

 Search help library....

Pgfpplots package

The **pgfpplots** package is a powerful tool, based on `tikz` (/learn/TikZ_package), dedicated to create scientific graphs.

Contents

- 1 Introduction
 - 2 The document preamble
 - 3 2D plots
 - 3.1 Plotting mathematical expressions
 - 3.2 Plotting from data
 - 3.3 Scatter plots
 - 3.4 Bar graphs
 - 4 3D Plots
 - 4.1 Plotting mathematical expressions
 - 4.2 Contour plots
 - 4.3 Plotting a surface from data
 - 4.4 Parametric plot
 - 5 Reference guide
 - 6 Further reading
-

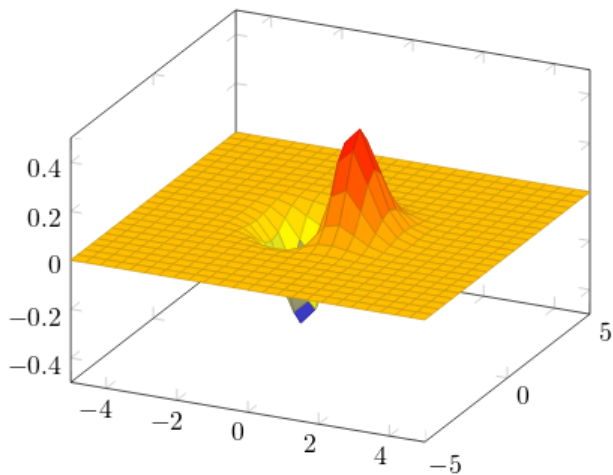
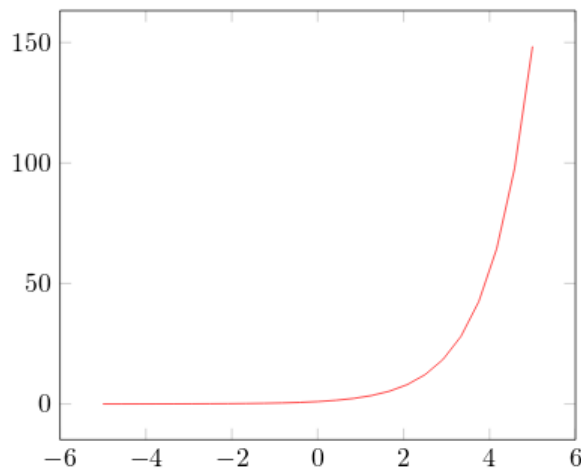
Introduction

Pgfpplots is a visualization tool to make simpler the inclusion of plots in your documents. The basic idea is that you provide the input data/formula and *pgfpplots* does the rest.

```

\begin{tikzpicture}
\begin{axis}
\addplot[color=red]{exp(x)};
\end{axis}
\end{tikzpicture}
%Here ends the first plot
\hskip 5pt
%Here begins the 3d plot
\begin{tikzpicture}
\begin{axis}
\addplot3[
    surf,
]
{exp(-x^2-y^2)*x};
\end{axis}
\end{tikzpicture}

```



(/learn/File:Pgplots3.png)

Since **pgfplot** is based on **tikz** the plot must be inside a *tikzpicture* environment. Then the environment declaration `\begin{axis}`, `\end{axis}` will set the right scaling for the plot, check the Reference guide for other axis environments.

To add an actual plot, the command `\addplot[color=red]{log(x)}`; is used. Inside the squared brackets some options can be passed, in this case we set the colour of the plot to *red*; the squared brackets are mandatory, if no options are passed leave a blank space between them. Inside the curly brackets you put the function to plot. Is important to remember that this command must end with a *semicolon*;

To put a second plot next to the first one declare a new *tikzpicture* environment. Do not insert a new line, but a small blank gap, in this case `\hskip 10pt` will insert a 10pt-wide blank space.

The rest of the syntax is the same, except for the `\addplot3 [surf,]{exp(-x^2-y^2)*x}`; . This will add a 3dplot, and the option *surf* inside squared brackets declares that it's a surface plot. The function to plot must be placed inside curly brackets. Again, don't forget to put a *semicolon*; at the end of the command.

Note: *It's recommended as a good practice to indent the code - see the second plot in the example above - and to add a comma , at the end of each option passed to \addplot. This way the code is more readable and is easier to add further options if needed.*

➡ Open an example of the pgplots package in Overleaf

([https://www.sharelatex.com/project/new/template?](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d1baf908c6363f001e35/download/zip&templateName=Pgplots%20Example&compiler=pdflatex)

[zipUrl=/project/52e9d1baf908c6363f001e35/download/zip&templateName=Pgplots%20Example&compiler=pdflatex](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d1baf908c6363f001e35/download/zip&templateName=Pgplots%20Example&compiler=pdflatex))

The document preamble

To include **pgfplots** in your document is very easy, add the next line to your preamble and that's it:

```
\usepackage{pgfplots}
```

Some additional tweaking for this package can be made in the preamble. To change the size of each plot and also guarantee backwards compatibility (recommended) add the next line:

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

This changes the size of each *pgfplot* figure to 10 centimeters, which is huge; you may use different units (pt, mm, in). The *compat* parameter is for the code to work on the package version 1.9 or later.

Since L^AT_EX was not initially conceived with plotting capabilities in mind, when there are several *pgfplot* figures in your document or they are very complex, it takes a considerable amount of time to render them. To improve the compiling time you can configure the package to export the figures to separate PDF files and then import them into the document, add the code shown below to the preamble:

```
\usepgfplotslibrary{external}
```

```
\tikzexternalize
```

See this help article

([/learn/Questions/I_have_a_lot_of_tikz_matlab2tikz_or_pgplots_figures_so_I%27m_getting_a_compilation_timeout._Can_I_externalise_my_figures%3F](#)) for further details on how to set up tikz-externalization in your Overleaf project.

