Вычислительные методы

Лекция 3:
Итерационные методы решения СЛАУ

Сивцева Вера Исаевна

Прямые методы (точные)

Метод решения задачи относят к классу точных, если в предположении отсутствия округлений он дает точное решение задачи после конечного числа арифметических и логических операций.

- метод Гаусса
- LU-разложение
- метод Холецкого
- QR-разложение
- Метод наименьших квадратов

Итерационные методы

Итерационные методы позволяют найти приближенное решение системы путем построения последовательности приближений (итераций), начиная с некоторого начального приближения.

- метод простых итераций
- метод Якоби
- метод Зейделя
- метод сопряженных градиентов

Недостатки прямых методов

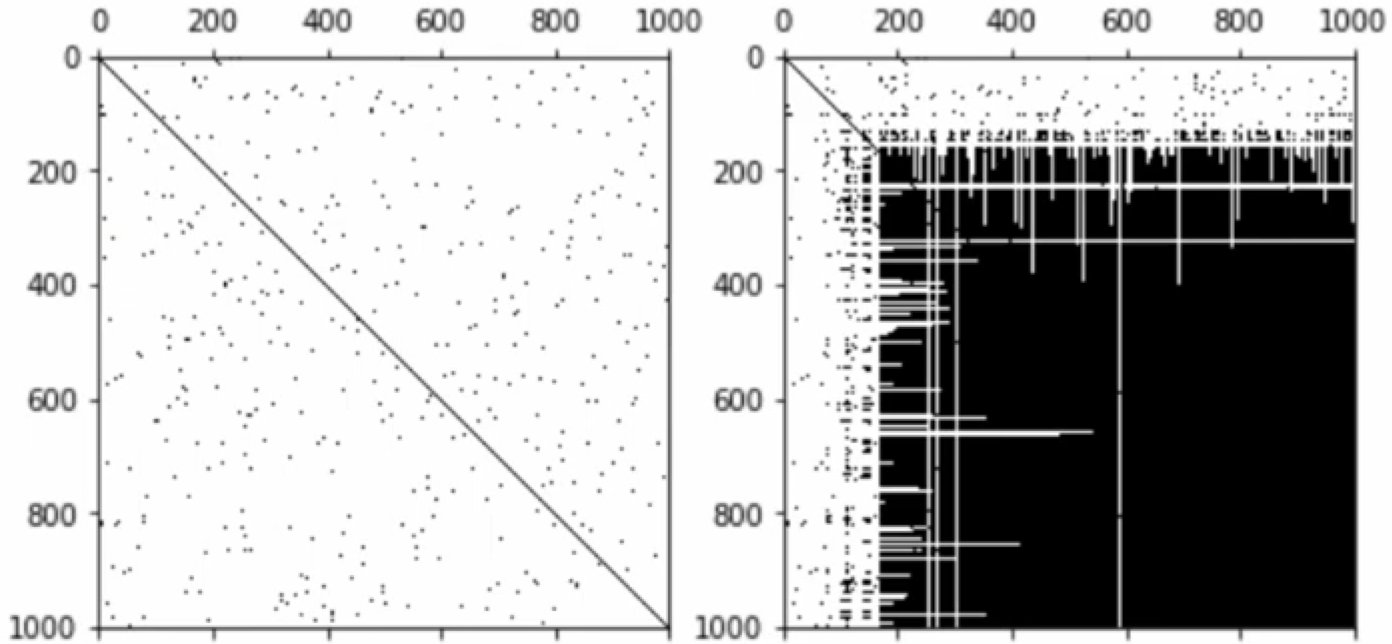
- Сложность $O(n^3)$ для матриц общего вида
- Нужны элементы матрицы в явном виде

$$F : \mathbb{R}^n \to \mathbb{R}^n; \; J = \frac{\partial F}{\partial x} = [\frac{\partial F_i}{\partial x_j}(x_0)]$$

$$J(x_0)\Delta x = F(x_0 + \Delta x) - F(x_0) + O(\Delta x^2)$$

- Нельзя найти приближение к решению с заданной погрешностью
- Сложно использовать специальную структуру матрицы
  Пример: если A - разреженная, матрицы L и U, Q и R не будут в общем случае разреженными

Потеря разреженности



Общий вид итерационного метода

$$
\begin{aligned}
Ax = b \iff x &= Sx + f \\
x^k &= Sx^{k-1} + f \\
e^k = (x^k - x) &= S(x^{k-1} - x) = \\
&= S^k(x^0 - x) = S^k e^0
\end{aligned}
$$

