

情報科学

$$\text{Int}[19]= \int \sqrt{1+x} \sqrt{2+x} dx$$

$$\frac{1}{3} (\sqrt{1+x} \sqrt{2+x} (3+2x) - \text{ArcSinh}[\sqrt{1+x}])$$

$$\text{Int}[20]= \int \sqrt{x+\sqrt{1+x}} dx$$

$$\frac{1}{12} \sqrt{x+\sqrt{1+x}} (-3+8x+2\sqrt{1+x}) + \frac{5}{8} \text{Log}[1+2\sqrt{1+x}+2\sqrt{x+\sqrt{1+x}}]$$

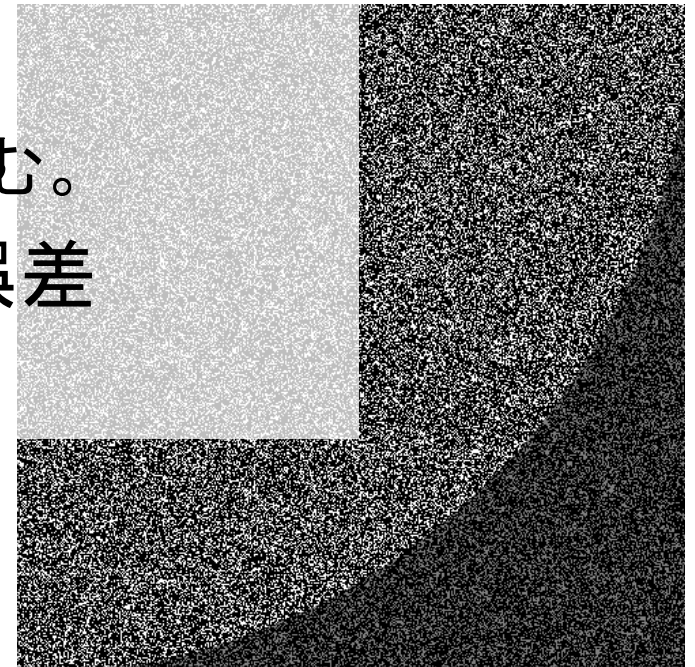
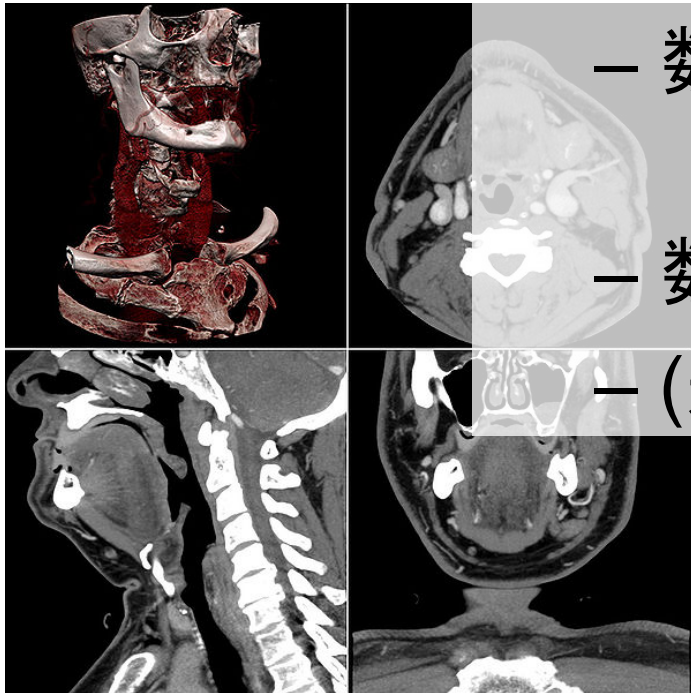
- 第5章 計算量とアルゴリズム
 - ビン整列法と空間計算量
- 第6章 数値計算

