Computer Science Department
Bogor Agricultural University
http://cs.ipb.ac.id/

Pertemuan 2

# Teknik Pencarian Akar pada Persamaan Non-linier

Komputasi Numerik (KOM325)

# Review materi pertemuan sebelumnya

- Apa itu error ?
- Sebutkan sumber-sumber error !
- Apa yang dimaksud dengan Loss of Significance Error ?
- Apa yang dimaksud dengan propagasi error ?

# Subtopik

- Metode dalam pencarian akar pada persamaan non-linier meliputi:
  – Metode Bisection,
  – Metode Newton,
  – Metode Secant,
  – Metode Iterasi satu titik
- Studi kasus pencarian akar pada persamaan non-linier secara numerik
- Algoritme pencarian akar pada persamaan non-linier

# Pendahuluan

- Masalah pencarian akar yang lazim disebut akar persamaan adalah mencari solusi persamaan sebuah fungsi yang membuat **nilai-nilai nol.**
- **Misal: 2x-3=0**

  Solusi adalah memindahkan nilai -3 ke ruas kanan sehingga dihasilkan 2x=3, dan hasilnya x=3/2
- $\boldsymbol{x^2 - 4x - 5 = 0}$

  Solusi adalah yang membuat nilai-nilai nol
- Umumnya permasalahan akar muncul dalam bentuk non linear yang melibatkan fungsi-fungsi trigonometri, eksponensial, logaritma, transeden dan lain-lain

# Pendahuluan

- Ingat kembali persamaan berikut dalam kalkulus

$$f(x) = ax^2 + bx + c = 0 \qquad (1)$$

- Penyelesaian fungsi tersebut diperoleh dengan menggunakan rumus berikut

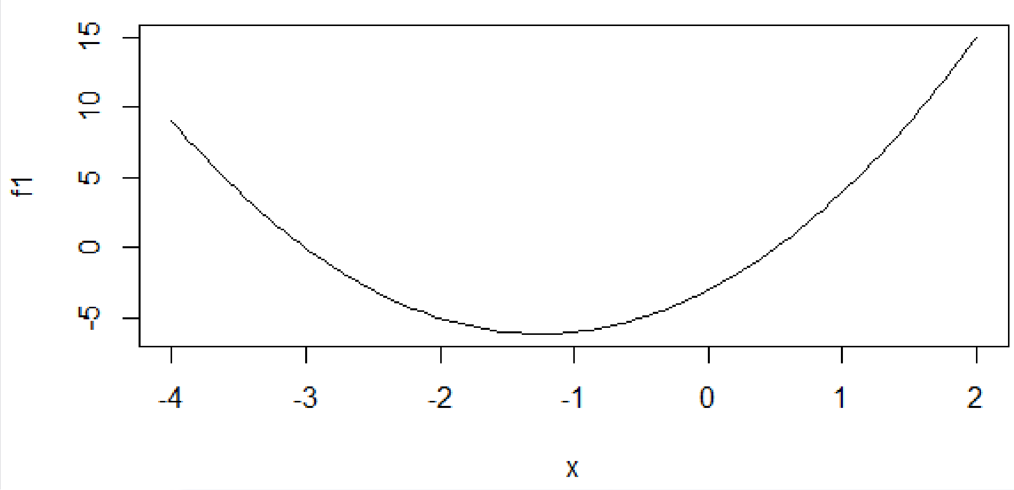
$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \qquad (2)$$

- $x_1$ dan $x_2$ adalah akar dari persamaan kuadratik (1). Akar tersebut menggambarkan nilai-nilai *x* yang membuat persamaan (1) sama dengan nol.
- Dengan demikian akar dari suatu persamaan merupakan nilai *x* yang memenuhi *f*(*x*) = 0

# Pendahuluan

- Contoh: tentukan akar dari $2x^2 + 5x - 3 = 0$
- Akar dari persamaan kuadratik tersebut adalah $x_1$ = 0.5 dan $x_2$ = -3.

```
> f1 <- function(x) 2*x^2+5*x-3
> plot(f1,-4, 2)
```



# Pendahuluan

Teorema:

Misal *f*(*x*) adalah fungsi kontinu. Setiap bilangan *r* dimana *f*(*r*) = 0 dinamakan akar dari persamaan *f*(*x*) = 0. Juga dikatakan bahwa *r* adalah titik nol persamaan. Persamaan (2) cukup sesuai digunakan untuk mencari akar persamaan kuadrat seperti dalam bentuk (1).

