Lagrangian Interpolation

Tim Dosen Matematika Lanjut

11/28/2020

Polynomial interpolation is the method of determining a polynomial that fits a set of given points. There are several approaches to polynomial interpolation, of which one of the most well known is the Lagrangian method. This post will introduce the Lagrangian method of interpolating polynomials and how to perform the procedure in R.

Lagrangian Polynomial Interpolation

The Lagrangian method of polynomial interpolation uses Lagrangian polynomials to fit a polynomial to a given set of data points. The Lagrange interpolating polynomial is given by the following theorem:

For a set of data points $(x_0, y_0), (x_1, y_1), \cdots, (x_n, y_n)$ with no duplicate x and there exists a function f which evaluates to these points, then there is a unique polynomial $P(x)$ with degree $\leq n$ also exists. The polynomial is given by:

$P(x) = f(x_o)L_{n,0}(x) + \cdots + f(x_n)L_{n,n}(x) = \sum_{k=0}^{n} f(x_k)L_{n,k}(x)$

Where each k in $k = 0, 1, \cdots, n$ is:

$L_{n,k} = \frac{(x-x_0)(x-x_1)\cdots(x-x_{k-1})(x-x_{k+1})\cdots(x-x_n)}{(x_k-x_0)(x_k-x_1)\cdots(x_k-x_{k-1})(x_k-x_{k+1})\cdots(x_k-x_n)} = \prod_{\substack{i=0 \\ i\neq k}}^{n} \frac{(x-x_i)}{(x_k-x_i)}$

Lagrangian Polynomial Interpolation Example

Consider the following set of data points:

x	y
0	7
2	11
3	28
4	63

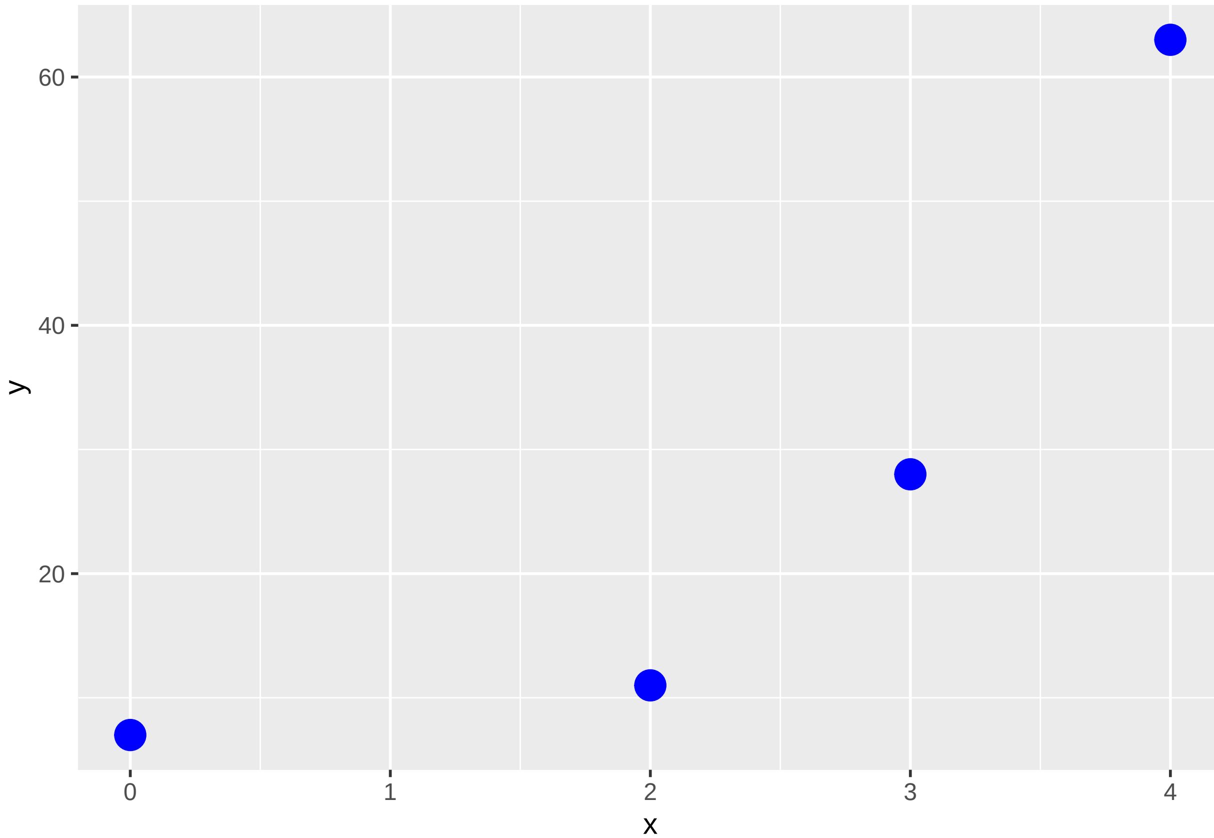
The graph of the points looks like:

```
library(ggplot2)

x <- c(0, 2, 3, 4)
y <- c(7, 11, 28, 63)

dat <- data.frame(cbind(x, y))

ggplot(dat, aes(x=x, y=y)) + geom_point(size=5, col='blue')
```



We wish to find a polynomial that passes through these points. Note that much of the intermediate algebra will be skipped for the sake of brevity.

