Visualization of Vectors and Span in R^2 and R^3

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

Expanded by Regina, Hugh and Harry --- Draft 2

```
> 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 the basic mechanics of entering vectors, and producing linear combinations with either addition or scalar multiplication.

2) Learn to plot a set of vectors in R^2 and R^3.

3) Visualize the effects of multiplication by scalar and of addition of vectors in R^2 and R^3

4) Using a random number generator, see what typical linear combinations of a pair of vectors look like.

5) See the effect of linear transformations on the linear combination of vectors

6) Apply these concepts to understand and visualize the parametric description of a line and a plane in R^3

1. Vectors in R^2 and R^3:

The easiest way to enter a vector in R^2 and R^3 is as a list with angle brackets. In Maple you separate the coordinates with commas for a column vector, and with vertical bars (|) for a row vector. The whole vector is surrounded the with angle brackets (< and >).

```
> v1 := < 1, 1 >; v2 := < 1, 3 >;
```

$$v1 := \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$v2 := \begin{bmatrix} 1 \\ 3 \end{bmatrix} \qquad (2.1)$$

```
> w1 := < 1| 1| 1 >; w2 := < -1| 3| 2 >;
```

$$w1 := \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

$$w2 := \begin{bmatrix} -1 & 3 & 2 \end{bmatrix} \qquad (2.2)$$

When vectors are entered this way we use normal mathematics notation to add two vectors or to multiply by a scalar.

```
2 * v1;
```

```
v1 + v2;
```

```
2 * w1 + 3 * w2;
```

$$\begin{bmatrix} -1 & 11 & 8 \end{bmatrix} \qquad (2.5)$$

$\begin{bmatrix} 2 \\ 2 \end{bmatrix}$ (2.3)	$\begin{bmatrix} 2 \\ 4 \end{bmatrix}$ (2.4)	(2.5)

Mentally check the computations Maple is doing.

Notice that vectors need to have the same length before we can add them:

```
> <1,2> + <3,4,5>;
Error, (in rtable/Sum) invalid arguments
```

We can also enter vectors in Maple with the Vector command, which is part of the LinearAlgebra package.

> $a1 :=$ *Vector*([1, 1]); $a1 := \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ (2.6)	> $a2 :=$ *Vector*([1, 3]); $a2 := \begin{bmatrix} 1 \\ 3 \end{bmatrix}$ (2.7)	> $a1 + a2$; $\begin{bmatrix} 2 \\ 4 \end{bmatrix}$ (2.8)	> $3 * a1$; $\begin{bmatrix} 3 \\ 3 \end{bmatrix}$ (2.9) >

Mentally check the computations Maple is doing.

Exercises:

1.1) Use the last 4 digits of your telephone number to create two vectors u1 and u2 in R^2. Use Maple to compute the linear combination 2*u1 + 3*u2. (Be sure to label answers to all exercises. You can either add a comment like "The answer is ..." to the Maple worksheet, or write a comment on your printout.)

>

2.1) Pick six integers from -10 to 10 (repetitions are allowed) to create two distinct nonzero vectors z1 and z2 in R^3. Use Maple to compute 1.0*z1 + 2.0*z2. Compare this to z1+2*z2.

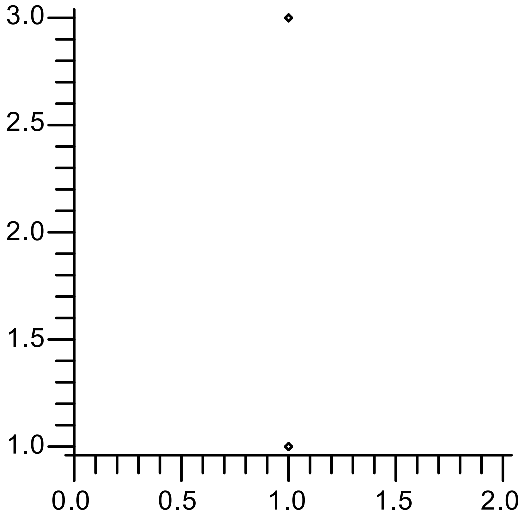
>

2. Plotting Lists of Points:

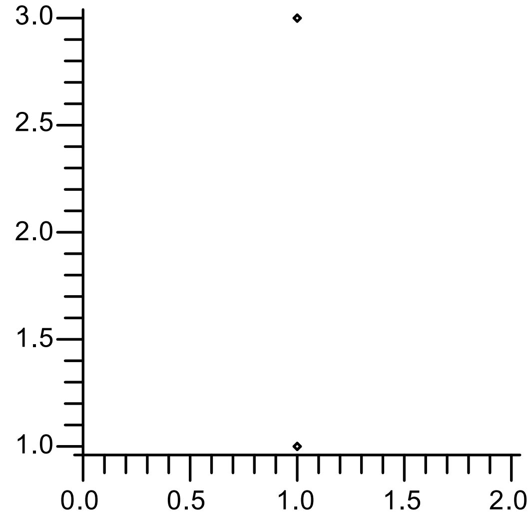
We plot points representing vectors with the command pointplot, which is part of the plot package.

$v1 := \langle 1, 1 \rangle;\ v2 := \langle 1, 3 \rangle;$ $v1 := \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ $v2 := \begin{bmatrix} 1 \\ 3 \end{bmatrix}$ (3.1)	$a1 :=$ *Vector*([1, 1]); $a2 :=$ *Vector*([1, 3]); $a1 := \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ $a2 := \begin{bmatrix} 1 \\ 3 \end{bmatrix}$ (3.2)