- Критерий сходимости:
  $e^k \to 0 \iff S^k e^0 \to 0 \; \forall e^0 \iff \rho(S) < 1$
- Достаточное условие сходимости: $\|S\| = q < 1$

$$\|e^k\| = \|S^k e^0\| \leq \|S\|^k \|e^0\| = q^k \|e^0\|$$

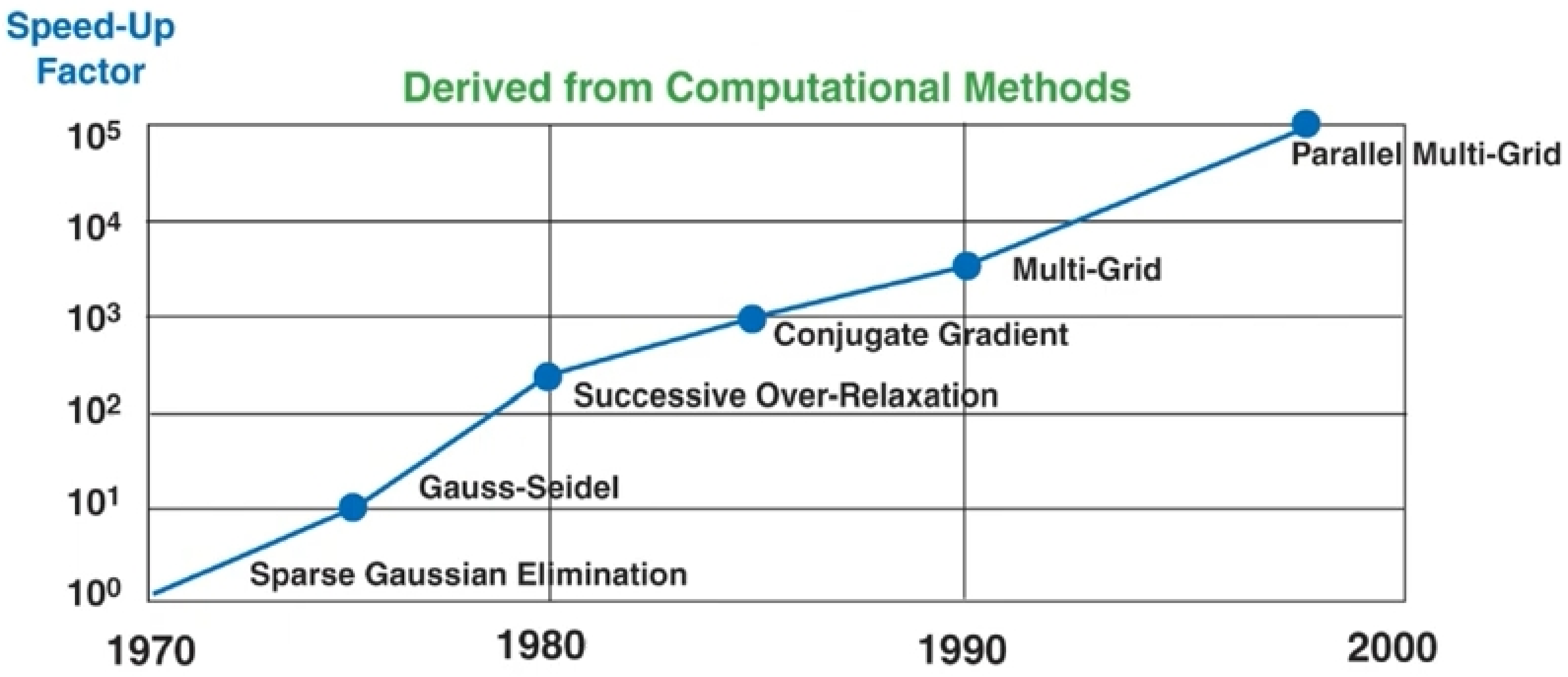
- Нужно задать начальное приближение x^0 и *критерий остановки*, например: $\|Ax^k - b\| \leq \epsilon$

Преимущества итерационных методов

$$x^k = Sx^{k-1} + f$$

- Сложность $O(Kn^2)$, где K - число итераций.
- $\|e^k\| \leq q^k\|e^0\| \leq \epsilon \Rightarrow k \geq \dfrac{\log(\epsilon/\|e^0\|)}{\log q}$
- Нужна только процедура для вычисления Sx
- Матрично-векторное умножение легко распараллеливается с высокой эффективностью.
- Для разреженных матриц сложность $O(K \times NNZ(S))$

