Introduction to pgfplots

Martin Maina

JKUAT

May 12, 2023

Introduction

The pgfplots package, based on TikZ, is a powerful tool for creating scientific/technical graphics. It simplifies the process by allowing you to provide the input data or formula, and pgfplots takes care of the rest.

Document Preamble

To use `pgfplots` in your document, add the following line to your preamble:

```
\usepackage{pgfplots}
```

You can also configure the behavior of `pgfplots` in the preamble. For example, to change the size of each plot and ensure backward compatibility, add the following line:

```
\pgfplotsset{width=10cm,compat=1.9}
```

This sets the size of each `pgfplots` figure to 10 centimeters. You can use different units such as pt, mm, or in.

Compilation Time

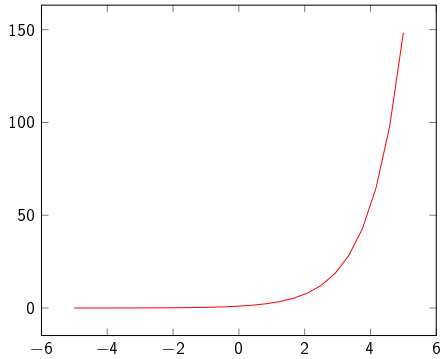
When the original TeX engine was created, it wasn't designed for direct production of graphics. The introduction of pdfTeX allowed for direct graphics creation using built-in TeX language commands. This led to the development of LaTeX graphics packages like TikZ and `pgfplots`.
However, the processing of high-level LaTeX graphics commands into low-level pdfTeX commands can take time. Documents with multiple `pgfplots` figures or complex graphics may have longer compilation times.

