連続系アルゴリズム レポート課題2

- 宿題(レポート)
  - 提出状況に応じて，期末試験の点に最大 50 点を加算します
    - まったく提出しなくても，期末試験の点で成績が出ます
    - ただし，試験を受けなければ単位は出ません
  - プログラムだけはメールの添付ファイルで
    reiji@is.s.u-tokyo.ac.jp
    に送付してください
    - 所属学科，学年，学籍番号，氏名をメール本文に明記すること
    - メールの題名(Subject)は「連続系レポート課題第2回提出者氏名」とすること
  - 手計算をするものや，計算結果と考察など，プログラム以外は A4 の紙 1 枚(裏もつかってよい)にまとめてください
    - 所属学科，学年，学籍番号，氏名をレポートの最初に明記すること
    - 次回の講義の前に集めます
    - あるいは PDF で A4 サイズ 2 ページでもよい(プログラムと一緒にメールで送付してください)
  - メールの提出はできるだけ次回の講義の前日までにお願いします

- 問題1
  - 1 以上 2 以下の実数区間を I とする
  - I 上で $1/x$ を(以下の意味で)最適に近似する 1 次式 $p(x) = ax + b$ を求めよ
    - 誤差関数を $E(x) = 1/x - p(x)$ とする
    - $E(x)$ は I 上の 3 点で極値をとる
    - その 3 点は、I の両端 ($x = 1$ と $x = 2$) 2 箇所と、I の内部 1 箇所 ($x = \xi$ とする) である
    - その 3 箇所の極値の絶対値は等しい
      $|E(1)| = |E(2)| = |E(\xi)|$
    - その 3 箇所の極値の符号は交互になる
      $E(1) = E(2) = -E(\xi)$

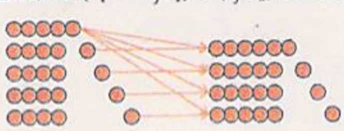
- 問題2
  - $x = 0.5$ から 1.0 まで 0.1 刻みで次の関数の値を計算し、配列に格納するプログラムを作れ
    - $1 + x + \cdots + x^n$ ($n = 2$ から 7 まで、Horner 法で)
    - 指数関数 $\exp(x)$、対数関数 $\log(x)$
  - 上記の配列の値を参照し、Neville または Aitken のアルゴリズムにより多項式補間を行い、$x = 0.75$ における値を求めよ
    - もともとの関数値と、補間値との差を求めよ

作り方ぬきtype

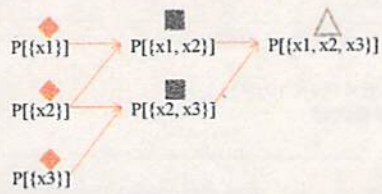
Neville のアルゴリズム, Aitken のアルゴリズム

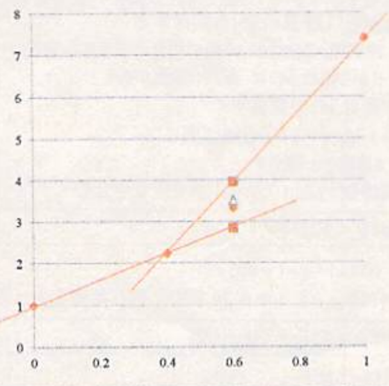
- 初期値: $P[\{x_k\}] = f(x_k)$ (x によらないので)
- Neville のアルゴリズム
  - for $j = 1$ to $n - 1$
    for $k = 1$ to $n - j$
    差分商で $\{x_{k+1} \cdots x_{k+j-1}\} \cup \{x_k, x_{k+j}\}$ の補間

- Aitken のアルゴリズム
  - for $j = 1$ to $n - 1$
    for $k = j + 1$ to n
    差分商で $\{x_1 \cdots x_{j-1}\} \cup \{x_j, x_k\}$ の補間



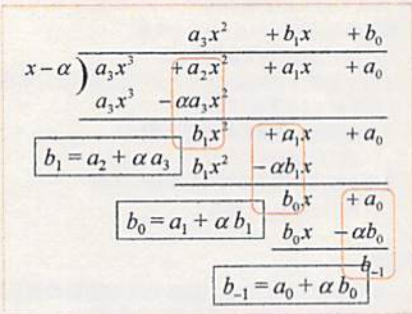
- Neville のアルゴリズムの様子





組み立て除法＋おまけ

- $(a_3 x^3 + a_2 x^2 + a_1 x + a_0) \div (x - \alpha)$ の筆算
  $= a_3 x^2 + b_1 x + b_0$ あまり b_{-1}