➡ Open an example of the pgfplots package in Overleaf

([https://www.sharelatex.com/project/new/template?](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d1baf908c6363f001e35/download/zip&templateName=Pgfplots%20Example&compiler=pdflatex)

[zipUrl=/project/52e9d1baf908c6363f001e35/download/zip&templateName=Pgfplots%20Example&compiler=pdflatex](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d1baf908c6363f001e35/download/zip&templateName=Pgfplots%20Example&compiler=pdflatex))

2D plots

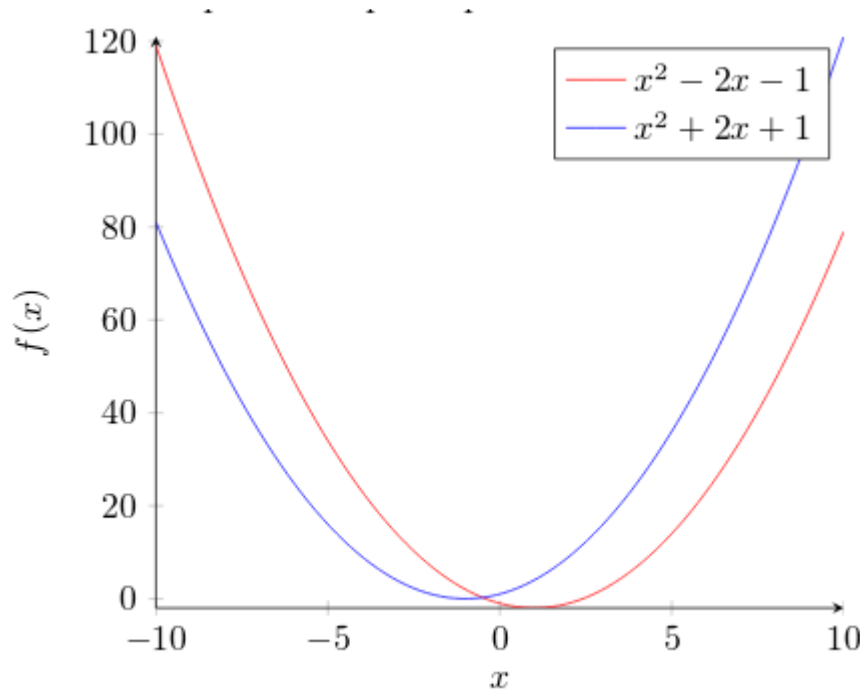
Pgfplots 2D plotting functionalities are vast, you can personalize your plots to look exactly what you want. Nevertheless, the default options usually give very good result, so all you have to do is feed the data and \LaTeX will do the rest:

Plotting mathematical expressions

To plot mathematical expressions is really easy:

```
\begin{tikzpicture}
\begin{axis}[
    axis lines = left,
    xlabel =  $x$ ,
    ylabel =  $f(x)$ ,
]
%Below the red parabola is defined
\addplot [
    domain=-10:10,
    samples=100,
    color=red,
]
{ $x^2 - 2x - 1$ };
\addlegendentry{ $x^2 - 2x - 1$ }
%Here the blue parabola is defined
\addplot [
    domain=-10:10,
    samples=100,
    color=blue,
]
{ $x^2 + 2x + 1$ };
\addlegendentry{ $x^2 + 2x + 1$ }

\end{axis}
\end{tikzpicture}
```



(/learn/File:Pgplots4.png)

Let's analyse the new commands line by line:

axis lines = left.

This will set the axis only on the left and bottom sides of the plot, instead of the default *box*.

Further customisation options at the reference guide.

xlabel = x and ylabel = $f(x)$.

Self-explanatory parameter names, these will let you put a label on the horizontal and vertical axis. Notice the *ylabel* value in between curly brackets, this brackets tell pgfplots how to group the text. The *xlabel* could have had brackets too. This is useful for complicated labels that may confuse pgfplot.

\addplot.

This will add a plot to the axis, general usage was described at the introduction. There are two new parameters in this example.

domain=-10:10.

This establishes the range of values of x .

samples=100.

Determines the number of points in the interval defined by *domain*. The greater the value of *samples* the sharper the graph you get, but it will take longer to render.

\addlegendentry{ $x^2 - 2x - 1$ }.

This adds the legend to identify the function $x^2 - 2x - 1$.

To add another graph to the plot just write a new `\addplot` entry.

➡ Open an example of the pgfplots package in Overleaf

([https://www.sharelatex.com/project/new/template?](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d283f908c6363f001e3d/download/zip&templateName=Maths%20Expression%20d&compiler=pdflatex)

[zipUrl=/project/52e9d283f908c6363f001e3d/download/zip&templateName=Maths%20Expression%20d&compiler=pdflatex](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d283f908c6363f001e3d/download/zip&templateName=Maths%20Expression%20d&compiler=pdflatex))

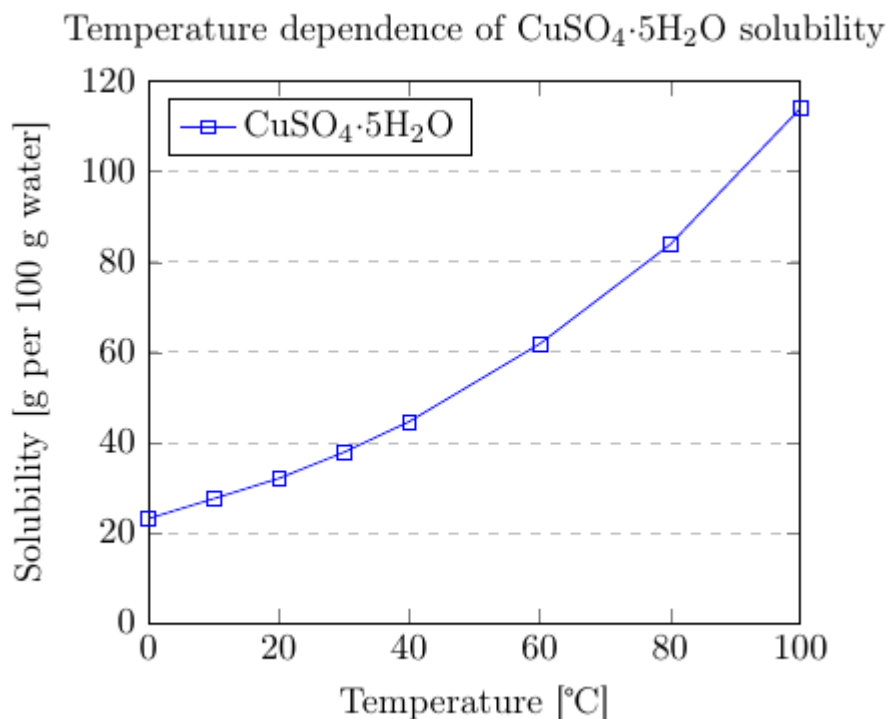
Plotting from data

Scientific research often yields data that has to be analysed. The next example shows how to plot data with *pgfplots*:

```
\begin{tikzpicture}
\begin{axis}[
    title={Temperature dependence of CuSO4·5H2O
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,
]

\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{CuSO4·5H2O}

\end{axis}
\end{tikzpicture}
```



(/learn/File:Pgplots2.png)

There are some new commands and parameters here:

title={Temperature dependence of $\text{CuSO}_4 \cdot 5\text{H}_2\text{O}$ solubility}.

