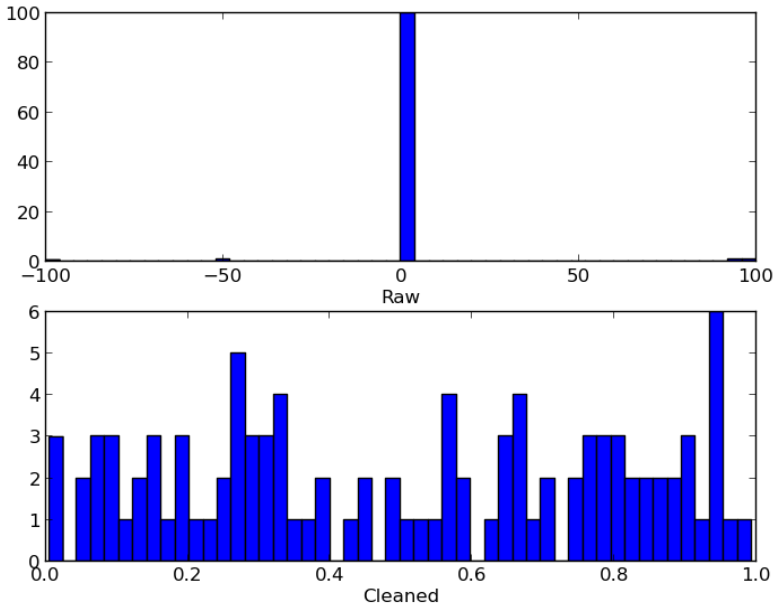
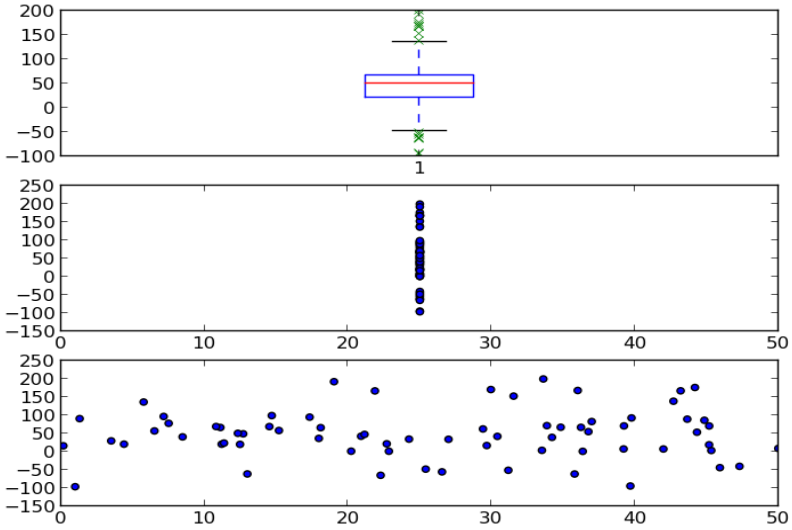
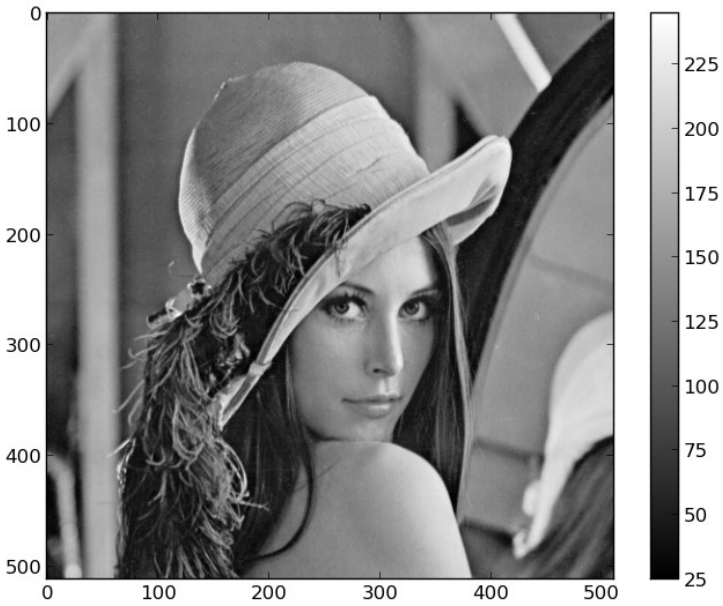
# 2
# Knowing Your Data

