Graphing Functions and Solving Equations in Sage

Graphing Functions and Solving Equations in Sage

In this lesson, we will first learn how to graph functions, and then we will talk about different methods of solving equations in Sage.

Basic Graphing

You can graph a function in Sage using the "plot" command.

Example 1

Graph the function $f(x) = x^2$.

```
1   plot(x^2)
```
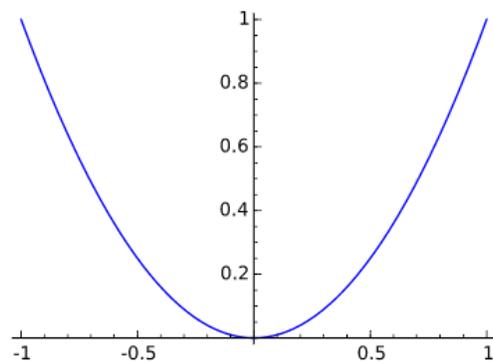

By default, the plot command creates a graph with $-1 \le x \le 1$. The range of values on the y-axis is chosen by Sage to fit the graph to the window.

```
2
```

You may also define the function $f(x) = x^2$ first, and then plot using the function name.

```
3   f(x)=x^2
4   plot(f)
```
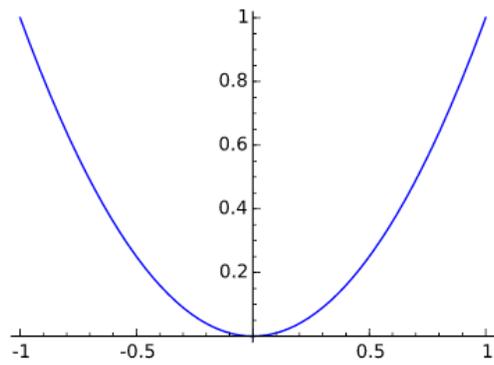
Changing the Plot Window - Horizontal Axis

To change the input values (horizontal axis), use the xmin and xmax options. Plot options are separated by commas.

5

Example 2

Graph $f(x) = x^2$ for $-10 \leq x \leq 10$.

6 `plot(x^2,xmin=-10,xmax=10)`
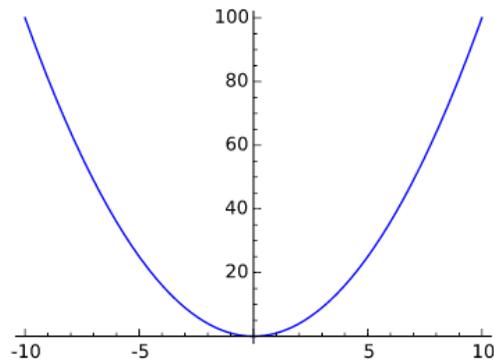
Example 3

Graph $f(x) = x^2$ for $-5 \leq x \leq 20$.

7 `plot(x^2,xmin=-5,xmax=20)`
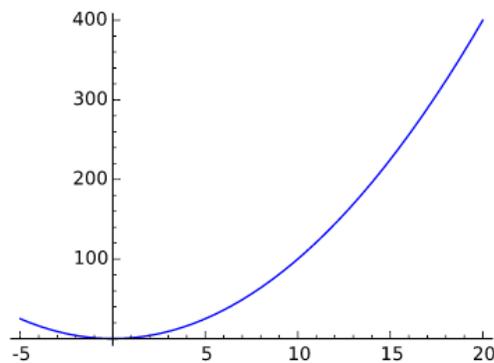
Changing the Plot Window - Vertical Axis

Unless you specify the range on the y-axis, Sage will choose it for you.

To change the output values (vertical axis), use the ymin and ymax options.

Example 4

Graph $f(x) = x^2$ for $-10 \leq x \leq 10$ and $-10 \leq y \leq 50$.

9 | `plot(x^2,xmin=-10,xmax=10,ymin=-10,ymax=50)`
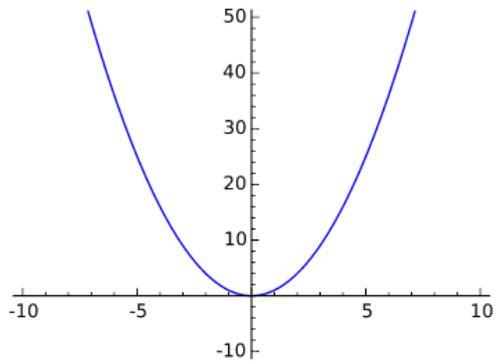
Example 5

Here's another window. Notice that I did not specify xmin, so it defaults to -1.

10 | `plot(x^2,xmax=20,ymin=-100,ymax=100)`
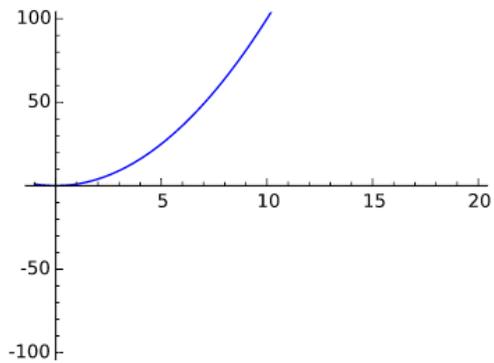
Example 6

Graph $f(t) = 2t + 4$ for $5 \leq t \leq 10$.

11 | `f(t)=2*t+4`
12 | `plot(f,xmin=5, xmax=10)`
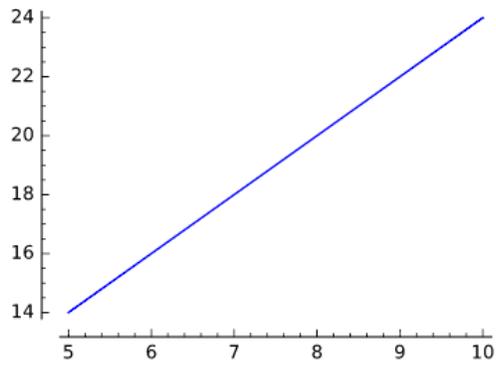
Note: Use the plot options xmin and xmax even if your input variable is something other than x.

