

Visualizing Linear Transformations in R^2

Worksheet by Michael K. May, S.J. and Russell Blyth

```
> restart: with(LinearAlgebra): with(plots): with(plottools):  
Warning, the name changecoords has been redefined  
Warning, the assigned name arrow now has a global binding
```

Outline

The basic objectives are:

- 1) Learn how to get Maple to draw simple stick figures in R^2 , and how to do simple translations of the figures.
- 2) Learn to represent a linear transformation in R^2 as multiplication by a matrix.

```
>
```

Drawing Stick Figures and Translation of Objects

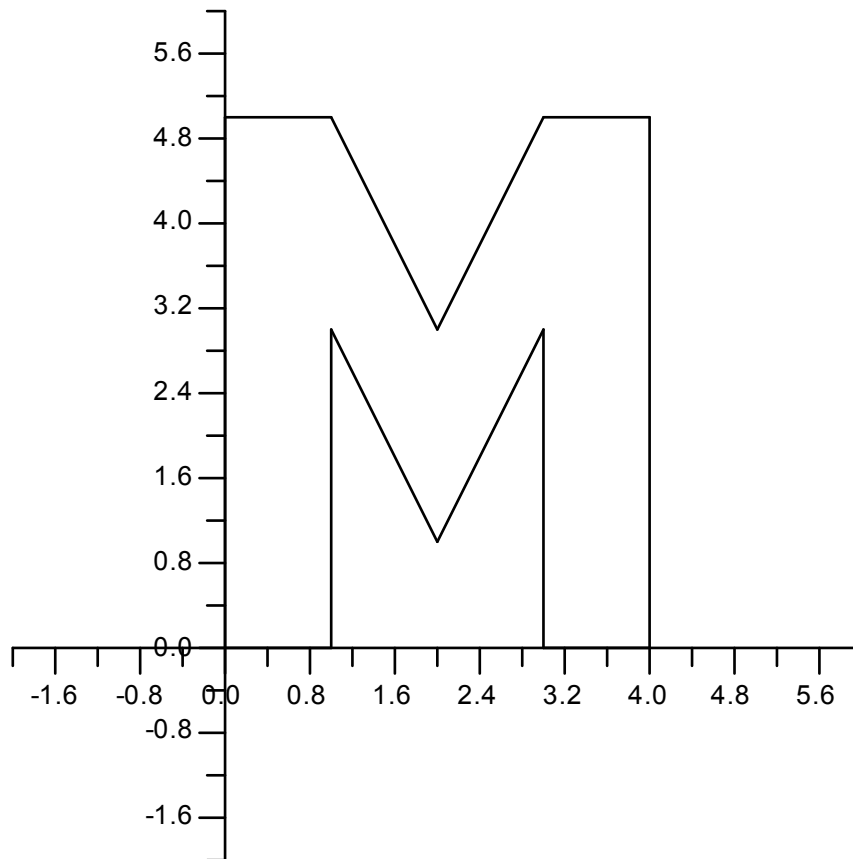
To see the action of linear transformations on objects, we first want to be able to easily draw objects, which we do by specifying outlines. In this worksheet we work with figures in 2 dimensions. The techniques will be extended in the next worksheet to look at figures in 3 dimensions. We draw a polygon by plotting a list of points that outline the shape we are drawing.

We start with a set of points that outline the letter M.

```
> mlist := [<0,0>, <0,5>, <1,5>, <2,3>, <3,5>, <4,5>, <4,0>, <3,  
0>, <3,3>, <2,1>, <1,3>, <1,0>, <0,0>];  
mlist :=  $\begin{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 5 \end{bmatrix}, \begin{bmatrix} 1 \\ 5 \end{bmatrix}, \begin{bmatrix} 2 \\ 3 \end{bmatrix}, \begin{bmatrix} 3 \\ 5 \end{bmatrix}, \begin{bmatrix} 4 \\ 5 \end{bmatrix}, \begin{bmatrix} 4 \\ 0 \end{bmatrix}, \begin{bmatrix} 3 \\ 0 \end{bmatrix}, \begin{bmatrix} 3 \\ 3 \end{bmatrix}, \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ 3 \end{bmatrix}, \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \end{bmatrix}$  (2.1)
```