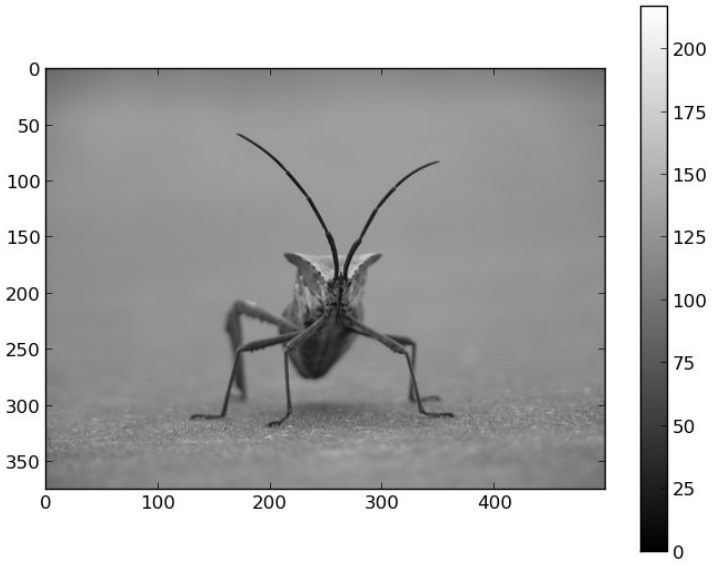
## Cleaning up data from outliers

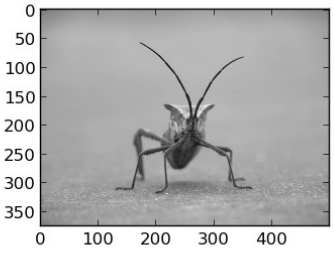


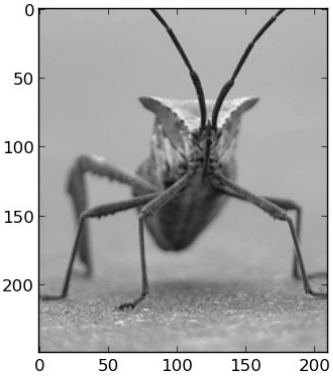

200
150
100
50
0
−50
−100
1
250
200
150
100
50
0
−50
−100
−150
0
10
20
30
40
50
250
200
150
100
50
0
−50
−100
−150
0
10
20
30
40
50

