Линейная алгебра

Дима Трушин

Семинар 1

Немного общих слов

В линейной алгебре основные объекты изучения – векторные пространства, линейные отображения и билинейные формы на них. Существует два языка говорить об этих объектах:

1. Язык конкретных векторных пространств.

2. Язык абстрактных векторных пространств.

Давайте я пару слов скажу о каждой из составляющих, чтобы ориентироваться, что там есть и для чего все это нужно.

Конкретные векторные пространства Когда мы в школе изучали геометрию, то мы запомнили, что $\mathbb{R}$ – это прямая. Когда надо было работать с плоскостью, мы на ней вводили координаты и потому каждая точка описывалась парой чисел (x, y). Множество таких пар есть $\mathbb{R}^2$ – плоскость. В трех мерном пространстве все описывалось тремя координатами и мы получаем $\mathbb{R}^3$. По аналогии получается пространство $\mathbb{R}^n$ – n-мерное векторное пространство. Оно удобно тем, что с ним понятно как работать. Тут возникают такие вещи как: системы линейных уравнений, матрицы, определители, ранги матриц и прочие технические вещи. Эта часть линейной алгебры полезна, когда надо что-то посчитать руками и по-другому никак. Однако, основной недостаток в том, что на этом языке не все удобно делать и не все связи между объектами можно легко увидеть. Именно поэтому математики придумали абстрактный подход, а не для того, чтобы замучить людей до смерти.

Абстрактные векторные пространства Утиный принцип из программирования на самом деле был заимствован из математики, как и интерфейсы. Для того, чтобы было проще работать со сложными объектами, мы их скрываем за удобными лаконичными интерфейсами. У векторных пространств тоже есть свой интерфейс, он называется определением. Грубо говоря, к векторным пространствам относится все что угодно, у чего есть элементы при этом эти элементы можно складывать и умножать на числа.[1] На этом языке в линейной алгебре возникают не только векторные пространство, но и линейные отображения, операторы, билинейные формы, скалярные произведения, собственные значения и собственные векторы и куча других любопытных вещей. Понимание этих геометрических слов, помогает искать правильные подходы к решению задач. Однако, для начала я бы хотел поговорить о простых вещах. Потому начнем с систем линейных уравнений и матриц. Уже на эти темы полно серьезных задач.

Системы линейных уравнений

Наша задачи научиться решать Системы Линейных Уравнений (СЛУ), то есть находить все их решения или доказывать, что решений нет. Общий вид СЛУ и ее однородная версия (ОСЛУ):

$$\begin{cases} a_{11}x_1 + \ldots + a_{1n}x_n = b_1 \\ \ldots \\ a_{m1}x_1 + \ldots + a_{mn}x_n = b_m \end{cases} \qquad \begin{cases} a_{11}x_1 + \ldots + a_{1n}x_n = 0 \\ \ldots \\ a_{m1}x_1 + \ldots + a_{mn}x_n = 0 \end{cases}$$

[1] Еще требуются некоторые свойства, но это уже детали контракта интерфейса.

Коэффициенты

Где живут коэффициенты a_{ij} и b_j? Варианты:

- Вещественные числа $\mathbb{R}$
- Комплексные числа $\mathbb{C}$
- Рациональные числа $\mathbb{Q}$

Для решения СЛУ **НЕ** имеет значения откуда берутся коэффициенты, так как решения будут лежать там же. Потому мы будем работать с числами из $\mathbb{R}$.

Матрицы связанные со СЛУ

Для каждой СЛУ введем следующие обозначения:

$$A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix} \quad b = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix} \quad x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} \quad (A|b) = \left(\begin{array}{ccc|c} a_{11} & \dots & a_{1n} & b_1 \\ \vdots & \ddots & \vdots & \vdots \\ a_{m1} & \dots & a_{mn} & b_m \end{array}\right)$$

Названия:

- A – матрица системы
- b – вектор правой части
- $(A|b)$ – расширенная матрица системы
- x – вектор решений

Будем кратко записывать СЛУ и ее однородную версию так: $Ax = b$ и $Ax = 0$.

Количество решений

Случай одного уравнения и одной неизвестной

- $x = 0$ – одно решение
- $0x = 0$ – бесконечное число решений
- $0x = 1$ – нет решений

Элементарные преобразования

$$\text{I тип: } \left(\begin{array}{ccc|c} a_{11} & \dots & a_{1n} & b_1 \\ a_{i1} & \dots & a_{in} & b_i \\ a_{j1} & \dots & a_{jn} & b_j \\ a_{m1} & \dots & a_{mn} & b_m \end{array}\right) \mapsto \left(\begin{array}{ccc|c} a_{11} & \dots & a_{1n} & b_1 \\ a_{i1} & \dots & a_{in} & b_i \\ a_{j1} + \lambda a_{i1} & \dots & a_{jn} + \lambda a_{in} & b_j + \lambda b_i \\ a_{m1} & \dots & a_{mn} & b_m \end{array}\right) \quad i \neq j$$

