Find the root of .

1. Bisection method:

Step1.找出使得, 根據勘根定理我們能確保之間有根. 最簡單的方式是窮舉法很多函數值或是直接畫圖.

Step2.令

Step3.若則設,

若則設. 重複Step2~3直到

|  |  |  |
| --- | --- | --- |
|  | Iteration time | The root of f |
|  | 36 | 1.887208 |
|  | 35 | 1.887208 |
|  | 36 | -1.887208 |
|  | 35 | -1.887208 |

我們可以發現f 有兩個根,大約在: 1.887208 , -1.887208. 而bisection method基本上每次朝根的步伐會縮減一半.

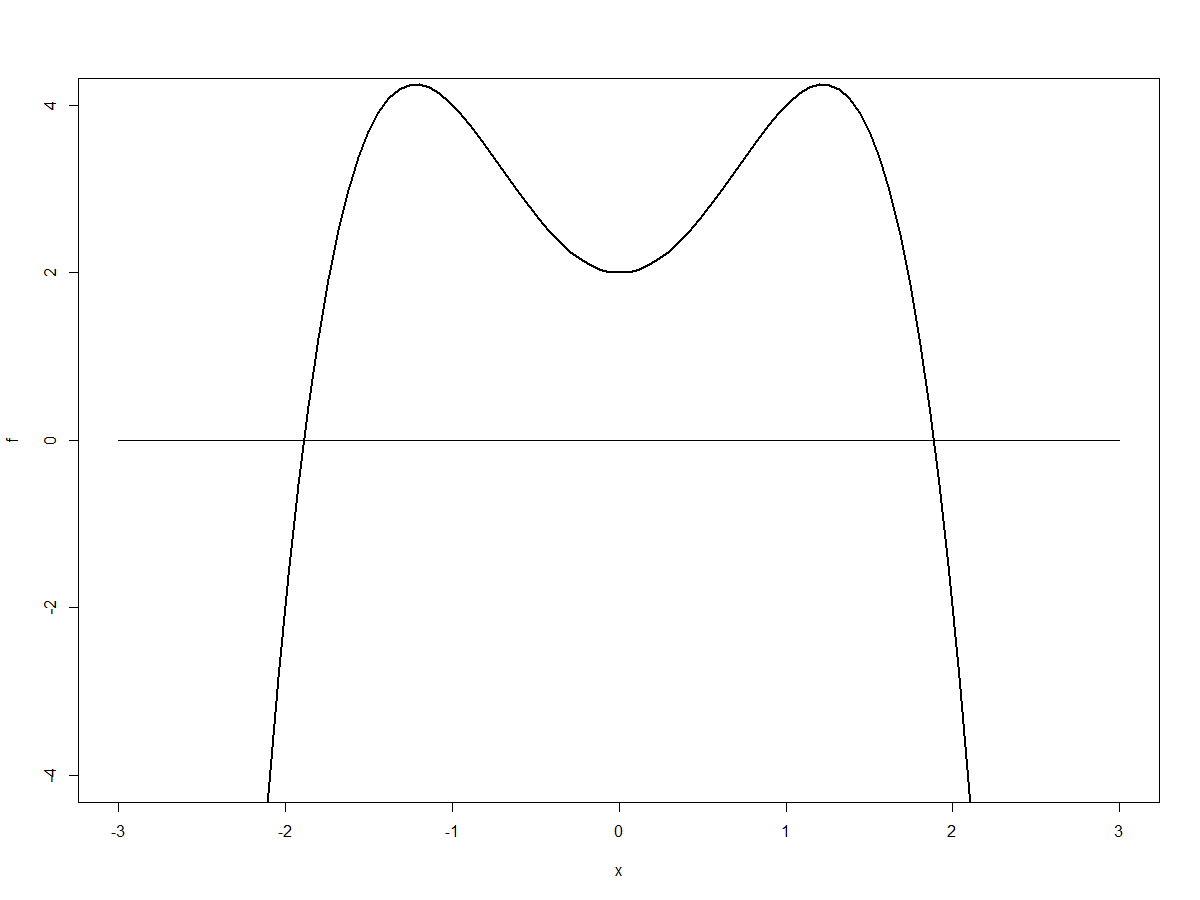
1. Newton method:

Step1. 選定初始值.