Graphing More Than One Function

To graph multiple functions you add together multiple plots (using +).

13

Example 7

Graph $f(x) = x^2$ and $g(x) = (x-2)^2 - 3$.

(Note: You can get the graph of g by shifting the graph of f two units right and three units down.)

```
14  f(x)=x^2
15  g(x)=(x-2)^2-3
16  plot(f)+plot(g)
```
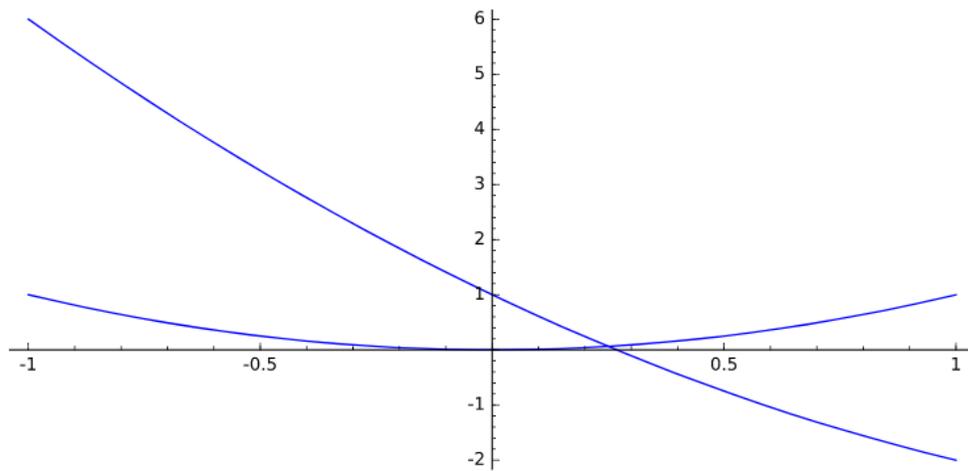

Example 8

Let's change the window so that $-5 \le x \le 5$.

```
17  plot(f,xmin=-5,xmax=5)+plot(g)
```
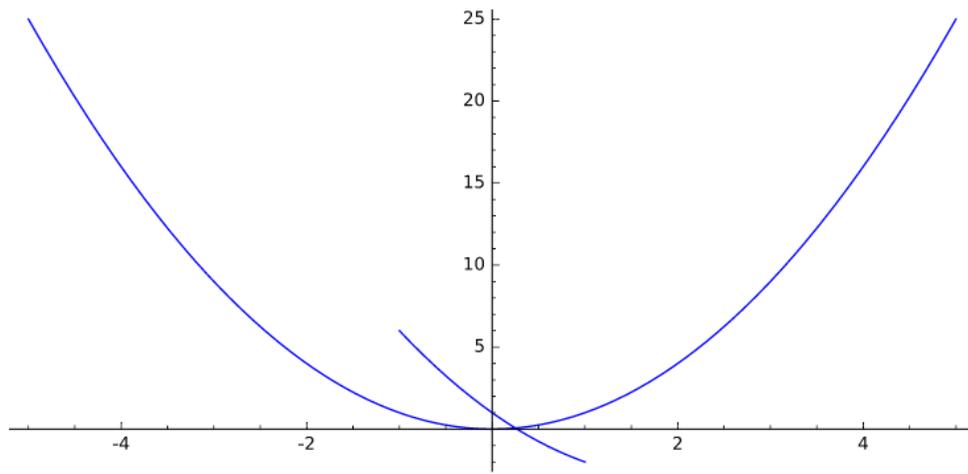
I changed the window for f, but g is still the default $-1 \le x \le 1$. I have to specify the window for each plot.

```
18  plot(f,xmin=-5,xmax=5)+plot(g,xmin=-5,xmax=5)
```
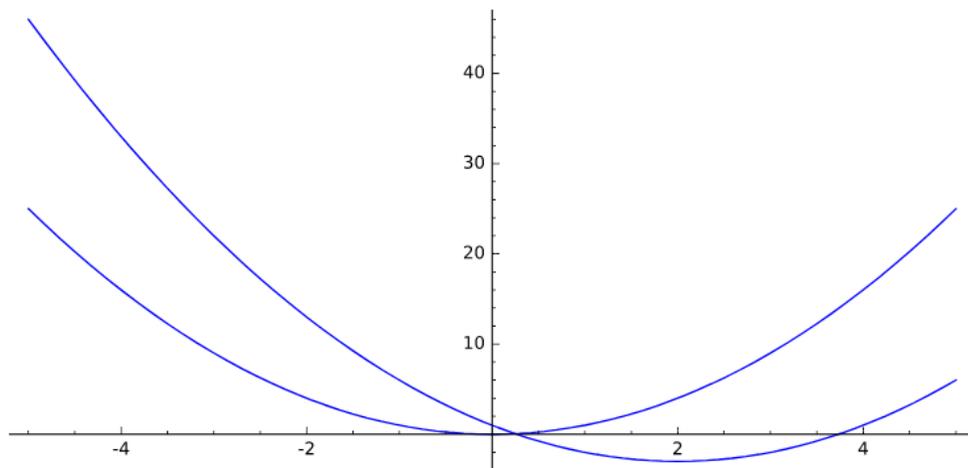

Changing Colors

We can change the color of the curves using the "color" option within a plot.

```
19
```

