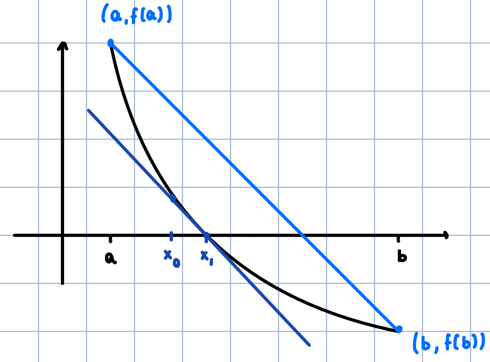
$q_k \simeq f'(\alpha)$ $\qquad x_{k+1} = x_k - \dfrac{f(x_k)}{q_k}$

α = radice



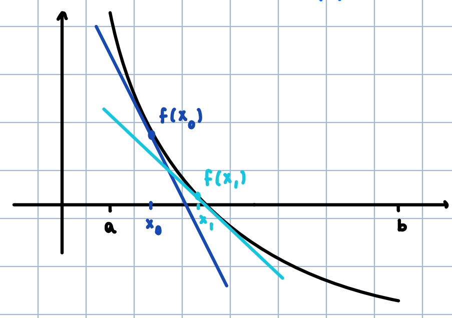
METODO delle CORDE: $\quad q_k = q = \dfrac{f(b) - f(a)}{b - a}$

METODO di NEWTON (o METODO delle TANGENTI): $\quad q_k = f'(x_k)$

$$\begin{cases} x_0 \text{ dato} \\ x_{k+1} = x_k - \dfrac{f(x_k)}{f'(x_k)} \end{cases} \qquad k = 0, 1, \dots \quad x_0 \text{ dato}$$

$\downarrow$

non deve essere $f'(x_k) \neq 0$



OSS: Il metodo di NEWTON è localmente convergente

TEOREMA: Sia $f \in C^2([a,b])$ se:

1. $f(a)f(b) < 0$
2. $f'(x) \neq 0 \quad \forall x \in [a,b]$
3. $f''(x) \geq 0 \quad \forall x \in [a,b] \quad$ o $\quad f''(x) \leq 0 \quad \forall x \in [a,b]$
4. $\dfrac{f(a)}{f'(a)} < (b-a)$ e $\dfrac{f(b)}{f'(b)} < b - a$

$\rightarrow$ allora il metodo di NEWTON converge all'unica $\alpha \in [a,b]$, $\forall x_0 \in [a,b]$

OSS: Il metodo di NEWTON ha ordine di convergenza $p = 2$, tranne in alcune eccezioni, risulta quindi un po' più veloce del metodo delle corde

ESERCIZIO: Confrontare il comportamento dei metodi sopra introdotti nel calcolo di radice α di

$f(x) = \cos^2(2x) - x^2$ nell'intervallo $[0, 1.5]$

RISOLUZIONE: $\quad x_0 = 0.6 \quad$ tolleranza $\varepsilon = 10^{-10}$ sul residuo

$\alpha \simeq 0.5149$

CRITERI D'ARRESTO

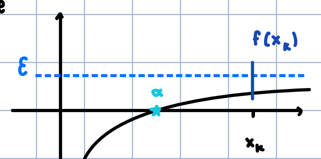
Noi ricaviamo una sucessione $\{x_k\}_{k=0,1,\dots}$ t.c. $\lim\limits_{k \to +\infty} x_k = \alpha$

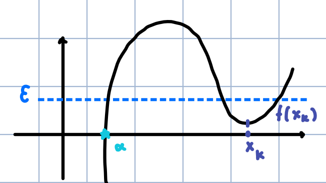
Fissiamo una tolleranza $\varepsilon \in \mathbb{R}^+$

• CRITERIO del RESIDUO:

Il processo iterativo si arresta al minimo k tale per cui $|f(x_k)| < \varepsilon$

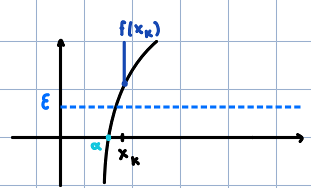
Attenzione





OSS: Dire che $|f(x_k)| < \varepsilon$ non implica $|x_k - \alpha| < \varepsilon$.

L'errore su α può essere comunque grande: sono vicina alla radice ma la mia f è molto grande



• CRITERIO DELL'INCREMENTO:

Il processo iterativo si arresta al minimo k per cui $|x_{k+1} - x_k| < \varepsilon$

Se l'avanzamento verso la radice è "lento", non è un buon test

$$x_{k+1} - x_k = x_{k+1} \pm \alpha - x_k = (x_{k+1} - \alpha)(\alpha - x_k) = e_{k+1} - e_k \simeq 0$$

• **NUMERO MASSIMO DI ITERAZIONI : N_{max}**

Oss: Sarà opportuno abbinare più criteri d'arresto

Esercizio: Calcolare la radice α della funzione $f(x) = e^{-x} - \eta \overset{?}{=} 0$

Per $\eta = 10^{-9}$ $\quad \alpha \cong 20.723$ $\qquad \alpha = ?$

con tolleranza $\varepsilon_1 = 10^{-3}$ $\quad \varepsilon_2 = 10^{-10}$, con criterio residuo/incremento

con metodo di NEWTON, $x_0 = 0$.

