Vector Fields and Mathematica

Note: Vector Fields change often with new versions of Mathematica. These notes were most recently updated in November 2020 for Version 12.1.1.

Basic Vector Fields

Start with **F**(*x*, *y*) = -*y* **i** + (*x* + *y*)**j** by evaluating the following. Note that here color represents magnitude.

```
VectorPlot[{-y,x+y},{x,-3,3},{y,-3,3}]
```

We use options after the domain inside the last bracket] as below to make adjustments. PlotLegends will show the connection between color and magnitude.

```
VectorPlot[{-y,x+y},{x,-3,3},{y,-3,3}, PlotLegends->Automatic]
```

To focus on vectors rather than color, delete PlotLegends and insert the following:

```
VectorColorFunction->None,VectorScaling->Automatic
```

Note: The vectors are scaled according to Mathematica and have lengths proportional to their actual magnitudes. I have not figured out how to get the vectors to have their actual lengths in version 12.

For arrowheads a consistent size, use `VectorStyle->Arrowheads[0.025]` to set them.

Also try `Frame->False,Axes->True,AxesLabel->{"x","y"}` for a traditional look.

For fewer points (or more points) use `VectorPoints->7` or another value. `Automatic` is default.

Using `VectorPoints->{"Regular",7}` in this example forces one vector on each integer point. Vectors are graphed centered at the point. Use `GridLines->Automatic` to see this. To have the vectors start at the point of calculation use `VectorMarkers->Placed["Arrow","Start"]` (however, this seems to cancel the arrowhead size setting).

Try using `VectorPoints→{{2,2},{-1,2}}`. This plots vectors only at (2, 2) and (-1,2).
Evaluate `RandomReal[{-3,3},{20,2}]`. This generates 20 ordered pairs within our domain.
Now try: `VectorPoints→RandomReal[{-3,3},{20,2}]`. Evaluate. Try changing 20 to 400.
Try changing `VectorPlot` to `StreamPlot`.

<u>Graph these vector fields:</u> (a) **F**(*x*, *y*) = *x* **i** + **j** (b) **F**(*x*, *y*) = (*y* − *x*) **i** + (*y* + 3)**j** (c) **F**(*x*, *y*) = 3**i** + 4**j**

Three-Dimensional Examples:

Start with **F**(*x*, *y*, *z*) = *x***i** + (*x* + *z*)**j** + (*y* – *z*)**k** as follows:

```
VectorPlot3D[{x,x+z,y-z},{x,-3,3},{y,-3,3},{z,-3,3}]
```

Remember, you can rotate 3D graphs and use Ctrl while dragging to zoom.

The 3D graph uses three-dimensional arrows. Try `VectorMarkers->"Arrow"` for another style.
Try adjusting VectorScaling, VectorColorFunction, and VectorPoints as in the two-dimensional graphs.

<u>Graph these vector fields:</u> Remember: cos *x* = Cos[x]

(a) **F**(*x*, *y*, *z*) = *x***i** + *y* **j** + *z* **k** (b) **F**(*x*, *y*, *z*) = 2**i** + 2**j** + 2**k** (c) **F**(*x*, *y*, *z*) = cos *x* **i** + *z* **j** + sin 2*y* **k**

Functions, Contours, and Gradient Fields (3D)

Let's take a look at the function $f(x,y) = x^2 y - y^3$.

```
Plot3D[x^2y-y^3,{x,-4,4},{y,-4,4},BoxRatios→{1,1,2}]
```

To graph the level curves:

```
levels=ContourPlot[x^2y-y^3,{x,-4,4},{y,-4,4},
        Contours->35,ContourShading->False]
```

Find the gradient of *f*. Type and evaluate: `D[x^2y-y^3,{{x,y}}]`

Graph the gradient:

```
field=VectorPlot[{2x*y,x^2-3y^2},{x,-4,4},{y,-4,4},
        VectorColorFunction->None,VectorScaling->Automatic]
```

Graph both together: `Show[levels,field]`

What do you notice?

Functions, Contours, and Gradient Fields (4D)

Consider the function $f(x,y,z) = x^2 + y^2 + z^2$.

This is now a 4-dimensional situation, we cannot graph it. We can, however, understand the function by graphing the level surfaces. Let's start with the level surface when $f(x,y,z) = 4$.

```
levels=ContourPlot3D[x^2+y^2+z^2,{x,-3,3},{y,-3,3},{z,-3,3},Contours->{4}]
```

To show more level surfaces, change `Contours->{4}` to `Contours->{1,2,3,4,5}` and add the options `ContourStyle→Opacity[0.5]` and `Mesh→False`.

Where are the level surfaces?
Change domain for *y* to `{y,-3,0}`, add option `BoxRatios->Automatic` and rotate the result.

Find the gradient of *f*: `D[x^2+y^2+z^2,{{x,y,z}}]`

Graph the vector field:

```
field=VectorPlot3D[{2x,2y,2z},{x,-3,3},{y,-3,0},{z,-3,3},
    VectorMarkers->"Arrow",VectorColorFunction->None,
    VectorScaling->Automatic]
```

All together now: `Show[levels,field]`

Be sure to look or zoom in closely. *What do you notice?*

Additional Practice

Repeat the section "Surfaces, Contour Lines, and Gradient Fields (4D)" with the function:

$$f(x,y,z)=\frac{x^2}{3}+\frac{y^2}{4}-z$$

Or try using the following code so you can change the function and plot range easily.

```
f[x_,y_,z_]=x^2/3+y^2/4-z;
r=3;

levels=ContourPlot3D[f[x,y,z],{x,-r,r},{y,-r,r},{z,-r,r},
    Contours->{0,1,2,3,4,5},ContourStyle->Opacity[0.5],Mesh->False];

field=VectorPlot3D[
   Evaluate[D[f[x,y,z],{{x,y,z}}]],{x,-r,r},{y,-r,r},{z,-r,r},
   VectorColorFunction->None,VectorMarkers->"Arrow",
   VectorScaling->Automatic];

Show[levels,field,BoxRatios->Automatic]
```

Try using the code for $f(x,y,z)=x^3-yz^2$

Interactive 2D Vector Field

Try this:

```
Manipulate[
  VectorPlot[{a*x,b*y},{x,-3,3},{y,-3,3},
    Frame→False,Axes→True,PlotRange→3,
    VectorPoints→10,VectorColorFunction→ColorData["Rainbow"]
  ],
 {a,-3,3},{b,-3,3}]
```

Go to "ColorSchemes" in the "Palettes" menu for more color choices.

Also . . .

If you have time, check out the function `StreamPlot`. You can use it like VectorPlot.
No 3D version of this one. Try it with `StreamColorFunction->ColorData["Rainbow"]`.