

Neville

Uriel Paluch

17/10/2021

Neville

Una dificultad práctica con la interpolación de Lagrange es que el término de error es difícil de aplicar, por lo que el grado del polinomio que se necesita para la precisión deseada en general se desconoce hasta que se realizan los cálculos.

```
# Configuración -----
options(scipen = 999)

# Metodo de Neville -----

Neville <- function(x, y, interpolator){
  #cantidad de iteraciones que voy a hacer
  n <- length(x)-1

  #Hago un vector vacio para llenar el df
  empty_vec <- rep(0, times = length(x))

  df <- data.frame(x, y)

  for (i in 1:n) {
    df[glue::glue("Q",i)] <- empty_vec

    for (j in (i+1):(n+1)) {

      df[j, (i+2)] <- ( (interpolator-x[(j-i)]) * df[j,(i+1)] - (interpolator-x[j]) * df[(j-1),(i+1)] ) /

    }
  }

  print(df)
}

# Llamado a la función -----
# x es la preimagen
# y es la imagen
# interpolator es el número que se desea interpolar
Neville(x = c(0, 0.25, 0.5, 0.75), y = c(1, 1.64872, 2.71828, 4.48169), interpolator = 0.43)
```

```
##      x      y      Q1      Q2      Q3
```

```
## 1 0.00 1.00000 0.000000 0.000000 0.000000
## 2 0.25 1.64872 2.115798 0.000000 0.000000
## 3 0.50 2.71828 2.418803 2.376383 0.000000
## 4 0.75 4.48169 2.224525 2.348863 2.360605
```

El número que aparece en el último escalon es el resultado de la interpolación