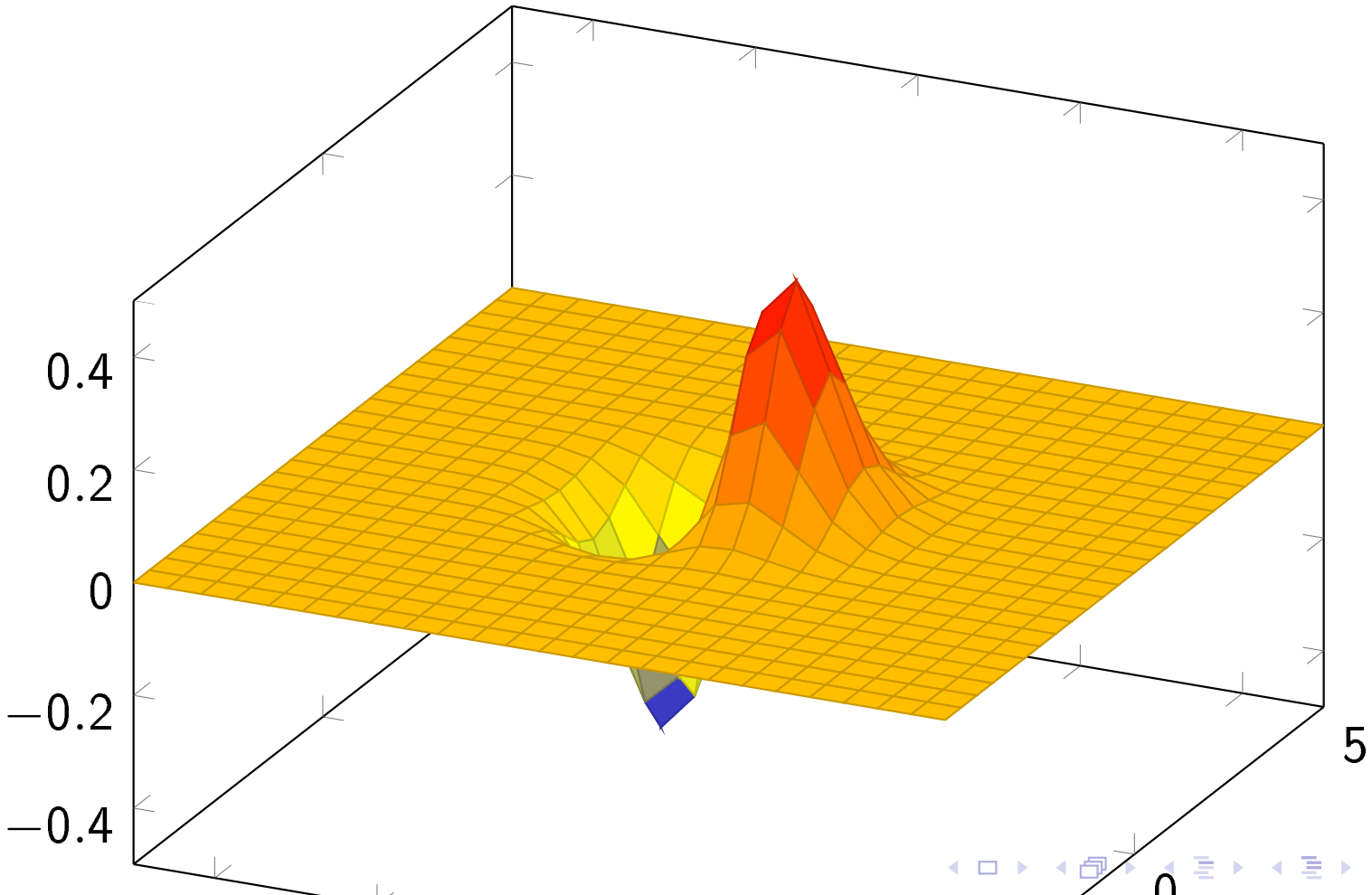
Reducing Compilation Time

To increase the speed of document compilation, you can configure `pgfplots` to export the figures to separate PDF files and then import them into the document. This allows you to compile once and reuse the figures.
To enable this, add the following code to the preamble:

```
\usepgfplotslibrary{external}
\tikzexternalize
```

Basic Example (Externalizing Figures)





Plotting from Data: Explanation

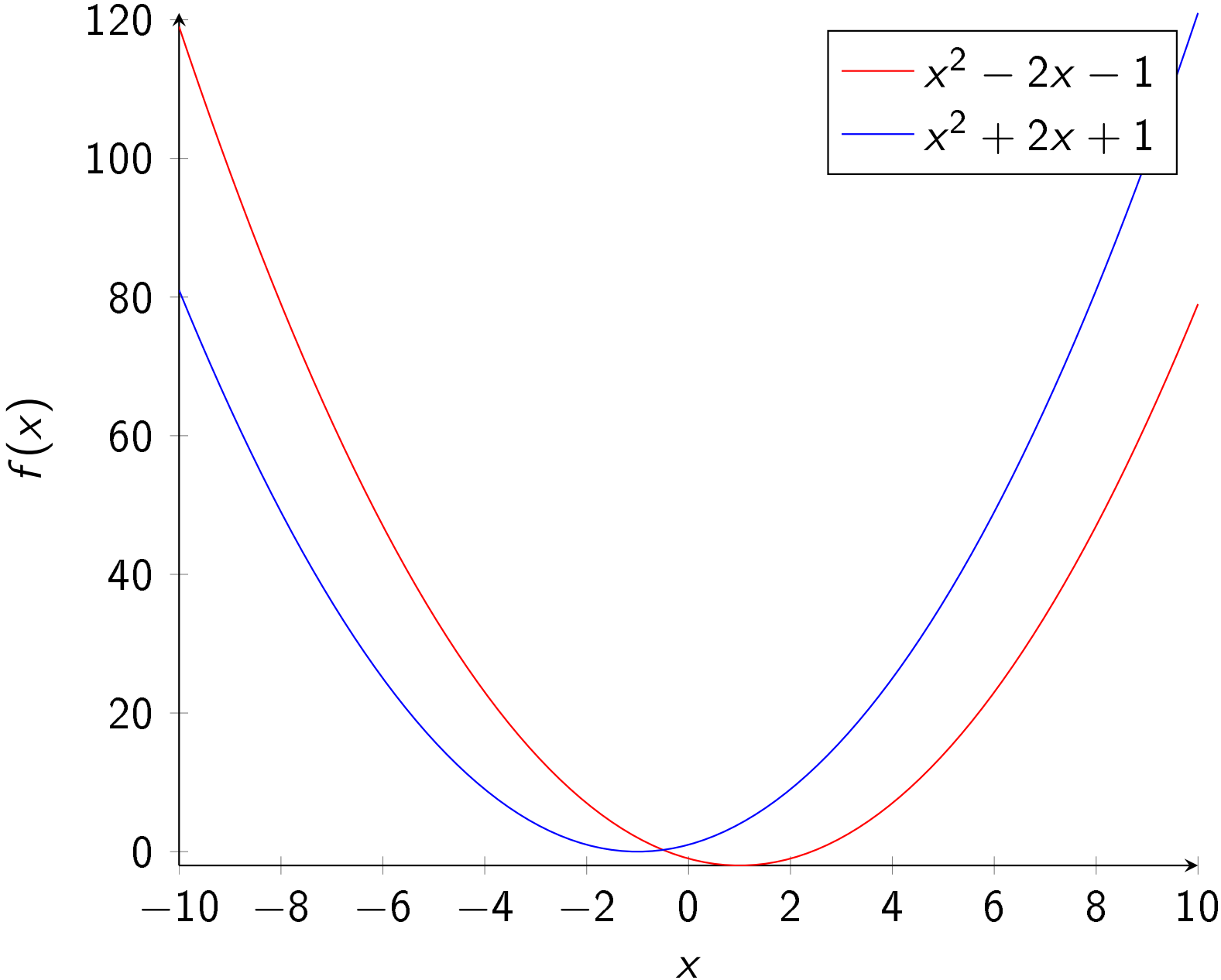
- The code snippet demonstrates how to plot data with `pgfplots`.
- The plot showcases the temperature dependence of $CuSO_4 \cdot 5H_2O$ solubility.
- The `title` command sets the title of the plot.
- The `xlabel` and `ylabel` specify the labels for the x-axis and y-axis, respectively.
- The `xmin`, `xmax`, `ymin`, and `ymax` options define the range of the x-axis and y-axis.
- The `xtick` and `ytick` parameters determine the tick positions on the x-axis and y-axis.
- The `legend pos` sets the position of the legend box (northwest in this case)

