PERTEMUAN 8
INTERPOLASI DAN EKSTRAPOLASI (2)

TUJUAN PRAKTIKUM

Mahasiswa mampu mengimplementasikan teknik-teknik dalam analisis data menggunakan interpolasi dan ekstrapolasi untuk menyelesaikan studi kasus yang diberikan menggunakan Program R.

TEORI PENUNJANG

Interpolasi *Piecewise*

Interpolasi Linier

Interpolasi linier *Piecewise* adalah metode interpolasi yang paling sederhana dengan cara menghubungkan titik-titik yang diberikan dengan menggunakan segmen garis lurus. Fungsi yang menginterpolasi data tersebut dinotasikan sebagai l(x). Selain dengan menggunakan interpolasi piecewise linier, interpolasi juga dapat dilakukan dengan menggunakan interpolasi polinomial.

Contoh 1

x	0	1	2	2.5	3	3.5	4
y	2.5	0.5	0.5	1.5	1.5	1.125	0

Interpolasi dapat menggunakan interpolasi piecewise linier dan interpolasi polinomial. Karena terdapat 7 titik data yang diberikan, sehingga interpolasi polinomialnya yaitu P6(x) memiliki derajat 6. Walaupun grafik P6(x) adalah grafik licin, tetapi terdapat perbedaan yang cukup besar dengan l(x), sebagai contoh untuk $0 \leq x \leq 1$lihat hasil plot.

Implementasi pada R

```
> x = c(0,1,2,2.5,3,3.5,4)
> y = c(2.5,0.5,0.5,1.5, 1.5, 1.125, 0)
> polyfit = poly.calc(x,y) #mencari interpolasi piecwise polinomial
> polyfit #hasilnya berupa fungsi
2.5 + 20.41548*x - 53.89663*x^2 + 46.83929*x^3 - 18.6002*x^4 +
3.495238*x^5 - 0.2531746*x^6
> plot(x,y)
> curve(polyfit,add=T) # Polynomial curve fit
> plot.new()
> curve(polyfit,ylim = c(-1, 5)) # Polynomial
```

Interpolasi Kuadratik

Pilihan ketiga untuk menginterpolasi data yang diberikan (Contoh 1) adalah dengan menggunakan Interpolasi kuadratik *piecewise*. Dengan mengunakan metode ini, titik-titik dihubungkan dengan menggunakan polinomial-polinomial interpolasi kuadratik. Fungsi yang dihasilkan dinotasikan q(x) pada [0, 4] yang disebut interpolasi kuadratik piecewise, yang ditunjukkan pada plot. Pada setiap interval bagian [0, 2], [2, 3] dan [3, 4], q(x) menginterpolasi data hanya pada interval bagian tersebut. Dapat dilhat bahwa grafik fungsi q(x) lebih licin dari *l*(x) dan lebih mendekati *l*(x) dibandingkan P6(x). Walaupun demikian, pada x = 2 dan x = 3 fungsi tersebut membentuk sudut sehingga q'(x) diskontinu pada titik x = 2 dan x = 3 (Perhatikan plot).

Interpolasi *Spline*

Misalkan diberikan titik-titik (x_i, y_i) untuk i = 1, 2,…, n, dengan asumsi $x_1 < x_2 < \ldots < x_n$, dan misalkan a = x_1 dan b = x_n. Selanjutnya kita akan menentukan fungsi *s(x)* yang didefinisikan dalam [a, b] yang menginterpolasi data:

$$s(x_i) = y_i,\ i = 1, 2, \ldots, n.$$

Grafik *s(x)* adalah kurva licin yang memenuhi *s'(x)* dan *s''(x)* kontinu, yang memenuhi

1. *s(x)* adalah polinomial derajat ≤ 3 pada setiap interval bagian [x_{j-1}, x_j] untuk j = 2, 3,…,n.
2. *s(x)*, *s'(x)* dan *s''(x)* kontinu untuk a ≤ *x* ≤ b.
3. $s''(x_1) = s''(x_n) = 0$.

Fungsi *s(x)* dikatakan fungsi spline kubik natural yang menginterpolasi data {(x_i, y_i)}.

