

Интерфейс, типы данных, операторы

Основы MATLAB

Юдинцев В. В.

Кафедра теоретической механики

13 сентября 2024 г.



САМАРСКИЙ УНИВЕРСИТЕТ
SAMARA UNIVERSITY

Содержание

1 Введение

2 Основы работы в MATLAB

3 Операторы

Введение

История создания

- MATLAB как язык программирования был разработан Кливом Моулером (Cleve Moler) в конце 1970-х годов для упрощения использования программных библиотек LINPACK и EISPACK без необходимости изучения Фортрана.
- В начале 80-х Джек Литл (Jack Little) Модернизировал эту систему для персональных компьютеров типа IBM PC, VAX и Macintosh.
- В 1984 основана компания The MathWorks inc.

Структура MATLAB

- Высокоуровневый **интерпретируемый** язык программирования, включающий основанные на матрицах структуры данных, широкий спектр функций, интегрированную среду разработки, объектно-ориентированные возможности и интерфейсы к программам, написанным на других языках программирования.
- **Toolboxes** – коллекции MATLAB-функций, для решения определённого класса задач (Optimization Toolbox, Partial Differential Equation Toolbox, Spline Toolbox, Statistic Toolbox).
- **Simulink** – приложение для анализа динамических систем.

Свободное ПО

Близкое по функциональности свободное ПО:

- **GNU Octave**

<https://www.gnu.org/software/octave>

Онлайн версия <https://octave-online.net>

- **Python** с библиотеками numpy, scipy, matplotlib;

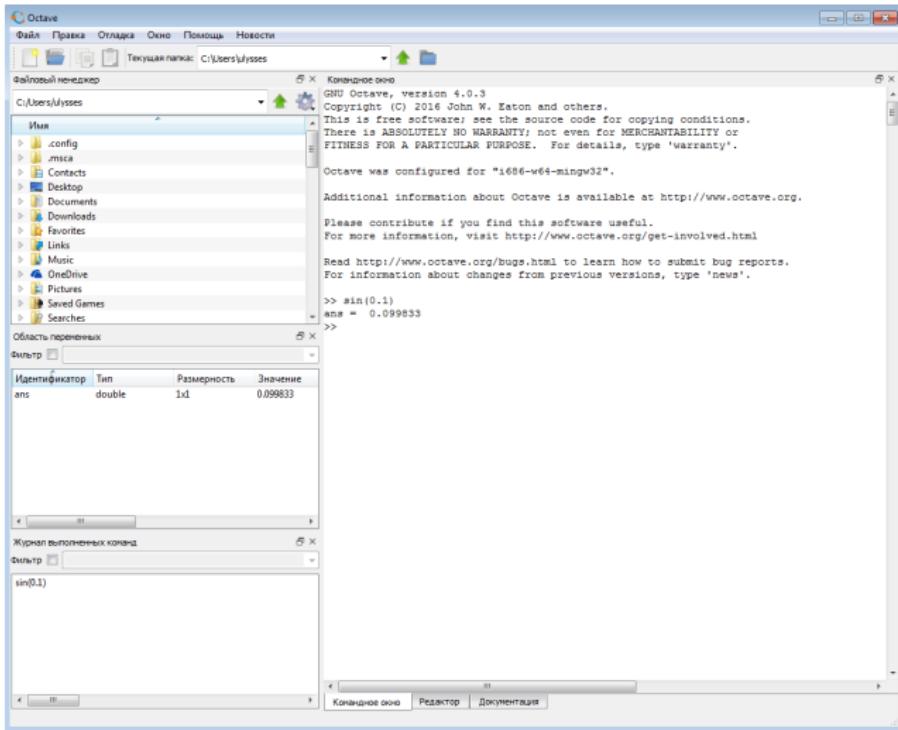
- **Sagemath** <http://www.sagemath.org>.

- **FreeMat** <http://freemat.sourceforge.net>;

- **Scilab** <http://www.scilab.org>;

- **R** (для статистических расчётов) <https://www.r-project.org>.

