財務工程導論 HW1

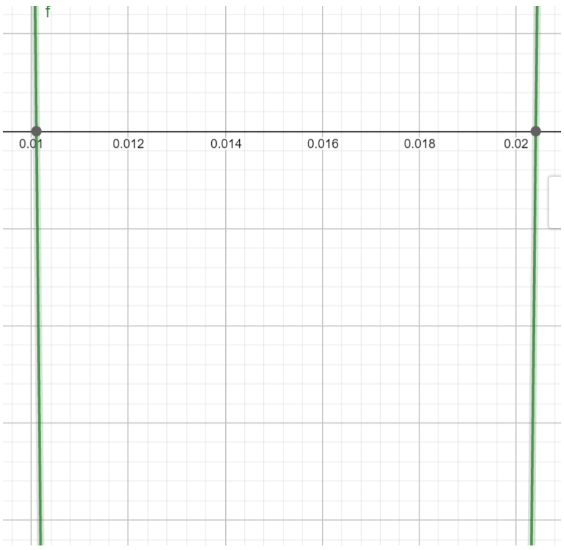
109550025 謝翔丞

Part1.

利用二分法(bisection method)雖然可以有效地朝特定值逼近，但是經由範例程式碼，我們會發現一旦 value > 0 時會將 low 的值往上調整，而這是建立在 f(x)是嚴格遞減函數的情況下才會成立。舉本題為例，有兩個可能的 IRR 值，且函數呈現開口向上的曲線，如果將 high 與 low 的值設為範例的 1 與 0，會造成 middle 在一開始就是 0.5，超過答案的 0.0101...與 0.02...，再加上此函數 x > 0.02…的部分 y 都呈現大於 0 且遞增的趨勢，一旦照著原本程式碼的 value > 0 就增加 low 的值，將會造成 low 不斷往 high 逼近，但實際上卻早已超過正確答案的值。

因此利用二分法解 IRR 的題目有以下幾點須注意:

1. high 與 low 的初始值需要謹慎設定
2. 當可能的 IRR 值超過一個時，如何正確地逼近與調整 high、low 值需要更嚴謹的判斷，if(value > 0)這條判斷式只限用於嚴格遞減函式。



Part2.

使用 newton method，可以較為精準的找到答案，不像二分法會因為 high、low 值的初始值而有所限制。但需要注意的是，以本題的答案 0.01…與 0.02…為例，當我們初始 x 值的 f’(x) < 0 時，他會找出較小的解，反之，則會逼近出較大者，言下之意就是只會找出一個答案，但都是屬於正解之一，而這都根據該函式的長相而有所差別。

//以上二分法和用來驗證的牛頓法都有附上 code