arcs from each type to the sink
# also add the dummy arc from sink to source
arcs = vcat(arcs_1, arcs_2, arcs_3, arcs_4, arcs_5, arcs_6,
         [(:cake,:sink),(:pastry,:sink),(:cookie,:sink),(:candy,:sink)], [(:sink,:source)])
## Problem 1 -- Patty cake, patty cake
# create a dictionary of capacities indexed over arcs
capacity = Dict()
for a in arcs
    # capacity on each arc to the features areas is 1 (if max flow < 11, no solution exists)
    if a[1] == :source
        capacity[a] = 1
    # capacity on each arc between features and desserts is 1 (dessert can only have a feature once)
    elseif a[1] in features
        capacity[a] = 1
    # capacity on each arc between desserts and types is upper bound on total flow
    # (a dessert could theoretically have every feature)
   elseif a[1] in desserts
        capacity[a] = 11
    # capacity on each arc between types and sink is 3
    elseif a[1] in types
        capacity[a] = 3
    # capacity on dummy arc is an upper bound on total flow
    elseif a == (:sink,:source)
        capacity[a] = 11
    end
end

# create a dictionary of costs, indexed over arcs
cost = Dict()
for a in arcs
    # dummy arc cost is 1
    if a == (:sink,:source)
        cost[a] = -1
    # all other arc costs are 0
    else
        cost[a] = 0
    end
end

using JuMP, Clp

m = Model(Clp.Optimizer)

@variable(m, x[arcs] >= 0)

@constraint(m, cap[a in arcs], x[a] <= capacity[a])
@constraint(m, constr[n in nodes], sum(x[a] for a in arcs if a[1] == n) == sum(x[a] for a in arcs if a[2] == n)

@objective(m, Max, x[(:sink,:source)])

set_optimizer_attribute(m, "LogLevel",false)
optimize!(m)

if objective_value(m) == 11.0
    println("Solution found:" )
    println("Desserts to make:")
    for i in arcs
        if i[2] in desserts
            if value(x[i]) > 0.00001
                println(i[2])
            end
        end
    end
else
    println("No solution found" )
end
```

