Matt Grogan

Midterm Exam I

PREDICT 401: Introduction to Statistical Analysis

1. *Joanne sells silk-screened t-shirts at a community festival. Her marginal cost to produce one t-shirt is $3.50. Her total cost to produce 60 t-shirts is $300, and she sells them for $7 each. Use Python to graph this information and determine the number of t-shirts Joanne must produce and sell to break even*.

import numpy as np

import matplotlib.pyplot as plt

x = np.arange(0,60,1)

marginal\_cost = 3.5

# total cost to produce 60 shirts is $300. 300/3.5 = 210. 300-210= 90. Fixed cost

fixed\_cost = 90

# total\_cost = marginal\_cost\*x + fixed\_cost

# marginal\_rev = 7\*x

marginal\_rev = 7

break\_even = 90//(7.0 - 3.5)

# you can't sell fractions of a shirt.

# You have to round up if there is a remainder.

remainder = 90%3.5

if remainder > 0: break\_even += 1

def cost(x):

return marginal\_cost\*x + fixed\_cost

def revenue(x):

return marginal\_rev\*x

plt.xlim(0,60)

plt.ylim(0,300)

plt.xlabel('Sales')

plt.ylabel('$ Amount')

plt.plot(x,cost(x))

plt.plot(x,revenue(x))

plt.scatter(break\_even, 7\*break\_even)

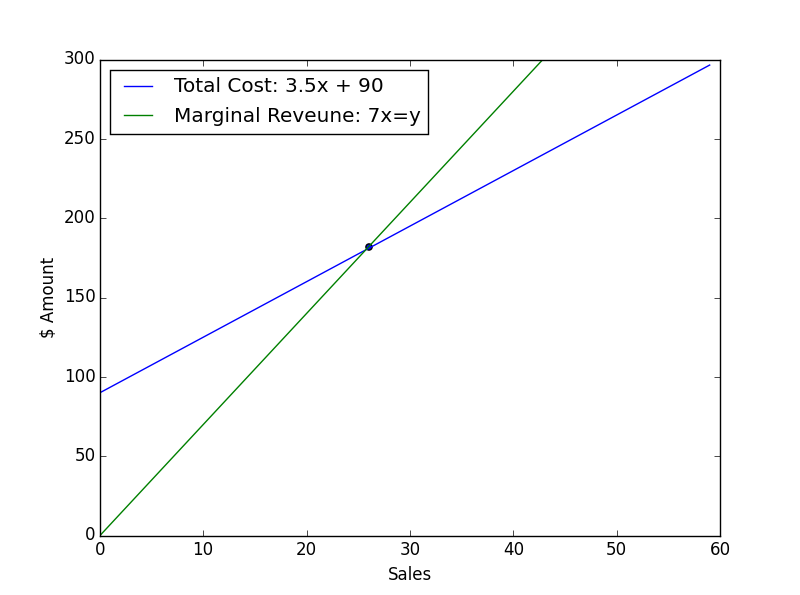
plt.legend(['Total Cost: 3.5x + 90','Marginal Reveune: 7x=y'],loc='best')

plt.show()

print 'They must sell {} to break even'.format(break\_even)

OUTPUT:

***They must sell 26 to break even***



1. *An electronics company produces transistors, resistors, and computer chips. Each transistor requires 3 units of copper, 1 unit of zinc and 2 units of glass. Each resistor requires 3, 2 and 1 units of the three materials, and each computer chip requires 2, 1, and 2 units of these materials respectively. How many of each product can be made with the following amount of material? 900 units of copper, 500 units of zinc, and 610 units of glass?*