$$\text{II тип: } \left(\begin{array}{ccc|c} a_{11} & \dots & a_{1n} & b_1 \\ a_{i1} & \dots & a_{in} & b_i \\ a_{j1} & \dots & a_{jn} & b_j \\ a_{m1} & \dots & a_{mn} & b_m \end{array}\right) \mapsto \left(\begin{array}{ccc|c} a_{11} & \dots & a_{1n} & b_1 \\ a_{j1} & \dots & a_{jn} & b_j \\ a_{i1} & \dots & a_{in} & b_i \\ a_{m1} & \dots & a_{mn} & b_m \end{array}\right)$$

$$\text{III тип: } \left(\begin{array}{ccc|c} a_{11} & \dots & a_{1n} & b_1 \\ a_{i1} & \dots & a_{in} & b_i \\ a_{m1} & \dots & a_{mn} & b_m \end{array}\right) \mapsto \left(\begin{array}{ccc|c} a_{11} & \dots & a_{1n} & b_1 \\ \lambda a_{i1} & \dots & \lambda a_{in} & \lambda b_i \\ a_{m1} & \dots & a_{mn} & b_m \end{array}\right) \quad \lambda \neq 0$$

Приведение к ступенчатому виду (алгоритм Гаусса)

Основной способ решения СЛУ – привести ее элементарными преобразованиями к простому виду, где множество решений очевидно.[2]

Разберем типичный ход алгоритма Гаусса на примере 3 уравнений и 4 неизвестных.[3]

Прямой ход алгоритма Гаусса

$$\left(\begin{array}{cccc|c} a_{11} & a_{12} & a_{13} & a_{14} & b_1 \\ a_{21} & a_{22} & a_{23} & a_{24} & b_2 \\ a_{31} & a_{32} & a_{33} & a_{34} & b_3 \end{array}\right) \quad \text{2-я строка } - \frac{a_{21}}{a_{11}} \cdot \text{1-я строка}$$

$$\left(\begin{array}{cccc|c} a_{11} & a_{12} & a_{13} & a_{14} & b_1 \\ 0 & a_{22} & a_{23} & a_{24} & b_2 \\ a_{31} & a_{32} & a_{33} & a_{34} & b_3 \end{array}\right) \quad \text{3-я строка } - \frac{a_{31}}{a_{11}} \cdot \text{1-я строка}$$

$$\left(\begin{array}{cccc|c} a_{11} & a_{12} & a_{13} & a_{14} & b_1 \\ 0 & a_{22} & a_{23} & a_{24} & b_2 \\ 0 & a_{32} & a_{33} & a_{34} & b_3 \end{array}\right) \quad \text{3-я строка } - \frac{a_{32}}{a_{22}} \cdot \text{2-я строка}$$

$$\left(\begin{array}{cccc|c} a_{11} & a_{12} & a_{13} & a_{14} & b_1 \\ 0 & a_{22} & a_{23} & a_{24} & b_2 \\ 0 & 0 & a_{33} & a_{34} & b_3 \end{array}\right)$$

В результате данного хода какие-то коэффициенты, например a_{33}, могли занулиться, потому возможны следующие принципиально другие случаи[4]

$$\left(\begin{array}{cccc|c} \underline{a_{11}} & a_{12} & a_{13} & a_{14} & b_1 \\ 0 & \underline{a_{22}} & a_{23} & a_{24} & b_2 \\ 0 & 0 & 0 & \underline{a_{34}} & b_3 \end{array}\right) \quad \left(\begin{array}{cccc|c} \underline{a_{11}} & a_{12} & a_{13} & a_{14} & b_1 \\ 0 & 0 & \underline{a_{23}} & a_{24} & b_2 \\ 0 & 0 & 0 & \underline{a_{34}} & b_3 \end{array}\right) \quad \left(\begin{array}{cccc|c} \underline{a_{11}} & a_{12} & a_{13} & a_{14} & b_1 \\ 0 & \underline{a_{22}} & a_{23} & a_{24} & b_2 \\ 0 & 0 & 0 & 0 & \underline{b_3} \end{array}\right) \quad \left(\begin{array}{cccc|c} \underline{a_{11}} & a_{12} & a_{13} & a_{14} & b_1 \\ 0 & \underline{a_{22}} & a_{23} & a_{24} & b_2 \\ 0 & 0 & 0 & 0 & 0 \end{array}\right)$$

Подчеркнутые элементы считаются не равными нулю. В ступенчатом виде все переменные (и соответственно коэффициенты перед ними) делятся на главные и неглавные. Главные коэффициенты – это первые ненулевые коэффициенты в строке (подчеркнутые). Переменные при них называются главными, остальные ненулевые коэффициенты и переменные – неглавные или свободными.

