MATHEMATICA – AN INTRODUCTION

Additional about:

LISTS

VECTORS

MATRICES

EIGENVALUES

Dealing with Array Variables: Lists and Tables

A **list** can have any kind of object, numbers, variables, functions, and equations. It is entered in the form {x, y, z}, where x, y, and z are **individual element** of the list.

```
In[1]:={1,3,6}
Out[1]= {1, 3, 6}
```

We can find the cube of the elements collectively as follows:

```
In[2]:= {1,3,6}^3
Out[2] = {1, 27, 216}
```

One can **add, multiply, subtract, divide** lists;
plot lists of functions; and solve lists of equations.

```
In[3]:= {9,5,3}-{1,3,2}
Out[3]= {8, 2, 1}
```

Mathematica can do operation with **list** having **symbols**.

```
In[4]:= {x,y,z}.{a,b,c}
Out[4]= a x + b y + c z
```

A **list** can be treated exactly like a **single object**.

```
In[5]:= u = {2, 5, 3}
Out[5] = {2, 5, 3}

In[6]:=u^2+2 u + 1
Out[5] = {9, 36, 16}

In[7]:= N[Exp[u], 4]
Out[7] = {7.389, 148.4, 20.09}
```

To extract the *n-th* element from a list, say u, type either u[[n]]

```
In[8]:= u[[2]]
Out[8]= 5
```

Generation of list

Mathematica provides the Table command to generate the lists.

Table[function, {j, n}]

builds a n-component vector by evaluating the function with j = 1, 2, ..n.

Different initial and final values may be assigned to the counter,

Table[function, {j, jmin, jmax, dj}]

In[9]:= Table[3 x^2 -4 x + 7, {x, 1, 9, 2}]
Out[9]= {6, 22, 62, 126, 214}

If **step-size** dj is not given, Mathematica takes the **default value** 1.

The output of the Table command can be displayed in rows and columns by:

wrapping TableForm [ ] around the command

(or appending the command//TableForm)

```
In[10]:= TableForm[{{1, 6}, {2, 11}, {3, 22}}]
Out[10] =
     1   6
     2   11
     3   22
```

One may construct lists that depend on two or more parameters.

```
In[4]:= Table[m / n, {m, 1, 3}, {n, 2, 4}]
```

Out[4]= $\left\{\left\{\frac{1}{2}, \frac{1}{3}, \frac{1}{4}\right\}, \left\{1, \frac{2}{3}, \frac{1}{2}\right\}, \left\{\frac{3}{2}, 1, \frac{3}{4}\right\}\right\}$

```
In[5]:= % // TableForm
```

Out[5]//TableForm=

$\frac{1}{2}$	$\frac{1}{3}$	$\frac{1}{4}$
1	$\frac{2}{3}$	$\frac{1}{2}$
$\frac{3}{2}$	1	$\frac{3}{4}$

Vector Operations

Conventional Way:-

```
In[11]:= vec[1]= 2.3; vec[2]= 3.4; vec[3]=7.3;

    sum = 0;
    Do[ sum = sum + vec[i]^2; Print[sum], {i, 1,3}]
    Out[11]=
            5.29
            16.85
            70.14
```

A vector is a **list** consisting of the **components** of the **vector**.

```
    In[12]:=Clear[x, y, z, a, b, c]
            v={x, y, z};
            w={a, b, c};
```

For **vector addition**,

```
In[13]:= v + w
Out[13] = {a + x, b + y, c + z}
```

For **scalar product** of two vectors, type

```
In[14]:= v . w
Out[14] = a x + b y + c z
```

For cross product of two vectors, type

```
In[15]:= Cross[v, w]
Out[15] = Cross[{x, y, z}, {a, b, c}]
```

Cross (×)

Cross[a, b]
gives the vector cross product of a and b.

Details

- If a and b are lists of length 3, corresponding to vectors in three dimensions, then Cross[a, b] is also a list of length 3.
- Cross[a, b] can be entered in StandardForm and InputForm as $a \times b$, a esc cross esc b or a\[Cross] b. Note the difference between \[Cross] and \[Times].
- Cross is antisymmetric, so that Cross[b, a] is −Cross[a, b]. »
- Cross[{x, y}] gives the perpendicular vector {− y, x}.
- In general, Cross[v_1, v_2, …, v_{n-1}] is a totally antisymmetric product which takes vectors of length n and yields a vector of length n that is orthogonal to all of the v_i.
- Cross[v_1, v_2, …] gives the dual (Hodge star) of the wedge product of the v_i, viewed as one-forms in n dimensions.

