TikZ/pgfplots tutorial

Lucas Unnerfelt

March 2020

Basic plotting

Contour plots

Misc

Relevant packages

1. tikz

Relevant packages

1. tikz
2. pgfplots

Relevant packages

1. tikz
2. pgfplots
3. pgfplotstable

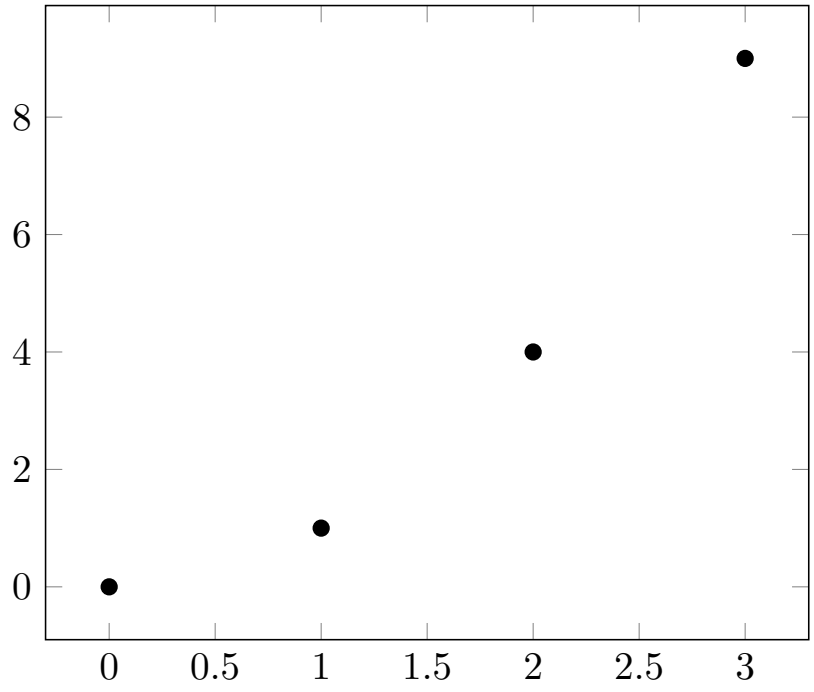
Relevant packages

1. tikz
2. pgfplots
3. pgfplotstable

All pacakges have extensive documentation on ctan.org Pgfplots and pgfplotstable have reasonably large documentations. TikZ has more than 1000 pages. Lots of code and examples.

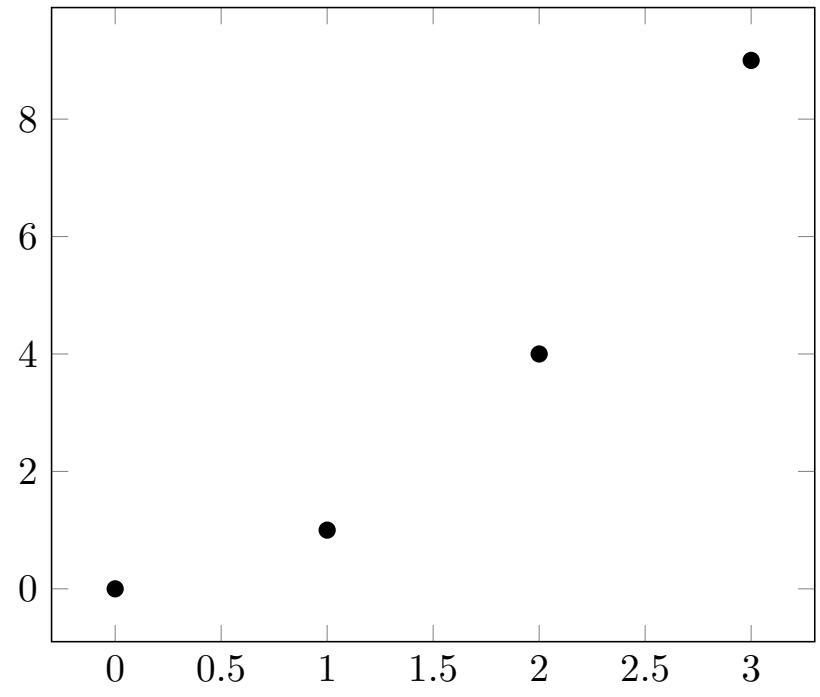
Including packages, configuration.

```
\documentclass{standalone}

\usepackage{pgfplots}
\pgfplotsset{compat=1.16}

\begin{document}
\begin{tikzpicture}
  \begin{axis}
    \addplot [black, only marks]
    table [x=x, y=f(x)] {
      x f(x)
      0 0
      1 1
      2 4
      3 9
    };
  \end{axis}
\end{tikzpicture}
\end{document}
```



