Отчет по лабораторной работе № 1

«Освоение интерфейса MATLAB»

по дисциплине

«Теория вероятностей и математическая статистика»

Выполнил студент группы ИС/б-22о

Горбенко К.Н.

Проверил:

Кузнецов С.А.

* 1. Цель работы

Получение общего представления о математическом пакете MATLAB - одного из наиболее популярных представителей семейства систем автоматизации решений научно-технических задач. Изучение особенностей интерфейса, функциональных основных возможностей, формирования навыков практической работы в среде MATLAB, математических вычислений, моделирования, разработки приложений и анализа данных.

* 1. Краткое теоретическое введение
     1. Что такое MATLAB?

MATLAB - это высокопроизводительный язык для технических расчетов, выполняемых на ЭВМ. Он включает в себя вычисления, визуализацию и программирование в удобной среде, где задачи и решения выражаются в форме, близкой к математической.

Система MATLAB (сокращение от MATrix LABoratory - МАТричная ЛАБоратория) разработана фирмой The MathWorks, Inc) (США, г. Нейтик, шт. Массачусетс). Она распространяется боле 10 лет и постоянно совершенствуется. Ее первоначальная версия была написана в начале 80-х годов на языке FORTRAN. Современные версии MATLAB допускают возможность обращения к программам, написанным на языках FORTRAN, C и C++.

Типичное использование MATLAB - это:

1. математические вычисления;
2. создание алгоритмов;
3. моделирование;
4. анализ данных, исследование и визуализация;
5. научная и инженерная графика;
6. разработка приложений, включая графический интерфейс.

MATLAB - это интерактивная система, в которой основным элементом данных является массив. Это позволяет решать различные задачи, связанные с техническими вычислениями, особенно в которых используются матрицы и вектора, в несколько раз быстрее, чем при написании программ с использованием "скалярных" языков программирования, таких как C или PASCAL.

MATLAB развивался в течение ряда лет, ориентируясь на различных пользователей. В университетской среде он представлял собой стандартный инструмент для работы в различных областях математики, машиностроения и науки. Для промышленности - это высокопродуктивный инструмент исследований, разработок и анализа данных.

В MATLAB важная роль отводится специализированным группам программ, называемых toolboxes (" ящики с инструментами " ). В нашей литературе их принято называть пакетами прикладных программ (ППП). Они очень важны для большинства пользователей MATLAB, так как позволяют применять высокоэффективные специализированные методы. Toolboxes это всесторонняя коллекция функций MATLAB (М-файлов), которые позволяют решать частные классы задач. Эти ППП применяются для обработки сигналов, анализа и синтеза систем контроля, нейронных сетей, нечеткой логики, вейвлетов, моделирования и т.д.

* + 1. Система MATLAB

Система MATLAB состоит из пяти основных частей:

**Язык MATLAB**. Это язык матриц и массивов высокого уровня с управлением потоками, функциями, структурами данных, вводом-выводом, обладающий особенностями объектно-ориентированного программирования. Это позволяет программировать как в "небольшом" масштабе для быстрого создания черновых программ, так и в "большом" для создания больших и сложных приложений.

**Среда MATLAB**. Это набор инструментов и приспособлений, с которыми работает пользователь или программист MATLAB. Она включает в себя средства для управления переменными в рабочем пространстве MATLAB, вводом и выводом данных, а также создания, контроля и отладки М-файлов и приложений MATLAB.

**Управляемая графика**. Это - графическая система MATLAB, которая включает в себя команды высокого уровня для визуализации двух- и трехмерных данных, обработки изображений, анимации и иллюстрированной графики. В нее входят также команды низкого уровня, позволяющие полностью редактировать внешний вид графики, так же, как при создании Графического Пользовательского Интерфейса (GUI) для MATLAB - приложений.

**Библиотека математических функций**. Это обширная коллекция эффективных вычислительных алгоритмов, начиная от элементарных функций (сумма, тригонометрические функции, комплексная арифметика) и кончая сложными (обращение матриц, нахождение собственных значений и собственных векторов, функции Бесселя, преобразование Фурье и т.д.)

**Программный интерфейс**. Это библиотека, которая позволяет писать программы на C и FORTAN, которые взаимодействуют с MATLAB. Она включает средства для вызова программ из MATLAB (динамическая связь), вызывая MATLAB как вычислительный инструмент и для чтения-записи MAT-файлов.

