# -\*- coding: utf-8 -\*-

"""

Created on Mon Jul 10 18:12:07 2023

@author: Sunggu

"""

import numpy as np

import matplotlib.pyplot as plt

def LinearExIn(xd,yd,x):

val = [];

for r in x:

idx = np.where(xd >= r)[0]

if len(idx) == 0:

idx = len(xd)-1

else:

idx = idx[0]

if idx == 0:

yv = 0.0;

xv = 0.0;

val.append(yd[idx])

else:

yv = yd[idx-1];

xv = xd[idx-1];

val.append( (r - xd[idx])\*(yd[idx] -yv)/(xd[idx] -xv) +yd[idx] )

return val

# Interest rate Curve set

tenors = np.array([0.25, 0.5, 1, 2, 3, 4, 5, 7, 10]);

rates = np.array([3.0, 3.4, 3.7, 4.0, 3.8, 4.2, 4.4, 4.7, 4.6])/100 # We s'pse that rates is given as zeros

x = np.linspace(0,10,100);

spots = LinearExIn(tenors, rates, x)

fwds = [spots[0]];

for i in range(1,len(x)):

# exp(spot[i]\*tenors[i]) = ...exp(tenors[i-1]\*fwds[i-1])\*exp(tenors[i]\*fwds[i])

fwds.append( np.log( np.exp(x[i]\*spots[i])\*np.exp(np.sum(-np.insert(np.diff(x[:i]),0,0.0)\*fwds[:i])) )/(x[i]-x[i-1]) );

# print( np.abs(np.exp(spots[i]\*x[i]) -np.exp(np.sum(np.insert(np.diff(x[:i+1]),0,0.0)\*fwds[:i+1]))) )

plt.figure(1)

plt.title("Curves")

plt.plot(x, spots, 'k-')

plt.plot(x, fwds, 'b-')

plt.legend(["spots", "fwds"])

plt.xlim(min(x),max(x));

plt.ylim(0, 1.2\*max(fwds));

# Fwd Contract Long under Stock

S0 = np.linspace(0,200,201);

K = 120;

T = 5;

idx = np.where(x >= T)[0][0]

payoff = S0\*np.exp(x[idx]\*spots[idx]) -K;

FWD = payoff\*np.exp(-x[idx]\*spots[idx])

plt.figure(2);

plt.plot(S0, FWD, 'k-')

plt.plot(S0, payoff, 'r-')

# IRS Fixed pay, float receive

T = 0.5

Notional = 10000;

fixed = np.mean(rates[:2])

x = np.linspace(0,0.5,3);

spots = LinearExIn(tenors, rates, x)

fwds = [spots[0]];

for i in range(1,len(x)):

# exp(spot[i]\*tenors[i]) = ...exp(tenors[i-1]\*fwds[i-1])\*exp(tenors[i]\*fwds[i])

fwds.append( np.log( np.exp(x[i]\*spots[i])\*np.exp(np.sum(-np.insert(np.diff(x[:i]),0,0.0)\*fwds[:i])) )/(x[i]-x[i-1]) );

# print( np.abs(np.exp(spots[i]\*x[i]) -np.exp(np.sum(np.insert(np.diff(x[:i+1]),0,0.0)\*fwds[:i+1]))) )

fixedleg = []; floatleg = [];

for i in range(len(spots)):

fixedleg.append(Notional\*fixed\*x[i]\*np.exp(-x[i]\*spots[i]))

floatleg.append(Notional\*fwds[i]\*x[i]\*np.exp(-x[i]\*spots[i]))

IRS = np.sum(floatleg) - np.sum(fixedleg)

print(IRS)