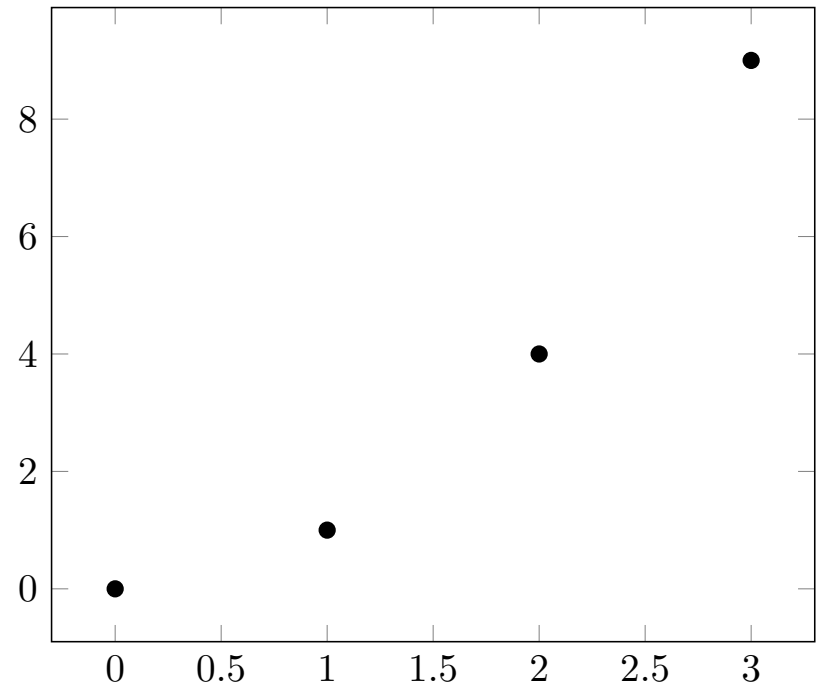
Including packages, configuration.

```
\documentclass{standalone}

\usepackage{pgfplots}
\pgfplotsset{compat=1.16}

\begin{document}
\begin{tikzpicture}
  \begin{axis}
    \addplot [black, only marks]
    table [x=x, y=f(x)] {
      x f(x)
      0 0
      1 1
      2 4
      3 9
    };
  \end{axis}
\end{tikzpicture}
\end{document}
```



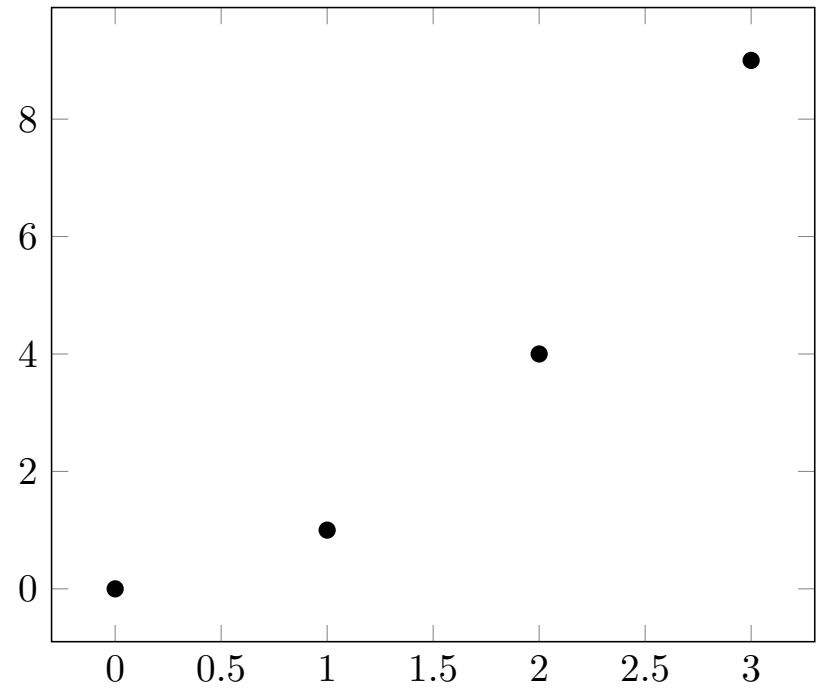
Including packages, configuration.

```
\documentclass{standalone}

\usepackage{pgfplots}
\pgfplotsset{compat=1.16}

\begin{document}
\begin{tikzpicture}
  \begin{axis}
    \addplot [black, only marks]
    table [x=x, y=f(x)] {
      x f(x)
      0 0
      1 1
      2 4
      3 9
    };
  \end{axis}
\end{tikzpicture}
\end{document}
```



Including packages, configuration.

```
\documentclass{standalone}

\usepackage{pgfplots}
\pgfplotsset{compat=1.16}

\begin{document}
\begin{tikzpicture}
  \begin{axis}
    \addplot [black, only marks]
    table [x=x, y=f(x)] {
      x f(x)
      0 0
      1 1
      2 4
      3 9
    };
  \end{axis}
\end{tikzpicture}
\end{document}
```



Example of data file.

```
x       f(x)    g(x)    h(x)
0       0       0       0
1       1       1       1
2       4       8       16
3       9       27      81
4       16      64      256
5       25      125     625
6       36      216     1296
```

Saved as data.txt. It is possible to reconfigure pgfplots(table) to load a lot of different text formats. But I have found it most simple to reformat in MATLAB/Python instead of in LaTeXand TikZ.

Save in correct format using Python/numpy

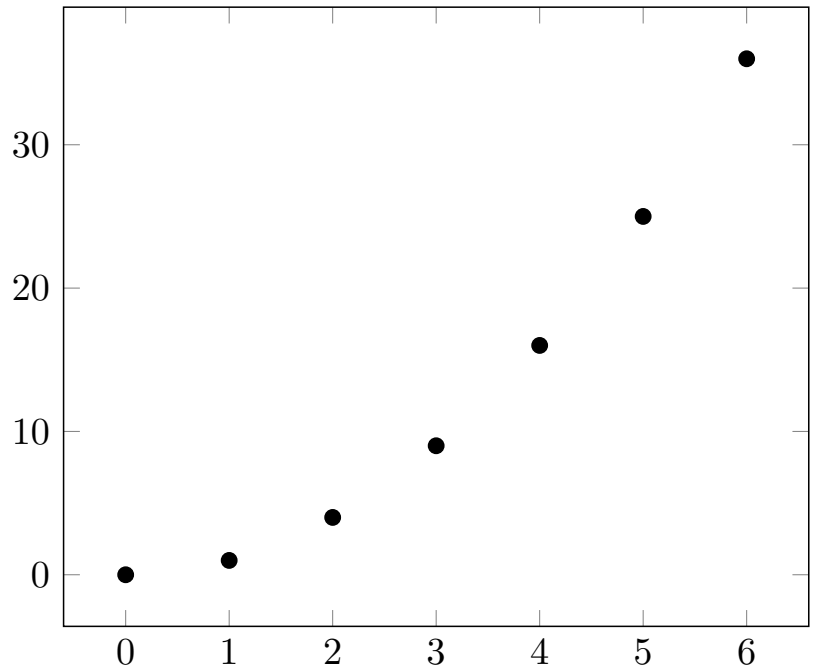
```
import numpy as np

x = np.arange(7)

a = np.stack([x, x ** 2, x ** 3, x ** 4]).transpose()

np.savetxt('data-py.txt', a, header='x f(x) g(x) h(x)', comments='')
```