```
pointplot({v1, v2});
```

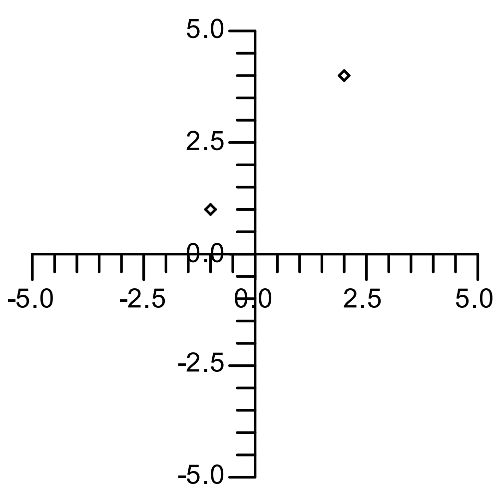



```
pointplot([a1, a2]);
```

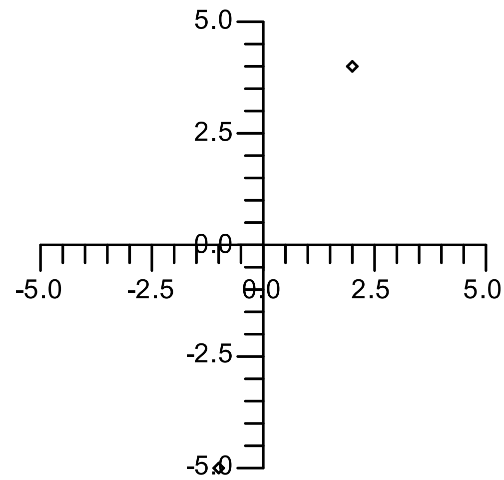


Notice that we can plot either a set of points (sets are enclosed in curly braces and are unordered) or a list of points (lists are ordered and enclosed in square brackets). When plotting, you may want to use the view option to specify the viewing window of the plot. For the two plots above, letting x and y both range from -5 to 5 is convenient. You can also specify a symbol size to make the points easier to see.

