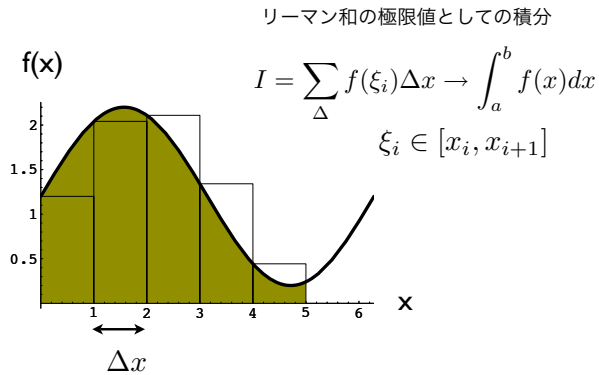


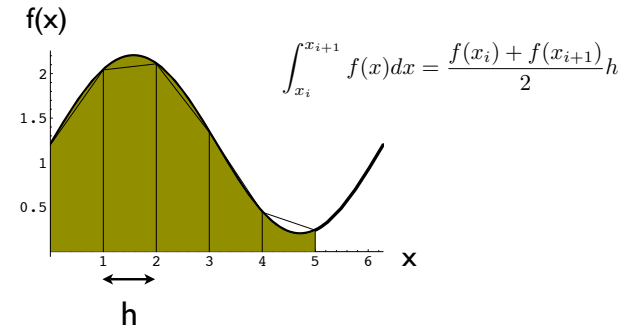
## 関数の積分

与えられた関数の（リーマン）積分 = 区間面積



## 台形公式

- 被積分関数を1次関数の接合として近似し、台形の面積の和として積分を求める方法



$$I = \int_a^b f(x) dx = \left\{ \frac{f(a) + f(b)}{2} + \sum_{i=1}^{n-1} f(a + i \times h) \right\} h$$

## 台形公式の誤差 1

- 関数を直線で近似することによる誤差が生じる（関数が一次関数でない場合）

真の値

$$I = \int_{x_i}^{x_i+h} f(x) dx = \int_0^h f(x_i + x) dx$$

$$= \int_0^h \left[ f(x_i) + f'(x_i)x + \frac{f''(x_i)}{2}x^2 \dots \right] dx$$

$$= f(x_i)h + \frac{f'(x_i)}{2}h^2 + \frac{f''(x_i)}{6}h^3 + O(h^4)$$

近似値

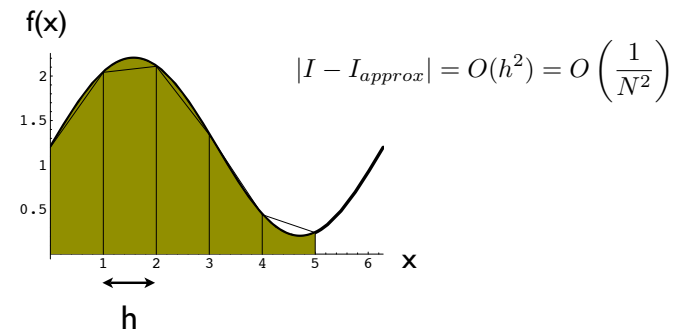
$$I_{approx} = \frac{h}{2} [f(x_i) + f(x_i + h)]$$

$$= \frac{h}{2} \left[ f(x_i) + f(x_i) + f'(x_i)h + \frac{f''(x_i)}{2}h^2 \dots \right]$$

$$= f(x_i)h + \frac{f'(x_i)}{2}h^2 + \frac{f''(x_i)}{4}h^3 + O(h^4)$$

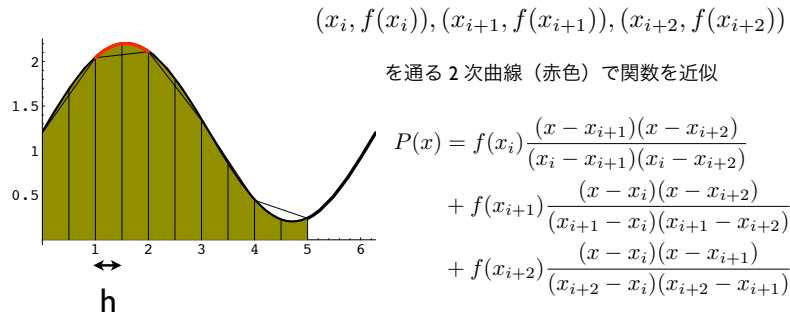
## 台形公式の誤差 2

- 一つの台形積分にかかる誤差のオーダーは、刻み幅  $h$  の 3 乗に比例
- 積分範囲を  $N$  分割した場合  $h = (b - a)/N$ 。累積誤差は  $h$  の 2 乗に比例