– 数値積分

- モンテカルロ法を含む。

– 数値計算における誤差

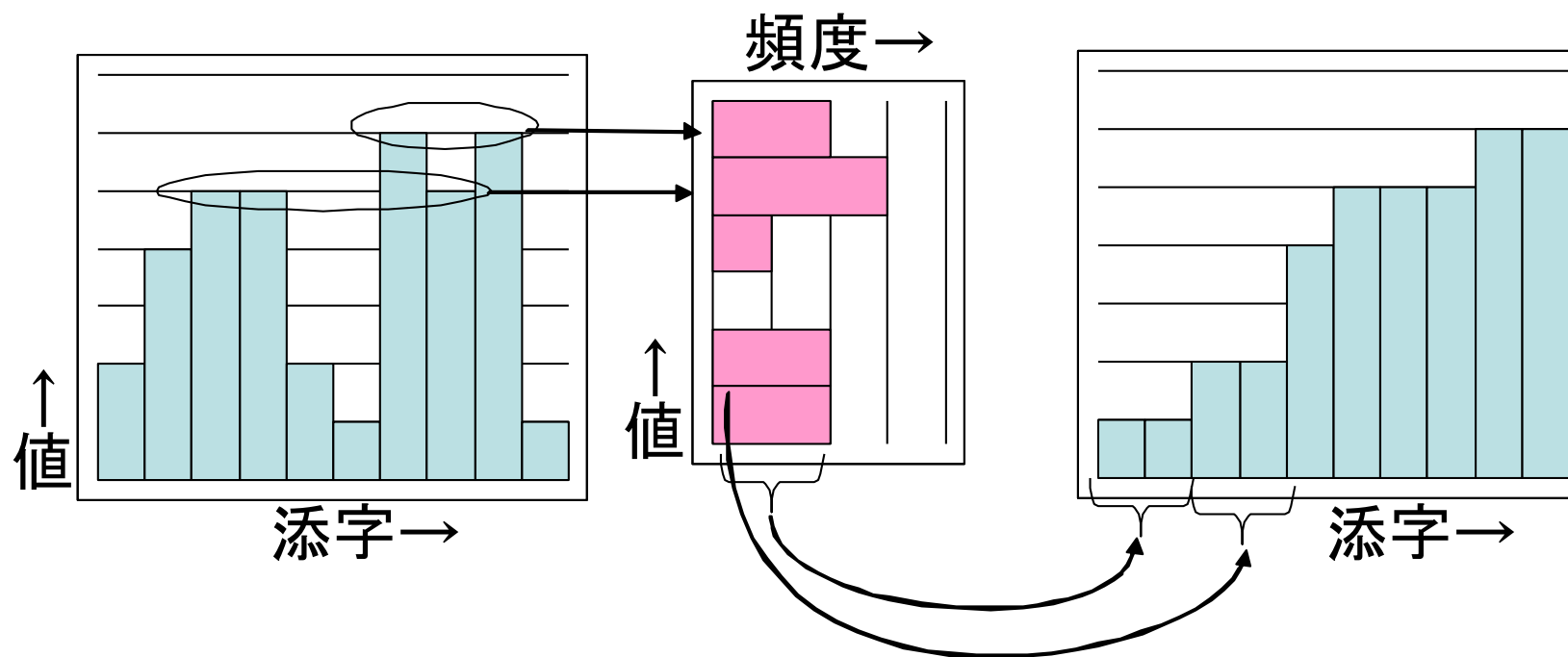
– (連立方程式)



ビン整列法

- 考え方

- 頻度分布(ヒストグラム)を作る
- 各値の頻度だけ値を並べる



ビン整列法

1. 大きな配列を用意し、全て0にしておく
2. 列の各要素 x について、配列の x 番目を1増やす
3. 配列を順に調べ、 x 番目が n ならば x を n 個並べる

9 3 1 4 6 4 3 1 3 9 4 8 7 3 3 1 2



0 1 2 3 4 5 6 7 8 9



計算量は?

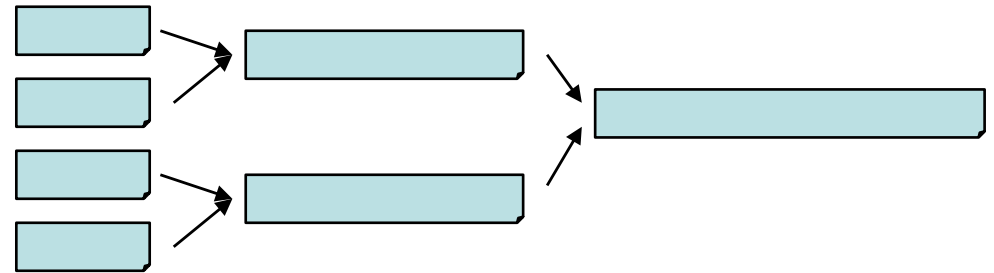
時間計算量と空間計算量

- いままでの計算量は「計算回数」に注目していた
→ 時間計算量という
- 配列やリストのようなデータ構造を使う場合は、必要とする記憶領域の大きさも問題となる
→ 空間計算量
- 考え方は時間計算量と同様

空間計算量の例

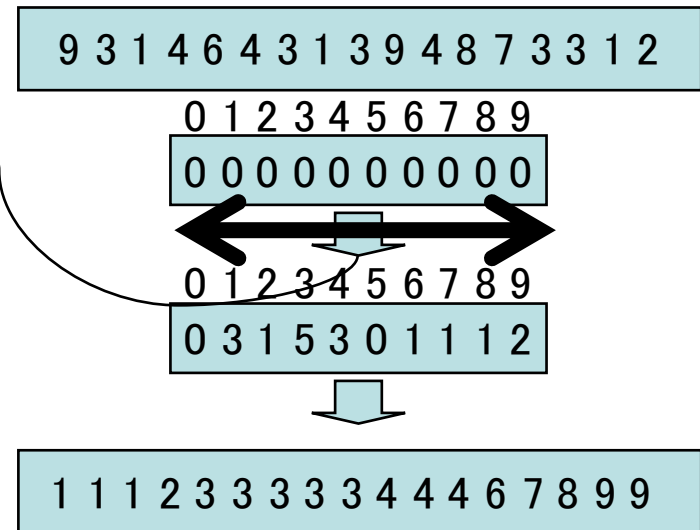
- 併合ソート

- データ個数 n の2倍
→ $O(n)$



- ビンソート

- データ範囲の幅 k とデータの個数 n の大きい方 → $O(k+n)$
- 範囲が狭い場合のみ使える



情報科学

$$\text{Int}[19]= \int \sqrt{1+x} \sqrt{2+x} dx$$

$$\frac{1}{12} \sqrt{1+x} \sqrt{2+x} (3+2x) - \text{ArcSinh}[\sqrt{1+x}]$$

$$\text{Int}[20]= \int \sqrt{x+\sqrt{1+x}} dx$$

$$\frac{1}{12} \sqrt{x+\sqrt{1+x}} (-3+8x+2\sqrt{1+x}) + \frac{5}{8} \text{Log}[1+2\sqrt{1+x}+2\sqrt{x+\sqrt{1+x}}]$$