# Importing image data into NumPy arrays

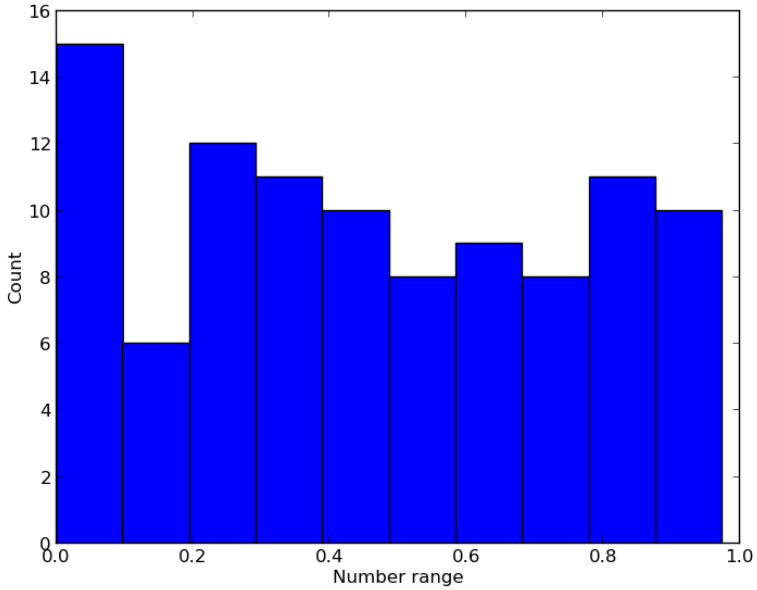


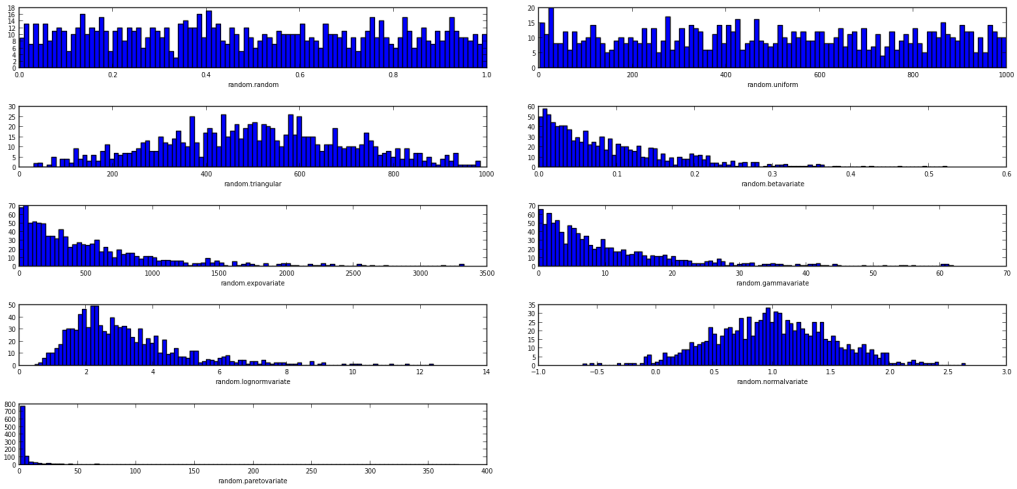

0
50
100
150
200
250
300
350
0
100
200
300
400
200
175
150
125
100
75
50
25
0


0
50
100
150
200
250
300
350
0
100
200
300
400

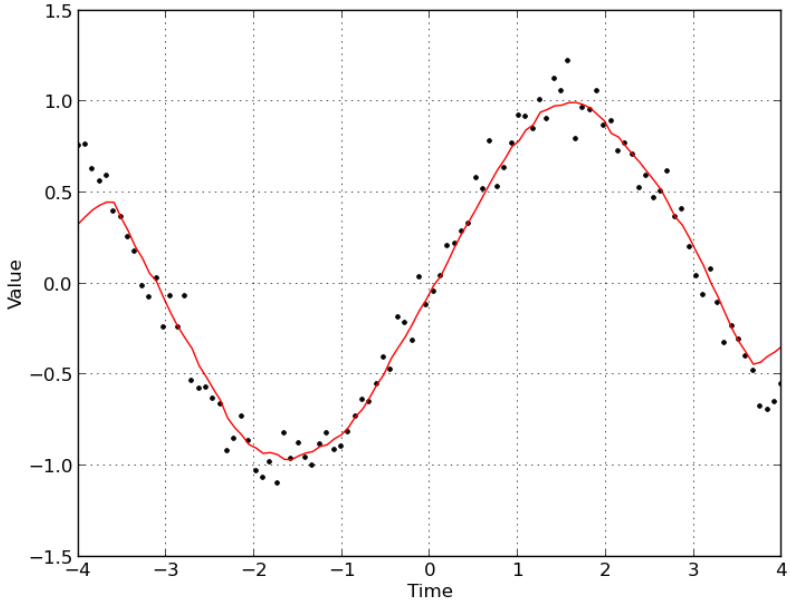

0
50
100
150
200
0
50
100
150
200