MATLAB

```
x = 0:6;

a = [x; x .^ 2; x .^ 3; x .^ 4];

fid = fopen('data-mat.txt', 'w');

fprintf(fid, 'x f(x) g(x) h(x)\n');
fprintf(fid, '%f %f %f %f\n', a);

fclose(fid);
```

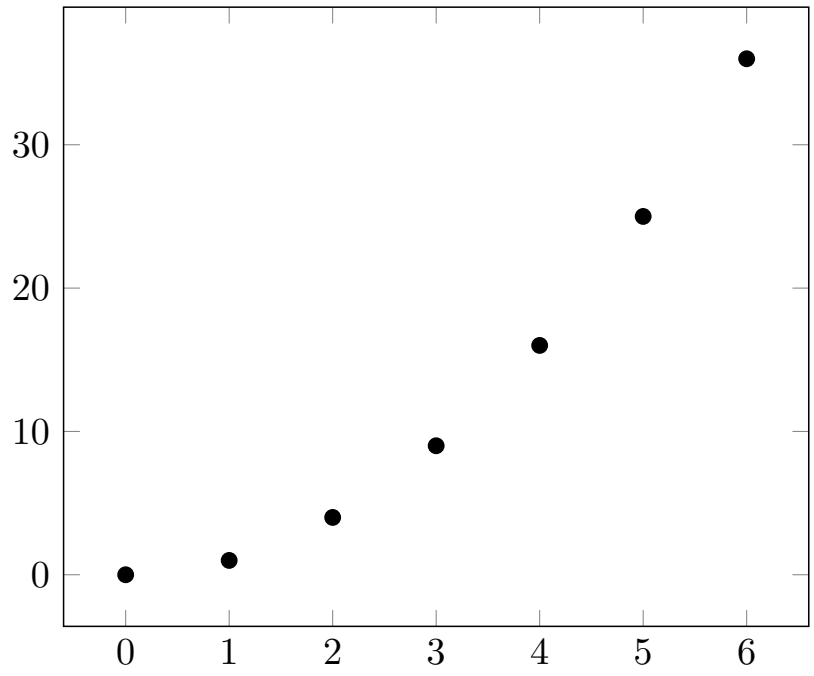
Loading data file

```
\pgfplotstableread{data.txt}\data

\begin{tikzpicture}
  \begin{axis}
    \addplot [black, only marks]
    table [x=x, y=f(x)] {\data};
  \end{axis}
\end{tikzpicture}
```



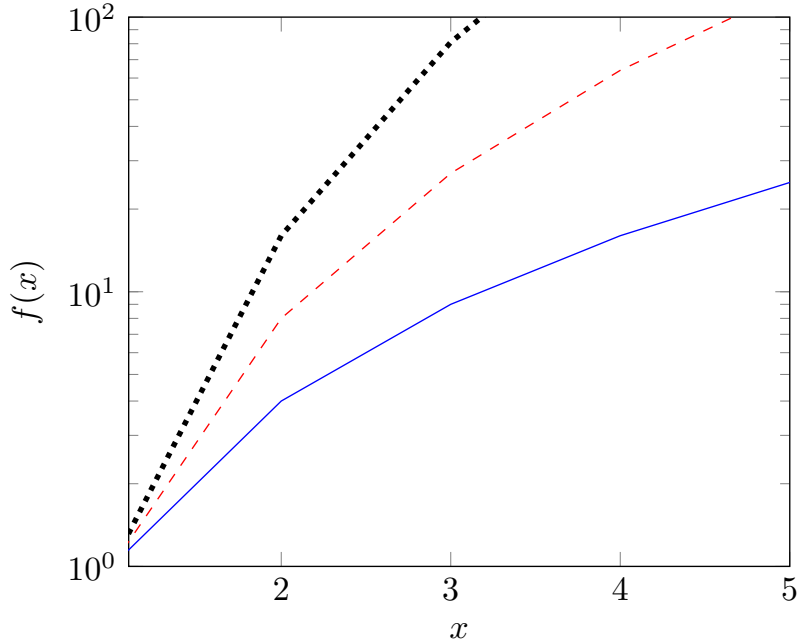
Loading data file

```
\pgfplotstableread{data.txt}\data

\begin{tikzpicture}
  \begin{axis}
    \addplot [black, only marks]
    table [x=x, y=f(x)] {\data};
  \end{axis}
\end{tikzpicture}
```



