数理解析特論 (1)

MATLAB 入門, 連立1次方程式に対する直接法

桂田 祐史

2005 年 4 月 26 日

1 連絡事項

数理解析特論のページは

http://www.math.meiji.ac.jp/~mk/lecture/suurikaisekitokuron/

2 6701 号室の使い方

- http://www.math.meiji.ac.jp/howto/ に 6701 号室利用のルールがある。
- パスワードは今日中に変えてください (ee18044, suuri1, suuri2 というアカウントを準備)。

3 Octave, Scilab 入門および本日の課題

数学科では MATLAB のネットワーク・ライセンスを 10 個所有している。もちろん 6701 号室の Windows パソコンから利用できる。

ここでは、実例で MATLAB の使い方を紹介する。

以下は、以前に MATLAB 互換ソフトである Octave を利用していたときの資料である。

Octave マニュアル: http://www.math.meiji.ac.jp/~mk/labo/howto/octave.pdf

また個人的なメモであるが、

http://www.math.meiji.ac.jp/~mk/lecture/suurikaisekitokuron/
private-matlab-notebook/ 『MATLAB 手習い』

を公開する。

3.1 三重対角行列の逆行列と LU 分解

(以下、Octave を用いた説明である (申し訳ない)。基本的に同じコマンドが使えるはずである。)

行列 A が疎行列であっても、その逆行列 A^{-1} は疎行列とは限らない。しかし $PA=LU$ と LU 分解したときの[1]、L, U は疎行列である。

このことを 3 重対角行列

$$A = \begin{pmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{pmatrix}$$

について調べてみよう。

```
mathpc% octave
GNU Octave, version 2.0.16 (i386-unknown-freebsd3.4).
Copyright (C) 1996, 1997, 1998, 1999, 2000 John W. Eaton.
This is free software with ABSOLUTELY NO WARRANTY.
For details, type ‘warranty’.

octave:1> n=5
n = 5
octave:2> ones(n-1,1)
ans =

  1
  1
  1
  1

octave:3> J=diag(ones(n-1,1),1)
J =

  0  1  0  0  0
  0  0  1  0  0
  0  0  0  1  0
  0  0  0  0  1
  0  0  0  0  0

octave:4> J+J’
ans =

  0  1  0  0  0
  1  0  1  0  0
  0  1  0  1  0
  0  0  1  0  1
  0  0  0  1  0

octave:5> a=2*eye(n,n)-J-J’
a =
```

[1] P は行の交換を表わす置換行列。$P^2=I$ (単位行列) という性質がある。

```
   2  -1   0   0   0
  -1   2  -1   0   0
   0  -1   2  -1   0
   0   0  -1   2  -1
   0   0   0  -1   2

octave:6> inv(a)
ans =

  0.83333  0.66667  0.50000  0.33333  0.16667
  0.66667  1.33333  1.00000  0.66667  0.33333
  0.50000  1.00000  1.50000  1.00000  0.50000
  0.33333  0.66667  1.00000  1.33333  0.66667
  0.16667  0.33333  0.50000  0.66667  0.83333

octave:7> [L U P]=lu(a)
L =

   1.00000   0.00000   0.00000   0.00000   0.00000
  -0.50000   1.00000   0.00000   0.00000   0.00000
   0.00000  -0.66667   1.00000   0.00000   0.00000
   0.00000   0.00000  -0.75000   1.00000   0.00000
   0.00000   0.00000   0.00000  -0.80000   1.00000

U =

   2.00000  -1.00000   0.00000   0.00000   0.00000
   0.00000   1.50000  -1.00000   0.00000   0.00000
   0.00000   0.00000   1.33333  -1.00000   0.00000
   0.00000   0.00000   0.00000   1.25000  -1.00000
   0.00000   0.00000   0.00000   0.00000   1.20000

P =

  1  0  0  0  0
  0  1  0  0  0
  0  0  1  0  0
  0  0  0  1  0
  0  0  0  0  1

octave:8> U\(L\P)        あるいは  inv(U)*inv(L)*P
ans =

  0.83333  0.66667  0.50000  0.33333  0.16667
  0.66667  1.33333  1.00000  0.66667  0.33333
  0.50000  1.00000  1.50000  1.00000  0.50000
  0.33333  0.66667  1.00000  1.33333  0.66667
  0.16667  0.33333  0.50000  0.66667  0.83333

octave:9> n=8;J=diag(ones(n-1,1),1);a=2*eye(n,n)-J-J';[L U P]=lu(a);inv(a)
ans =

  0.88889  0.77778  0.66667  0.55556  0.44444  0.33333  0.22222  0.11111
  0.77778  1.55556  1.33333  1.11111  0.88889  0.66667  0.44444  0.22222
  0.66667  1.33333  2.00000  1.66667  1.33333  1.00000  0.66667  0.33333
  0.55556  1.11111  1.66667  2.22222  1.77778  1.33333  0.88889  0.44444
  0.44444  0.88889  1.33333  1.77778  2.22222  1.66667  1.11111  0.55556
```

```
  0.33333  0.66667  1.00000  1.33333  1.66667  2.00000  1.33333  0.66667
  0.22222  0.44444  0.66667  0.88889  1.11111  1.33333  1.55556  0.77778
  0.11111  0.22222  0.33333  0.44444  0.55556  0.66667  0.77778  0.88889

octave:10> U\(L\P)     あるいは   inv(U)*inv(L)*P
ans =

  0.88889  0.77778  0.66667  0.55556  0.44444  0.33333  0.22222  0.11111
  0.77778  1.55556  1.33333  1.11111  0.88889  0.66667  0.44444  0.22222
  0.66667  1.33333  2.00000  1.66667  1.33333  1.00000  0.66667  0.33333
  0.55556  1.11111  1.66667  2.22222  1.77778  1.33333  0.88889  0.44444
  0.44444  0.88889  1.33333  1.77778  2.22222  1.66667  1.11111  0.55556
  0.33333  0.66667  1.00000  1.33333  1.66667  2.00000  1.33333  0.66667
  0.22222  0.44444  0.66667  0.88889  1.11111  1.33333  1.55556  0.77778
  0.11111  0.22222  0.33333  0.44444  0.55556  0.66667  0.77778  0.88889

octave:11> quit
mathpc%
```