Plotting from Data: Explanation

- `ymajorgrids` enables the grid lines on the y-axis, and `grid style` determines their style.
- The `color` option sets the color of the plotted line, and `mark` specifies the shape of the data points (squares in this case).
- The `coordinates` section contains the data points to be plotted.
- The `\legend` command adds a legend entry for the plot.

This example demonstrates how to plot data from coordinates using `pgfplots`. It provides a clear visual representation of the temperature dependence of $CuSO_4 \cdot 5H_2O$ solubility.

2D Plots



Plotting from Data

```
\begin{figure}
  \centering
  \begin{tikzpicture}
    \begin{axis}[
      title={Temperature dependence of
 CuSO\(_4\cdot\)5H\(_2\)O solubility},
      xlabel={Temperature [\textcelsius]},
      ylabel={Solubility [g per 100 g water]},
      xmin=0, xmax=100,
      ymin=0, ymax=120,
      xtick={0,20,40,60,80,100},
      ytick={0,20,40,60,80,100,120},
      legend pos=north west,
      ymajorgrids=true,
      grid style=dashed,
    ]
\end{figure}
```

```
\begin{figure}
\centering
 \addplot[
      color=blue,
      mark=square,
    ]
    coordinates {
      (0,23.1)(10,27.5)(20,32)(30,37.8)
(40,44.6)(60,61.8)(80,83.8)(100,114)
    };
    \legend{CuSO\(_4\cdot\)5H\(_2\)O}

    \end{axis}
  \end{tikzpicture}
  \caption{Temperature dependence of
CuSO\(_4\cdot\)5H\(_2\)O solubility}
\end{figure}
```

Plotting from Data: Explanation

- The code snippet demonstrates how to plot data with ‘pgfplots‘.
- The plot showcases the temperature dependence of $CuSO_4 \cdot 5H_2O$ solubility.
- The ‘title‘ command sets the title of the plot.
- The ‘xlabel‘ and ‘ylabel‘ specify the labels for the x-axis and y-axis, respectively.
- The ‘xmin‘, ‘xmax‘, ‘ymin‘, and ‘ymax‘ options define the range of the x-axis and y-axis.
- The ‘xtick‘ and ‘ytick‘ parameters determine the tick positions on the x-axis and y-axis.
- The ‘legend pos‘ sets the position of the legend box (northwest in this case).
- ‘ymajorgrids‘ enables the grid lines on the y-axis, and ‘grid style‘ determines their style

- ‘ymajorgrids‘ enables the grid lines on the y-axis, and ‘grid style‘ determines their style.
- The ‘color‘ option sets the color of the plotted line, and ‘mark‘ specifies the shape of the data points (squares in this case).
- The ‘coordinates‘ section contains the data points to be plotted.
- The `\legend` command adds a legend entry for the plot.