This problem involves 3 equations with 3 unknowns, therefore we are trying to solve a system of equations like this Ax = b. I will use the Gauss Jordan method to solve this problem.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  | A = |  |  |  | b = |
|  | transistor | resistor | chip |  |  |
| copper | 3 | 3 | 2 |  | 900 |
| zinc | 1 | 2 | 1 |  | 500 |
| glass | 2 | 1 | 2 |  | 610 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| Finds a vector x that solves Ax = b. | | |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| A augmented | 3 | 3 | 2 | 900 |  |
| with b | 1 | 2 | 1 | 500 |  |
|  | 2 | 1 | 2 | 610 |  |
|  |  |  |  |  |  |
| Use row operations to reduce the A into an identity matrix. | | | | |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| R1 = R1/3 | 1 | 1 | 0.66667 | 300 |  |
|  | 1 | 2 | 1 | 500 |  |
|  | 2 | 1 | 2 | 610 |  |
|  |  |  |  |  |  |
|  | 1 | 1 | 0.66667 | 300 |  |
| R2 = R2-R1 | 0 | 1 | 0.33333 | 200 |  |
| R3 = R3-2\*R1 | 0 | -1 | 0.66667 | 10 |  |
|  |  |  |  |  |  |
|  | 1 | 1 | 0.66667 | 300 |  |
|  | 0 | 1 | 0.33333 | 200 |  |
| R3 = R2 + R3 | 0 | 0 | 1 | 210 |  |
|  |  |  |  |  |  |
| R1 = R1 – R2 | 1 | 0 | 0.33333 | 100 |  |
|  | 0 | 1 | 0.33333 | 200 |  |
|  | 0 | 0 | 1 | 210 |  |
|  |  |  |  |  |  |
| R1= R1-1/3\*R3 | 1 | 0 | 0 | 30 |  |
| R2 = r2-1/3\*r3 | 0 | 1 | 0 | 130 |  |
|  | 0 | 0 | 1 | 210 |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
| After reducing out vector x is the last column of the augmented matrix. | | | | | |
| We can check by multiplying Ax and verify the result is b. And it is. | | | | |  |
|  |  |  |  |  |  |
| Therefore we can have: | |  |  |  |  |
| **30 transistors** |  |  |  |  |  |
| **130 resistors** |  |  |  |  |  |
| **210 chips** |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |

1. *A pharmacist mixes together three types of vitamin tablets. Each tablet A contains, among other things, 13 mg of niacin and 11 IU of vitamin E. The amounts for a tablet B are 18 mg and 14 IU, and for a tablet C are 23 mg and 36 IU. Use Python to determine how many of each tablet there are if there are 225 total tablets, 4300 mg of niacin, and 5200 IU of vitamin E.*

import numpy as np

# [1,1,1,] [x1] = 225 total tablets

# [13,18,23] \* [x2] = 4300 mg of niacin

# [11,14,36] [x3] = 5200 IU vitamin E

A = np.array([[1,1,1,],

[13,18,23],

[11,14,36]])

b = np.array([225,4300,5200])

x = np.linalg.solve(A,b)

tablets = ['A', 'B', 'C']

for i in enumerate(x):

print 'There are {} of tablet {}'.format(i[1], tablets[i[0]])

print 'Ax = {}'.format(A.dot(x))

print 'b = {}'.format(b)

OUTPUT:

There are 50.0 of tablet A

There are 75.0 of tablet B

There are 100.0 of tablet C

Ax = [ 225. 4300. 5200.]

b = [ 225 4300 5200]

1. *Jayla is raising money for the homeless and discovers each church group requires 2 hours of letter writing and 1 hour of follow-up calls, while each labor union needs 2 hours of letter writing and 3 hours of follow-up. She can raise $100 from each church group and $175 from each union. She has a maximum of 20 hours of letter writing and 14 hours of follow-up available each month. Determine the most profitable mixture of groups she should contact and the most money she can raise in a month.*