As you might expect, assigns a title to the figure. The title will be displayed above the plot.

xmin=0, xmax=100, ymin=0, ymax=120.

Minimum and maximum bounds of the x and y axes.

xtick={0,20,40,60,80,100}, ytick={0,20,40,60,80,100,120}.

Points where the marks are placed. If empty the ticks are set automatically.

legend pos=north west.

Position of the legend box. Check the reference guide for more options.

ymajorgrids=true.

This Enables/disables grid lines at the tick positions on the y axis. Use `xmajorgrids` to enable grid lines on the x axis.

grid style=dashed.

Self-explanatory. To display dashed grid lines.

mark=square.

This draws a squared mark at each point in the *coordinates* array. Each mark will be connected with the next one by a straight line.

coordinates `{(0,23.1)(10,27.5)(20,32) ...}`

Coordinates of the points to be plotted. This is the data you want analyse graphically.

If the data is in a file, which is the case most of the time; instead of the commands `\addplot` and `coordinates` you should use `\addplot table {file_with_the_data.dat}`, the rest of the options are valid in this environment.

➡ Open an example of the pgfplots package in Overleaf

([https://www.sharelatex.com/project/new/template?](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d314f908c6363f001e44/download/zip&templateName=Data%20Plot%202d&compiler=pdflatex)

[zipUrl=/project/52e9d314f908c6363f001e44/download/zip&templateName=Data%20Plot%202d&compiler=pdflatex](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d314f908c6363f001e44/download/zip&templateName=Data%20Plot%202d&compiler=pdflatex))

Scatter plots

Scatter plots are used to represent information by using some kind of marks, these are common, for example, when computing statistical regression. Lets start with some data, the sample below is to show the structure of the data file we are going to plot (see the end of this section for a link to the \LaTeX source and the data file):

GPA ma ve co un

3.45 643 589 3.76 3.52

2.78 558 512 2.87 2.91

2.52 583 503 2.54 2.4

3.67 685 602 3.83 3.47

3.24 592 538 3.29 3.47

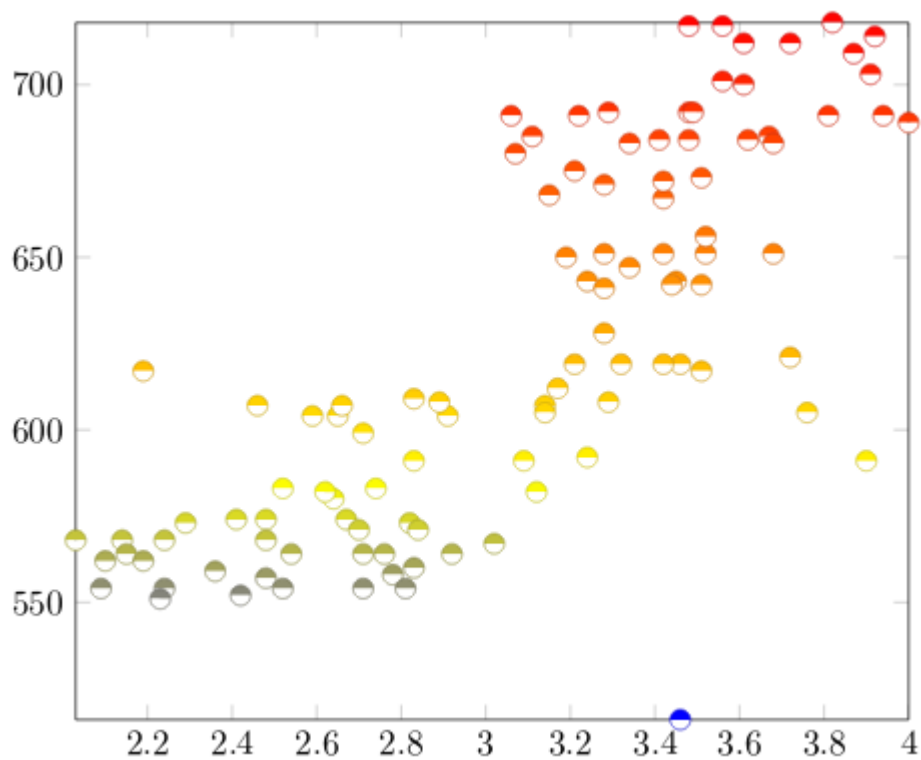
2.1 562 486 2.64 2.37

The next example is a scatter plot of the first two columns in this table:


```

\begin{tikzpicture}
\begin{axis}[
  enlargelimits=false,
]
\addplot+[
  only marks,
  scatter,
  mark=halfcircle*,
  mark size=2.9pt]
table[meta=ma]
{scattered_example.dat};
\end{axis}
\end{tikzpicture}

```



(/learn/File:Scattered.png)

The parameters passed to the *axis* and *addplot* environments can also be used in a data plot, except for *scatter*. Below the description of the code:

enlarge limits=false

This will shrink the axes so the point with maximum and minimum values lay on the edge of the plot.

only marks

Really explicit, will put a mark on each point.

scatter

When *scatter* is used the points are coloured depending on a value, the colour is given by the *meta* parameter explained below.

mark=halfcircle*

The kind of mark to use on each point, check the reference guide for a list of possible values.

mark size=2.9pt

The size of each mark, different units can be used.

table[meta=ma]{scattered_example.dat};

Here the *table* command tells latex that the data to be plotted is in a file. The *meta=ma* parameter is passed to choose the column that determines the colour of each point. Inside curly brackets is the name of the data file.