Обратный ход алгоритма Гаусса

Разберем типичный обратный ход алгоритма Гаусса. Подчеркнутые элементы считаются не равными нулю.

$$\left(\begin{array}{cccc|c} \underline{a_{11}} & a_{12} & a_{13} & a_{14} & b_1 \\ 0 & \underline{a_{22}} & a_{23} & a_{24} & b_2 \\ 0 & 0 & \underline{a_{33}} & a_{34} & b_3 \end{array}\right) \quad \text{разделить } i\text{-ю строку на } a_{ii}$$

$$\left(\begin{array}{cccc|c} 1 & a_{12} & a_{13} & a_{14} & b_1 \\ 0 & 1 & a_{23} & a_{24} & b_2 \\ 0 & 0 & 1 & a_{34} & b_3 \end{array}\right) \quad \text{2-я строка } - a_{23} \cdot \text{3-я строка}$$

$$\left(\begin{array}{cccc|c} 1 & a_{12} & a_{13} & a_{14} & b_1 \\ 0 & 1 & 0 & a_{24} & b_2 \\ 0 & 0 & 1 & a_{34} & b_3 \end{array}\right) \quad \text{1-я строка } - a_{13} \cdot \text{3-я строка}$$

$$\left(\begin{array}{cccc|c} 1 & a_{12} & 0 & a_{14} & b_1 \\ 0 & 1 & 0 & a_{24} & b_2 \\ 0 & 0 & 1 & a_{34} & b_3 \end{array}\right) \quad \text{1-я строка } - a_{12} \cdot \text{2-я строка}$$

$$\left(\begin{array}{cccc|c} 1 & 0 & 0 & a_{14} & b_1 \\ 0 & 1 & 0 & a_{24} & b_2 \\ 0 & 0 & 1 & a_{34} & b_3 \end{array}\right)$$

[2] Данный метод является самым быстрым возможным как для написания программ, так и для ручного вычисления. При вычислениях руками, однако, полезно местами пользоваться «локальными оптимизациями», то есть, если вы видите, что какая-то хитрая комбинация строк сильно упростит вид системы, то сделайте ее.

[3] При переходе от одной матрицы к другой я новым коэффициентам даю старые имена, чтобы не захламлять текст новыми обозначениями.

[4] Это не полный список всех случаев.

В специальных случаях приведенных выше, получим

$$\left(\begin{array}{cccc|c}1 & 0 & a_{13} & 0 & b_1\\ 0 & 1 & a_{23} & 0 & b_2\\ 0 & 0 & 0 & 1 & b_3\end{array}\right)\quad \left(\begin{array}{cccc|c}1 & a_{12} & 0 & 0 & b_1\\ 0 & 0 & 1 & 0 & b_2\\ 0 & 0 & 0 & 1 & b_3\end{array}\right)\quad \left(\begin{array}{cccc|c}1 & 0 & a_{13} & a_{14} & 0\\ 0 & 1 & a_{23} & a_{24} & 0\\ 0 & 0 & 0 & 0 & 1\end{array}\right)\quad \left(\begin{array}{cccc|c}1 & 0 & a_{13} & a_{14} & b_1\\ 0 & 1 & a_{23} & a_{24} & b_2\\ 0 & 0 & 0 & 0 & 0\end{array}\right)$$

Полученный в результате обратного хода вид расширенной матрицы называется улучшенным ступенчатым видом, т.е., это ступенчатый вид, где все коэффициенты при главных неизвестных – единицы, и все коэффициенты над ними равны нулю.

Получение решений

В системе ниже, выберем переменную x_4 как параметр

$$\left(\begin{array}{cccc|c}1 & 0 & 0 & a_{14} & b_1\\ 0 & 1 & 0 & a_{24} & b_2\\ 0 & 0 & 1 & a_{34} & b_3\end{array}\right)$$

Тогда решения имеют вид[5]

$$\begin{pmatrix}x_1\\x_2\\x_3\end{pmatrix} = \begin{pmatrix}b_1\\b_2\\b_3\end{pmatrix} - x_4\begin{pmatrix}a_{14}\\a_{24}\\a_{34}\end{pmatrix}$$

Специальные случаи:

$$\left(\begin{array}{cccc|c}1 & 0 & a_{13} & 0 & b_1\\ 0 & 1 & a_{23} & 0 & b_2\\ 0 & 0 & 0 & 1 & b_3\end{array}\right)\quad \text{Решения:}\quad \begin{pmatrix}x_1\\x_2\\x_4\end{pmatrix} = \begin{pmatrix}b_1\\b_2\\b_3\end{pmatrix} - x_3\begin{pmatrix}a_{13}\\a_{23}\\0\end{pmatrix}$$

$$\left(\begin{array}{cccc|c}1 & a_{12} & 0 & 0 & b_1\\ 0 & 0 & 1 & 0 & b_2\\ 0 & 0 & 0 & 1 & b_3\end{array}\right)\quad \text{Решения:}\quad \begin{pmatrix}x_1\\x_3\\x_4\end{pmatrix} = \begin{pmatrix}b_1\\b_2\\b_3\end{pmatrix} - x_2\begin{pmatrix}a_{12}\\0\\0\end{pmatrix}$$

$$\left(\begin{array}{cccc|c}1 & 0 & a_{13} & a_{14} & 0\\ 0 & 1 & a_{23} & a_{24} & 0\\ 0 & 0 & 0 & 0 & 1\end{array}\right)\quad \text{Решения:}\quad \text{Нет решений, т.к. последнее уравнение } 0 = 1$$

$$\left(\begin{array}{cccc|c}1 & 0 & a_{13} & a_{14} & b_1\\ 0 & 1 & a_{23} & a_{24} & b_2\\ 0 & 0 & 0 & 0 & 0\end{array}\right)\quad \text{Решения:}\quad \begin{pmatrix}x_1\\x_2\end{pmatrix} = \begin{pmatrix}b_1\\b_2\end{pmatrix} - x_3\begin{pmatrix}a_{13}\\a_{23}\end{pmatrix} - x_4\begin{pmatrix}a_{14}\\a_{24}\end{pmatrix}$$