The problem asks to maximize profit such that total hours spend writing letters and following up does not exceed available time.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| maximize 100\*x1 + 175\*x2 | | |  |  |  |  |  |
| st |  |  |  |  |  |  |  |
| 2\*x1 | 2\*x2 | <=20 |  |  |  |  |  |
| 1\*x1 | 3\*x2 | <=14 |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| Coefficient with most negative value | | | | |  |  |  |
| Entering basic variable | | | | |  |  |  |
| Exiting basic variable | | | | |  |  |  |
|  |  |  |  |  |  |  |  |
| I will use the simplex method to solve this problem. First set up the simplex Tableau by adding slack variables, and adding the negative coefficient of the function to be maximized to the last row. At each iteration I select the most negative coefficient (yellow). I test how much each constraint will allow the selected coefficient to increase by using the minimum ration test. Then divide the new basic variable by its coefficient to produce a 1 in its spot. Next i use row operations to produce zeros in all the rows above and below the new basic variable. This is repeated until there are no negative coefficient left. | | | | | |  |  |
|  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | Min ratio test | |
| basic vars | x1 | x2 | s1 | s2 | z | b |  |
| s1 | 2 | 2 | 1 | 0 | 0 | 20 | 10 |
| s2 | 1 | 3 | 0 | 1 | 0 | 14 | 4.66667 |
|  | -100 | -175 | 0 | 0 | 1 | 0 |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  | Min ratio test | |
| basic vars | x1 | x2 | s1 | s2 | z | b |  |
| s1 | 1.33333 | 0 | 1 | -0.66667 | 0 | 10.6667 | 8 |
| x2 | 0.33333 | 1 | 0 | 0.33333 | 0 | 4.66667 | 14 |
|  | -41.6667 | 0 | 0 | 58.3333 | 1 | 816.667 |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| basic vars | x1 | x2 | s1 | s2 | z | b |  |
| x1 | 1 | 0 | 0.75 | -0.5 | 0 | 8 |  |
| x2 | 0 | 1 | -0.25 | 0.5 | 0 | 2 |  |
|  | 0 | 0 | 31.25 | 37.5 | 1 | $ 1,150 |  |
| All coefficients are positive so we cannot increase the value any more. | | | | | | |  |
|  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |
| **Hours spend writing letters = x1 = 8** | | | |  |  |  |  |
| **Hours spent following up = x2 = 2** | | | |  |  |  |  |
| **At these levels you will maximize profit at $1,150** | | | | |  |  |  |

1. To be at his best as a teacher, Phil needs at least 10 units of vitamin A, 12 units of vitamin B, and 20 units of vitamin C per day. Pill #1 contains 4 units of A and 3 of B. Pill #2 contains 1 unit of A, 2 of B, and 4 of C. Pill #3 contains 10 units of A, 1 of B, and 5 of C. Pill #1 costs 6 cents, pill #2 costs 8 cents, and pill #3 costs 1 cent. How many of each pill must Phil take to minimize his cost?