Развитие итерационных методов



Список 10 важнейших алгоритмов 20-го века, *Computing in Science and Engineering*, 2001 год:

- *итерационные методы* подпространств Крылова

Метод Ричардсона

- Рассмотрим систему $Ax = b,\ A = A^T > 0$:

$$\begin{aligned}
&Ax = b \\
&\tau(Ax - b) = 0 \\
&x - \tau(Ax - b) = x \\
&\underline{x^{k+1} = x^k - \tau(Ax^k - b) = (I - \tau A)x^k + \tau b}
\end{aligned}$$

τ - итерационный параметр. $S = I - \tau A$

- Критерий сходимости:

$$\mathrm{sp}(I - \tau A) = 1 - \tau\ \mathrm{sp}(A),\ \lambda_i(A) > 0$$
$$|1 - \tau\lambda_i| < 1,\ i = 1, \ldots, n \Rightarrow$$
$$(1 - \tau\lambda_i)^2 < 1 \Rightarrow\ \tau\lambda_i(\tau\lambda_i - 2) < 0 \Rightarrow \underline{0 < \tau < \frac{2}{\lambda_{max}}}$$

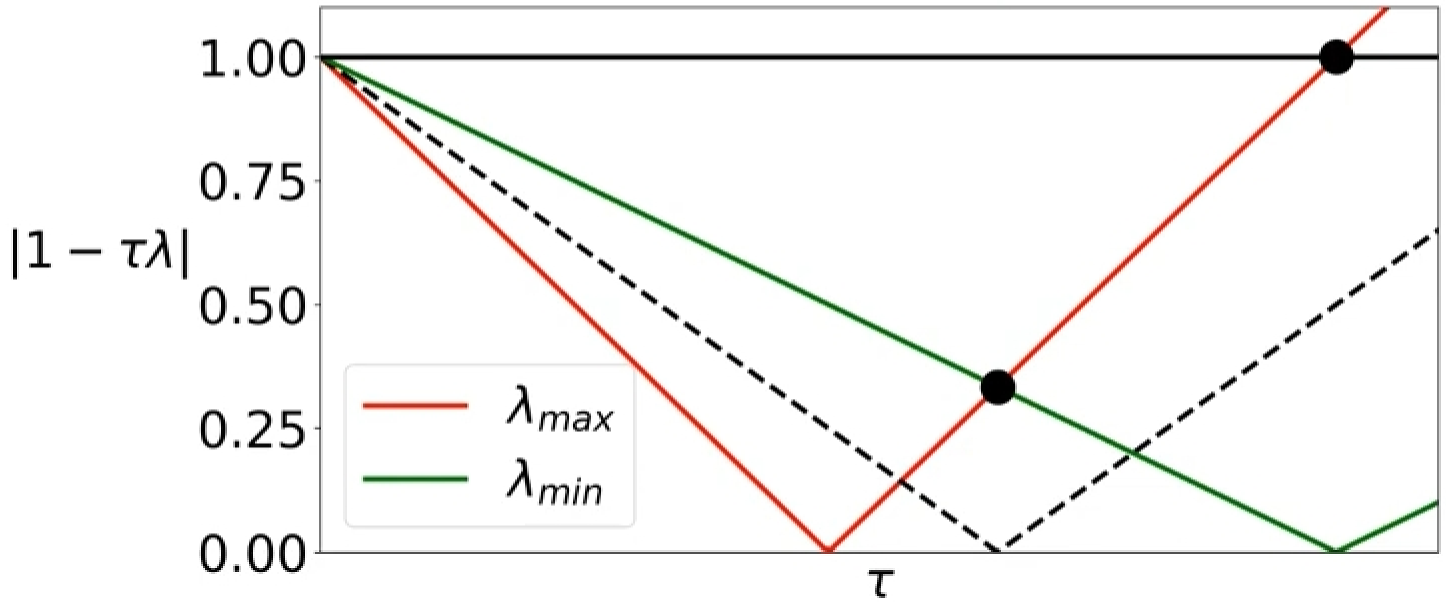
Оптимальный параметр

- $\|e^k\|_2 = \|(I - \tau A)^k e^0\|_2 \le \|(I - \tau A)\|_2^k \|e^0\|_2$
- $\|(I - \tau A)\|_2 = \max\limits_{\lambda \in sp(A)} |1 - \tau\lambda|$
- Часто, известны только оценки границы спектра:

$$sp(A) \in [m, M]$$

- $\tau^* = \arg\min\limits_{\tau} \max\limits_{\lambda \in [m, M]} |1 - \tau\lambda|$

Оптимальный параметр



Оптимальное значение:

$$1 - \tau m = \tau M - 1 \Rightarrow \tau^* = \frac{2}{M+m} \Rightarrow$$

$$\|(I - \tau^* A)\|_2 = q^* = \frac{M-m}{M+m} = \frac{M/m - 1}{M/m + 1} = \frac{cond_2(A) - 1}{cond_2(A) + 1}$$

Скорость сходимости зависит от обусловленности!