$b_{-1} = p(\alpha)$

a_3	a_2	a_1	a_0
	αa_3	αb_1	αb_0
a_3	b_1	b_0	b_{-1}

- $x = a$ でテーラー展開すると

$$p(x) = p(a) + (x-a)p'(a) + (x-a)^2 \frac{p''(a)}{2} + \cdots (x-a)^n \frac{p^{(n)}(a)}{n!}$$

- $p(x)$ を $x - a$ で割ると，余りが $p(a)$
  - 実は、筆算式の割り算と Horner 法は同じ！
  - 商を $q(x)$ とする
- $q(x)$ を $x - a$ で割ると，余りが $p'(a)$
  - 続ければ $x = a$ でのすべての微係数が計算できる
  - 微係数の計算と，原点移動は同じ！

- おまけ
  - n 次多項式の和・差は自明に $O(n)$ でできる
  - 積は，来週やる FFT で $O(n \log n)$ でできる
    - Karatsuba 法という $O(n^{3/2})$ のアルゴリズムもある
    - FFT より Karatsuba 法のほうがメモリが少なくて済む
  - 商(ここに書いてあるの)も，実は FFT を使って $O(n \log n)$ で実現できる
    - 一旦「逆数」$\lfloor x^{2n} / p(x) \rfloor$ を作る
    - Aho, Hopcroft, Ullman “the design and analysis of computer algorithms” Addison Wesley 1974, § 8.3
    - Karatsuba 法でも割り算ができる

本日の C 言語

- 繰り返し

```
int i;  // 整数 i を宣言

for (i = 0; i < 10; i++) {
  計算内容
}
```

  - 変数 i を 0 から 10 の手前(つまり 9)まで変化させながら，「計算内容」を実行する
    - 繰り返しの回数が 10 回の場合の標準的な書き方
    - 0 から始めるのは，配列と関係がある
  - 変数は int で宣言する
  - i ++ は，i = i + 1 とほぼ同じ
    - i の値を 1 増やす
  - 別の例：i を 10 から 1 まで変化させる

```
for (i = 10; i >= 1; i--)
```

    - i -- は i = i - 1 と同じ
  - 別の例：i を 0 から 20 まで 1 飛びに変化させる

```
for (i = 1; i <= 20; i+=2)
```

    - i += 2 は i = i + 2 と同じ

- 配列
  - 同じ型の変数が一列に並んだようなもの
  - 宣言

```
double a[10];
```

    - double 型の要素 10 個からなる配列
  - 参照

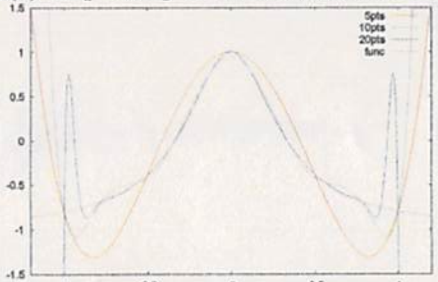
```
a[0] = 1.0;
a[1] = 2.0 * a[0];
```

    - 10 個ある要素のうち，どれを指定するかを，[] の中に書く
    - [ ] の中を添字(そえじ)という
    - 要素数が 10 の場合，添字は 0 から 9 まで
      - a[10] というのは存在しないことに注意
      - これを使ってしまうと，動作がおかしくなる
        - 普通，関係ないところで関係ないエラーが発生する
      - しかし，誰もが一度はやる失敗
        - 「添字を間違ったよ」とは教えてくれない
- 多次元配列

```
double a[10][10];
a[0][0] = 1.0;
```

Runge の現象

- $f(x)$ を $[-1, 1]$ で等間隔標本点で多項式補間

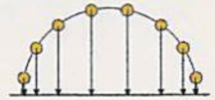


$$f(x) = \frac{2}{1+9x^2} - 1$$

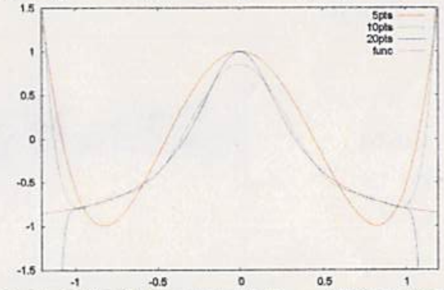
  - 区間の端の近くでは，次数を上げるほど精度が悪化!