If we simply plot the points we see that they form a large block M in a 5 by 4 square with legs having thickness one. (Notice that the ending point repeats the starting point to close the outline.)

```
> pointplot(mlist, view=[-2..6, -2..6], connect=true, axes=normal);
```



>

▼ Exercise:

1) Create three lists of points that define a block polygon descriptions of three distinct letters in your last name (at this point you may want to avoid letter like A and R that would need two polygons to describe them). Have Maple draw the letters in separate plots.

>
>
>
>

Once we have plotted the outline of a polygon, we want to be able to systematically make modifications to our list of points and see how the changes modify the pictures. The most obvious transformation to start with is a translation. We first define a function to carry out a translation:

```
> AddVectorToList :=(vectorToAdd, listOfVectors)->map(Add,
  listOfVectors, vectorToAdd):
```

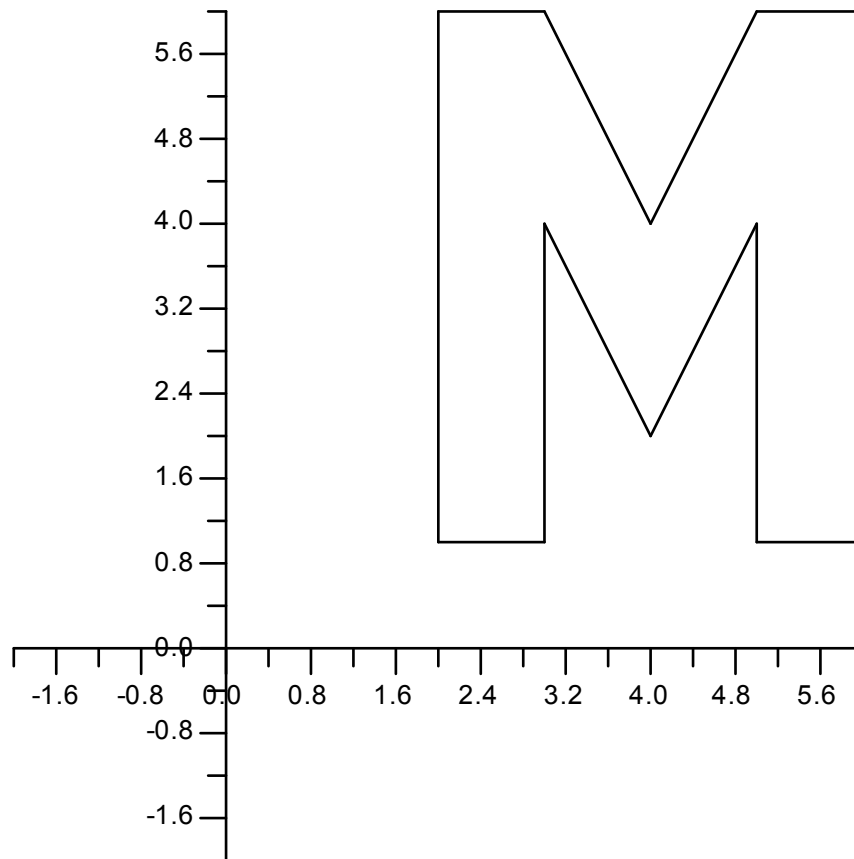
We can use this command to move the corners of M right 2 and up 1.

```
> mlist;
mlist2 := AddVectorToList(<2,1>,mlist);
pointplot(mlist2, view=[-2..6, -2..6],connect=true, axes=
```

```

normal);
[[ [0] [0] [1] [2] [3] [4] [4] [3] [3] [2] [1] [1] [0] ]
  [0] [5] [5] [3] [5] [5] [0] [0] [3] [1] [3] [0] [0] ]
mlist2 := [[ [2] [2] [3] [4] [5] [6] [6] [5] [5] [4] [3] [3] [2] ]
            [1] [6] [6] [4] [6] [6] [1] [1] [4] [2] [4] [1] [1] ]