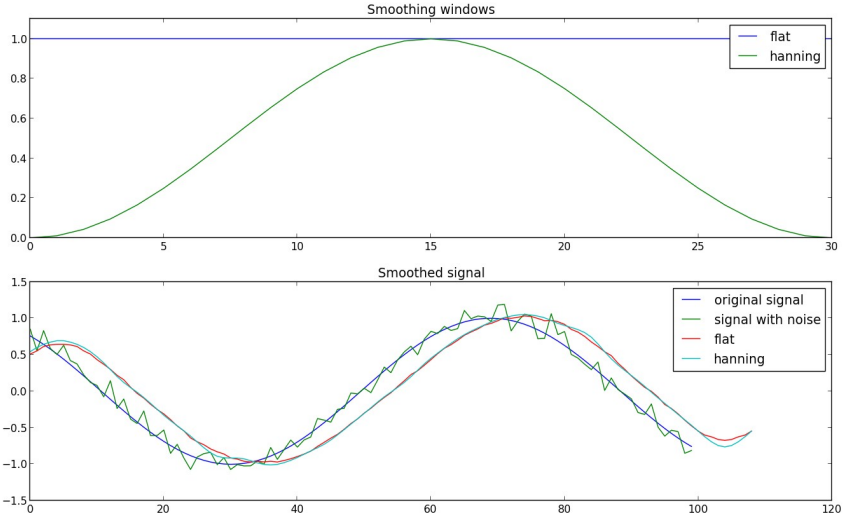
# Generating controlled random datasets

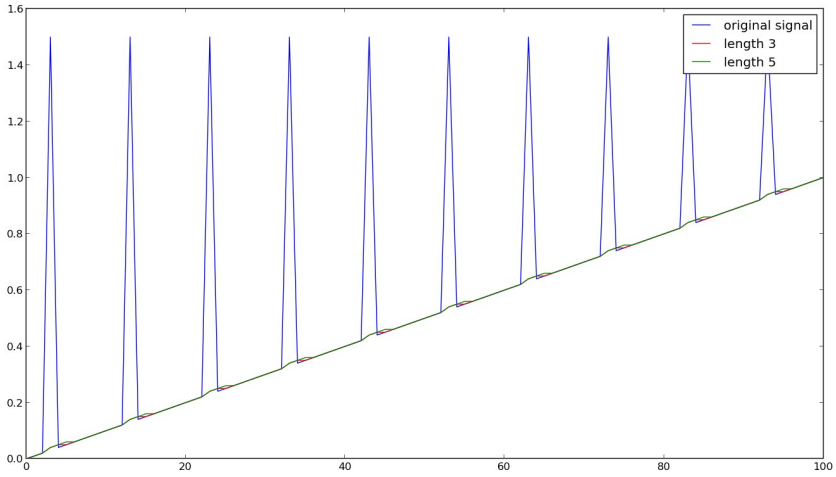


# Smoothing the noise in real-world data



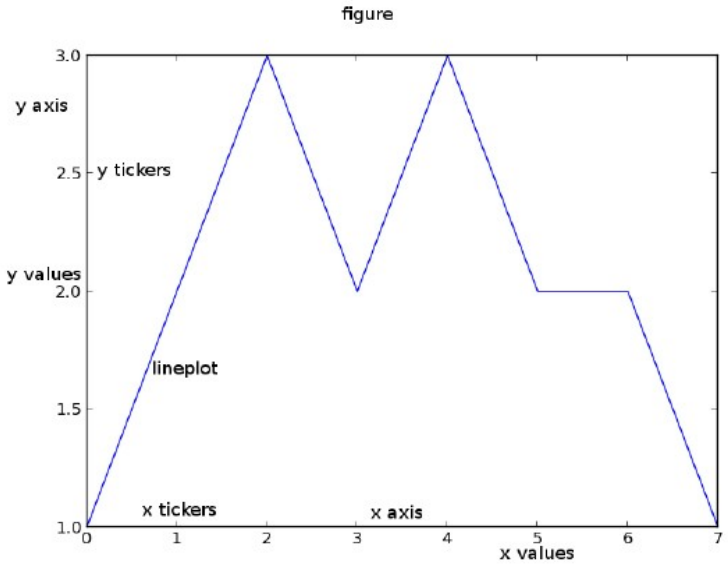



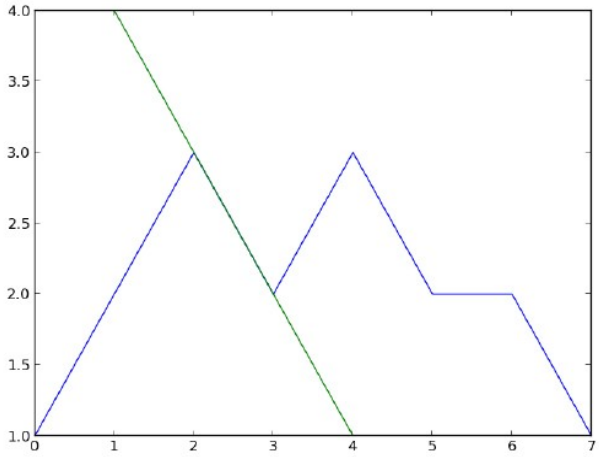

original signal
length 3
length 5
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
0
20
40
60
80
100

