Санкт-Петербургский политехнический университет Петра Великого
Институт компьютерных наук и технологий
Высшая школа интеллектуальных систем и суперкомпьютерных технологий

Отчёт по лабораторной работе № 2

Дисциплина: Вычислительная математика

Выполнил студент гр. 3530901/10001 _______________ Д.Л. Симоновский
(подпись)

Руководитель __________________________ В.Н. Цыган
(подпись)

“17” февраль 2023 г.

Санкт-Петербург
2023

Оглавление

Задание: 2

Инструменты: 2

Ход выполнения работы: 2

Порядок действий: 2

Первая задача: 2

Полная формулировка: 2

Разбиваем на задачи: 2

Пункт 1: 3

Пункт 2: 3

Вторая задача: 4

Полная формулировка: 4

Решение: 4

Результат: 6

Анализ: 7

Вывод: 7

Листинг кода: 8

Ссылки: 9

Задание:

Вариант 11:

Семейство линейных систем, представленных в следующем виде зависит от p:

$$\begin{pmatrix} p-29 & 6 & -6 & -4 & -3 & -8 & -5 & 5 \\ 6 & -13 & -3 & 5 & 4 & 3 & 1 & 7 \\ 5 & -5 & -1 & 7 & 2 & 0 & 7 & 1 \\ 5 & -5 & 5 & 6 & 4 & -7 & 4 & 0 \\ 4 & 4 & 7 & -4 & 9 & -8 & -8 & -4 \\ -4 & 5 & -4 & 1 & 0 & 12 & 0 & 6 \\ -3 & -2 & -4 & 2 & -8 & -3 & 16 & 4 \\ 7 & 5 & 0 & 2 & 0 & -6 & 8 & -12 \end{pmatrix} \begin{pmatrix} x^1 \\ x^2 \\ x^3 \\ x^4 \\ x^5 \\ x^6 \\ x^7 \\ x^8 \end{pmatrix} = \begin{pmatrix} 4p-175 \\ 133 \\ 110 \\ 112 \\ 17 \\ 32 \\ 13 \\ -18 \end{pmatrix}$$

Решить линейные системы, используя программы DECOMP и SOLVE, при $p = 1.0, 0.1, 0.01, 0.0001, 0.000001$. Сравнить решение системы $Ax_2 = b$ с решением системы $A^T A x_1 = A^T b$, полученной из исходной, левой трансформацией Гаусса.

Проанализировать связь числа обусловленности *cond* и величины $\delta = \frac{\|x_1 - x_2\|}{\|x_1\|}$

Инструменты:

Для работы был выбран язык программирования Python версии 3.11 по причине удобства его использования для поставленной задачи. Были выбраны следующие библиотеки:

- NumPy – для большей скорости расчетов, для функций расчета нормы и обусловленности матрицы
- SciPy – для функции расчета решения матрицы
- PrettyTable – для красивого вывода в консоль таблицы

Ход выполнения работы:

Порядок действий:

Поставленное задание легко можно разбить на две глобальные задачи:

1. Вычислительная математика.
   Проанализировать связь между системами $Ax_1 = b$ и $A^T A x_1 = A^T b$.
   Проанализировать теоретическую связь между *cond* и величиной $\delta = \frac{\|x_1 - x_2\|}{\|x_1\|}$
2. Выполнить расчеты для проверки теоретических выкладок, а также решить систему при $p = 1.0, 0.1, 0.01, 0.0001, 0.000001$

Первая задача:

Полная формулировка:

Сравнить решение системы $Ax_1 = b$ с решением системы $A^T A x_2 = A^T b$, полученной из исходной, левой трансформацией Гаусса.

Проанализировать связь числа обусловленности *cond* и величины $\delta = \frac{\|x_1 - x_2\|}{\|x_1\|}$.

Разбиваем на задачи:

В поставленной задаче явно есть несколько этапов:

1. Сравнить решение системы $Ax_1 = b$ с решением системы $A^T A x_2 = A^T b$
2. Проанализировать связь числа обусловленности *cond* и величины $\delta = \frac{\|x_1 - x_2\|}{\|x_1\|}$.

Пункт 1:

Первый пункт является очевидным, поэтому опустим доказательство и перейдем сразу к выводу.

