Зачетные задания по курсу «MATLAB в научных исследованиях»

Н. Ю. Золотых

Выполнение зачетного задания включает:

1. Изучение темы
2. Реализацию программы
3. Подготовку письменного отчета

Оформление отчета

Отчет должен быть подготовлен в электронном виде и должен содержать:

1. ФИО студента и номер группы;
2. постановку задачи;
3. краткое описание алгоритма (алгоритмов) ее решения;
4. аргументированные ответы на вопросы, приведенные в задании;
5. текст программы;
6. распечатку выдаваемых программой результатов (таблицы, графики и т. п. — в зависимости от задания).

Порядок сдачи

Отчет (в любом из форматов: TEX, LaTeX, Word, pdf, ps) и m-файлы нужно направить по e-mail на адрес Nikolai.Zolotykh@gmail.com. Тема письма должна быть следующая: MATLABcourse2011 (в противном случае я не гарантирую, что письмо не попадет в спам или потеряется еще где-нибудь). Крайний срок сдачи: *15 декабря 2011*. После этой даты отчеты не принимаются.

Номер своего варианта можно посмотреть здесь: https://docs.google.com/spreadsheet/ccc?key=0AkB-xd6y1Lm3dEozLUhpUDdWWThRSkhJOW50V3FxcGc&hl=en_US Там же можно отслеживать ход проверки вашей работы.

Список зачетных заданий

1. Напишите функцию `mysin(x, n)`, вычисляющую сумму первых n членов ряда Тейлора для функции $\sin x$:

   $$\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \ldots$$

   Функция должна выполнять $O(n)$ операций. Вариант этой функции `mysin(x)` должен вычислять столько слагаемых, пока результат перестанет изменяться. Протестируйте функцию на значениях $x = \frac{\pi}{2}, \frac{11\pi}{2}, \frac{21\pi}{2}, \frac{31\pi}{2}$. Какова точность результатов, полученных для таких x? Сколько членов потребовалось? Каков максимальный по абсолютному значению член? Попробуйте объяснить, почему результаты неудовлетворительные.

2. Используя интеграл

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

   можно найти приближение к π.

   Используйте правило прямоугольников, трапеций и Симпсона с 2, 4, 8, 16, 32, 64 и 128 узлами. Затабулируйте погрешность. Как уменьшается погрешность при удвоении числа элементарных отрезков? Почему с некоторого момента погрешность не уменьшается? Составленная программа должна выводить графики подинтегральных функций, табулированные погрешности и значения интегральных сумм с графиками.

3. Используя интеграл

$$-\frac{4}{9} = \int_0^1 \sqrt{x} \ln x dx$$

можно найти приближение к $-4/9$. Задание такое же, как в № 2.

4. Постройте 11 точек, взяв

$$x_i = \frac{i-1}{10}, \quad y_i = \mathrm{erf} x_i \quad (i = 1, \ldots, 11),$$

где

$$\mathrm{erf} x = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

— *функция ошибок*. Для вычисления $\mathrm{erf} x$ воспользуйтесь функцией `erf`.

   (a) Составьте систему линейных уравнений для нахождения коэффициентов интерполирующего многочлена 10-ой степени. С помощью функций `cond` и `condest` найдите/оцените число обусловленности этой системы. Решите составленную систему (команда \). Насколько достоверно полученное решение? С помощью построенного интерполянта протабулируйте значение функции в промежуточных точках. Чему равны абсолютная и относительная ошибки?

   (b) Сравните коэффициенты построенного интерполирующего многочлена с коэффициентами многочлена, который можно получить с помощью функции `polyfit`. В чем разница?

   (c) Постройте кубический эрмитов интерполянт (`pchip`) и кубический сплайн (`spline`). Найдите значения интерполянтов в тех же точках, что и в предыдущих пунктах и сравните результаты.

Составленная программа должна выводить исходные точки, графики интерполируемой функции и интерполянтов и графики зависимости чисал обусловленности от степени интерполяционного многочлена

5. Рассмотрим задачу поиска интерполирующей кривой по точкам $p_1, p_2, \dots, p_n$, $p_i = (x_i, y_i)$ $(i = 1, \dots, n)$. Предположим, что форму этой кривой нельзя описать функцией вида $y = f(x)$. Например, этого нельзя сделать для кривой в форме буквы S и др. (функция была бы неоднозначной).

   Рассмотрим два набора данных (i, x_i) и (i, y_i) $(i = 1, \dots, n)$. Их можно независимо интерполировать функциями $f(t)$, $g(t)$. Тогда кривая, заданная параметрически,

$$\begin{cases} x = f(t) \\ y = g(t) \end{cases}$$

   будет интерполировать наши данные.