```

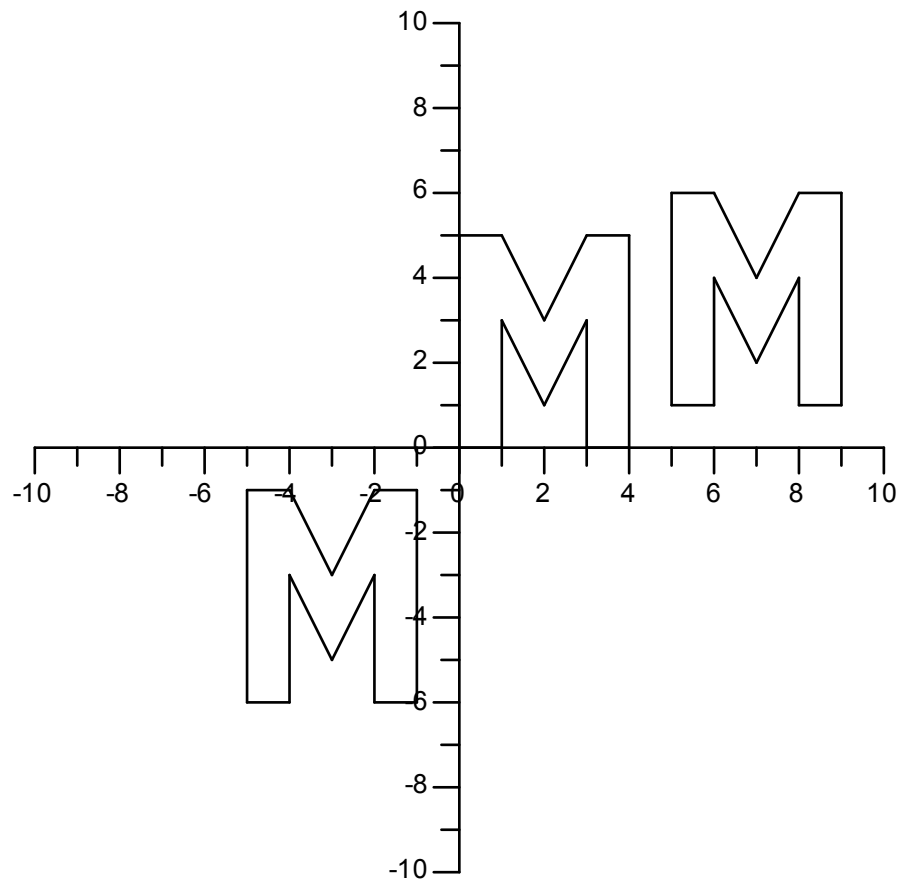


To plot several letters together, we use the display command. First we create named plots of each letter. Then we display the list of plots. (Note that when we name a plot, we end the command with a colon rather than a semicolon, so that we don't get a printout of the formal plot structure.)

```

> m1 := pointplot(mlist,connect=true):
   m2 := pointplot(AddVectorToList(<5,1>,mlist),connect=true):
   m3 := pointplot(AddVectorToList(<-5,-6>,mlist),connect=true):
   display({m1,m2,m3},view=[-10..10,-10..10], axes=normal);

```



Exercises:

2) Use the lists created in exercise 1 above to plot the first three letters of your last name in a single plot.

```
>
```

3) Replot the same letters, but put them in a more interesting arrangement this time. (They should not be in a flat line.)

```
>
```

4) Translate the letters you plotted in exercise 3 so that they are balanced around the origin.

```
>
```

```
>
```