Система уравнений $Ax_1 = b$ эквивалентна системе $A^T A x_2 = A^T b$ с точностью до погрешности вычислений.

Пункт 2:

Этот пункт я бы хотел вывести подробнее т.к. зависимость не столь очевидна:

$$\begin{cases} Ax_1 = b \\ A^T A x_2 = A^T b \end{cases}$$

Умножим второе уравнение на $A^{T^{-1}}$, таким образом получим $Ax_2 = b$, однако необходимо не забыть про вычислительную погрешность, для её симуляции добавим вектор r:

$$\begin{cases} Ax_1 = b \\ Ax_2 = b + r \end{cases}$$

Вычтем из первой системы уравнений, вторую (r для простоты запишем со знаком плюс т.к. это не влияет на ход решения):

$$Ax_1 - Ax_2 = b - b + r$$
$$A(x_1 - x_2) = r$$
$$(x_1 - x_2) = A^{-1} r$$

Далее воспользуемся свойством нормы матрицы $\|A + B\| \le \|A\| + \|B\|$:

$$\|x_1 - x_2\| \le \|A^{-1} r\|$$

Теперь используем свойство нормы матрицы $\|A * B\| \le \|A\| * \|B\|$:

$$\|x_1 - x_2\| \le \|A^{-1} r\| \le \|A^{-1}\| * \|r\|$$

Поделим обе части неравенства на $\|x_1\|$:

$$\frac{\|x_1 - x_2\|}{\|x_1\|} \le \frac{\|A^{-1}\| * \|r\|}{\|x_1\|}$$

Умножим правую часть неравенства на дробь $\frac{\|b\|}{\|b\|}$:

$$\frac{\|x_1 - x_2\|}{\|x_1\|} \le \frac{\|A^{-1}\| * \|r\|}{\|x_1\|} \frac{\|b\|}{\|b\|} \quad (1)$$

Сделаем преобразования над первым уравнением в исходной системе, используя описанные выше свойства:

$$Ax_1 = b$$
$$\|A\| * \|x_1\| \ge \|b\|$$
$$\frac{1}{\|x_1\|} \le \frac{\|A\|}{\|b\|}$$

Подставим полученное в (1):

$$\frac{\|x_1 - x_2\|}{\|x_1\|} \le \frac{\|A^{-1}\| * \|r\| * \|A\|}{\|b\|} \frac{\|b\|}{\|b\|}$$

Как мы знаем, $\|A^{-1}\| * \|A\|$ это определение числа обусловленности cond(A):

$$\frac{\|x_1 - x_2\|}{\|x_1\|} \le cond(A) \frac{\|r\|}{\|b\|}$$

Таким образом мы получили отношение величины $\delta = \frac{\|x_1 - x_2\|}{\|x_1\|}$ и cond(A)

Вторая задача:

Полная формулировка:

Семейство линейных систем, представленных в следующем виде зависит от p:

$$\begin{pmatrix} p-29 & 6 & -6 & -4 & -3 & -8 & -5 & 5 \\ 6 & -13 & -3 & 5 & 4 & 3 & 1 & 7 \\ 5 & -5 & -1 & 7 & 2 & 0 & 7 & 1 \\ 5 & -5 & 5 & 6 & 4 & -7 & 4 & 0 \\ 4 & 4 & 7 & -4 & 9 & -8 & -8 & -4 \\ -4 & 5 & -4 & 1 & 0 & 12 & 0 & 6 \\ -3 & -2 & -4 & 2 & -8 & -3 & 16 & 4 \\ 7 & 5 & 0 & 2 & 0 & -6 & 8 & -12 \end{pmatrix} \begin{pmatrix} x^1 \\ x^2 \\ x^3 \\ x^4 \\ x^5 \\ x^6 \\ x^7 \\ x^8 \end{pmatrix} = \begin{pmatrix} 4p-175 \\ 133 \\ 110 \\ 112 \\ 17 \\ 32 \\ 13 \\ -18 \end{pmatrix}$$

Решить линейные системы, используя программы DECOMP и SOLVE, при $p = 1.0, 0.1, 0.01, 0.0001, 0.000001$.