Example 9

Let's change the graph of g to red so we can tell the two graphs apart. You do this by adding color='red' to the g plot. Notice the quotes around red.

```
20  f(x)=x^2
21  g(x)=(x-2)^2-3
22  plot(f,xmin=-5,xmax=5)+plot(g,xmin=-5,xmax=5,color='red')
```
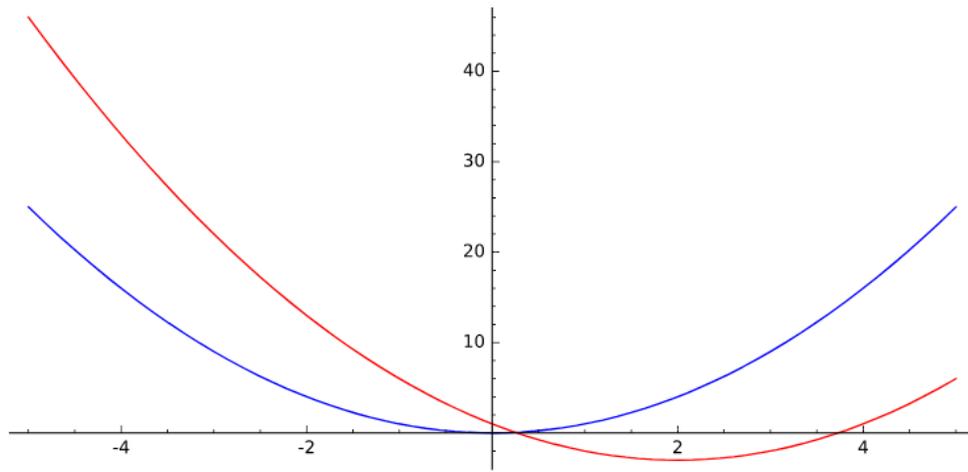
Example 10

Sage knows quite a few colors by name. Try some yourself.

```
23  plot(f,xmin=-5,xmax=5,color='salmon')+plot(g,xmin=-5,xmax=5,color='fuchsia')
```
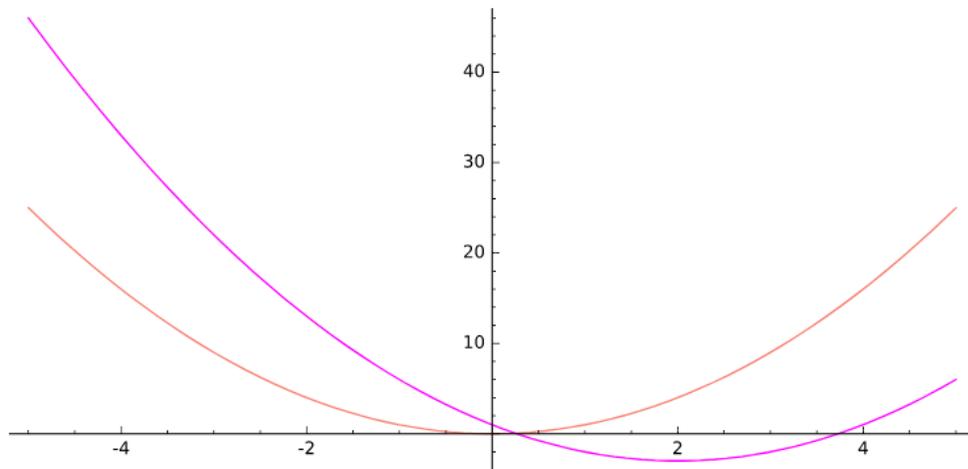

Example 11

Let's add a graph of $h(x) = x^3$ to get a graph of three functions.

```
24  f(x)=x^2
25  g(x)=(x-2)^2-3
26  h(x)=x^3
27  plot(f,xmin=-5,xmax=5,color='salmon')+plot(g,xmin=-5,xmax=5,color='fuchsia')+plot(h,xmin=-5,xmax=5,ymin=-50,ymax=50,color='teal')
```
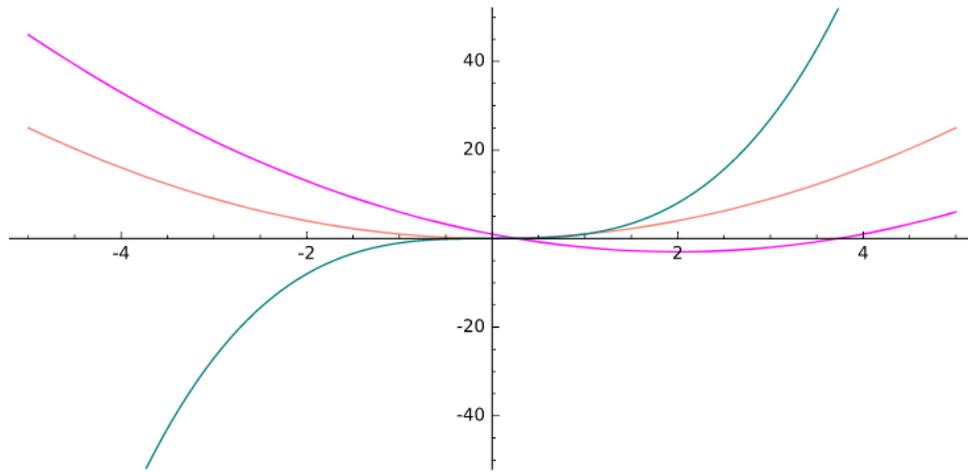
Note: I have adjusted the ymin and ymax in the plot of h above; otherwise, h would dominate the graph.

Changing the Line Style

You can also change the style of line using the "linestyle" option. The default is "solid," but you can also used dashed or dotted lines or a combination of the two.

28

Example 12

Let's make the graph of g using a dashed line by typing linestyle='dashed'

```
29  f(x)=x^2
30  g(x)=(x-2)^2-3
31  plot(f,xmin=-5,xmax=5)+plot(g,xmin=-5,xmax=5,color='red',linestyle='dashed')
```
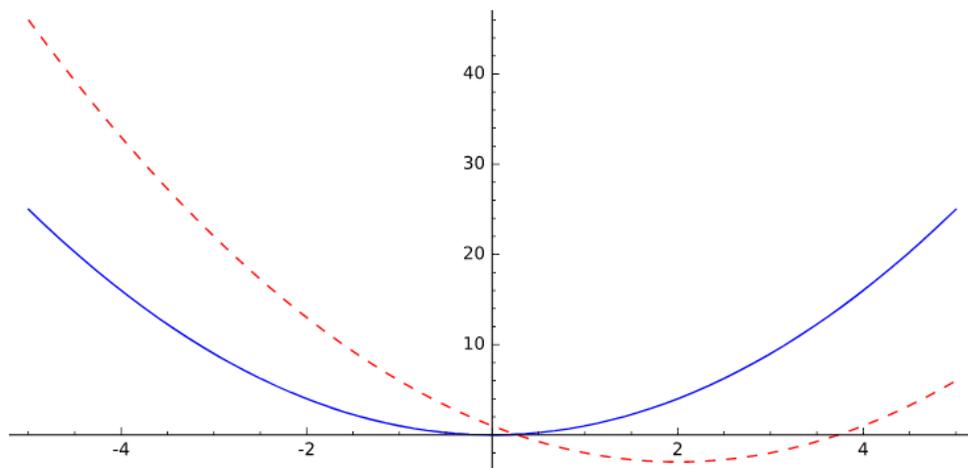