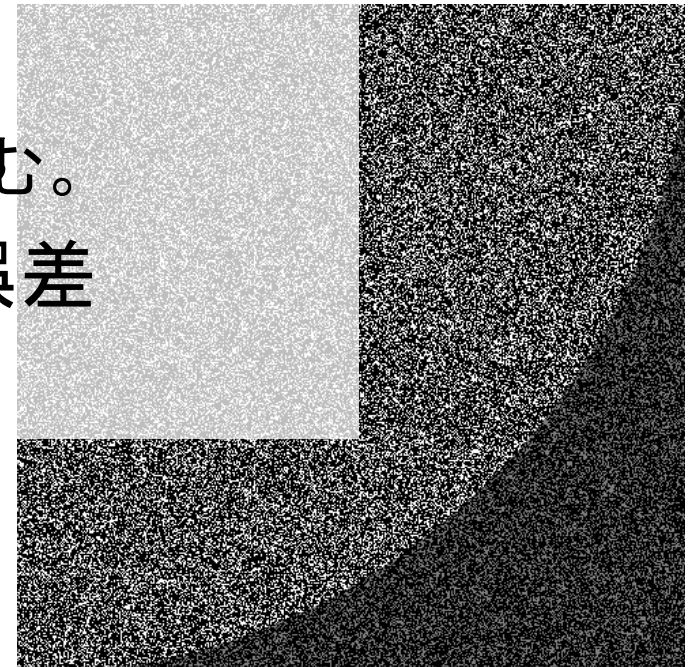
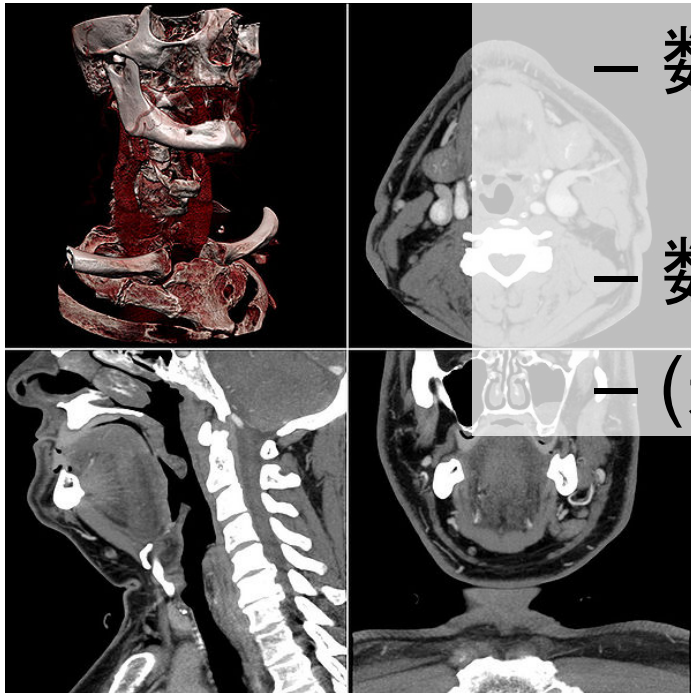
- 第5章 計算量とアルゴリズム
 - ビン整列法と空間計算量
- 第6章 数値計算

– 数値積分

- モンテカルロ法を含む。

– 数値計算における誤差

– (連立方程式)



