

# LaTeX Posters and Vectorized Graphics

Bryan Herman

Massachusetts Institute of Technology - Nuclear Science and Engineering

## Introduction

The `baposter` class provides a great way to make a LaTeX poster. The documentation for this class is online. It should be noted that I have fixed some problems with the background function. The `tikz` (0,0) origin does not show up at the bottom left corner sometimes. I have added an extra inch when defining the background rectangle coordinates. Below are some necessary steps when constructing a poster.

1. Choose a poster size and either portrait or landscape
2. Customize the Background Color
3. Add titleheader information, this takes some customization
4. You can add a logo if you wish on the right
5. Use eyecatcher if you want a logo on the left
6. Decide the number of rows and columns. It should be noted that row heights will automatically adjust with content. Column widths will not expand. You can span multiple columns but it will be in an integer amount.
7. Layout you content boxes where you can use a combination of absolute and relative positioning which we will do as an example

## Vectorized Images

It is important that you take the time to produce high quality images for your work. No matter if we are doing simple documentation, theses, publications, presentations or even posters, we can use the same procedure to develop plots and images. Below is a list of programs I use for certain applications:

- Images
  - Tikz - amazing capabilities
  - Inkscape - general drawings
  - Geogebra - geometry drawings
- Plottings
  - Gnuplot - very easy from terminal
  - Matplotlib - integrated into python
  - Matlab - not a huge fan of this but it works well in certain situations

Some common extensions that we use:

**PDF** Portable Document Format

**SVG** Scalable Vector Graphics

**EPS** Encapsulated Post Script

EPS and SVG must be converted to PDF format before including them in latex.

## Gnuplot

I use gnuplot a lot for line plots and surface plots to generate pdfs. I then include these pdfs as images just like we learned before. Let's go ahead and plot a sine curve. With any new thing, just have to read online and learn as you go.

