1010602 YiWen Chen 2015.3.31

Secant method

● Introduction

The Secant method is very similar to the Newton's method. We can see Figure 1 that we have two points located at x_n and x_{n-1}. Use the slope of secant line,

$\frac{f(x_{n-1})-f(x_n)}{x_{n-1}-x_n} = \frac{f(x_n)}{x_n-x_{n+1}}$, and then $x_{n+1} = x_n - f(xk) * \frac{x_n-x_{n-1}}{f(x_n)-f(x_{n+1})}$.

For optimization problem,

$x_{n+1} = x_n - f'(xk) * \frac{x_n-x_{n-1}}{f'(x_n)-f'(x_{n+1})}$

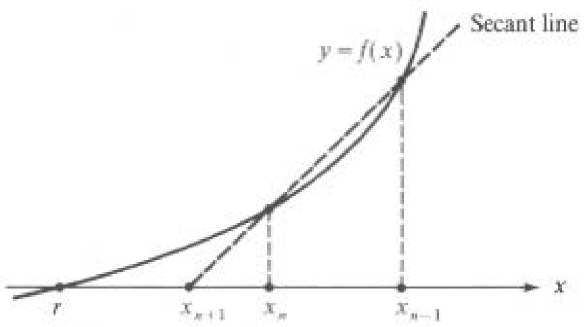

Figure 1the Secant method

● Compare the Secant method with the Newton's method.

➢ **An example is f(x) = $\frac{x^2}{2}$ - sin(x).**

```
subplot(1,2,1);      t = -5:0.01:5;            %plot problem on left
plot(t,(t.^2)/2-sin(t));
subplot(1,2,2);      t = -5:0.01:5;            %plot problem on right
hold on;   plot(t,(t.^2)/2-sin(t));
times = 15;                                    %iteration number
xk = 1.5;                                      %initial point
xi = 3;                                        %point k-1 for secant method
subplot(1,2,1);      hold on;                  %plot initial point on left
plot(xk,(xk.^2)/2-sin(xk),'kx')
subplot(1,2,2);      hold on;                  %plot initial point on right
plot(xk,(xk.^2)/2-sin(xk),'kx')
hold on;   plot(xi,(xi.^2)/2-sin(xi),'kx')     %plot k-1 point on right
```

```
syms x;
g = (x.^2)/2-sin(x);                        %problem function
g1 = diff(g);                               %find first differential equation
g2 = diff(g1);                              %find second differential equation

new = [0];                                  %save values from newton’s
sec = [0];                                  %save values from secant
i = 1;
for i=1:times;                              %for times of loop
    x=xk;                                   %substitute variable
    value1=subs(g1);                        %get first differential value
    value2=subs(g2);                        %get second differential value
    x  =xi;                                 %substitute variable
    value3=subs(g1);                        %get first differential value

    xk = xk - (value1/value2);              %newton's method
    new(i) = xk;
    subplot(1,2,1);     hold on;            %plot point on left
    plot(xk,(xk.^2)/2-sin(xk),'rx')

    xk = xk - value1*(xk-xi)/(value1-value3);     %secant method
    sec(i) = xk;
    subplot(1,2,2);     hold on;            %plot point on right
    plot(xk,(xk.^2)/2-sin(xk),'rx')
 end
```

By using iteration number = 15, initial point = 1.5 and x_{k-1} = 2, I got Figure 2 as my output. The minimizer from newton’s method is at x = 0.7392. The minimizer from secant method is at x = 0.7603.

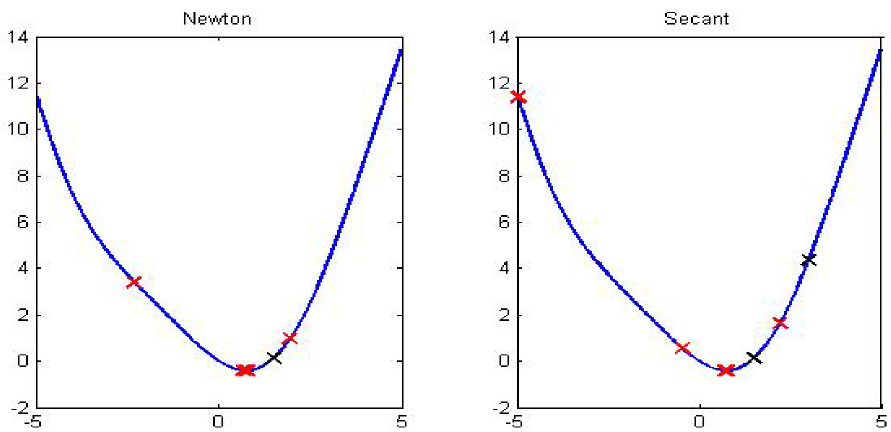

Figure 2output

- **Calculate their difference between current and last point.**

```
for i=1:times
    m(i) = i;
    if i<times
        new2(i) = abs(new(i+1) - new(i));        %save value for newton's
    else
        new2(times) = new2(times-1);
    end
end
plot(m,new2);                                    %plot newton's
for i=1:times
    if i<times
        sec2(i) = abs(sec(i+1) - sec(i));        %save value for secant
    else
        sec2(times) = sec2(times-1);
    end
end
hold on;plot(m,sec2,'k');                        %plot secant
```

In Figure 3, they finished converging after iteration times was 4. We can see that the Secant method is more quickly than the Newton's method.

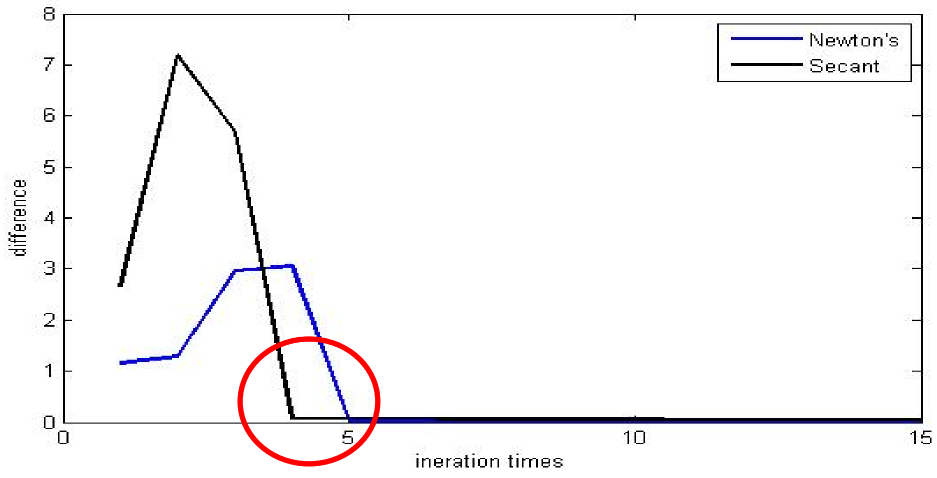

Figure 3output for difference

Last but not least, the most different is the Newton's method only needs one initial value but needs the function is twice differentiable and the Secant method needs two initial value but only needs once differentiable.