数値計算とは？

Mathematica を
実行してみよう

- 代数演算

- 等式を一定の規則のもとに変形して解析的に求める

- 数値計算

- 反復計算によって近似的に解を求める
- 解析的に求めることが困難な方程式を解く場合に用いる

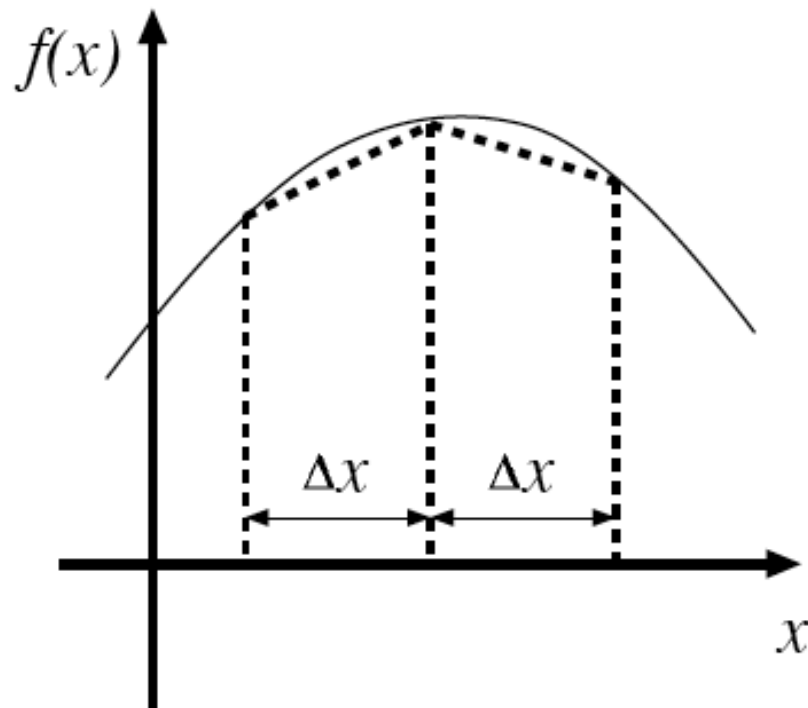
例) 物理学、化学、天文学などにおける数値積分
や微分方程式の解法など

数値積分

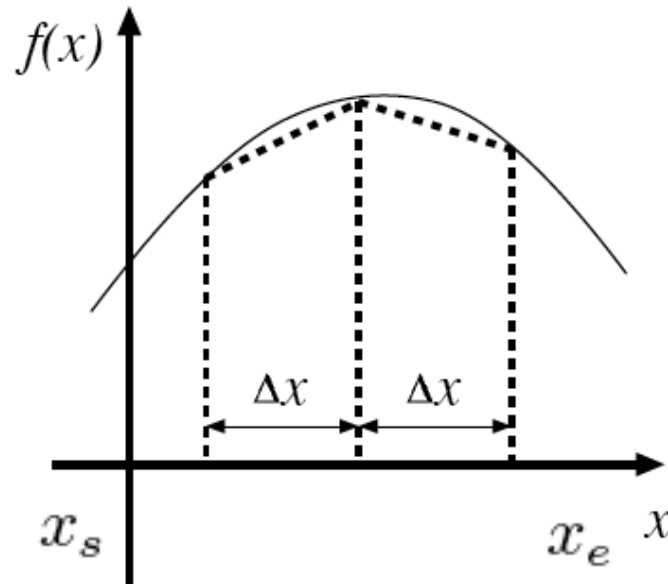
1. 台形公式
2. シンプソン公式
3. モンテカルロ法

台形公式

- 積分を区分線形(piecewise linear)で近似する方法
- 全積分区間を n 等分し各部分区間 Δx の積分を台形の面積として計算し総和をとる



台形公式(続き)



ある関数 $f(x)$ の x_s から x_e までの
積分の近似値を求めるには、

各部分区間 $\Delta x = \frac{x_e - x_s}{n}$

における台形の総和で近似する

$$\begin{aligned}\int_{x_s}^{x_e} f(x)dx &\approx \sum_{i=0}^{n-1} \frac{1}{2} \{f(x_s + i\Delta x) + f(x_s + (i+1)\Delta x)\} \Delta x \\ &= \Delta x \left\{ \frac{f(x_s) + f(x_e)}{2} + \sum_{i=1}^{n-1} f(x_s + i\Delta x) \right\}\end{aligned}$$

```
def f(x)
  x/((x+1.0)*(x+2.0))
```

```
end
```

$$\Delta x \left\{ \frac{f(x_s) + f(x_e)}{2} + \sum_{i=1}^{n-1} f(x_s + i\Delta x) \right\}$$

```
def trapezoid(xs,xe,n)
```

```
  deltax = (xe-xs)*1.0/n
```

```
  sum = (f(xs)+f(xe))/2.0
```

```
  for i in 1..(n-1)
```

```
    sum = sum + f(xs+i*deltax)
```

```
  end
```

```
  deltax * sum
```

```
end
```

trapezoid.rb

練習

- 以下の積分を台形公式で近似して円周率を求めよ。 n を引数として円周率を返す関数を定義せよ。

$$\pi = 4 \int_0^1 \frac{1}{1+x^2} dx$$

進捗確認(台形公式)

- 計算できたら `pi_trapezoid(100)` の値と π が一致している桁数を投票せよ (π の値は定数 `PI` に)

```
cd Downloads
ruby c.rb 桁数
```

```
include(Math)
load("./trapezoid.rb")
def f(x)
  ...
end

def pi_trapezoid(n)
  4.0 * trapezoid(...)
end
```

trapezoid.rb
を変更せずに
fを再定義
する

pi.rb

収束に関するコメント

$$f(x+t) = f(x) + f'(x)t + \frac{f''(x)}{2}t^2 + \dots$$

$$f'(x) = \frac{f(x+\Delta x) - f(x)}{\Delta x} - \frac{f''(x)}{2}\Delta x - \dots$$