32

Example 13

To get a dotted line, type linestyle='dotted'

```
33  plot(f,xmin=-5,xmax=5)+plot(g,xmin=-5,xmax=5,color='red',linestyle='dotted')
```
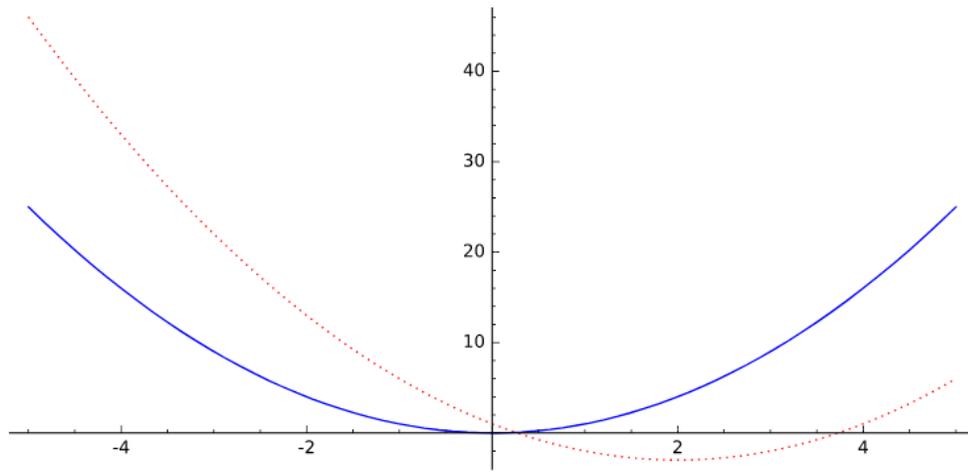
Example 14

To get a dash-dot pattern, use linestyle='dashdot'

```
34  plot(f,xmin=-5,xmax=5)+plot(g,xmin=-5,xmax=5,color='red',linestyle='dashdot')
```
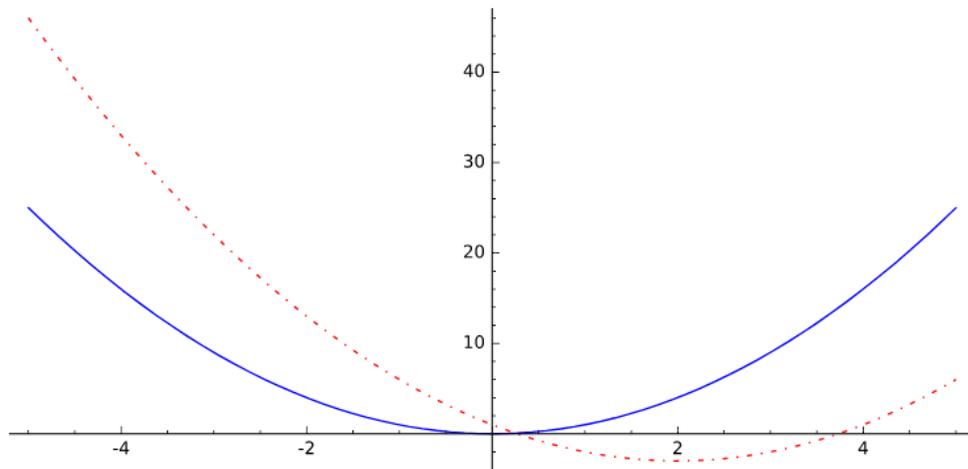

Solving Equations

There are several different ways to solve equations in Sage, including:

- Approximate solutions graphically
- Use the solve command
- Use the find_root command

We'll start with a graphical approach.

```
35
```

Example 15

Solve the equation $x^2 - 10 = 2x - x^2$ for x.

We'll graph both sides of this equation and see where the two curves cross.

```
36  f(x)=x^2-10
37  g(x)=2*x-x^2
38  plot(f)+plot(g,color='red')
```
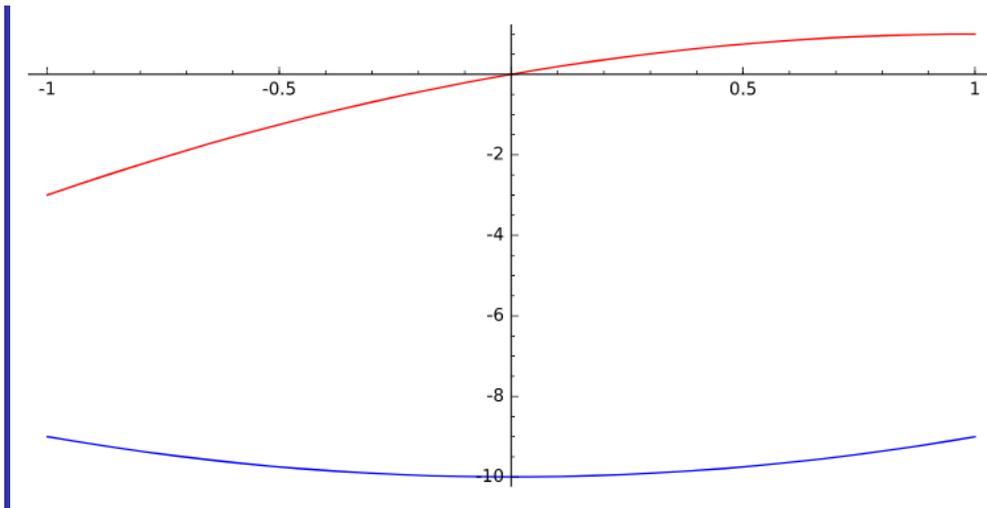
On the default plot, we don't see any points of intersection, so let's zoom out.