The cross product of two vectors:

```
In[1]:= Cross[{a, b, c}, {x, y, z}]
Out[1]= {-c y + b z, c x - a z, -b x + a y}
```

The cross product of a single vector:

```
In[1]:= Cross[{x, y}]
Out[1]= {-y, x}
```

Find a vector perpendicular to a vector in the plane:

```
In[1]:= u = RandomReal[1, 2]
Out[1]= {0.282533, 0.431914}

In[2]:= v = Cross[u]
Out[2]= {-0.431914, 0.282533}

In[3]:= u.v
Out[3]= 0.
```

Find the normal to the plane spanned by two vectors:

```
In[1]:= u = {1, 2, 3};
        v = {1, 4, 9};

In[2]:= w = Cross[u, v]
Out[2]= {6, -6, 2}
```

The equation for the plane:

```
In[3]:= w.{x, y, z} == 0
Out[3]= 6 x - 6 y + 2 z == 0
```

```
In[18]:= u = RandomReal[1, 3]
         v = RandomReal[1] u
Out[18]= {0.292174, 0.675329, 0.837668}
Out[19]= {0.026958, 0.0623106, 0.0772891}

In[21]:= Cross[u, v]
```

Out[21]= $\{-2.94486\times10^{-18},\ 1.27406\times10^{-18},\ 2.63792\times10^{-19}\}$

```
In[22]:= Chop[Cross[u, v]]
Out[22]= {0, 0, 0}
```

Chop

Chop[*expr*]
replaces approximate real numbers in *expr* that are close to zero by the exact integer 0.

Details

- Chop[*expr*, *delta*] replaces numbers smaller in absolute magnitude than *delta* by 0.
- Chop uses a default tolerance of 10^{-10}.
- Chop works on both Real and Complex numbers.

If u and v are linearly independent, $u \times v$ is nonzero and orthogonal to u and v:

```
In[1]:= {u, v} = RandomReal[1, {2, 3}]
Out[1]= {{0.703636, 0.356147, 0.340631}, {0.267011, 0.8348, 0.304179}}

In[2]:= w = Cross[u, v]
Out[2]= {-0.176027, -0.123079, 0.4923}

In[3]:= Chop[{u.w, v.w}]
Out[3]= {0, 0}
```

Cross is antisymmetric:

```
In[25]:= {u, v} = RandomReal[1, {2, 3}]

Out[25]= {{0.433995, 0.869812, 0.105013}, {0.939323, 0.639939, 0.834624}}
```

```
In[26]:= Cross[u, v] == -Cross[v, u]

Out[26]= True
```

For vectors in 3 dimensions, Cross is bilinear:

In[1]:= $u_1 = \{x_1, y_1, z_1\};$
$u_2 = \{x_2, y_2, z_2\};$

In[2]:= Expand[Cross[a u_1, u_2] == a Cross[u_1, u_2]]

Out[2]= True

In[3]:= Expand[Cross[u_1, b u_2] == b Cross[u_1, u_2]]

Out[3]= True

Cross product computed with exact arithmetic:

```
In[1]:= u = {1, 2, 3};
        v = {1, 1/2, 1/3};

In[2]:= Cross[u, v]
```

Out[2]= $\left\{-\frac{5}{6}, \frac{8}{3}, -\frac{3}{2}\right\}$

Computed with machine arithmetic:

```
In[3]:= Cross[N[u], N[v]]

Out[3]= {-0.833333, 2.66667, -1.5}
```

Computed with arbitrary-precision arithmetic:

```
In[4]:= Cross[N[u, 20], N[v, 20]]

Out[4]= {-0.83333333333333333333, 2.6666666666666666667, -1.5000000000000000000}
```

Cross of one vector in 2 dimensions:

```
In[1]:= Cross[{x, y}]

Out[1]= {-y, x}


In[2]:= {x, y}.%

Out[2]= 0
```

Matrix Operations

Conventional Way

```
In[15]:=   mat[1,1]= 2.1; mat[1,2]=1.8;
     mat[2,1]=4.5; mat[2,2]=7.2;
      tr = 0;
     Do[ tr = tr + mat[i,i]; Print[tr], {i, 1,2}]
     Out[15] =
               2.1
               9.3
```

