NCERT-6.5.14

EE24BTECH11065 - Spoorthi yellamanchali

Question:
Find two positive numbers x and y such that $x + y = 60$ and xy^3 is maximum.

Theoretical Solution:
From the question, we can write,

$$y = 60 - x \tag{0.1}$$

On substituting equation (0.1) in xy^3, we get,

$$x\,(60 - x)^3 \tag{0.2}$$

On differentiating equation (0.2) with respect to x and equating it to zero, we get,

$$(60 - x)^3 - 3x\,(60 - x)^2 = 0 \tag{0.3}$$

$$(60 - x)^2\,(60 - 4x) = 0 \tag{0.4}$$

From the equation (asd), we get , $x = 60$ and $x = 15$,
On differentiating equation (asd), with respect x, and substituting both values of x, we get,

$$-3\,(60 - x)^2 - 3\,(60 - x)^2 + 6x\,(60 - x) \tag{0.5}$$

$$\therefore \frac{d^2y}{dx^2} = (60 - x)\,(-360 + 12x) \tag{0.6}$$

On substituting $x = 15$, we get ,

$$\frac{d^2y}{dx^2} < 0 \tag{0.7}$$

that means, xy^3 is maximum when $x = 15$, then , $y = 60 - 15 = 45$
$\therefore$ xy^3 is maximum when $x = 15$ and $y = 45$.

Solution using gradient descent method.
The gradient descent(in case of finding minima) or the gradient ascent method(in case of finding maxima) is a computational algorithm which optimizes and maximises/minimizes the functional curve.
It uses the concept of gradient or slope to do so as we know the fact that slope or gradient is zero or almost negligible at points of maxima and minima.
It works by itertively adjusting the input variable x in the direction of function's gradient. starting from initial guess x_0,the algorithm iteratively updates x using the rule

$$x_{new} = x_{current} + \alpha \times f'(x_{current}) \tag{0.8}$$

Here α is the learning rate which controls the size of each step.
The iteration stops when the change in x between iterations becomes smaller than a predefined tolerance.
We know that, in our question,

$$f'(x) = -3x(60-x)^2 + (60-x)^3 \tag{0.9}$$

On substituing equation (0.9) in (0.8) , we get,

$$x_{new} = x_{current} + \alpha\left(-3x_{current}(60-x_{current})^2 + (60-x_{current})^3\right) \tag{0.10}$$

and by taking,

$$\alpha = 0.000001 \tag{0.11}$$

$$\tag{0.12}$$

And convergence = $1e-6$
initial guess $x_0 = 14.0$, We get,
x value at maxima = 14.999878397267214
y value at maxima = 45.00012160273279
Solution using geometric programming:
Geometric programming is a type of mathematical optimization that is used for problems where the objective and constraints can be expressed as posynomials or monomials.

A monomial is a product of variables raised to constant powers. A monomial is of the form: ${x_1}^{a_1}{x_2}^{a_2}{x_3}^{a_3}.....{x_n}^{a_n}$
where $x_1, x_2, x_3.....x_n$ are the variables and $a_1, a_2, a_3....a_n$ are the real constant powers(can be positive or negative or zero.)

posynomial is a sum of monomials, where each monomial in the sum has non-negative coefficients. A posynomial is of the form:

$$f(x_1, x_2, x_3..., x_n) = c_1{x_1}^{a_{11}}{x_2}^{a_{12}}...{x_n}^{a_{1n}} + c_2{x_1}^{a_{21}}{x_2}^{a_{22}}....{x_n}^{a_{2n}} + ... + c_n{x_1}^{a_{n1}}{x_2}^{a_{n2}}...{x_n}^{a_{nn}} \tag{0.13}$$

Where $c_i \geq 0$ (non-negative constants)
The equation above is the expression of the objective function which we wish to maximise or minimise.

The constraints of the problem are typically posynomials, and they are usually represented as:

$$g_i(x_1, x_2, x_3....x_n) \leq 0; i = 1, 2, 3...m. \tag{0.14}$$

$g_i(x_1, x_2, x_3....x_n)$ is a posynomial.