5

Chebyshev 補間



- 標本点を $x_k = \cos((2k-1)\pi / (2n))$ とする



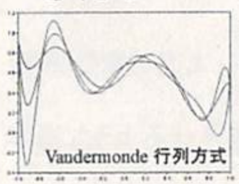
  - 区間の外の値は役に立たない・・・補外なんて論外！

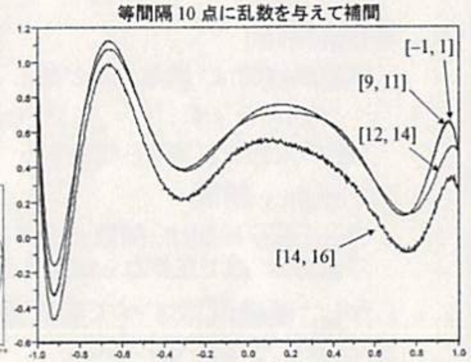
と言いながら，区間の端点は標本点ではなく，補外になっている
しかし，端点のわずかでも外では，精度が保証されない

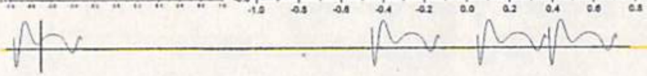
6

べき表現の危険性

- $[-1, 1]$ の範囲外になると丸め誤差が大きくなる







7

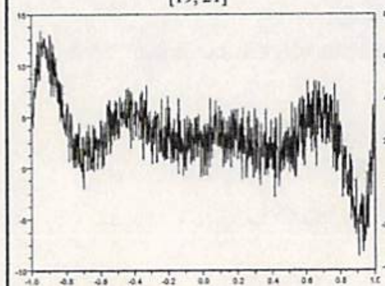
平行移動すると，誤差も大きくなる
0に近いものにうまくいく方が

べき表現の危険性

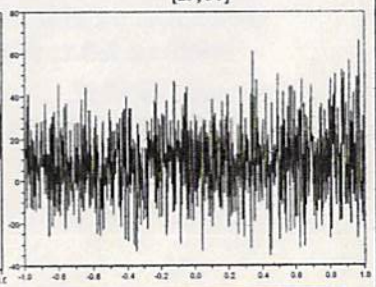
- もっとずらすと

・べき表現は [-1, 1] のみで(に移動して)使え
・もっと精度のよい表現方法もある
・(半)無限区間なら $1/x$ の多項式にする

[19, 21]



[29, 31]



8

べき表現の値の計算（1）

わりでかけるね

$\frac{1}{1-x}$

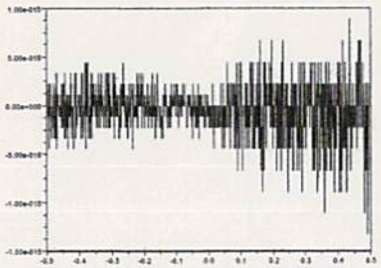
- $1+x+x^2+\cdots+x^{64} \approx (1-x)^{-1}$ を (–0.5, 0.5) で計算
- 誤差をプロット

$x^0 = 1;\ v = 1$

for $i = 1$ to 64

  $x^i = x^{i-1} * x$

  $v = v + x^i$



1

べき表現の値の計算（2）

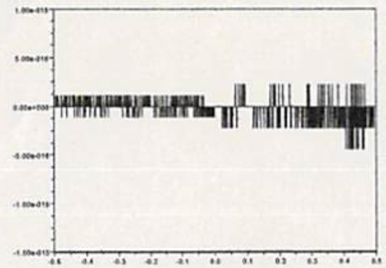
- 同じ計算を Horner 法で行い，誤差をプロット
- 誤差が小さい

$v = 1$

for $i = 63$ to 0

  $v = v * x + 1$

Horner 法の方が
誤差が少ない。



2

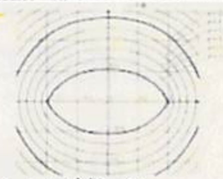
Weierstrass の近似定理

- Weierstrass の定理
  - 実軸上の閉区間で連続な関数は，この区間で一様に多項式によって近似できる
    - 連続だけの条件で，多項式でいくらでもよい近似ができる
- Bernstein 多項式
  - Weierstrass の定理の有名な証明だが・・・
  - 収束が遅く（$1/n$ 程度），実用にはならない
- 直交多項式（次週）を用いるべし

