Задача 1.

Напишите свою собственную функцию вычисления SVD итерационным методом. Сравните её работу с работой библиотечной функции.

Задача 2.

Для разреженных матриц SVD считают иначе. Для таких матриц обычно нужны не все сингулярные числа и вектора, а только первые k. Для этого используются специальные алгоритмы. Просто вызовём функцию из модуля scipy.sparse.linalg

```
u, sigma, v = spsplin.svds(X.asfptype(), k=7)
```

Написать программу, которая генерирует случайные разреженные матрицы и строит график убывания сингулярных чисел.

Задача 3.

Пусть $A = U\Sigma V^*$ SVD для матрицы A. Тогда псевдообратная матрица равна:

$$A^\dagger = V\Sigma^\dagger U^*$$

где $\Sigma^\dagger$ состоит из обращённых ненулевых сингулярных чисел матрицы A. Действительно,

$$A^\dagger = \lim_{\alpha\to 0}(\alpha I + A^*A)^{-1}A^* = \lim_{\alpha\to 0}(\alpha VV^* + V\Sigma^2V^*)^{-1}V\Sigma U^* = \lim_{\alpha\to 0}(V(\alpha I + \Sigma^2)V^*)^{-1}V\Sigma U^*$$
$$= V\lim_{\alpha\to 0}(\alpha I + \Sigma^2)^{-1}\Sigma U^* = V\Sigma^\dagger U^*$$

- Вы можете проверить, что $\Sigma^\dagger$ состоит из обращённых ненулевых сингулярных чисел
- Если сингулярные числа малы, их можно не обращать. Это даст решение менее чувствительное к шуму в правой части

Напишите программу, которая вычисляет псевдообратную матрицу с помощью SVD. Исследуйте, что происходит с числом обусловленности.

Задача 4.

Псевдообратная матрица решает задачу минимизации методом наименьших квадратов. Однако она не является самым оптимальным способом решить эту задачу. Более эффективно использовать QR-разложение. Если A имеет полный ранг, то

$$x = A^\dagger b = (A^*A)^{-1}A^*b = ((QR)^*(QR))^{-1}(QR)^*b = (R^*Q^*QR)^{-1}R^*Q^*b = R^{-1}Q^*b.$$

Таким образом, необходимо решить следующую квадратную систему: $Rx = Q^*b$

- R верхнетреугольная
- Решение требует $\mathcal{O}(n^2)$ операций
- Более устойчивый способ, чем использование псевдообратной матрицы напрямую.

Реализуйте МНК этим методом и сравните с работой алгоритма из задачи 3 на случайных матрицах.