```
pointplot({v1 + v2, v2-2 * v1}, view = [-5 ..5,-5 ..5], symbolsize = 15);
```




```
pointplot({(a1 -2 * a2),(a1 + a2)}, view = [-5 ..5, -5 ..5], symbolsize = 15);
```

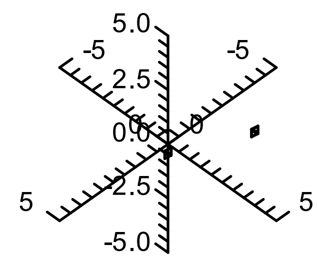


If the vectors are in R^3 instead of R^2, we use the command pointplot3d

Unfortunately, the default option for 3-dimensional plots in Maple is to hide the axes. This can be fixed by either clicking once on the 3-D plot above and then clicking on the icon for normal axes or by using the axes=normal option. Once again there is a view option for these graphs.

```
pointplot3d({w1, w2}
    , color = blue,  symbol = diamond,
    symbolsize = 15);
```

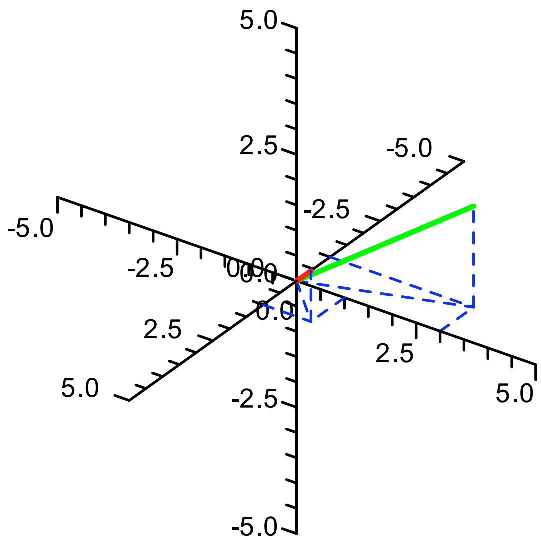
```
pointplot3d({w1, w2}
    , axes = normal, view = [-5 ..5, -5 ..5,
    -5 ..5], color = blue,  symbol =
    diamond,  symbolsize = 15);
```



Click on the graph and rotate the plot to get a good idea of the location of the two points.

To help visualize the point in space, it might be helpful to plot the dashed lines (which indicate the projection of the point on the coordinate planes).

```
> l1 := line([- 1, 0, 0],[- 1, 3, 0]
    , color = blue, thickness = 1, linestyle = DASH) :
  l2 := line([0, 3, 0],[- 1, 3, 0]
  , color = blue, thickness = 1, linestyle = DASH) :
  l3 := line([- 1, 3, 0],[- 1, 3, 2]
  , color = blue, thickness = 1, linestyle = DASH) :
  l4 := line([0, 0, 0],[- 1, 3, 0]
  , color = blue, thickness = 1, linestyle = DASH) :
  l5 := line([0, 0, 0],[-1, 3, 2], color = green, thickness = 2) :
  l11 := line([1, 0, 0],[1, 1, 0], color = blue, thickness = 1, linestyle = DASH
  ) :
  l12 := line([0, 1, 0],[1, 1, 0]
  , color = blue, thickness = 1, linestyle = DASH) :
  l13 := line([1, 1, 0],[1, 1, 1], color = blue, thickness = 1, linestyle = DASH
  ) :
  l14 := line([0, 0, 0],[1, 1, 0]
  , color = blue, thickness = 1, linestyle = DASH) :
  l15 := line([0, 0, 0],[1, 1, 1], color = red, thickness = 2) :
  display([l11, l12, l13, l14, l15, l1, l2, l3, l4, l5], axes = normal, view =[
  -5 .. 5, -5 .. 5, -5 .. 5]);
```



We can think of the tip of the red and green segments as the points, and of the segments itself as the vectors.

Click on the graph and rotate the plot to get a good idea of the location of the two points.
After you click on the graph, try typing $\theta=-160$ [Enter] and $\varphi=60$ [Enter] on the boxes (located at the top left of the tool bar).
Which angles θ and φ give you a good view of these vectors?

Exercises:

2.1) Plot the points [1, 1], [2, -2], [-3, 3], and [4, -4] all on the same graph.

>

2.2) Using the points z1 and z2 you defined in Exercise 2 above, plot z1, z2, z1 + z2, and 2*z1 - z2 all on the same graph.

>

2.3) Include the dashed lines and vectors to the pictures of z1, z2 and z1+z2. Choose angles θ and φ that give you a good view of these three vectors.

>

3. Visualizing operations with vectors:

Now we are ready to visualize the result of vector operations (multiplication by scalar and addition)

This is a section that I, Harry Mills, volunteered to do for Hugh and Regina. Bogged down with details of other matters, I didn't get to it. The idea was to do a VERY modest enhancement of what Russell and Mike did with the vector space operations on vectors. About all that I was thinking of adding to what's been done already was to superimpose the original vectors in harpoon form on a "cloud" of points that resulted from various operations.

The main place I saw this being of use was in the linear combinations of vectors example already done by our leaders. Adding the original "basis vectors for the subspace" in harpoon form to the resulting convex hull of linear combos is something I mentioned and then our workshop leaders did on their own.

Something that has NOT been done, here or there, is, by taking scalars from a bounded set, it may be useful to illustrate, for instance, a 3-D convex hull constructed from a random triple of linearly independent vectors in 3-space. It wouldn't fill the screen, but the students could convince themselves that they could fill up a ball, and eventually absorb all of 3-space (by "un-bounding" their set of scalars), which would plant the "absorbing" idea in their heads, just in case they stumble into Functional Analysis in grad school.

