Section: Numerical integration - Lecture 1

Goal: Given a function $f(x)$, $a \leq x \leq b$, we want to evaluate or approximate $\int_a^b f(x)dx$.

[1] Integration via Polynomial Interpolation

Suppose that we want to approximate $\int_a^b f(x)dx$. We can select nodes $x_0, x_1, \cdots, x_n$ in $[a, b]$ and set up a Langrange interpolation process by defining

$$\ell_i(x) = \prod_{\substack{j=0\\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j}, \quad 0 \leq i \leq n. \quad p(x) = \sum_{i=0}^{n} f(x_i)\ell_i(x).$$

Then

$$\int_a^b f(x)dx \approx \int_a^b p(x)dx = \sum_{i=0}^{n} f(x_i) \int_a^b \ell_i(x)dx.$$

We obtain that

$$\int_a^b f(x)dx \approx \sum_{i=0}^{n} A_i f(x_i), \text{ where } A_i = \int_a^b \ell_i(x)dx$$

This gives us a fourmula how an integration can be approximate numerically.

So, the integration depends on how the nodes $x_0, x_1, \cdots, x_n$ being chosen. An example is the so-called **Newton-Cotes formula** where the nodes are equally spaced.

We define a function to evaluate the integral.

We then evaluate the absolute difference between the numerical approximated integration and the exact integration value. We "hope" that as the number of points increases, the error will decreases.

[1.1] Midpoint rule

Suppose we want to use one single point to approximate the integral, then $n = 0$ and the natural choice is the midpoint $x_0 = \frac{a+b}{2}$. Therefore

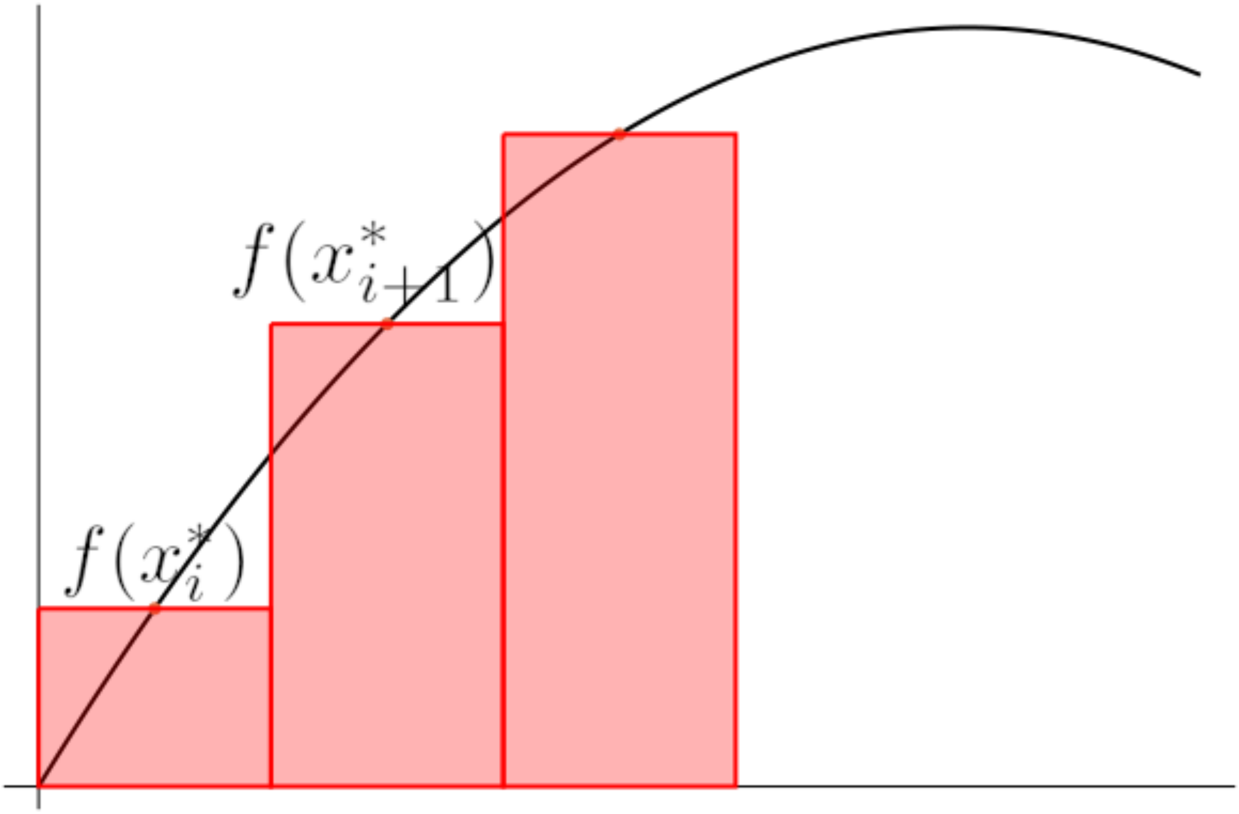
$$\int_a^b f(x)dx \approx (b - a)f\left(\frac{a + b}{2}\right)$$

