Решение задач линейной алгебры в системе Matlab.

Д. К. Разманова

Руководитель В. П. Поплёвко

MATLAB – это высокоуровневый язык технических расчетов, интерактивная среда разработки алгоритмов и современный инструмент анализа данных. Основной особенностью данного языка является его широкие возможности по работе с матрицами. В то время как другие языки программирования обычно работают с числами по одному, он работает с целыми матрицами и массивами, т.к. самая основная структура данных в MATLAB – это матрица. Ее элементы могут быть числами, логическими значениями, строками или другим типом данных. Чтобы ввести матрицу необходимо использовать квадратные скобки, пробелы или запятые, точку с запятой, разделяющую строки. Например, B = [12 62; 93 -8] – квадратная матрица (2×2). MATLAB имеет много функций, которые помогают создавать матрицы с определенными значениями или конкретной структурой. Например, A = zeros (3,2), т.е. A – это матрица (3×2), заполненная нулями. Квадратные скобки используются для соединения существующих матриц вместе. Этот способ создания матрицы называется конкатенацией. Чтобы конкатенировать две матрицы, у них должны быть совместимые размеры, т.е. при конкатенации горизонтально – одинаковое число строк, вертикально – одинаковое число столбцов. Также можно добавить один или несколько элементов в матрицу путем размещения их за пределами существующих контуров индекса строки и столбца или расширить размер путем вставки новой матрицы за пределами существующих областей значений индекса.

Матричные операции в Matlab: 1)Добавление и вычитание матриц. Выполняются поэлементно и требуют, чтобы обе матрицы имели совместимые размерности. 2)Векторные произведения и транспонирование. Функции transpose(транспонирование), ctranspose(комплексное сопряженное транспонирование). Вектор – строка и вектор – столбец той же длины могут быть умножены в любом порядке. Результатом является или скалярное произведение или векторное. 3)Умножение матриц. Функция mtimes. Прямоугольные умножения матриц должны удовлетворить условия совместимости размерности. 4) Обращение матриц. Если матрица A квадратная, ее определитель ненулевой, и уравнения A*X = I и X*A =I имеют одно решение X. Это решение называется инверсией A и обозначается A^-1(функция inv). 5)Также часто используются опрецации: mpower(матричная степень), sqrtm(матричный квадратный корень), logm(матричный логарифм) и т.д.

Определителем вычисляется мера масштабного коэффициента линейного преобразования, описанного матрицей. Когда определитель нулевой, матрица необратима, т.е. ее инверсии не существует. Некоторые матрицы почти обратимы, и несмотря на то, что обратная матрица существует, вычисление восприимчиво к числовым ошибкам. Функция cond вычисляет число обусловленности для инверсии, а именно, точность результатов матричной инверсии.

СЛАУ в Matlab. В системах линейных уравнений используется терминология деления. Два символа деления, наклонная черта вправо (mrdivide) и наклонная черта влево (mldivide). Эти операторы используются в двух ситуациях: 1) x = b/A – решение матричного уравнения x*A = b. 2) x = A\b – решение матричного уравнения A*x = b.

Методы решения систем линейных уравнений. Кроме стандартного решения систем x = A \ b, существуют: 1)Метод обратной матрицы. Его суть состоит в том, что сначала необходимо выписать коэффициенты при a, b и c (то есть те коэффициенты, которые находятся слева) в одну матрицу, а свободный член в другую. В итоге у нас получится 2 матрицы: A = [4 1 -1; 1 -1 1; 2 -3 -3];   B=[6; 4; 4]; Для реализации этого метода (и следующих методов тоже) требуется чтобы определитель матрицы, составленной из коэффициентов левой части не был равен нулю. После проверки этого условия находим обратную матрицу с помощью оператора inv. Данное решение в Matlab находится как перемножение найденной обратной матрицы на матрицу свободных членов. 2)Метод Гаусса. Сформируем расширенную систему.Например, A=[123;1-32;111]; b=[7;5;3]; C = [A b]; Приведём её к ступенчатому виду, выполнив прямой и обратный ход метода Гаусса D = rref(C), где последний столбец матрицы есть решение x = D(:,4); Проверяем его A*x – b. 3)Правило Крамера. Проверим невырожденность системы (rank(A)). По правилу Крамера: A1 = A; и т.д. A1(:,1) = b; и т.д. x1 = det(A1) / det(A); и т.д. Получаем x=[x1;x2;x3;x4]; и проверяем решение с помощью A*x – b. Также можно решать СЛАУ с помощью обратной матрицы или ее ранга.

Ключевые особенности Matlab: 1)Платформонезависимый высокоуровневый язык программирования, ориентированный на матричные вычисления и разработку алгоритмов. 2)Содержит функции линейной алгебры, статистики, анализа Фурье, дает возможность решения дифференциальных уравнений и т.д. 3)Удобен в использовании и не требует особых навыков в программировании.