1. Fungsi *s(x)* dibentuk dengan langkah-langkah sebagai berikut:
   Langkah pertama mengkonstruksi *s(x)* dilakukan dengan menyatakan peubah M_1, M_2,…, Mn, dengan $M_i = s''(x_i)$, i = 1, 2,…, n. (1)
2. Menyatakan *s(x)* dalam bentuk M_i yang merupakan nilai-nilai yang tak diketahui.
3. Selanjutnya nilai-nilai tersebut akan diperoleh dengan menyelesaikan sistem persamaan linier yang akan ditentukan kemudian.
4. Karena *s(x)* adalah polinomial kubik pada setiap interval [x_{j-1}, x_j] maka fungsi *s''(x)* adalah linier pada setiap interval tersebut. Suatu fungsi linier ditentukan dengan menggunakan dua titik yang dalam hal ini, kita gunakan

$$s''(x_{j-1}) = M_{j-1},\ s''(x_j) = M_j \qquad (2)$$

Maka $s''(x) = \frac{(x_j-x)M_{j-1}+(x_j-x)M_j}{x_j-x_{j-1}}$, $x_{j-1} \le x \le x_j$ (3)

Pada titik-titik ujung interval [x_{j-1}, x_j] nilai *s(x)* dinyatakan sebagai berikut

$$s(x_{j-1}) = y_{j-1},\ s(x_j) = y_j \qquad (4)$$

Dengan melakukan manipulasi aljabar (4) diperoleh polinomial kubik sebagai berikut

$$s(x) = \frac{(x_j-x)^3 M_{j-1}+(x-x_{j-1})^3 M_j}{6(x_j-x_{j-1})} + \frac{(x_j-x)y_{j-1}+(x-x_{j-1})y_j}{x_j-x_{j-1}} - \frac{x_j-x_{j-1}}{6}\left[(x_j-x)M_{j-1} + (x - x_{j-1})M_j\right],\ x_{j-1} \le x \le x_j \qquad (5)$$

Dengan mencari turunan kedua dari persamaan (5) diperoleh persamaan (3). Dengan melakukan substitusi langsung dapat ditunjukkan bahwa persamaan (5) memenuhi kondisi interpolasi (4). Karena *s'(x)* kontinu pada [x_0, x_n] maka *s'(x)* pada interval [x_{j-1}, x_j] dan [x_j, x_{j+1}] akan memberikan nilai yang sama pada $x = x_j$ untuk j = 2, 3,…, n-2, sehingga diperoleh sistem persamaan linier berikut

$$\frac{x_j-x_{j-1}}{6}M_{j-1} + \frac{x_{j+1}-x_{j-1}}{3}M_j + \frac{x_{j+1}-x_j}{6}M_{j+1}$$

$$\frac{y_{j+1}-y_j}{x_{j+1}-x_j} - \frac{y_j-y_{j-1}}{x_j-x_{j-1}}, j = 2,3,\ldots,n-1 \qquad (6)$$

Dari sistem persamaan linier (6) terdapat n- 2 persamaan. Dengan menggunakan asumsi bahwa $s''(x_1) = s''(x_n) = 0$, yang berarti $M_1 = M_n = 0$, dan nilai-nilai $M_2, M_3, \ldots, M_{n-1}$ yang merupakan solusi dari sistem persamaan linier (6), akan diperoleh fungsi *s(x)* yang menginterpolasi data yang diberikan.

Contoh 2

Tentukan fungsi spline kubik natural yang menginterpolasi data berikut {(1, 1), (2, 1/2), (3, 1/3), (4, 1/4)}. Banyaknya persamaan adalah n = 4, dan untuk semua j, $x_j - x_{j-1} = 1$. Sistem persamaan linier (6) adalah

$$\frac{1}{6}M_1 + \frac{2}{3}M_2 + \frac{1}{6}M_3 = \frac{1}{3}$$
$$\frac{1}{6}M_2 + \frac{2}{3}M_3 + \frac{1}{6}M_4 = \frac{1}{12}$$

Karena $M_1 = M_4 = 0$ maka diperoleh $M_2 = 1/2$, $M_3 = 0$. Dengan mensubstitusikan nilai-nilai tersebut ke dalam persamaan (5) diperoleh

$$s(x) = \begin{cases} \frac{1}{12}x^3 - \frac{1}{4}x^2 - \frac{1}{3}x + \frac{3}{2}, & 1 \le x \le 2 \\ -\frac{1}{12}x^3 + \frac{3}{4}x^2 - \frac{7}{3}x + \frac{17}{6}, & 2 \le x \le 3 \\ -\frac{1}{12}x + \frac{7}{12}, & 3 \le x \le 4 \end{cases}$$