Вычислительная практика

Найти решения СЛУ соответствующих следующим расширенным матрицам:

$$\left(\begin{array}{ccc|c}2 & 1 & 1 & 3\\ 1 & 2 & 1 & 0\\ 1 & 1 & 2 & 0\end{array}\right)\quad \left(\begin{array}{cc|c}9 & 3 & 5\\ 7 & 5 & 3\end{array}\right)\quad \left(\begin{array}{ccc|c}1 & 1 & -1 & a\\ 1 & -1 & 1 & b\\ -1 & 1 & 1 & c\end{array}\right)\quad \left(\begin{array}{ccc|c}1 & 1 & 2 & 0\\ -1 & 0 & -1 & -1\\ 1 & 1 & 2 & \lambda\end{array}\right)\quad \left(\begin{array}{ccc|c}\lambda & 1 & 1 & 1\\ 1 & \lambda & 1 & 1\\ 1 & 1 & \lambda & 1\end{array}\right)\quad \left(\begin{array}{ccc|c}1 & -1 & -2 & 0\\ 0 & -1 & -1 & 0\\ 1 & 2 & 1 & 1\end{array}\right)$$

Замечание. Работая с целочисленными матрицами, старайтесь во время прямого хода алгоритма Гаусса не выходить за рамки целых чисел.

- Используйте элементарные преобразования I типа только с целым параметром.
- Полезно не злоупотреблять умножением на ненулевое целое, умножайте только на ± 1. Иначе придется работать с большими числами.

На этапе обратного хода алгоритма Гаусса избавиться от деления уже не возможно.

[5] Операция умножения матрицы на число покомпонентная (умножаем каждый элемент на число). Сумма и разность двух матриц покомпонентная (складываем или вычитаем числа на одних и тех же позициях).

Матрицы

Матрица – это прямоугольная таблица чисел

$$A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix},\ \text{где } a_{ij} \in \mathbb{R}$$

Множество всех матриц с m строками и n столбцами обозначается $\mathrm{M}_{m\,n}(\mathbb{R})$. Множество квадратных матриц размера n будем обозначать $\mathrm{M}_n(\mathbb{R})$. Матрицы с одним столбцом или одной строкой называются векторами (вектор-столбцами и вектор-строками соответственно). Множество всех векторов с n координатами обозначается через $\mathbb{R}^n$. Мы по умолчанию считаем, что наши вектора – вектор-столбцы.[6]

Операции над матрицами

Сложение Пусть $A, B \in \mathrm{M}_{m\,n}(\mathbb{R})$. Тогда сумма $A + B$ определяется покомпонентно, т.е. $C = A + B$, то $c_{ij} = a_{ij} + b_{ij}$ или

$$\begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix} + \begin{pmatrix} b_{11} & \dots & b_{1n} \\ \vdots & \ddots & \vdots \\ b_{m1} & \dots & b_{mn} \end{pmatrix} = \begin{pmatrix} a_{11}+b_{11} & \dots & a_{1n}+b_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1}+b_{m1} & \dots & a_{mn}+b_{mn} \end{pmatrix}$$

Складывать можно только матрицы одинакового размера.

Умножение на скаляр Если $\lambda \in \mathbb{R}$ и $A \in \mathrm{M}_{m\,n}(\mathbb{R})$, то λA определяется так: $\lambda A = C$, где $c_{ij} = \lambda a_{ij}$ или

$$\lambda \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix} = \begin{pmatrix} \lambda a_{11} & \dots & \lambda a_{1n} \\ \vdots & \ddots & \vdots \\ \lambda a_{m1} & \dots & \lambda a_{mn} \end{pmatrix}$$

Умножение матриц Пусть $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ и $B \in \mathrm{M}_{n\,k}(\mathbb{R})$, то произведение $AB \in \mathrm{M}_{m\,k}(\mathbb{R})$ определяется так: $AB = C$, где $c_{ij} = \sum_{t=1}^{n} a_{it} b_{tj}$ или

$$\begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix} \begin{pmatrix} b_{11} & \dots & b_{1k} \\ \vdots & \ddots & \vdots \\ b_{n1} & \dots & b_{nk} \end{pmatrix} = \begin{pmatrix} \sum_{t=1}^{n} a_{1t} b_{t1} & \dots & \sum_{t=1}^{n} a_{1t} b_{tk} \\ \vdots & \ddots & \vdots \\ \sum_{t=1}^{n} a_{mt} b_{t1} & \dots & \sum_{t=1}^{n} a_{mt} b_{tk} \end{pmatrix}$$

На умножение матриц можно смотреть следующим образом. Чтобы получить коэффициент c_{ij} надо, из матрицы A взять i-ю строку (она имеет длину n), а из матрицы B взять j-ый столбец (он тоже имеет длину n). Тогда их надо скалярно перемножить и результат подставить в c_{ij}.