40 `plot(f,xmin=-10,xmax=10)+plot(g,xmin=-10,xmax=10,color='red')`
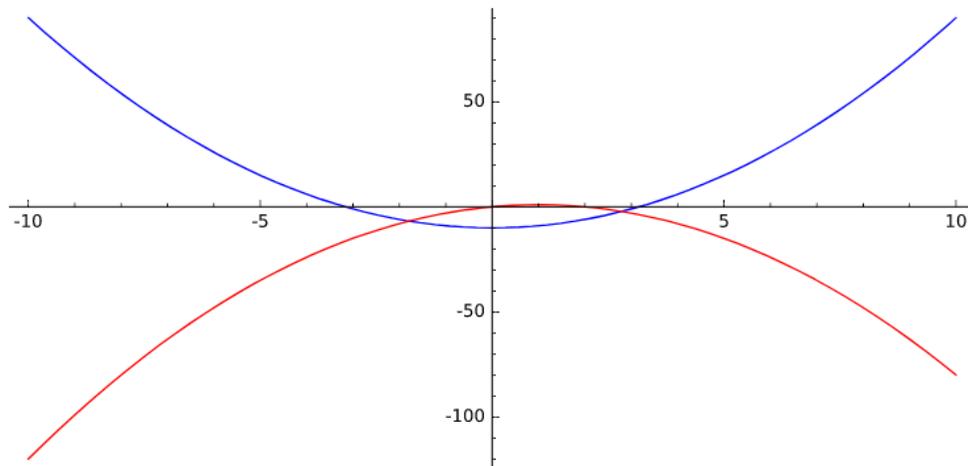
Now we see two points of intersection, around $x = 3$ and $x = -2$. Let's zoom in around $x = 3$.

42 `plot(f,xmin=2,xmax=4)+plot(g,xmin=2,xmax=4,color='red')`
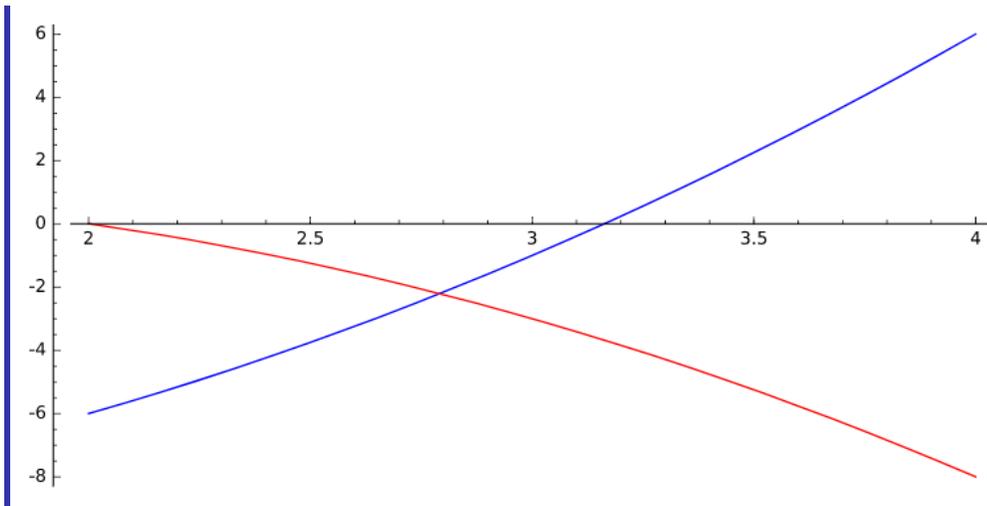
43

It looks like the curves cross near the point $(2.8, -2)$, so let's zoom in some more:

44 `plot(f,xmin=2.75,xmax=2.85)+plot(g,xmin=2.75,xmax=2.85,color='red')`
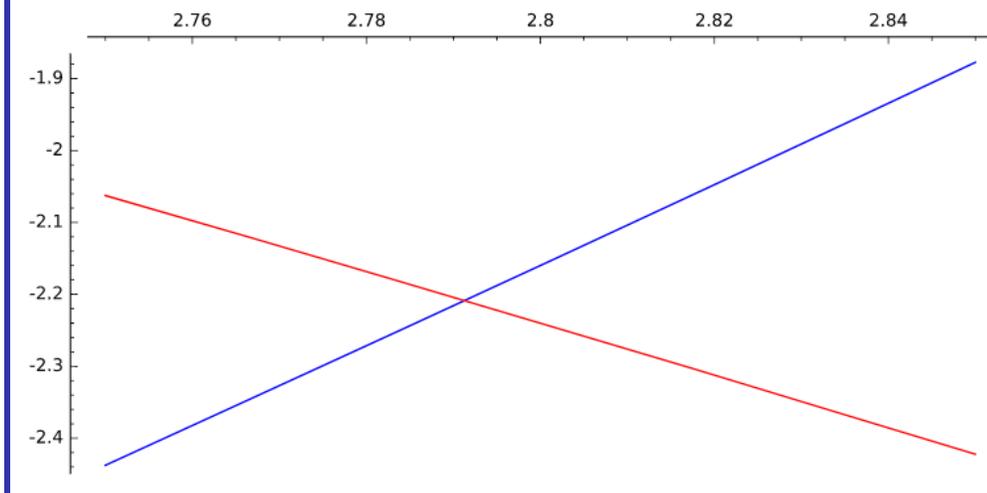
45

Depending on how accurate we want to be, we could keep doing this over and over again, but this is a tedious process.

46

Example 16

It is possible to plot an equation in Sage. This produces a plot of the difference of the two sides.

Note: When you write an equation in Sage, you have to use two equal signs, since a single equal sign is for assignment.

Solve $x^2 - 10 = 2x - x^2$ by graphing this equation.