参考書：「数値計算の常識」伊理正夫＆藤野和建，共立

3

補間の近似精度



- 等間隔補間
  - 等間隔補間は，関数 $f(x)$ がある $\rho > 1$ に対して
    $$A(\rho) = \{z \in \mathbf{C} : |(z+1)^{z+1} / (z-1)^{z-1}| = 4\rho^2\}$$
    とその内部で正則なら収束する
- Chebyshev 補間
  - Chebyshev 補間は，関数 $f(x)$ が [– 1, 1] を含む（複素平面の）領域で正則なら収束する
- しかし，連続関数すべてを収束させるような標本点の取り方は存在しない

参考書：杉原・室田「数値計算法の数理」9 章

4

べき級数の危険 p7.

$$p(x) = a_0 + a_1 \cdots + \underbrace{a_{n-1}x^{n-1}}_{\text{すごい大きい}} + \underbrace{a_n x^n}_{\text{すごい大きい}}$$

大 − 大 + 大 … なので、けた落ちがおこる.

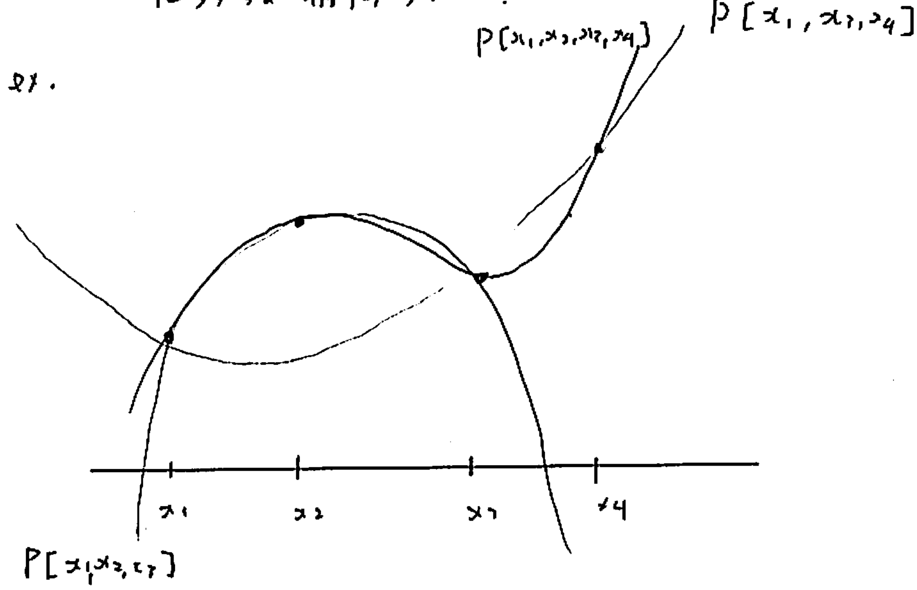
差分商

$P[X]$ = 点列の集合が

$X = \{x_1, x_3, x_4\}$

に対する補間多項式.

例.