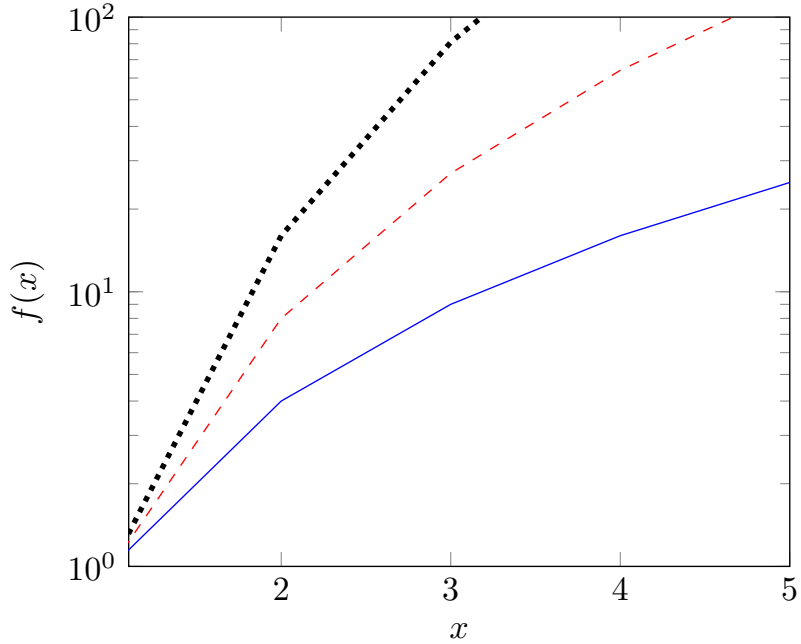
Axis settings

```
\pgfplotstableread{data.txt}\data

\begin{tikzpicture}
  \begin{axis}
    [xlabel={$x$}, ylabel={$f(x)$},
      ymode=log,
      xmin=1.1, xmax=5,
      ymin=1, ymax=100]

    \addplot [blue]
    table [x=x, y=f(x)] {\data};

    \addplot [red, dashed]
    table [x=x, y=g(x)] {\data};

    \addplot [black, dotted,
      ultra thick]
    table [x=x, y=h(x)] {\data};

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



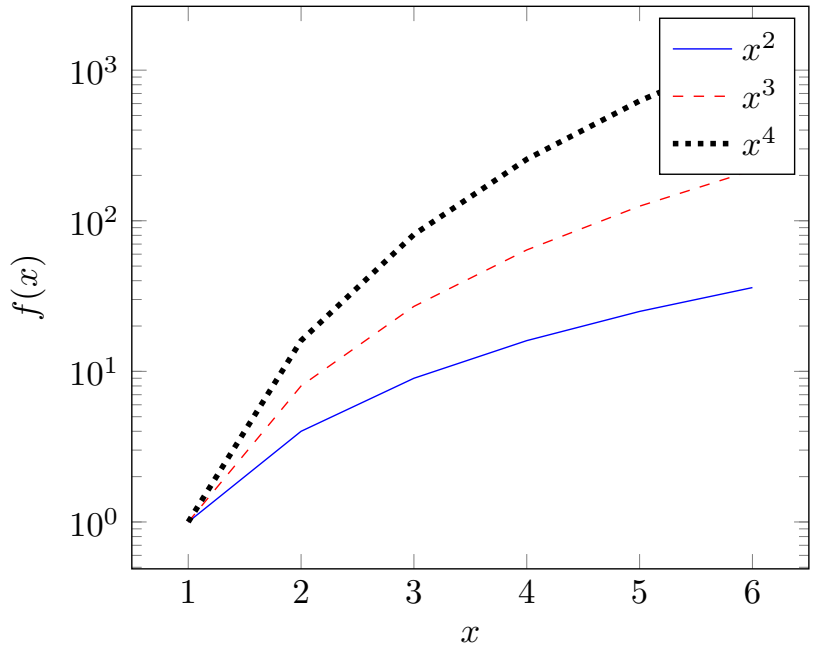
Axis settings

```
\pgfplotstableread{data.txt}\data

\begin{tikzpicture}
  \begin{axis}
    [xlabel={$x$}, ylabel={$f(x)$},
      ymode=log,
      xmin=1.1, xmax=5,
      ymin=1, ymax=100]

    \addplot [blue]
    table [x=x, y=f(x)] {\data};

    \addplot [red, dashed]
    table [x=x, y=g(x)] {\data};

    \addplot [black, dotted,
      ultra thick]
    table [x=x, y=h(x)] {\data};