Bagaimana dengan fungsi non-linier berikut? Apakah bisa diselesaikan dengan persamaan (2)?

$$f(x) = e^{-x} - 3x$$

$$f(x) = x\sin(x) - 1$$

$$f(x) = \ln(x) - 5 + x$$

# Pendahuluan

- Dalam metode numerik, pencarian akar $f(x) = 0$ dilakukan secara iteratif dengan menelusuri skuens nilai x yang mungkin dan menyebabkan solusinya adalah nol
- Banyak algoritme untuk pencarian akar secara numerik dan dikelompokan ke dalam dua metode:
  - Metode Tertutup: Bisection, regula falsi
  - Metode Terbuka: Fixed point, Newthon-Rahpson dan Secant

# Metode Tertutup

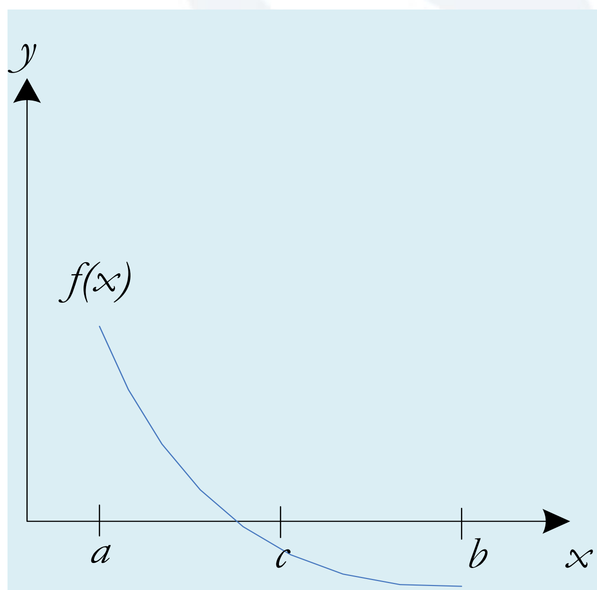
- Sering disebut dengan metode konvergen. Metode yang mencari akar dari suatu persamaan/fungsi dengan selang tertutup antara titik a dan b atau [a, b]. Sudah dipastikan berisi minimal 1 akar. Dengan kata lain iterasinya konvergen menuju 1 akar.
- Dalam sebuah selang pencarian akar tergantung untuk karakteristik fungsi dengan nilai berikut:
  - $f(a)f(b) < 0$ → jumlah akarnya adalah ganjil
  - $f(a)f(b) > 0$ → jumlah akarnya adalah genap atau tidak ada sama sekali