Linear Transformations in R^2

In section 3.1 of our text we learn that mappings of vector spaces that satisfy the linearity properties can be represented as multiplication by a matrix. We follow the text's convention, treating vectors as column vectors so that the matrix is always on the left and the vector is on the right.

```
> A := <<1,-2> | <0,1>>;
```

```
b := <1,3>;
"A b " =A.b;
```

$$A := \begin{bmatrix} 1 & 0 \\ -2 & 1 \end{bmatrix}$$

$$b := \begin{bmatrix} 1 \\ 3 \end{bmatrix}$$

$$\text{"A b"} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

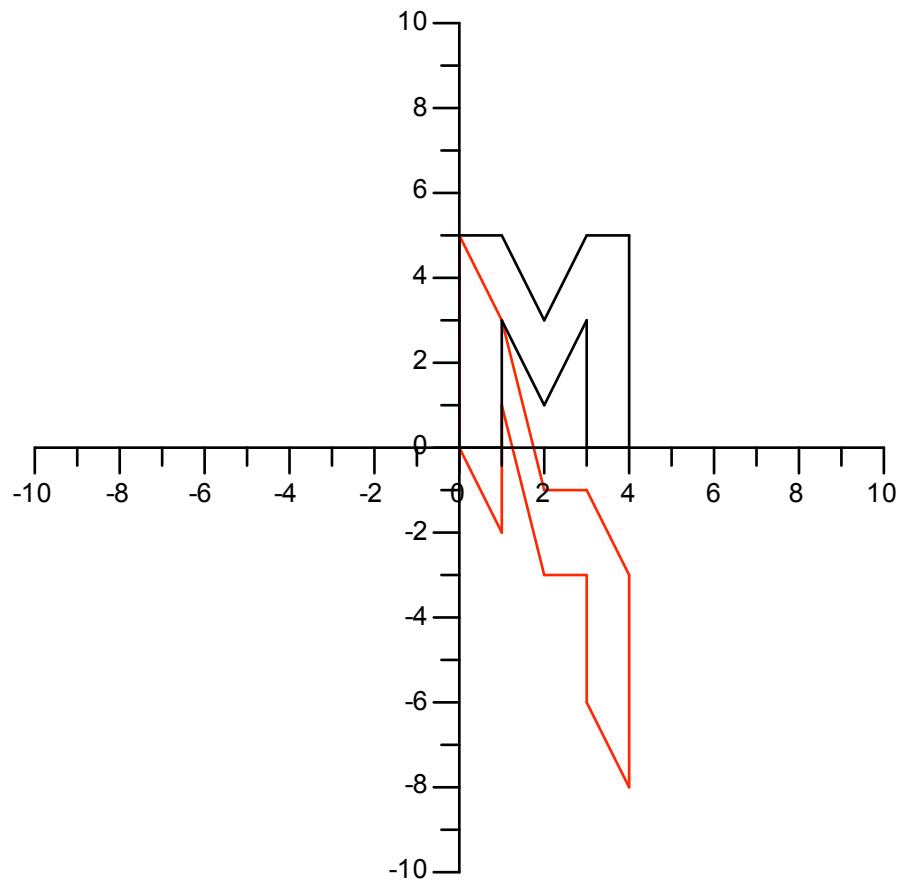
(3.1)

To see how multiplication by a matrix changes a figure, we apply the matrix to a list of vectors.

```
> multmatbylist :=(multmat, listofvecs)->
  map((x,y)-> y.x,listofvecs,multmat):
```

```
> mlistA := multmatbylist(A,mlist);
plota := pointplot(mlist,color=black,connect=true):
plotb := pointplot(mlistA,color=red,connect=true):
display({plota, plotb}, view=[-10..10,-10..10],axes=normal);
```

$$mlistA := \left[\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 5 \end{bmatrix}, \begin{bmatrix} 1 \\ 3 \end{bmatrix}, \begin{bmatrix} 2 \\ -1 \end{bmatrix}, \begin{bmatrix} 3 \\ -1 \end{bmatrix}, \begin{bmatrix} 4 \\ -3 \end{bmatrix}, \begin{bmatrix} 4 \\ -8 \end{bmatrix}, \begin{bmatrix} 3 \\ -6 \end{bmatrix}, \begin{bmatrix} 3 \\ -3 \end{bmatrix}, \begin{bmatrix} 2 \\ -3 \end{bmatrix}, \begin{bmatrix} 1 \\ 1 \end{bmatrix}, \begin{bmatrix} 1 \\ -2 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right]$$