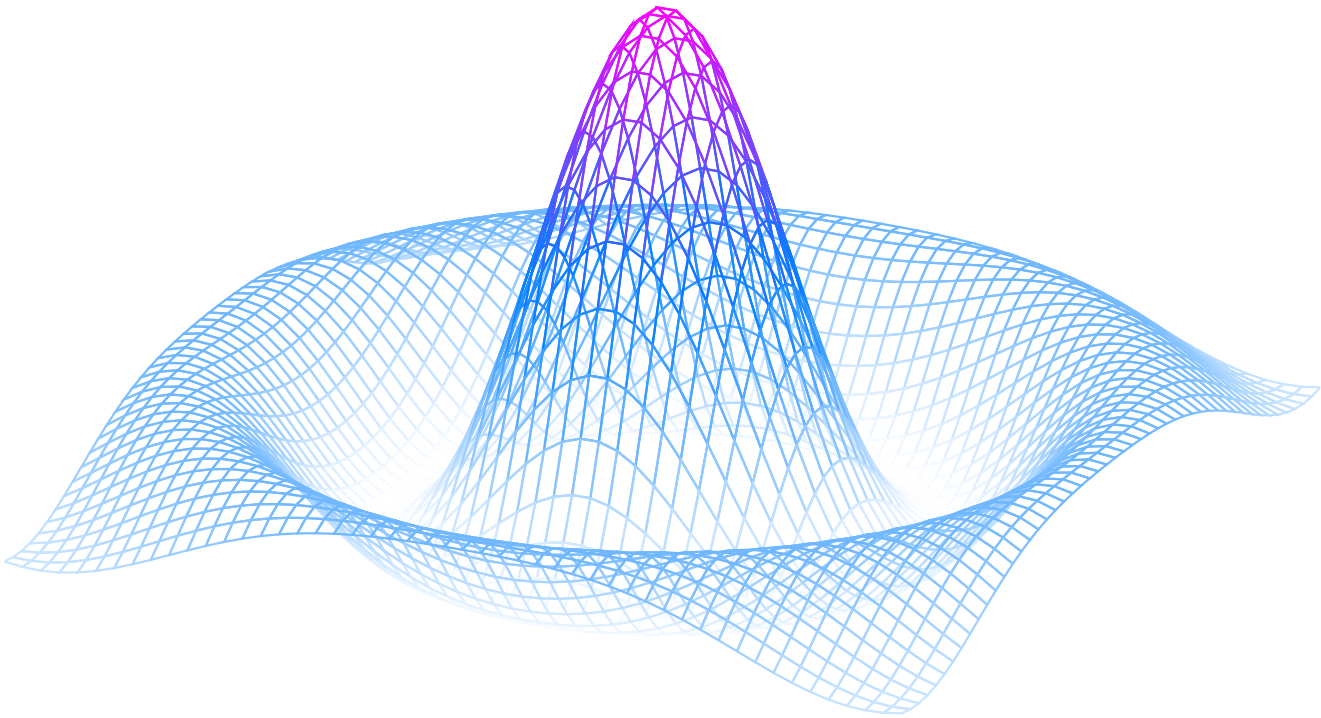
This example demonstrates how to plot data from coordinates using ‘pgfplots‘. It provides a clear visual representation of the temperature dependence of $CuSO_4 \cdot 5H_2O$ solubility.

3D Plots

pgfplots has the 3D Plotting capabilities that you may expect in a plotting software.