# Metode Bagi Dua (*bisection*)

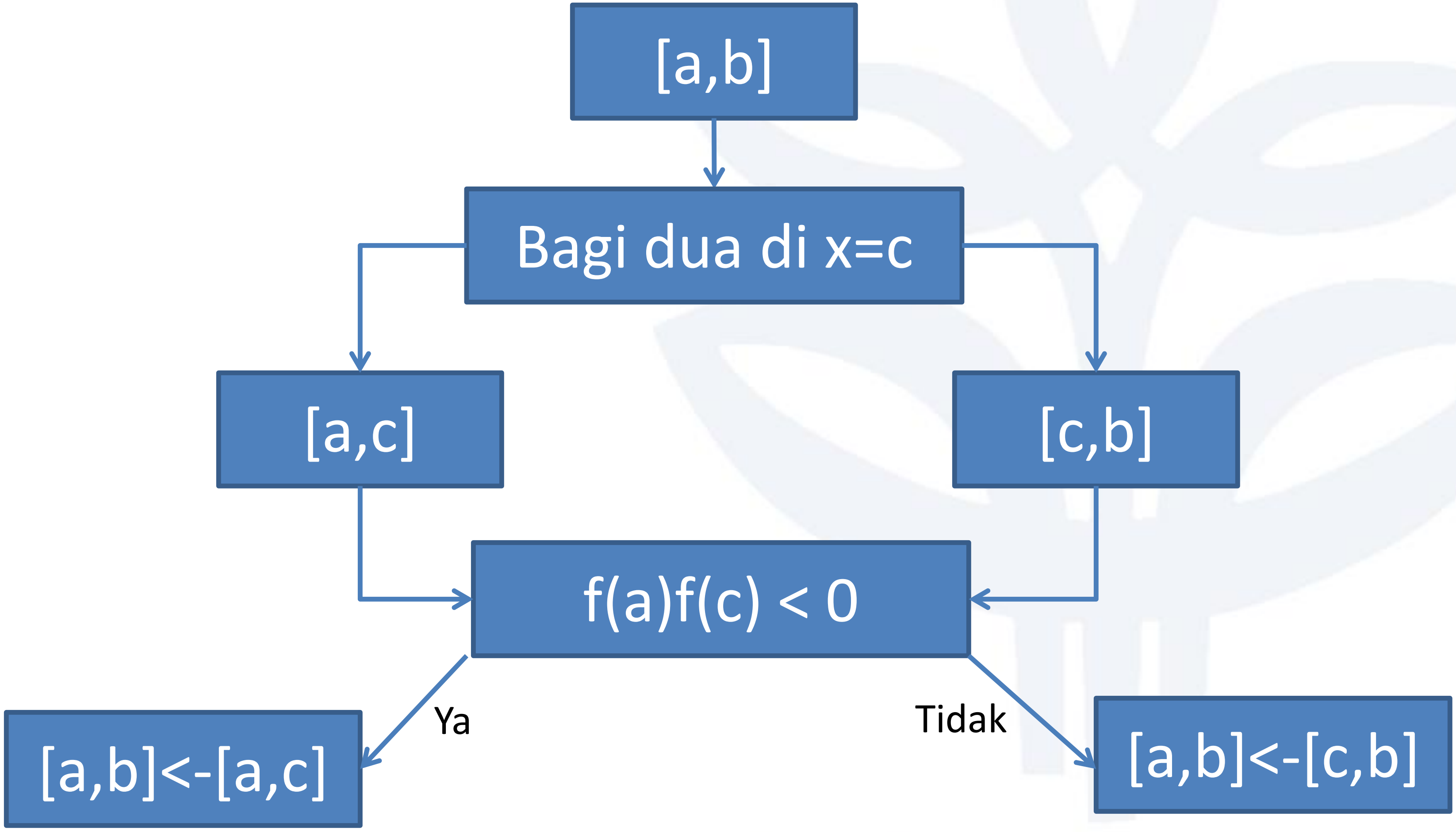
- Misalkan sudah ditentukan selang [a,b] sehingga $f(x)$ kontinu pada selang [*a, b*] dimana $f(a).f(b) \quad 0$.
- Hal ini menunjukkan bahwa $f(x)$ berubah tanda pada [*a, b*] yang artinya $f(x) = 0$ mempunyai sedikitnya 1 akar pada [*a, b*]
- Metode bagi dua, pada setiap iterasinya, membagi selang [a,b] menjadi 2 bagian, misalkan x=c sehingga didapatkan dua subselang tutup dengan ukuran yang sama yaitu [a,c] dan [c,b]
- Selang yang dipilih untuk iterasi berikutnya adalah subselang yang memuat akar dan bergantung pada $f(a)f(c) < 0$ atau $f(c)f(b) < 0$
- Selang baru dibagi dua denga cara yang sama sampai dengan ukuran selang dengan sangat kecil (Lebar selang baru [a-c] atau [c-b] lebih kecil dari nilai toleransi lebar selang yang mengurung akar)

# Metode Bagi Dua

- Misalkan $f(x)$ kontinu pada selang [*a*, *b*] dimana $f(a)f(b)$ $0$.
- Hal ini menunjukkan bahwa $f(x)$ berubah tanda pada [*a*, *b*] yang artinya $f(x) = 0$ mempunyai sedikitnya 1 akar pada [*a*, *b*]



# Metode Bagi Dua



# Metode Bagi Dua

- Proses pencarian akar pada $f(x) = 0$ pada selang [*a*, *b*] dapat dilakukan dengan membagi selang tersebut menjadi dua bagian, yaitu [*a*, *c*] dan [*c*, b], sehingga berlaku

$$f(a)f(c) = \begin{cases} < 0, akar\ \ pada\ selang\ (a,c) \\ = 0, akar = c \\ > 0, akar\ \ pada\ selang\ (c,b) \end{cases}$$

$$c = \frac{a+b}{2}$$

$$f(c)f(b) = \begin{cases} < 0, akar\ \ pada\ selang\ (c,b) \\ = 0, akar = c \\ > 0, akar\ \ pada\ selang\ (a,c) \end{cases}$$