3.2 課題 1

(1) 行列の積の計算や連立 1 次方程式を解く時間を測って、それが行列の寸法 n にどのように依存しているかを調べよ。なお、コマンドの実行時間は `tic; コマンド;toc` で計測できる。例えば

```
octave:1> n=100;a=rand(n,n);b=rand(n,1);tic;x=a\b;toc
```

とすると、連立 1 次方程式の解の計算 `a\b` の実行時間が表示される。n を色々変えて計測し、横軸 n, 縦軸が計算時間であるグラフを描いて[2]分析せよ (対数グラフが適当かも知れない)。

(2) 三角行列の逆行列が三角行列であることを実験で確かめよ[3]。また、疎行列であるという性質は逆行列には遺伝しないことを上の実験で確かめたが、LU 分解したときの L, U に遺伝するかどうか (上の例とは異なる場合で) 確かめよ。

4 参考書案内

MATLAB, Octave, Scilab については、WWW 上に解説文書があふれているが、日本語で読める書籍としては 大石 [2] がある。

参考文献

[1] 有木進, 工学のための線形代数, 日本評論社 (2000).

[2]Octave 自身を使ってもよいし、**gnuplot** を使っても良い。

[3]与えられた行列の下三角部分を求める `tril()`, 上三角部分を求める `triu()` という関数が利用できるかもしれない。

[2] 大石進一, Linux 数値計算ツール, コロナ社 (2000).

[3] 大石進一, MATLAB による数値計算, 培風館 (2001).

[4] 小国力/Dongarra, Jack J., MATLAB による線形計算ソフトウェア, 丸善 (1998).

[5] 小国力, MATLAB と利用の実際 — 現代の応用数学と CG —, サイエンス社 (1995).

A 数値計算ソフトウェアの発展 (駆け足説明)

主に行列計算関係に焦点を当てて説明する (先週の講義の補足)。

A.1 数値計算ライブラリィ

数値計算ライブラリィの歴史 (概略)

1. **サブルーチン**[a](subroutine) の誕生、**サブルーチン・ライブラリィ**の誕生
2. (汎用) プログラミング言語[b]の誕生 (**FORTRAN**[c], **LISP** などが最初の例)
3. 固有値計算ライブラリィ **EISPACK**
   (論文誌 "Numerische Mathematic" で発表されたアルゴリズムを元に最初は **ALGOL** で書かれ、後に FORTAN に移植される。主宰者は有名な数値解析学者である Wilkinson である。)
4. 連立 1 次方程式の解法ライブラリィ **LINPACK**
   途中から **BLAS** が生まれ、LINPACK は BLAS の上に構築される。
5. 線形計算ライブラリィ **LAPACK**
   (**メモリー階層**を考慮した BLAS を全面的に採用、EISPACK & LINPACK の現代化)
6. 他のプログラミング言語への移植 — **TNT**[d] (C++) など。

[a]機械語 (machine language) や、Fortran 言語における、あるまとまった処理をするプログラムの単位を呼ぶ言葉。C 言語における「関数」に相当すると考えて構わない。

[b]当時は「自動プログラミング言語」と呼ばれたそうである。

[c]FORTRAN は、IBM が線形計画法のプログラムを効率的に作成するために開発した言語であると言われている。

[d]C++ 向けには **LAPACK++** があったが、C++ 言語の ANSI 規格の進展に伴い、新しく設計し直されたのが TNT である。まだ発展途上で、LAPACK の機能のすべては実装されていない。