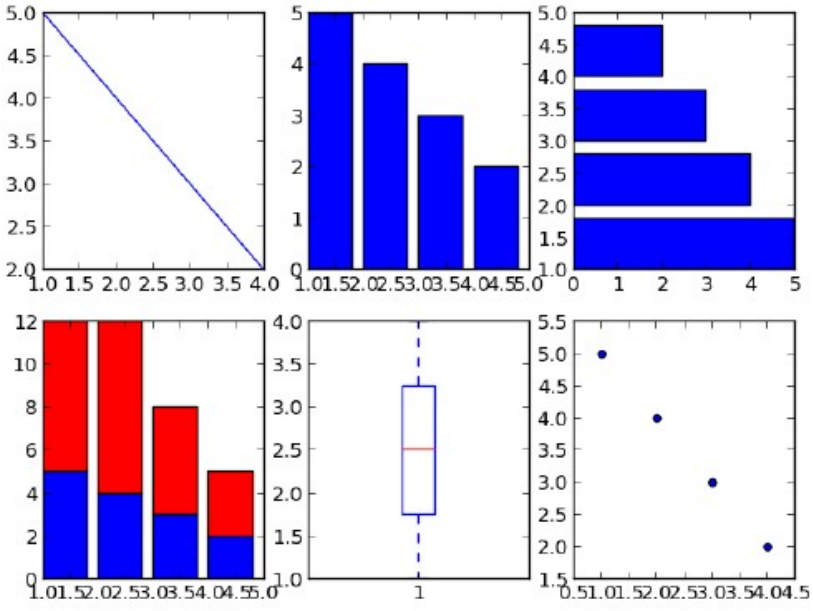
# 3

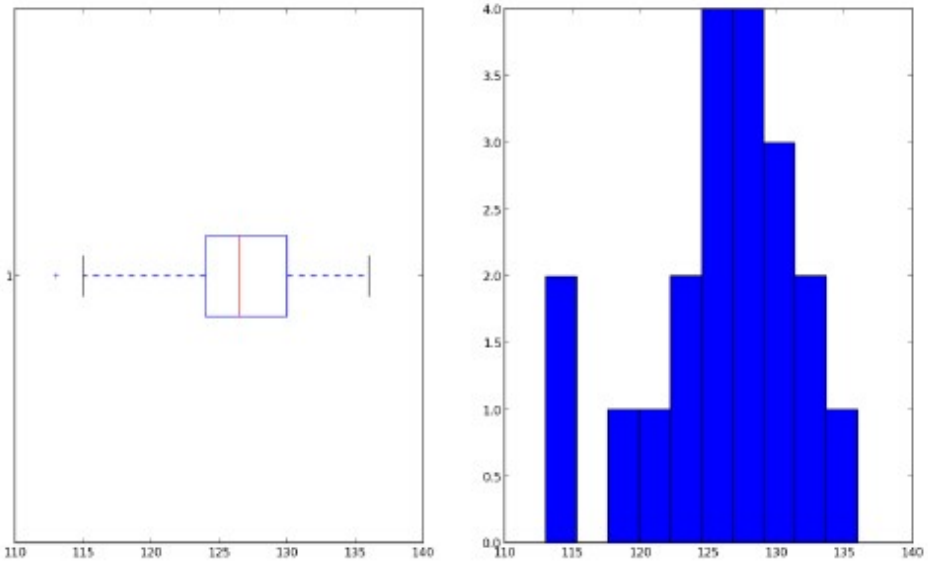
# Drawing Your First Plots and Customizing Them