The transformation produced by multiplication by A can either be described by saying that it is a shear in the y direction by a factor of -2 (a holistic geometric description), or by saying that the first element of the standard basis is taken to $[1, -2]$ and the second element of the standard basis is taken to $[0, 1]$.

An important example discussed in the text is rotation around the origin by angle α radians. We see that this rotation takes $[1, 0]$ to $[\cos(\alpha), \sin(\alpha)]$ and takes $[0, 1]$ to $[-\sin(\alpha), \cos(\alpha)]$. This allows us to produce a matrix that can be used for the rotation (these two images are placed into the matrix as its columns).

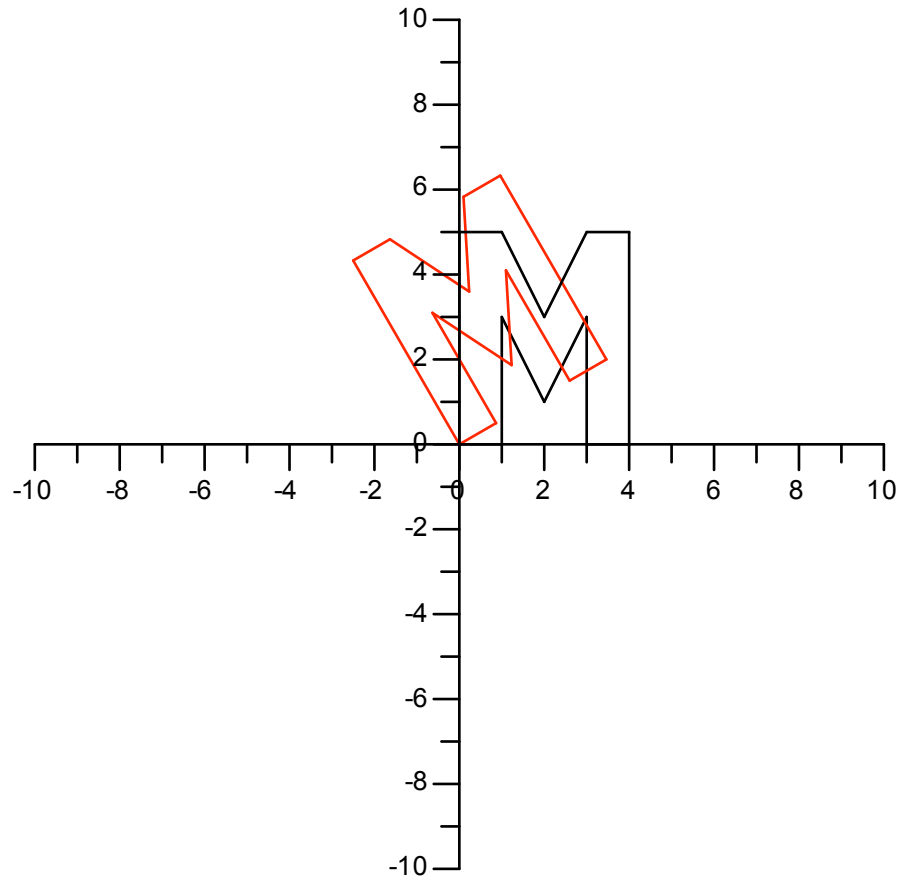
```
> rotmat := alpha -> <<cos(alpha), sin(alpha)> | < -sin(alpha), cos(alpha)>>;
Rot30 := rotmat(Pi/6);
```

