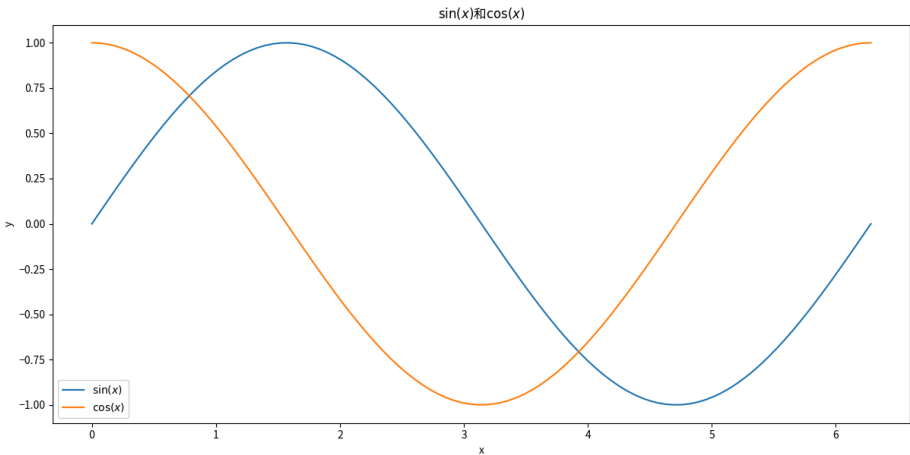


初等代数

一元函数绘图

绘制 $\sin(x)$ 和 $\cos(x)$ 函数在 $x \in [0, 2\pi]$ 的图像。

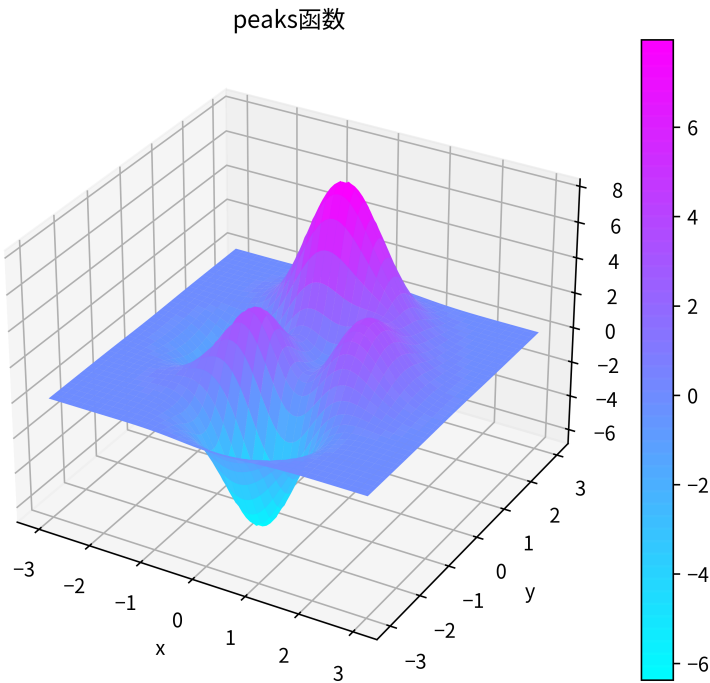


二元函数绘图

给定二元函数 $peaks(x, y)$ 如下：

$$peaks(x, y) = 3(1 - x)^2 e^{-x^2 - (y+1)^2} - 10(\frac{x}{5} - x^3 - y^5) e^{-x^2 - y^2} - \frac{1}{3} e^{-(x+1)^2 - y^2}$$

试绘制它在 $x, y \in [-3, 3]$ 的图像。



```

local plt = require('package.plot')
local image = require('package.image')
local mol = libminoptlab

local imgs = {}

local title1 = '$\\sin(x)$和$\\cos(x)$'
local funs= {math.sin,math.cos}
local labels = {'$\\sin(x)$', '$\\cos(x)$'}
local lb = {0,0}
local ub = {2*math.pi,2*math.pi}

local img1 = plt.fplot(funs,lb,ub,labels)
local fw1 = plt.framework(111,title1,{'x','y'})
plt.show({{fw1,{img1}}}, 'images/base1.png',false)
table.insert(imgs,{'images/base1.png',121,'一元函数绘图'})

local title2 = 'peaks函数'
local peaks = function(xx)
    local x, y = xx[1], xx[2]
    return
        3 * (1 - x) ^ 2 * math.exp(-(x ^ 2) - (y + 1) ^ 2) - 10 * (x / 5 - x ^ 3 - y ^ 5) * math.exp(-x ^ 2 - y ^ 2) - 1 /
        3 * math.exp(-(x + 1) ^ 2 - y ^ 2)
end
local lb = {-3, -3}
local ub = {3, 3}
local n = 80

local img2 = plt.fplot3d({peaks}, {lb}, {ub}, n, {{
    cmap = 'cool'
}})
local fw2 = plt.framework3d(111, title2, {'x', 'y', 'z'})
plt.show({{fw2,{img2}}}, 'images/base2.png',false)
table.insert(imgs,{'images/base2.png',122,'二元函数绘图'})

image.show(imgs)

```