## シンプソンの公式

- 被積分関数を2次関数の接合として近似し、2次曲線の面積の和として積分を求める方法



ラグランジュの補完

## シンプソンの公式 2

- 2次関数による近似値。誤差は  $h$  の5乗のオーダー

$$I_{approx} = \int_{x_i}^{x_{i+2}} P(x) dx = \frac{h}{3} \left[ f(x_i) + 4f\left(\frac{x_i + x_{i+2}}{2}\right) + f(x_{i+2}) \right]$$

$$\left| \int_{x_i}^{x_{i+2}} f(x) dx - I_{approx} \right| = O(h^5)$$

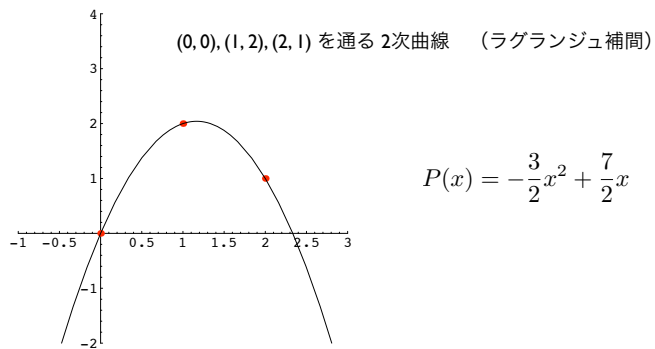
- 累積誤差は  $h$  の4乗のオーダー

$$|I - I_{approx}| = O(h^4) = O\left(\frac{1}{N^4}\right)$$

シンプソンの公式の区間分割数は偶数であることに注意

## 多項式による補間

- $N$  個の離散点  $(x_i, y_i)$  を通る  $N-1$  次の多項式は唯一存在する ( $i = 1, 2, \dots, N$ )



## 関数へのポインタ

- 関数へのポインタを用いて可搬性のある関数を定義する

```

int main (int argc, const char * argv[]) {
    double a = 0.0, b = 1.0;
    int n = 100;
    double (*myfunc)(double);

    myfunc = fn;
    printf("%f, %f\n", a, (*myfunc)(a)); // a での関数 myfunc の値を表示
    // リーマン和を計算する関数 riemann_sum の呼び出し
    printf("%f, %f, %f\n", a, b, riemann_sum(*myfunc, a, b, n));

    return 0;
}

double fn(double x)
{
    return 1.0/(1 + x*x);
}
  
```

## 積分を行う関数の定義

```
double riemann_sum( double (*func)(double), double a, double b, int n)
{
    double h, sum = 0.0;
    int i;

    .....

    return sum*h;
}

double trapzd( double (*func)(double), double a, double b, int n)
{
    .....
}

double simpson( double (*func)(double), double a, double b, int n)
{
    ...
}
```

## 問題 1

以下の積分それぞれについて、

1) リーマン和、2) 台形公式、3) シンプソンの公式、を用いた数値積分を求めよ。

区間分割数  $N$  を 10, 100, 1000 と変えたとき、真の積分値と 3 つの方法による近似値との誤差を比較すること。

$$\int_0^1 x^2 dx$$

$$\int_0^1 \frac{1}{1+x^2} dx$$

$$\int_0^\pi \sin(x) dx$$

$$\int_{-1}^1 \sqrt{1-x^2} dx$$

	$N=10$	$N=100$	$N=1000$
1) Riemann sum	...	...	...
2) Trapezoidal	...	...	...
3) Simpson	...	...	...
Error	...	...	...