```
>
```

Exercises:

3.1)

```
>
```

3.2)

```
>
```

4. Random Number Generators and Long Lists:

It is useful to be able to generate random vectors and matrices. The rand function in Maple returns random integers in a specified range. We can use rand to create functions that produce random 3 digit numbers either from 0 to 1 or from -1 to 1.

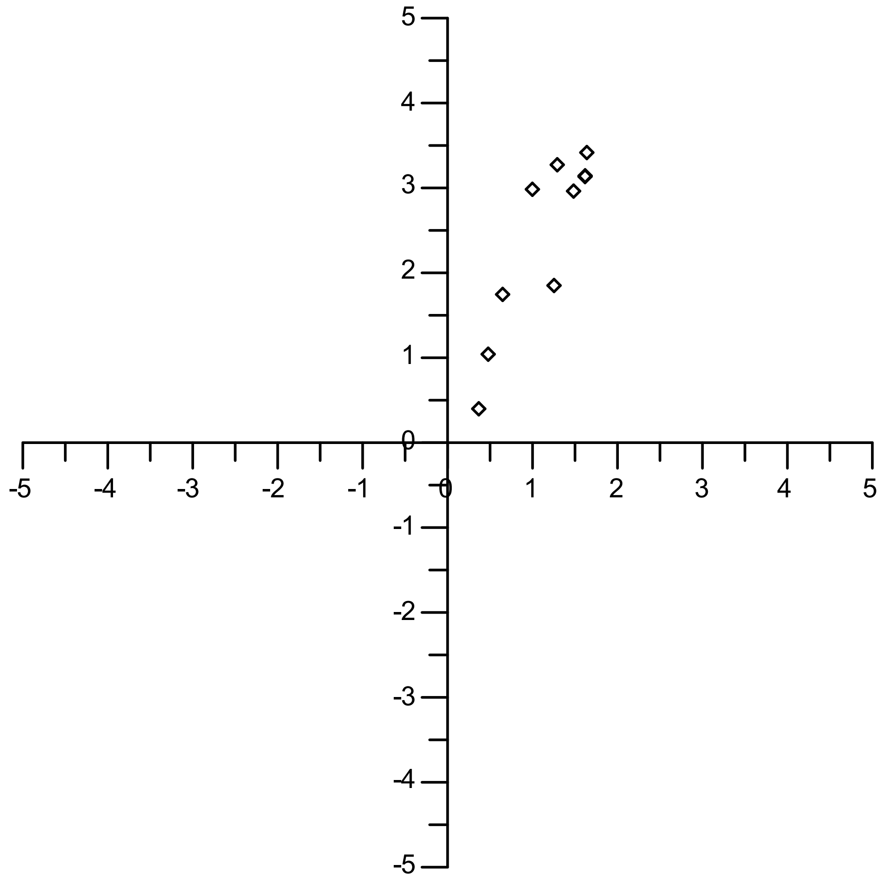
```
> rand0to1  := rand(0..1000)/1000.0:
  randneg1to1  := rand(-1000..1000)/1000.0:
```

With the first of these functions it is easy to produce a list of 10 random linear combinations of the form A*v1+B*v2, where A and B are both between 0 and 1.