```
Solution found:
Desserts to make:
handpie
molasses
biscotti
handpie
angelfood
fruitcake
croissant
caramel
chocolatechip
cakepop
toffee
```

Solution to (b)

The arcs from the source to the features give a minimum capacity cut (11). We know this from the fact that min cut = max flow, and the max flow in the network is 11. Furthermore, we know if a primal solution exists and is finite, by strong duality the dual solution also exists and has objective equal to that of the primal.

Problem 2 - Duality!

Suppose you are a manager at a local Chipotle restaurant and you are trying to schedule your staff for the next day. There are four time periods during the day that must be covered ($t = 1, 2, 3, 4$), and 3 possible shifts, each of which corresponds to working 2 consecutive time periods. The cost per employee assigned to shift i is c_i, for $i = 1, 2, 3$. Each time period has a required number of employees who must be working in order to cover demand for your delicious customizable burritos (required employees are $7, 8, 5, 9$, in that order). Suppose you have already built an optimizaiton model to deterimine how many employees to hire for each shift (model is shown below). The decision variables for this model are x_i for $i = 1, 2, 3$, where x_1 represents the number of employees scheduled to work shift 1 (time periods 1 and 2), x_2 represents the number of employees scheduled to work shift 2 (time periods 2 and 3), and x_3 represents the number of employees hired to work shift 3 (time periods 3 and 4).

The complete linear program is as follows:

$$\begin{array}{llllll} \min & z = c_1x_1 + c_2x_2 + c_3x_3 & & & & \\ \text{s.t.} & x_1 & & & \geq 7 & (\text{Period } 1) \\ & x_1 + & x_2 & & \geq 8 & (\text{Period } 2) \\ & & x_2 + & x_3 & \geq 5 & (\text{Period } 3) \\ & & & x_3 & \geq 9 & (\text{Period } 4) \end{array}$$

$$x_1 \geq 0,\ x_2 \geq 0, x_3 \geq 0$$

(a) Write the dual of this linear program. (Use y_1, y_2, y_3, y_4 as the dual variables.)

In the following questions, you will need the information that an optimal dual solution is $y^* = (0, 90, 0, 80)$.

(b) What is the optimal primal objective value?

(c) The marketing department is considering running a promotion that would increase the required number of employees to 10 in period 2, to 9 in period 3 and to 11 in period 4 (no change in period 1). Provide an estimate of the cost of running this promotion (i.e., how much will the optimal value change). Also indicate whether the actual cost might be higher or lower than this estimate.

(d) Asumme the shift requirements are the original requirements in the problem statement. The HR department has increased the salary for employees by $\$5$ for all shifts (so cost of shift 1 is now $c_1 + 5$, shift 2 is $c_2 + 5$, and shift 3 is $c_3 + 5$). Provide an estimate of the new cost of hiring employees to cover shifts (i.e., how much will the optimal value change). Also indicate whether the actual cost might be higher or lower than this estimate.

(a)

Solution

$$\begin{array}{llllll} \max & z = 7y_1 + 8y_2 + 5y_3 + 9y_4 & & & & \\ \text{s.t.} & y_1 + & y_2 & & \leq c_1 & (\text{Shift } 1) \\ & & y_2 + & y_3 & \leq c_2 & (\text{Shift } 2) \\ & & & y_3 + \ y_4 & \leq c_3 & (\text{Shift } 3) \end{array}$$

$$y_1 \geq 0,\ y_2 \geq 0, y_3 \geq 0,\ y_4 \geq 0$$

In the following questions, you will need the information that an optimal dual solution is $y^* = (0, 90, 0, 80)$.

(b) What is the optimal primal objective value?

Solution

By strong duality, we know $p* = 7(0) + 8(90) + 5(0) + 9(80) = 1440$.

(c) The marketing department is considering running a promotion that would increase the required number of employees to 10 in period 2, to 9 in period 3 and to 11 in period 4 (no change in period 1). Provide an estimate of the cost of running this promotion (i.e., how much will the optimal value change). Also indicate whether the actual cost might be higher or lower than this estimate.

Solution

The coefficients on the objective in the dual will change, but the feasible region will not. Thus, the old point remains feasible, but may no longer be optimal. The dual objective will now be $7y_1 + 10y_2 + 9y_3 + 11y_4$. If the same point remains optimal, the new cost will be 1780. This is a lower bound on the new optimal cost, since a new point may be "more optimal" than the old point.

(d) Asumme the shift requirements are the original requirements in the problem statement. The HR department has increased the salary for employees by 5 dollars for all shifts (so cost of shift 1 is now $c_1 + 5$, shift 2 is $c_2 + 5$, and shift 3 is $c_3 + 5$). Provide an estimate of the new cost of hiring employees to cover shifts (i.e., how much will the optimal value change). Also indicate whether the actual cost might be higher or lower than this estimate.

Solution

The coefficients on the objective in the primal will change, but the feasible region will not. Thus, the old primal point remains feasible, but may no longer be optimal. The primal objective at this solution will now be $1440 + 5x_1^* + 5x_2^* + 5x_3^*$. If the same point remains optimal, the new cost will be 1440 + $5(x_1^* + x_2^* + x_3^*)$. This is an upper bound on the new optimal cost, since a new point may be "more optimal" than the old point.

Problem 3 -- Duality, Take 2

Consider the following linear program:

$$\begin{aligned} &\max -3x_1 + x_2 + 2x_3 + x_4 \\ &\text{s.t.} -x_1 + x_3 + 2x_4 \leq 6 \\ &\quad 4x_2 - x_3 \leq 1 \\ &\quad x_1, x_2, x_3, x_4 \geq 0 \end{aligned} \tag{1}$$

(a) Write the dual of this linear program.

(b) Solve the (dual) LP from Part (a) using the graphical method.

(c) Using your graphical solution, give the range of values for which you could change the right-hand-side of the second constraint in the primal ($4x_2 - x_3 \leq 1$) before a new *dual* solution becomes optimal. In other words, what is the smallest possible right-hand-side value and what is the largest possible right-hand-side value for which the dual solution stays the same? Note that the dual optimal objective value will change.

(d) Use weak duality to prove that there cannot be an optimal solution to the primal problem with value larger than 100.

(e) Find the optimal solution to the primal problem using complementary slackness. Check that the strong duality theorem holds.

(a) Write the dual of this linear program.

Solution

$$\begin{aligned} &\min 6\lambda_1 \\ &\text{s.t.} -\lambda_1 \geq -3 \\ &\quad 4\lambda_2 \geq 1 \\ &\quad \lambda_1 - \lambda_2 \geq 2 \\ &\quad 2\lambda_1 \geq 1 \\ &\quad \lambda_1, \lambda_2 \geq 0 \end{aligned} \tag{2}$$

(b) Solve the (dual) LP from Part (a) using the graphical method.

Solution

The optimal solution is $\lambda_1 = 2.25$, $\lambda_2 = 0.25$. The objective is $6(2.25) = 13.5$.

(c) Using your graphical solution, give the range of values for which you could change the right-hand-side of the second constraint in the primal ($4x_2 - x_3 \leq 0$) before a new *dual* solution becomes optimal. In other words, what is the smallest possible right-hand-side value and what is the largest possible right-hand-side value for which the dual solution stays the same? Note that the dual optimal objective value will change.

Solution

Using the graph from part (b), we see that the same solution is optimal unless the objective is parallel to the constraint $4\lambda_2 \geq 1$ or the constraint $\lambda_1 - \lambda_2 \geq 2$. The coefficient on λ_2 (rhs of constraint in primal) would need to be increased to ∞ before the objective is parallel to the constraint $4\lambda_2 \geq 1$ (in other words, there is no upper bound). The lower bound on the value is -6. When the rhs of the second primal constraint is -6, the dual objective is $6\lambda_1 - 6\lambda_2$, so it is parallel to the constraint $\lambda_1 - \lambda_2 \geq 2$.

(d) Use weak duality to prove that there cannot be an optimal solution to the primal problem with value larger than 100.

Solution

By weak duality, $p* \leq d*$. We have found $d* = 13.5$. Since $100 > d*$, it cannot be an optimal solution to the primal.

(e) Find the optimal solution to the primal problem using complementary slackness. Check that the strong duality theorem holds.

Solution

At the dual optimal solution, the first and last dual constraints have slack, so $x_1 = x_4 = 0$. Since both dual variables are nonzero in the solution, both primal constraints must be satisfied at equality (compl. slackness). Therefore, $-x_1 + x_3 + 2x_4 = 6 \Rightarrow x_3 = 6$. The second constraint gives $4x_2 - x_3 = 0 \Rightarrow 4x_2 = 6 \Rightarrow x_2 = 1.5$. Verifying strong duality holds:
$-3x_1 + x_2 + 2x_3 + x_4 = 0 + 1.5 + 2(6) + 0 = 13.5$. Strong duality holds.