This is a minimization problem. I will take advantage of duality here and solve the corresponding maximization problem using the same technique from question 4.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Stated Problem** | | | | | | | | | |
| Minimize .06x1 + .08x2 + .01x3 | | | | | | | | | |
| st |  |  |  |  |  |  |  |  |  |
| Pill 1 | Pill 2 | Pill 3 |  |  |  |  |  |  |  |
| 4 | 1 | 10 | >= | 10 | Vitamin A |  |  |  |  |
| 3 | 2 | 1 | >= | 12 | Vitamin B |  |  |  |  |
| 0 | 4 | 5 | >= | 20 | Vitamin C |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| **Solve the dual problem** | | | | | | | | | |
| Maximize 10y1 + 12y2 + 20y3 (maximize the amount of vitamins, without going over the per pill cost) | | | | | | | | | |
| St  Vitamin A | Vitamin B | Vitamin C |  |  |  |  |  |  |  |
| 4 | 3 | 0 | <= | 0.06 | Cost for 1 |  |  |  |  |
| 1 | 2 | 4 | <= | 0.08 | Cost for 2 |  |  |  |  |
| 10 | 1 | 5 | <= | 0.01 | Cost for 3 |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| I will use the simplex method to solve this problem. First set up the simplex Tableau by adding slack variables, and adding the negative coefficient of the function to be maximized to the last row. At each iteration I select the most negative coefficient (yellow). I test how much each constraint will allow the selected coefficient to increase by using the minimum ration test. Then divide the new basic variable by its coefficient to produce a 1 in its spot. Next i use row operations to produce zeros in all the rows above and below the new basic variable. This is repeated until there are no negative coefficient left. | | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | x1 | x2 | x3 | s1 | s2 | s3 | z | b | min ratio |
| s1 | 4 | 3 | 0 | 1 | 0 | 0 | 0 | 0.06 | null |
| s2 | 1 | 2 | 4 | 0 | 1 | 0 | 0 | 0.08 | 0.02 |
| s3 | 10 | 1 | 5 | 0 | 0 | 1 | 0 | 0.01 | 0.002 |
|  | -10 | -12 | -20 | 0 | 0 | 0 | 1 | 0 |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | x1 | x2 | x3 | s1 | s2 | s3 | z | b | min ratio |
| s1 | 4 | 3 | 0 | 1 | 0 | 0 | 0 | 0.06 | 0.02 |
| s2 | -7 | 1.2 | 0 | 0 | 1 | -0.8 | 0 | 0.072 | 0.06 |
| x3 | 2 | 0.2 | 1 | 0 | 0 | 0.2 | 0 | 0.002 | 0.01 |
|  | 30 | -8 | 0 | 0 | 0 | 4 | 1 | 0.04 |  |
| *\* Interesting here x3 was the first entering basic variable, and is the first chosen to leave* | | | | | | | | | |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | x1 | x2 | x3 | s1 | s2 | s3 | z | b |  |
| s1 | -26 | 0 | -15 | 1 | 0 | -3 | 0 | 0.03 |  |
| s2 | -19 | 0 | -6 | 0 | 1 | -2 | 0 | 0.06 |  |
| x2 | 10 | 1 | 5 | 0 | 0 | 1 | 0 | 0.01 |  |
|  | 110 | 0 | 40 | 0 | 0 | 12 | 1 | **0.12** |  |
|  |  |  |  |  |  |  |  |  |  |
| **Here our problem in maximized with x2 = .01** | | | | | | | | | |
| **However we also have the values we need to solve the stated original minimization problem.** | | | | | | | | | |
| **The values for the original problem are read from the slack variables coefficients (highlighted in orange)** | | | | | | | | | |
| **s1 = x1 = 0** | |  |  |  |  |  |  |  |  |
| **s2 = x2 = 0** | |  |  |  |  |  |  |  |  |
| **s3 = x3 = 12** | |  |  |  |  |  |  |  |  |
| **Minimum Cost = .12** | | |  |  |  |  |  |  |  |
| **The teacher can minimize his cost at 12 cents and still meet his nutritional needs by** | | | | | | | | |  |
| **taking 12 of pill C.** | |  |  |  |  |  |  |  |  |

1. *Recreate the following graph using Python. Be sure to replace ‘Red Line’ and ‘Blue Line with the correct equations.*

import numpy as np

import matplotlib.pyplot as plt

x = np.arange(0,26,1)

### equation 1 red line ####

y1 = -2\*x +20

### equation 2 blue line ####

y2 = -1.0\*x +15

### corners ###

xx = [5, 0, 0, 25, 25, 15]

yy = [10,20,25,25, 0, 0]

##### set plot attributes #####

plt.xlim(0,25)

plt.ylim(0,25)

plt.xlabel('x-axis')

plt.ylabel('y-axis')