➡ Open an example of the pgfplots package in Overleaf

([https://www.sharelatex.com/project/new/template?](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9bb56f908c6363f001dc7/download/zip&templateName=Scatter%20Plot%20Example&compiler=pdflatex)

[zipUrl=/project/52e9bb56f908c6363f001dc7/download/zip&templateName=Scatter%20Plot%20Example&compiler=pdflatex](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9bb56f908c6363f001dc7/download/zip&templateName=Scatter%20Plot%20Example&compiler=pdflatex))

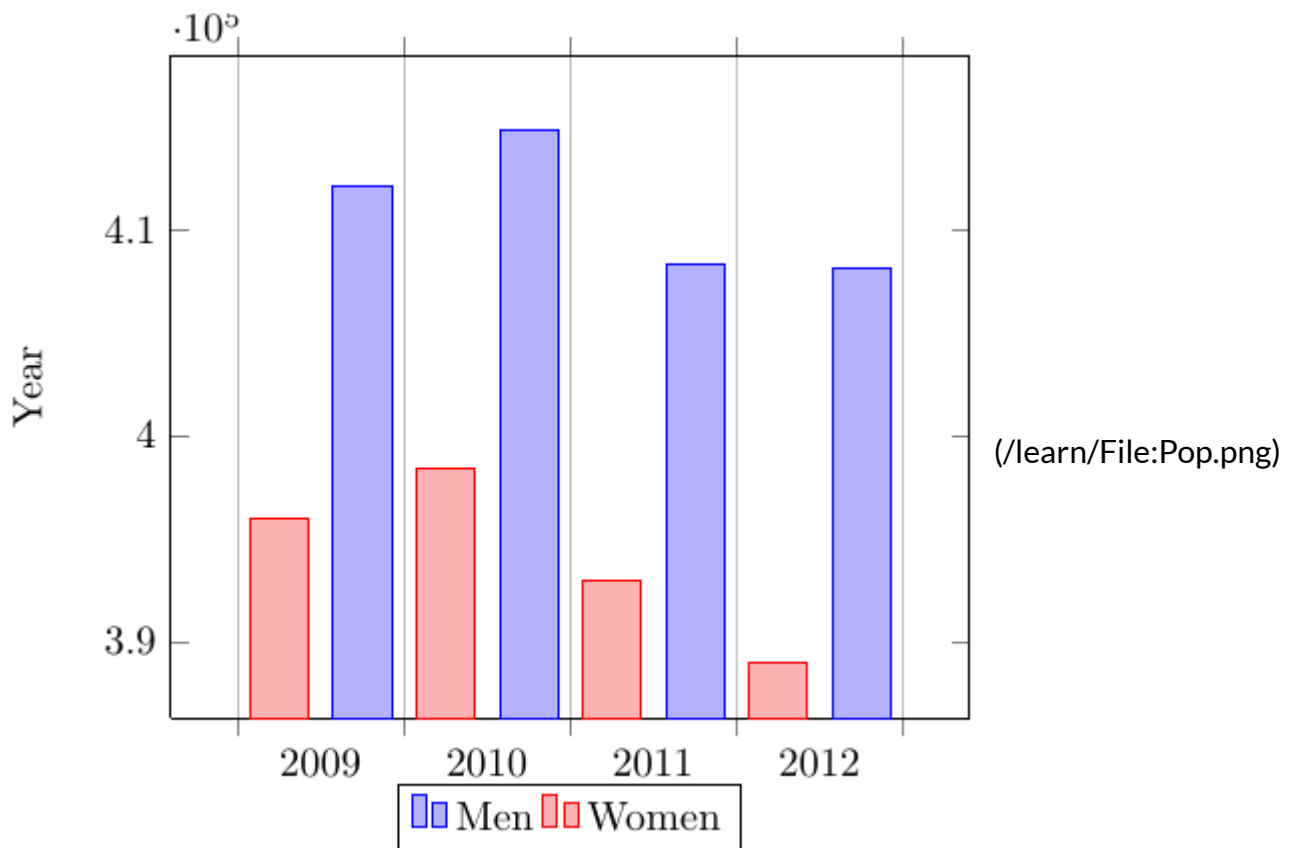
Bar graphs

Bar graphs (also known as bar charts and bar plots) are used to display gathered data, mainly statistical data about a population of some sort. Bar plots in **pgfplots** are highly customisable, but here we are going to show an example that 'just works':

```

\begin{tikzpicture}
\begin{axis}[
  x tick label style={
    /pgf/number format/1000 sep=},
  ylabel=Year,
  enlargelimits=0.05,
  legend style={at={(0.5,-0.1)},
    anchor=north,legend columns=-1},
  ybar interval=0.7,
]
\addplot
  coordinates {(2012,408184) (2011,408348)
    (2010,414870) (2009,412156)};
\addplot
  coordinates {(2012,388950) (2011,393007)
    (2010,398449) (2009,395972)};
\legend{Men,Women}
\end{axis}
\end{tikzpicture}

```



The figure starts with the already explained declaration of the *tikzpicture* and *axis* environments, but the *axis* declaration has a number of new parameters:

x tick label style={/pgf/number format/1000 sep=}

This piece of code defines a complete style for the plot. With this style you may include several `\addplot` commands within this *axis* environment, they will fit and look nice together with no further tweaks (the `ybar` parameter described below is mandatory for this to work).

`enlargetimits=0.05.`

Enlarging the limits in a bar plot is necessary because these kind of plots often require some extra space above the bar to look better and/or add a label. Then number 0.05 is relative to the total height of of the plot.

`legend style={at={(0.5,-0.2)}, anchor=north,legend columns=-1}`

Again, this will work just fine most of the time. If anything, change the value of -0.2 to locate the legend closer/farther from the x-axis.

`ybar interval=0.7,`

Thickness of each bar. 1 meaning the bars will be one next to the other with no gaps and 0 meaning there will be no bars, but only vertical lines.

The `coordinates` in this kind of plot determine the base point of the bar and its height.

The labels on the y-axis will show up to 4 digits. If in the numbers you are working with are greater than 9999 **pgfplot** will use the same notation as in the example.

➡ Open an example of the pgfplots package in Overleaf

([https://www.sharelatex.com/project/new/template?](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d42df908c6363f001e47/download/zip&templateName=Bar%20Chart&compiler=pdflatex)

[zipUrl=/project/52e9d42df908c6363f001e47/download/zip&templateName=Bar%20Chart&compiler=pdflatex](https://www.sharelatex.com/project/52e9d42df908c6363f001e47/download/zip&templateName=Bar%20Chart&compiler=pdflatex))