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



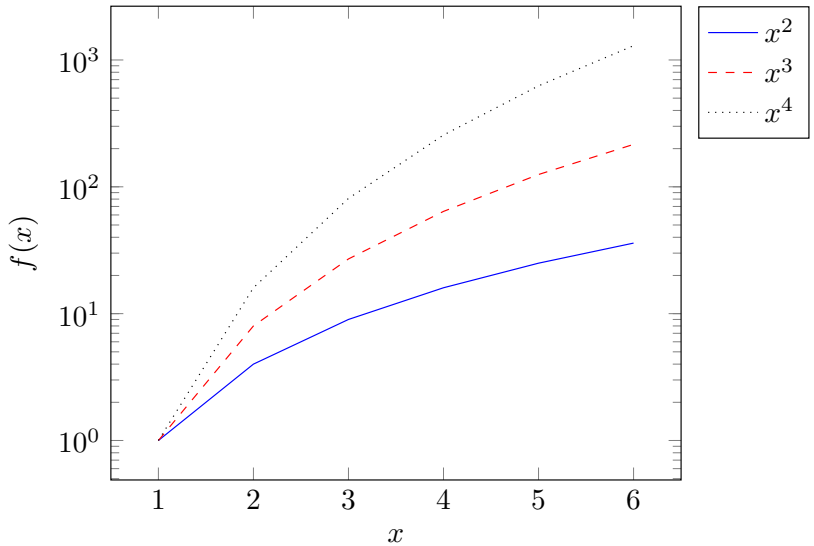
Legends

```
\pgfplotstableread {data.txt}\data

\begin{tikzpicture}
  \begin{axis}
    [xlabel={$x$}, ylabel={$f(x)$},
      ymode=log,
      legend entries={
        $x^2$,$x^3$,$x^4$},
    ]

    \addplot [blue]
    table [x=x, y=f(x)] {\data};

    \addplot [red, dashed]
    table [x=x, y=g(x)] {\data};

    \addplot [black, dotted,
      ultra thick]
    table [x=x, y=h(x)] {\data};

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



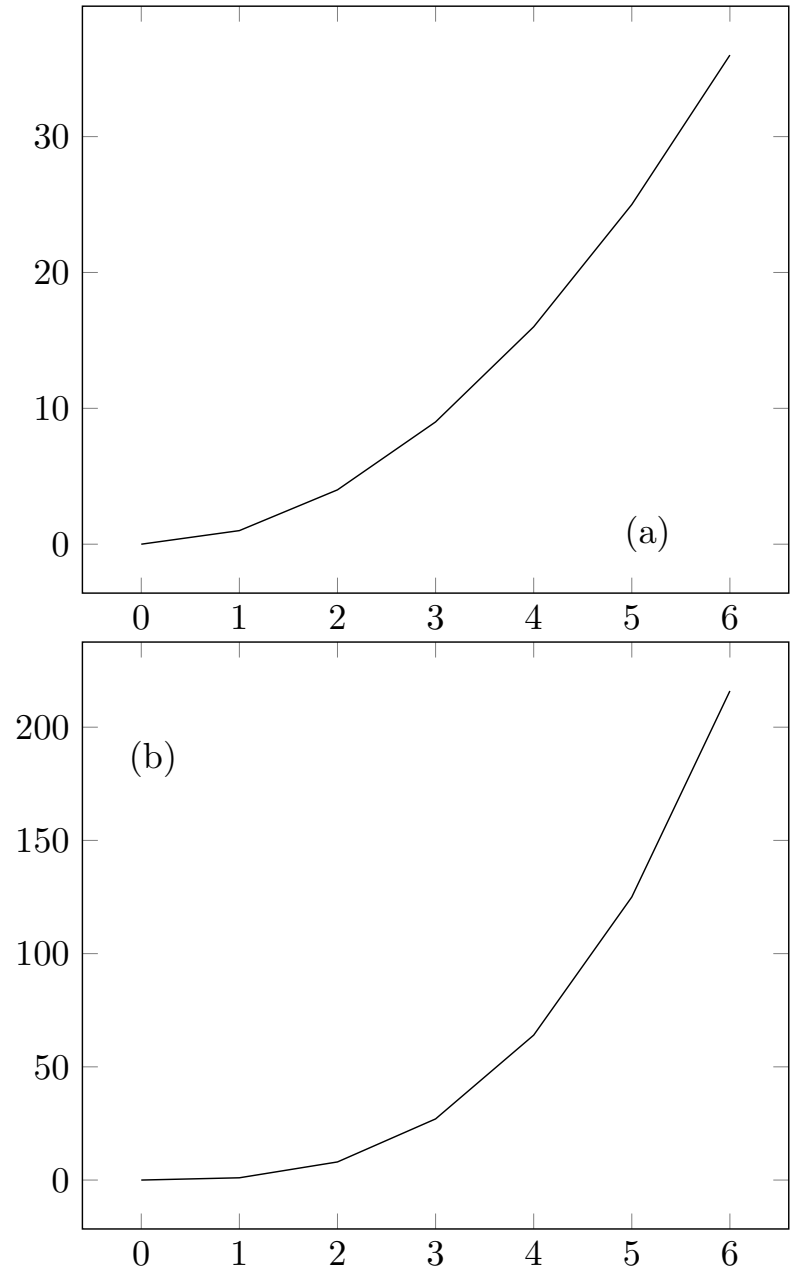
Legends

```
\pgfplotstableread{data.txt}\data

\begin{tikzpicture}
  \begin{axis}
    [xlabel={$x$}, ylabel={$f(x)$},
      ymode=log,
      legend entries={
        $x^2$,$x^3$,$x^4$},
      legend pos=outer north east,
    ]

    \addplot [blue]
    table [x=x, y=f(x)] {\data};

    \addplot [red, dashed]
    table [x=x, y=g(x)] {\data};

    \addplot [black, dotted]
    table [x=x, y=h(x)] {\data};