Методы Якоби и Гаусса-Зейделя

- Общая идея: $A = M - N$, M - обратимая и *легко обращается* (система решается за $\ll O(n^3)$):
  $Mx = Nx + b \Rightarrow Mx^{k+1} = Nx^k + b$, $x^{k+1} = M^{-1}(Nx^k + b)$
- Легко обращаются:
  - Диагональные матрицы – $O(n)$
  - Треугольные матрицы – $O(n^2)$
- $A = L + D + U$,
  L - строго нижнетреугольная часть
  D - диагональная часть
  U - строго верхнетреугольная часть.
- **Предполагается, что D - невырожденная.**
- Метод Якоби: $Dx^{k+1} = -(L + U)x^k + b$
- Метод Зейделя: $(L + D)x^{k+1} = -Ux^k + b$

Метод Якоби

$$x^{k+1} = -D^{-1}(L+U)x^k + D^{-1}b$$

Достаточное условие сходимости метода Якоби

Если матрица A имеет строчное диагональное преобладание, то метод Якоби сходится.

- докажем, что $\|D^{-1}(L+U)\|_\infty < 1$
- $|a_{ii}| > \sum\limits_{j \neq i} |a_{ij}|,\ i = 1, \ldots, n$ (определение диагонального преобладания) $\Rightarrow$
- $\|D^{-1}(L+U)\|_\infty = \max\limits_i \sum\limits_{j \neq i} \left|\frac{a_{ij}}{a_{ii}}\right| = \max\limits_i \frac{\sum_{i \neq j} |a_{ij}|}{|a_{ii}|} < 1 \square$

Алгоритм метода Якоби

```
A
b
xkp1[:] = 0 # начальное приближение
xk[:] = 0
while (||dot(A, xkp1) - b|| > tol)
    xk = xkp1
    for i = 1, n
        xkp1[i] = b[i]
        for j = 1, n; j \= i
            xkp1[i] = xkp1[i] - A[i,j] * xk[j]
        xkp1[i] = xkp1[i] / A[i,i]
```

- Метод Якоби редко используется в *чистом виде*
- Acceleration of the Jacobi iterative method by factors exceeding 100 using scheduled relaxation, JCP, 2014

Метод Зейделя

$$x^{k+1} = -(L+D)^{-1}Ux^k + (L+D)^{-1}b$$

Достаточное условие сходимости метода Зейделя

Если $A \in \mathbb{R}^{n \times n}$, $A = A^T > 0$, то метод Зейделя сходится.

- $A = A^T > 0 \Rightarrow \quad A = L + D + L^T$
- $\langle Ae_k, e_k \rangle = a_{kk} > 0 \Rightarrow D > 0$
- $x^{k+1} = -(L+D)^{-1}L^Tx^k + (L+D)^{-1}b = \boxed{(I - (L+D)^{-1}A)}\, x^k + (L+D)^{-1}b$
- Можно ввести A-норму: $\|v\|_A^2 \equiv \langle Av, v \rangle = \langle v, v \rangle_A$, где $\langle \cdot, \cdot \rangle$ - обычное скалярное произведение

Метод Зейделя(2)

$$\|e^{k+1}\|_A^2 \equiv \left\langle Ae^{k+1}, e^{k+1} \right\rangle =$$

$$= \left\langle A(I-(L+D)^{-1}A)e^k, (I-(L+D)^{-1}A)e^k \right\rangle =$$

$$= \left\langle Ae^k - A(L+D)^{-1}Ae^k, e^k - (L+D)^{-1}Ae^k \right\rangle \boxed{=}$$