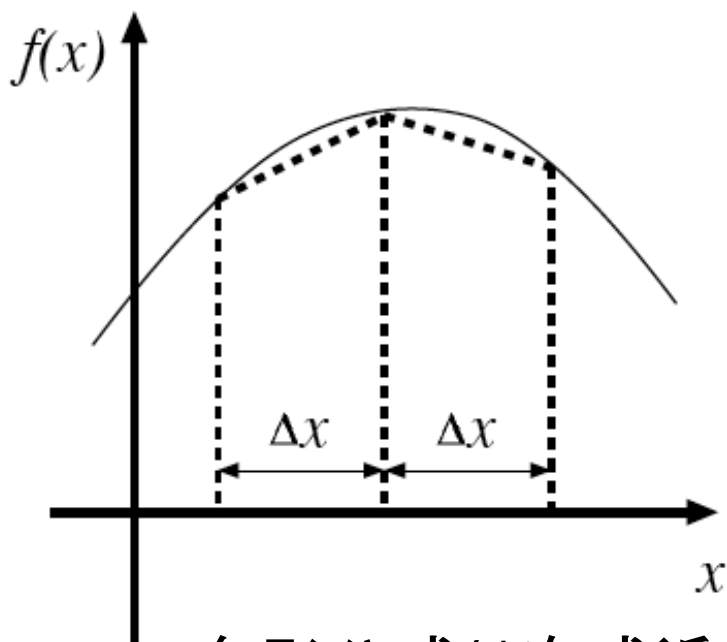
$$\begin{aligned} \int_x^{x+\Delta x} f(x)dx &= \int_0^{\Delta x} f(x+t)dt = f(x)\Delta x + f'(x)\frac{\Delta x^2}{2} + \frac{f''(x)}{2}\frac{\Delta x^3}{3} + \dots \\ &= f(x)\Delta x + \left(\frac{f(x+\Delta x) - f(x)}{\Delta x} - \frac{f''(x)}{2}\Delta x - \dots \right) \frac{\Delta x^2}{2} + \frac{f''(x)}{2}\frac{\Delta x^3}{3} + \dots \\ &= \frac{f(x) + f(x+\Delta x)}{2}\Delta x + O(\Delta x^3) \end{aligned}$$

Δx^3 に比例した項

- 刻み一つごとに Δx^3 に比例した誤差
- これが積もると Δx^2 に比例した誤差になる¹⁴

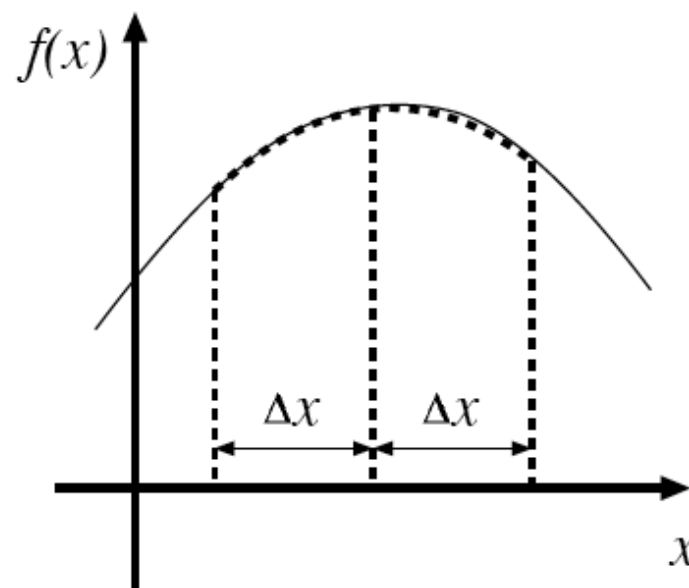
シンプソン公式

- 積分を1次式ではなく2次式で近似する方法



台形公式(1次式近似)

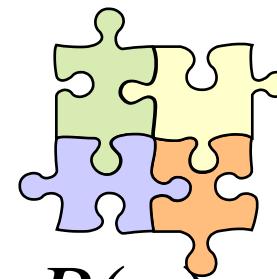
$$\frac{1}{2} \{f(x) + f(x + \Delta x)\} \Delta x$$
$$\Delta x = \frac{x_e - x_s}{n}$$



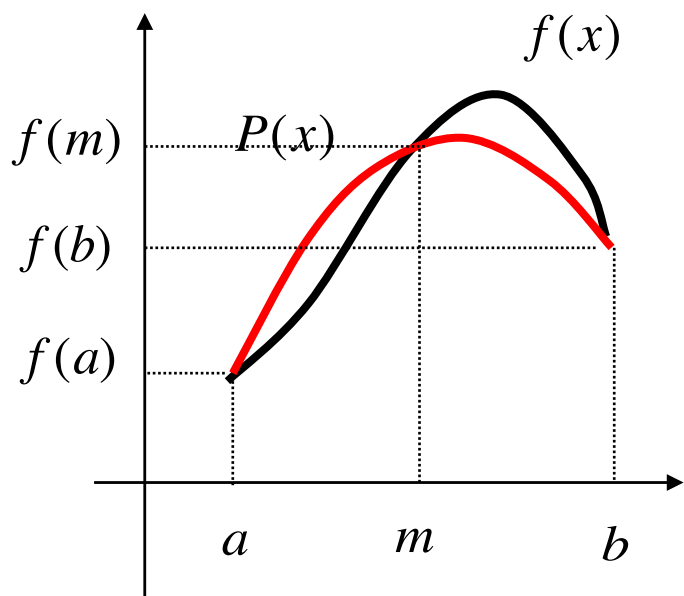
シンプソン公式(2次式近似)

$$\frac{1}{3} \{f(x) + 4f(x + \Delta x) + f(x + 2\Delta x)\} \Delta x$$
$$\Delta x = \frac{x_e - x_s}{2n}$$

