Matlab and similar matrix handlers

Bernt Arne Ødegaard

15 March 2023

1 Matrix handlers (matlab, scilab, octave, etc)

Purpose: Manipulating numerical relationships using notation similar to linear algebra.
   Usually work on collections of numbers, represented by vectors and/or matrices.
   Name the objects of interest (vectors/matrices) and specify manipulations of these.
   Create a vector:

```
v=[1 ; 1]
v =
   1
   1
```

Create a matrix:

```
>> A = [1 1;0 1]
A =
   1   1
   0   1
```

Multiply the two

```
>> A*v
ans =
   2
   1
```

2 Linear Algebra

2.1 Basic matrix operations

Defining variables   The basic variables we consider are numbers, vectors and matrices. The type of each variable is determined as it is defined:

```
> a=1
a = 1
> b=2
b = 2
> c=3
c = 3
> y=[1;2;3]
y =
   1
```

```
    2
    3
> x=[1,2,3]
x =
   1  2  3
> A=[1 2 3;4 5 6]
A =
   1  2  3
   4  5  6
```

A space or a comma signifies a new number, a semicolon a new row. To suppress the printing of what is defined, end the line with a semicolon:

```
> A=[1,2,3,4];
> A=[1,2,3,4]
A =
   1  2  3  4
```

One can also use defined variables to define new variables, as long as the dimensions make sense. For example, given the above definitions:

```
>B=[c x]
B =
   3  1  2  3
> C = [A;x]
C =
   1  2  3
   4  5  6
   1  2  3
> D = [A y]
error: number of rows must match
error: evaluating assignment expression near line 22, column 3
```

If the dimensioning is wrong, an error message is outpu, and the variable is not defined.

To see what is in a variable, print the value by giving its name:

```
> a
a = 1
> b
b = 2
> A
A =
   1  2  3
   4  5  6
> B
B =
   5  1  2  3
```

Note that definitions are case-sensitive, both `A` and `a` are defined.

2.1.1 Arithmetic Matrix Operations

Let us look at how matrix arithmetic is performed
  Addition and subtraction.
  Numbers

```
> a=1;
> b=2;
> c=3;
> a+b
ans = 3
> b+c
ans = 5
```

Vectors

```
> x=[1 2 3 4 ]
x =
  1  2  3  4
> y=[4 3 2 1]
y =
  4  3  2  1
> x+y
ans =
  5  5  5  5
> y-x
ans =
   3   1  -1  -3
```

Take linear combinations of vectors:

```
> a*x + b*y
ans =
  9  8  7  6
> 0.5*x -1.5*y
ans =
  -5.50000  -3.50000  -1.50000   0.50000
```

Matrices

```
> A=[1 2 3; 4 5 6];
> B=[6 5 4; 3 2 1];
> A+B
ans =
  7  7  7
  7  7  7
> A-B
ans =
  -5  -3  -1
   1   3   5
> a*A + b*B
ans =
  13  12  11
  10   9   8
```

Matrix Multiplication   When multiplying matrices, one must be more careful about dimensions, but otherwise it is just like vector multiplication.

```
> A=[1 2 3;4 5 6]
A =
  1  2  3
  4  5  6
> B = [1 2;3 4; 5 6]
B =
  1  2
  3  4
  5  6
> A*B
ans =
  22  28
  49  64
> B*A
ans =
   9  12  15
  19  26  33
  29  40  51
```

For these matrices, both **AB** and **BA** are defined operations, but note that the results are different, in fact, even the dimension of the product is different.

If we let **B** be a 2×2 matrix, then multiplying **AB** is an error.

```
> B=[1 2;3 4]
B =
  1  2
  3  4
>
error: nonconformant matrices (op1 is 2x3, op2 is 2x2)
> B*A
ans =
   9  12  15
  19  26  33
```

Matrix Operations   Transpose

The transpose of a matrix `a` is `a'`:

```
> A
A =
  1  2  3
  4  5  6
> A'
ans =
  1  4
  2  5
  3  6
```

Null and identity matrix

```
> null = zeros(3,3)
null =
  0  0  0
  0  0  0
```

```
  0  0  0
> ident = eye(3,3)
ident =
  1  0  0
  0  1  0
  0  0  1
```