A **rectangular matrix** is represented by a **list of lists**:
the **sublists** denotes the **rows** of the matrix, and
elements in the sublists denote elements in corresponding columns.

```
     In[16]:= Clear[a,b,c,d,p, q, f, g, m, w, z]
               z = { p ,  q };
               w = {f, g};
                m = {{a, b}, {c, d}};
```

To get the first row,

```
In[17]:= m[[1]]
Out[17] = {a, b}
```

To get a particular matrix element, give its row and column,

```
In[18]:= m[[1,2]]
Out[18] =  b
```

You can **multiply** a **matrix** by a **vector**,

```
In[19]:= m.z
Out[19] = {a p + b q, c p + d q}
```

or a **vector** by a **matrix**

```
In[20]:=  w.m
Out[20] = {a f + c g, b f + d g}
```

or their combination to **form a scalar**,

```
In[21]:= w.m.z
Out[21] =  (a f + c g) p + (b f + d g) q
```

Type another matrix,

```
In[22]:= n = {{a1, b1}, {c1, d1}}
```

Mathematica performs **addition, subtraction,** and **multiplication** of two **matrices.**

```
In[23]:= m + n
Out[23] = {{a + a1, b + b1}, {c + c1, d + d1}}

In[24]:= m - n
Out[24] = {{a - a1, b - b1}, {c - c1, d - d1}}

In[25]:= m.n
Out[25]={{a a1 + b c1, a b1 + b d1},{a1 c + c1 d, b1 c + d d1}}
```

Result can be printed in **matrix form** as,

```
In[26]:= MatrixForm[%]
Out[26] =
          a a1 + b c1   a b1 + b d1
          a1 c + c1 d   b1 c + d d1
```

IdentityMatrix

IdentityMatrix[n]

gives the $n \times n$ identity matrix.

Details and Options

- IdentityMatrix[{m, n}] gives the $m \times n$ identity matrix.
- IdentityMatrix by default creates a matrix containing exact integers.
- The option WorkingPrecision can be used to specify the precision of matrix elements.
- IdentityMatrix[n, SparseArray] gives the identity matrix as a SparseArray object.

```
IdentityMatrix[3] // MatrixForm
```

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

```
IdentityMatrix[{3, 4}] // MatrixForm
```

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

Transpose

Transpose[*list*]
transposes the first two levels in *list*.

Transpose[*list*, {n_1, n_2, ...}]
transposes *list* so that the k^{th} level in *list* is the ${n_k}^{th}$ level in the result.

- Transpose gives the usual transpose of a matrix.
- Transpose[*m*] can be input as $m^{\intercal}$.
- $^{\intercal}$ can be entered as esc tr esc or \[Transpose].
- Acting on a tensor $T_{i_1 i_2 i_3 \ldots}$ Transpose gives the tensor $T_{i_2 i_1 i_3 \ldots}$. »
- Transpose[*list*, {n_1, n_2, ...}] gives the tensor $T_{i_{n_1} i_{n_2} \ldots}$.
- So long as the lengths of the lists at particular levels are the same, the specifications n_k do not necessarily have to be distinct.
- Transpose works on SparseArray objects.

```
In[1]:= m = {{a, b, c}, {x, y, z}};

In[2]:= MatrixForm[m]
```

Out[2]//MatrixForm= $\begin{pmatrix} a & b & c \\ x & y & z \end{pmatrix}$

```
In[3]:= Transpose[m] // MatrixForm
```

Out[3]//MatrixForm= $\begin{pmatrix} a & x \\ b & y \\ c & z \end{pmatrix}$

```
In[42]:= m = Array[a, {2, 3, 2}]

Out[42]= {{{a[1, 1, 1], a[1, 1, 2]}, {a[1, 2, 1], a[1, 2, 2]}, {a[1, 3, 1], a[1, 3, 2]}},
  {{a[2, 1, 1], a[2, 1, 2]}, {a[2, 2, 1], a[2, 2, 2]}, {a[2, 3, 1], a[2, 3, 2]}}}

In[43]:= m = Array[a, {2, 3, 2}] // MatrixForm
```

Out[43]//MatrixForm=

$$\begin{pmatrix} \begin{pmatrix} a[1,1,1] \\ a[1,1,2] \end{pmatrix} & \begin{pmatrix} a[1,2,1] \\ a[1,2,2] \end{pmatrix} & \begin{pmatrix} a[1,3,1] \\ a[1,3,2] \end{pmatrix} \\ \begin{pmatrix} a[2,1,1] \\ a[2,1,2] \end{pmatrix} & \begin{pmatrix} a[2,2,1] \\ a[2,2,2] \end{pmatrix} & \begin{pmatrix} a[2,3,1] \\ a[2,3,2] \end{pmatrix} \end{pmatrix}$$

```
In[44]:= Transpose[Array[a, {2, 3, 2}]] // MatrixForm
```

