Graphing Functions of Two Variables

You’ve already seen how to plot a graph of a function of one variable, for instance:

   plotFun( 95 - 73\*exp(-.2\*t) ~ t, tlim=range(0,20) )



This lesson is about plotting functions of two variables. For the most part, the format used will be a contour plot, but it’s also possible to make the graph of the function, as you’ll see later.

You use the same plotFun( ) function to plot with two input variables. The only change is that you need to list the two variables on the right of the ~ sign, and you need to give a range for each of the variables. For example:

   plotFun( sin(2\*pi\*t/10)\*exp(-.2\*x) ~ t & x, tlim=range(0,20), xlim=range(0,10))



Each of the contours is labeled, and by default the plot is filled with color to help guide the eye. If you prefer just to see the contours, without the color fill, use the filled=FALSE argument.

   plotFun( sin(2\*pi\*t/10)\*exp(-.2\*x) ~ t & x,
     t.lim=range(0,20), x.lim=range(0,10), filled=FALSE)



Occasionally, people want to see the function as a surface, plotted in 3 dimensions. You can get the computer to display a perspective 3-dimensional plot by giving the optional argument surface=TRUE.

   plotFun( sin(2\*pi\*t/10)\*exp(-.2\*x) ~ t & x,
     t.lim=range(0,20), x.lim=range(0,10), surface=TRUE)



If you are using RStudio, you can press on the little “gear” icon in the plot and you will have a slider to control the viewpoint. (Try moving the slider to the right, release it, and wait for the picture to update.)

It’s very hard to read quantitative values from a surface plot — the contour plots are much more useful for that. On the other hand, people seem to have a strong intuition about shapes of surfaces. Being able to translate in your mind from contours to surfaces (and vice versa) is a valuable skill.

Finally, you can ask plotFun( ) to create a function that you can evaluate numerically, not just graphically. For example:

   g = plotFun( sin(2\*pi\*t/10)\*exp(-.2\*x) ~ t & x,
     t.lim=range(0,20), x.lim=range(0,10))
   g(x=4, t=7)

  [1] -0.4273372

Make sure to name the arguments explicitly when inputting values. That way you will be sure that you haven’t reversed them by accident. For instance, note that this statement gives a different value than the above:

   g(4, 7)

  [1] 0.1449461

but this one is the same

   g(t=7, x=4)

  [1] -0.4273372

Exercise 1 Refer to this contour plot: 

Approximately what is the value of the function at each of these (x,t) pairs? Pick the closest value

(a)

x = 4,t = 10:

 -6  -5  -4  -2  0  2  4  5  6

(b)

x = 8,t = 10:

 -6  -5  -4  -2  0  2  4  5  6

(c)

x = 7,t = 0:

 -6  -5  -4  -2  0  2  4  5  6

(d)

x = 9,t = 0:

 -6  -5  -4  -2  0  2  4  5  6

Exercise 2 Describe the shape of the contours produced by each of these functions. (Hint: Make the plot! Caution: Use the mouse to make the plotting frame more-or-less square in shape.)

(a)

The function

   plotFun( sqrt( (v-3)^2 + 2\*(w-4)^2 ) ~ v&w, v.lim=range(0,6), w.lim=range(0,6))

has contours that are

 Parallel Lines  Concentric Circles  Concentric Ellipses  X Shaped

(b)

The function

   plotFun( sqrt( (v-3)^2 + (w-4)^2 ) ~ v&w, v.lim=range(0,6), w.lim=range(0,6))

has contours that are

 Parallel Lines  Concentric Circles  Concentric Ellipses  X Shaped

(c)

The function

   plotFun( 6\*v - 3\*w + 4 ~ v&w, v.lim=range(0,6), w.lim=range(0,6))

has contours that are:

 Parallel Lines  Concentric Circles  Concentric Ellipses  X Shaped