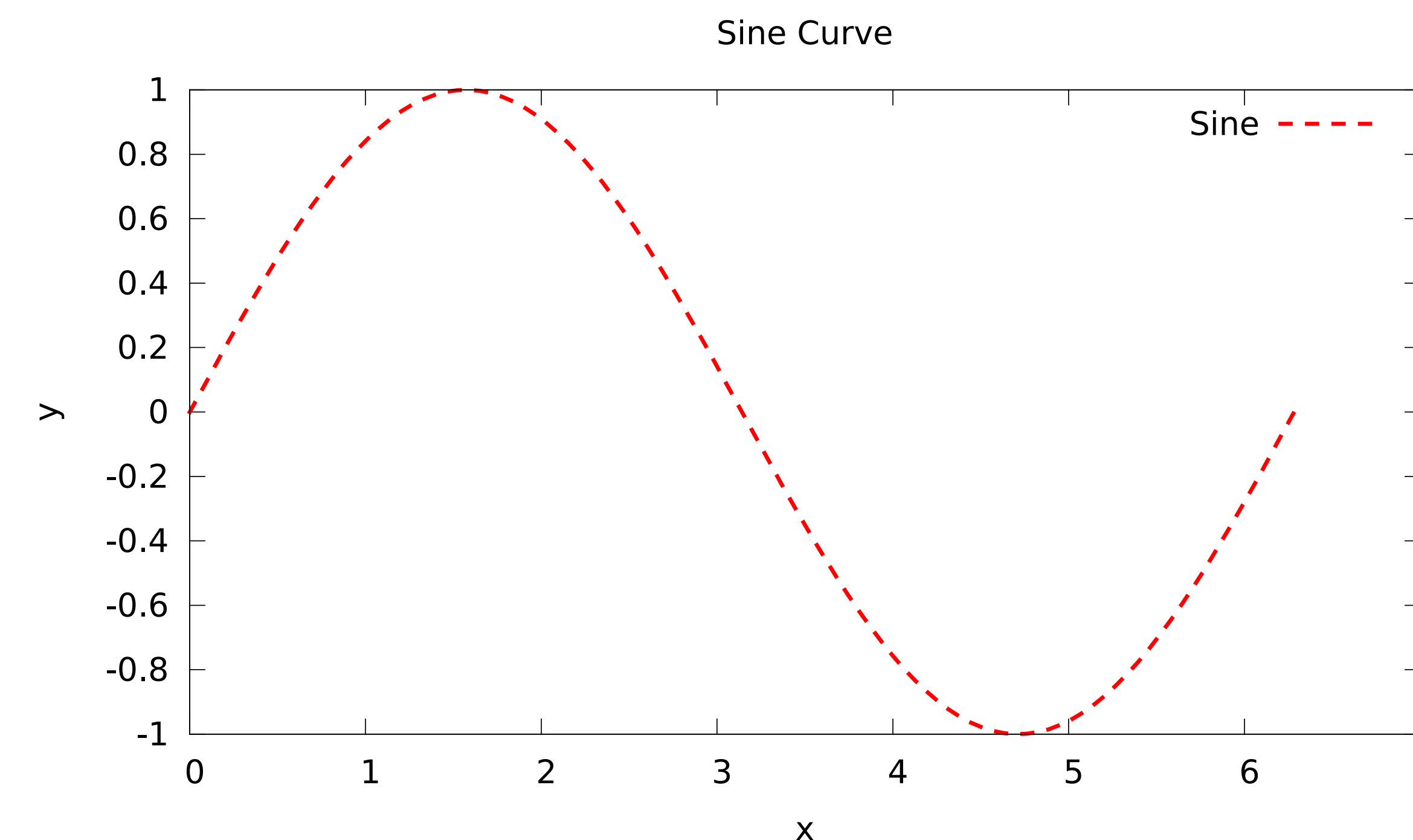
Here is an example input file for this:

```
#!/bin/sh env gnuplot

set terminal pdf enhanced
set output 'sine.pdf'

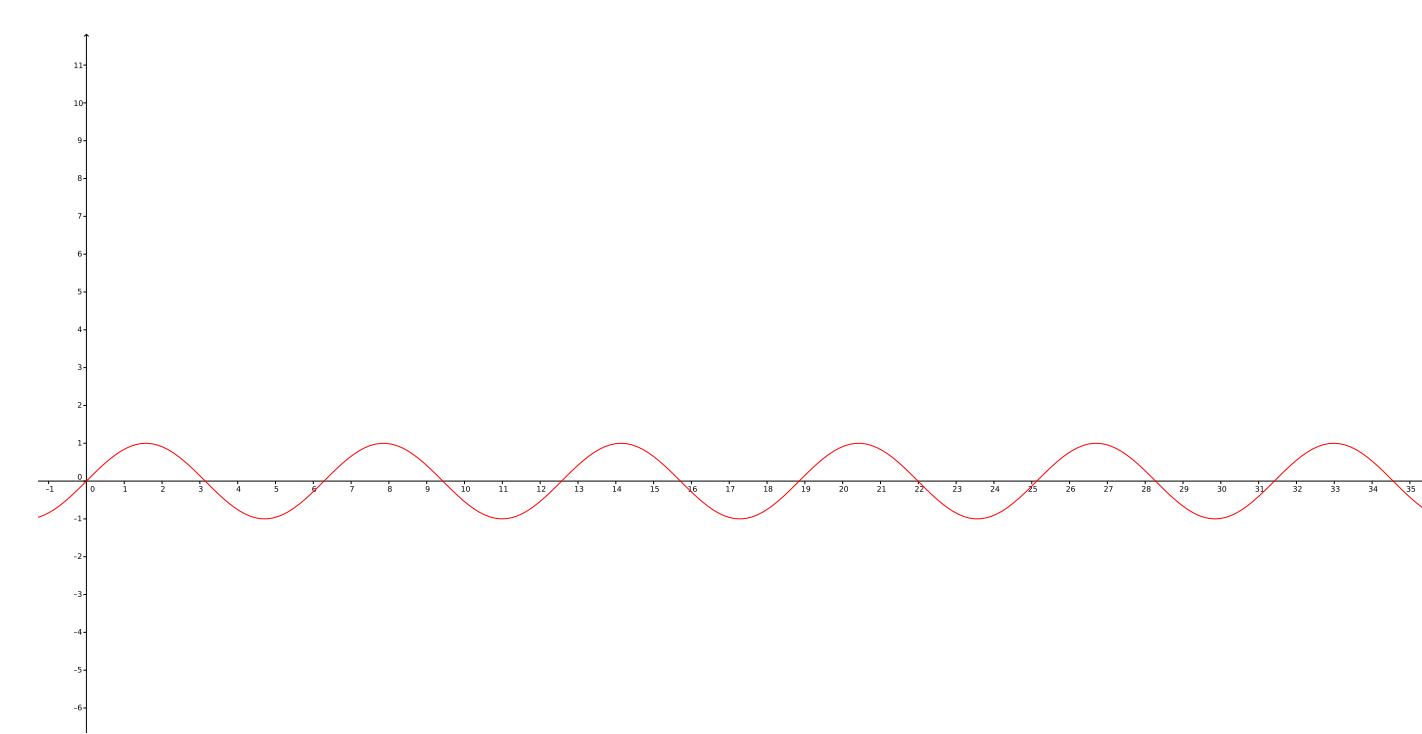
set xlabel "x"
set ylabel "y"
set title "Sine_Curve"

plot 'sine.dat' using 1:2
with lines lc rgb "red"
lt 2 lw 4.0 title "Sine"
```