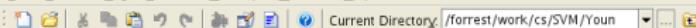
GNU Octave



Основы работы в MATLAB

Окно программы

- Command window – окно команд;
- Command history – окно истории истории команд;
- Current directory – окно, содержащие список файлов и папок текущего каталога;
- Editor – текстовый редактор.
- Workspace – окно со списком переменных текущей сессии.



Shortcuts How to Add What's New

Current Directory Workspace

Name	Value	Min	Max
C	100	100	100
X	<600x784 double>	0	255
X1	<300x784 double>	0	255
Y	<600x1 double>	-1	1
Y1	<300x1 double>	-1	1
accuracy	0.9600	0.96...	0.96...
ans	0	0	0
digit23Trn	<600x784 double>	0	255
digit23Tst	<300x784 double>	0	255
digit23TstT	<300x1 double>	-1	1
err	12	12	12
err_rate	0.0400	0.04...	0.04...
kernel	0	0	0
options	<1x1 struct>		
predictions	<300x1 double>	-1	1
sigma	0.5000	0.50...	0.50...
x	300	300	300

Command History

```

load predictions
predictions
predictions = sign(predictions)
%-> 4/17/08 9:42 PM --%
SA
loss
%-> 4/18/08 11:06 AM --%
svmclassify
%-> 4/19/08 3:33 PM --%
%-> 4/19/08 3:38 PM --%
-X= load('digit23Trn.mat')
-X= X.digit23Trn
-X= X.digit23Trn
-Y= load('digit23TrnT')
-X= load('digit23Trn.mat');
-X= X.digit23Trn;
-Y= load('digit23TrnT');
-svmwrite('Train',X,Y);
-X1 = load('digit23Tst.mat');
-X1 = X1.digit23Tst;
-Y1 = load('digit23TstT');
-svmwrite('Test',X1,Y1);

```

Command Window

1 New to MATLAB? Watch this [Video](#), see [Demos](#), or read [Getting Started](#).

Runtime (without I/O) in cpu-seconds: 0.00

Accuracy on test set: 96.00% (288 correct, 12 incorrect, 300 total)

Precision/recall on test set: 96.62%/95.33%

err_rate =

0.0400

??? Error: File: PROJmain.m Line: 38 Column: 24

The expression to the left of the equals sign is not a valid target for an assignment.

Writing 100 200 300 400 500 600 done.

Writing 100 200 300 done.

Calling SVMlight:

svm_learn -c 100 -t 0 Train model

Scanning examples...done

Reading examples into memory...100..200..300..400..500..600..0K. (600 examples read)

Optimizing.....

Optimization finished (0 misclassified, maxdiff=0.00099).

Runtime in cpu-seconds: 0.21

Number of SV: 88 (including 0 at upper bound)

L1 loss: loss=0.00000

Norm of weight vector: |w|=0.00907

Norm of longest example vector: |x|=3499.11360

Estimated VCdim of classifier: VCdim<=1008.17259

Computing XiAlpha-estimates...done

Runtime for XiAlpha-estimates in cpu-seconds: 0.00

XiAlpha-estimate of the error: error<=13.67% (rho=1.00,depth=0)

XiAlpha-estimate of the recall: recall=>85.00% (rho=1.00,depth=0)

XiAlpha-estimate of the precision: precision=>87.33% (rho=1.00,depth=0)

Number of kernel evaluations: 29199

Writing model file...done

Calling SVMlight:

svm_classify Test model predictions

Reading model...OK. (88 support vectors read)

Classifying test examples...100..200..300..done

Runtime (without I/O) in cpu-seconds: 0.00

Accuracy on test set: 96.00% (288 correct, 12 incorrect, 300 total)

Precision/recall on test set: 96.62%/95.33%

accuracy =

0.9600

MATLAB как калькулятор

- `a=1.2` – присвоение некоторого значения переменной `a`

```
1 >> a=1.2  
2 a = 1.2
```

- `b=sin(a)*sqrt(a+2)` – вычисление выражения и вывод результата

```
1 >> b=sin(a)*sqrt(a+2)  
2 b =  
3 1.6673
```

- `;` – точка с запятой в конце выражения подавляет вывод результата:

```
1 >> b=sin(a)*sqrt(a+2);  
2 >>
```

Базовые команды редактора (Command window)

- ↑ – возврат к предыдущей команде.
- `help имя функции` – справка по функции (или F1).
- `cls` – очистить окно команд (Command window).
- `clear` – удалить все переменные в текущей сессии.
-   - (в окне команд) прерывание вычислений.