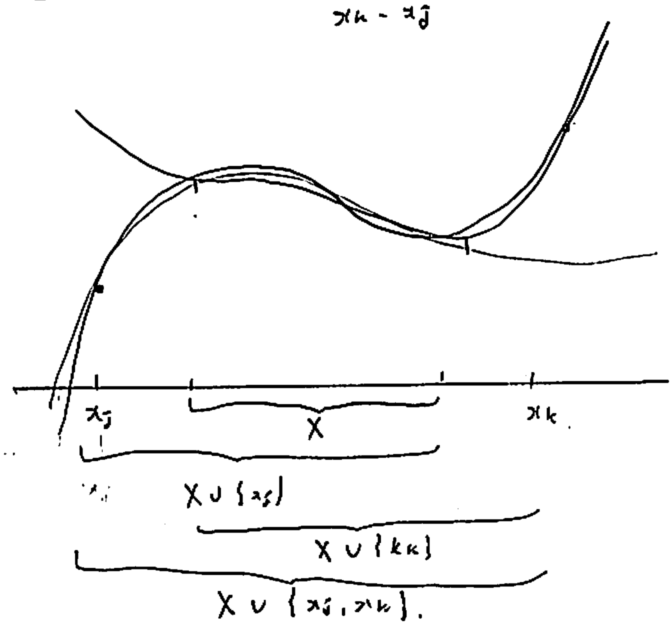
$X \cap \{x_j, x_k\} = \phi$ のとき.

$P[X \cup \{x_j, x_k\}]$ がほしい.

左辺は右辺よりも1次高い

$P[X \cup \{x_j, x_k\}]$

$$= \frac{(x-x_j)\,P[X \cup \{x_k\}] - (x-x_k)\,P[X \cup \{x_j\}]}{x_k - x_j}$$

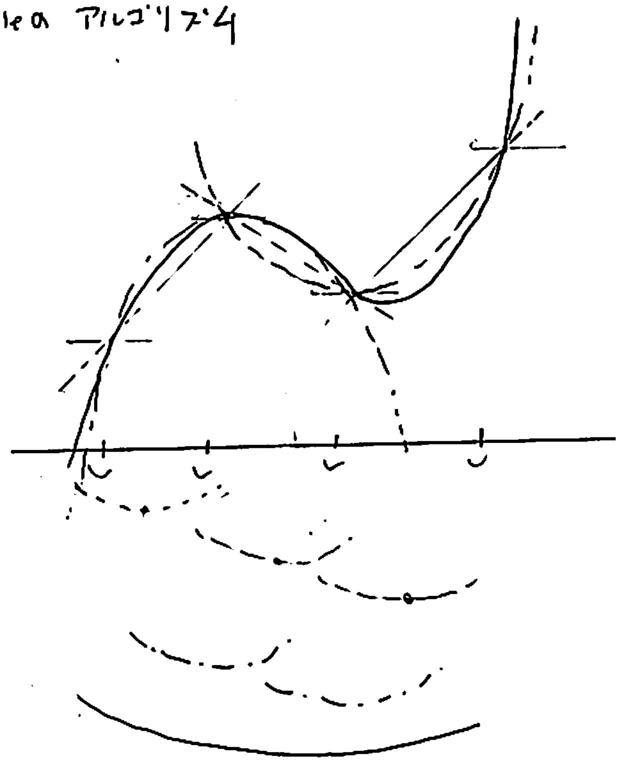


$x = x_j$ を代入すると v_j
$x = x_k$ 〃 v_k
$x = x_i \in X$ 〃 v_i

これより $P[X \cup \{x_j, x_k\}]$ は補間多項式であることがわかる

多項式を作ることができるし
値を求めることもできる

Nevilleのアルゴリズム



Aitken

やり方ちがう.

連続系アルゴリズム 10/17

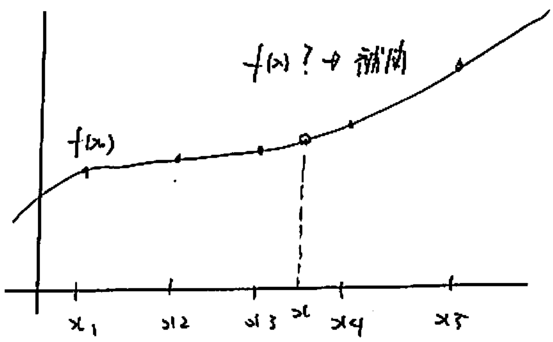
関数の近似

$f: x \longmapsto f(x)$

① 基底関数の線形結合

→ 多項式 $1, x, x^2, x^3, \dots$
　 三角関数
　 直交多項式

② 座標と値 の組
(入力)　(出力)



$\{ (x_1, f(x_1)), (x_2, f(x_2)), \dots \}$

多項式

$p(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_n x^n$

「べき表現」 $n, a_0, a_1, \dots, a_n$

問題

xの値を与えた，$p(x)$の値を求める．

cf. $a * x * x * x + b * x * x + c$.