3D Plots

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

Plotting mathematical expressions

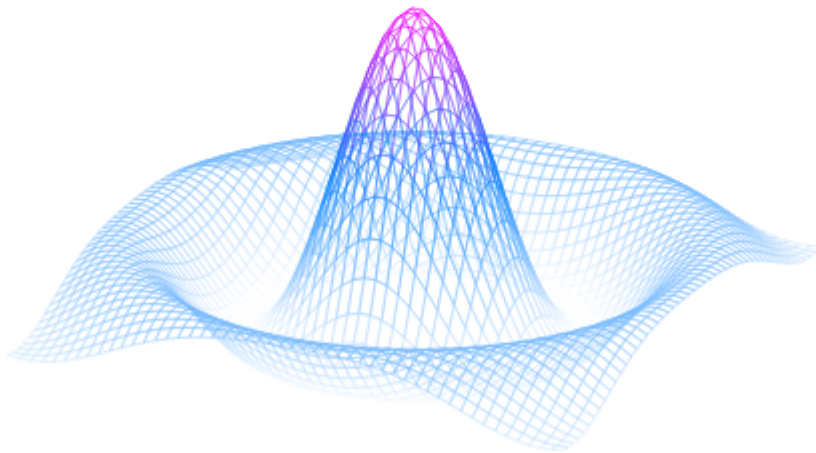
There's a simple example about this at the introduction, let's work on something slightly more complex:

```

\begin{tikzpicture}
\begin{axis}[
  title=Exmple using the mesh parameter,
  hide axis,
  colormap/cool,
]
\addplot3[
  mesh,
  samples=50,
  domain=-8:8,
]
{sin(deg(sqrt(x^2+y^2)))/sqrt(x^2+y^2)};
\addlegendentry{$\frac{\sin(r)}{r}$}
\end{axis}
\end{tikzpicture}

```

Exmple using the mesh parameter



(/learn/File:Pgplots3d1.png)

Most of the commands here have already been explained, but there are 3 new things:

hide axis

This option in the *axis* environment is self descriptive, the axis won't be shown.

colormap/cool

Is the colour scheme to be used in the plot. Check the reference guide for more colour schemes.

mesh

This option is self-descriptive too, check also the *surf* parameter in the introductory example.

Note: *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 `deg` function to convert to degrees.*

➞ Open an example of the pgfplots package in Overleaf

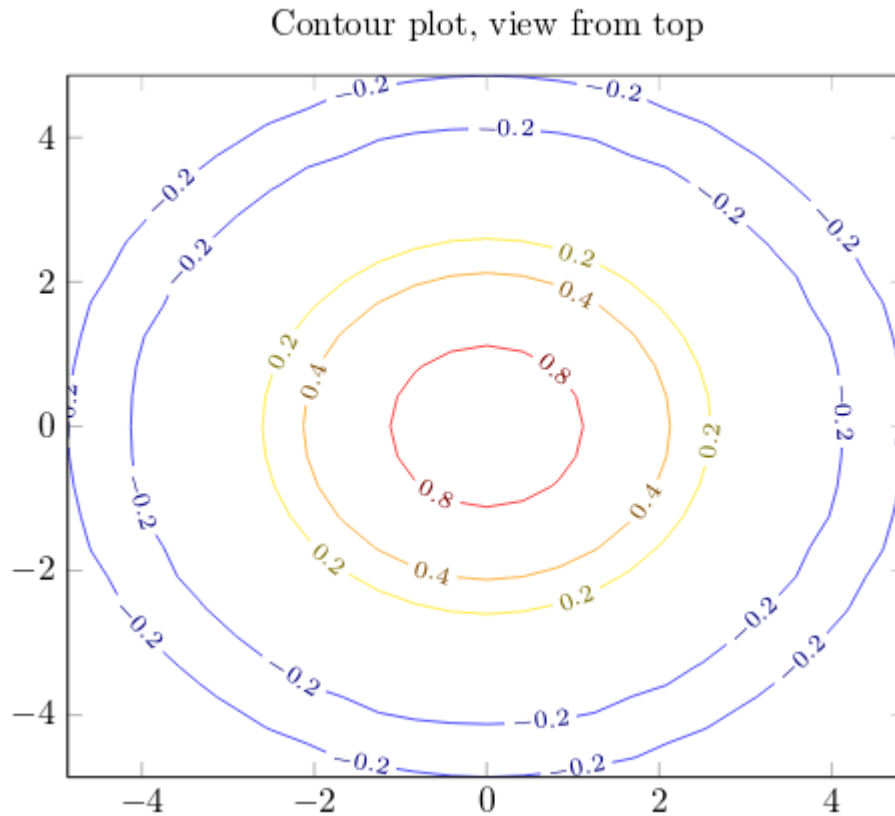
([https://www.sharelatex.com/project/new/template?](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d4bef908c6363f001e4a/download/zip&templateName=Maths%20Expression%203d&compiler=pdflatex)

[zipUrl=/project/52e9d4bef908c6363f001e4a/download/zip&templateName=Maths%20Expression%203d&compiler=pdflatex](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d4bef908c6363f001e4a/download/zip&templateName=Maths%20Expression%203d&compiler=pdflatex))

Contour plots

In **pgfplots** is possible to plot contour plots, but the data has have to be pre calculated by an external program. Let's see:

```
\begin{tikzpicture}
\begin{axis}
[
    title={Contour plot, view from top},
    view={0}{90}
]
\addplot3[
    contour gnuplot={levels={0.8, 0.4, 0.2, -0.2}}
]
{sin(deg(sqrt(x^2+y^2)))/sqrt(x^2+y^2)};
\end{axis}
\end{tikzpicture}
```



(/learn/File:Pgplot3d2.png)

This is a plot of some contour lines for the same equation used in the previous section. 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. There are two new commands:

`view={0}{90}`

This changes the view of the plot. The parameter is passed to the *axis* environment, which means this can be used in any other type of 3d 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}}`

This line of code does two things: First, tells LaTeX to use the external software *gnuplot* to compute the contour lines; this works fine in *ShareLaTeX* 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). Second, the sub parameter `levels` is a list of values of elevation levels where the contour lines are to be computed.