Step2. . 重複Step2直到

由牛頓法收斂的定理告訴我們，對於起始點必須要落在根的附近，如此我們的牛頓法才能夠找到根。

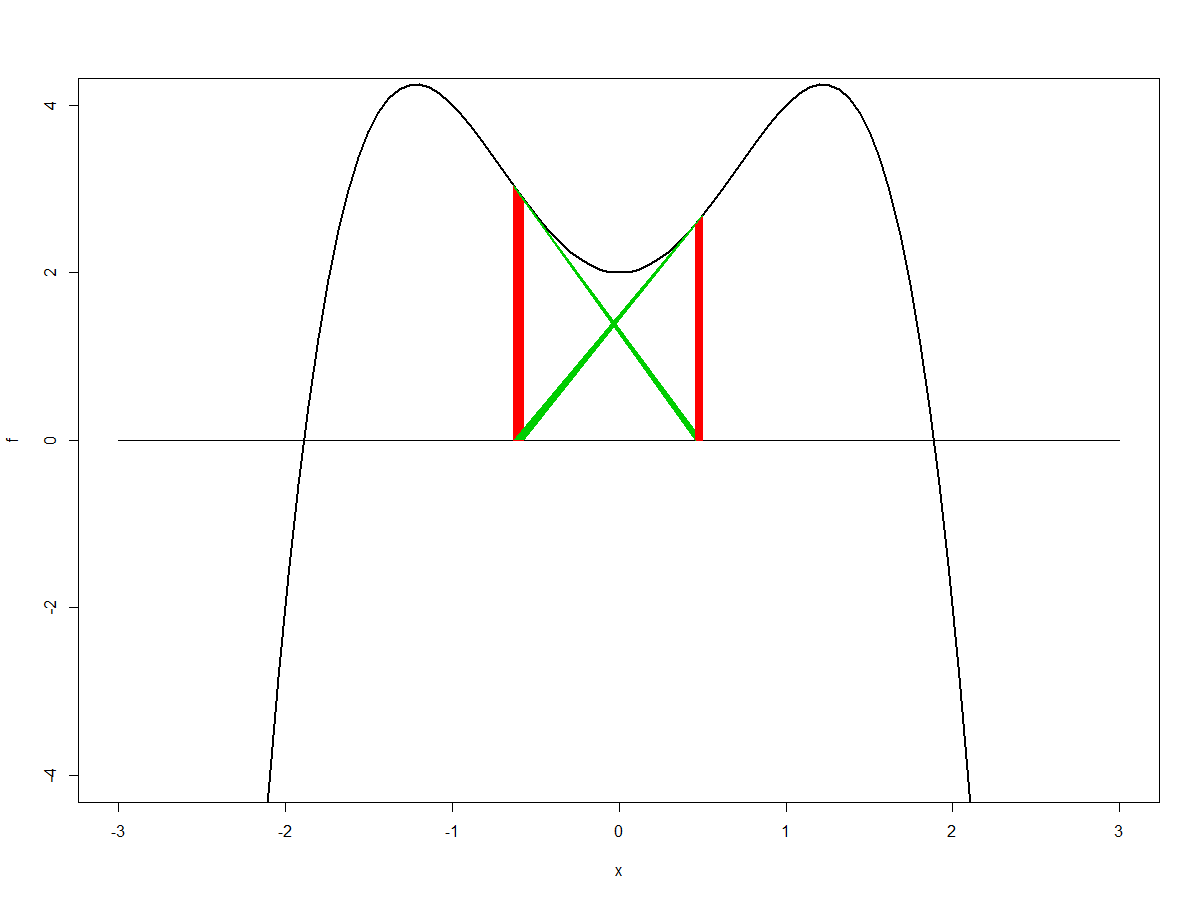
首先，畫初函數圖:



由於不確定能否收斂，設定最大疊代次數為500，我取 ,

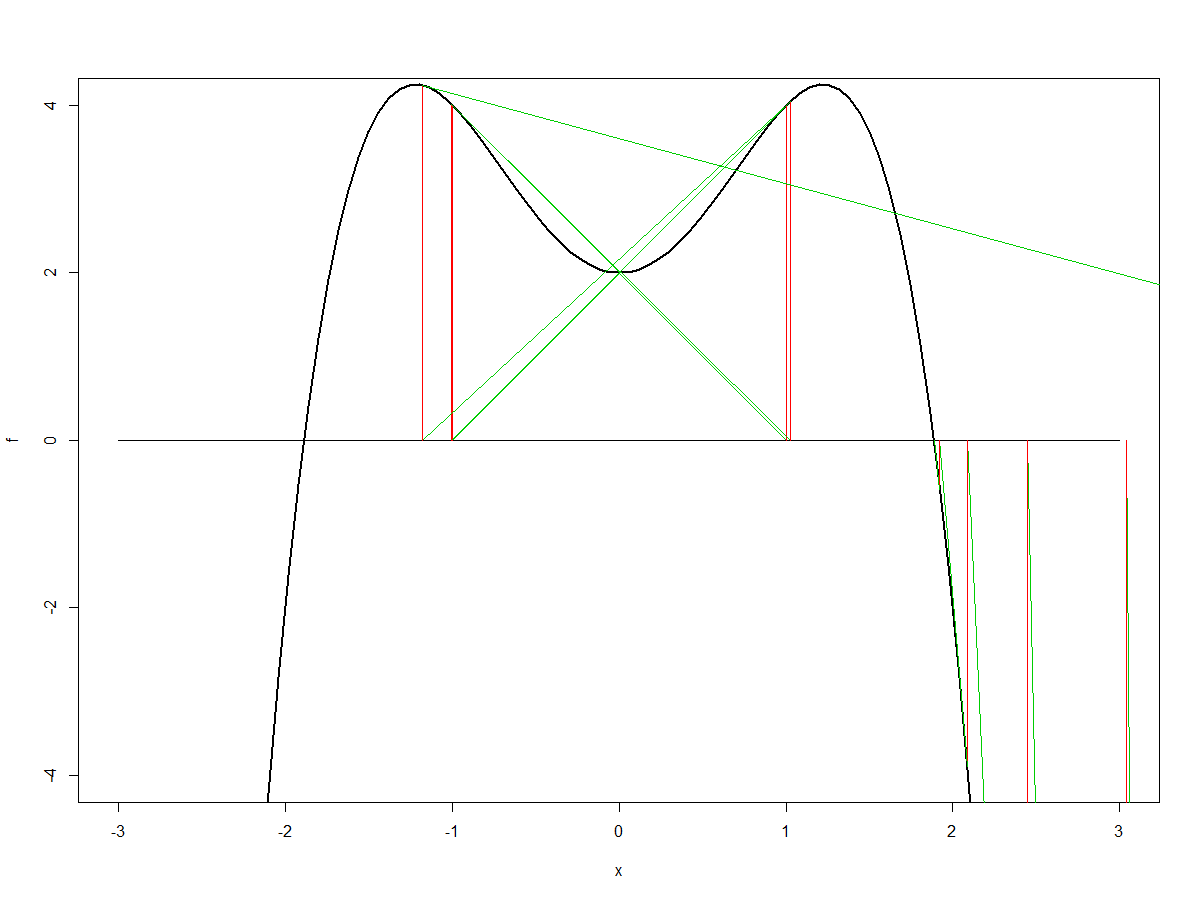
發現達到最大疊代次數仍無法收斂.