   Другой способ — интерполировать данные $(\sum_{i=1}^{i-1} d_i, x_i)$ и $(\sum_{i=1}^{i-1} d_i, y_i)$ $(i = 1, \dots, n)$, где d_i расстояние между точками p_i и p_{i+1}.

   Нарисуйте букву S (или любую другую) и измерьте координаты 8 точек p_i, лежащих на ней. Интерполируйте эти данные обоими способами, описанными выше

   (a) кубическим интерполяетом;
   (b) кубическим сплайном.

   Изобразите полученные кривые и сравните результаты.

6. Провести серию экспериментов, тестирующую команды `\`, `cond`, `condest` на хорошо обусловленных матрицах. Число обусловленности на следующих классах матриц с ростом порядка матрицы n растет *умеренно*:

   (a) $A = LU$, где L, U — нижняя треугольная и верхняя треугольная матрицы соответственно с диагональным преобладанием.
   (b) $A = LL^T$, где L — нижняя треугольная матрица с диагональным преобладанием.

   Экспериментальным путем для каждого из перечисленных классов установить диапазон изменения порядка матрицы, при котором число обусловленности не превосходит ведичины порядка 10^16. Построить графики значений, выдаваемых функциями `cond` и `condest`.

Сравнить их. Построить графики используемого этими функциями времени. Для каждой матрицы задать случайно решение x. По матрице A и решению x сгенерировать правую часть системы линейных уравнений b. Решить систему $Ax = b$, используя команду \. Построить графики относительной ошибки решения и невязки. Объяснить результаты.

7. Провести серию экспериментов, тестирующую команды \, `cond`, `condest` на хорошо обусловленных матрицах вида $A = (a_{ij})$, где

$$a_{ij} = \begin{cases} 2i+1, & \text{если } i = j, \\ i, & \text{если } i = j+1, \\ j, & \text{если } i = j-1, \\ 0, & \text{если } i = j. \end{cases}$$

   Задание такое же, как в № 6.

8. Провести серию экспериментов, тестирующую команды \, `cond`, `condest` на хорошо обусловленных матрицах вида $A = (a_{ij})$, где

$$a_{ij} = \begin{cases} j+1, & \text{если } i \geq j, \\ i+1, & \text{если } i < j, \end{cases}$$

   Задание такое же, как в № 6.

9. Провести серию экспериментов, тестирующую команды \, `cond`, `condest` на матрицах вида $A = (a_{ij})$, где a_{ij} — псевдослучайные равномерно распределенные на отрезке $[-1, 1]$ числа. Задание такое же, как в № 6.

10. Провести серию экспериментов, тестирующую команды \, `cond`, `rcond`, `condest` на плохо обусловленных матрицах вида $A = LU$, где L, U — нижняя треугольная и верхняя треугольная матрицы соответственно с малыми диагональными элементами.

    Экспериментальным путем для каждого из перечисленных классов установить диапазон изменения порядка матрицы, при котором число обусловленности не превосходит ведичины порядка $10^1$6. Построить графики значений, выдаваемых функциями `cond`, `rcond`, `condest`. Сравнить их. Построить графики используемого этими функциями времени. Для каждой матрицы задать случайно решение x. По матрице A и решению x сгенерировать правую часть системы линейных уравнений b. Решить систему $Ax = b$, используя команду \. Построить графики относительной ошибки решения и невязки. Объяснить результаты.

11. Провести серию экспериментов, тестирующую команды \, `cond`, `rcond`, `condest` на плохо обусловленных матрицах вида $A = (a_{ij})$, где
$$a_{ij} = \sin\frac{(i+1)(j+1)}{2n} + 10^{-6} \times \mathrm{rand},$$
rand — псевдослучайное число с равномерным распределением на отрезке $[-1/2,\ 1/2]$. Задание такое же, как в № 10.

12. Провести серию экспериментов, тестирующую команды \, `cond`, `rcond`, `condest` на плохо обусловленных матрицах вида $A = (a_{ij})$, где
$$a_{ij} = \frac{1}{1 + 66 \times \left(\frac{(i+1)(j+1)}{n}\right)^4}$$
Задание такое же, как в № 10.

13. (M. T. Heath. ScientiÏc Computing: An Introductory Survey. McGraw-Hill, New York, 1997.) Вычисление К. Гауссом орбиты астероида Церера — по-видимому, первое использование метода наименьших квадратов. Данная задача составлена по мотивам этого исследования. Выражение $z = ax2 + bxy + cy2 + dx + ey + f$ называется квадратичной формой. Множество точек (x, y), для которых $z = 0$ называется кривой второго порядка или коническим сечением. К коническим сечениям относятся эллипсы, гиперболы и параболы. При некоторых значениях коэффициентов можно получить пустое множество, точку, прямую или пару прямых (параллельных или пересекающихся). Уравнение $z = 0$ можно нормализовать, разделив квадратичную форму на один из коэффициентов. Например, если $f \neq 0$, то мы можем разделить уравнение на f и получить тем самым квадратичную форму с единичным свободным коэффициентом. Для изображения конических сечений можно воспользоваться функцией `contour`.