➡ Open an example of the pgfplots package in Overleaf

([https://www.sharelatex.com/project/new/template?](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d52af908c6363f001e4d/download/zip&templateName=Contour%20Plot&compiler=pdflatex)

[zipUrl=/project/52e9d52af908c6363f001e4d/download/zip&templateName=Contour%20Plot&compiler=pdflatex](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d52af908c6363f001e4d/download/zip&templateName=Contour%20Plot&compiler=pdflatex))

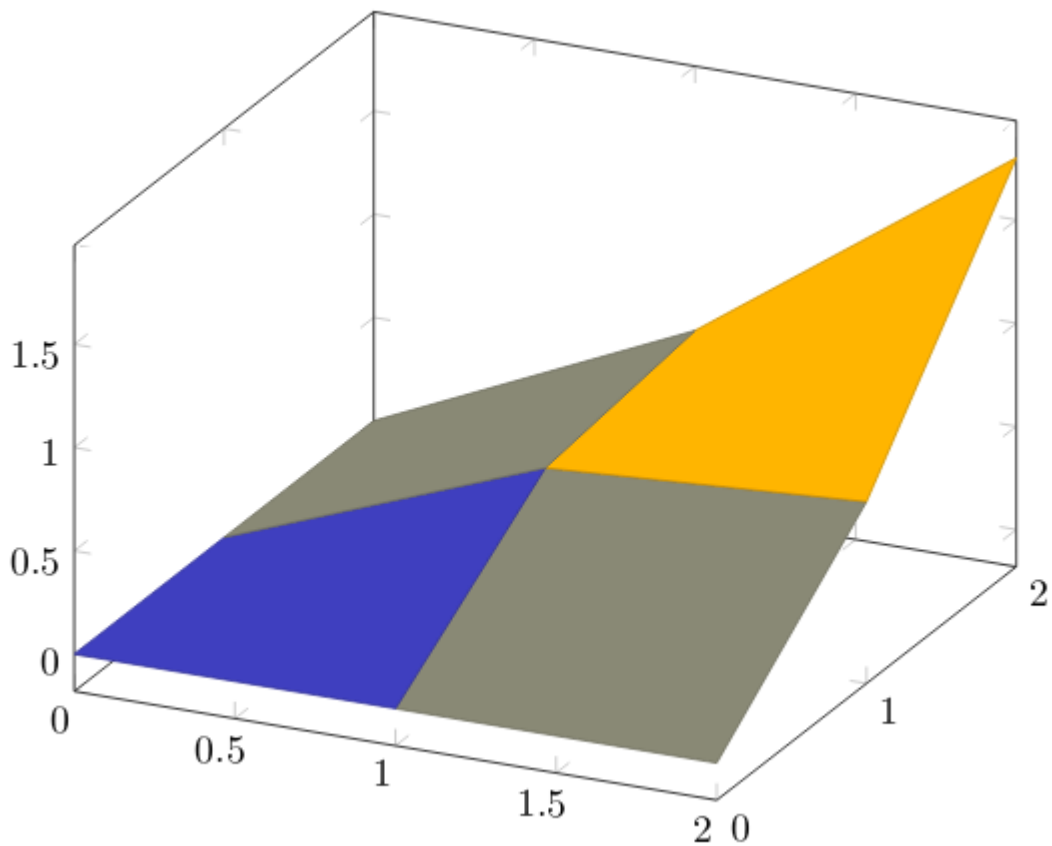
Plotting a surface from data

To plot a set of data into a 3d surface all we need is the coordinates of each point. These coordinates could be an unordered set or, in this case, a matrix:

```
\begin{tikzpicture}
\begin{axis}
\addplot3[
    surf,
]
coordinates {
(0,0,0) (0,1,0) (0,2,0)

(1,0,0) (1,1,0.6) (1,2,0.7)

(2,0,0) (2,1,0.7) (2,2,1.8)
};
\end{axis}
\end{tikzpicture}
```

(/learn/File:Pgplot3d3.png)

The points passed to the `coordinates` parameter are treated as contained in a 3 x 3 matrix, being a white row space the separator of each matrix row.

All the options for 3d plots in this article apply to data surfaces.

➡ Open an example of the pgplots package in Overleaf

([https://www.sharelatex.com/project/new/template?](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d57cf908c6363f001e51/download/zip&templateName=Plot%20Data%203d&compiler=pdflatex)

[zipUrl=/project/52e9d57cf908c6363f001e51/download/zip&templateName=Plot%20Data%203d&compiler=pdflatex](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9d57cf908c6363f001e51/download/zip&templateName=Plot%20Data%203d&compiler=pdflatex))

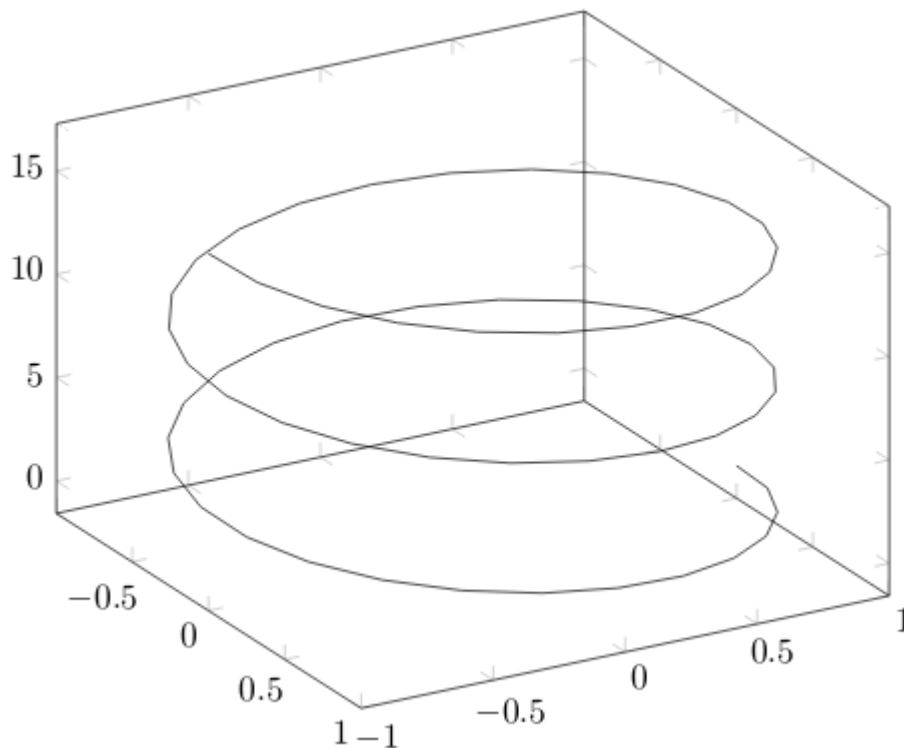
Parametric plot

The syntax for parametric plots is slightly different. Let's see:

```

\begin{tikzpicture}
\begin{axis}
[
view={60}{30},
]
\addplot3[
domain=0:5*pi,
samples = 60,
samples y=0,
]
({sin(deg(x))},
{cos(deg(x))},
{x});
\end{axis}
\end{tikzpicture}

```



(/learn/File:Pgplots3d4.png)

There are only two new things in this example: first, the `samples y=0` to prevent *pgfplots* from joining the extreme points of the spiral and; second, the way the function to plot is passed to the `addplot3` environment. Each parameter function is grouped inside curly brackets and the three parameters are delimited with parenthesis.