Замечание Пусть у нас задана система линейных уравнений с матрицей $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ и правой частью $b \in \mathbb{R}^m$. Пусть $x = (x_1, \dots, x_n)$ – будет столбцом из переменных. Тогда систему $(A|b)$ можно задать в матричном виде $Ax = b$.

Свойства операций

Все три операции на матрицах обладают «естественными свойствами» и согласованы друг с другом. Вот перечень базовых свойств операций над матрицами:[7]

1. **Ассоциативность сложения** $(A + B) + C = A + (B + C)$ для любых $A, B, C \in \mathrm{M}_{m\,n}(\mathbb{R})$

[6] Важно, directX и openGL используют вектор-строки! Потому часть инженерной литературы на английском связанной с трехмерной графикой оперирует со строками.

[7] Все эти свойства объединяет то, что они являются аксиомами в различных определениях для алгебраических структур. Позже мы столкнемся с такими структурами.

2. **Существование нейтрального элемента для сложения** Существует единственная матрица $0 \in \mathrm{M}_{m\,n}(\mathbb{R})$ обладающая следующим свойством $A + 0 = 0 + A = A$ для всех $A \in \mathrm{M}_{m\,n}(\mathbb{R})$. Такая матрица целиком заполнена нулями.

3. **Коммутативность сложения** $A + B = B + A$ для любых $A, B \in \mathrm{M}_{m\,n}(\mathbb{R})$.

4. **Наличие обратного по сложению** Для любой матрицы $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ существует матрица $-A$ такая, что $A + (-A) = (-A) + A = 0$. Такая матрица единственная и состоит из элементов $-a_{ij}$.

5. **Ассоциативность умножения** Для любых матриц $A \in \mathrm{M}_{m\,n}(\mathbb{R})$, $B \in \mathrm{M}_{n\,k}(\mathbb{R})$ и $C \in \mathrm{M}_{k\,t}(\mathbb{R})$ верно $(AB)C = A(BC)$.

6. **Существование нейтрального элемента для умножения** Для каждого k существует единственная матрица $E \in \mathrm{M}_k(\mathbb{R})$ такая, что для любой $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ верно $EA = AE = A$. У такой матрицы $E_{ii} = 1$, а $E_{ij} = 0$. Когда нет путаницы матрицу E обозначают через 1.

7. **Дистрибутивность умножения относительно сложения** Для любых матриц $A, B \in \mathrm{M}_{m\,n}(\mathbb{R})$ и $C \in \mathrm{M}_{n\,k}(\mathbb{R})$ верно $(A + B)C = AC + BC$. Аналогично, для любых $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ и $B, C \in \mathrm{M}_{n\,k}(\mathbb{R})$ верно $A(B + C) = AB + AC$.

8. **Умножение на числа ассоциативно** Для любых $\lambda, \mu \in \mathbb{R}$ и любой матрицы $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ верно $\lambda(\mu A) = (\lambda\mu)A$. Аналогично для любого $\lambda \in \mathbb{R}$ и любых $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ и $B \in \mathrm{M}_{n\,k}(\mathbb{R})$ верно $\lambda(AB) = (\lambda A)B$.

9. **Умножение на числа дистрибутивно относительно сложения матриц и сложения чисел** Для любых $\lambda, \mu \in \mathbb{R}$ и $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ верно $(\lambda+\mu)A = \lambda A+\mu A$. Аналогично, для любого $\lambda \in \mathbb{R}$ и $A, B \in \mathrm{M}_{m\,n}(\mathbb{R})$ верно $\lambda(A + B) = \lambda A + \lambda B$.

10. **Умножение на скаляр нетривиально** Если $1 \in \mathbb{R}$, то для любой матрицы $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ верно $1A = A$.

11. **Умножение на скаляр согласовано с умножением матриц** Для любого $\lambda \in \mathbb{R}$ и любых $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ и $B \in \mathrm{M}_{n\,k}(\mathbb{R})$ верно $\lambda(AB) = (\lambda A)B = A(\lambda B)$.

К этим свойствам надо относиться так. Доказывая что-то про матрицы, можно лезть внутрь определений операций над ними, а можно пользоваться свойствами операций. Так вот, список выше – это минимальный набор свойств операций, из которых можно вытащить базовую информацию про эти операции и при этом не лезть внутрь определений.

Нехорошие свойства операций

Матричные операции обладают несколькими аномалиями по сравнению со свойствами операций над обычными числами.

1. Существование вычитания следует из «хорошести» операции сложения. Она позволяет определить вычитание без проблем. Однако, операция умножения уже хуже, чем на обычных числах, потому не получится определить на матрицах операцию деления.

2. Умножение матриц НЕ коммутативно. Действительно
$$\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \quad \text{но} \quad \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

3. В матрицах есть ненулевые «делители нуля», т.е. существуют две ненулевые матрицы A и B такие, что $AB = 0$.[8] Пример:
$$\begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix} = 0$$

4. В матрицах есть ненулевые «нильпотенты», то есть можно найти такую ненулевую матрицу A, что $A^n = 0$.[9] Пример,
$$\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = 0$$

[8] По определению матрица $A \in \mathrm{M}_n(\mathbb{R})$ называется делителем нуля, если для некоторой ненулевой матрицы $B \in \mathrm{M}_n(\mathbb{R})$ выполнено $AB = 0$ или $BA = 0$.