```
> setofpoints  := {seq(rand0to1()*v1+rand0to1()*v2,i=1..10)};
  pointplot(setofpoints,view=[-5..5,-5..5]);
```

$$setofpoints := \left\{ \begin{bmatrix} 0.366999999999999993 \\ 0.398999999999999966 \end{bmatrix}, \begin{bmatrix} 0.997999999999999998 \\ 2.98399999999999999 \end{bmatrix}, \begin{bmatrix} 1.29099999999999992 \\ 3.27299999999999968 \end{bmatrix}, \begin{bmatrix} 1.61799999999999988 \\ 3.13399999999999990 \end{bmatrix}, \begin{bmatrix} 0.648000000000000020 \\ 1.74600000000000022 \end{bmatrix}, \right.$$

$$\begin{bmatrix} 1.25299999999999989 \\ 1.85099999999999997 \end{bmatrix}, \begin{bmatrix} 1.63900000000000001 \\ 3.41699999999999981 \end{bmatrix}, \begin{bmatrix} 1.48300000000000010 \\ 2.96299999999999964 \end{bmatrix},$$
$$\begin{bmatrix} 0.477999999999999981 \\ 1.04199999999999981 \end{bmatrix}, \begin{bmatrix} 1.61900000000000000 \\ 3.14299999999999979 \end{bmatrix} \}$$



Exercise:

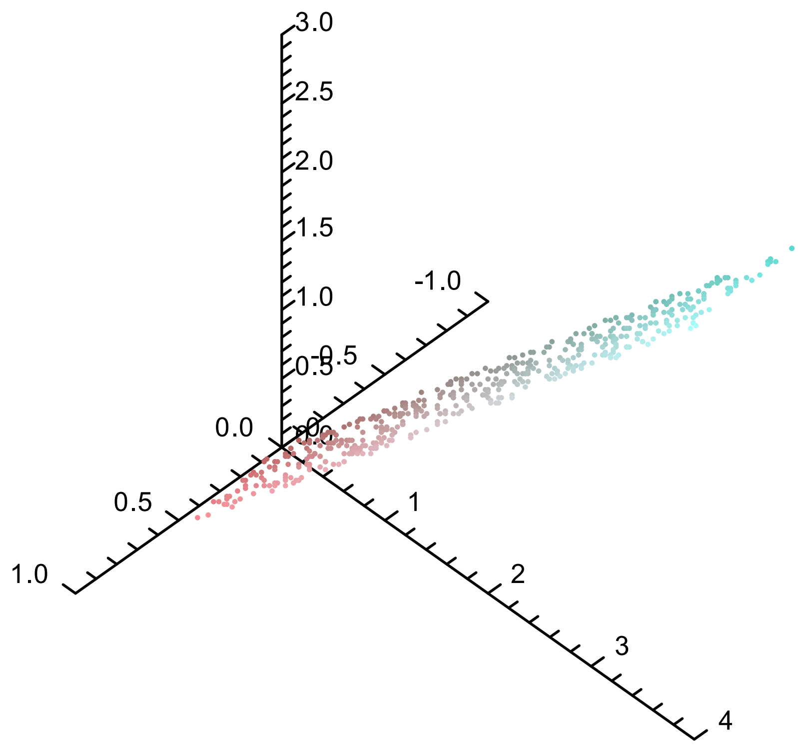
4.1) Use the rand0to1 function to create a list of 500 random linear combinations of v1 and v2. (You probably want to end the command with a colon rather than a semicolon so the list is not printed out.) Plot the points in the list and describe the geometric figure that they make. Include the coordinates of the vertices in your description.

>

When we try the same trick with vectors in R^3, we find that the points all lie in a plane. To see this, rotate the figure below in such a way that the plane is viewed edge-on - that is, so that it appears to be a line.

