**Newton – Raphson method**

**201710793 박진홍**

1. **What is Newton-Raphson Method**

This is a numerical solution that can solve equations. By using this, solutions and parameters of various nonlinear models can be solved approximately.

1. **Description**
   1. **Background**

* We have to know background information on the numerical approximation of the derivative of a function.
* Calculating derivative Using Finite Difference Method(유한차분법)

Taylor series expansion of for a small value h around a given point 



The 1st order approximation of the derivative of using one of the above equation



Therefore, the first order numerical approximation becomes



* 1. **How to Use**
* NR method begins from the idea of geometrical interpretation of differentials that derivative function at a point is same with gradient of tangent line.

First of all, 1st order Taylor Series Approximation of a function can be written as



* 1. **Features and Limitations of Newton-Raphson Method**
* 우선 이 방법을 사용하기 위해서 주어진 함수는 연속이고 미분가능해야 한다. 또 해가 여러 개 있다 하더라도 그 중 하나의 해 만을 찾을 수 있다. 이것은 초기값을 어떠한 값으로 잡느냐에 따라 도출되는 해가 달라지는 것 또한 의미한다.
* 초기값을 잘못주는 경우를 보완하기 위해서 먼저 일정한 간격으로 x 값을 변화시키면서 함수값의 변화를 본 후 함수값의 부호가 바뀌는 구간을 보간법을 통해 초기값을 잡는 방법을 이용할 수 있다.

1. **Pseudo Code**

텍스트이(가) 표시된 사진

자동 생성된 설명

1. **Fortran Code**

테이블이(가) 표시된 사진

자동 생성된 설명

|  |
| --- |
| Main |
| PROGRAM NR\_MAIN  !-----------------------------------------------------------------------------------  IMPLICIT NONE  !-----------------------------------------------------------------------------------  INTEGER :: Kp , ITERMAX  INTEGER :: NFile\_Out  !  DOUBLE PRECISION :: XX(100000)  DOUBLE PRECISION :: FF(100000)  DOUBLE PRECISION :: FFD(100000)  DOUBLE PRECISION :: E  DOUBLE PRECISION :: error  DOUBLE PRECISION :: X0  !-----------------------------------------------------------------------------------  X0 = 3  E = 10E-7  ITERMAX = 100000  !  NFile\_Out = 1000  OPEN(NFile\_Out, FILE = 'OUT\_NR.m')  WRITE(NFile\_Out,\*) ' result = ['  XX(1) = X0  CALL function\_NR(X0,FF(1))  DO Kp=1, ITERMAX  CALL functionD\_NR(XX(Kp),FFD(Kp))  CALL function\_NR(XX(Kp),FF(Kp))  !WRITE (\*,\*) FF(Kp)  error = ABS(FF(Kp))    IF (error < E) THEN  EXIT;  END IF    PRINT 106, Kp, XX(Kp), FF(Kp), FFD(Kp), Error  WRITE(NFile\_Out, 106) Kp, XX(Kp), FF(Kp), FFD(Kp), Error  XX(Kp+1) = XX(Kp) - FF(Kp)/FFD(Kp)  END DO  WRITE(NFile\_Out,\*) '];'  106 FORMAT(2X, I8, 100(5X, E16.8))  !------------------------------------------------------------------------------------------  END PROGRAM NR\_MAIN |
| Function |
| SUBROUTINE function\_NR(XX, FF)  !------------------------------------------------------------------------------------------  IMPLICIT NONE  !------------------------------------------------------------------------------------------  DOUBLE PRECISION :: XX, FF  !------------------------------------------------------------------------------------------  FF = 2\*XX\*\*3 + 4\*XX + 1  RETURN  END |
| Function\_D |
| SUBROUTINE functionD\_NR(XX, FF)  !------------------------------------------------------------------------------------------  IMPLICIT NONE  !------------------------------------------------------------------------------------------  DOUBLE PRECISION :: XX, FF  !------------------------------------------------------------------------------------------  FF = 6\*XX\*\*2 + 3  RETURN  END |