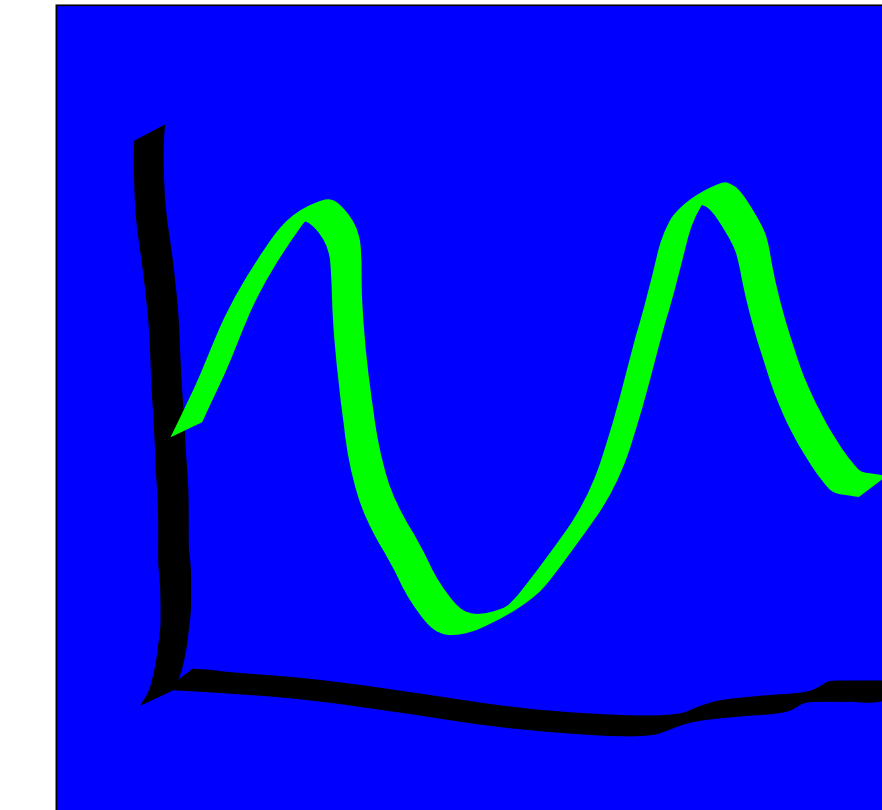
## Geogebra

A nice program for drawing geometry or simple plot sketches. Just draw your picture and then export as pdf.



## Inkscape

Just draw your image and save as pdf. I would uncheck rasterize option.