➡ Open an example of the *pgfplots* package in Overleaf

([https://www.sharelatex.com/project/new/template?](https://www.sharelatex.com/project/new/template?zipUrl=/project/52e9c7daf908c6363f001e02/download/zip&templateName=Parametric%20)

[zipUrl=/project/52e9c7daf908c6363f001e02/download/zip&templateName=Parametric%20](https://www.sharelatex.com/project/52e9c7daf908c6363f001e02/download/zip&templateName=Parametric%20)

3d%20Plot&compiler=pdflatex)

Reference guide

Option	Description	Possible Values
axis	Normal plots with linear scaling	
semi logx axis	logarithmic scaling of x and normal scaling for y	
semi logy axis	logarithmic scaling for y and normal scaling for x	
loglogaxis	logarithmic scaling for the x and y axes	
axis lines	changes the way the axes are drawn. default is ' <i>box</i> '	box, left, middle, center, right, none
legend pos	position of the legend box	south west, south east, north west, north east, outer north east

mark	type of marks used in data plotting. When a single-character is used, the character appearance is very similar to the actual mark.	*, x, +, , o, asterisk, star, 10-pointed star, oplus, oplus*, otimes, otimes*, square, square*, triangle, triangle*, diamond, halfdiamond*, halvesquare*, right*, left*, Mercedes star, Mercedes star flipped, halfcircle, halfcircle*, pentagon, pentagon*, cubes. (cubes only work on 3d plots).
colormap	colour scheme to be used in a plot, can be personalized but there are some predefined colormaps	hot, hot2, jet, blackwhite, bluered, cool, greenyellow, redyellow, violet.

Further reading

For more information see:

- Using colours in LaTeX (/learn/Using_colours_in_LaTeX)
- TikZ package (/learn/TikZ_package)
- Externalising pgfplots and tikzpictures (/learn/Questions/I_have_a_lot_of_tikz_matlab2tikz_or_pgfplots_figures_so_im_getting_a_compilation_timeout_Can_I_externalise_my_figures%3F)
- Inserting Images (/learn/Inserting_Images)
- Lists of tables and figures (/learn/Lists_of_tables_and_figures)
- Positioning images and tables (/learn/Positioning_images_and_tables)
- Drawing diagrams directly in LaTeX (/learn/Picture_environment)
- The **pgfplots** package documentation (<http://mirrors.ctan.org/graphics/pgf/contrib/pgfplots/doc/pgfplots.pdf>).
- The TikZ and PGF Packages: Manual for version 3.0.0 (<http://mirror.utexas.edu/ctan/graphics/pgf/base/doc/pgfmanual.pdf>)
- TikZ and PGF examples at TeXample.net (<http://www.texample.net/tikz/examples/all/>)

Documentation Home (/learn/Main_Page)

Learn LaTeX in 30 minutes (/learn/Learn_LaTeX_in_30_minutes)

Overleaf guides

[Creating a document in Overleaf \(/learn/Kb/Creating_a_document_in_Overleaf\)](/learn/Kb/Creating_a_document_in_Overleaf)

[Uploading a project \(/learn/Kb/Uploading_a_project\)](/learn/Kb/Uploading_a_project)

[Copying a project \(/learn/Kb/Copying_a_project\)](/learn/Kb/Copying_a_project)

[Creating a project from a template \(/learn/Kb/Creating_a_project_from_a_template\)](/learn/Kb/Creating_a_project_from_a_template)

[Including images in Overleaf \(/learn/Kb/Including_images_in_ShareLaTeX\)](/learn/Kb/Including_images_in_ShareLaTeX)

[Exporting your work from Overleaf \(/learn/Kb/Exporting_your_work_from_ShareLaTeX\)](/learn/Kb/Exporting_your_work_from_ShareLaTeX)

[Working offline in Overleaf \(/learn/Kb/Working_Offline_in_Overleaf_v2\)](/learn/Kb/Working_Offline_in_Overleaf_v2)

[Using Track Changes in Overleaf \(/learn/Kb/Track_Changes_in_Overleaf_v2\)](/learn/Kb/Track_Changes_in_Overleaf_v2)

[Using bibliographies in Overleaf \(/learn/Kb/Using_bibliographies_in_ShareLaTeX\)](/learn/Kb/Using_bibliographies_in_ShareLaTeX)

[Sharing your work with others \(/learn/Kb/Sharing_your_work_with_others\)](/learn/Kb/Sharing_your_work_with_others)

[Using the History feature \(/learn/Using_the_History_feature\)](/learn/Using_the_History_feature)

[Debugging Compilation timeout errors \(/learn/Kb/Debugging_Compilation_timeout_errors\)](/learn/Kb/Debugging_Compilation_timeout_errors)

[How-to guides \(/learn/Kb/Knowledge_Base\)](/learn/Kb/Knowledge_Base)

LaTeX Basics

[Creating your first LaTeX document \(/learn/Creating_a_document_in_LaTeX\)](/learn/Creating_a_document_in_LaTeX)

[Choosing a LaTeX Compiler \(/learn/Choosing_a_LaTeX_Compiler\)](/learn/Choosing_a_LaTeX_Compiler)

[Paragraphs and new lines \(/learn/Paragraphs_and_new_lines\)](/learn/Paragraphs_and_new_lines)

[Bold, italics and underlining \(/learn/Bold,_italics_and_underlining\)](/learn/Bold,_italics_and_underlining)

[Lists \(/learn/Lists\)](/learn/Lists)

[Errors \(/learn/Errors\)](/learn/Errors)

Mathematics

[Mathematical expressions \(/learn/Mathematical_expressions\)](/learn/Mathematical_expressions)

[Subscripts and superscripts \(/learn/Subscripts_and_superscripts\)](/learn/Subscripts_and_superscripts)

[Brackets and Parentheses \(/learn/Brackets_and_Parentheses\)](/learn/Brackets_and_Parentheses)

[Matrices \(/learn/Matrices\)](/learn/Matrices)

[Fractions and Binomials \(/learn/Fractions_and_Binomials\)](/learn/Fractions_and_Binomials)

[Aligning Equations \(/learn/Aligning_equations\)](/learn/Aligning_equations)

[Operators \(/learn/Operators\)](/learn/Operators)

[Spacing in math mode \(/learn/Spacing_in_math_mode\)](/learn/Spacing_in_math_mode)