Memorizzare il numero di iterazioni.

STUDIAMO IL CONDIZIONAMENTO DEL PROBLEMA DEL CALCOLO DI UNA RADICE

$f(x) = 0$ riscriviamo l'eq.ne nella forma

$$f(x) = \varphi(x) - d = 0 \qquad d \in \mathbb{R}$$

$$f(x) = \sin(x)\tan(x) = \underbrace{\sin(x)\tan(x) + 3}_{\varphi(x)} \underbrace{- 3}_{d} = 0$$

$$d + \delta d = \varphi(x + \delta x) = \varphi(x) + \varphi'(x)\,\delta x + \frac{\varphi''(x)}{2!}(\delta x)^2 + \frac{\varphi'''(x)}{3!}(\delta x)^3 + \dots + \frac{\varphi^{(k)}(x)}{k!}(\delta x)^k + \dots + o(\delta x)^m$$

Supponiamo che x sia radice di ordine m

$\varphi^{(k)}(x) = 0 \quad k = 1, \dots, m-1 \qquad f(x) = 0 \quad \dots \quad f'(x) = 0 \quad f''(x) = 0 \quad f^{(m-1)}(x) = 0$

$f'(x) = \varphi'(x) \quad \dots \quad f^{(m-1)}(x) = \varphi^{(m-1)}(x)$

$$\delta d = \frac{\varphi^{(m)}(x)}{m!}(\delta x)^m + o(\delta x)^m = \frac{f^{(m)}(x)}{m!}(\delta x)^m + \cancel{o(\delta x)^m}$$

lo trascuro

$$(\delta x)^m \cong \frac{\delta d\, m!}{f^{(m)}(x)} \qquad = \sup \frac{\|\delta x\|}{\left\|\frac{f^{(m)}(x)}{m!}\right\| \, \|(\delta x)^m\|}$$

$$= \frac{\|\delta x\|}{\|\delta d\|} = \left|\frac{\delta d\, m!}{f^{(m)}(x)}\right|^{\frac{1}{m}} \frac{1}{|\delta d|}$$

Oss: $m = 1$

Il problema è MAL CONDIZIONATO se $f^{(m)}(x) \simeq 0$

$m > 1$

Il problema potrebbe essere MAL CONDIZIONATO

Es: Osserviamo la "sensibilità" del calcolo delle radici di un polinomio al variare dei suoi coefficienti $\qquad$ lezione 3 2021

Produttoria = analogo alla sommatoria ma ha il prodotto

$$P_{10}(x) = \prod_{k=1}^{10} (x + k) = x^{10} + a_1 x^9 + a_2 x^8 + \dots + a_{10}$$

POLINOMIO di WILKINSON

$= (x+1)(x+2)(x+3)\cdots(x+10)$

le 10 radici : $\alpha = -k \qquad k = 1, \dots, 10$

$\hat{P}_{10} = P_{10} + \varepsilon x^9 \qquad \varepsilon = 10^{-7}$

Matlab:

Comando *poly* : converte le radici a un polinomio.

poly(V), quando V è un vettore, è un vettore i cui elementi sono i coefficienti del polinomio, le cui radici sono gli elementi di V.

```
>> coeff= poly ([-1: -1: -10])
 coeff=

  Columns 1 through 6

  1   55   1320   18150   157773   902055

  Columns 7 through 11

  3416930   8409500   12753576   10628640   3628800
```

Comando *roots* : trova le radici del polinomio

roots(C) computa le radici del polinomio i cui coefficienti sono gli elementi del vettore C. Se C ha N+1 componenti, il polinomio è C(1)*X^N + … + C(N)*X + C(N +1).

coeff = poly([-1: -1: 10]) Calcolo coefficienti polinomio di Wilkinson

radici = roots(coeff) Calcolo le radici "approssimate" attraverso un alg. di Matlab

r = [-10 : +1 : -1] → radici esatte

Err = abs (r - radici) $\sim 10^{-8}$ errore assoluto

coeff (2) = coeff (2) + 10^{-9} perturbo un coefficiente

radici1= roots(coeff)

Err = abs (r - radici1) $\sim 10^{-4}$

Oss: L'approssimazione numerica di una radice α di f si basa in genere sull'uso di metodi iterativi, si costruisce cioè una successione di valori reali x^k t.c.

$$\lim_{k \to +\infty} x^k = \alpha$$