MP-1 TUTORIAL-2

PRELAB

1. Write the algorithm to solve the simple simplex method. What is the usage of slack variables

MP-1 Lab-2 190031187
prelab Radhakrishna

1. Algorithm for simplex method

step-1 :- check whether objective function is to minimized or maximized. If it is to be minimized then we convert it into a problem of maximizing it by using result Minimum $z = -$ Maximum $(-z)$.

step-2 :- check whether all b_i $(i=1,2\ldots m)$ are non-negative, If any one of b_i is negative then multiply the corresponding eqn of the constraints by -1.

step-3 :- convert all the inequations of the constraints into equations by introducing slack / surplus and artificial variables in the constraints

step-4 :- obtain an initial basic feasible solution by setting $x_1 = x_2 = \ldots = x_n = 0$. in equations obtained in step-3

step-5 :- Determine which variable to enter into solution next. Identify the column - hence the variable - with the largest +ve number in the $c_j - z_j$ row of the previous tableau.

Scanned with CamScanner

Step-6 : Determine which variable to replace

Step-7 : Compute new values for the pivot row

In pivot row each value is $\frac{\text{previous value}}{\text{pivot}}$

In other rows each new value is

previous value $-$ (previous value in pivot column $*$ New value in pivot row)

Step-8 :- compute the z_j and $c_j - z_j$ rows as demonstrated in the initial tableau. If all numbers in $c_j - z_j$ rows are zero or negative. we have found an optimal solution. If this is not the case, we must return to step 5.

Usage of slack variable

A slack variable is a variable that added to an inequality constraint to transform it into an equality.

2. Consider the following linear programming problem
   Maximize:
   $P = 7x + 12y$
   Subject to:
   $2x + 3y \leq 6$
   $3x + 7y \leq 12$
   SetUp the Initial Simplex Tableau.

2. Maximize: $P = 7x + 12y$

Subject to: $2x + 3y \le 6$

$3x + 7y \le 12$

Adding slack variables

$2x + 3y + S_1 = 6$

$3x + 7y + S_2 = 12$

For instance suppose that

$x = 1, y = 1$ then $S_1 = 1$

$S_2 = 2$

Scanned with CamScanner

Initial simplex tableau

C_{Bi}	C_j	7	12	0	0	
	Basic variable	x	y	S_1	S_2	sol
0	S_1	2	3	1	0	6
0	S_2	3	7	0	1	12
	Z_j	0	0	0	0	0
	$C_j - Z_j$	7	12	0	0	

190031187 NERELLA VENKATA RADHAKRISHNA

INLAB

1. A hotel has requested a manufacturer to produce pillows and blankets for their room service. For materials, the manufacturer has 750 m^2 of cotton textile and 1,000 m^2 of silk. Every pillow needs 2 m^2 of cotton and 1 m^2 of silk. Every blanket needs 2 m^2 of cotton and 5 m^2 of silk. The price of the pillow is fixed at $5 and the blanket is fixed at $10.What is the number of pillows and blankets that the manufacturer must give to the hotel so that these items obtain a maximum sale? Formulate and Solve using Python.

Inlab :-

1. let x = no.of pillows
y = no.of blankets

Maximize $z = 5x + 10y$ (objective func)

subject To $2x + 2y \leq 750$
$x + 5y \leq 1000$
$x, y \geq 0$

using slack variables

$2x + 2y + s_1 = 750$
$x + 5y + s_2 = 1000$

Objective function

Max $z = 5x + 10y + 0s_1 + 0s_2$

Scanned with CamScanner

Code:

```
import numpy as np
import scipy as sp
#create the matrix for objective function and constraints
z=np.array([-5,-10])
a=np.array([[2,2],[1,5]])
b=np.array([750,1000])
x1_bound=(0,None)
x2_bound=(0,None)
from scipy.optimize import linprog
#Required command to get result
result=linprog(z,A_ub=a,b_ub=b,bounds=[x1_bound,x2_bound])
print result
```

OUTPUT

jupyter Inlab2_1 Last Checkpoint: 14 minutes ago (autosaved)

In [1]:
```
import numpy as np
import scipy as sp
#create the matrix for objective function and constraints
z=np.array([-5,-10])
a=np.array([[2,2],[1,5]])
b=np.array([750,1000])
x1_bound=(0,None)
x2_bound=(0,None)
from scipy.optimize import linprog
#Required command to get result
result=linprog(z,A_ub=a,b_ub=b,bounds=[x1_bound,x2_bound])
print result
```

```
     con: array([], dtype=float64)
     fun: -2656.25
 message: 'Optimization terminated successfully.'
     nit: 2
   slack: array([0., 0.])
  status: 0
 success: True
       x: array([218.75, 156.25])
```

2. Kiran owns a petroleum company, which consists of two refineries. Refinery 1 costs \$32,000 per day to operate, and it can produce 455 barrels of high-grade oil, 280 barrels of medium-grade oil, and 232 barrels of low-grade oil each day. Refinery 2 is newer and more modern. It costs \$40,000 per day to operate, and it can produce 350 barrels of high-grade oil, 438 barrels of medium-grade oil, and 522 barrels of low-grade oil each day. The company has orders totaling 25,625 barrels of high-grade oil, 24,992 barrels of medium-grade oil, and 29,568 barrels of low-grade oil. How many days should it run each refinery to minimize its costs and still refine enough oil to meet its orders? Formulate the problem and solve using Python.