$a_4 * x * x * x * x$.
$+ \quad \vdots$
$+$
$+ a_1 * x$
$+ a_0$

12ムダ．

$x_2 = x * x$
$x_3 = x * x_2$
$x_4 = x * x_3$

→ $a_4 x_4 + a_3 x_3 + a_2 x_2 + a_1 x + a_0$

◦ $(((a_4 * x + a_3) * x + a_2) * x + a_1) * x + a_0$.

Honer 法．

```
v = a_n                  乗算 n回
for i = n-1, ..., 0      加算 n回
    v = v * x + a_i
```

補間.

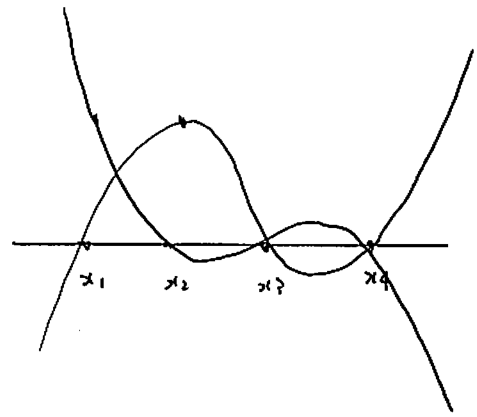
値を求めるもの

多項式を求めるもの

問題　n点の座標 $\{x_1, \dots, x_n\}$

次の条件をみたす多項式 $\pi_i(x)$ を求めよ

$$\pi_i(x_j) = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases} \quad (j = 1 \dots n)$$



補間多項式.

$\pi_i(x)$ は $n-1$ 次以上.

$$w_i(x) = \prod_{j \neq i} (x - x_j).$$

cf. $w_1(x) = (x - x_2)(x - x_3)(x - x_4)$

$w_2(x) = (x - x_1)(x - x_3)(x - x_4).$

$\vdots$

w を用いて,

$$\pi_i(x) = \frac{w_i(x)}{w_i(x_i)} \quad \begin{matrix} \leftarrow \text{多項式,} \\ \leftarrow \text{定数.} \end{matrix}$$

が解

Lagrange 補間

問題.　n個の座標と値のペア

$\{(x_1, v_1), (x_2, v_2), \dots, (x_n, v_n)\}$

が与えられていて, $p(x_i) = v_i$

となる多項式 $p(x)$ を求めよ.

解.　$p(x) = \sum_{i=1}^{n} v_i \pi_i(x)$.　$n-1$ 次多項式

$p(x_1) = v_1$

$p(x_2) = v_2$

$\vdots$

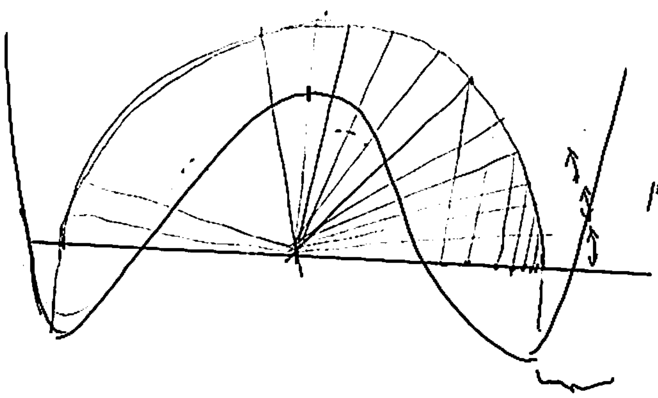
プリント p5.

10点 中はあっているが、
外がはずれる

20点 外がより悪化.

→ はじめより次数を上げるほど悪化する
(Rungeの現象)

Chebyshev 補間.



補間ができる条件

プリントp4

○ 等間隔補間

$$A(\rho) = \{ z \in \mathbb{C} \mid |(z+1)^{z+1}/(z-1)^{z-1}| = 4\rho^2 \}$$

とするの内部で正則なら収束.

○ Chebyshev 〃

$[-1, 1]$ で f が正則なら収束.

それではムリ