Решение:

В основе задания лежат матрица A и вектор b, зависящие от константы p. Напишем функции для получения этих объектов:

```python
def get_matrix_A(p):
    """Для получения значения матрицы A по p"""
    return np.array([
        [p - 29, 6, -6, -4, -3, -8, -5, 5],
        [6, -13, -3, 5, 4, 3, 1, 7],
        [5, -5, -1, 7, 2, 0, 7, 1],
        [5, -5, 5, 6, 4, -7, 4, 0],
        [4, 4, 7, -4, 9, -8, -8, -4],
        [-4, 5, -4, 1, 0, 12, 0, 6],
        [-3, -2, -4, 2, -8, -3, 16, 4],
        [7, 5, 0, 2, 0, -6, 8, -12]
    ])


def get_vector_b(p):
    """Для получения значения матрицы b по p"""
    return np.vstack(np.array([4 * p - 175, 133, 110, 112, 17, 32, 13, -18]))
```

Далее напишем цикл для p = 1.0, 0.1, 0.01, 0.0001, 0.000001, используем для этого for-each по массиву из требуемых элементов, в начале выведем на экран переменную p, чтоб было понятно, для какого этапа производятся вычисления, а также получим матрицу A и вектор b:

```python
for p in [1.0, 0.1, 0.01, 0.0001, 0.000001]:
    print(f'p={p}')
    matrix_A = get_matrix_A(p)
    vector_b = get_vector_b(p)
```

Далее необходимо получить транспонированную матрицу A для второй системы уравнений, сделать это можно вызвав соответствующий метод на матрице A:

```python
matrix_A_transpose = matrix_A.transpose()
```

Теперь посчитаем произведение матрицы A^T на матрицу A и матрицы A^T на вектор b:

```python
# Получение матрицы A_trans * A
matrix_A_transpose_A = matrix_A_transpose.dot(matrix_A)
# Получение вектора A_trans * b
vector_A_ransponse_b = matrix_A_transpose.dot(vector_b)
```

На данный момент у нас были получены все компоненты для решения систем уравнений $\begin{array}{c} Ax_1 = b \\ A^T A x_2 = A^T b \end{array}$, осталось вызвать для них DECOMP и SOLVE, однако в python есть метод scipy.linalg.solve, он принимает на вход матрицу A и матрицу b, в том числе в общем виде, и выдает результирующую матрицу. Таким образом нет необходимости раскладывать матрицу на LU, можно сразу передать полученные выше матрицы для системы в ответ:

```python
# Решение уравнения Ax=b
result = solve(matrix_A, vector_b)
# Решение уравнения A_t * A * x = A_t * b
result_transform = solve(matrix_A_transpose_A, vector_A_ransponse_b)
```

Выведем на экран информацию об обусловленности матрицы A и матрицы $A^T A$. В качестве функции для вычисления нормы будем использовать норму Фробениуса. Для этого используем numpy.linalg.cond, передавая в качестве параметра p="fro":

```python
print(f'Обусловленность матрицы A: {cond(matrix_A, p="fro")} '
      f'A_T * A: {cond(matrix_A_transpose_A, p="fro")}')
```

Так же необходимо вывести полученные результаты решения систем уравнений $Ax_1 = b$ и $A^T A x_2 = A^T b$ и разницу между их решениями, для этого напишем функцию, воспользовавшись библиотекой prettytable:

```python
def print_answer(x_1, x_2):
    """Выводит на экран два решения и разницу между ними"""
    pt = PrettyTable()
    pt.add_column('x', [f'x_{i + 1}' for i in range(0, len(x_1))])
    pt.add_column('x_1', [x[0] for x in x_1])
    pt.add_column('x_2', [x[0] for x in x_2])
    pt.add_column('delta', [x[0] for x in x_1 - x_2])
    print(pt)
```

Ну и последнее – выведем на экран значение $\delta = \frac{\|x_1 - x_2\|}{\|x_1\|}$, чтоб проверить теоретические выкладки из пункта 1. Для вычисления нормы будем использовать numpy.linalg.norm и передавать в качестве аргумента ord="fro", дабы в качестве нормы функция использовала норму Фробениуса:

```python
print(f'||x_1-x_2||/||x_1||='
      f'{norm(result - result_transform, ord="fro") / norm(result, ord="fro")}')
```

Результат:

В итоге программа выдала такие значения:

p=1.0

Обусловленность матрицы A: 8519.140985499118   A_T * A: 39385779.397530325

x	x_1	x_2	delta
x_1	3.9999999999995453	4.000000001103602	-1.1040564018571786e-09
x_2	-4.4082894516311453e-13	1.0732149439944653e-09	-1.0736557729396284e-09
x_3	3.0000000000004263	2.9999999989634807	1.0369456404646371e-09
x_4	5.999999999999566	6.000000001052607	-1.0530412097864428e-09
x_5	8.000000000000455	7.9999999989036095	1.0968452812676333e-09
x_6	1.0000000000004332	0.9999999989468525	1.0535806671541081e-09
x_7	4.0000000000004405	3.9999999989342157	1.0662248861592616e-09
x_8	6.999999999999556	7.000000001081983	-1.0824274809806411e-09

||x_1-x_2||/||x_1||=2.1920150022324254e-10

==

p=0.1

Обусловленность матрицы A: 88317.35680165827   A_T * A: 4263883451.436119

x	x_1	x_2	delta
x_1	4.000000000001828	4.0000000395963	-3.95944717013208e-08
x_2	1.8225821205505956e-12	3.9487278308107974e-08	-3.948545572598742e-08
x_3	2.9999999999981837	2.9999999606443706	3.93538130971649e-08
x_4	6.000000000001817	6.000000039413292	-3.9411474972439464e-08
x_5	7.999999999998175	7.999999960429615	3.956855998410447e-08
x_6	0.9999999999981809	0.9999999605847291	3.941345183555711e-08
x_7	3.9999999999981797	3.9999999605394136	3.94587660323964e-08
x_8	7.000000000001824	7.000000039518743	-3.9516918626247843e-08

||x_1-x_2||/||x_1||=8.07895675426055e-09

==

p=0.01

Обусловленность матрицы A: 886347.6011370448   A_T * A: 429776614560.4676

x	x_1	x_2	delta
x_1	3.999999999979987	3.9999609237916287	3.90761883584112e-05
x_2	-2.0008643552401878e-11	-3.906544928547747e-05	3.9065429276833914e-05
x_3	3.0000000000200013	3.000039052457923	-3.905243792168278e-05
x_4	5.999999999979998	5.999960941852425	3.9058127573099455e-05
x_5	8.00000000002001	8.000039073650692	-3.907363068123004e-05
x_6	1.0000000000200047	1.0000390583432857	-3.905832328099379e-05
x_7	4.000000000020005	4.000039062815148	-3.906279514254152e-05
x_8	6.9999999999799885	6.999960931445632	3.906853435609037e-05

||x_1-x_2||/||x_1||=7.9948412920894e-06

==

p=0.0001

Обусловленность матрицы A: 88669727.7088329   A_T * A: 4520269532764875.0

x	x_1	x_2	delta
x_1	4.000000008337465	4.428797856603765	-0.4287978482663002
x_2	8.337442102242623e-09	0.4287966759683974	-0.4287966763095525
x_3	2.999999991662582	2.571204749621178 3	0.4287952420414038 5
x_4	6.000000008337426	6.428795874723415 5	-0.4287958663859896 5
x_5	7.999999991662542	7.571202424060814 5	0.428797567601727 7
x_6	0.9999999916625724	0.571204103799177 7	0.428795887863394 6
x_7	3.999999991662563	3.57120361308545 6	0.428796378577107 2
x_8	7.000000008337448	7.4287970167007975	-0.4287970083633494 4

||x_1-x_2||/||x_1||=0.08775656200579741

==

p=1e-06

Обусловленность матрицы A: 8867008514.835018 A_T * A: 1.591928701060035e+17

x	x_1	x_2	delta
x_1	4.000000416316432	13.79099402211575	-9.790993605799319
x_2	4.163164196638241e-07	9.790993752534318	-9.790993336217898
x_3	2.99999583683593	-6.790993427021074	9.79099301070466 8
x_4	6.000000416316413	15.79099356958131 8	-9.790993153264903
x_5	7.999999583683571	-1.7909939580299374	9.79099354171350 8
x_6	0.9999995836835879	-8.790993574485396	9.790993158168984
x_7	3.9999995836835827	-5.790993686532946	9.790993270216529
x_8	7.000000416316423	16.79099383033575	-9.790993414019328

||x_1-x_2||/||x_1||=2.0038031860726955

==

Так же в ходе работы было выдано предупреждение:

```
LinAlgWarning: Ill-conditioned matrix (rcond=2.92164e-18): result may not be accurate.
```