[Integrals, sums and limits \(/learn/Integrals,_sums_and_limits\)](/learn/Integrals,_sums_and_limits)

[Display style in math mode \(/learn/Display_style_in_math_mode\)](/learn/Display_style_in_math_mode)

[List of Greek letters and math symbols \(/learn/List_of_Greek_letters_and_math_symbols\)](/learn/List_of_Greek_letters_and_math_symbols)

[Mathematical fonts \(/learn/Mathematical_fonts\)](/learn/Mathematical_fonts)

Figures and tables

[Inserting Images \(/learn/Inserting_Images\)](#)

[Tables \(/learn/Tables\)](#)

[Positioning Images and Tables \(/learn/Positioning_images_and_tables\)](#)

[Lists of Tables and Figures \(/learn/Lists_of_tables_and_figures\)](#)

[Drawing Diagrams Directly in LaTeX \(/learn/Picture_environment\)](#)

[TikZ package \(/learn/TikZ_package\)](#)

References and Citations

[Bibliography management in LaTeX \(/learn/Bibliography_management_in_LaTeX\)](#)

[Bibliography management with biblatex \(/learn/Bibliography_management_in_LaTeX\)](#)

[Biblatex bibliography styles \(/learn/Biblatex_bibliography_styles\)](#)

[Biblatex citation styles \(/learn/Biblatex_citation_styles\)](#)

[Bibliography management with natbib \(/learn/Bibliography_management_with_natbib\)](#)

[Natbib bibliography styles \(/learn/Natbib_bibliography_styles\)](#)

[Natbib citation styles \(/learn/Natbib_citation_styles\)](#)

[Bibliography management with bibtex \(/learn/Bibliography_management_with_bibtex\)](#)

[Bibtex bibliography styles \(/learn/Bibtex_bibliography_styles\)](#)

Languages

[Multilingual typesetting on Overleaf using polyglossia and fontspec \(/learn/Multilingual_typesetting_on_Overleaf_using_polyglossia_and_fontspec\)](#)

[Multilingual typesetting on Overleaf using babel and fontspec \(/learn/Multilingual_typesetting_on_Overleaf_using_babel_and_fontspec\)](#)

[International language support \(/learn/International_language_support\)](#)

[Quotations and quotation marks \(/learn/Typesetting_quotations\)](#)

[Arabic \(/learn/Arabic\)](#)

[Chinese \(/learn/Chinese\)](#)

[French \(/learn/French\)](#)

[German \(/learn/German\)](#)

[Greek \(/learn/Greek\)](#)

[Italian \(/learn/Italian\)](#)

[Japanese \(/learn/Japanese\)](#)

[Korean \(/learn/Korean\)](#)

[Portuguese \(/learn/Portuguese\)](#)

[Russian \(/learn/Russian\)](#)

[Spanish \(/learn/Spanish\)](#)

Document structure

[Sections and chapters \(/learn/Sections_and_chapters\)](#)

[Table of contents \(/learn/Table_of_contents\)](#)

[Cross referencing sections and equations \(/learn/Cross_referencing_sections_and_equations\)](#)

[Indices \(/learn/Indices\)](#)

[Glossaries \(/learn/Glossaries\)](#)

[Nomenclatures \(/learn/Nomenclatures\)](#)

[Management in a large project \(/learn/Management_in_a_large_project\)](#)

[Multi-file LaTeX projects \(/learn/Multi-file_LaTeX_projects\)](#)

[Hyperlinks \(/learn/Hyperlinks\)](#)

Formatting

[Lengths in L^AT_EX \(/learn/Lengths_in_LaTeX\)](#)

[Headers and footers \(/learn/Headers_and_footers\)](#)

[Page numbering \(/learn/Page_numbering\)](#)

[Paragraph formatting \(/learn/Paragraph_formatting\)](#)

[Line breaks and blank spaces \(/learn/Line_breaks_and_blank_spaces\)](#)

[Text alignment \(/learn/Text_alignment\)](#)

[Page size and margins \(/learn/Page_size_and_margins\)](#)

[Single sided and double sided documents \(/learn/Single_sided_and_double_sided_documents\)](#)

[Multiple columns \(/learn/Multiple_columns\)](#)

[Counters \(/learn/Counters\)](#)

[Code listing \(/learn/Code_listing\)](#)

[Code Highlighting with minted \(/learn/Code_Highlighting_with_minted\)](#)

[Using colours in LaTeX \(/learn/Using_colours_in_LaTeX\)](#)

[Footnotes \(/learn/Footnotes\)](#)

[Margin notes \(/learn/Margin_notes\)](#)

Fonts

[Font sizes, families, and styles \(/learn/Font_sizes,_families,_and_styles\)](#)

[Font typefaces \(/learn/Font_typefaces\)](#)

[Supporting modern fonts with X_YL^AT_EX \(/learn/XeLaTeX\)](#)

Presentations

[Beamer \(/learn/Beamer\)](#)

[Powerdot \(/learn/Powerdot\)](#)

[Posters \(/learn/Posters\)](#)

Commands

[Commands \(/learn/Commands\)](#)

[Environments \(/learn/Environments\)](#)

Field specific

[Theorems and proofs \(/learn/Theorems_and_proofs\)](#)

[Chemistry formulae \(/learn/Chemistry_formulae\)](#)

[Feynman diagrams \(/learn/Feynman_diagrams\)](#)

[Molecular orbital diagrams \(/learn/Molecular_orbital_diagrams\)](#)

[Chess notation \(/learn/Chess_notation\)](#)

[Knitting patterns \(/learn/Knitting_patterns\)](#)

[Circuitikz package \(/learn/Circuitikz_package\)](#)

[Pgfpplots package \(/learn/Pgfpplots_package\)](#)

[Typing exams in LaTeX \(/learn/Typing_exams_in_LaTeX\)](#)

[Knitr \(/learn/Knitr\)](#)

[Attribute Value Matrices \(/learn/Attribute_Value_Matrices\)](#)

Class files

[Understanding packages and class files \(/learn/Understanding_packages_and_class_files\)](#)

[List of packages and class files \(/learn/List_of_packages_and_class_files\)](#)

[Writing your own package \(/learn/Writing_your_own_package\)](#)

[Writing your own class \(/learn/Writing_your_own_class\)](#)

[Tips \(/learn/Tips\)](#)

Advanced TeX/LaTeX

[In-depth technical articles on TeX/LaTeX \(/learn/Articles\)](#)