Синтаксис

- Имена переменных могут состоять из латинских букв, цифр, знаков подчёркивания: `a`, `a1`, `x1`, `x_1`. Имена переменных чувствительны к регистру: `A` и `a` – это разные переменные.
- При создании переменной может быть явно указан тип (по умолчанию `double`):

```
1 >> a=int16(25);
```

Типы данных

Тип	Описание
<code>double</code>	вещественный, 64 бит
<code>single</code>	вещественный, 32 бит
<code>int8</code>	знаковый целочисленный, 8 бит
<code>int16</code>	знаковый целочисленный, 16 бит
<code>int32</code>	знаковый целочисленный, 32 бит
<code>int64</code>	знаковый целочисленный, 64 бит
<code>uint8</code>	беззнаковый целочисленный, 8 бит
<code>uint16</code>	беззнаковый целочисленный, 16 бит
<code>uint32</code>	беззнаковый целочисленный, 32 бит
<code>uint64</code>	беззнаковый целочисленный, 64 бит

Синтаксис

- Квадратные скобки используются для создания векторов и матриц: `a=[1,2,3,4]`.
- Круглые скобки используются для вызова функций и для группировки выражений: `sin(1.2)+a*(c+1)`.
- Фигурные скобки для создания массивов ячеек:
`a={'Масса', 10}`

Управление переменными

- `whos` показать переменные текущей сессии

```
1 >> whos
2   Name      Size            Bytes  Class    Attributes
3
4   a            1x1              8  double
5   ans          1x5             40  double
6   b            1x1              8  double
```

- `clear` удалить все переменные текущей сессии
- `clear f1, f2` удалить переменные `f1, f2` текущей сессии

Ведение “протокола” работы

- `diary filename` – ведет запись на диск всех команд в строках ввода и полученных результатов в виде текстового файла.
- `diary off` – приостанавливает запись в файл.
- `diary on` – вновь начинает запись в файл.

Сохранение сессии

Сохранение значения всех переменных в файл `*.mat`

- `save filename` – запись в файл `filename.mat` текущей сессии (значение всех переменных).
- `load filename` – загрузка значений переменных из файла `filename`.

Ввод чисел

a=1.25

a=9.3e10

$9.3 \cdot 10^{10}$

a=2.36e-5

$2.36 \cdot 10^{-5}$

```
1 >> a=1.25
2 a = 1.2500
```

```
1 >> a=9.3e10
2 a = 9.3000e+10
```

```
1 >> a=2.36e-5
2 a = 2.3600e-05
```

Формат отображения результата

```
1 >> format short;
2 >> pi
3 ans =
4     3.1416
```

```
1 >> format long;
2 >> pi
3 ans =
4     3.141592653589793
```

```
1 >> format short e;
2 >> pi
3 ans =
4     3.1416e+00
```

```
1 >> format long e;
2 >> pi
3 ans =
4     3.141592653589793e+00
```

Константы

- `pi` – число π

```
1 >> pi  
2 ans = 3.1416
```

- `realmin` – минимальное положительное число

```
1 >> realmin  
2 ans = 2.2251e-308
```

- `realmax` – максимальное положительное число

```
1 >> realmax  
2 ans = 1.7977e+308
```

Использование имен переменных, совпадающих со встроенными именами, не рекомендуется

Комплексные числа

- `complex(1,3)` – комплексное число $1 + 3i$

```
1 >> a = complex(1,3);
```

- `i` или `j` – мнимая единица $\sqrt{-1}$

```
1 >> b = 1+2i;
2 >> a*b
3 ans =
4 -5.0000 + 5.0000i
```

$$(1 + 3i) \cdot (1 + 2i) = 1 + 2i + 3i - 6 = -5 + 5i$$

ans – результат вычисления предыдущего действия

Встроенная переменная **ans** хранит результат последнего действия:

```
1 >> 2+2  
2 ans =  
3     4
```