ラグランジュ補間



- ある関数 $f(x)$ を3点 a, m, b で補間する2次関数 $P(x)$

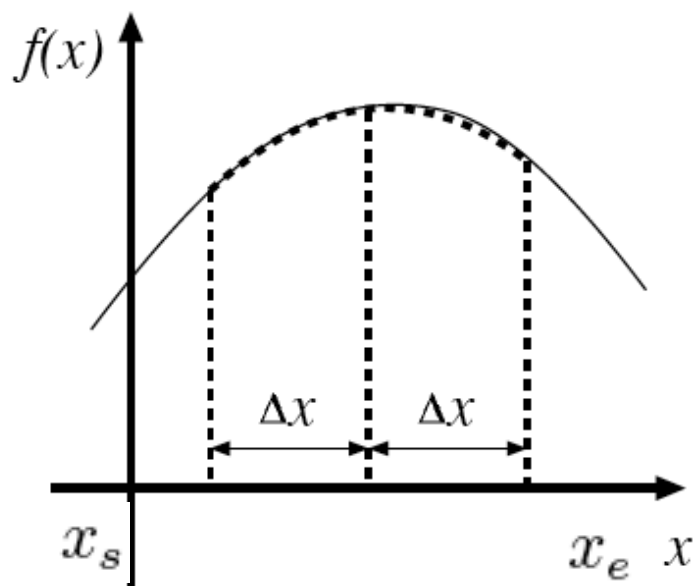


$$P(x) = \frac{(x-b)(x-m)}{(a-b)(a-m)} f(a) + \frac{(x-a)(x-b)}{(m-a)(m-b)} f(m) + \frac{(x-a)(x-m)}{(b-a)(b-m)} f(b)$$

$$m = \frac{(b+a)}{2} \Rightarrow \int_a^b P(x) dx = \frac{(b-a)}{6} [f(a) + 4f(m) + f(b)]$$

$$\begin{array}{l} a \rightarrow x \\ b \rightarrow x + 2\Delta x \\ m \rightarrow x + \Delta x \end{array} \Rightarrow = \frac{1}{3} \{f(x) + 4f(x + \Delta x) + f(x + 2\Delta x)\} \Delta x$$

シンプソン公式(続き)



部分区間 $\Delta x = \frac{x_e - x_s}{2n}$ とし

区間 $[2\Delta x]$ の部分の積分近似値は

$$\frac{1}{3} \{f(x) + 4f(x + \Delta x) + f(x + 2\Delta x)\} \Delta x$$

となるため、 x_s から x_e までの

積分近似値の総和は

$$\begin{aligned} \int_{x_s}^{x_e} f(x) dx &\approx \sum_{i=0}^{n-1} \frac{1}{3} \{f(x_s + 2i\Delta x) + 4f(x_s + (2i+1)\Delta x) + f(x_s + (2i+2)\Delta x)\} \Delta x \\ &= \frac{\Delta x}{3} \left\{ f(x_s) + 4f(x_s + \Delta x) + f(x_e) + \sum_{i=1}^{n-1} (2f(x_s + 2i\Delta x) + 4f(x_s + (2i+1)\Delta x)) \right\} \end{aligned}$$

練習6.2 +α

- 練習6.2 (注意: $\Delta x = (x_e - x_s)/(2n)$)
- 以下の積分をシンプソン公式で近似して円周率を求めよ。 n を引数として円周率を返す関数を定義せよ。

進捗確認(シンプソン法)

$$\pi = 4 \int_0^1 \frac{1}{1+x^2} dx$$

- 計算できたらn=100のときの値とπが一致している桁数を投票せよ(πの値は定数PIに)

投票して下さい
cd Downloads
ruby c.rb 桁数

台形公式との近づき方の違いは?