## Defining plot types – bar, line, and stacked charts

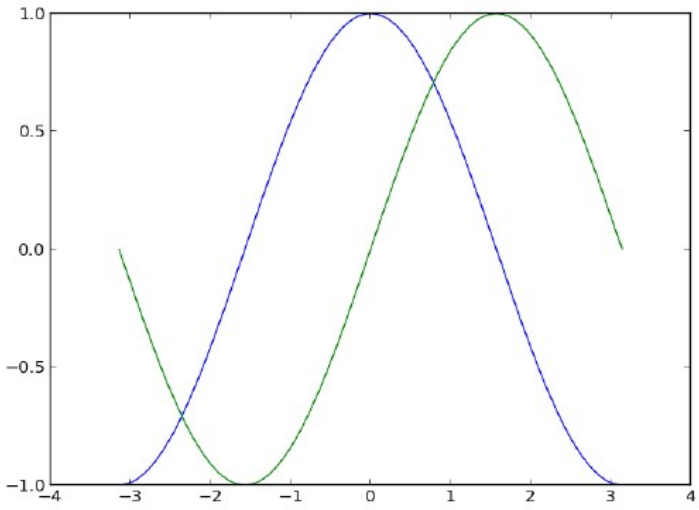


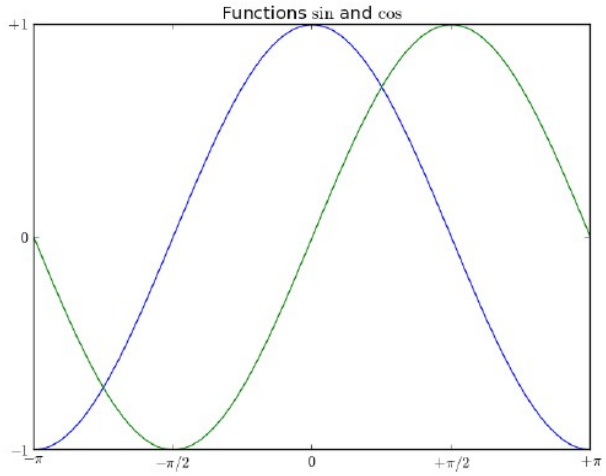




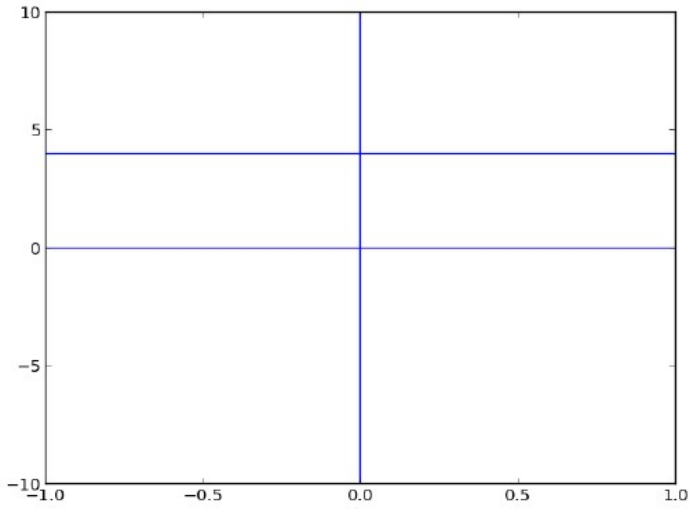