[9] По определению матрица $A \in \mathrm{M}_n(\mathbb{R})$ называется нильпотентной или нильпотентом, если для некоторого натурального $n > 0$ верно $A^n = 0$.

Блочное умножение матриц

Пусть даны две матрицы, которые разбиты на блоки как показано ниже:

$$\begin{array}{c|cc} & k & s \\ \hline m & \multicolumn{2}{c}{\multirow{2}{*}{$\begin{pmatrix} A & B \\ C & D \end{pmatrix}$}} \\ n & & \end{array} \qquad \begin{array}{c} \\ m \\ n \end{array}\begin{array}{c} k\quad s \\ \begin{pmatrix} A & B \\ C & D \end{pmatrix} \end{array} \qquad \begin{array}{c} \\ k \\ s \end{array}\begin{array}{c} u\quad v \\ \begin{pmatrix} X & Y \\ W & Z \end{pmatrix} \end{array}$$

Числа m, n, k, s, u, v – размеры соответствующих блоков. Наша цель понять, что эти матрицы можно перемножать блочно. А именно, увидеть, что результат умножения этих матриц имеет вид

$$\begin{array}{c} \\ m \\ n \end{array}\begin{array}{c} \qquad u \qquad\qquad\quad v \\ \begin{pmatrix} AX + BW & AY + BZ \\ CX + DW & CY + DZ \end{pmatrix} \end{array}$$

Делается это таким трюком. В начале заметим, что

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix} = \begin{pmatrix} A & 0 \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & B \\ 0 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ C & 0 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & D \end{pmatrix}$$

После чего методом «пристального взгляда» перемножаем матрицы с большим количеством нулей (попробуйте проделать это!).

На этот факт можно смотреть вот как. Матрица – это прямоугольная таблица заполненная числами. А можно составлять прямоугольные таблица заполненные другими объектами, например матрицами. Тогда они складываются и перемножаются так же как и обычные матрицы из чисел. Единственное надо учесть, что в блочном умножении есть разница между $AX + BW$ и $XA + BW$, так как A, B, X и W не числа, а матрицы, то их нельзя переставлять местами, порядок теперь важен.

Вот полезный пример. Пусть дана матрица из $\mathrm{M}_{n+1}(\mathbb{R})$ вида

$$\begin{pmatrix} A & v \\ 0 & \lambda \end{pmatrix}, \quad \text{где} \quad A \in \mathrm{M}_n(\mathbb{R}), \quad v \in \mathbb{R}^n, \quad \lambda \in \mathbb{R}$$

Тогда

$$\begin{pmatrix} A & v \\ 0 & \lambda \end{pmatrix}\begin{pmatrix} A & v \\ 0 & \lambda \end{pmatrix} = \begin{pmatrix} A^2 & Av + v\lambda \\ 0 & \lambda^2 \end{pmatrix} = \begin{pmatrix} A^2 & Av + \lambda v \\ 0 & \lambda^2 \end{pmatrix} = \begin{pmatrix} A^2 & (A + \lambda E)v \\ 0 & \lambda^2 \end{pmatrix}$$

Предпоследнее равенство верно, так как не важно с какой стороны умножать v на скаляр λ.

Транспонирование

Пусть A – матрица вида

$$\begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \dots & a_{mn} \end{pmatrix} \quad \text{или} \quad \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{pmatrix} \quad \text{или} \quad \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

Определим транспонированную матрицу $A^t = (a'_{ij})$ так: $a'_{ij} = a_{ji}$. Наглядно, транспонированная матрица для приведенных выше

$$\begin{pmatrix} a_{11} & \dots & a_{m1} \\ \vdots & \ddots & \vdots \\ a_{1n} & \dots & a_{mn} \end{pmatrix} \quad \text{или} \quad \begin{pmatrix} a_{11} & a_{21} \\ a_{12} & a_{22} \\ a_{13} & a_{23} \end{pmatrix} \quad \text{или} \quad \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix}$$

Задача. Пусть $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ и $B \in \mathrm{M}_{n\,k}(\mathbb{R})$. Покажите, что

1. $(AB)^t = B^t A^t$.

2. Если A – блочная матрица следующего вида

$$A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad \text{тогда} \quad A^t = \begin{pmatrix} A_{11}^t & A_{21}^t \\ A_{12}^t & A_{22}^t \end{pmatrix}$$