Анализ:

Из результата работы программы можно сделать ряд выводов:

1. Чем меньше переменная p, тем больше обусловленность матрицы A
2. Чем больше обусловленность матрицы A, тем больше обусловленность матрицы $A^{\mathrm{T}}A$
3. Чем больше обусловленность матрицы A, тем больше разброс между решениями системы уравнений $Ax_1 = b$ и $A^TAx_2 = A^Tb$
4. Чем больше обусловленность матрицы A, тем больше $\delta = \frac{\|x_1-x_2\|}{\|x_1\|}$

Все эти пункты соответствуют математическим ожиданиям.

Так же хотелось бы отдельно отметить LinAlgWarning. Из его описания понятно, что он говорит о том, что обусловленность матрицы A слишком большая, что приведет к неточностям.

Вывод:

В ходе работы я ознакомился с аналогом подпрограммы SOLVE, поработал с функциями для вычисления нормы матрицы и обусловленности в языке Python. По результатам работы видно, что δ зависит от обусловленности матрицы A, а также, что при увеличении этой обусловленности погрешность решения сильно возрастает.

Листинг кода:

```
import numpy as np
from prettytable import PrettyTable
from scipy.linalg import solve
from numpy.linalg import cond, norm


def get_matrix_A(p):
    """Для получения значения матрицы A по p"""
    return np.array([
        [p - 29, 6, -6, -4, -3, -8, -5, 5],
        [6, -13, -3, 5, 4, 3, 1, 7],
        [5, -5, -1, 7, 2, 0, 7, 1],
        [5, -5, 5, 6, 4, -7, 4, 0],
        [4, 4, 7, -4, 9, -8, -8, -4],
        [-4, 5, -4, 1, 0, 12, 0, 6],
        [-3, -2, -4, 2, -8, -3, 16, 4],
        [7, 5, 0, 2, 0, -6, 8, -12]
    ])


def get_vector_b(p):
    """Для получения значения матрицы b по p"""
    return np.vstack(np.array([4 * p - 175, 133, 110, 112, 17, 32, 13, -18]))


def print_answer(x_1, x_2):
    """Выводит на экран два решения и разницу между ними"""
    pt = PrettyTable()
    pt.add_column('x', [f'x_{i + 1}' for i in range(0, len(x_1))])
    pt.add_column('x_1', [x[0] for x in x_1])
    pt.add_column('x_2', [x[0] for x in x_2])
    pt.add_column('delta', [x[0] for x in x_1 - x_2])
    print(pt)


def main():
    for p in [1.0, 0.1, 0.01, 0.0001, 0.000001]:
        print(f'p={p}')
        # Получение матрицы A
        matrix_A = get_matrix_A(p)
        # Получение вектора b
        vector_b = get_vector_b(p)
        # Получение транспонированной матрицы A
        matrix_A_transpose = matrix_A.transpose()
        # Получение матрицы A_trans * A
        matrix_A_transpose_A = matrix_A_transpose.dot(matrix_A)
        # Получение вектора A_trans * b
        vector_A_ransponse_b = matrix_A_transpose.dot(vector_b)
        # Решение уравнения Ax=b
        result = solve(matrix_A, vector_b)
        # Решение уравнения A_t * A * x = A_t * b
        result_transform = solve(matrix_A_transpose_A, vector_A_ransponse_b)
        # Выводим обусловленность матриц
        print(f'Обусловленность матрицы A: {cond(matrix_A, p="fro")} '
              f'A_T * A: {cond(matrix_A_transpose_A, p="fro")}')
        # Выводим вектора ответа и дельту
        print_answer(result, result_transform)
        # Выведем ||x_1-x_2||/||x_1||
        print(f'||x_1-x_2||/||x_1||='
              f'{norm(result - result_transform, ord="fro") / norm(result, ord="fro")}')
        print('=' * 80)


if __name__ == '__main__':
    main()
```

Ссылки:

Листинг код: github.com

Документация по SciPy: docs.scipy.org