Системе MATLAB сопутствует программа **Simulink** - интерактивная система для моделирования нелинейных динамических систем. Она представляет собой среду, управляемую мышью, которая позволяет моделировать процесс путем перетаскивания блоков диаграмм на экране и их манипуляцией. Simulink работает с линейными, нелинейными, непрерывными, дискретными, многомерными системами.

Дополнениями к Simulink являются **Blocksets** ("Наборы блоков"), обеспечивающие библиотеки блоков для специализированных приложений, таких как связь, обработка сигналов, энергетические системы.

Программа **Real-Time Workshop** позволяет генерировать С - код из блоков диаграмм и запускать их на выполнение на различных системах реального времени.

* 1. Перевод справки раздела matlab\ elmat (вариант № 4)

Элементарные матрицы и манипуляции над матрицами.

**Элементарные матрицы:**

zeros - Массив, содержащий нули.

ones - Массив, содержащий единицы.

eye - Единичная матрица.

repmat - Скопировать матрицу с заданием размера.

repelem - Скопировать элементы массива.

linspace - Вектор с линейной разницей.

logspace - Вектор с логарифмической разницей.

freqspace - Разница в частоте для частотного ответа.

meshgrid - Массивы координат X и Y для диаграм.

accumarray - Construct an array with accumulation.

: - Массив с фиксированной разностью между элементами.

**Базовая информация о массивах:**

size - Размер массива.

length - длина вектора.

ndims - Количество размерностей.

numel - Количество элементов.

disp - Показать значение переменной.

isempty - Истина в случае, когда массив пустой.

isequal - Истина в случае, когда массивы численно идентичны.

isequaln - Истина в случае, когда массивы численно идентичны, NaN воспринимаются как числа.

**Манипуляции над матрицами:**

cat - Склеить массивы.

reshape - Переформировать массивы.

diag - Диагональные матрицы и извлечение диагоналей.

blkdiag - Поблочное диагональное склеивание.

tril - Извлечь нижнюю треугольную матрицу.

triu - Извлечь верхнюю треугольную матрицу.

fliplr - Повернуть матрицу влево/вправо.

flipud - Повернуть матрицу вверх/вниз.

flip - Повернуть порядок элементов.

rot90 - Повернуть матрицу на 90 градусов.

: - Массив с фиксированной разностью между элементами.

find - Получить индексы ненулевых элементов.

end - Индекс последнего элемента.

sub2ind - Получить индексы многих позиций.

ind2sub - Получить многие позиции по индексам.

bsxfun - Функция бинарного одиночного расширения.

**Функции над многомерными массивами:**

ndgrid - Сгенерировать массивы для функций N-D и для интерполяции. permute - Переставить размерности массива.

ipermute - Инверсная смена размерностей массива.

shiftdim - Сдвинуть размерности.

circshift - Сдвинуть массив циклически.

squeeze - Удалить единичные размерности.

**Полезные функции для работы с массивами:**

isscalar - Истина для скаляра.

isvector -Истина для вектора.

isrow - Истина для вектора-строки.

iscolumn - Истина для вектора-столбца.

ismatrix - Истина для матрицы.

**Специальные переменные и константы:**

eps - Относительная точность числа с плавающей точкой.

realmax - Наибольшее число с плавающей точкой.

realmin - Наименьшее число с плавающей точкой.

intmax - Наибольшее целое число.

intmin - Наименьшее целое число.

flintmax - Наибольшее целое в форме с плавающей точкой.

pi - 3.1415926535897....

i - Мнимая единица.

inf - Бесконечность.

nan - Не-число.

isnan - Истина для NaN.

isinf -Истина для бесконечных чисел.

isfinite - Истина для конечных чисел.

j - Мнимая единица.

true - Массив с элементами-истинами.

false - Массив с ложными элементами.

**Специальные матрицы:**

compan -Сопутствующая матрица.

gallery - Тестовые матрицы.

hadamard - Матрица Адамара.

hankel -Генкелева матрица.

hilb - Матрица Гильберта.

invhilb - Обращенная матрица Гильберта.

magic - Волшебный квадрат.

pascal - Матрица Паскаля.

rosser - Классическая симметричная тестовая задача на собственные значения.

toeplitz - Матрица Тёплица.

vander - Матрица Вандермонда.

wilkinson - Матрица собственных значений Вилкинсона.