Maximization of a Posynomial Objective Function:

$$f(x_1, x_2, x_3..., x_n) = \sum c_i x_1^{a_{i1}} x_2^{a_{i2}} ... x_n^{in} \tag{0.15}$$

Subject to constraints of the form:

$$g_i(x_1, x_2, x_3....x_n) \leq 0; \;\; i = 1, 2, 3..., m \tag{0.16}$$

The objective function in our problem is

$$f(x, y) = xy^3 \tag{0.17}$$

To apply Geometric Programming, we take the logarithm of both the objective function and the constraint.
This step simplifies the problem and makes it easier to handle by converting products into sums.
For the objective $f(x, y) = xy^3$, we take the natural logarithm:

$$\ln xy^3 = \ln x + 3 \ln y \tag{0.18}$$

The constraint is $x + y = 60$.
Taking logarithm of this constraint directly is not meaningful, as it is a sum , not a product. we will simply proceed with it as it is since it's already linear and suitable for GP.
In order to combine objective and constraint we use lagrange multipliers
Lagrangian combines the objective function and the constraint by introducing a Lagrange multiplier λ for the constraint.
Lagrangian for an objective function $f(x, y)$ and constraint $g(x, y)$ is given by:

$$L(x, y, \lambda) = f(x, y) + \lambda (g(x, y)) \tag{0.19}$$

we can take our objective function to be $\ln x + 3 \ln y$ as , since *ln* is an increasing function, $\ln xy^3$ is maximized whenever xy^3 is maximized. $\therefore$ On applying lagrangian for equation (d) and (a), we get,

$$L(x, y, \lambda) = \ln x + 3 \ln y + \lambda (x - y - 60) \tag{0.20}$$

To find the optimal solution, we take the partial derivatives of the Lagrangian with respect to x, y, and λ, and set them equal to zero.

On partially derivating equation (a) with respect to x:

$$\frac{\partial L}{\partial x} = \frac{1}{x} - \lambda = 0 \tag{0.21}$$

$$\lambda = \frac{1}{x} \tag{0.22}$$

On partially derivating equation (a) with respect to y:

$$\frac{\partial L}{\partial y} = \frac{3}{y} - \lambda \tag{0.23}$$

$$\lambda = \frac{3}{y} \tag{0.24}$$

On partially derivating equation (a) with respect to λ:

$$\frac{\partial L}{\partial \lambda} = 60 - x + y = 0 \tag{0.25}$$

$$x + y = 60 \tag{0.26}$$

From the first two equations, we can equate λ:

$$\frac{1}{x} = \frac{3}{y} \tag{0.27}$$

Solving for y in terms of x, we get:

$$y = 3x \tag{0.28}$$

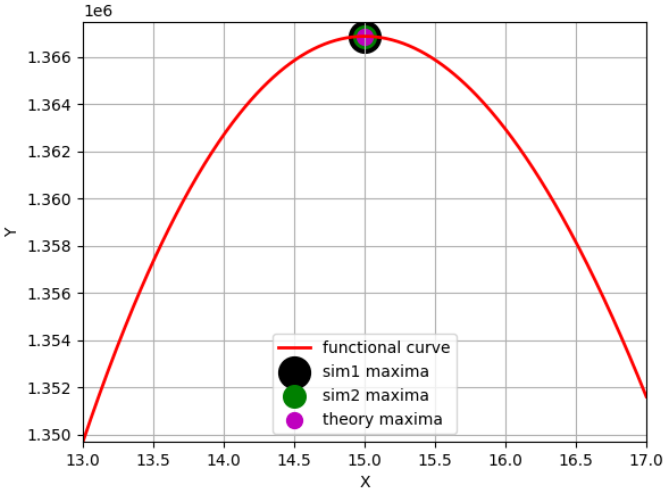
Now substitute $y = 3x$ into the constraint $x + y = 60$, then , we get,

$$x + 3x = 60 \tag{0.29}$$

$$x = 15 \tag{0.30}$$

$$y = 45 \tag{0.31}$$

$\therefore$ $x = 15$ and $y = 45$ at maxima. On solving for values of x and y using a python function , we get
value for x using geometric programming 14.99878409289183
value for y using geometric programming 45.00121536318684.
Due to floating-point rounding errors, the computed solution will be an approximation, leading to slight deviations.


1e6
1.366
1.364
1.362
1.360
1.358
1.356
1.354
1.352
1.350
Y
13.0
13.5
14.0
14.5
15.0
15.5
16.0
16.5
17.0
X
functional curve
sim1 maxima
sim2 maxima
theory maxima