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



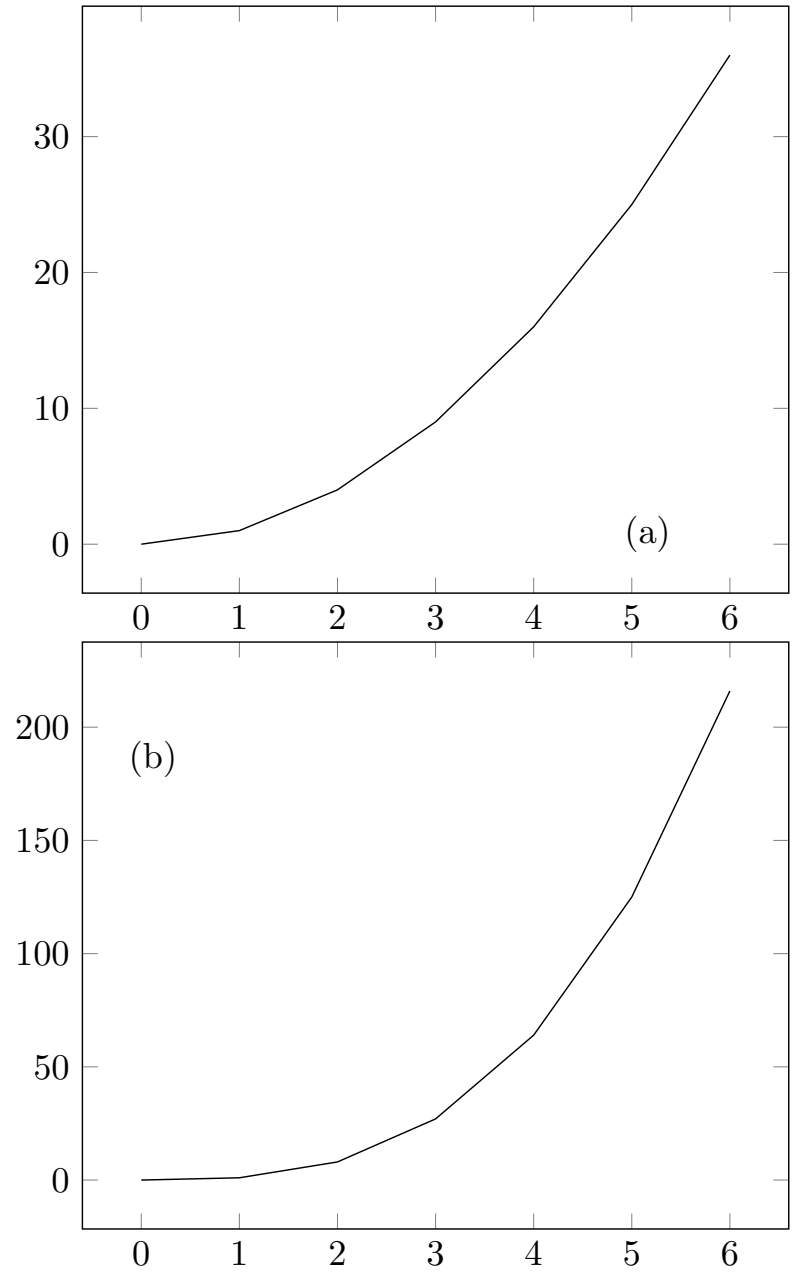
Multiple plots with labels.

```
\pgfplotstableread{data.txt}\data

\begin{tikzpicture}
  \matrix{
  \begin{axis}
    \node at (rel axis cs:0.8, 0.1)
          {(a)};

    \addplot [black]
    table [x=x, y=f(x)] {\data};
  \end{axis}\\
  \begin{axis}
    \node at (rel axis cs:0.1, 0.8)
          {(b)};

    \addplot [black]
    table [x=x, y=g(x)] {\data};
  \end{axis}\\
  };
\end{tikzpicture}
```



Multiple plots with labels.

```
\pgfplotstableread{data.txt}\data

\begin{tikzpicture}
  \matrix{
  \begin{axis}
    \node at (rel axis cs:0.8, 0.1)
          {(a)};

    \addplot [black]
    table [x=x, y=f(x)] {\data};
  \end{axis}\\
  \begin{axis}
    \node at (rel axis cs:0.1, 0.8)
          {(b)};

    \addplot [black]
    table [x=x, y=g(x)] {\data};
  \end{axis}\\
  };
\end{tikzpicture}
```



Contour plots

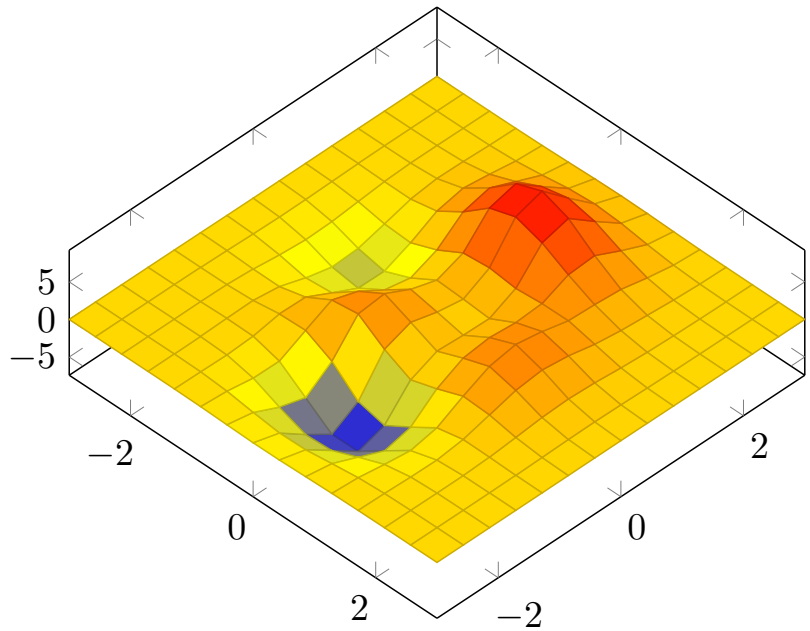
- Should be simple (and is for non-filled contour plots)
- Built in pgfplots functionality for filled contour plots only works in Acrobat Reader
- Therefore we need workarounds

Data we will be working with

```
[X, Y, Z] = peaks(15);

fid = fopen('datac.txt', 'w');

fprintf(fid, 'x y z\n');
fprintf(fid, '%f %f %f\n', [X(:) Y(:) Z(:)]');

fclose(fid);
```