ここで名を紹介した EISPACK, LINPACK, LAPACK, TNT はいずれもソースが公開されているフリーソフトである[4]。

[4] このあたりに欧米文化の強さが感じられる。ここで紹介したソフトの中には、博士号クラスの研究者が数十人、何年も作業して始めて開発できたものもある。日本でも大学を中心に様々なライブラリィの開発がされたが、全面公開までこぎつけたものは少なく (途中で企業に売ってしまったものもある)、大変もったいない事態になっていると筆者は感じている。こうなってしまった背景には、ソフトウェアの開発を研究業績とは認めない風潮など、二三の理由が考えられる…(脱線)

数値計算ライブラリィを採用で実現できること

(1) 高い生産性

(2) 高い信頼性 (バグが少ない、高精度、条件が悪い問題でも崩れないタフさ)

(3) 高い効率性 (速度、メモリー利用効率)

なぜ高い実行効率が得られるか

速度をあげるために考えねばならないこととして、講義では**計算量**を説明した。これらのソフトウェアでは計算量の観点から無駄がないことはもちろんであるが、それ以外にも重要な要素がある。

1. ループのアンロールなどのテクニック
2. メモリー階層を考慮したプログラミング

これらを追求すると、利用するコンピューター・システムに合わせたプログラムのチューニングが必要になり、プログラムの汎用性が低くなる恐れがある。しかし、システムごとにチューニングが必要な部分を小さな部分に凝縮し、他と分離することにより、この問題をある程度解決できる。LAPACK については、BLAS がこのチューニングが必要な部分を担当している。LAPACK に付属する BLAS は “reference (参考) BLAS” と呼ばれ、FORTRAN で書かれているので移植性があるが、高速な計算を望む場合は最適化された BLAS に差し換えることになる。例えば Sun Workstation の場合、Sun Microsystem 製ソフトウェア開発環境 Sun Workshop では、コンパイラーと一緒に BLAS が (と実は LAPACK も) 提供されている)。Windows や Intel CPU 向けの Linux の場合は Intel から BLAS が提供されている (無償)。これらは人手で最適化されたものである (と思われる) が、色々なパラメーターを変化させながら性能を測定することで、最適なパラメーターを実験的に「算出」し、その結果を元に BLAS プログラムを自動生成するソフトウェアもある (例えばフリーソフトの **ATLAS**)。

A.2 MATLAB とその互換システムの登場

MATLAB は EISPACK の開発にも関わった C.Moler が作成したシステムで、彼が創立した MathWorks 社から販売されている。

MATLAB の特徴

- インタープリター型言語である。そのため[a]、
  - 対話的で使いやすいシステムになっている。
  - 注意深く利用しないと実行効率が低くなる[b](個々の命令の実行時に命令解釈のコストが必要なため、繰り返し処理を多用すると計算時間が長くなりがちである)。
- LAPACK などの各種数値計算ライブラリィを内蔵している (これらのライブラリィ群へのインターフェイスであると理解すべきかもしれない)。
- ベクトル、行列などのデータの型が始めから定義されているので、命令が簡潔になっていて、プログラミングも楽になった[c]。

[a]ここで指摘することは、例えば **Mathematica**, **Maple** のような**数式処理系**にも当てはまる。
[b]ここで述べたような注意は、かつてはパソコン上で BASIC 言語を使ってプログラムを開発する際の常識であったのだが、今ではあまり知られていないことなのだろう。
[c]オブジェクト指向であり、データ構造が隠蔽されていると言って良いかもしれない。LAPACK などの利用で面倒な点の一つに、プログラマーにライブラリィ中で定義されたデータ構造を正しくなぞったプログラムを書く努力が要求されるというものがあるが、MATLAB ではこれがなくなっている。この点は C++ で書かれたライブラリィでも期待できることであるが。

MATLAB をいかに評価するかであるが、筆者は最近

大したものではないか、ひょっとするとコロンブスの卵で大発明？

と考えるようになった。このような**システムを作るのは実は簡単**で (実際、以下紹介するように「真似」がたくさん出て来た)、しかし使ってみると分るが、**非常に便利**である。日本の数学界ではあまり人気がない (というか知られていない) ようであるが、欧米や日本でも工学の世界では浸透している。

MATLAB は現在も改良が続けられていて、行列計算関係では、**疎行列向きの処理法**や**反復法**なども採り入れられている。簡単な偏微分方程式のシミュレーションへの応用も十分可能なレベルに成長した。

MATLAB を後を追ったシステムがたくさん開発されたが、MATLAB の言語仕様は「準標準」となっている。以下 MATLAB と似たシステムをいくつか紹介しよう[5]。いずれもソース公開のフリーソフトウェアである。

Octave MATLAB との互換性が高い。残念ながら疎行列専用の処理が用意されていないが、入門には十分であるし、用途を選べば実用性も高い。

Scilab MATLAB との互換性の程度は Octave よりも低いが、ソフトウェアとしての完成度はやや高い (と思われる)。疎行列処理も完全ではないが、LU 分解とそれに基づく連立 1 次方程式の解法程度はサポートしている。開発元から Windows 向けのバイナリーが配布されている。

[5]`http://www.dspguru.com/sw/opendsp/mathclo2.htm` などが参考になる。