MSMS 106

Ananda Biswas

Practical 04

✎ **Implement Lagrange's Interpolation method to approximate value of a function at a given point.**

➔ The Lagrange Interpolating polynomial $P(x)$ of degree n that passes through the data points $(x_1, y_1 = f(x_1))$, $(x_2, y_2 = f(x_2))$, ..., $(x_n, y_n = f(y_n))$ is given by

$$P(x) = y_1 \cdot \frac{(x - x_2)(x - x_3)\dots(x - x_n)}{(x_1 - x_2)(x_1 - x_3)\dots(x_1 - x_n)} + y_2 \cdot \frac{(x - x_1)(x - x_3)\dots(x - x_n)}{(x_2 - x_1)(x_2 - x_3)\dots(x_2 - x_n)} + \dots + y_n \cdot \frac{(x - x_1)(x - x_2)\dots(x - x_{n-1})}{(x_n - x_1)(x_n - x_2)\dots(x_n - x_{n-1})}.$$

```r
lagrange_interpolation <- function(x, xi, yi){

  num <- rep(1, length(xi))
  denom <- rep(1, length(xi))

  for (i in 1:length(xi)) {

    for (j in 1:length(xi)) {
      if(i != j){
        num[i] <- num[i] * (x - xi[j])
        denom[i] <- denom[i] * (xi[i] - xi[j])
      }
    }
  }

  pred <- sum((num / denom) * yi)
  return(pred)
}
```

- **Example 1**

```r
temp <- c(361, 367, 378, 387, 399)
pres <- c(154.9, 167, 191, 212.5, 244.2)
```

```r
lagrange_interpolation(371.2, temp, pres)

## [1] 175.8824
```

- **Example 2**

```r
x <- c(2, 2.5, 3)
y <- c(0.69315, 0.91629, 1.09861)
```

```r
lagrange_interpolation(2.7, x, y)

## [1] 0.9941164
```

- **Example 3**

```r
x <- c(5, 6, 9, 11)
y <- c(12, 13, 14, 16)
```

```r
lagrange_interpolation(10, x, y)

## [1] 14.66667
```