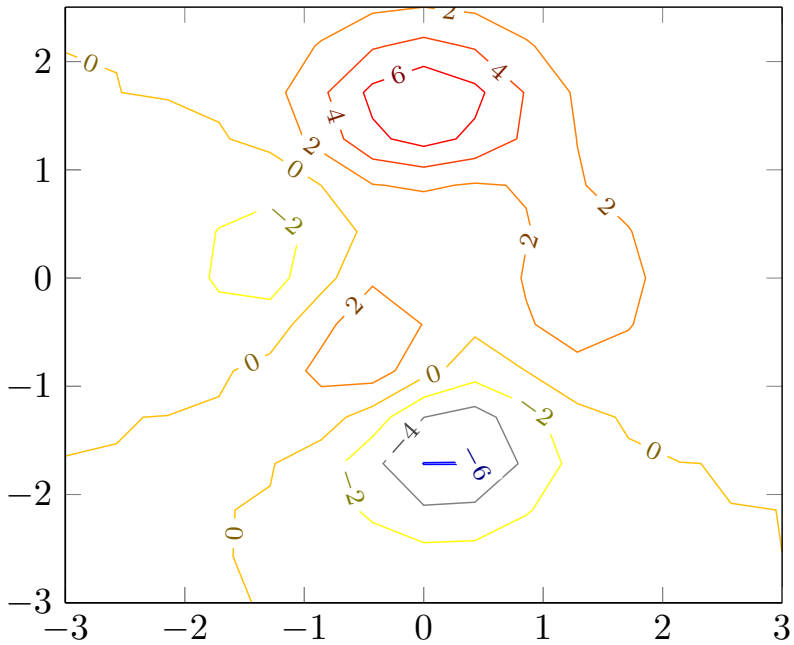
3D plot.

```
\pgfplotstableread{datac.txt}\data

\begin{tikzpicture}
  \begin{axis}[view={45}{70}]
    \addplot3 [
      surf,
      z buffer=sort]
    table {\data};
  \end{axis}
\end{tikzpicture}
```



Contour plot.

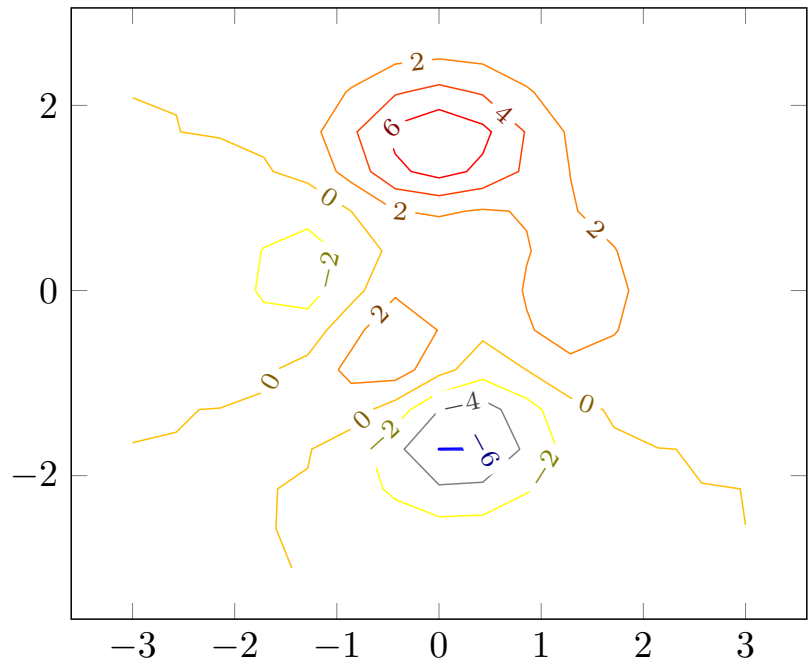
```
\pgfplotstableread{datac.txt}\data

\begin{tikzpicture}
  \begin{axis}[
      view={0}{90}]
    \addplot3 [
      contour gnuplot={
        number=10
      },
      mesh/rows=15, mesh/cols=15]
    table {\data};
  \end{axis}
\end{tikzpicture}
```



Using MATLAB

```
[X, Y, Z] = peaks(15);

M = contour(X, Y, Z); % Using contourc is also possible.
M = M'

save 'exportcontour.txt' M -ASCII
```

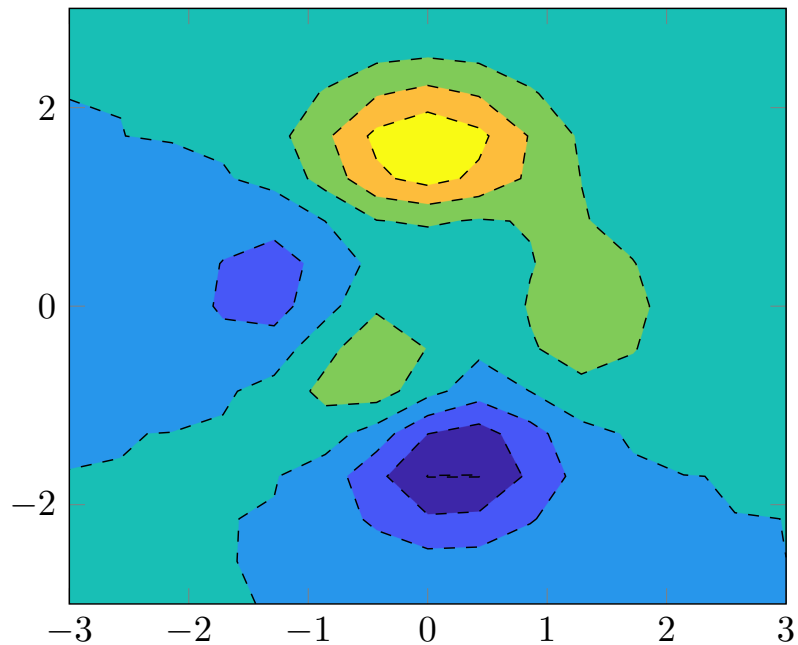
```
\begin{tikzpicture}
  \begin{axis}
    \addplot [
      contour prepared,
      contour prepared format=matlab
    ]
    table {exportcontour.txt};
  \end{axis}
\end{tikzpicture}
```



Using matlab, filled

```
[X, Y, Z] = peaks(15);

M = contourf(X, Y, Z, 'lines', 'none');
M = M'

axis off
saveas(gcf, 'exportedfig.png');
system('convert -trim exportedfig.png exportedfig_t.png');

save 'exportcontour.txt' M -ASCII
```