Вот еще один полезный пример блочного умножения. Пусть $x_1, \ldots, x_m \in \mathbb{R}^n$ и $y_1, \ldots, y_m \in \mathbb{R}^n$ – столбцы. Составим из этих столбцов матрицы $X = (x_1|\ldots|x_m)$ и $Y = (y_1|\ldots|y_m)$.[10] Заметим, что $X, Y \in \mathrm{M}_{n\,m}(\mathbb{R})$. Тогда

$$XY^t = (x_1|\ldots|x_m)(y_1|\ldots|y_m)^t = \sum_{i=1}^{m} x_i y_i^t$$

Умножение на специальные виды матриц

Пусть $A, B \in \mathrm{M}_n(\mathbb{R})$ имеют следующий вид

$$A = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1\,n-1} & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2\,n-1} & a_{2n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n-1\,1} & a_{n-1\,2} & \ldots & a_{n-1\,n-1} & a_{n-1\,n} \\ a_{n1} & a_{n2} & \ldots & a_{n\,n-1} & a_{nn} \end{pmatrix} \quad B = \begin{pmatrix} 0 & 1 & & & \\ & 0 & 1 & & \\ & & 0 & \ddots & \\ & & & \ddots & 1 \\ & & & & 0 \end{pmatrix}$$

где все незаполненные клетки считаются равными нулю. Тогда прямая проверка показывает, что

$$AB = \begin{pmatrix} 0 & a_{11} & \ldots & a_{1\,n-2} & a_{1\,n-1} \\ 0 & a_{21} & \ldots & a_{2\,n-2} & a_{2\,n-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & a_{n-1\,1} & \ldots & a_{n-1\,n-2} & a_{n-1\,n-1} \\ 0 & a_{n1} & \ldots & a_{n\,n-2} & a_{n\,n-1} \end{pmatrix} \quad \text{и} \quad BA = \begin{pmatrix} a_{21} & a_{22} & \ldots & a_{2\,n-1} & a_{2n} \\ a_{31} & a_{32} & \ldots & a_{3\,n-1} & a_{3n} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n1} & a_{n2} & \ldots & a_{n\,n-1} & a_{nn} \\ 0 & 0 & \ldots & 0 & 0 \end{pmatrix}$$

То есть умножение на B справа сдвигает все столбцы матрицы A вправо, а умножение на B слева сдвигает все строки матрицы A вверх. Если мы хотим переставлять столбцы и строки по циклу, то надо умножать на матрицу B следующего вида

$$B = \begin{pmatrix} 0 & 1 & & & \\ & 0 & 1 & & \\ & & 0 & \ddots & \\ & & & \ddots & 1 \\ 1 & & & & 0 \end{pmatrix}$$

то есть добавить единичку слева снизу к предыдущей матрице. Проверьте, что умножение на нее действительно сдвигает столбцы и строки по циклу.

Квадратные матрицы

След Для любой матрицы $A \in \mathrm{M}_n(\mathbb{R})$ определим ее след следующим образом $\operatorname{tr}(A) = a_{11} + \ldots + a_{nn}$, то есть это сумма диагональных элементов. Пока совсем не понятно зачем нам нужна такая странная величина. Давайте начнем с простых свойств следа

1. Для любых матриц $A, B \in \mathrm{M}_n(\mathbb{R})$ верно

$$\operatorname{tr}(A+B) = \operatorname{tr}(A) + \operatorname{tr}(B)$$

2. Для любой матрицы $A \in \mathrm{M}_n(\mathbb{R})$ и числа $\lambda \in \mathbb{R}$, верно

$$\operatorname{tr}(\lambda A) = \lambda \operatorname{tr}(A)$$

3. Для любых матриц $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ и $B \in \mathrm{M}_{n\,m}(\mathbb{R})$ выполнено[11]

$$\operatorname{tr}(AB) = \operatorname{tr}(BA)$$

[10] Данная запись означает, что мы берем столбцы x_i и записываем их подряд в одну большую таблицу.

[11] Обратите внимание, что слева считается след матрицы размером m на m, а справа считается след матрицы размером n на n.

Деление Заметим, что множество квадратных матриц $\mathrm{M}_n(\mathbb{R})$ замкнуто относительно сложения, умножения, содержит 0 и 1 (в смысле содержит единичную матрицу E, которая является аналогом числа 1 для чисел). На это множество можно смотреть как на обобщение обычных чисел. Однако, надо не забывать, что произведение ненулевых матриц может стать нулем или даже степень ненулевой матрицы может оказаться нулем.

Что значит деление в числах? Предположим, что у нас есть два числа $a, b \in \mathbb{R}$. Тогда деление $a/b = a \cdot b^{-1}$ – это просто умножение на обратный элемент, а обратный элемент b^{-1} определяется свойством $bb^{-1} = 1$. Данное наблюдение дает ключ к распространению деления и обращения на случай матриц. Обратите внимание, что из-за некоммутативности умножения матриц, нам бы пришлось вводить две операции деления: деление слева и справа. Кроме того, пришлось бы разбираться, как эти операции согласованы друг с другом. Вместо этого мы все сводим к унарной операции взятия обратной матрицы, а деление превращается в обычное умножение. Это сильно упрощает техническую часть.

Односторонняя обратимость Пусть $A \in \mathrm{M}_{m\,n}(\mathbb{R})$ и $B \in \mathrm{M}_{n\,m}(\mathbb{R})$ – некоторые матрицы. Говорят, что B является *левым обратным* к A, если выполняется равенство $BA = E$, где E – единичная матрица размера n. Аналогично, говорят, что B является *правым обратным* к A, если выполняется равенство $AB = E$, где E – единичная матрица размера m.