Start with $L_1(x)$:

$L_1(x) = \frac{(x-2)(x-3)(x-4)}{(0-2)(0-3)(0-4)} = -\frac{1}{24}(x-2)(x-3)(x-4)$

$L_2(x)$ is then:

$L_2(x) = \frac{(x-0)(x-3)(x-4)}{(2-0)(2-3)(2-4)} = \frac{1}{4}x(x-3)(x-4)$

Then finally, $L_3(x)$ and $L_4(x)$:

$L_3(x) = \frac{(x-0)(x-2)(x-4)}{(3-0)(3-2)(3-4)} = -\frac{1}{3}x(x-2)(x-4)$

$L_4(x) = \frac{(x-0)(x-2)(x-3)}{(4-0)(4-2)(4-3)} = \frac{1}{8}x(x-2)(x-3)$

These polynomials are then multiplied by they corresponding y value and added together, resulting in the following polynomial:

$7\left(-\frac{1}{24}(x-2)(x-3)(x-4)\right) + 11\left(\frac{1}{4}x(x-3)(x-4)\right) - 28\left(-\frac{1}{3}x(x-2)(x-4)\right) + 63\left(\frac{1}{8}x(x-2)(x-3)\right)$

Simplifying this equation yields:

$x^3 - 2x + 7$

Polynomial Interpolation in R

Lagrangian polynomial interpolation can be performed in R using the `poly.calc()` function from the polynom package.

```
library(polynom)
```

Use the `poly.calc()` function to interpolate the data points using the Lagrangian method.

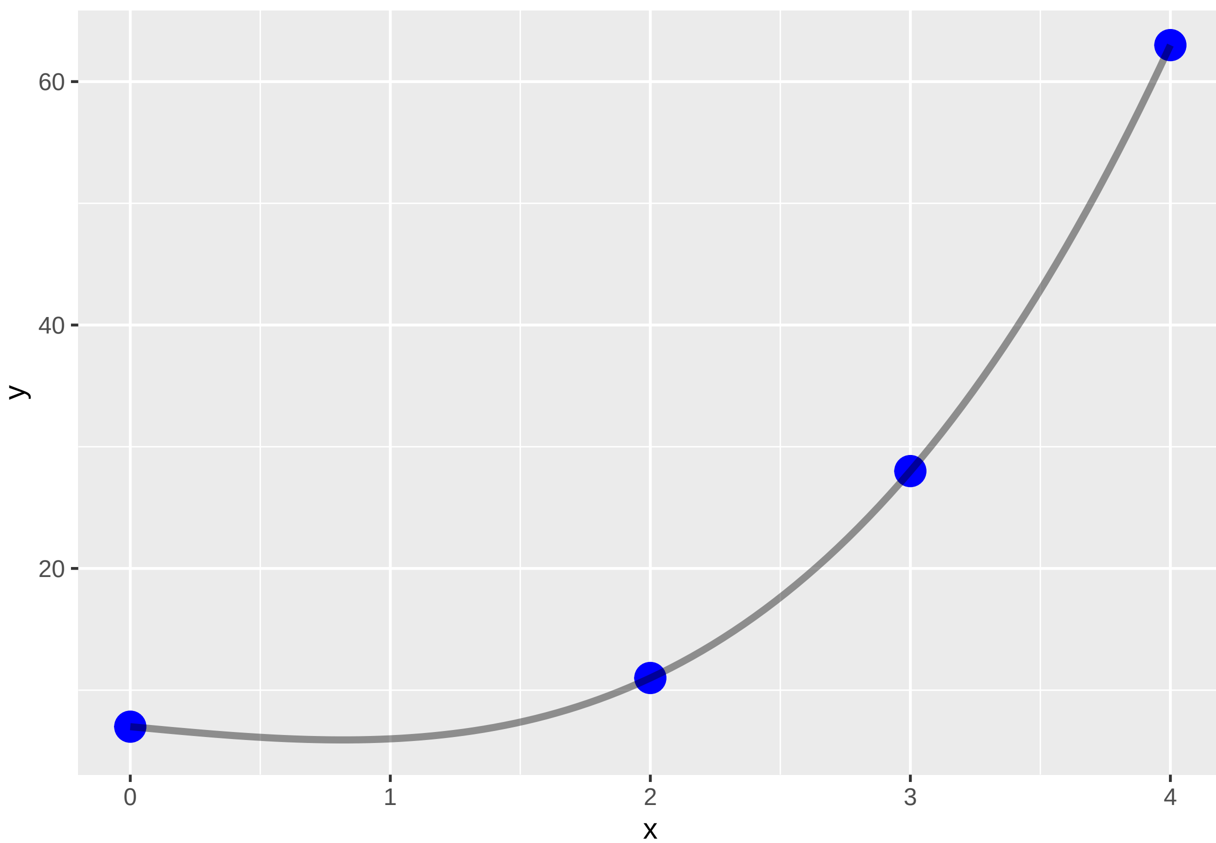
```
poly.calc(x, y)
```

```
## 7 - 2*x + x^3
```

We can plot this polynomial against the original data points to confirm visually that the polynomial does indeed pass through the points.

```
f <- function(x) {
  return(x^3 - 2 * x + 7)
}

ggplot(dat, aes(x=x, y=y)) +
  geom_point(size=5, col='blue') +
  stat_function(fun = f, size=1.25, alpha=0.4)
```



Exercise

Given the data below:

x	y
1	1
2	5
2.5	7
3	8
4	2
5	1

Calculate:

1. $f(1.5)$
2. $f(3.4)$

References

Burden, R. L., & Faires, J. D. (2011). Numerical analysis (9th ed.). Boston, MA: Brooks/Cole, Cengage Learning.

Cheney, E. W., & Kincaid, D. (2013). Numerical mathematics and computing (6th ed.). Boston, MA: Brooks/Cole, Cengage Learning.

Source: https://rpubs.com/aaronsc32/lagrangian-polynomial-interpolation-r