Implementasi pada R

```
> options(digits=4)
> x = c(1,2,3,4)
> y = c(1,1/2,1/3,1/4)
> require(PolynomF) #load package PolynomF
> polyfit=poly.calc(x,y)
> polyfit
2.083 - 1.458*x + 0.4167*x^2 - 0.04167*x^3
> plot(x,y) #Plot of points
> curve(polyfit, add = T, lty=3) #Polynomial curve fit
> splinefit = splinefun(x,y)
> curve(splinefit, add=T, lty=2) #Spline fit
> legend("bottomright", legend=c("polynom","spline"), lty=c(3:1),bty="n")

> x = c(0,0.5,1,2,3,4)
> y = c(0,0.93,1,1.1,1.15,1.2)
> require(PolynomF)
> polyfit=poly.calc(x,y)
> polyfit
3.638*x - 4.794*x^2 + 2.828*x^3 - 0.7438*x^4 + 0.07105*x^5
> plot(x,y) #Plot of points
> curve(polyfit, add = T, lty=1) #Polynomial curve fit
> splinefit = splinefun(x,y)
> curve(splinefit, add=T, lty=2) #Spline fit
> legend("bottomright", legend=c("polynom","spline"), lty=c(3:1),bty="n")
```

Interpolasi Rasional

Dinotasikan $\rho_{n,m}$ adalah himpunan semua fungsi rasional ρ dari bentuk

$$\rho(x) = \frac{p(x)}{q(x)} \qquad (1)$$

dimana p ∈ π_n, q ∈ π_m sehingga p dan q dapat ditulis sebagai

$$p(x) = a_0 + a_1 x + \cdots + a_n x^n \qquad (2)$$
$$q(x) = b_0 + b_1 x + \cdots + a_m x^m$$

perhatikan bahwa ruang $\rho_{n,m}$ bukan ruang linier karena tidak tertutup dibawah operasi penjumlahan. Perkalian p dan q dalam (1) dengan kuantitas skalar yang sama tidak merubah definisi dari ρ, dan dengan demikian kita dapat membuat normalisasi persamaan (2) dengan menetapkan $b_0 = 1$.

$$\rho(x) = \frac{a_0 + a_1 x + \cdots + a_n x^n}{1 + b_1 x + \cdots + a_m x^m} \qquad (3)$$

dalam persamaan (3) terdapat n + m + 1 parameter dalam definisi dari ρ, dan dengan demikian kita dapat menganggap bahwa n + m + 1 adalah banyaknya titik data yang diperlukan untuk interpolasi oleh suatu anggota dari himpunan $\rho_{n,m}$.

Misalkan $x_0, x_1, \ldots, x_N$, dimana N = n + m adalah titik-titik berbeda dimana nilai-nilai $f(x_0), f(x_1), \ldots, f(x_N)$ diketahui. Agar ρ yang diberikan oleh (1) sesuai dengan f pada titik-titik ini, maka untuk setiap I = 0, 1, ..., N,

$$f(x_i) = \frac{a_0 + a_1 x_i + \cdots + a_n x_i^n}{1 + b_1 x_i + \cdots + a_m x_i^m} \qquad (4)$$

bentuk ini dapat ditulis dengan sistem linier :

$$a_0 + a_1 x_i + \cdots + a_n x_i^n - f(x_i)(b_1 x_i + \cdots + a_m x_i^m) = f(x_i) \qquad (5)$$

tidak seperti interpolasi dan spline, sistem linier dapat tidak memiliki solusi dan solusi dapat tidak unik.