2. let x_1 be Refinery 1
x_2 be Refinery 2.

Objective function is

$$C = 32000\, x_1 + 40000\, x_2$$

Subject To:

$$455 x_1 + 350 x_2 \geq 25625$$
$$280 x_1 + 438 x_2 \geq 24992$$
$$232 x_1 + 522 x_2 \geq 29568$$
$$x_1, x_2 \geq 0$$

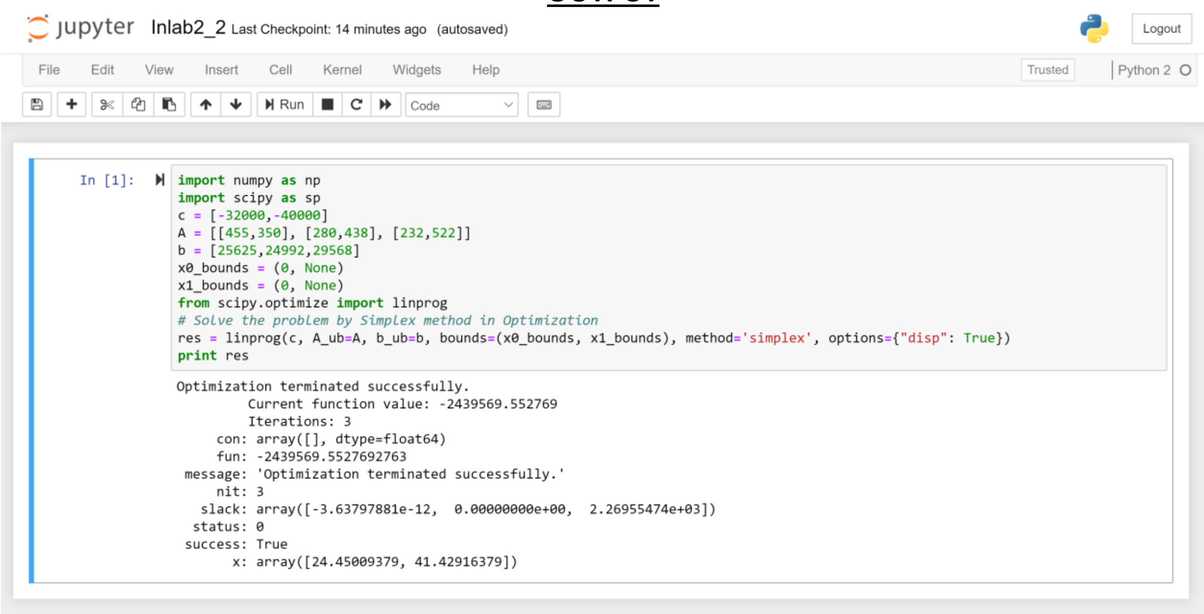
Code:

import numpy as np

import scipy as sp

```
c = [-32000,-40000]
A = [[455,350], [280,438], [232,522]]
b = [25625,24992,29568]
x0_bounds = (0, None)
x1_bounds = (0, None)
from scipy.optimize import linprog
# Solve the problem by Simplex method in Optimization
res = linprog(c, A_ub=A, b_ub=b, bounds=(x0_bounds, x1_bounds), method='simplex',
options={"disp": True})
print res
```

