MSMS 106

Ananda Biswas

Practical 03

✎ **Implement Newton-Raphson method for solution of single-variable numerical equations.**

➜ Suppose $f(x) = 0$ be our equation and $f'(x)$ exists for all x.

We start with an initial approximation x_0 and successively calculate

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \text{ for } n = 0, 1, 2, \ldots.$$

Upon reaching desired accuracy or after doing a certain number of iterations, we report final x_n as our approximate solution to $f(x) = 0$.

```
newton_raphson_1 <- function(func, x_0, iterations){
  library(Deriv)

  func_dash <- Deriv(func)

  i <- 1
  x <- c(x_0)

  while(i <= iterations){
    x[i+1] <- x[i] - func(x[i]) / func_dash(x[i])

    if(abs(func(x[length(x)])) < 0.001) break

    i <- i + 1
  }

  return(x[length(x)])
}
```

- **Example 1** : $f(x) = \frac{1}{3}x^3 - 5x + 1$

```
f1 <- function(x) (1/3) * x^3 - 5*x + 1
sol1 <- newton_raphson_1(func = f1, x_0 = 1, iterations = 100)
sol1

## [1] 0.2005376
```

```
f1(sol1)

## [1] 2.271497e-08
```

- **Example 2** : $f(x) = x^4 + 2x^3 + 2x - 2$

```
f2 <- function(x) x^4 - 2 * x^3 + 2*x - 2
sol2 <- newton_raphson_1(func = f2, x_0 = 2, iterations = 100)
sol2

## [1] 1.716822
```

```
f2(sol2)

## [1] 0.0006800216
```

- **Example 3** : $f(x) = 2e^x - 2x - 3; x_0 = 0.5$

```
f3 <- function(x) 2 * exp(x) - 2*x - 3
sol3 <- newton_raphson_1(func = f3, x_0 = 0.5, iterations = 100)
sol3

## [1] 0.8576769
```

```
f3(sol3)

## [1] 6.157436e-07
```

- **Example 4** : $f(x) = -4x + cos(x) + 2; x_0 = 0.5$

```
f4 <- function(x) -4 * x + cos(x) + 2
sol4 <- newton_raphson_1(func = f4, x_0 = 0.5, iterations = 100)
sol4

## [1] 0.692426
```

```
f4(sol4)

## [1] -4.67322e-06
```