Rank:

```
> A
A =
  1  2  3
  4  5  6
> rank(A)
ans = 2
```

Inverse

```
> D = [3 4;4 6]
D =
  3  4
  4  6
> inverse(D)
ans =
   3.0000  -2.0000
  -2.0000   1.5000
```

To make sure that this *is* the inverse, multiply D and `inverse(D)`:

```
> D * inverse(D)
ans =
  1  0
  0  1
```

Idempotent matrix

So that you can see that there really exists idempotent matrices, consider

$$\mathbf{F} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

```
> F = [0.5 0.5; 0.5 0.5]
F =
  0.50000  0.50000
  0.50000  0.50000
> F * F
ans =
  0.50000  0.50000
  0.50000  0.50000
```

Determinant

```
> B
B =
  1  2
  3  4
> det(B)
ans = -2
```

A Warning. Numerical errors.

One thing you should be aware of when using a matrix tool is that the underlying tools are floating point number routines. All these are subject to numerical round–off errors, that may aggregate, and give weird results. This is also a problem in doing numerical linear algebra, if a matrix is *ill-conditioned.* As an example, consider the matrix

```
> A
A =
   1  2  3
   3  2  1
   1  1  1
```

This matrix is singular, which you see if you take $\frac{1}{4}$ times the sum of the first two rows, which is equal to the third row. Since the matrix is singular, it does not have an inverse, and should have determinant equal to zero. Let us check.

```
> det(A)
ans =  -4.4409e-16
> inverse(A)
ans =
  -2251799813685248  -2251799813685246   9007199254740989
   4503599627370496   4503599627370494  -18014398509481980
  -2251799813685248  -2251799813685248   9007199254740992
> rank(A)
ans = 2
```

As you see, because of numerical inaccuracies, this particular implementaion does not "catch" that the determinant of **A** is actually *equal to* zero, and tries to invert **A** giving this extremely wrong estimate of the inverse. The algorithm for finding the rank of the matrix, on the other hand, is accurate enough to find the correct rank of 2. Taking determinants is thus useful as a diagnostic tool.[1]

Solving linear equations.

Remember the basic linear equation

$$\mathbf{Ax} = \mathbf{b}$$

We saw that this equation had a defined solution if the rank of **A** was the same as the rank of $[\mathbf{A}|\mathbf{b}]$. If **A** is nonsingular, we solve the linear equation by finding the unique solution

$$\mathbf{x} = \mathbf{A}^{-1}\mathbf{b}$$

Consider the linear equation

$$3x_1 + 4x_2 = 5$$
$$4x_1 + 6x_2 = 8$$

Solve this system of equations

Write this in matrix form by defining

$$\mathbf{A} = \left[\begin{array}{cc} 3 & 4 \\ 4 & 6 \end{array} \right]$$

$$b = \left[\begin{array}{c} 5 \\ 8 \end{array} \right]$$

Let us first check that this system is solvable

---

[1]There are actually better tools available to check the condition of a matrix, which you may want to look at if you are doing important computations.

```
> A = [3 4;4 6]
A =
  3  4
  4  6
> b=[5;8]
b =

  5
  8
> rank(A)
ans = 2
> rank ([A b])
ans = 2
```

Note how I create the augmented matrix $[A|b]$ by `[A b]`. The rank of the two is the same. Since **A** is square, we can calculate the solution as

```
> inverse(A)
ans =
   3.0000  -2.0000
  -2.0000   1.5000
> x = inverse(A) * b
x =
  -1
   2
```

The solution to the system of equations is

$$\mathbf{x} = \begin{bmatrix} -1 \\ 2 \end{bmatrix}$$

(In this case we calculated the solution by finding the inverse. But you should be aware that solving the system of equations by calculation of the inverse is not the numerically most stable way of doing the calculation. `Octave` has built in a direct linear matrix solver, which is invoked by the *left division* operator

```
> x = A\b
x =
  -1
   2
```

This solves the system of equations directly, and it is usually the way to do this operation, unless one needs the inverse for other purposes.)

3   Term Structure of Interest rates

The term structure of interest rates is a relationship between time to maturity and prices of dated future payments.