## Matplotlib

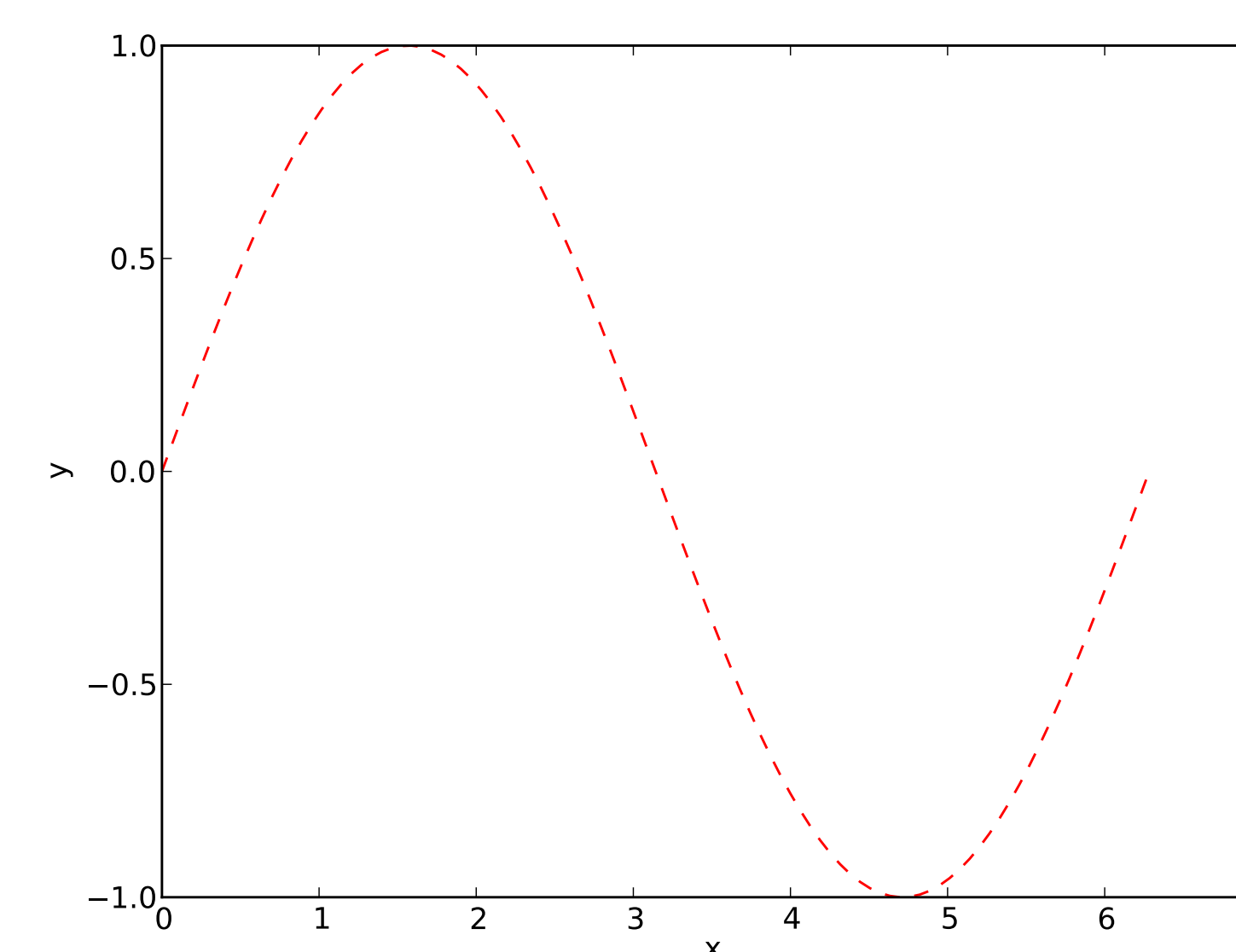
```
#!/bin/sh env python

# Packages
import matplotlib.pyplot as plt

# Read in data
with open('sine.dat', 'r') as fh:
    data = fh.readlines()

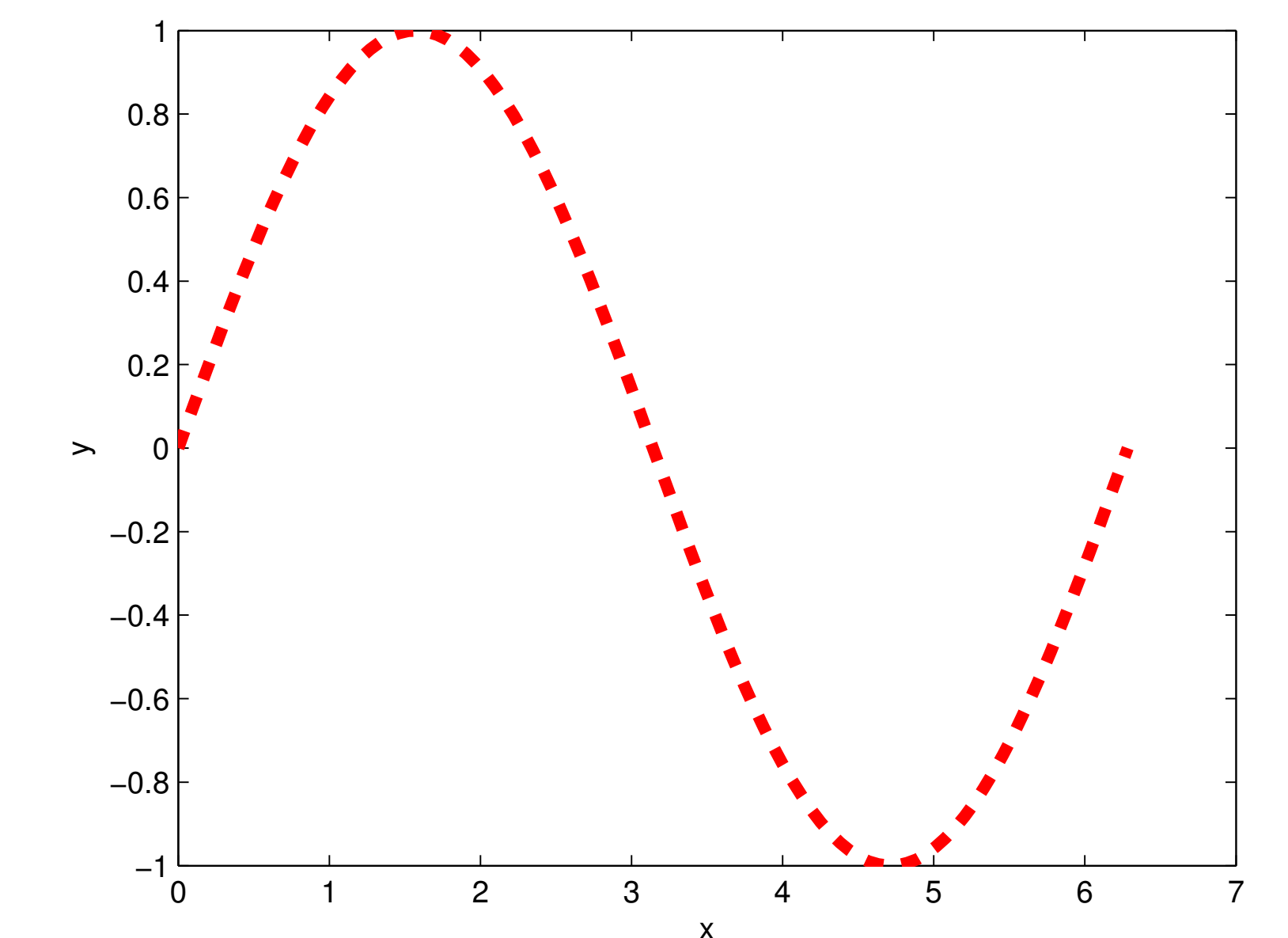
# Split data into x and y
x = []
y = []
for i in range(len(data)):
    x.append(float(data[i].split()[0]))
    y.append(float(data[i].split()[1]))

# Plot
plt.plot(x, y, 'r--')
plt.xlabel('x')
plt.ylabel('y')
plt.savefig('sine_matplotlib.pdf')
```



## Matlab

I am going to assume you know how to plot in Matlab. Just save as an eps. Once you have the eps file you need to convert it to pdf. If you use linux there is a command `epstopdf` that you can use.



## Comparison of Images