Out[44]//MatrixForm=

$$\begin{pmatrix} \begin{pmatrix} a[1,1,1] \\ a[1,1,2] \end{pmatrix} & \begin{pmatrix} a[2,1,1] \\ a[2,1,2] \end{pmatrix} \\ \begin{pmatrix} a[1,2,1] \\ a[1,2,2] \end{pmatrix} & \begin{pmatrix} a[2,2,1] \\ a[2,2,2] \end{pmatrix} \\ \begin{pmatrix} a[1,3,1] \\ a[1,3,2] \end{pmatrix} & \begin{pmatrix} a[2,3,1] \\ a[2,3,2] \end{pmatrix} \end{pmatrix}$$

Det

Det[m]
  gives the determinant of the square matrix m.

- Det[m, Modulus -> n] computes the determinant modulo n.

Find the determinant of a symbolic matrix:

In[8]:= Det$\left[\begin{pmatrix} a_{1,1} & a_{1,2} \\ a_{2,1} & a_{2,2} \end{pmatrix}\right]$

Out[8]= $-a_{1,2}\,a_{2,1}+a_{1,1}\,a_{2,2}$

In[8]:= Det$\left[\begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{pmatrix}\right]$

Out[8]= $-a_{1,3}\,a_{2,2}\,a_{3,1}+a_{1,2}\,a_{2,3}\,a_{3,1}+a_{1,3}\,a_{2,1}\,a_{3,2}-a_{1,1}\,a_{2,3}\,a_{3,2}-a_{1,2}\,a_{2,1}\,a_{3,3}+a_{1,1}\,a_{2,2}\,a_{3,3}$

Inverse

Inverse[m]
gives the inverse of a square matrix m.

Details and Options

- Inverse works on both symbolic and numerical matrices.
- For matrices with approximate real or complex numbers, the inverse is generated to the maximum possible precision given the input. A warning is given for ill-conditioned matrices.

Inverse of a 2×2 matrix:

```
In[1]:= Inverse[{{1.4, 2}, {3, -6.7}}]
Out[1]= {{0.435631, 0.130039}, {0.195059, -0.0910273}}
```

Enter the matrix in a grid:

In[1]:= $\texttt{Inverse}\left[\begin{pmatrix} 1 & 2 & 3 \\ 4 & 2 & 2 \\ 5 & 1 & 7 \end{pmatrix}\right]$

Out[1]= $\left\{\left\{-\frac{2}{7}, \frac{11}{42}, \frac{1}{21}\right\}, \left\{\frac{3}{7}, \frac{4}{21}, -\frac{5}{21}\right\}, \left\{\frac{1}{7}, -\frac{3}{14}, \frac{1}{7}\right\}\right\}$

Inverse of a symbolic matrix:

In[1]:= `Inverse[{{u, v}, {v, u}}]`

Out[1]= $\left\{\left\{\frac{u}{u^2 - v^2}, -\frac{v}{u^2 - v^2}\right\}, \left\{-\frac{v}{u^2 - v^2}, \frac{u}{u^2 - v^2}\right\}\right\}$

The inverse may not exist:

In[1]:= $\texttt{Inverse}\left[\begin{pmatrix} 1 & 2 \\ 1 & 2 \end{pmatrix}\right]$

··· Inverse: Matrix {{1, 2}, {1, 2}} is singular.

Out[1]= `Inverse[{{1, 2}, {1, 2}}]`

PseudoInverse

PseudoInverse[*m*]
finds the pseudoinverse of a rectangular matrix.

- PseudoInverse works on both symbolic and numerical matrices.
- For a square matrix, PseudoInverse gives the Moore–Penrose inverse.

For non-singular square matrices $\mathbf{M}$, the pseudoinverse $\mathbf{M}^{(-1)}$ is equivalent to the standard inverse.