Известно, что небесные тела двигаются по эллиптическим орбитам. Даны 10 наблюдений движения одной планеты:

```
x = [1.02 .95 .87 .77 .67 .56 .44 .30 .16 .01]';
y = [0.39 .32 .27 .22 .18 .15 .13 .12 .13 .15]';
```

(a) Методом наименьших квадратов определите коэффициенты квадратичной формы, положив $= 1$ и решив перепределенную 10×5 систему линейных уравнений относительно остальных коэффициентов. Изобразите орбиту планеты. На том же графике постройте заданные точки.

(b) Рассматриваемая задача близка к задаче с неполным рангом (т.е. задаче с рангом меньшим 5). Чтобы увидеть, к чему это приводит, внесите в векторы `x`, `y` небольшое возмущение, добавляя к каждой компоненте случайную величину, равномерно распределенную на отрезке $[-.005, .005]$. Вычислите новые коэффициенты и изобразите новую орбиту на прежнем графике. Объясните различие коэффициентов и орбит от прежних.

14. Робин Гуд бросает камень с привязанным к нему письмом через бойницу крепостной стены в открытое окно, стоящего за ней замка (см. рис. 1). Смоделируйте полет камня с помощью функции ode45. Рассмотрите следующие события:

    (a) камень упал на землю, не долетев до окна (либо до, либо после стены);

    (b) камень не попал в бойницу (ударился о стену);

    (c) камень застрял в бойнице (стукнулся о «пол» или «потолок» бойницы и упал внутри нее);

    (d) камень стукнулся о стену замка (не попав в окно);

    (e) камень попал в окно (для простоты считаем, что и крепостная стена, и замок очень высокие и сверху через них камень перебросить нельзя).

    В каждом из этих случаев вычисления прекращаются.