47 `plot(x^2-10==2*x-x^2,xmin=-5,xmax=5)`
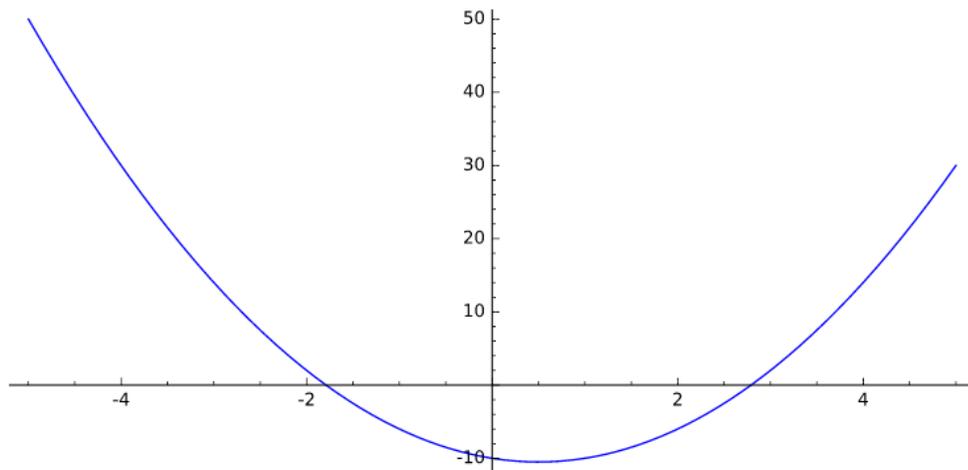
If we do the plot this way, then the solutions to our equation are the roots (zeros) of the graph. In other words, solving $f(x) = g(x)$ is equivalent to solving $f(x) - g(x) = 0$.

48

Using the solve Command

Sage has a command called "solve" that can solve many (but not all) equations. This command takes two arguments: the equation to solve and the variable to solve for.

Don't forget two equal signs when you type an equation.

Example 17

Solve $x^2 - 10 = 2x - x^2$ using the solve command.

```
49  solve(x^2-10==2*x-x^2,x)   #You could also type solve(f(x)==g(x),x)
50  show(_)
```

```
[x == -1/2*sqrt(21) + 1/2, x == 1/2*sqrt(21) + 1/2]
```

$$[x = -\frac{1}{2}\sqrt{21} + \frac{1}{2}, x = \frac{1}{2}\sqrt{21} + \frac{1}{2}]$$

There are two solutions, $-\frac{1}{2}\sqrt{21} + \frac{1}{2}$ and $\frac{1}{2}\sqrt{21} + \frac{1}{2}$.

Let's convert to decimals.

```
51  N(1/2*sqrt(21) + 1/2)
52  N(-1/2*sqrt(21) + 1/2)
```

```
2.79128784747792
-1.79128784747792
```

53

Example 18

Solve for x: $3x^2 + 2x - 4 = -x^3 + 5x^2 - 8x + 7$.

```
54  solve(3*x^2+2*x-4==-x^3+5*x^2-8*x+7,x)
```

```
[x == -1/2*(1/18*sqrt(3259)*sqrt(3) + 133/54)^(1/3)*(I*sqrt(3) + 1) + 1/9*(-13*I*sqrt(3) + 13)/(1/18*sqrt(3259)*sqrt(3) + 133/54)^(1/3) +
2/3, x == -1/2*(1/18*sqrt(3259)*sqrt(3) + 133/54)^(1/3)*(-I*sqrt(3) + 1) - 1/9*(-13*I*sqrt(3) - 13)/(1/18*sqrt(3259)*sqrt(3) +
133/54)^(1/3) + 2/3, x == (1/18*sqrt(3259)*sqrt(3) + 133/54)^(1/3) - 26/9/(1/18*sqrt(3259)*sqrt(3) + 133/54)^(1/3) + 2/3]
```

```
55  show(_[0]);show(_[1]);show(_[2])
```

Sage found three solutions, but two are complex; there is only one real solution.

```
56  N(-1/2*(1/18*sqrt(3259)*sqrt(3) + 133/54)^(1/3)*(I*sqrt(3) + 1) + 1/9*(-13*I*sqrt(3) + 13)/(1/18*sqrt(3259)*sqrt(3) + 133/54)^(1/3) + 2/:
57  N(-1/2*(1/18*sqrt(3259)*sqrt(3) + 133/54)^(1/3)*(-I*sqrt(3) + 1) - 1/9*(-13*I*sqrt(3) - 13)/(1/18*sqrt(3259)*sqrt(3) + 133/54)^(1/3) + 2/
58  N((1/18*sqrt(3259)*sqrt(3) + 133/54)^(1/3) - 26/9/(1/18*sqrt(3259)*sqrt(3) + 133/54)^(1/3) + 2/3)
```

```
0.392038325748806 - 2.98210141403215*I
0.392038325748806 + 2.98210141403215*I
1.21592334850239
```

Here is a graph of the two sides of this equation showing one real solution near $x \approx 1.2$.

```
59  plot(3*x^2+2*x-4,xmin=-7,xmax=7)+plot(-x^3+5*x^2-8*x+7,xmin=-7,xmax=7,ymin=-50,ymax=100,color='red')
```
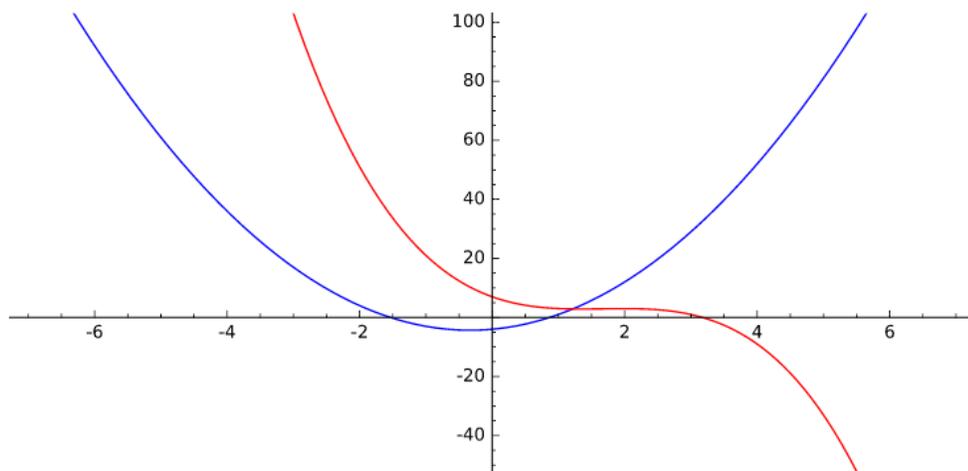


```
60
```

Using the find_root Command

There are some equations that Sage cannot solve exactly using the solve command. In this case, we can approximate the solutions using the find_root command.

```
61
```

Example 19

Solve for x: $\cos(x) = x$.