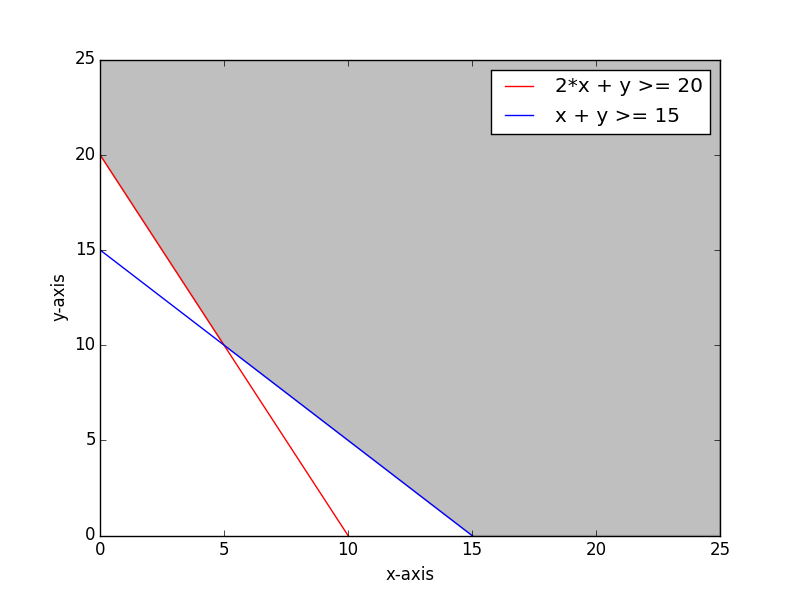
plt.plot(x,y1,c='red')

plt.plot(x,y2,c='blue')

plt.fill(xx,yy, color='grey',alpha=0.5)

plt.legend(['2\*x + y >= 20','x + y >= 15'],loc='best')

plt.show()



1. *Thor, a fitness trainer, has an exercise regimen that includes running, swimming, and walking. He has no more than 12 hours per week to devote to exercise, including at most 4 hours running. He wants to walk at least three times as many hours as he swims. Thor will burn on average 528 calories per hour running, 492 calories per hour swimming, and 348 calories per hour walking. Calculate how many hours per week Thor should spend on each exercise to maximize the number of calories he burns, as well as the maximum number of calories he will burn. (Hint: Write the constraint involving walking and swimming in the form ≤ 0.)*

This problem can be stated in the standard maximization form and solved using the simplex method.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Maximize | | | | | | |  |  |  |
| 528x1 + 492x2 + 348x3 (maximize calories burnt per week) | | | | | | |  |  |  |
| st |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| run | walk | swim |  | hours |  |  |  |  |  |
| 1 | 1 | 1 | <= | 12 | time on all 3 is less than 12 | | |  |  |
| 1 | 0 | 0 | <= | 4 | no more than 4 hrs running | | |  |  |
| 0 | 1 | -3 | <= | 0 | walk atleast 3 times as much as swim | | | |  |
|  |  |  |  |  |  |  |  |  |  |
| I will use the simplex method to solve this problem. First set up the simplex Tableau by adding slack variables, and adding the negative coefficient of the function to be maximized to the last row. At each iteration I select the most negative coefficient (yellow). I test how much each constraint will allow the selected coefficient to increase by using the minimum ration test. Then divide the new basic variable by its coefficient to produce a 1 in its spot. Next i use row operations to produce zeros in all the rows above and below the new basic variable. This is repeated until there are no negative coefficient left. | | | | | |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | x1 | x2 | x3 | s1 | s2 | s3 | z | b | min ratio |
| s1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 12 | 12 |
| s2 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 4 | 4 |
| s3 | 0 | 1 | -3 | 0 | 0 | 1 | 0 | 0 | null |
|  | -528 | -492 | -348 | 0 | 0 | 0 | 1 | 0 |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | x1 | x2 | x3 | s1 | s2 | s3 | z | b | min ratio |
| s1 | 0 | 1 | 1 | 1 | -1 | 0 | 0 | 8 | 8 |
| x1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 4 | null |
| s3 | 0 | 1 | -3 | 0 | 0 | 1 | 0 | 0 | 0 |
|  | 0 | -492 | -348 | 0 | 528 | 0 | 1 | 2112 |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | x1 | x2 | x3 | s1 | s2 | s3 | z | b | min ratio |
| s1 | 0 | 0 | 4 | 1 | -1 | -1 | 0 | 8 | 2 |
| x1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 4 |  |
| x2 | 0 | 1 | -3 | 0 | 0 | 1 | 0 | 0 |  |
|  | 0 | 0 | -1824 | 0 | 528 | 492 | 1 | 2112 |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | x1 | x2 | x3 | s1 | s2 | s3 | z | b |  |
| x3 | 0 | 0 | 1 | 0.25 | -0.25 | -0.25 | 0 | 2 |  |
| x1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 4 |  |
| x2 | 0 | 1 | 0 | 0.75 | -0.75 | 0.25 | 0 | 6 |  |
|  | 0 | 0 | 0 | 456 | 72 | 36 | 1 | 5760 |  |
|  |  |  |  |  |  |  |  |  |  |
| **x1 = hours running = 4** | | | |  |  |  |  |  |  |
| **x2 = hours walking =6** | | | |  |  |  |  |  |  |
| **x3 = hours swimming = 2** | | | |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| **Thor should spend 4 hours a week running, 6 hours walking and 2 hours swimming** | | | | | | | | |  |
| **This will maximize his calories burnt at 5760 while staying within this constraints.** | | | | | | | | |  |