    Написать функцию, на вход которой подаются:

```
                    стена                     замок
                    _____
                   [     ]                   ______
                   [     ]                  |      |
                   [_____]                  |      |
                   бойница                  |      |
                    _____                   |      |
                   [     ]                  |      |
                   [     ]                    окно |
                   [     ]                  |      |
                   [     ]                  |      |
.                  [     ]                  |      |
 \0 _              [     ]                  |      |
  I                [     ]                  |      |
 / \               [     ]                  |      |
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

Рис. 1: *Робин Гуд бросает камень с письмом в окно замка.*

(a) параметры: масса камня, расстояние от Робин Гуда до стены, толщину стены (равную длине бойницы), расстояние от земли до бойницы и высоту бойницы, расстояние от стены до замка, расстояние от земли до окна и высоту окна —

(b) начальные условия: угол броска и начальную скорость камня.

Программа должна отображать «вид сбоку» (многоугольники рисуются с помощью функций `fill` или `patch`) и траекторию полета.

15. Метод Ньютона нахождения корня уравнения $f(x) = 0$ заключается в итерациях вида

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}.$$

Написать функцию `newton(f, df, [x0, x1]`, реализующую метод Ньютона, где `f` — строка, задающая правую часть $f(x)$ уравнения, `df` — строка, задающая $f'(x)$, `[x0, x1]` — отрезок локализации. Функция должна возвращать найденный корень с макимально возможной точностью.

Написать программу (скрипт), тестирующую эту функцию и сравнивающую ее со стандартной функцией `fzero` на уравнениях:

$$x^3 - 2x - 5 = 0, \qquad 0 \le x \le 3$$

(исторический пример Валлиса),

$$\sin x = 0, \qquad 1 \le x \le 4,$$

$$x^3 = 0.001, \qquad -1 \le x \le 1,$$

$$\ln x + \frac{2}{3} = 0, \qquad 0 \le x \le 1,$$

$$\operatorname{sgn}(x-2)\sqrt{|x-2|} = 0, \qquad 1 \le x \le 4,$$

$$\arctan x = \frac{\pi}{3}, \qquad 0 \le x \le 5,$$

$$\frac{1}{x-\pi} = 0, \qquad 0 \le x \le 5.$$

Программа должна печатать таблицу, в которой указываются найденные функциями `newton`, `fzero` решения, их относительные ошибки, и количества затраченных итераций.

16. (Моулер) В следующей таблице приведены данные о длине тормозного пути d некоторого автомобиля в зависимости от его скорости v в начале пути:

v	d
0	0
10	5
20	20
30	46
40	70
50	102
60	153

Какова должна быть максимальная скорость автомобиля, чтобы длина тормозного пути не превосходила бы 60. Написать программу (скрипт), решающую задачу следующими способами:

(a) кусочной линейной интерполяцией,

(b) кусочной кубической интерполяцией,

(c) обратной кубической интерполяцией.

Придумать и реализовать новый способ. Нарисовать графики построенных интерполянтов. Сравнить результаты.

17. Вычисление элементов произведения C двух квадратных матриц $A = (a_{ij})$ и $B = (b_{ij})$ по формулам

$$c_{ij} = \sum_k a_{ik} b_{kj}$$

легко реализуется с помощью трех вложенных циклов: по i, по j и по k. Различные варианты алгоритма можно получить перестановкой циклов. Напишите шесть функций `multijk`, `multikj`, `multjik`, ..., реализующих каждый из этих вариантов. Для серии случайно сгенерированных матриц больших размеров замерьте время, требуемое этими функциями и стандартной операцией `*`. Нарисуйте графики зависимости времени от порядка матриц. Объясните, почему требуется различное время на выполнение каждой из написанных вами функций.

18. Система $(A^{\mathrm{T}}A + \alpha E)x = A^{\mathrm{T}}b$ называется *регуляризованной* по отношению к системе $Ax = b$. Параметр α называется параметром регуляризации. *Метод регуляризации* А.Н. Тихонова заключается в выборе последовательности $\alpha_k \to 0$ (например, геометрической прогрессии) и решении регуляризованных систем $(A^{\mathrm{T}}A + \alpha_k E)x = A^{\mathrm{T}}b$ для конечного числа значений α_k, и выборе из всех полученных решений того, при котором норма невязки для исходной системы минимальна.

    Написать функцию

    ```
    [x0, discr0, alpha0, x, discr] = tikhonov(A, b, alpha),
    ```

    в которой `A` и `b` — матрица и левая часть исходной системы, `alpha` — вектор, содержащий набор параметров регуляризации, `x` — матрица, по столбцам которой записаны регуляризованные решения для соответсвующего значения параметра регуляризации, `discr` — вектор полученных таким образом невязок, `x0` — регуляризованное решение (из найденных) с минимальной нормой невязки, `discr0`, `alpha0` — соответствующие ему значения нормы невязки и параметра регуляризации.

Провести эксперимент, аналогичный № 10.

19. Метод Штрассена умножения матриц заключается в разбиении матрицы на 4 равные клетки

$$\begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix} = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \cdot \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix},$$

и нахождения произведения по формулам:

$$\begin{aligned}
D_1 &= (A_{11} + A_{22}) \cdot (B_{11} + B_{22}),\\
D_2 &= (A_{12} - A_{22}) \cdot (B_{21} + B_{22}),\\
D_3 &= (A_{11} - A_{21}) \cdot (B_{11} + B_{12}),\\
D_4 &= (A_{11} + A_{12}) \cdot B_{22},\\
D_5 &= (A_{21} + A_{22}) \cdot B_{11},\\
D_6 &= A_{11} \cdot (B_{12} - B_{22}),\\
D_7 &= A_{22} \cdot (-B_{11} + B_{21}),
\end{aligned}$$

$$\begin{aligned}
C_{11} &= D_1 + D_2 - D_4 + D_7,\\
C_{12} &= D_4 + D_6,\\
C_{21} &= D_5 + D_7,\\
C_{22} &= D_1 - D_3 - D_5 + D_6.
\end{aligned}$$

Для нахождения произведений вида $(A_{11}+A_{22})\cdot(B_{11}+B_{22})$ и т. п. рекурсивно используется тот же алгоритм. Если обычный алгоритм умножения квадратных матриц порядка n использует $O(n^3)$ операций, то алгоритм Штрессана использует только $O(n^{\log_2 7}) = O(n^{2.8})$. Напишите функцию `strassen(A, B)`, реализующую этот алгоритм для матриц порядка 2^k. Экспериментально постройте графики зависимоти времени работы алгоритма Штрассена и обычного алгоритма от порядка матрицы. Удастся ли вам дойти до таких размеров матриц, что ваша реализация алгоритма Штрассена будет быстрее обычного алгоритма умножения?

20. Рассмотрите модель Вольтерра́–Лотке простой экосистемы

$$\begin{cases} \dfrac{dr}{dt} = 2r - \alpha r f,\\ \dfrac{df}{dt} = -f + \alpha r f, \end{cases}$$

$$r(0) = r_0, \qquad f(0) = f_0,$$

где f — численность лис, r — численность кроликов (α, например, 0.01; r_0, f_0 могут изменяться от нескольких единиц до нескольких тысяч)

Напишите функцию, на вход которой можно подавать величины α, r_0, f_0. По заданным параметру и начальным условиям функция должна строить три графика: графики зависимости $f(t)$, $r(t)$ и фазовый портрет (траекторию в плоскости (r, f)).

Экспериментально попытайтесь найти все качественно различные фазовые траектории. Каков содержательный смысл в соответствующих этим траекториям решениях?