```
> setofpoints := {seq(rand0to1()*w1+rand0to1()*w2,i=1..500)}:
```

```
pointplot3d(setofpoints,view=[-1..1,0..4,0..3], axes=normal);
```



When a 3-D plot is active, you see the present orientation in the second menu bar. θ gives the angle of view in degrees in the xy-plane around from the positive x-axis, while ϕ gives the view angle down from vertical. The default view orientation is $\theta = 45$ and $\phi = 45$. Trial and error rotations of the figure above show that an orientation of $[\theta, \phi] = [17,65]$ views the plane containing all the points on edge, while an orientation of [-11, 17] looks at that plane from the top so that the set of points looks like a figure in a plane.

>

Exercise:

4.2)

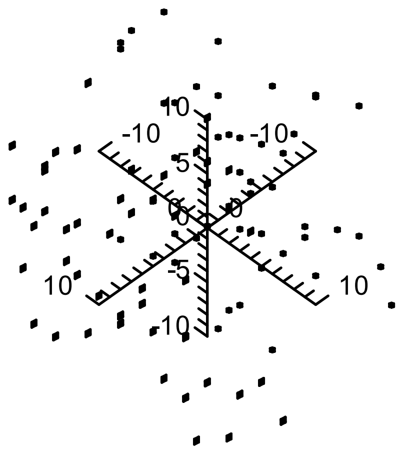
5. The effect of Linear Transformations on Linear Combinations (Might better be named "The nature of the image of a linear transformation")

This is Harry's (my) section, and I've done a lot of playing around with colors, procedures, and saveing and read-ing files, to little avail. I see very little that I have added to the original visualizations offered by our workshop leaders.

A Very lengthy procedure you don't want to see. So close this baby.

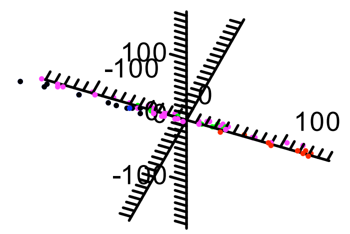
There are two plots given below. The plot on the left is a "cloud" of random points in $\mathbb{R}^3$. The plot on the right is the image of the cloud under a linear transformation.

> *MyCloud*



>

> *ImageOfCloud*



>

Exercises

(5.1) Use the mouse-arrow to rotate the plots, above. What does the image of the cloud (not the cloud, itself, but what happens to it after it is acted on by A in the plot on the right) look like as a geometric object?

>

(5.2) Question here trying to induce the student to say "Ah ha! It's a planar subspace of $\mathbb{R}^3$!"

>

(5.3) Relate the dimension of the image to the rank of the matrix A.

>

> *ImageOfCloud*

>

6. Parametric Equations for Lines and Planes:

>

>

Exercise:

6.1)

>

6.2)

>

>

In this part of the worksheet we are going to look at parametric equations for lines and planes in R^3.
First, for parametric equations of a line, let's enter a vector **v** in R^3 that will be in the direction of the line and a vector $\mathbf{r_0}$ in R^3 that will be pointing from the origin to a point on the line.
So, first enter the vector **v**: