MATH 100 – Introduction to the Profession

Matrices and Linear Transformations in MATLAB

Greg Fasshauer

Department of Applied Mathematics
Illinois Institute of Technology

Fall 2011

Basic Definitions of Vectors and Matrices

We go over some basic matrix and vector stuff in MATLAB:

- `matrix_vector.m` (definition of matrices and vectors)
- `arithmetic.m` (simple arithmetic with matrices and vectors)
- `lin_sys.m` (solving linear systems)
- `submatrices.m` (definition of submatrices)

Vector Equation of a Line

An arbitrary point $\boldsymbol{r}$ on a line through the point $\boldsymbol{r}_0$ with direction vector $\boldsymbol{v}$ is given by

$$\boldsymbol{r} = \boldsymbol{r}_0 + t\boldsymbol{v},$$

where the parameter t tells us how much of, and which direction, the vector $\boldsymbol{v}$ is added to $\boldsymbol{r}_0$.

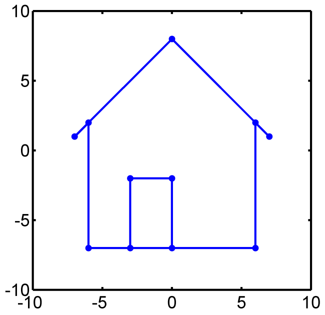
Look at the Mathematica demo `EquationOfALineInVectorForm2D.cdf`.

More details on vectors and equations of lines in 2D and 3D are given in [Stewart Calculus, Sections 12.2 and 12.5].

Matrices as Linear Transformations

We illustrate properties of linear transformations (matrix multiplication by A) with the following “data”:

```
X = house
dot2dot(X)
```

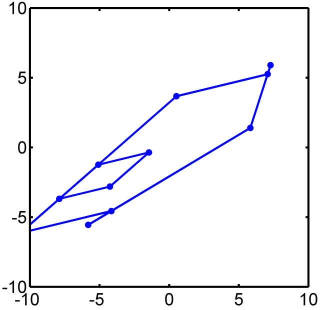


Straight lines are always mapped to straight lines.

```
A = rand(2,2)
dot2dot(A*X)
```

Sample matrix

$$A = \begin{bmatrix} 0.9357 & 0.7283 \\ 0.8187 & 0.1758 \end{bmatrix}$$

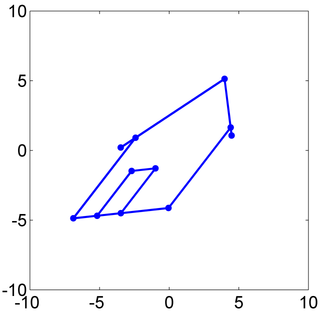


The transformation is orientation-preserving[1] if $\det A > 0$.

```
A = rand(2,2)
det(A)
dot2dot(A*X)
```

Sample matrix

$$A = \begin{bmatrix} 0.5694 & 0.4963 \\ 0.0614 & 0.6423 \end{bmatrix}$$



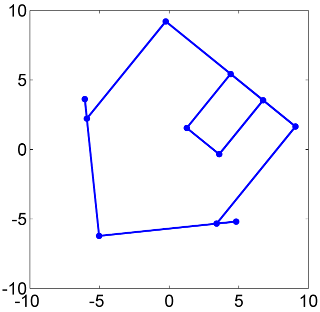
[1] The door always stays on the left.

The angles between straight lines are preserved if the matrix is orthogonal[2].

```
A = orth(rand(2,2));   % creates orthogonal matrix
A = A(:,randperm(2))   % randomly permute columns of A
det(A)
dot2dot(A*X)
```

Sample matrix

$$A = \begin{bmatrix} -0.7767 & -0.6299 \\ 0.6299 & -0.7767 \end{bmatrix}$$



[2] An orthogonal matrix A has $\det A \pm 1$ and represents either a rotation or a reflection.