Эту переменную можно использовать в выражениях:

```
1 >> ans*5  
2 ans =  
3     20
```

Типы данных: векторы

[1.2 1.5 1.9 0.5] или [1.2, 1.5, 1.9, 0.5] вектор-строка

$$\begin{bmatrix} 1.2 & 1.5 & 1.9 & 0.5 \end{bmatrix}$$

[1.2; 1.5; 1.9; 0.5] вектор-столбец

$$\begin{bmatrix} 1.2 \\ 1.5 \\ 1.9 \\ 0.5 \end{bmatrix}$$

Матрицы и скаляры

Каждая переменная в MATLAB – это матрица, поэтому переменная

```
1 >> a=1;
```

это матрица размерности 1x1:

```
1 >> a(1,1)
2 ans =
3     1
```

Типы данных: матрицы

- $A = [1.2 \ 1.5 \ 1.9; \ 0.5 \ 0.6 \ 0.7; \ 0.1 \ 1 \ 3]$

$$A = \begin{bmatrix} 1.2 & 1.5 & 1.9 \\ 0.5 & 0.6 & 0.7 \\ 0.1 & 1 & 3 \end{bmatrix}$$

- элементы вводятся построчно;
- элементы матрицы в строке можно разделять пробелами или запятыми;
- для разграничения строк используются точка с запятой;
- элемент матрицы может быть числом или выражением.

Операторы

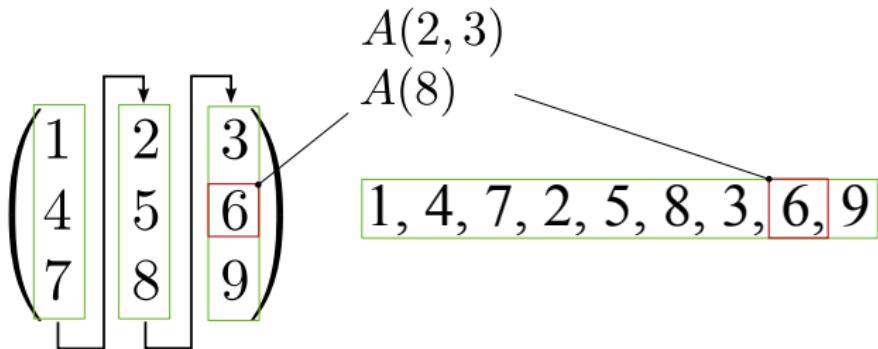
Оператор ()

`help ops` вывести список всех операторов

`(i,j,k,...)` – доступ к элементам матрицы. `A(i,j)` – i строка, j столбец.

Оператор ()

$A(i)$ – i элемент вектора (строки или столбца) или i элемент матрицы, при расположении элементов по столбцам.



Логическое индексирование

Логическое индексирование позволяет выбрать из вектора или матрицы элементы, удовлетворяющие заданному условию.

```
1 >> a = sin(1:0.5:5)
2 a =
3     0.8415    0.9975    0.9093    0.5985    0.1411   -0.3508
      -0.7568   -0.9775   -0.9589
```

```
1 >> ind=a<0
2 ind =
3     0     0     0     0     0     1     1     1     1
```

```
1 >> a(ind)
2 ans =
3     -0.3508   -0.7568   -0.9775   -0.9589
```

Скалярное произведение векторов

$$\vec{a} \cdot \vec{b} = |a||b| \cos \varphi = a_x b_x + a_y b_y + a_z b_z$$

Вычисление скалярного произведения, используя встроенную функцию `dot`:

```
1 >> a=[1 2 3];
2 >> b=[4 5 6];
3 >> dot(a,b)
4 ans =
5      32
```

Скалярное произведение векторов

$$\vec{a} \cdot \vec{b} = |a||b| \cos \varphi = a_x b_x + a_y b_y + a_z b_z$$

Вычисление скалярного произведения, используя оператор матричное умножение:

```
1 >> a*b'  
2 ans =  
3      32
```

Матричное умножение

A*B умножение чисел или матриц (матричное умножение)

```
1 >> a = [1 2 3  
2           4 5 6];  
3 >> b = [1 2  
4           3 4  
5           5 6];  
6 >> a*b  
7 ans =  
8     22      28  
9     49      64
```