# Metode Bagi Dua

- Selanjutnya untuk mencari akar $f(x) = 0$ pada selang $[a, c]$ dapat dilakukan dengan membagi selang tersebut menjadi dua bagian, yaitu $[a, c_1]$ dan $[c_1, c]$, sehingga berlaku

$$c_1 = \frac{a+c}{2}$$

$$f(a)f(c_1) = \begin{cases} < 0, akar\ pada\ selang\ (a, c_1) \\ = 0, akar = c_1 \\ > 0, akar\ pada\ selang\ (c_1, c) \end{cases} \qquad f(c_1)f(c) = \begin{cases} < 0, akar\ pada\ selang\ (c_1, c) \\ = 0, akar = c \\ > 0, akar\ pada\ selang\ (a, c_1) \end{cases}$$

- Demikian proses penentuan akar tersebut berlanjut, sehingga jika akar terdapat pada selang $[a, c_1]$, maka selang tersebut dibagi dua sehingga diperoleh selang-selang $[a, c_2]$ dan $[c_2, c_1]$ dengan $c_2 = \frac{a+c_1}{2}$

# Metode Bagi Dua

Proses akan berhenti bila

1. Akar telah ditemukan
2. Mencapai iterasi maksimum (N) yang telah ditetapkan sebelumnya
3. $|b - a| \leq \varepsilon$ (lebar selang cukup kecil).

Algoritme metode Bagi Dua untuk pencarian akar dari $f(x) = 0$:

1. Input : $a$, $b$, $\varepsilon$ dengan $f(a).f(b) < 0$.
2. Set $$c = \frac{a+b}{2}$$

   jika $f(c) = 0$ maka akar $= c \rightarrow$ proses selesai

   jika $f(a).f(c) < 0$ maka set $b = c$

   jika tidak, set $a = c$

   {periksa apakah $|b - a| \leq \varepsilon$}

   jika $|b - a| \leq \varepsilon \rightarrow$ proses selesai $\rightarrow$ akar $= c$

   jika tidak, kembali ke langkah 2

# Metode Bagi Dua

Contoh 1: tentukan akar dari persamaan

$$f(x) = e^{x} - 3x$$

pada selang [0, 1] dengan $\varepsilon = 10^{-2}$. (perhitungan menggunakan desimal 4 digit)

| iterasi | a | b | c | f (c ) |
|---|---|---|---|---|
| 1 | 0 | 1 | 0.5000 | 0.1487 |
| 2 | 0.5000 | 1 | 0.7500 | -0.1330 |
| 3 | 0.5000 | 0.7500 | 0.6250 | $-6.754 \times 10^{-3}$ |
| 4 | 0.5000 | 0.6250 | 0.5625 | 0.0676 |
| 5 | 0.5625 | 0.6250 | 0.5938 | 0.0295 |
| 6 | 0.5938 | 0.6250 | 0.6094 | 0.01112 |
| 7 | 0.6094 | 0.6250 | 0.6172 | $2.130 \times 10^{-3}$ |

# Metode Bagi Dua

Contoh 1 (lanjutan):

Lebar selang pada setiap iterasi

| iterasi | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| $\lvert b - a \rvert$ | 0.5 | 0.25 | 0.125 | 0.0625 | 0.0312 | 0.0156 | $7.8 \times 10^{-3}$ |

jika nilai ujung selang pada iterasi ke-i dinotasikan sebagai $a_i$ dan $b_i$, untuk $i$ = 1, 2,...,$n$, maka lebar selang pada saat ke-$i$ adalah

$$\frac{1}{2^i}(b-a) \tag{1}$$

Proses pencarian akar akan berhenti bila lebar selang sangat kecil, artinya $\lvert b_n - a_n \rvert \leq \varepsilon$

# Metode Bagi Dua

- Jika nilai ε yang diketahui, banyaknya iterasi sampai akar diperoleh dapat ditentukan.
- Pada iterasi ke-*n*, proses pencarian akar akan berhenti karena

$$\left| b_n - a_n \right| \le \varepsilon \qquad (2)$$

- Dengan menggunakan persamaan (1), persamaan (2) menjadi

$$\frac{1}{2^n}\left(b - a\right) \le \varepsilon \qquad (3)$$

- Selanjutnya dari persamaan (3) diperoleh

$$n = INT\left(\frac{\log\left(\frac{b-a}{\varepsilon}\right)}{\log 2}\right) + 1$$