Обозначим $\underline{v = (L+D)^{-1}Ae^k; Ae^k = (L+D)v \ (1)}$

$$\boxed{=} \left\langle Ae^k, e^k \right\rangle - \left\langle Av, e^k \right\rangle - \left\langle Ae^k, v \right\rangle + \left\langle Av, v \right\rangle =$$

$$= \|e^k\|_A^2 - \left(2\left\langle (L+D)v, v \right\rangle - \left\langle (L+D+L^T)v, v \right\rangle \right) =$$

$$= \|e^k\|_A^2 - \left\langle Dv, v \right\rangle \leq \underline{\|e^k\|_A^2 - d\|v\|_2^2 \ (2)}, \ d = \min_i D_{ii} = \min_i a_i$$

$$\|e^k\|_A^2 = \left\langle Ae^k, e^k \right\rangle = \left\langle (L+D)v, A^{-1}(L+D)v \right\rangle \leq$$

$$\|A^{-1}\|_2 \|(L+D)\|_2^2 \|v\|_2^2 = \lambda_{min}^{-1} \|(L+D)\|_2^2 \|v\|_2^2 \Rightarrow \text{ из } (2):$$

$$\underline{\|e^{k+1}\|_A^2 \leq \left(1 - \lambda_{min} d \|L+D\|_2^{-2}\right) \|e^k\|_A^2} \ \square$$

Алгоритм метода Зейделя

$$(L + D)x^{k+1} = -Ux^k + b$$

```
A
b
xkp1[:] = 0 # начальное приближение
xk[:] = 0
while (||dot(A, xkp1) - b|| > tol)
    xk = xkp1
    for i = 1, n
        xkp1[i] = b[i]
        for j = 1, i-1
            xkp1[i] = xkp1[i] - A[i,j] * xkp1[j]
        for j = i+1, n # матрица U
            xkp1[i] = xkp1[i] - A[i,j] * xk[j]
        xkp1[i] = xkp1[i] / A[i,i]
```

Квадратичные функционалы и линейные системы

- Для $A = A^T \in \mathbb{R}^{n \times n}$ рассмотрим:
$f(x) = \dfrac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle = \dfrac{1}{2}x^T Ax - x^T b$
- Пусть z - решение системы $Az = b$. Если $A > 0$, то:

$$E(x) \equiv f(x) - f(z) = 0.5\langle A(x-z), x-z\rangle > 0, \forall x \neq z$$

$\Rightarrow z$ - точка минимума
- Для произвольной A можно ввести *функционал невязки*:

$$R(x) = \|b - Ax\|_2$$

- Общая идея: $x^{k+1} = x^k - \tau^k(Ax^k - b) = x^k - \tau^k r^k$
$\tau^k = \arg\min\limits_{\tau} F(x^k - \tau r^k),\ F(x) = E(x), R(x)$

Метод наискорейшего спуска

- $F(x) = E(x) = \frac{1}{2}\langle Ax, x\rangle - \langle b, x\rangle + f(z)$
- $\nabla E(x^k) = Ax^k - b = r^k$ (градиентный спуск)
- $x^{k+1} = x^k - \tau^k(Ax^k - b) = x^k - \tau^k \nabla f(x^k)$

$$f(x^{k+1}) = \frac{1}{2}\langle A(x^k - \tau r^k), x^k - \tau r^k\rangle - \langle b, x^k - \tau r^k\rangle =$$

$$= \frac{1}{2}\langle Ax^k, x^k\rangle - \underline{\frac{1}{2}\tau\langle Ax^k, r^k\rangle - \frac{1}{2}\tau\langle Ar^k, x^k\rangle} +$$

$$+ \underline{\underline{\frac{1}{2}\tau^2\langle Ar^k, r^k\rangle}} - \langle b, x^k\rangle + \underline{\tau\langle b, r^k\rangle} =$$

$$= \frac{1}{2}\tau^2\langle Ar^k, r^k\rangle - \tau\langle Ax^k - b, r^k\rangle + \ldots =$$

$$= \frac{1}{2}\tau^2\langle Ar^k, r^k\rangle - \tau\langle r^k, r^k\rangle + \ldots = \Phi(\tau)$$

$$\Phi'(\tau) = \tau\langle Ar^k, r^k\rangle - \langle r^k, r^k\rangle = 0 \Rightarrow \tau^k = \frac{\langle r^k, r^k\rangle}{\langle Ar^k, r^k\rangle}$$

Заключение

- Сравнение прямых и итерационных методов
- Метод Ричардсона
- Методы Якоби и Зейделя
- Методы, основанные на минимизации функционала

Системы линейных алгебраических уравнений (СЛАУ)