**cat - Склеить массивы.**

cat(DIM, A, B) склеивает два массива в размерности DIM.

cat(2, A, B) – то же, что и [A, B].

cat(1, A, B) – то же, что и [A; B].

B = cat(DIM, A1, A2, A3, A4, ...) склеивает массивы A1, A2, A3, A4, … в размерности DIM.

Можно вызывать, используя синтаксис списков: cat(DIM, C{;}) или cat(DIM, C.FIELD). Таким образом, получаем удобный способ склеить структуру, содержащую матрицы, в одну матрицу.

Примеры:

a = magic(3); b = pascal(3);

c = cat(4, a, b);

Результат – матрица 3 x 3 x 1 x 2.

s = {a b};

for i = 1:length(s),

siz(i) = size(s{i});

end

sizes = cat (1, siz{;})

Результат – массив 2 x 2, содержащий размерности.

**squeeze - Удалить единичные размерности.**

B = squeeze(A) возвращает массив B с теми же элементами, что и в А, но без единичных размерностей. Единичная размерность – та, для которой size(A, dim)==1. Не работает для двумерных массивов.

Пример:

squeeze(rand(2, 1, 3))

Результат – матрица 2 x 3.

**flip - Повернуть порядок элементов.**

Y = flip(X) возвращает вектор Y с такими же размерностями, как и у X, но с обратным порядком элементов в них. Если X – матрица, элементы переворачиваются вертикально (по столбцам).

flip(X, DIM) работает только для размерности DIM.

Пример:

flip(X), где

X = 1 4 результирует в 3 6

2 5 2 5

3 6 1 4

* 1. Перевод содержимого раздела «The Colon Operator»

Colon (двоеточие): создание массивов, извлечение массивов, итерации в цикле for. Двоеточие - один из наиболее полезных операторов в MATLAB. С помощью него можно создавать векторы, извлекать массивы и организовывать итерирование в цикле for.

### Синтаксис

x = j:k

x = j:i:k

A(:,n)

A(m,:)

A(:)

A(j:k)

### Описание

x = j:k создает массив x, содержащий элементы [j, j+1, j+2, ..., j+m] с единичным шагом,  где m = fix(k - j). Если j и k - целые, тогда массив имеет вид [j, j+1, ..., k].

x = j:i:k создает массив x с шагом i между элементами. Члены вектора в точности соответствуют [j, j+i, j+2\*i, ..., j+m\*i] где m = fix((k - j) / i). Однако, если i - нецелое число, тогда от арифметики чисел с плавающей точкой зависит момент, когда следует прекратить заполнение создаваемого массива. Если вы обозначаете нескалярные массивы, тогда MATLAB интерпретирует запись j:i:k как j(1):i(1):k(1).

x = colon(j,k) и x = colon(j,i,k) - альтернативные способы выполнить команды j:k и j:i:k, но они редко используются. Такой синтаксис позволяет перегружать операторы классов.

A(:,n), A(m,:), A(:), и A(j:k)- обычные выражения для индексирования матрицы А, содержащие двоеточие. Если использовать двоеточие в качестве индекса в таком выражении, как A(:,n), то такая запись играет роль сокращения для получения всех элементов, находящихся в соответствующей размерности массива. Также обычным вариантом использования является создание вектора с помощью двоеточия для индексирования, например так: A(j:k). Некоторые выражения включают оба варианта использования оператора двоеточия, например: A(:,j:k).

Стандартные выражения для индексирования, содержащие двоеточие:

* A(:, n) возвращает n-й столбец матрицы A.
* A(m, :) возвращает m-ю строку матрицы A.
* A(:, :, p) возвращает p-й член трехмерного массива A.
* A(:) преобразует все элементы массива A в вектор-столбец. Ничего не делает в случае, когда A уже является вектором.
* A(:, :) преобразует все элементы массива A в матрицу. Ничего не делает в случае, когда A уже является матрицей.
* A(j:k) использует вектор j:k для получения значений из A: [A(j), A(j+1), …, A(k)].
* A(:, j:k) содержит все позиции первой размерности и использует вектор j:k для индексирования второй размерности. Возвращает матрицу со столбцами [A(:, j), A(:, j+1), …, A(:, k)].