First, we will try the solve command.

```
62  solve(cos(x)==x,x)
```

```
[x == cos(x)]
```

Sage returns the original equation again; it was unable to find exact answers. However, we can find a numerial approximation using the find_root command.

The find_root command requires that we specify an interval to search for a solution. It will return the first solution it finds, so we have to make sure that our interval contains only one solution. One way to do this is by graphing first.

```
63  plot(cos(x))+plot(x,color='red')
```
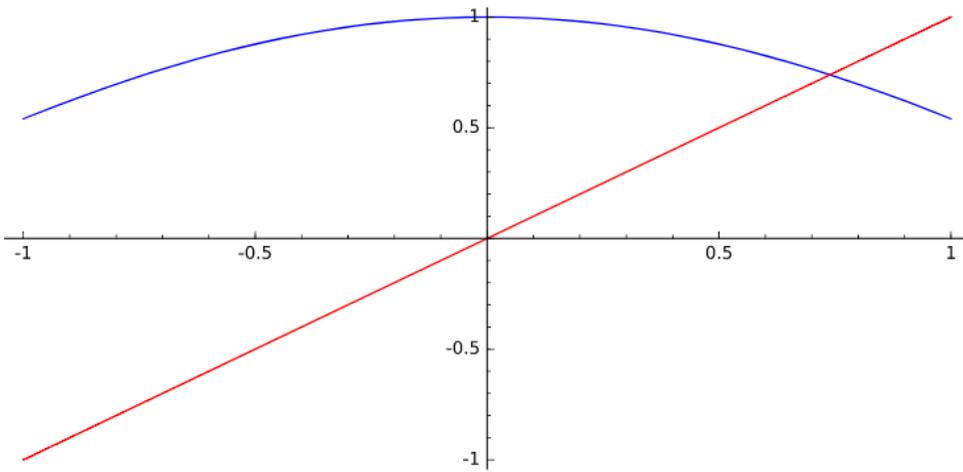
64

Here we have only one solution, near 0.75.

[Note: How do we know there are no more solutions outside this viewing window?]

The find_root command takes three arguments: an equation to solve, a lower bound, and an upper bound. It will search for solutions between the lower and upper bounds, ar it will return the first solution it finds.

65
```
find_root(cos(x)==x,-1,1)
```
```
0.7390851332151607
```

This command looks for solutions to the equation $\cos(x) = x$ on the interval $[-1, 1]$.

Notice that find_root returns a numerical approximation, while solve returns an exact answer.

66

Example 20

Solve for x: $e^x - 1.1 = \sin(x)$.

First, we'll try the solve command.

67
```
solve(e^x-1.1==sin(x),x)
```
```
[sin(x) == e^x - 11/10]
```

68

The solve command is not able to solve this equation (notice that there are still x's on both sides of the output), so we'll have to use find_root instead.

What interval should we give find_root? Let's try the interval from -10 to 10 just to see what happens:

69
```
find_root(e^x-1.1==sin(x),-10,10)
```
```
-1.891197831974637
```

The find_root command returns only one solution. It gives us no clue whether or not there are more solutions in this interval. To use find_root effectively, we need to know hc many solutions there are and their approximate location. So we'll look at a graph. It may take some trial and error to get a good window.

70
```
plot(e^x-1.1,xmin=-3,xmax=2)+plot(sin(x),xmin=-3,xmax=2,ymax=1,color='red')
```
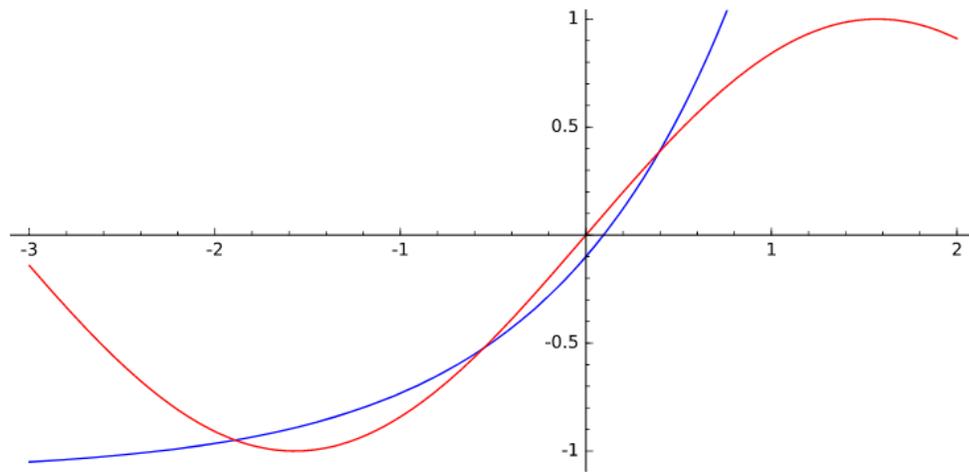
71

From the graph, we can see three solutions (near $x = -2$, $x = -0.5$, and $x = 0.5$). We'll use this to determine the intervals to give find_root.

[Note: How do we know there are no more solutions outside this viewing window?]

```
72  find_root(e^x-1.1==sin(x),-3,-1)
73  find_root(e^x-1.1==sin(x),-1,0)
74  find_root(e^x-1.1==sin(x),0,1)
```

```
-1.8911978319747127
-0.5513169357098279
0.395754320029599
```

Equations Involving Multiple Variables

You can also use the solve command to solve for one variable in an equation involving multiple variables. The answer will be an expression involving the other variables.

Note: find_root does not work with multiple variables, since the answer must be a number to use find_root.

75

Example 21

Solve $xy + 2 = 2x - 3y$ for y.