More generally, consider equally spaced nodes $a = x_0 < x_1 \cdots < x_n = b$, where $x_i = a + i\Delta x$, $\Delta x = \frac{b-a}{n}$. If we use midpoint rule in each interval $[x_{i-1}, x_i]$, then

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} \Delta x f(x_i^*)$$

where $x_i^* = (x_{i-1} + x_i)/2$.

An illustration of the midpoint rule is shown in the following figure:



Example 1

Evaulate $\int_{-1}^{1} e^x dx$ using the Midpoint rule with equally spaced nodes for $n = 10, 100, 1000$. Note that the exact solution for this integral is $e - \frac{1}{e}$.

```julia
exact = exp(1) - exp(-1)
```

```
2.3504023872876028
```

We define a function to evaluate the integral.

```julia
function Midpoint(f::Function,n)
    xi = range(-1,1;length=n+1)
    xi_s = zeros(n)
    for i=1:n
        xi_s[i] = (xi[i] + xi[i+1])/2
    end
    sum=0
    for i=1:n
        sum=sum+(f(xi_s[i]))*(2/n)
    end
    return sum
end
```

```
Midpoint (generic function with 1 method)
```

For different number of points, the error between the numerical approximation and the exact value are shown in the following:

```julia
abs(Midpoint(exp,10)-exact)
```

```
0.003912771899297507
```

```julia
abs(Midpoint(exp,100)-exact)
```

```
3.917291610378015e-5
```

```julia
abs(Midpoint(exp,1000)-exact)
```

```
3.9173368460154734e-7
```

[1.2] Trapezoidal rule

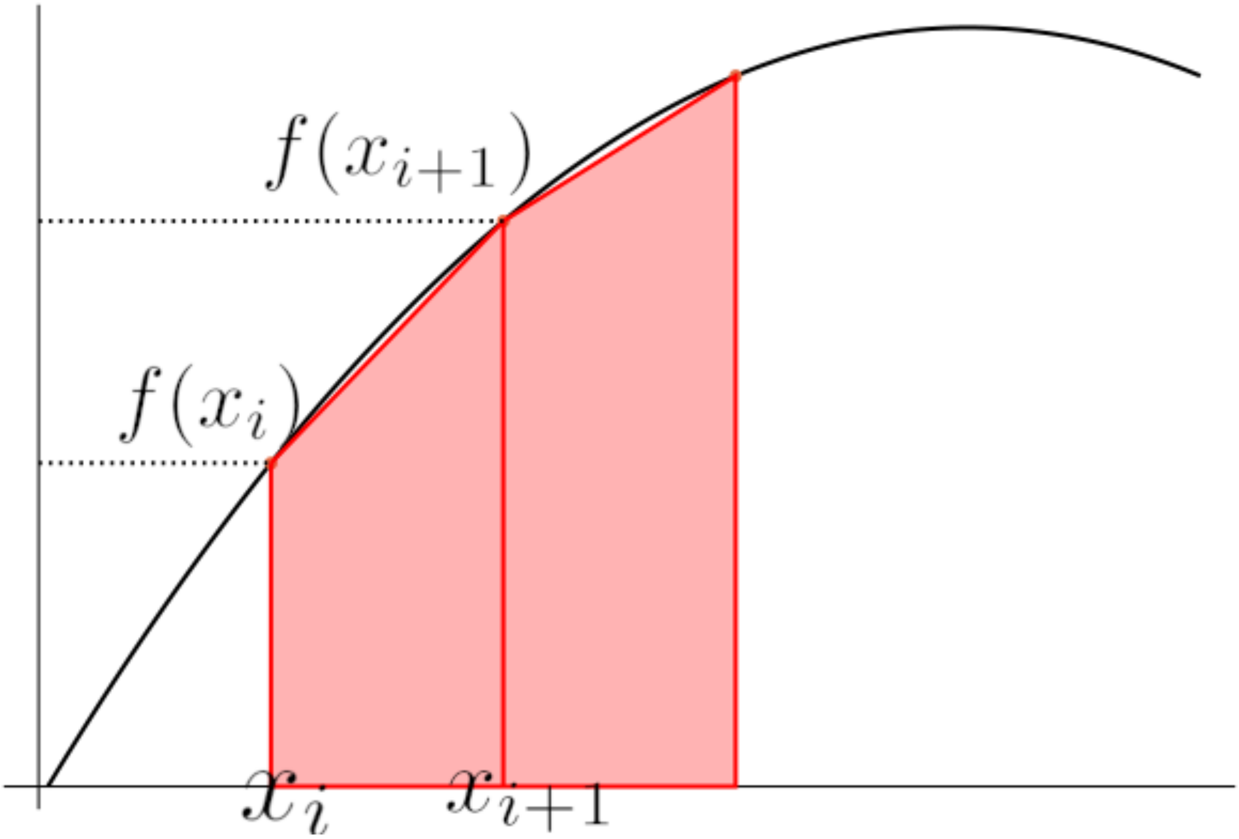
Suppose we want to approximate the integral using two points and we choose the two end points $x_0 = a$ and $x_1 = b$, we obtian