$rotmat := \alpha \rightarrow \langle \langle \cos(\alpha), \sin(\alpha) \rangle \mid \langle -\sin(\alpha), \cos(\alpha) \rangle \rangle$

$$Rot30 := \begin{bmatrix} \frac{1}{2}\sqrt{3} & -\frac{1}{2} \\ \frac{1}{2} & \frac{1}{2}\sqrt{3} \end{bmatrix}$$

(3.2)

```
> plota := pointplot(mlist, color=black,connect=true):
plotb := pointplot(multmatbylist(Rot30, mlist), color=red,
connect=true):
display({plota, plotb}, view=[-10..10,-10..10],axes=normal,
scaling=constrained);
```



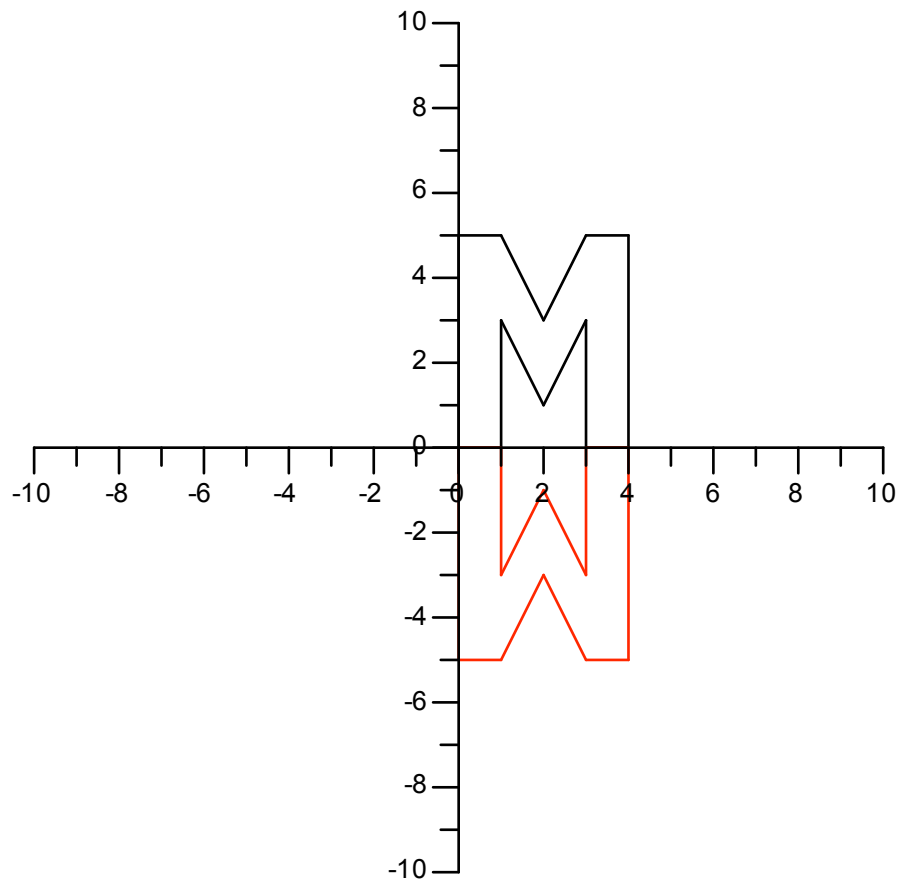
Exercise:

5) Find matrices to achieve the following actions. Verify the actions by applying them to the three letter combinations used in exercise 4 above:

MatA flips the letters upside down (reflection in x-axis).

```
> RF := <<1,0>|<0,-1>>;
plota := pointplot(mlist, color=black,connect=true):
plotb := pointplot(multmatbylist(RF, mlist), color=red,
connect=true):
display({plota, plotb}, view=[-10..10,-10..10],axes=normal,
scaling=constrained);
```

$$RF := \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$



MatB flips the letters right-to-left (reflection in y-axis).

>

MatC rotates the object by an angle of $2\pi/3$ (around the origin)

>

MatD shears along the y axis by a factor of 2.

>

>

>