A matrix has a pseudoinverse even if it is singular:

```
In[9]:= PseudoInverse[{{1, 2, 3}, {4, 5, 6}, {7, 8, 9}}]
```

Out[1]= $\left\{\left\{-\frac{23}{36}, -\frac{1}{6}, \frac{11}{36}\right\}, \left\{-\frac{1}{18}, 0, \frac{1}{18}\right\}, \left\{\frac{19}{36}, \frac{1}{6}, -\frac{7}{36}\right\}\right\}$

m is a 4×3 matrix:

```
In[2]:= m = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}, {10, 11, 12}};
```

Compute using exact arithmetic:

```
In[2]:= PseudoInverse[m]
```

Out[2]= $\left\{\left\{-\frac{29}{60}, -\frac{11}{45}, -\frac{1}{180}, \frac{7}{30}\right\}, \left\{-\frac{1}{30}, -\frac{1}{90}, \frac{1}{90}, \frac{1}{30}\right\}, \left\{\frac{5}{12}, \frac{2}{9}, \frac{1}{36}, -\frac{1}{6}\right\}\right\}$

Compute using machine arithmetic:

```
In[3]:= PseudoInverse[N[m]]

Out[3]= {{-0.483333, -0.244444, -0.00555556, 0.233333},
  {-0.0333333, -0.0111111, 0.0111111, 0.0333333}, {0.416667, 0.222222, 0.0277778, -0.166667}}
```

Compute the pseudoinverse for a random complex 3×2 matrix:

```
In[1]:= PseudoInverse[RandomComplex[1 + I, {3, 2}]]

Out[1]= {{0.53176 + 0.00122074 i, -0.711776 + 0.519616 i, 0.331812 - 0.966093 i},
  {-0.35849 - 0.323444 i, 0.973913 - 0.549196 i, -0.0211863 + 0.718957 i}}

In[67]:= t = RandomReal[1, 3]
  r = 3 + 14 t + 1.5 RandomReal[{-1, 1}, 3]

Out[67]= {0.0693972, 0.658357, 0.794431}

Out[68]= {4.13827, 13.1404, 13.9084}

In[69]:= Transpose[{t, r}]

Out[69]= {{0.0693972, 4.13827}, {0.658357, 13.1404}, {0.794431, 13.9084}}
```

Matrices may also be constructed using the Table[ ] command,

```
In[76]:= mat = Table[2 / (i + j), {i, 3}, {j, 3}]
```

Out[76]= $\left\{\left\{1, \frac{2}{3}, \frac{1}{2}\right\}, \left\{\frac{2}{3}, \frac{1}{2}, \frac{2}{5}\right\}, \left\{\frac{1}{2}, \frac{2}{5}, \frac{1}{3}\right\}\right\}$

```
In[77]:= invmat = Inverse[mat]

Out[77]= {{36, -120, 90}, {-120, 450, -360}, {90, -360, 300}}
```

To confirm the obtained inverse, multiply it by the original matrix.

```
In[78]:= invmat.mat

Out[78]= {{1, 0, 0}, {0, 1, 0}, {0, 0, 1}}
```

Eigenvalues of a Matrix

To find the **eigenvalues** of the matrix, u may solve the characteristic equation.

In[82]:= eigenmat = mat - x IdentityMatrix[3]

Out[82]= $\left\{\left\{1-x, \frac{2}{3}, \frac{1}{2}\right\}, \left\{\frac{2}{3}, \frac{1}{2}-x, \frac{2}{5}\right\}, \left\{\frac{1}{2}, \frac{2}{5}, \frac{1}{3}-x\right\}\right\}$

In[85]:= deter = Det[eigenmat]

Out[85]= $\frac{1-786\,x+9900\,x^2-5400\,x^3}{5400}$

In[86]:= NSolve[deter == 0, x]

Out[86]= $\{\{x \to 0.00129332\}, \{x \to 0.0818099\}, \{x \to 1.75023\}\}$

Mathematica can find these eigenvalues directly as:

```
Eigenvalues[N[mat]]
```

```
{1.75023, 0.0818099, 0.00129332}
```

Eigenvalues

Eigenvalues[*m*]
gives a list of the eigenvalues of the square matrix *m*.

Eigenvalues[{*m*, *a*}]
gives the generalized eigenvalues of *m* with respect to *a*.

Eigenvalues[*m*, *k*]
gives the first *k* eigenvalues of *m*.

Eigenvalues[{*m*, *a*}, *k*]
gives the first *k* generalized eigenvalues.