Don't forget to declare your variables.

```
76  %var y
77  solve(x*y+2==2*x-3*y,y)
78  show(_)
```

```
[y == 2*(x - 1)/(x + 3)]
```

$$[y = \frac{2\,(x-1)}{x+3}]$$

79

Now let's solve the same equation for x.

```
80  solve(x*y+2==2*x-3*y,x)
81  show(_)
```

```
[x == -(3*y + 2)/(y - 2)]
```

$$[x = -\frac{3\,y+2}{y-2}]$$

82

Example 22

If you forget the Quadratic Formula, then Sage will remind you (notice that you can leave off ==0).

Solve for x: $ax^2 + bx + c = 0$.

```
83  %var a,b,c
84  solve(a*x^2+b*x+c,x)
85  show(_)
```

```
[x == -1/2*(b + sqrt(b^2 - 4*a*c))/a, x == -1/2*(b - sqrt(b^2 - 4*a*c))/a]
```

$$[x = -\frac{b+\sqrt{b^2-4\,ac}}{2\,a},\, x = -\frac{b-\sqrt{b^2-4\,ac}}{2\,a}]$$

Solving Inequalities

You can also use Sage to solve inequalities, although we won't be needing this feature as much.

86

Example 23

Solve $x^2 - 8 > 3$.

```
87  solve(x^2-8>3,x)
88  show(_)
```

```
[[x < -sqrt(11)], [x > sqrt(11)]]
```

$$[[x < -\sqrt{11}], [x > \sqrt{11}]]$$

In interval notation, the solution is $(-\infty, -\sqrt{11}) \cup (\sqrt{11}, \infty)$.

Here is a graph:

```
89  plot(x^2-8,xmin=-5,xmax=5)+plot(3,xmin=-5,xmax=5,color='red',linestyle='dashed')
```
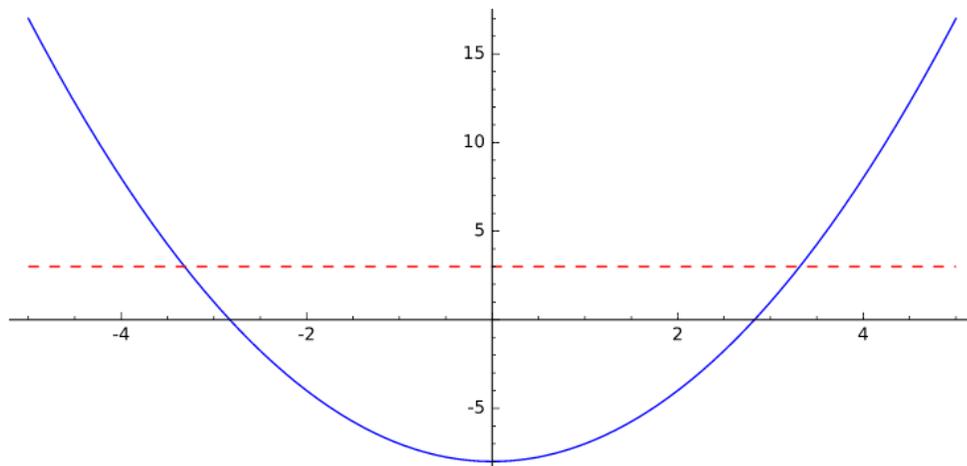

90

Example 24

Solve $x^3 - 5x \le 2$.

[Note: use "<=" for $\le$]

```
91  solve(x^3-5*x<=2,x)
92  show(_)
```

```
[[x <= -2], [x >= -sqrt(2) + 1, x <= sqrt(2) + 1]]
```

$$[[x \le (-2)], [x \ge -\sqrt{2} + 1, x \le \sqrt{2} + 1]]$$

```
93
```

In interval notation, the solution is $(-\infty, -2] \cup [-\sqrt{2} + 1, \sqrt{2} + 1]$.

A picture would be nice to help us interpret the output, so here's a graph:

```
94  plot(x^3-5*x,xmin=-4,xmax=4,ymin=-10,ymax=10)+plot(2,xmin=-4,xmax=4,color='red',linestyle='dashed')
```
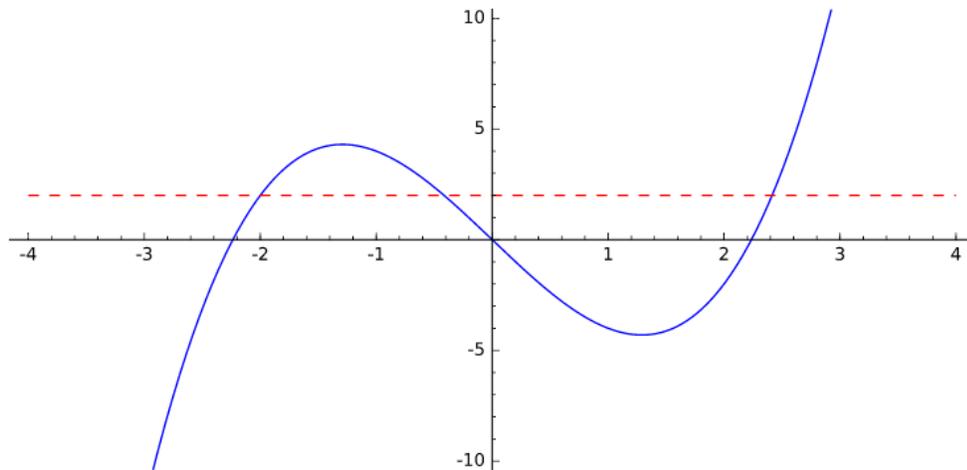

You have to be careful reading this solution. Notice that [[x <= -2], [x >= -sqrt(2) + 1, x <= sqrt(2) + 1]] is *not* the same as [[x <= -2], [x >= -sqrt(2) + 1], [x <= sqrt(2) + 1]].

The latter is $(-\infty, -2] \cup [-\sqrt{2} + 1, \infty) \cup (-\infty, \sqrt{2} + 1]$, which does not make sense anyway, because $[-\sqrt{2} + 1, \infty) \cup (-\infty, \sqrt{2} + 1] = \mathbb{R}$