下圖紅色線為每一疊代點所對應的函數值; 綠色線為該點的切線

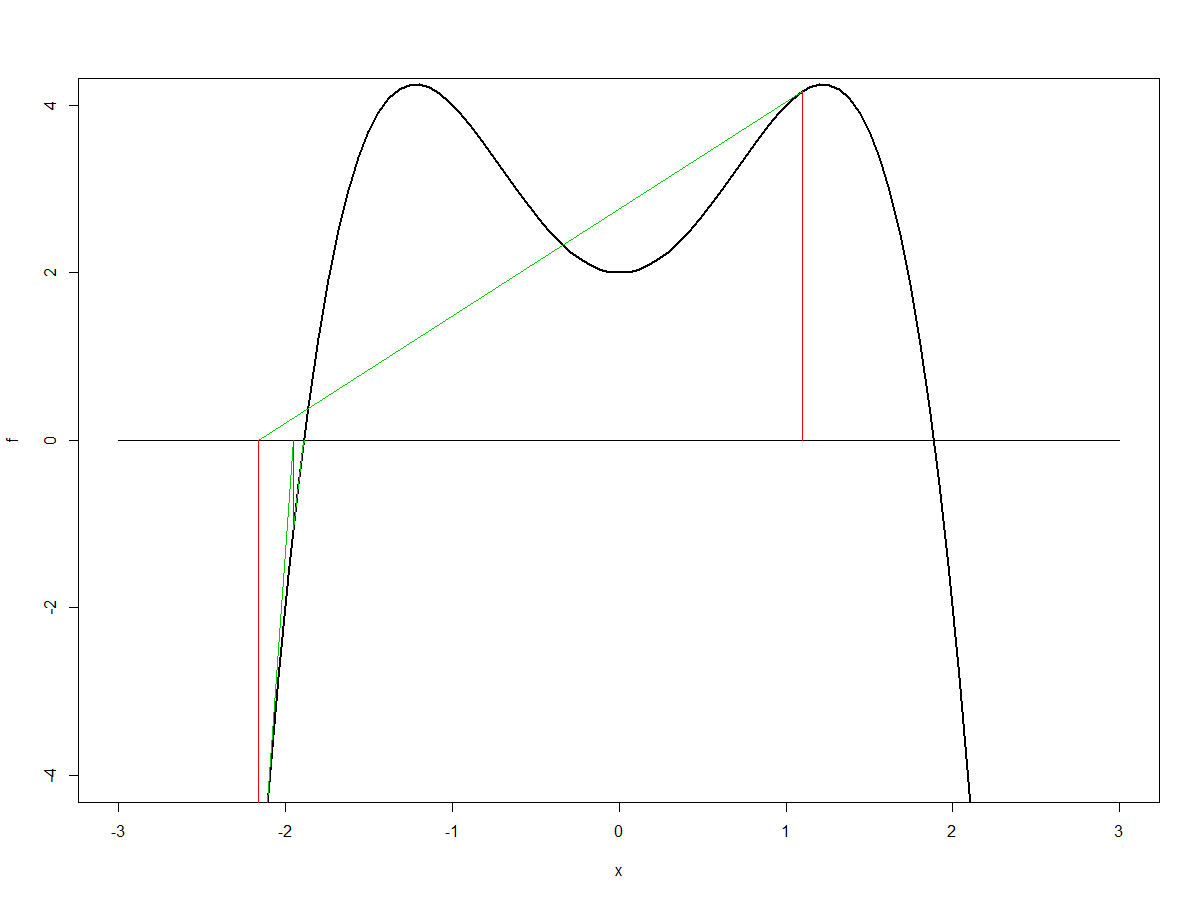


接著，我取,都和上面結果一樣無法收斂.

當 時, 經過18次疊代, 找到根為 1.887208



當 時, 經過5次疊代, 找到根為 -1.887208



經過測試後，我們發現當選擇作為起始點，牛頓法會收斂到根:1.887208或-1.887208.

而當選擇作為起始點，牛頓法無法收斂.

附上程式碼:

|  |
| --- |
| #newton method  f<-function(x){ -x^4+3\*x^2+2 }  plot(f ,xlim = c(-3,3) ,ylim = c(-4,4) ,lwd= 2) ;lines(c(-3,3),c(0,0))  epslon<-10^-9 ;h<-10^-8 ;loop<-0  seed<--0.5 ;x2<-seed  while(abs(f(x2))>=epslon && loop<500 ){  cat("the ",loop,"-th step, f(x) is ",abs(f(x2)),"\n")  loop<-loop+1  x1<-x2  dif<-(f(x1+h)-f(x1)) /h  x2<-x1 - f(x1)/dif  lines(c(x1,x1),c(0,f(x1)),col=2) ;lines(c(x1,x2),c(f(x1),0) ,col=3)  };cat("the root of f(x) is",x2)  g<-function(x){ x-(-x^4+3\*x^2+2)/(-4\*x^3+6\*x)}  windows();plot(g,xlim = c(-3,3), ylim=c(-4,4),lwd=2,type="p") ;lines(c(-3,3),c(1,1)) ;lines(c(-3,3),c(-1,-1)) ;lines(c(-3,3),c(-3,3),lwd=2)  x<-seq(0,2,0.001)  plot(x,abs((g(x+h)-g(x))/h), ylim = c(0,2))  x[abs((g(x+h)-g(x))/h) < 1]  #bisection  f<-function(x){ -x^4+3\*x^2+2 }  plot(f ,xlim = c(-3,3) ,ylim = c(-4,4) ,lwd= 2) ;lines(c(-3,3),c(0,0))  epslon<-10^-9 ;loop<-0  xl<--51 ;xr<--1 ;xm<-(xl+xr)/2  while(abs(xl-xr)>=epslon){  cat("the ",loop,"-th step, f(x) is ",abs(f(xm)),"\n")  loop<-loop+1  xm<-(xl+xr)/2  if(f(xl)\*f(xr)>0){  break  }else if(f(xl)\*f(xm)<0){  xr<-xm  }else if(f(xr)\*f(xm)<0){  xl<-xm }  };cat("the root of f(x) is",xm) |