Contoh 3

Tentukan interpolasi rasional dari data f(-1) = 0 , f(0) = 1, f(1) = 1 oleh $\rho_{1,1}$ bentuk fungsi rasional:

$$\rho(x) = \frac{a_0 + a_1 x}{1 + b_1 x}$$

dengan menggunakan titik data yang diberikan, diperoleh persamaan-persamaan sebagai berikut:

$$\frac{a_0 + a_1(-1)}{1 + b_1(-1)} = 0, \frac{a_0 + a_1(0)}{1 + b_1(0)} = 1, \frac{a_0 + a_1(1)}{1 + b_1(1)} = 0$$

sistem persamaan linier tersebut memiliki solusi unik yaitu $a_0 = 1$, $a_1 = 1$, $b_1 = 1$. Sehingga

$$\rho(x) = \frac{1 + x}{1 + x} = 1$$

dengan menghilangkan singularitas pada $x = -1$, diperoleh $\rho(x) = 1$ yang tidak memenuhi kondisi pertama interpolasi.

Implementasi pada R

Implementasi interpolasi rasional pada R memerlukan package pracma

```
> options(digits=4)
> x = c(-1,0,1)
> y = c(0,1,1)
> require(pracma)
Loading required package: pracma
Attaching package: 'pracma'
The following object is masked _by_ '.GlobalEnv':
polyfit
The following object is masked from 'package:PolynomF':
integral
Warning message:
```

```
package 'pracma' was built under R version 3.0.3
> ratinterp(x,y,0.5)
[1] 1
```

Ekstrapolasi

Diberikan tabel data yang menyatakan nilai-nilai *x* dan *y* = f(x).

x	x_0	x_1	x_2	x_3	...	x_n
y	y_0	y_1	y_2	y_3	...	y_n

Proses komputasi y terhadap x dimana $x_i \le x \le x_{i+1}$, i = 0,1,2,…,n−1 adalah interpolasi. Jika $x < x_0$ atau $x > x_n$, maka proses dinamakan ekstrapolasi. Penerapan ekstraspolasi diantaranya dalam formula integrasi Newton-Cotes, metode integrasi Romberg.

Contoh 4

Implementasi pada R

```
> options(digits=4)
> x = c(0.4, 1)
> y = sin(x)
> xExp = c(0.2,0.3,1.5,2)
> require(PolynomF)
Loading required package: PolynomF
> polyfit=poly.calc(x,y)
> polyfit
0.08805 + 0.7534*x
> polyfit(xExp)
[1] 0.2387 0.3141 1.2182 1.5949
> sin(xExp)
[1] 0.1987 0.2955 0.9975 0.9093
> splinefit = splinefun(x,y)
> splinefit(xExp)
[1] 0.2387 0.3141 1.2182 1.5949
```

```
> x = c(0.4, 0.6, 0.8, 1)
> y = sin(x)
> polyfit=poly.calc(x,y)
> polyfit
-0.004209 + 1.026*x - 0.05307*x^2 - 0.1268*x^3
> polyfit(xExp)
[1] 0.1978 0.2953 0.9867 0.8200
> sin(xExp)
[1] 0.1987 0.2955 0.9975 0.9093
> splinefit = splinefun(x,y)
> splinefit(xExp)
[1] 0.1978 0.2953 0.9867 0.8200
> plot(x,y)
> curve(polyfit, add = T, lty=3) #Polynomial curve fit
> curve(splinefit, add=T, lty=2) #Spline fit
> legend("bottomright", legend=c("polynom","spline"), lty=c(3:1),bty="n")
```

LAPORAN PENDAHULUAN

1. Jelaskan secara singkat bagaimana proses interpolasi spline *piecewise*!
2. Jelaskan apa yang dimaksud interpolasi rasional!
3. Jelaskan secara singkat bagaimana proses ekstrapolasi!

MATERI PRAKTIKUM

1. Buat program komputer menggunakan bahasapemograman R untuk menentukan interpolasi polinomial, piecwise, spline, dan rasional. Kemudian periksa dan bandingkan akurasi dari masing-masing metode untuk beberapa kasus seperti

    a. Diketahui beberapa titik x dan fungsi y diketahui

    b. Diketahui beberapa pasang titik x dan y

2. Lakukan ekstrapolasi di titik $x_1 < x$ dan $x_1 > x$ pada kasus yang dikerjakan di soal sebelumnya.

DAFTAR PUSTAKA

1. Atkinson K. 1994. Elementary Numerical Analysis, Second edition. Wiley
2. Victor A. Bloomfield. 2014. *Using R for Numerical Analysis in Science and Engineering*. 1 edition. Chapman and Hall/CRC