Рассмотрим систему из *n* линейных алгебраических уравнений с *N* неизвестными

$$\begin{array}{ccccccccc} a_{11}x_1 & + & a_{12}x_2 & + & \cdots & + & a_{1n}x_n & = & b_1, \\ a_{21}x_1 & + & a_{22}x_2 & + & \cdots & + & a_{2n}x_n & = & b_2, \\ \vdots & \vdots & \vdots & . & \cdots & \vdots & \vdots & \vdots & \vdots \\ a_{n1}x_1 & + & a_{n2}x_2 & + & \cdots & + & a_{nn}x_n & = & b_n. \end{array} \qquad (3.1)$$

В матричном виде система имеет вид

$$\boldsymbol{A}\vec{\boldsymbol{x}} = \vec{\boldsymbol{b}},$$

где ***A*** - квадратная матрица размером ***n***×***n***, ***X*** *и* ***B***- векторы порядка *n* :

$$\boldsymbol{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & . \\ a_{n1} & a_{n2} & \cdots & a_{1n} \end{pmatrix}, \vec{\boldsymbol{x}} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}, \quad \vec{\boldsymbol{b}} = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}.$$

Определение 1

Решением системы линейных уравнений называется такая упорядоченная совокупность чисел $\boldsymbol{x_1 = c_1}$, $\boldsymbol{x_2 = c_2}$, $\boldsymbol{x_n = c_n}$, которая обращает все уравнения системы в верные равенства.

Краткий математический аппарат

Нормой вектора $\vec{x}$ назовём поставленное в соответствие этому вектору неотрицательное число $\|\vec{x}\|$ такое, что

1. $\|\vec{x}\| > \mathbf{0}$ при $\vec{x} \neq \mathbf{0}$, $\|\vec{0}\| = \mathbf{0}$,

2. $\|\alpha\vec{x}\| = \alpha\|\vec{x}\|$, α**=const,**

3. $\|\vec{x} + \vec{y}\| \leq \|\vec{x}\| + \|\vec{y}\|$.

Примеры норм:

1. $\|\vec{x}\| = \sqrt{(\vec{x}, \vec{x})} = \sqrt{x_1^2 + x_2^2 + \cdots + x_n^2}$, **- сферическая норма,**

2. $\|\vec{x}\| = \max\limits_{1 \leq i \leq n} |x_i|$.

Нормой квадратной матрицы ***A*** назовём поставленное в соответствие этому вектору неотрицательное число $\|A\|$ такое, что

1. $\|A\| > \mathbf{0}$ при $A \neq 0$, $\|0\| = \mathbf{0}$;
2. $\|\alpha A\| = \alpha\|A\|$, α**=const;**
3. $\|A + B\| \leq \|A\| + \|B\|$;
4. $\|A \cdot B\| \leq \|A\| \cdot \|B\|$.

```python
# Метод Якоби
import numpy as np
n = 4
A = np.random.rand(n,n) + n * np.eye(n)
print(A)
```

```
[[4.8713287  0.95163772 0.81385025 0.77085225]
 [0.97989821 4.34316456 0.06548597 0.02870745]
 [0.33459906 0.64006384 4.15900138 0.95784457]
 [0.6416472  0.45952207 0.91214638 4.64156387]]
```

```python
b = np.random.rand(n)
# Находим точное решение
x_ex = np.linalg.solve(A, b)
print(A @ x_ex - b)
```

```
[ 2.77555756e-17  0.00000000e+00  0.00000000e+00 -1.11022302e-16]
```

```
%matplotlib inline
import matplotlib
matplotlib.rcParams.update({'font.size': 22})
from matplotlib import pyplot as plt

xkp1 = np.zeros(n)
xk = np.zeros(n)
err = np.array([])
tol = 1e-6
it = 0
while (np.linalg.norm(A @ xkp1 - b) > tol):
    it += 1
    xk[:] = xkp1[:]
    for i in range(n):
        xkp1[i] = b[i]
        for j in range(n):
            if(i != j):
                xkp1[i] = xkp1[i] - A[i,j] * xk[j]
        xkp1[i] = xkp1[i] / A[i,i]
    err = np.append(err, np.linalg.norm(xkp1 - x_ex))
print('it = ', it)
print('||x - x_ex|| = {0:5.2e}'.
      format(np.linalg.norm(xkp1 - x_ex)/np.linalg.norm(x_ex)))
fig, ax = plt.subplots(figsize = (8, 4))
ax.semilogy(err)
ax.grid(True)
```

```
it =  16
||x - x_ex|| = 5.59e-07
```