乱数とモンテカルロ法

1. モンテカルロ法
2. 乱数とは？
 - ・ 一様乱数
 - ・ 正規乱数
3. 疑似乱数



モンテカルロ法

- 数学的問題の解法に乱数(疑似乱数)を用いる方法の総称をモンテカルロ法という

モンテカルロとはモナコにある賭博で有名な都市(賭博場が国営)であり、多くの賭博は乱数を利用して行われるためこのように呼ばれる

- 一般に以下の2種類の問題を扱う

- 決定論的問題

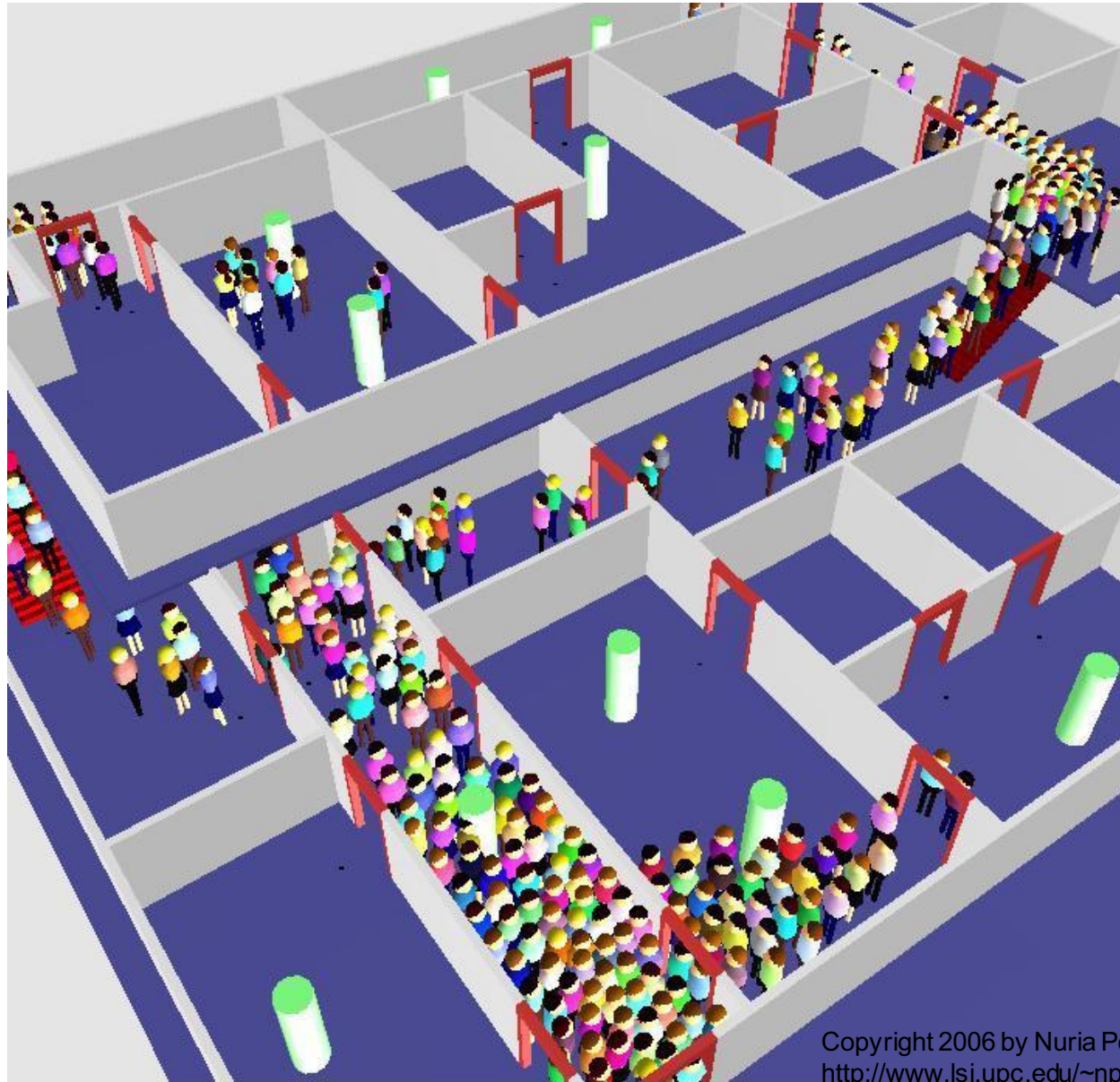
- 乱数を用いる多次元積分 (例)円周率の近似計算

- 非決定論的(確率的変動を含む)問題

- 自然科学・社会科学におけるシミュレーション
(実際に実験が困難であったり多大な費用がかかる場合)
例えば、ランダムウォーク(ブラウン運動)



