Справочник по формулам Scilab, используемых при работе с матрицами

1.Ввод матриц

1)Поэлементный ввод:

Вектор-строка: ввести имя массива, а затем после знака присваивания, в квадратных скобках через пробел или запятую, перечислить элементы массива:

name=[x1 x2 ... xn]

или

name=[x1, x2, ..., xn]

Вектор-столбец: ввести имя массива, а затем после знака присваивания, в квадратных скобках через точку с запятой, перечислить элементы массива:

name=[x1; x2; ...; xn]

Обратиться к элементу вектора можно, указав имя массива и порядковый

номер элемента в круглых скобках: name(индекс)

Ввод элементов матрицы также осуществляется в квадратных скобках, при этом элементы строки отделяются друг от друга пробелом или запятой, а строки разделяются между собой точкой с запятой:

name=[x_{11}, x_{12}, ..., x_{1n}; x_{21}, x_{22}, ..., x_{2n}; ...; x_{m1}, x_{m2}, ..., x_{mn};…]

Обратиться к элементу матрицы можно, указав после имени матрицы, в круглых скобках через запятую, номер строки и номер столбца на пересечении которых элемент расположен: name(индекс1, индекс2)

2) Составляя из ранее заданных матриц и векторов:

<u>Ex 1:</u>

v1=[1 2 3]; v2=[4 5 6]; v3=[7 8 9]

V = [v1 v2 v3]

Получим:

V = 1 2 3 4 5 6 7 8 9

<u>Ex 2:</u>

V = [v1; v2; v3]

V = 1 2 3

4 5 6

7 8 9

Ex 3:

M = [V V V]

M = 1 2 3 1 2 3 1 2 3

4 5 6 4 5 6 4 5 6

7 8 9 7 8 9 7 8 9

Ex 4:

M = [V;V]

M = 1 2 3

4 5 6

7 8 9

1 2 3

4 5 6

7 8 9

2.Для работы с матрицами и векторами в Scilab предусмотрены следующие операции:

+ — сложение

- — вычитание[1]

’ — транспонирование[2]

* — матричное умножение[3]

* — умножение на число

ˆ — возведение в степень[4]

\ — левое деление[5]

/ — правое деление[6]

.* — поэлементное умножение матриц

.ˆ — поэлементное возведение в степень

.\ — поэлементное левое деление

./ — поэлементное правое деление

1Операции сложения и вычитания определены для матриц одной размерности или векторов одного типа, т.е. суммировать (вычитать) можно либо векторы-столбцы, либо векторы-строки одинаковой длины.

2) Если в некоторой матрице заменить строки соответствующими столбцами, то получится транспонированная матрица.

3) Операция умножения вектора на вектор определена только для векторов одинакового размера, причем один из них должен быть вектором-столбцом, а второй вектором-строкой. Матричное умножение выполняется по правилу «строка на столбец» и допустимо, если количество строк во второй матрице совпадает с количеством столбцов в первой. Кроме того, переместительный закон на произведение матриц не распространяется.

4) Возвести матрицу в n-ю степень значит умножить ее саму на себя n раз. При этом целочисленный показатель степени может быть как положительным, так и отрицательным. В первом случае выполняется алгоритм умножения матрицы на себя указанное число раз, во втором умножается на себя матрица, обратная к данной.

5) (A\B) ⇒ (A−1B), операция может быть применима для решения матричного уравнения вида $A \cdot X = B$, где X — неизвестный вектор.

6) (B/A) ⇒ (B · A−1), используют для решения матричных уравнений вида $X \cdot A = B$.

3. Функции применимые к матрицам

1) Важную роль при работе с матрицами играет знак двоеточия «:». Указывая его вместо индекса при обращении к массиву, можно получать доступ к группам его элементов.

A=[5 7 6 5; 7 10 8 7;6 8 10 9;5 7 9 10]

Выделение отдельной строки/столбца:

A(:,2)	A(3,:)
ans = 7	ans = 6 8 10 9
10	
8	
7	

Выделение подматрицы:

M=A(3:4,2:3)

M = 8 10

7 9

Удаление столбца/строки:

A(:,2)=[]

A = 5 8 10

7 7 9

6 10 9

5 9 10

A(3,:)=[]

A = 5 8 10

7 7 9

5 9 10

2) Специальные матричные функции:

- matrix(A [,n,m]) — преобразует матрицу A в матрицу другого размера;
  D=[1 2;3 4;5 6];
  matrix(D,2,3)
  ans = 1. 5. 4.
  3. 2. 6.
- ones(m,n) — создает матрицу единиц из m строк и n столбцов;
  ones(2,2)
  ans = 1. 1.
  1. 1.
- zeros(m,n) — создает нулевую матрицу из m строк и n столбцов;
  zeros(3,2)
  ans = 0. 0.
  0. 0.
  0. 0.
- eye(m,n) — формирует единичную матрицу из m строк и n столбцов;
  m=3; n=4;
  E=eye(m,n)
  E = 1. 0. 0. 0.
  0. 1. 0. 0.
  0. 0. 1. 0.
- rand(n1,n2,...nn[,fl]) — формирует многомерную матрицу случайных чисел;
  rand(2,2)
  ans = 0.2113249 0.0002211
  0.7560439 0.3303271
- sparse([i1 j1;i2 j2;...;in jn],[n1,n2,...,nn]) — формирует разреженную матрицу. Для создания матрицы такого типа необходимо указать индексы ее ненулевых элементов — [i1 j1,i2 j2,...,in jn], и их значения — [n1,n2,...,nn]. Индексы одного элемента отделяются друг от друга либо пробелом, либо запятой, а пары индексов — соответственно точкой с запятой, значения элементов разделяются запятыми.
- full(M) — вывод разреженной матрицы M в виде таблицы;

A = sparse([1 3;3 2;3 5],[4,5,6])
A = (3, 5) sparse matrix
(1, 3) 4.
(3, 2) 5.
(3, 5) 6.
full(A)
ans =0. 0. 4. 0. 0.
0. 0. 0. 0. 0.
0. 5. 0. 0. 6.

- hypermat(D[,V]) — создание многомерной матрицы с размерностью, заданной вектором D и значениями элементов, хранящихся в векторе V (использование параметра V необязательно);
  M = hypermat([2 2 3],0:11)
  M = (:,:,1)
  0. 2.
  1. 3.
  (:,:,2)
  4. 6.
  5. 7.
  (:,:,3)
  8. 10.
  9. 11.
- diag(V[,k]) — возвращает квадратную матрицу с элементами V на главной или на k-й диагонали; функция diag(A[,k]), где A — ранее определенная матрица, в качестве результата выдаст вектор-столбец, содержащий элементы главной или k-ой диагонали матрицы A;
  V=[1,2,3];
  diag(V)
  ans = 1 0 0
  0 2 0
  0 0 3
- cat(n, A, B, [C, ...]) — объединяет матрицы A и B или все входящие матрицы, при n=1 по строкам, при n=2 по столбцам; то же что [A; B] или [A, B];

A=[1 2;3 4]; B=[5 6 ;7 8];
cat(2,A,B)
ans =1 2 5 6
        3 4 7 8

- tril(A[,k]) — формирует из матрицы A нижнюю треугольную матрицу, начиная с главной или с k-й диагонали;
  A=[1 2 3;4 5 6;7 8 9]
  tril(A)
  ans = 1 0 0
          4 5 0
          7 8 9
- triu(A[,k]) — формирует из матрицы A верхнюю треугольную матрицу, начиная с главной или с k-й диагонали;
  A=[1 2 3;4 5 6;7 8 9];
  triu(A)
  ans =1 2 3
          0 5 6
          0 0 9
- sort(X) — выполняет упорядочивание массива X; если X — матрица, сортировка выполняется по столбцам;
  b=[2 0 1]; sort(b)
  ans =2. 1. 0.

Функции вычисления различных числовых характеристик матриц:

- size(V[,fl]) — определяет размер массива V; если V — двумерный массив, то size(V,1) или size(V,’r’) определяют число строк матрицы V, а size(V,2) или size(V,’c’) — число столбцов;
  M=[1 2;3 4;5 6;7 8];
  [n,m]=size(M)
  m =
  2.
  n =
  4.
- length(X) — определяет количество элементов массива X; если X — вектор, его длину; если X — матрица, вычисляет общее число ее элементов;
  V=[-1 0 3 -2 1 -1 1];
  length(V)//Длина вектора
  ans =7