Plotting mathematical expressions

Example using the mesh parameter

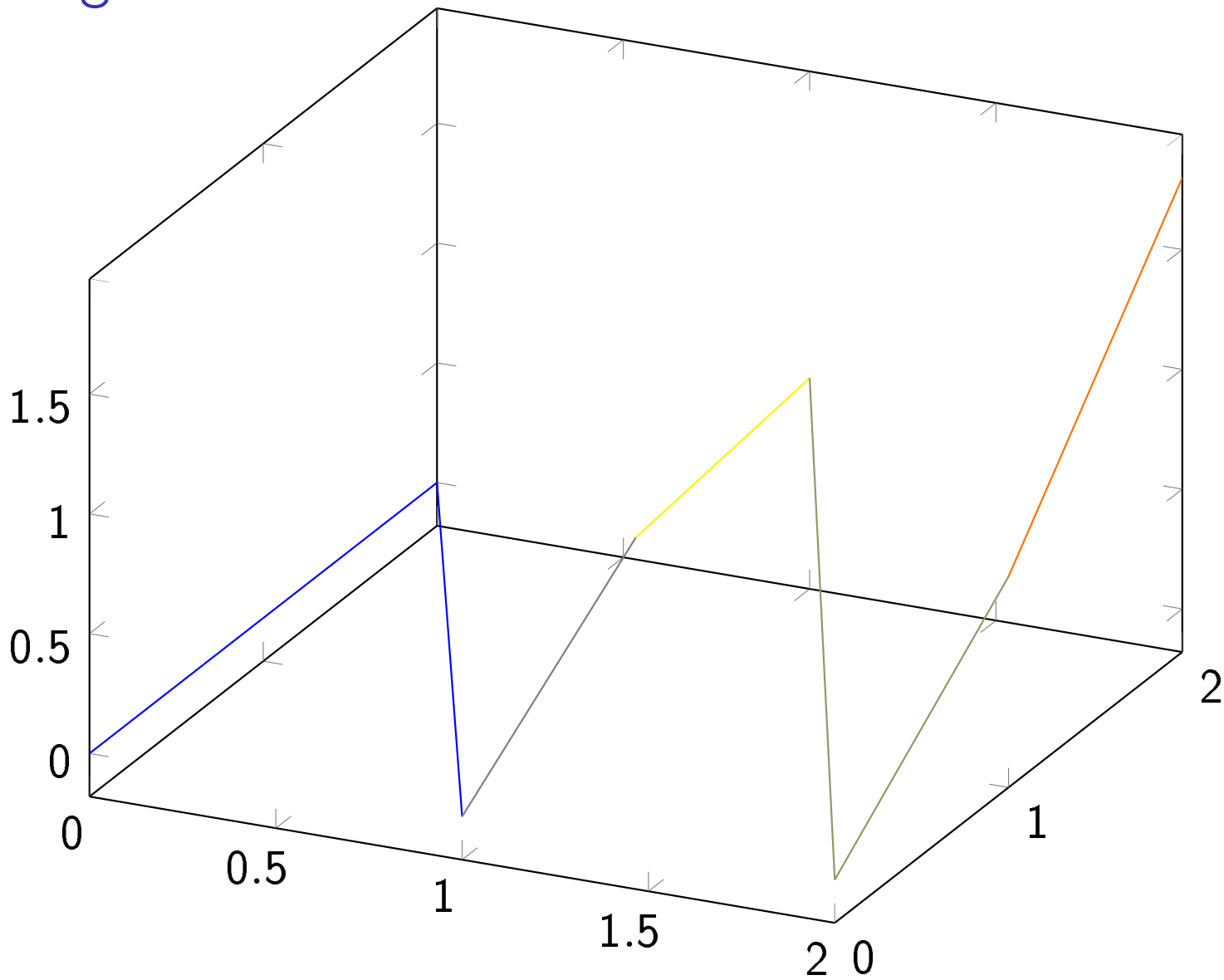


When working with trigonometric functions pgfplots uses degrees as default units, if the angle is in radians (as in this example) you have to use the deg function to convert to degrees.

