OPER 527 Sept 13 Classwork Code

Christopher Flippen, Sahil Chindal, Eunice Clark, Richard Foster

9/13/2022

```
import itertools

A = [[3, 2, 5],
     [2, 1, 1],
     [1, 1, 3],
     [5, 2, 4],
     [-1, 0, 0],
     [0, -1, 0],
     [0, 0, -1]]

B = [[55],
     [26],
     [30],
     [57],
     [0],
     [0],
     [0]]

# the max and the x vector which achieves the max are stored as \
   maxVal and maxX
maxVal = 0
maxX = []
results = []
# iterate through all the 7 choose 3 equations by using the \
   combinations function
for currentIndexSet in itertools.combinations([0..6],3):
    Arows = []
    Brows = []
    # store our current equations in Arows and Brows
    for i in currentIndexSet:
        Arows.append(A[i])
        Brows.append(B[i])
    # store these rows as matrices
    Amat = matrix(Arows)
```

```
    Bmat = matrix(Brows)
    Ainv = Amat.inverse()
    # x = A^-1 * B
    xVec = Ainv*Bmat
    # split x into its components
    x1 = xVec[0][0]
    x2 = xVec[1][0]
    x3 = xVec[2][0]
    # compute P(x)
    Px = 20*x1 + 10*x2 + 15*x3
    # store x and P(x) in the results list
    results.append((x1,x2,x3,Px))

# sort the results in descending order sorted by the P(x) value
sortedResults = sorted(results,key=lambda t:-t[3])

# printing sortedResults
from pprint import pprint as pp
print(f'The corner points listed in the form (x1, x2, x3, P(x)):')
pp(sortedResults)

# examine the results in descending order of P(x) value
for r in sortedResults:
    i = 0
    failed = False
    x1 = r[0]
    x2 = r[1]
    x3 = r[2]
    # keep looking at the current x point until it either passes all\
    the constraints or fails one
    while not failed and i < len(A):
        # evaluate row i of matrix A with the current x
        evalRow = A[i][0]*x1 + A[i][1]*x2 + A[i][2]*x3
        # if evalRow > B[i], then the current x does not satisfy the\
    constraints
        if evalRow > B[i][0]:
            failed = True
        i += 1
    # if the current point did not fail the constraints,
    # then it must be the maximizing point because the list was \
   iterated through in descending order
    if not failed:
        # print the best point and stop running
        print(f'The maximum x value which satisfies the constraints \
   is {x1, x2, x3} which gives value P(x) = {r[3]}')
        break
```

The corner points listed in the form (x1, x2, x3, P(x)):

[(30, 0, 0, 600),
 (0, 0, 26, 390),
 (55/3, 0, 0, 1100/3),
 (0, 30, 0, 300),
 (0, 65/2, -2, 295),
 (48/5, 0, 34/5, 294),
 (-1, 31, 0, 290),
 (0, 57/2, 0, 285),
 (75/7, 0, 32/7, 1980/7),
 (1, 26, 0, 280),
 (0, 51/2, 3/2, 555/2),
 (0, 55/2, 0, 275),
 (0, 47/2, 5/2, 545/2),
 (1/2, 91/4, 9/4, 1085/4),
 (0, 24, 2, 270),
 (9/5, 104/5, 8/5, 268),
 (0, 25, 1, 265),
 (-3, 32, 0, 260),
 (-4, 34, 0, 260),
 (5, 16, 0, 260),
 (0, 26, 0, 260),
 (13, 0, 0, 260),
 (-6, 39, -1, 255),
 (-5, 35, 0, 250),
 (47/3, 0, -16/3, 700/3),
 (57/5, 0, 0, 228),
 (0, 15, 5, 225),
 (5, 0, 8, 220),
 (51/11, 0, 93/11, 2415/11),
 (0, 0, 57/4, 855/4),
 (15/4, 0, 35/4, 825/4),
 (7, -13, 12, 190),
 (0, 0, 11, 165),
 (0, 0, 10, 150),
 (0, 0, 0, 0)]
The maximum x value which satisfies the constraints is (9/5, 104/5, 8/5) which gives value P(x) = 268