College of the Redwoods
Mathematics Department
Multivariable Calculus

# Level Curves in Matlab

**David Arnold**

## Directory

- **Table of Contents.**
- **Begin Article.**


Last Revision Date: May 4, 1999 Version 1.00

# Table of Contents

1. Introduction and Prerequisites
2. Level Sets and Contours
    2.1. Matlab's Contour Command
    2.2. Labeling the Contours
    2.3. Choosing Particular Contours
    2.4. Implicit Function Plotting
    2.5. Homework Exercises

# 1. Introduction and Prerequisites

In this activity, Matlab is used to explore the level curve concept of functions mapping $R^2$ into $R$. Some familiarity with Matlab's `meshgrid` command is required, as well as rudimentary knowledge of Matlab's element wise operators (`.*`, `./`, `.^`).

# 2. Level Sets and Contours

Let's begin with a definition.

**Definition 1** *Let $f : R^2 \to R$. The set $\{(x, y) : f(x, y) = c\}$, where $c$ is an arbitrary constant, is called a level set of the function $f$.*

Consider the function $f : R^2 \to R$ defined by $f(x, y) = x^2 + y^2$. The level sets of $f$ are then defined by

$$\{(x, y) : f(x, y) = c\} \quad \text{or} \quad \left\{(x, y) : x^2 + y^2 = c\right\} \tag{1}$$

If you choose $c = 1$ in Equation 1, then the set of points

$$\{(x, y) : x^2 + y^2 = 1\} \tag{2}$$

is a circle of radius 1, centered at the origin (See Figure 1), and is called a level set of the function $f$.

The usual practice is to sketch several level sets by selecting different values for the constant $c$. For example, if you let $c = 1, 2, 3, 4, 5$ in Equation 1, then the following level sets are obtained.

$$\{(x, y) : x^2 + y^2 = 1\}$$
$$\{(x, y) : x^2 + y^2 = 2\}$$
$$\{(x, y) : x^2 + y^2 = 3\}$$
$$\{(x, y) : x^2 + y^2 = 4\}$$
$$\{(x, y) : x^2 + y^2 = 5\}$$

Each of these level sets is a circle, centered at the origin, with radius 1, $\sqrt{2}$, $\sqrt{3}$, 2, and $\sqrt{5}$. Note that it is customary to label each level

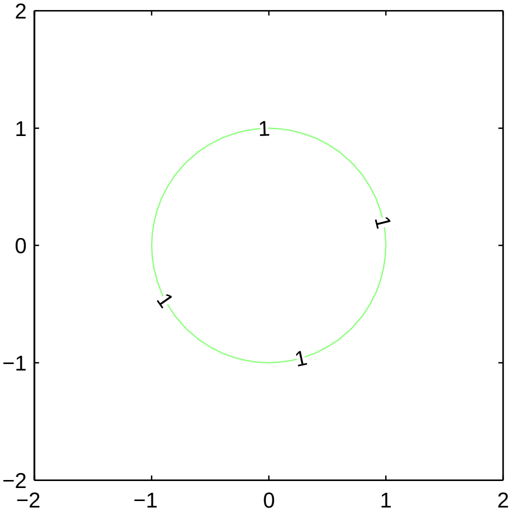


Figure 1: The level set $f(x, y) = 1$.

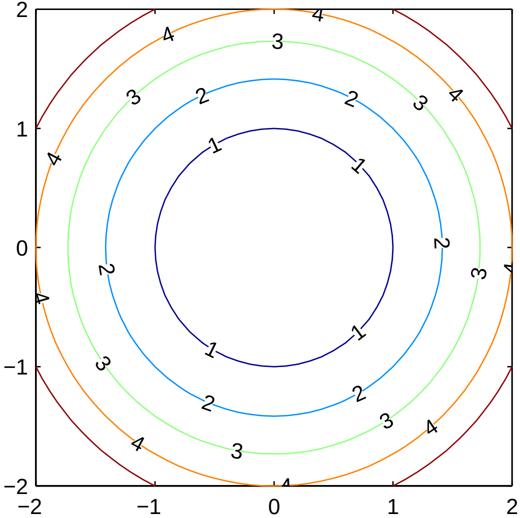


Figure 2: Level curves for $c = 1, 2, 3, 4, 5$.

curve with its $c$-value, as shown in Figure 2.

The level sets pictured in Figure 2 offer a wealth of visual information about the function $f$. For example, as you move away from the point $(0, 0)$, located at the center of Figure 2, the function values increase. As you move toward the point $(0, 0)$, the function values decrease. It is no coincidence that the level sets in Figure 2 closely resemble a *topographical* map, where each contour represents a constant height.

There are numerous applications where level curves can be very useful. For example, suppose that the function $f(x, y) = x^2 + y^2$ used to generate the level curves in Figure 2 represents the temperature (in degrees Fahrenheit) at the position $(x, y)$. Any point selected from the curve $x^2 + y^2 = 1$ will have temperature $1\,^\circ$F, points selected from the curve $x^2 + y^2 = 2$ will have temperature $2\,^\circ$F, and so on.