- Eigenvalues finds numerical eigenvalues if m contains approximate real or complex numbers.
- Repeated eigenvalues appear with their appropriate multiplicity.
- An $n \times n$ matrix gives a list of exactly n eigenvalues, not necessarily distinct.
- If they are numeric, eigenvalues are sorted in order of decreasing absolute value.
- The eigenvalues of a matrix m are those λ for which $m.v = \lambda\, v$ for some nonzero eigenvector v.
- The generalized eigenvalues of m with respect to a are those λ for which $m.v = \lambda\, a.v$.

Machine-precision numerical eigenvalues:

```
In[1]:= Eigenvalues[Table[N[1 / (i + j + 1)], {i, 3}, {j, 3}]]
Out[1]= {0.657051, 0.0189263, 0.000212737}
```

Approximate 20-digit precision eigenvalues:

```
In[1]:= Eigenvalues[Table[N[1 / (i + j + 1), 20], {i, 3}, {j, 3}]]
Out[1]= {0.65705142829757924544, 0.018926310974070914307, 0.00021273691882603073422}
```

Exact eigenvalues:

```
In[1]:= Eigenvalues[Table[1/(i+j+1), {i, 2}, {j, 2}]]
```

Out[1]= $\left\{\frac{1}{60}\left(16+\sqrt{241}\right), \frac{1}{60}\left(16-\sqrt{241}\right)\right\}$

Symbolic eigenvalues:

```
In[1]:= Eigenvalues[{{a, b}, {c, d}}]
```

Out[1]= $\left\{\frac{1}{2}\left(a+d-\sqrt{a^2+4\,b\,c-2\,a\,d+d^2}\right), \frac{1}{2}\left(a+d+\sqrt{a^2+4\,b\,c-2\,a\,d+d^2}\right)\right\}$

```
In[2]:= CharacteristicPolynomial[{{a, b}, {c, d}}, x]
```

Out[2]= $-b\,c+a\,d-a\,x-d\,x+x^2$

```
In[3]:= Solve[% == 0, x]
```

Out[3]= $\left\{\left\{x\to\frac{1}{2}\left(a+d-\sqrt{a^2+4\,b\,c-2\,a\,d+d^2}\right)\right\}, \left\{x\to\frac{1}{2}\left(a+d+\sqrt{a^2+4\,b\,c-2\,a\,d+d^2}\right)\right\}\right\}$

CharacteristicPolynomial

CharacteristicPolynomial[*m*, *x*]
gives the characteristic polynomial for the matrix *m*.

CharacteristicPolynomial[{*m*, *a*}, *x*]
gives the generalized characteristic polynomial with respect to *a*.

- *m* must be a square matrix.
- It can contain numeric or symbolic entries.
- CharacteristicPolyonomial[*m*, *x*] is essentially equivalent to Det[*m* - *id* *x*] where *id* is the identity matrix of appropriate size. »
- CharacteristicPolynomial[{*m*, *a*}, *x*] is essentially Det[*m* - *a* *x*]. »

```
In[1]:= CharacteristicPolynomial[{{a, b}, {c, d}}, x]
```

Out[1]= $-b\,c + a\,d - a\,x - d\,x + x^2$

```
In[2]:= Det[{{a, b}, {c, d}} - x IdentityMatrix[2]]
```

Out[2]= $-b\,c + a\,d - a\,x - d\,x + x^2$

Find the eigenvalues of a matrix as the roots of the characteristic polynomial:

```
In[1]:= m = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}};

In[2]:= Solve[CharacteristicPolynomial[m, x] == 0, x]
```

Out[2]= $\left\{\{x \to 0\}, \left\{x \to \frac{3}{2}\left(5 - \sqrt{33}\right)\right\}, \left\{x \to \frac{3}{2}\left(5 + \sqrt{33}\right)\right\}\right\}$

```
In[3]:= Eigenvalues[m]
```

Out[3]= $\left\{\frac{3}{2}\left(5 + \sqrt{33}\right), \frac{3}{2}\left(5 - \sqrt{33}\right), 0\right\}$

The generalized characteristic polynomial is equivalent to Det[*m* - *a x*]:

```
In[1]:= {m, a} = RandomInteger[9, {2, 5, 5}];

In[2]:= CharacteristicPolynomial[{m, a}, x]
```

Out[2]= $-783 + 79x - 2671x^2 + 16020x^3 - 26327x^4 + 9390x^5$

```
In[3]:= Det[m - a x]
```

Out[3]= $-783 + 79x - 2671x^2 + 16020x^3 - 26327x^4 + 9390x^5$