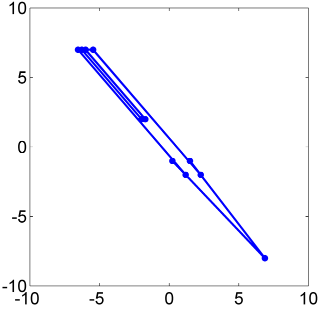
A linear transformation is invertible[3] only if $\det A \neq 0$.

```
a22 = randi(3,1,1)-2  % creates random {-1,0,1}
A = triu(rand(2,2)); A(2,2) = a22
det(A)
dot2dot(A*X)
```

Sample matrix

$$A = \begin{bmatrix} 0.0903 & 0.8586 \\ 0 & -1.0000 \end{bmatrix}$$

$$\det A = -0.0903$$



[3] If the transformation is not invertible, then the 2D image collapses to a line or even a point.

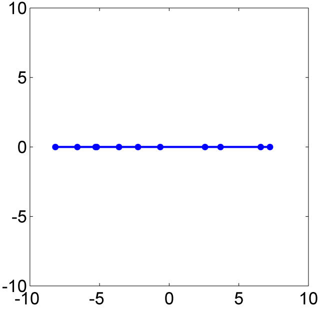
A linear transformation is invertible[3] only if $\det A \neq 0$.

```
a22 = randi(3,1,1)-2  % creates random {-1,0,1}
A = triu(rand(2,2)); A(2,2) = a22
det(A)
dot2dot(A*X)
```

Sample matrix

$$A = \begin{bmatrix} 0.9884 & 0.3209 \\ 0 & 0 \end{bmatrix}$$

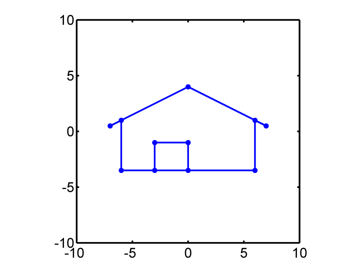
$$\det A = 0$$



[3]If the transformation is not invertible, then the 2D image collapses to a line or even a point.

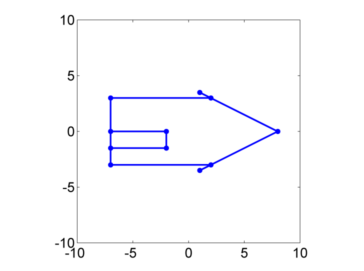
A diagonal matrix stretches the image or reverses its orientation.

$$A = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{2} \end{bmatrix}, \quad \det A = \frac{1}{2}$$



A anti-diagonal matrix in addition interchanges coordinates.

$$A = \begin{bmatrix} 0 & 1 \\ \frac{1}{2} & 0 \end{bmatrix}, \quad \det A = -\frac{1}{2}$$



The action of a diagonal matrix provides an interpretation of the effect of eigenvalues. Note that these matrices have orthogonal columns, but their determinant is not ± 1, so they are **not** orthogonal matrices. These matrices preserve right angles only.

Any rotation matrix can be expressed in terms of trigonometric functions:
The matrix

$$G(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

represents a counter-clockwise rotation by the angle θ (measured in radians).

Look at `wiggle.m`.

Look through `matrices_recap.m`.

References I

T. A. Driscoll.
Learning MATLAB.
SIAM, Philadelphia, 2009.
http://epubs.siam.org/ebooks/siam/other_titles_in_applied_mathematics/ot115

D. J. Higham and N. J. Higham.
MATLAB Guide (2nd ed.).
SIAM, Philadelphia, 2005.
http://epubs.siam.org/ebooks/siam/other_titles_in_applied_mathematics/ot92

C. Moler.
Numerical Computing with MATLAB.
SIAM, Philadelphia, 2004.
http://www.mathworks.com/moler/index_ncm.html

References II

C. Moler.
Experiments with MATLAB.
Free download at
http://www.mathworks.com/moler/exm/chapters.html

J. Stewart.
Calculus (7th ed.).
Brooks/Cole, 2011.

The MathWorks.
MATLAB 7: Getting Started Guide.
http://www.mathworks.com/access/helpdesk/help/pdf_doc/matlab/getstart.pdf