## 2.1. Matlab's Contour Command

Matlab simplifies the process of constructing level curves, even for the most difficult of functions.

**Example 1** *Sketch several level curves of the function* $f : R^2 \to R$ *defined by*

$$f(x, y) = \frac{-3y}{x^2 + y^2 + 1} \tag{3}$$

*over the region* $\{(x, y) : -2 \le x \le 2, -2 \le y \le 2\}$ *and label each level curve with its constant function value.*

**Solution.** First use the meshgrid command to create a grid of $x$ and $y$-values on the given domain. Calculate the function value at each point and use Matlab's `contour` command to draw the level curves. The following commands should produce an image similar to that in Figure 3.

```
>> [x,y]=meshgrid(-2:.1:2);
>> z=-3*y./(x.^2+y.^2+1);
>> contour(x,y,z)
```

If you are not satisfied by the number of level curves produced, it is a simple matter to add more. The following command should produce 10 level curves, similar to those in Figure 4.

```
>> contour(x,y,z,10)
```

## 2.2. Labeling the Contours

It is a simple task to label each level curves with its constant function value. The following commands were used to produce the image in Figure 5.

```
>> [c,h]=contour(x,y,z);
>> clabel(c,h);
```

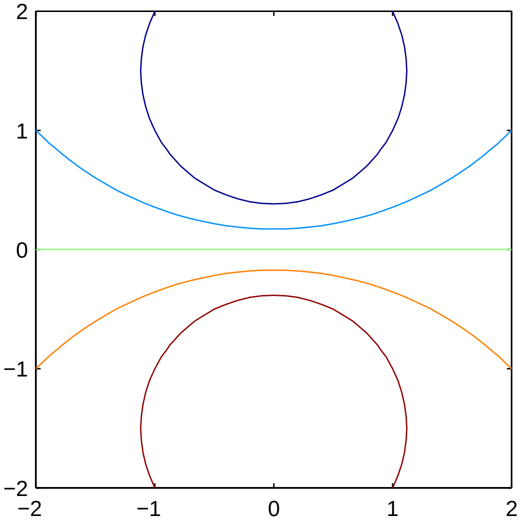


Figure 3: Level curves of $f(x, y) = -3y/(x^2 + y^2 + 1)$.

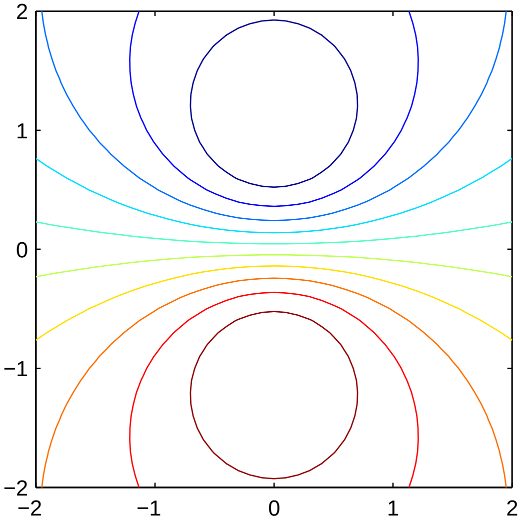


Figure 4: Ten level curves.

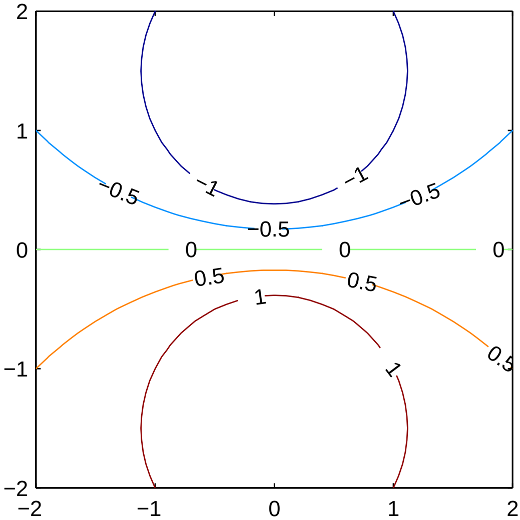


Figure 5: Labeling the level curves.