1. *A plant food is made from three chemicals, labeled I, II, and III. In each batch of the plant food, the amounts of chemicals II and III must be in the ratio of 5 to 3. The amount of nitrogen must be at least 29 kg. The percent of nitrogen in the three chemicals is 8%, 4%, and 5%, respectively. If the three chemicals cost $1.03, $0.83, and $0.68 per kilogram, respectively, how much of each should be used to minimize the cost of producing at least 650 kg of the plant food?*

This problem is similar to problem 5 in that it is stated as a minimization problem, and we will use the dual to solve for the max, and use the slack variables from the dual to have our answers for the minimization problem. One interesting about this problem is the minimization problem has a constraint the forces the ratio between to variables to be 5 to 3. When set up as a constraint in the problem this constraint must be met with equality. If we solve the dual problem all the constraints are stated in the standard form (no constraints have to be met with equality)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Stated Problem** | | | |  |  |  |  |  |  |
| Minimize 1.03x1 + 0.83x2 + 0.68x3 | | | | |  |  |  |  |  |
| st |  |  |  |  |  |  |  |  |  |
| chem 1 | chem 2 | chem 3 |  |  |  |  |  |  |  |
| 0 | 1 | -5/3 | = | 0 | chem 2 & 3 must be in 5 to 3 ratio | | | |  |
| 0.08 | 0.04 | 0.05 | >= | 29 | nitrogen must be atleast 29kg | | |  |  |
| 1 | 1 | 1 | >= | 650 | produce atleast 650kg | | |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
| **Dual of Stated Problem** | | | |  |  |  |  |  |  |
| \* basically we want to the quality of 650 kg of product | | | | | |  |  |  |  |
| Maximize 0x1 + 52x2 + 650x3 | | | | |  |  |  |  |  |
| st |  |  |  |  |  |  |  |  |  |
| 0 | 0.08 | 1 | <= | 1.03 | cost per unit chem1 | |  |  |  |
| 1 | 0.04 | 1 | <= | 0.83 | cost per unit chem2 | |  |  |  |
| -5/3 | 0.05 | 1 | <= | 0.68 | cost per unit chem3 | |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | x1 | x2 | x3 | s1 | s2 | s3 | z | b | min ratio |
| s1 | 0 | 0.08 | 1 | 1 | 0 | 0 | 0 | 1.03 | 1.03 |
| s2 | 1 | 0.04 | 1 | 0 | 1 | 0 | 0 | 0.83 | 0.83 |
| s3 | -1.66667 | 0.05 | 1 | 0 | 0 | 1 | 0 | 0.68 | 0.68 |
|  | 0 | -29 | -650 | 0 | 0 | 0 | 1 | 0 |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | x1 | x2 | x3 | s1 | s2 | s3 | z | b | min ratio |
| s1 | 1.66667 | 0.03 | 0 | 1 | 0 | -1 | 0 | 0.35 | 0.21 |
| s2 | 2.66667 | -0.01 | 0 | 0 | 1 | -1 | 0 | 0.15 | 0.05625 |
| s3 | -1.66667 | 0.05 | 1 | 0 | 0 | 1 | 0 | 0.68 | null |
|  | -1083.33 | 3.5 | 0 | 0 | 0 | 650 | 1 | 442 |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | x1 | x2 | x3 | s1 | s2 | s3 | z | b | min ratio |
| s1 | 0 | 0.03625 | 0 | 1 | -0.625 | -0.375 | 0 | 0.25625 | 7.06897 |
| x2 | 1 | -0.00375 | 0 | 0 | 0.375 | -0.375 | 0 | 0.05625 | null |
| x3 | 0 | 0.04375 | 1 | 0 | 0.625 | 0.375 | 0 | 0.77375 | null |
|  | 0 | -0.5625 | 0 | 0 | 406.25 | 243.75 | 1 | 502.938 |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  | x1 | x2 | x3 | s1 | s2 | s3 | z | b |  |
| s1 | 0 | 1 | 0 | 27.5862 | -17.2414 | -10.3448 | 0 | 7.06897 |  |
| x2 | 1 | 0 | 0 | 0.10345 | 0.31034 | -0.41379 | 0 | 0.08276 |  |
| x3 | 0 | 0 | 1 | -1.2069 | 1.37931 | 0.82759 | 0 | 0.46448 |  |
|  | 0 | 0 | 0 | 15.5172 | 396.552 | 237.931 | 1 | 506.914 |  |

The answer to the original minimization problem can be read from the slack variables highlighted in orange.

**To minimize the cost of producing 650 kg of plant food they should use:**

**Chem 1: 15.5172**

**Chem 2: 396.552**

**Chem 3: 237.931**

**For a minimum cost of $506.91**

*9. Among uses of automated teller machines (ATMs), 93% use ATMs to withdraw cash and 32% use them to check their account balance. Suppose that 96% use ATMs to either withdraw cash or check their account balance (or both). Given someone who uses an ATM to check his or her balance, what is the probability that this person also uses an ATM to withdraw cash?*

I wrote a python script to help solve this problem:

print '\n'

# probability of withdrawl and check balance

pw = .93

pb = .32

print 'w = make a withdraw'

print 'b = check balance'

print "p(w) = {}".format(pw)

print "p(b) = {}".format(pb)

# probability of using ATM to withdrawl or check balance

pw\_or\_pb = .96

print "P(w or b) = {}".format(pw\_or\_pb)

print '\n'

# probability of using ATM and not withdrawling

# and probability of not checking balance

npw = 1-pw

npb = 1-pb

print "p(w') = {}".format(npw)

print "p(b') = {}".format(npb)

print '\n'

# probability of checking balance and withdrawling

# p(A & B) = P(A) + P(B) - P(A or B)

print "p(A & B) = P(A) + P(B) - P(A or B)"

print "p(w & b) = P(w) + p(b) - P(w or b)"

print "p(w & b) = {} + {} - {}".format(pw,pb,pw\_or\_pb)

pw\_a\_pb = pw + pb - pw\_or\_pb

print "p(w & b) = {}".format(pw\_a\_pb)

print '\n'

# P(B|A) = P(A & B)/P(A)

print "P(B|A) = P(A & B)/P(A)"

print "P(b|w) = p(w & b)/P(w)"

print "P(b|w) = {} / {}".format(pw\_a\_pb,pw)

pb\_given\_w = float(pw\_a\_pb) / pw

print "P(b|w) = {}".format(pb\_given\_w)

print '\n'