# Metode Bagi Dua

Tentukan akar terbesar dari persamaan

$$f(x) = x^6 - x - 1 = 0$$

pada selang [1, 2] dengan $\varepsilon$ =0.001

| iterasi | a | b | c | b - c | f(c ) |
|---|---|---|---|---|---|
| 1 | 1.0000 | 2.0000 | 1.5000 | 0.5000 | 8.8906 |
| 2 | 1.0000 | 1.5000 | 1.2500 | 0.2500 | 1.5647 |
| 3 | 1.0000 | 1.2500 | 1.1250 | 0.1250 | -0.0977 |
| 4 | 1.1250 | 1.2500 | 1.1875 | 0.0625 | 0.6167 |
| 5 | 1.1250 | 1.1875 | 1.1562 | 0.0312 | 0.2333 |
| 6 | 1.1250 | 1.1562 | 1.1406 | 0.0156 | 0.0616 |
| 7 | 1.1250 | 1.1406 | 1.1328 | 0.0078 | -0.0196 |
| 8 | 1.1328 | 1.1406 | 1.1367 | 0.0039 | 0.0206 |
| 9 | 1.1328 | 1.1367 | 1.1348 | 0.0020 | 0.0004 |
| 10 | 1.1328 | 1.1348 | 1.1338 | 0.00098 | -0.0096 |

# Metode Terbuka

- Tidak memerlukan selang yang mengurung akar
- Memerlukan tebakan awal akar atau lebih dari dua yang tidak perlu mengurung akar
- Hampiran akar didasarkan pada hampiran akar sebelumnya
- Iterasi bisa lebih cepat dibandingkan tertutup namun nilai bisa mencapai konvergen pada akar atau malah divergen

# Metode Iterasi titik tetap

- Sering disebut dengan metode iterasi sedarhana, dilakukan secara langsung
- Kesederhanaan metode ini karena pembentukan prosedur iterasinya mudah dibentuk sebagai berikut:
  - Susun persamaan $f(x) = 0$ menjadi bentuk $x = g(x)$. Lalu lakukan prosedur iterasi:
  - $x_{r+1} = g(x_r)$. Terka nilai x berikutnya sampai dengan konvergen sampai ditemukan akar s sedemikian sehingga $f(s) = 0$ atau $s = g(s)$ dengan kondisi berhenti adalah $|x_{r+1} - x_r| < \varepsilon$

# Metode Iterasi titik tetap

- Tentukan akar persamaan $f(x) = x^2 - 2x - 3 = 0$ dengan metode iterasi titik tetap dengan $\varepsilon$=0.000001
- Algoritme:
  - Konversi $f(x) = 0$ menjadi $x = g(x)$
  - Terka misal nilai $x_0 = a$
  - Prosedur iterasi $x_1 = g(a)$