*Note: If you are unhappy with the placement of the labels, then you might want to try* `clabel(c,h,'manual')` *instead. This command will allow you to place the labels with your mouse.*

## 2.3. Choosing Particular Contours

You might have noted by now that Matlab automatically decides on the optimum $c$-values when plotting the level curves $f(x, y) = c$. You can easily override this automatic selection and plot contours for particular $c$-values. For example, suppose that you want level curves for $c = -1.25, -1.00, -0.75, \ldots, 1.5$. Recall that the Matlab code `-1.25:.25:1.25` will produce this vector of $c$-values. The following command was used to create the image in Figure 6.

```
>> [c,h]=contour(x,y,z,-1.25:.25:1.25);
>> clabel(c,h)
```

*Note: Again, if you do not care for the crowded appearance of the labels in Figure 6, try the command* `clabel(c,h,'manual')`*, which*

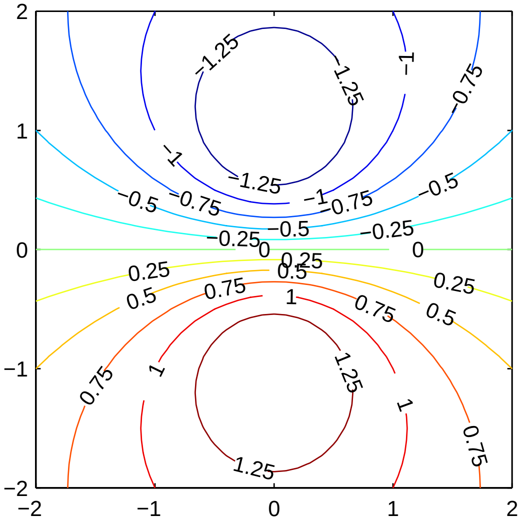


Figure 6: Specifying particular contours.

*will allow you to set the labels individually with the mouse.*

## 2.4. Implicit Function Plotting

Often, an equation such as

$$x^3 + y^3 = 3xy \tag{4}$$

is difficult (or impossible) to solve for $y$ in terms of $x$. However, you can use Matlab's `contour` command as an implicit function plotter, eliminating the need to explicitly solve the equation for $y$ in terms of $x$ before plotting. Begin by making one side of Equation 4 equal to zero.

$$x^3 + y^3 - 3xy = 0 \tag{5}$$

Next, define a function $f : R^2 \to R$ by $f(x, y) = x^3 + y^3 - 3xy$. Equation 5 now reads

$$f(x, y) = 0, \tag{6}$$

where $f(x,y) = x^3 + y^3 - 3xy$. Consequently, Equation 6 is the level curve $f(x,y) = 0$ of the function $f(x,y) = x^3+y^3-3xy$. You can plot a single level curve of a function by using Matlab's `contour` command in the form `contour(x,y,z,[c c])`. The following commands should produce an image similar to that in Figure 7. Note how a finer mesh is used in this example to improve the accuracy of the plot.

```
>> [x,y]=meshgrid(-2:.05:2);
>> z=x.^3+y.^3-3*x.*y;
>> contour(x,y,z,[0,0])
```

## 2.5. Homework Exercises

1. Consider the function defined by the equation

$$f(x,y) = -xye^{-x^2-y^2}$$

   (a) Use Matlab's `contour` command to plot twenty (20) contours over the domain $\{(x,y) : -2 \leq x,y \leq 2\}$.

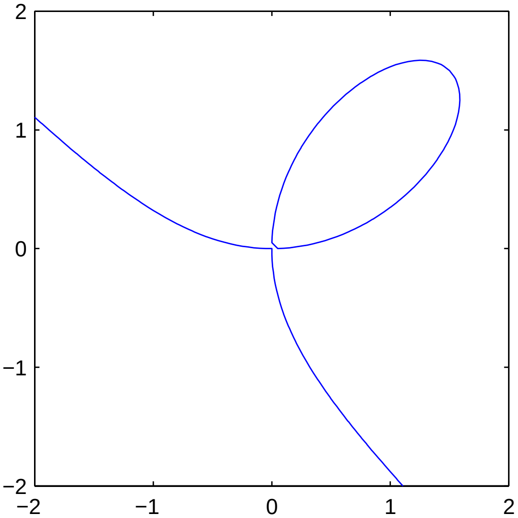


Figure 7: Plotting a single contour.

(b) Use the form `clabel(c,h,'manual')` to selectively label several contours with the mouse.

(c) Obtain a printout of your result.

2. Consider the function defined by the equation

$$f(x, y) = -xye^{-x^2-y^2}$$

(a) Use Matlab's `contour3` command to plot twenty (20) contours over the domain $\{(x, y) : -2 \le x, y \le 2\}$. *Hint: Type* `help contour3` *to obtain help on the* `contour3` *command.*

(b) Turn off the grid with the command `grid off`.

(c) Obtain a printout of your result.

3. Consider the function defined by the equation

$$f(x, y) = -xye^{-x^2-y^2}$$

(a) Use Matlab's `meshc` command to obtain a simultaneous plot of the surface and the level curves of the function over the domain $\{(x, y) : -2 \leq x, y \leq 2\}$. *Hint: Type* `help meshc` *to obtain help on the* `meshc` *command.*

(b) Remove hidden line removal with the command `hidden off`.

(c) Obtain a printout of your result.

4. Sketch the graph of the famous *knot curve* whose points satisfy the equation

$$(x^2 - 1)^2 = y^2(3 + 2y)$$

Obtain a printout of your result.