OUTPUT



POSTLAB

1. Maximize Z=3 x1 + 5 x2
Subject TO:

3 x1 + 2 x2 =18
X1 <= 4
2 x2 <= 12
X1 >=0
X2 >= 0

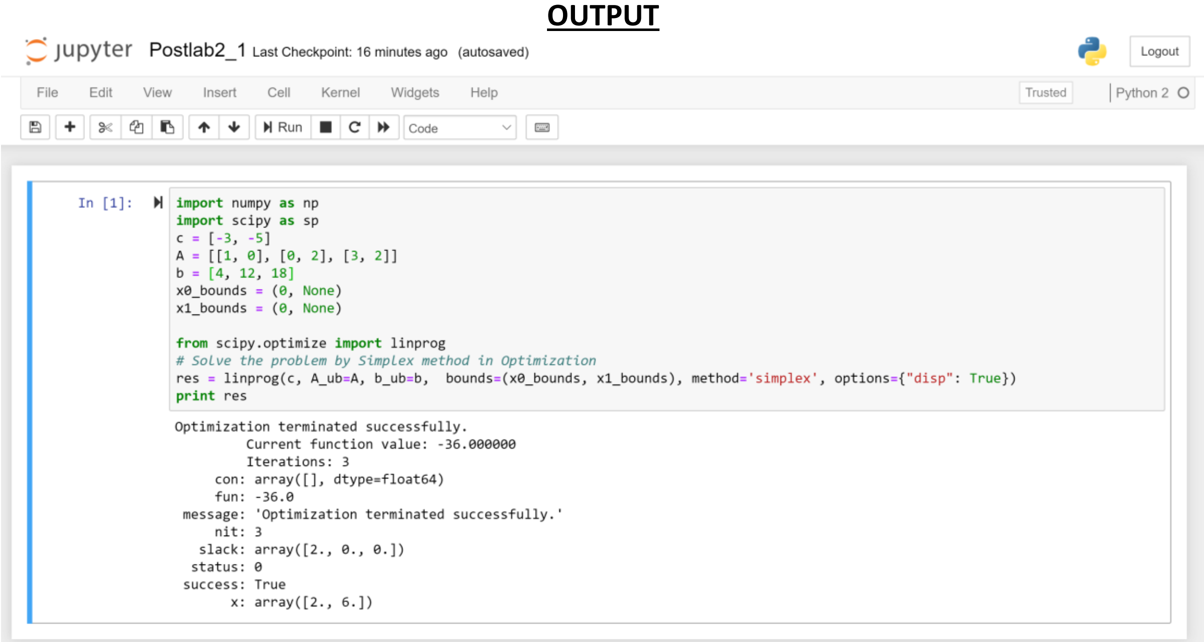
Solve LP using simplex method using Python

Code:

```
import numpy as np
import scipy as sp
c = [-3, -5]
A = [[1, 0], [0, 2], [3, 2]]
b = [4, 12, 18]
x0_bounds = (0, None)
x1_bounds = (0, None)

from scipy.optimize import linprog
# Solve the problem by Simplex method in Optimization
res = linprog(c, A_ub=A, b_ub=b,  bounds=(x0_bounds, x1_bounds), method='simplex', options={"disp": True})
print res
```

OUTPUT




```
import numpy as np
import scipy as sp
c = [-3, -5]
A = [[1, 0], [0, 2], [3, 2]]
b = [4, 12, 18]
x0_bounds = (0, None)
x1_bounds = (0, None)

from scipy.optimize import linprog
# Solve the problem by Simplex method in Optimization
res = linprog(c, A_ub=A, b_ub=b,  bounds=(x0_bounds, x1_bounds), method='simplex', options={"disp": True})
print res
```

```
Optimization terminated successfully.
         Current function value: -36.000000
         Iterations: 3
     con: array([], dtype=float64)
     fun: -36.0
 message: 'Optimization terminated successfully.'
     nit: 3
   slack: array([2., 0., 0.])
  status: 0
 success: True
       x: array([2., 6.])
```

2. Solve Linear Programming Model Using Simplex Method
Maximize: P=40x1+35x2
Subject To: x1 +x2 <=24

3x1 +2x2<=60
X1,x2 >=0

post - Lab

2. Maximize : $P = 40x_1 + 35x_2$

subject To : $x_1 + x_2 \leq 24$

$3x_1 + 2x_2 \leq 60$

$x_1, x_2 \geq 0$

Adding slack variables

$x_1 + x_2 + s_1 = 24$

$3x_1 + 2x_2 + s_2 = 60$

objective function $p = 40x_1 + 35x_2 + 0s_1 + 0s_2$

Initial table

CB_i	c_j	40	35	0	0		
	Basic variable	x_1	x_2	s_1	s_2	sol	Ratio min
0	s_1	1	1	1	0	24	24
0	s_2	3	2	0	1	60	20
	z_j	0	0	0	0		
	$c_j - z_j$	40	35	0	0		

Iteration - 1

C_{B_i}	C_j	40	35	0	0		
	Basic variable	x_1	x_2	s_1	s_2	sol	
0	s_1	0	1/3	1	-1/3	4	12
40	x_1	1	2/3	0	1/3	20	30
	z_j	40	80/3	0	40/3		
	$C_j - z_j$	0	25/3	0	-40/3		

Iteration - 2

C_{B_i}	C_j	40	35	0	0	
	Basic variables	x_1	x_2	s_1	s_2	sol
35	x_2	0	1	3	-1	12
40	x_1	1	0	-2	1	12
	z_j	40	35	25	5	
	$C_j - z_j$	0	0	-25	-5	

$\therefore \; x_1 = 12 \quad x_2 = 12$

$P = 40(12) + 35(12)$

$= 480 + 420$

$P = 900$