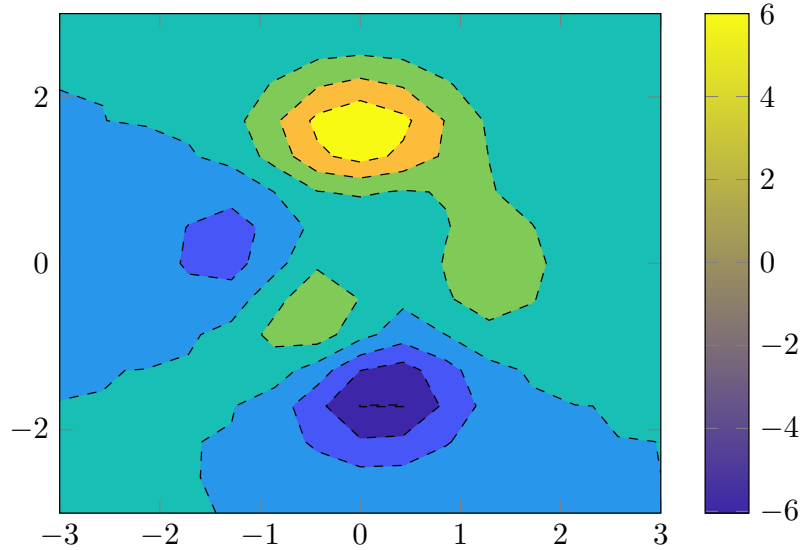
```
\begin{tikzpicture}
  \begin{axis} [
      enlargelimits=false,
      axis on top=true]
    \addplot graphics
              [xmin=-3,xmax=3,
                ymin=-3,ymax=3]
              {exportedfig_t.png};
    \addplot [
        contour prepared = {
          draw color = black,
          labels = false
        },
      contour prepared format=matlab,
      dashed
    ]
    table {exportcontour.txt};
  \end{axis}
\end{tikzpicture}
```



Colorbar

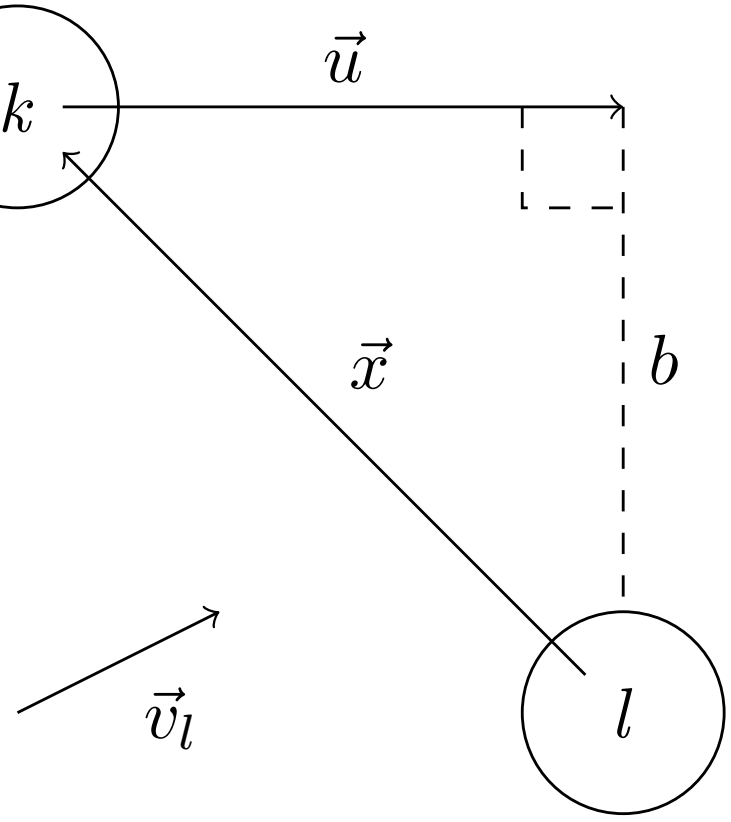
```
cb = colorbar;

fprintf('%f %f\n', cb.Limits);
fprintf('rgb=(%f %f %f)\n', colormap');
```

```
\begin{tikzpicture}
  \begin{axis}[
      enlargelimits=false,
      axis on top=true,
      colormap={mymap}{
rgb=(0.242200,0.150400,0.660300)
% ...
rgb=(0.976900,0.983900,0.080500)},
      colorbar,
      point meta min=-6.0419,
      point meta max=6
    ]
    \addplot graphics
             [xmin=-3,xmax=3,
               ymin=-3,ymax=3]
             {exportedfig_t.png};
    \addplot [
        contour prepared = {
          draw color = black,
          labels = false
        },
      contour prepared format=matlab,
      dashed
    ]
    table {exportcontour.txt};
  \end{axis}
\end{tikzpicture}
```



Illustrations (inkscape might be a better choice for this)

```
\begin{tikzpicture}
  \node(k) at (0,3) {\textit{k}};
  \node(l) at (3,0) {\textit{l}};
  \draw(k) circle [radius=0.5];
  \draw(l) circle [radius= 0.5];
  \draw[->](k) --(3,3) node
       [midway,above] {$\vec{u}$};
  \draw[->](l) -- (k) node
       [midway, above right]
       {$\vec{x}$};
  \draw[dashed] (3,3)--(3,0.5) node
       [midway, right] {$b$};
  \draw[dashed] (2.5,3)--(2.5,2.5)
       --(3,2.5);
  \draw[->] (0,0) -- (1,0.5) node
       [midway, below right]
       {$\vec{v}_l$};
\end{tikzpicture}
```



Speed up compilation

```
\usetikzlibrary{external}
\tikzexternalize
```

Documentation for pgfplots, tikz, and even pgfplotstable are really good.