# Metode Newton

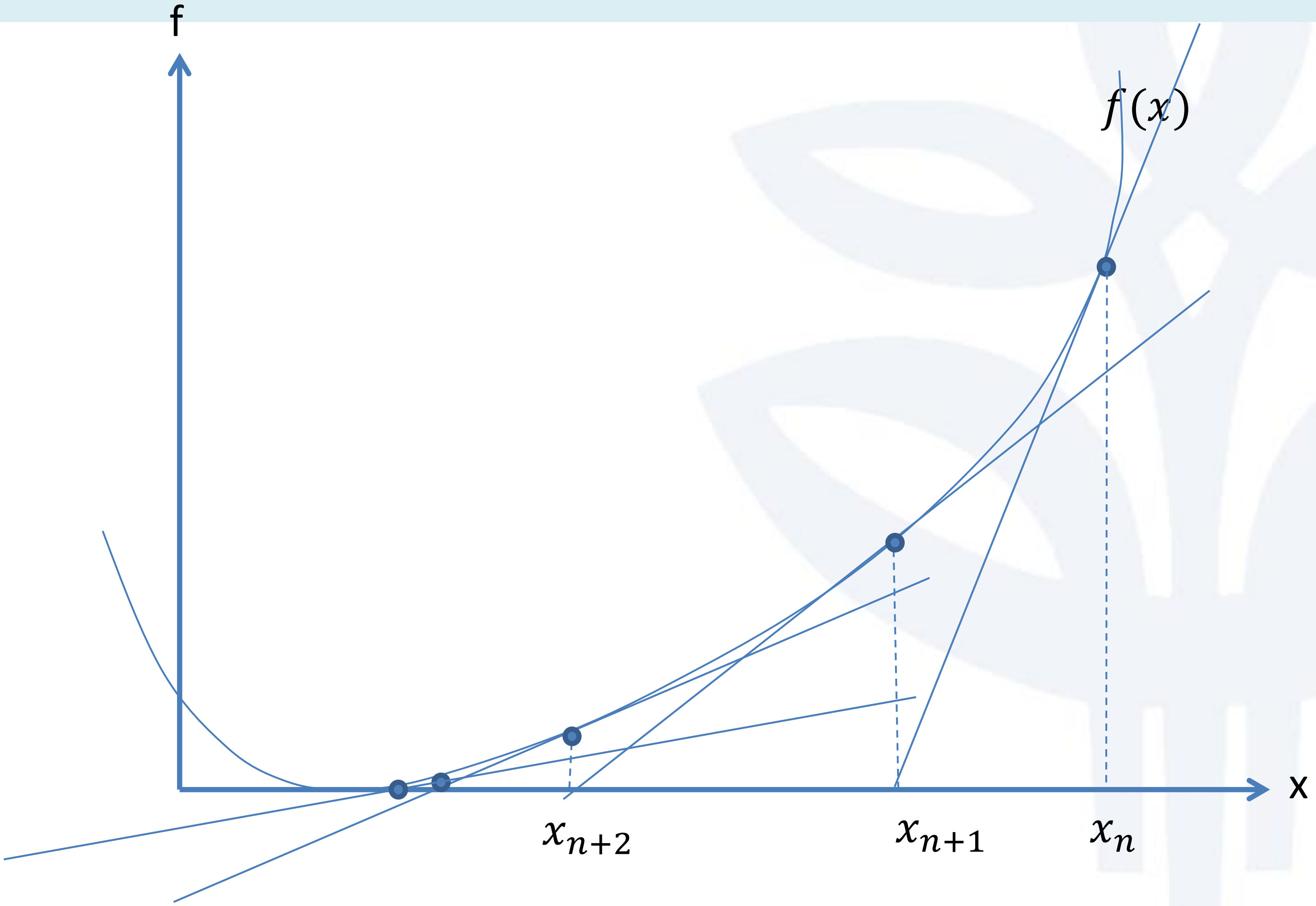
- Metode ini paling banyak digunakan dalam mencari akar – akar dari suatu persamaan.
- Jika perkiraan awal dari akar adalah $X_i$ , suatu garis singgung dapat dibuat dari titik $(Xi\ (\ f(xi))$
- Titik  dimana garis singgung tersebut memotong sumbu x biasanya memberikan perkiraan yang lebih dekat dari nilai akar
- Turunan pertama pada Xi adalah ekivalen dengan kemiringan

# Metode Newton

- metode pendekatan yang menggunakan satu titik awal dan mendekatinya dengan memperhatikan slope atau gradien pada titik tersebut.Titik pendekatan ke n+1 dituliskan dengan :
- Jenis I (Raphson)

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

# Metode Newton




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

# Algoritma Metode Newton

1. Definisikan fungsi $f(x)$ dan turunanya $f'(x)$
2. Tentukan toleransi error $(\in)$ dan iterasi maksimum $(n)$
3. Tentukan nilai pendekatan awal $x_0$
4. Hitung $f(x_0)$ dan $f'(x_0)$
5. Untuk iterasi 1 s/d n atau $|x_{i+1} - x_i| > \in$
   - Hitung $f(xi)$ dan $f'(xi)$

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}$$

6. Nilai minimizer (akar persamaan) adalah nilai $x_i$ yang terakhir diperoleh.

# Tugas Mandiri

Konversikan algoritme metode newton menggunakan Proogram R untuk menentukan akar dari

$$f(x) = x^3 + sin^2 x - \ln x$$

Nilai inisial $x_0 = 1$, toleransi error (e) dan iterasi maksimum (n) adalah $\varepsilon = 10^{-5}$

Tugas dikumpulkan di LMS dalam file *.doc yang berisikan:

- Penghitungan manual
- Script program R dan *scrshoot* hasilnya
- Plot fungsi untuk skuens 1:3 dengan beda nilai 0.1

# Referensi


- Victor A. Bloomfield. 2014. *Using R for Numerical Analysis in Science and Engineering*. 1 edition. Chapman and Hall/CRC
- Elementary Numerical Analysis, Second edition, Kendall Atkinson