$$\int_a^b f(x)dx \approx \frac{b-a}{2}[f(a)+f(b)].$$

More generally, if we choose nodes $x_0, x_1, \cdots, x_n$ and we use trapezoidal rule to approximate the integral in each subinterval, we have

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} \frac{f(x_{i-1})+f(x_i)}{2}\Delta x_i, \text{ where } \Delta x_i = x_i - x_{i-1}.$$

An illustration of the trapezoidal rule is shown in the following figure:



See Trapezoidal rule in wiki for further details.

Example 1

Evaulate $\int_{-1}^{1} e^x dx$ using the trapezoidal rule with equally spaced nodes for $n = 10, 100, 1000$.

```julia
function Trapezoidal(f::Function,n::Int)
    xi = range(-1,1;length=n+1)
    sum=0
    for i=1:n
        sum=sum+((f(xi[i])+f(xi[i+1]))/2)*(xi[2]-xi[1])
    end
    return sum
end
```

```
Trapezoidal (generic function with 1 method)
```

```julia
abs(Trapezoidal(exp,10)-exact)
```

```
0.007829456477730271
```

```julia
abs(Trapezoidal(exp,100)-exact)
```

```
7.834622393865232e-5
```

```julia
abs(Trapezoidal(exp,1000)-exact)
```

```
7.834674118356588e-7
```

[1.3] Simpson's rule

Suppose we want to use three points to approximate the integral and we choose the two end points and the midpoint, that is, $x_0 = a, x_1 = \frac{a+b}{2}$ and $x_2 = b$, this gives us

$$\int_a^b f(x)dx \approx \frac{b-a}{6}\left[f(a) + 4f\left(\frac{a+b}{2}\right) + f(b)\right]$$

This is called the Simpson's rule.

More genrally, we have

$$\int_a^b f(x)dx \approx \frac{b-a}{3n}\left[f(x_0) + 4f(x_1) + 2f(x_2) + 4f(x_3) + 2f(x_4) + \cdots + 4f(x_{n-1}) + f(x_n)\right]$$

for equally spaced nodes $x_0, x_1, \cdots, x_n$ and even n.

See Simpson's rule in wiki for further details.

Example 3

Evaulate $\int_{-1}^{1} e^x dx$ using the Simpson's rule with equally spaced nodes for $n = 10, 100, 1000$.

We define a function to evaulate the integral.

```julia
function Simpson(f::Function, n::Int)
    x_i = range(-1,1;length=n+1)
    sum=0
    for i=1:n+1
        if (i==1 || i== n+1)
            sum = sum + (2/(3*n))*f(x_i[i])
        elseif (i%2 == 0)
            sum = sum + 4*(2/(3*n))*f(x_i[i])
        else
            sum = sum + 2*(2/(3*n))*f(x_i[i])
        end
    end
    return sum
end
```

```
Simpson (generic function with 1 method)
```

```julia
abs(Simpson(exp,10)-exact)
```

```
2.079339388094681e-5
```

```julia
abs(Simpson(exp,100)-exact)
```

```
2.089148409822883e-9
```

```julia
abs(Simpson(exp,1000)-exact)
```

```
2.0961010704922955e-13
```

Conclusion

A comparison between the results of the above three methods is shown in the following

n\Method	Midpoint	Trapezoidal	Simpson's
10	3.91e-3	7.83e-3	2.08e-5
100	3.92e-5	7.83e-5	2.09e-9
1000	3.92e-7	7.83e-7	2.10e-13

In particular, one can see that the Midpoint rule and Trapezoidal rule both have error in the order $O(n^{-2})$ while the error in Simpson's method is $O(n^{-4})$.

Furthermore, as we pointed out in the very begining of this lecture, the Newton Cotes formula are groups of formula of numerical quadrature rule in equally spaced points. See Newton–Cotes formulas in wiki for further details.