Contour plots

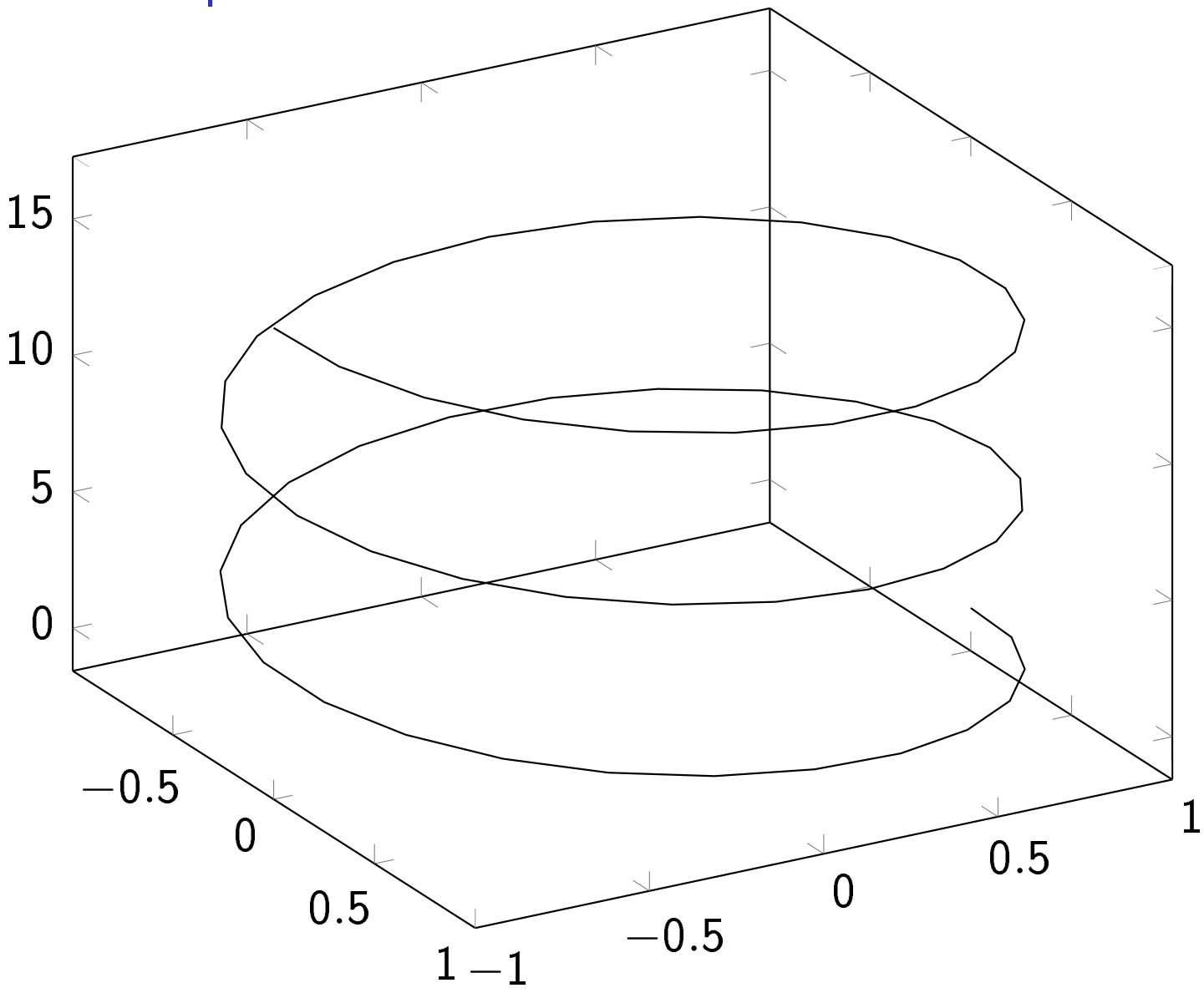
- The value of the title parameter is inside curly brackets because it contains a comma, so we use the grouping brackets to avoid any confusion with the other parameters passed to the `\begin{axis}` declaration.
- `view={0}{90}` changes the view of the plot. The first value is a rotation, in degrees, around the z-axis; the second value is to rotate the view around the x-axis. In this example when we combine a 0° rotation around the z-axis and a 90° rotation around the x-axis we end up with a view of the plot from top.
- `contour gnuplot={levels={0.8, 0.4, 0.2, -0.2}}` tells LaTeX to use the external software gnuplot to compute the contour lines; this works fine in Overleaf but if you want to use this command in your local LaTeX installation you have to install gnuplot first (matlab will also work, in such case write matlab instead of gnuplot in the command). The sub parameter levels is a list of values of elevation levels where the contour lines are to be computed.

Plotting a surface from data



The points passed to the `coordinates` parameter are treated as contained in a 3×3 matrix, using a blank line as the separator for

Parametric plot



- `samples y=0` is used to prevent pgfplots from joining the extreme points of the spiral.