Вообще говоря, существование левого обратного не влечет существование правого и наоборот. Более того, односторонних обратных может быть много. Например, рассмотрим матрицу $A \in \mathrm{M}_{1\,2}(\mathbb{R})$ следующего вида $A = (1, 0)$. Тогда A^t является правой обратной к A. У A нет левого обратного, так как матрица BA всегда имеет нулевой второй столбец. Кроме того, любая матрица вида $(1, a)^t$ является правой обратной к A.

Обратимость Пусть теперь $A \in \mathrm{M}_n(\mathbb{R})$ – некоторая матрица. Говорят, что A *обратима*, если она обладает левым и правым обратным. Пусть $B_1 \in \mathrm{M}_n(\mathbb{R})$ является левым обратным к A, а $B_2 \in \mathrm{M}_n(\mathbb{R})$ – правым обратным к A. Тогда $B_1 A = E$ и $AB_2 = E$. Рассмотрим выражение $B_1 A B_2$. Тогда по ассоциативности матричного умножения выполнено следующее

$$B_1 = B_1 E = B_1(AB_2) = (B_1 A)B_2 = EB_2 = B_2$$

Более того, если B_1 и B_1' – два левых обратных, а B_2 – какой-то правый обратный, то из выше написанного следует, что $B_1 = B_2$ и в тоже время $B_1' = B_2$. В частности, все левые обратные между собой совпадают и аналогично все правые обратные между собой совпадают. А значит, существует единственная матрица, которую мы обозначим A^{-1} такая, что $AA^{-1} = A^{-1}A = E$. Такая матрица называется *обратной* к A.

Почему нет обратимых прямоугольных матриц Заметим, что определение обратимой матрицы и обратной к ней мы могли бы дать в случае не только квадратной матрицы, а именно: пусть $A \in \mathrm{M}_{m\,n}(\mathbb{R})$, тогда $B \in \mathrm{M}_{n\,m}(\mathbb{R})$ назовем обратной к A, если $AB = E \in \mathrm{M}_m(\mathbb{R})$ и $BA = E \in \mathrm{M}_n(\mathbb{R})$. Оказывается, что таких прямоугольных неквадратных матриц просто не существует. То есть двусторонний обратный бывает только у квадратных матриц. Доказывается это так. Если верны равенства $AB = E$ и $BA = E$ в предположениях выше, то мы можем посчитать след от обоих равенств. Тогда из первого равенства следует, что $\operatorname{tr}(AB) = m$, а из второго, что $\operatorname{tr}(BA) = n$. Но мы уже знаем, что $\operatorname{tr}(AB) = \operatorname{tr}(BA)$. А отсюда следует, что $m = n$.

Смысл обратной матрицы в том, что если она существует для матрицы $A \in \mathrm{M}_n(\mathbb{R})$, то на матрицу A можно сокращать в выражениях. Например, если $AB = AC$ и матрица A обратима, то умножая на A^{-1} слева, мы получим $A^{-1}AB = A^{-1}AC$, а следовательно $B = C$. То есть умножение на A^{-1} слева сокращает A слева.

Существование обратного Пусть $A, B \in \mathrm{M}_n(\mathbb{R})$ – две ненулевые матрицы (то есть не состоящие целиком из нулей). Если $AB = 0$, то для матриц A и B не существует обратной. Действительно, предположим, что A обратима, тогда равенство $AB = 0$ можно умножить на A^{-1} слева, получим $A^{-1}AB = A^{-1}0$. Но это равносильно $B = 0$, что противоречит нашему предположению о том, что B не ноль. Аналогично, не существует обратной матрицы, если $A^n = 0$ для некоторого $n > 0$.

Пусть как и выше $A \in \mathrm{M}_n(\mathbb{R})$ – некоторая квадратная матрица. Рассмотрим систему линейных уравнений вида $Ax = 0$. Если бы любая ненулевая квадратная матрица A была обратима, то уравнение $Ax = 0$ имело бы одно единственное решение $x = 0$, для этого надо просто сократить на A слева. Однако, если взять матрицу $A = \left(\begin{smallmatrix} 1 & 1 \\ 1 & 1 \end{smallmatrix}\right)$, то очевидно, что у системы $Ax = 0$ есть ненулевые решения, все решения имеют вид $x_1 = x_2 = t \in \mathbb{R}$. Потому необратимость матрицы связана с наличием ненулевых решений у однородной системы.

Утверждение. *Пусть $A \in \mathrm{M}_n(\mathbb{R})$ – произвольная квадратная матрица. Тогда следующие условия эквивалентны:*

1. *Система линейных уравнений $Ax = 0$ имеет только нулевое решение $x \in \mathbb{R}^n$.*
2. *Система линейных уравнений $A^t y = 0$ имеет только нулевое решение $y \in \mathbb{R}^n$.*
3. ...[12]
4. *Матрица A обратима.*
5. *Матрица A обратима слева, т.е. существует $L \in \mathrm{M}_n(\mathbb{R})$ такая, что $LA = E$.*
6. *Матрица A обратима справа, т.е. существует $R \in \mathrm{M}_n(\mathbb{R})$ такая, что $AR = E$.*

[12] Этот пункт я уточню в следующий раз, а сейчас не хочу сбивать нумерацию.