# P(b|w') = 1 - P(b|w)

print "P(b|w') = 1 - P(b|w)"

print "P(b|w') = 1 - {}".format(pb\_given\_w)

pb\_given\_nw = 1.0 - pb\_given\_w

print "P(b|w') = {}".format(pb\_given\_nw)

print '\n'

print "P(A|B) = P(A)\*P(B|A)/ P(A)\*(B|A) + P(A')\*P(B|A')"

print "P(w|b) = P(w)\*P(b|w)/ P(w)\*P(b|w)+ P(w')\*P(b|w')"

print "P(w|b) = {}\*{}/ ({}\*{})+({}\*{})".format(pw,pb\_given\_w,pw,pb\_given\_w,npw,pb\_given\_nw)

pw\_given\_b = float(pw\*pb\_given\_nw)/((pw\*pb\_given\_nw)+(npw\*pb\_given\_nw))

print "The probability of making a withdrawl given a balance check is:\nP(w|b) = {}".format(pw\_given\_b)

OUTPUT:

w = make a withdraw

b = check balance

p(w) = 0.93

p(b) = 0.32

P(w or b) = 0.96

p(w') = 0.07

p(b') = 0.68

p(A & B) = P(A) + P(B) - P(A or B)

p(w & b) = P(w) + p(b) - P(w or b)

p(w & b) = 0.93 + 0.32 - 0.96

p(w & b) = 0.29

P(B|A) = P(A & B)/P(A)

P(b|w) = p(w & b)/P(w)

P(b|w) = 0.29 / 0.93

P(b|w) = 0.311827956989

P(b|w') = 1 - P(b|w)

P(b|w') = 1 - 0.311827956989

P(b|w') = 0.688172043011

P(A|B) = P(A)\*P(B|A)/ P(A)\*(B|A) + P(A')\*P(B|A')

P(w|b) = P(w)\*P(b|w)/ P(w)\*P(b|w)+ P(w')\*P(b|w')

P(w|b) = 0.93\*0.311827956989/ (0.93\*0.311827956989)+(0.07\*0.688172043011)

**The probability of making a with draw given a balance check is:**

**P(w|b) = 0.93 = 93%**

Given the the probability of making a withdraw is 93% this implies making a withdraw is independent of checking your balance.

*10. A study showed that in 1990, 49% of all those involved in a fatal car crash wore seat belts. Of those in a fatal crash who wore seat belts, 46% were injured and 27% were killed. For those not wearing seat belts, the comparable figures were 41% and 52%, respectively. Find the probability that a randomly selected person who was unharmed in a fatal crash was not wearing a seat belt.*

I wrote a python script to help solve this problem:

seat\_belt = .49

no\_seat\_belt = .51

seat\_belt\_injured = .46

seat\_belt\_killed = .27

seat\_belt\_unharmed = .27

no\_seat\_belt\_injured = .41

no\_seat\_belt\_killed = .52

no\_seat\_belt\_unharmed = .07

pct\_unharmed\_w\_sb = seat\_belt\*seat\_belt\_unharmed

pct\_unharmed\_wo\_sb = no\_seat\_belt\*no\_seat\_belt\_unharmed

prob\_unharmed = pct\_unharmed\_wo\_sb+pct\_unharmed\_w\_sb

print 'The probability of selecting an unharmed person from a fatal crash:'

print prob\_unharmed

prob\_unharmed\_wo\_sb = pct\_unharmed\_wo\_sb/prob\_unharmed

print 'The probability that a randomly selected person who was unharmed in a fatal crash\nwas not wearing a seat belt:'

print prob\_unharmed\_wo\_sb

OUTPUT:

The probability of selecting an unharmed person from a fatal crash:

0.168

**The probability that a randomly selected person who was unharmed in a fatal crash**

**was not wearing a seat belt:**

**0.2125**