Example

   A two-year Treasury bond with a face value of 1000 and an annual coupon payment of 8% sells for 982.50. A one-year T bill, with a face value of 100, and no coupons, sells for 90. Compounding is discrete, annual.
   Given these market prices,

1. Find the prices $d(0,1)$ and $d(0,2)$ of one dollar received respectively one and two years from now.

2. Find the corresponding interest rates.

7

1. Discount factors (prices):
$$\begin{bmatrix} 982.50 = d(0,1)80 + d(0,2)1080 \\ 90 = d(0,1)100 \end{bmatrix}$$

Solving these equations we find prices
$$\begin{bmatrix} d(0,1) = 0.9 \\ d(0,2) = 0.84 \end{bmatrix}$$

2. and interest rates
$$\begin{bmatrix} r(0,1) = 11\% \\ r(0,2) = 9\% \end{bmatrix}$$

How is this example calculated with a matrix tool?
Define the vector of discount factors
$$\mathbf{d} = \begin{bmatrix} d(0,1) \\ d(0,2) \end{bmatrix}$$

the set of cash flows
$$\mathbf{C} = \begin{bmatrix} 80 & 1080 \\ 100 & 0 \end{bmatrix}$$

and the set of bond prices
$$\mathbf{B} = \begin{bmatrix} 982.50 \\ 90 \end{bmatrix}$$

The set of bond prices satisfies
$$\mathbf{B} = \mathbf{Cd}'$$

which means we can find $\mathbf{d}$ as
$$\mathbf{d}' = \mathbf{C}^{-1}\mathbf{B} = \begin{bmatrix} 80 & 1080 \\ 100 & 0 \end{bmatrix}^{-1} \begin{bmatrix} 982.50 \\ 90 \end{bmatrix} = \begin{bmatrix} 0.9 \\ 0.84 \end{bmatrix}$$

```
> C=[80 1080; 100 0 ]
C =
    80   1080
   100      0
2> B=[982.5;90]
B =
  982.500
   90.000
> d=inv(C)*B
d =
  0.90000
  0.84306
```

Example
You have estimated spot interest rates as follows:

Year	Spot Rate, Percent
1	$r(0,1) = 5.00$
2	$r(0,2) = 5.40$
3	$r(0,3) = 5.70$
4	$r(0,4) = 5.90$
5	$r(0,5) = 6.00$

Compounding is discrete, annual.

1. What are the discount factors for each date (that is, the present value of $1 paid in year t)?

2. What are the forward rates for each period?

3. Calculate the prices of the following Treasury notes. Assume they pay interest once a year, and have face values of 100.

    (a) 5 percent, 2-year note

    (b) 5 percent, 5-year note

    (c) 10 percent, 5-year note

4. Compare the duration of the two five year bonds.

Discount factors and forward prices

Year	$r(0,t)$	$d(0,t)$	$f(0, t-1, t)$
1	5.0	$\frac{1}{(1.05)} = 0.952$	5.0%
2	5.4	$\frac{1}{(1.054)^2} = 0.900$	$\frac{(1.054)^2}{(1.05)} - 1 = 5.8\%$
3	5.7	$\frac{1}{(1.057)^3} = 0.846$	$\frac{(1.057)^3}{(1.05)(1.058)} - 1 = 6.3\%$
4	5.9	$\frac{1}{(1.059)^4} = 0.795$	$\frac{(1.059)^4}{(1.05)(1.058)(1.063)} - 1 = 6.5\%$
5	6.0	$\frac{1}{(1.06)^5} = 0.747$	$\frac{(1.06)^5}{(1.05)(1.058)(1.063)(1.065)} - 1 = 6.4\%$

Prices of coupon bonds, assuming a face value of 100.

1. 5%, 2 years:
$$PV = 0.05 \cdot 0.952 + 1.05 \cdot 0.9 = 99.279$$

2. 5%, 5 years:
$$PV = 0.05 \cdot 0.952 + 0.05 \cdot 0.9 + 0.05 \cdot 0.846 + 0.05 \cdot 0.795 + 1.05 \cdot 0.74 = 95.934$$

3. 10%, 5 years
$$PV = 0.10 \cdot 0.952 + 0.10 \cdot 0.9 + 0.10 \cdot 0.846 + 0.10 \cdot 0.795 + 1.10 \cdot 0.747 = 117.14$$

Now, here is how this is done in a matrix tool:

Discount factors:

```
> r=[5 5.4 5.7 5.9 6]/100
r =
   0.050000   0.054000   0.057000   0.059000   0.060000
> t=1:5
t =
   1   2   3   4   5
> d=(1+r).^(-t)
d =
   0.95238   0.90016   0.84679   0.79509   0.74726
> f(1)=r(1);
> for i=2:5
> f(i) = ((1+r(i))^i)/((1+r(i-1))^(i-1))-1;
> endfor
f =
   0.050000   0.058015   0.063026   0.065023   0.064009
```

9

Calculating bond prices:

```
> b=[5 105 0 0 0]*d'
b = 99.279
> b=[5 5 5 5 105]*d'
b = 95.934
>b=[10 10 10 10 110]*d'
b = 117.14
```

Calculating duration

```
> C1=[5 5 5 5 105]
C1 =
     5    5    5    5  105
> B1=C1*d'
B1 = 95.934
> Dur1 = 1/B1 * (t.*C1)*d'
D1 = 4.5310
> C2=[10 10 10 10 110]
C2 =
    10   10   10   10  110
> B2 = C2*d'
B2 = 117.14
> Dur2 =  1/B2 * (t.*C2)*d'
Dur2 = 4.2318
```

Note the use of the element by element multiplication .* in this calculation.

Example

A 10%, two year bond is traded at a price of 90. The current one year spot rate is $r(0,1) = 12\%$ (with discrete, annual compounding).

1. Determine the duration and convexity of the bond, using both the full term structure and the Macaulay style calculations.

First need to find the two year spot rate

$$90 = 10d_1 + 110d_2$$

$$d_1 = \frac{1}{1 + 0.12} = 0.89286$$

$$d_2 = \frac{90 - 10d_1}{110} = 0.73701$$

$$r_2 = 0.16483$$

Here are some of the calculation in a matrix tool

```
> C=[10 110]
C =
    10  110
> r(1)=0.12
r =   0.12000
> d(1)=1/(1+r(1))
d =   0.89286
> d(2)=(90-10*d(1))/110
d =
```

```
   0.89286   0.73701
> d=d'
d =
   0.89286
   0.73701
> BondPrice=C*d
BondPrice = 90
> r(2)=d(2)^(-1/2)-1
r =
   0.12000   0.16483
> y = irr([-BondPrice C],0)
y = 0.16249x
> checkprice=C(1)/(1+y)+C(2)/(1+y)^2
checkprice = 90
```

We calculate duration using the two definitions

```
> Duration=1/BondPrice * ( 1*d(1)*C(1) + 2*d(2)*C(2))
Duration = 1.9008
> Duration=1/BondPrice * ( 1*C(1)/(1+y) + 2*C(2)/((1+y)^2) )
Duration = 1.9044
```

Using the term structure we find duration as
$$D = 1.9008$$
using the Macaulay definition we find
$$D = 1.9044$$

Thus, not a major difference.

We also calculate the convexity for the two definitions

```
> Cx=1/BondPrice * 1/(1+y)^2 * ( (1+1)*d(1)*C(1) + (2+2^2)*d(2)*C(2))
Cx = 4.1463
> Cx=1/BondPrice * 1/(1+y)^2 * ( (1+1)*C(1)/(1+y) + (2+2^2)*C(2)/(1+y)^2)
Cx = 4.1570
```

Again, not a major difference with the two methods of calculating

Consider a second version of the matrix calculations, with slighly more compact use of notation in the calculations.

```
> C=[10 110]
C =
    10   110
> r=[0.12]
r = 0.12000
> d=(1+r)^(-1)
d = 0.89286
> d(2)=(90-10*d(1))/110
d =
   0.89286   0.73701
> r=d.^(-1./t)-1
r =
  0.12000   0.16483
> B=C*d'
```

```
B = 90
> y=irr([-B C])
y = 0.16249
8> D=1/B*(t.*C)*d'
D = 1.9008
> DMc = 1/B* (t.*C)*((1+y).^(-t))'
DMc = 1.9044
> Cx = 1/B*(1/(1+y))*((t+t.^2).*C)*d'
Cx = 4.8200
> Cx = 1/B*(1/(1+y))*((t+t.^2).*C)*((1+y).^(-t))'
Cx = 4.8324
```