非決定論的問題の例 群集シミュレーション



決定論的問題: 円周率の近似計算

- 平面上の正方領域

$$P = \{0 \leq x \leq 1, 0 \leq y \leq 1\}$$

にランダムに n 個の点を配置し、

その中で、四半円領域

$$Q = \{x^2 + y^2 < 1, 0 \leq x \leq 1, 0 \leq y \leq 1\}$$

にある点の数 m を求める。

- 一様乱数を用いる場合、

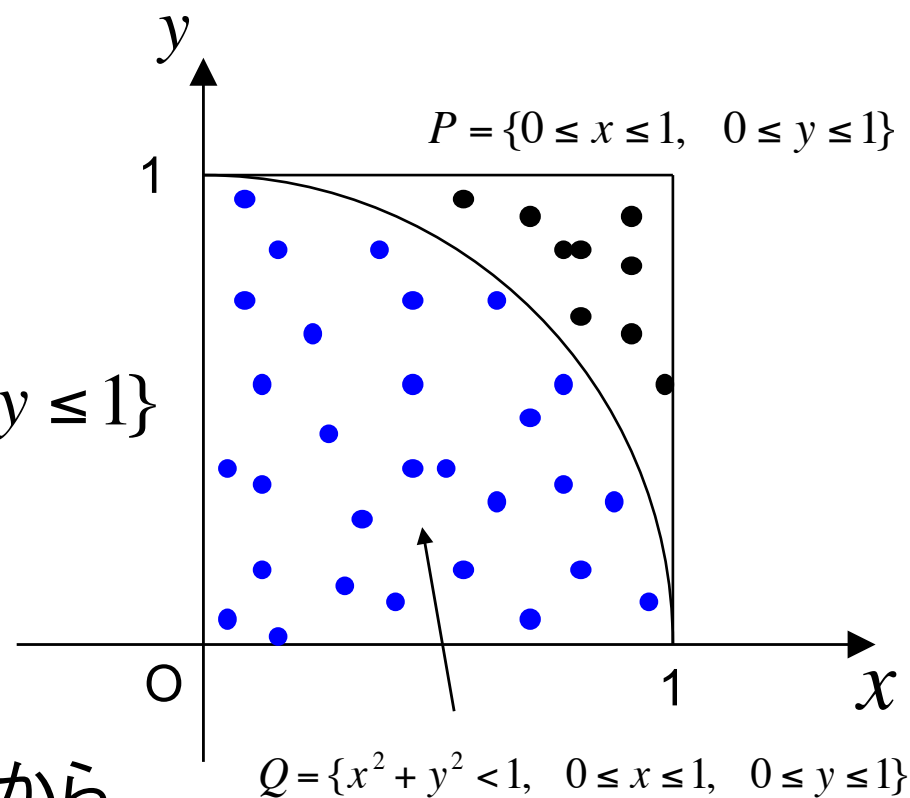
配置される点の数は

領域の面積に比例するはずであるから、

$$m/n \approx Q/P = \pi/4$$

- つまり円周率は $\pi \approx 4m/n$ で近似できる。

ただし、精度を上げるためには、非常に多くの点を用いる必要がある



$x_i \in [0,1)$ $y_i \in [0,1)$ $i = 1..n$ のうち

- $x_i^2 + y_i^2 \geq 1$ を満たす点
- $x_i^2 + y_i^2 < 1$ を満たす点

名前をmcplotに変える

```
def montecarlo(n)
```

```
  m = 0
```

a=make2d(500,500)

```
  for i in 1..n
```

```
    x = rand() # random number in [0,1)
```

```
    y = rand()
```

```
    if x*x + y*y < 1.0
```

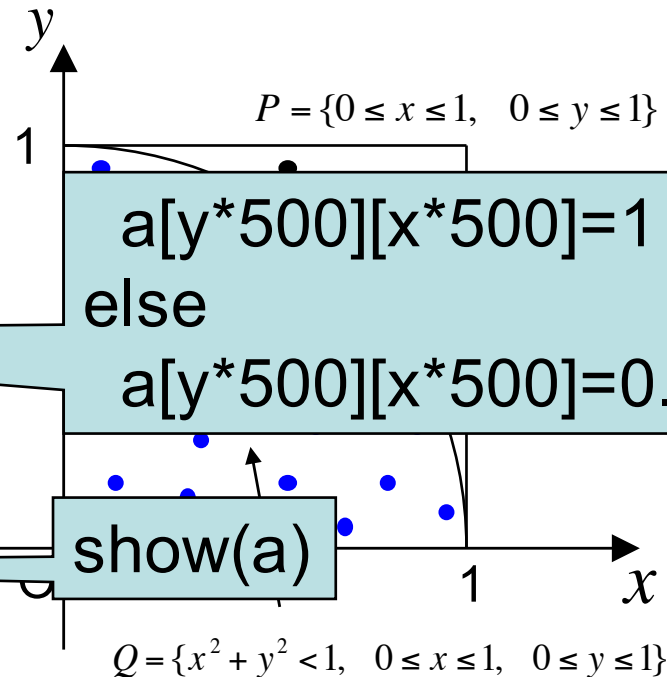
```
      m = m + 1
```

```
    end
```

```
  end
```

```
  m*1.0/n
```

```
end
```



montecarlo.rb

乱数とは？

- 乱数とは、ある一つの数が現れたとき、それに続く数が前の数と全く関係なく現れるような列(乱数列)の要素

もう少し丁寧に定義すると...

- 乱数列とは、ある分布に従う(互いに独立な事象を表す)確率変数の実現値の列をいう
- 乱数とは、乱数列の各要素である

ある確率変数 x の実現値の列である乱数列

$\{x_1, x_2, \dots, x_n, x_{n+1}, \dots\}$ において
 x が x_{n+1} の値をとることは x_1, x_2, \dots, x_n からは予測できない

確率分布によって、**一様乱数**、**正規乱数**などがある。

疑似乱数

(pseudorandom numbers)

- 乱数列のように見えるが、計算機である計算を行うことによって求めることができる数列をいう
- 区間 $[0, 1)$ の一様乱数を近似する数列をさすことが多い

疑似乱数は、 n 個目までの m 個の既知要素 $x_n, x_{n-1}, \dots, x_{n-m+1}$ を用いて以下の漸化式により $n+1$ 番目の要素を計算する

$$x_{n+1} = f(x_n, x_{n-1}, \dots, x_{n-m+1})$$

疑似乱数列 $\{x_n\}$ は周期が大きいことが望ましい

(実際には、一様分布に関する適合度、統計的独立性などを様々な乱数検定法で検定し疑似乱数としての資格を与²⁵える)