### Примеры

Создать вектор с числами от 1 до 10 и единичным шагом. Оператор –двоеточие по умолчанию использует инкремент +1.

x = 1:10

1 2 3 4 5 6 7 8 9 10

Создать вектор, значения элементов которого различаются на заданное значение.

x = 0:0.1:1

0 0.1000 0.2000 0.3000 0.4000 0.5000 0.6000 0.7000 0.8000 0.9000 1.0000

Создать вектор, значения элементов которого декрементируются.

y = 10:-2:0

10 8 6 4 2 0

Создать матрицу 3x3, индексировать первую строку.

A = magic(3)

8 1 6

3 5 7

4 9 2

A(1,:)

8 1 6

Индексировать второй и третий столбец.

A(:,2:3)

1 6

5 7

9 2

Преобразовать матрицу в вектор-столбец.

A(:)

8

3

4

1

5

9

6

7

2

Задать цикл for

В контексте цикла for оператор-двоеточие задает количество его итераций. Написать цикл, который возводит в квадрат числа от 1 до n.

for n = 1:4

n^2

end

ans = 1

ans = 4

ans = 9

ans = 16

### Советы

* Страница справочника о цикле for содержит описание использования оператора-двоеточия в контексте циклических операторов.
* linspace идентичен оператору-двоеточию, но дает полные контроль над количеством точек и всегда содержит конечные точки. Функция logspase создает значения, отличающиеся на логарифмические величины.
* При создании вектора для индексирования в массиве с ячейками или структурном массиве MATLAB возвращает список значений, разделенных запятыми.
  1. Перевод раздела Basic matrix operations

Эти примеры показывают базовые методы и функции для работы с матрицами в языке MATLAB.

Создадим вектор с 9 элементами под названием a:

a = [1 2 3 4 6 4 3 4 5]

Теперь добавим 2 к каждому элементу вектора a и поместим результат в новый вектор. Отметим, что MATLAB использует один синтаксис для работы как с матрицами, так и с векторами.

b = a + 2

Создание графиков в MATLAB вызывается одной командой. Создадим график созданного ранее вектора с разметкой.

plot(b)

grid on

В MATLAB можно создавать и другие виды графиков.

bar(b)

xlabel('Sample #')

ylabel('Pounds')

Также MATLAB в графиках может использовать символы. Пример использования звездочек для обозначения точек на графике.

plot(b,'\*')

axis([0 10 0 10])

MATLAB удобен в вычислении матриц.

Создание матриц так же просто, как и создание векторов, только с использованием «;» для разделения строк матриц.

A = [1 2 0; 2 5 -1; 4 10 -1]

Можно легко транспонировать матрицу:

B = A'

Теперь перемножим эти две матрицы. Заметьте, что MATLAB не заставляет вас рассматривать матрицы как коллекции чисел. MATLAB знает, когда вы работаете с матрицами, и выбирает необходимые операции.

C = A \* B

Вместо умножения матриц, мы можем умножить соответствующие элементы двух матриц, используя оператор «.\*».

C = A .\* B

Используем MATLAB для решения уравнения A\*x=b. Для этого используем оператор «\».

b = [1;3;5]

x = A\b

Теперь можно показать, что A\*x действительно равно b.

r = A\*x – b

MATLAB содержит функции почти для каждой операции над матрицами. Вот функция для получения собственных значений:

eig(A)

Вот функция для получения особенных значений

svd(A)

Функция «poly» генерирует вектор, содержащий коэффициенты характеристического полинома.

p = round(poly(A))

Мы можем легко найти корни полинома используя функцию roots

roots(p)

MATLAB имеет множество приложений кроме работы над матрицами. Чтобы свернуть векторы:

q = conv(p,p)

Или свернуть снова и сделать график по результату:

r = conv(p,q);

plot(r);

В любое время можно получить список переменных, содержащихся в памяти, используя команду who или whos:

whos

Можно получить значение определенной переменной, введя ее имя:

A

На одной строке может быть введено несколько выражений. Для этого выражения необходимо разделять запятыми или точками с запятой.

Если не присвоить результат операции переменной, результат будет помещен во временной переменной под названием ans.

* 1. Вывод

В ходе лабораторной работы было получено общее представление о математическом пакете MATLAB, главное окно программы, операции helpwin, helpdesk, demo, окно поиска документации.

Также были изучены следующие функции оператора «двоеточие»: создание массивов, индексирование массивов, инициализация цикла for.

Кроме того, были изучены способы построения графиков: функции plot, figure, xlabel, ylabel, title.