- sum(X[,fl]) — вычисляет сумму элементов массива X, имеет необязательный параметр fl. Если параметр fl отсутствует, то функция sum(X) возвращает скалярное значение, равное сумме элементов массива. Если fl=’r’ или fl=1, что то же самое, то функция вернет строку, равную поэлементной сумме столбцов матрицы X. Если fl=’c’ или fl=2, то результатом работы функции будет вектор-столбец, каждый элемент которого равен сумме элементов строк матрицы X.
  M=[1 2 3;4 5 6;7 8 9];
  Y=sum(M)
  Y = 45.
- prod(X[,fl]) — вычисляет произведение элементов массива X, работает аналогично функции sum;
  prod(M)
  ans = 362880.
  p1=prod(M,1)
  p1 =
  28 80 162
- max(M[,fl]) — вычисляет наибольший элемент в массиве M, имеет необязательный параметр fl. Если параметр fl отсутствует, то функция max(M) возвращает максимальный элемент массива M; если fl=’r’, то функция вернет строку максимальных элементов столбцов матрицы M; если fl=’c’, то результатом работы функции будет вектор-столбец, каждый элемент которого равен максимальному элементу соответствующих строк матрицы M. Функция [x, nom]=max(M[,fl]) вернет значение максимального элемента x и его номер в массиве nom;
  M=[5 0 3;2 7 1;0 4 9];
  max(M)
  ans =
  9.
- min(M[,fl]) — вычисляет наименьший элемент в массиве M, работает аналогично функции max;
  A=[5 10 3 2 7 1 25 4 0];
  [x,nom]=min(A)
  nom =
  7.
  x =
  25.
- mean(M[,fl]) — вычисляет среднее значение массива M; если M двумерный массив, то mean(M,1) или mean(M,’r’) определяют среднее

значение строк матрицы M, а mean(M,2) или mean(M,'c') — среднее значение столбцов;
mean(M)
ans =
3.4444444

- median(M[,fl]) — вычисляет медиану1 массива M, работает аналогично функции mean;
M=[5 0 3;2 7 1;0 4 9];
median(M)
ans = 3.
- det(M) — вычисляет определитель квадратной матрицы M;
M=[1 0 2;3 2 1;0 3 1];
det(M)
ans = 17.
- rank(M[,tol]) — вычисление ранга матрицы M с точностью tol.
M=[1 0 2;3 2 1;0 3 1];
rank(M)
ans = 3.
- norm(M[,fl]) — вычисление нормы квадратной матрицы M; тип нормы определяется необязательной строковой переменной fl, по умолчанию fl=2. Функции norm(M) и norm(M,2) эквивалентны и вычисляют вторую норму матрицы M. Первая норма определяется функцией norm(M,1). Функции norm(M,'inf') и norm(M,'fro') вычисляют соответственно бесконечную и евклидову нормы. Если V — вектор, то результатом работы функции norm(V,1) будет сумма модулей всех элементов вектора V. С помощью функции norm(V,2) можно вычислить модуль вектора V. Значение norm(V,'inf') равно модулю максимального элемента вектора по модулю;
M=[1 0 2;3 2 1;0 3 1];
norm(M,1)
ans = 5.
- cond(M) — вычисляет число обусловленности1 матрицы M по второй норме;
A=[5 7 6 5;7 10 8 7;6 8 10 9;5 7 9 10];
cond(A)
ans =
2984.0927

Функции, реализующие численные алгоритмы решения задач линейной алгебры:

- spec(M) — вычисляет собственные значения и собственные векторы квадратной матрицы M.
  M=[3 -2;-4 1]
  M =
  3. - 2.
  - 4. 1.
  spec(M)
  ans =
  - 1.
  5.
- inv(A) — вычисляет матрицу, обратную к A;
  A=[1 2 3 5;0 1 3 2;4 2 1 1;2 3 0 1];
  inv(A)
  ans =
  ! 0.0285714 - 0.1428571 0.3428571 - 0.2 !
  ! - 0.1428571 0.2142857 - 0.2142857 0.5 !
  ! - 0.2 0.5 0.1 - 0.1 !
  ! 0.3714286 - 0.3571429 - 0.0428571 - 0.1 !
- pinv(A[,tol]) — вычисляет псевдообратную матрицу для матрицы A с точностью tol (необязательный параметр);
  pinv(A)
  ans =
  0.0285714 - 0.1428571 0.3428571 - 0.2
  - 0.1428571 0.2142857 - 0.2142857 0.5
  - 0.2 0.5 0.1 - 0.1
  0.3714286 - 0.3571429 - 0.0428571 - 0.1
- linsolve(A,b) — решает систему линейных алгебраических уравнений вида A · ~x −~b = 0.
- rref(A) — осуществляет приведение матрицы A к треугольной форме, используя метод исключения Гаусса;
- lu(M) — выполняет треугольное разложение матрицы M;
- qr(M) — выполняет разложение матрицы M на ортогональную и верхнюю треугольную матрицы;
- svd(M) — выполняет сингулярное разложение1 размером n×m; результатом работы функции может быть либо сингулярное разложение, либо вектор, содержащий сингулярные значения матрицы;

- kernel(M[,tol[,fl]]) — определение ядра матрицы2 M, параметры tol и fl являются необязательными. Первый задает точность вычислений, второй — используемый при вычислении алгоритм и принимает значения ’qr’ или ’svd’.