Поэлементное умножение

$C = A \cdot * B$ поэлементное умножение матриц $C_{ij} = A_{ij}B_{ij}$

```
1 >> a = [1 2;
2           3 4];
3 >> b = [4 7;
4           1 3];
5 >> a.*b
6 ans =
7     4      14
8     3      12
```

Возведение в степень

Возведение в степень матрицы или числа

```
1 >> a=[1 2  
2           3 4];  
3 >> a^2  
4 ans =  
5     7      10  
6     15      22
```

$$a^2 = a \cdot a = \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} = \begin{pmatrix} 7 & 10 \\ 15 & 22 \end{pmatrix}$$

Возведение в степень

Возведение в степень всех элементов матрицы

```
1 >> a=[1 2  
2      3 4];  
3 >> a.^2  
4 ans =  
5      1      4  
6      9     16
```

Операторы

- \mathbf{A}' – транспонирование матрицы:

```
1 >> A=[1.2  0.3;
2           0.5  2.1];
3 >> A'
4 ans =
5     1.2000    0.5000
6     0.3000    2.1000
```

- $\mathbf{C}=\mathbf{A}/\mathbf{B}$ – деление $C = AB^{-1}$
- $\mathbf{C}=\mathbf{A}./\mathbf{B}$ – поэлементное деление $C_i = A_i/B_i$

Решение системы линейных уравнений

Деление $A \setminus B$ выполняет операцию $C = A^{-1}B$ – решение СЛУ с матрицей коэффициентов A и матрицей правой части B :

$$\begin{cases} 1.2x_1 + 0.3x_2 = 1; \\ 0.5x_1 + 2.1x_2 = 2. \end{cases}$$

```
1 >> A=[1.2 0.3;0.5 2.1];
2 >> B=[1;2];
3 >> A\B
4 ans =
5      0.6329
6      0.8017
```

Логические операторы < > = >= <= ~=

Результатом операции над матрицей или вектором является логический вектор (матрица):

```
1    >> A=[1 ,2 ,3 ,4 ,5 ,6];
2    >> A<5
3    ans=
4        [1 ,1 ,1 ,1 ,0 ,0]
5    >> A==3
6    ans=
7        [0 ,0 ,1 ,0 ,0 ,0]
8    >> A~=5
9    ans=
10       [1 ,1 ,1 ,1 ,0 ,1]
11
```

Оператор :

- Для создания списков с равноотстоящими значениями используется оператор :
 $n1:s:n2$ $n1, n1 + s, n1 + 2s, \dots, n_k, n_k \leq n2$
- Оператор : может использоваться для доступа к элементам матрицы и вектора
- $A=1:10$ – 1, 2, 3, 4, 5, 6, 7, 8, 9, 10
- $A(1:2:10)$ – нечетные элементы вектора: [1, 3, 5, 7, 9]

Оператор :

Для прямоугольной матрицы A :

- $A(:, k)$ – k -ый столбец матрицы A
- $A(k, :)$ – k -ая строка матрицы A
- $A(:, [2, 4, 5])$ – второй, четвертый и пятый столбец матрицы A

```
1 >> a = magic(4)
2 a =
3     16      2      3     13
4     5     11      10      8
5     9      7      6     12
6     4     14     15      1
7 >> a([1 3], :)
8 ans =
9     16      2      3     13
10    9      7      6     12
```

Блоки матриц

$A(1:2,3:4)$ – блок матрицы A, лежащий на пересечении строк 1 и 2, и столбцов 3 и 4.

```
1 >> a = magic(4)
2 a =
3     16      2      3     13
4     5     11     10      8
5     9      7      6     12
6     4     14     15      1
7 >> a([1 2],[3 4])
8 ans =
9     3     13
10    10      8
```

Удаление строк и столбцов из матрицы

$A(3, :) = []$ – удалить третью строку из матрицы A

```
1 >> a = magic(4)
2 a =
3 16 2 3 13
4 5 11 10 8
5 9 7 6 12
6 4 14 15 1
7 >> a(3,:) = []
8 a =
9 16 2 3 13
10 5 11 10 8
11 4 14 15 1
```