## Drawing a simple sine and cosine plot

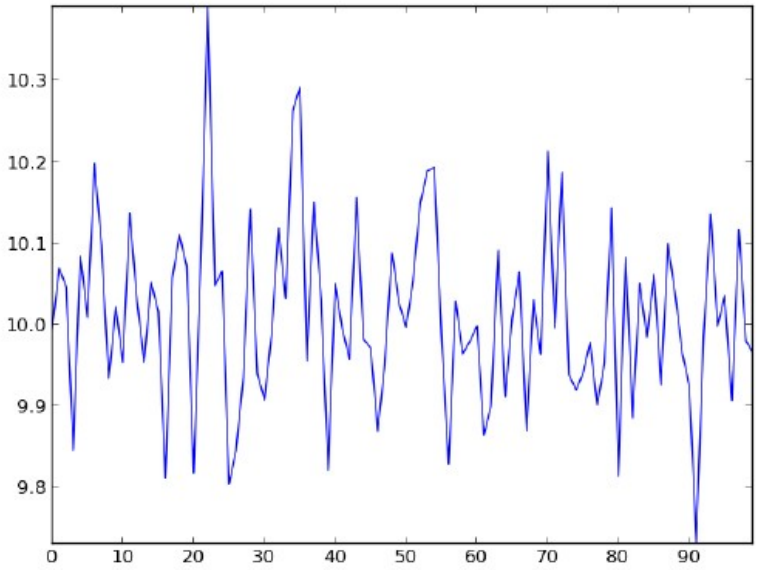


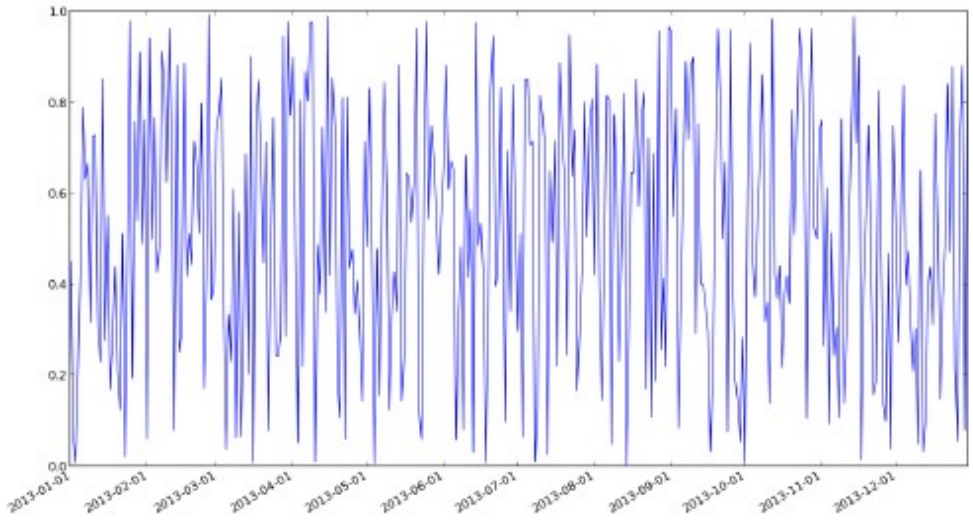



## Defining axis lengths and limits

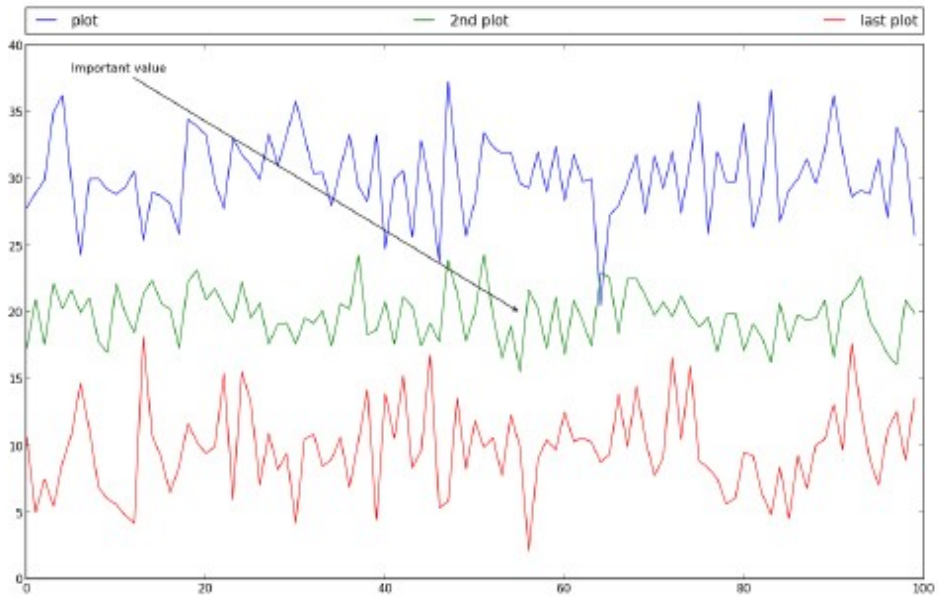


# Setting ticks, labels, and grids

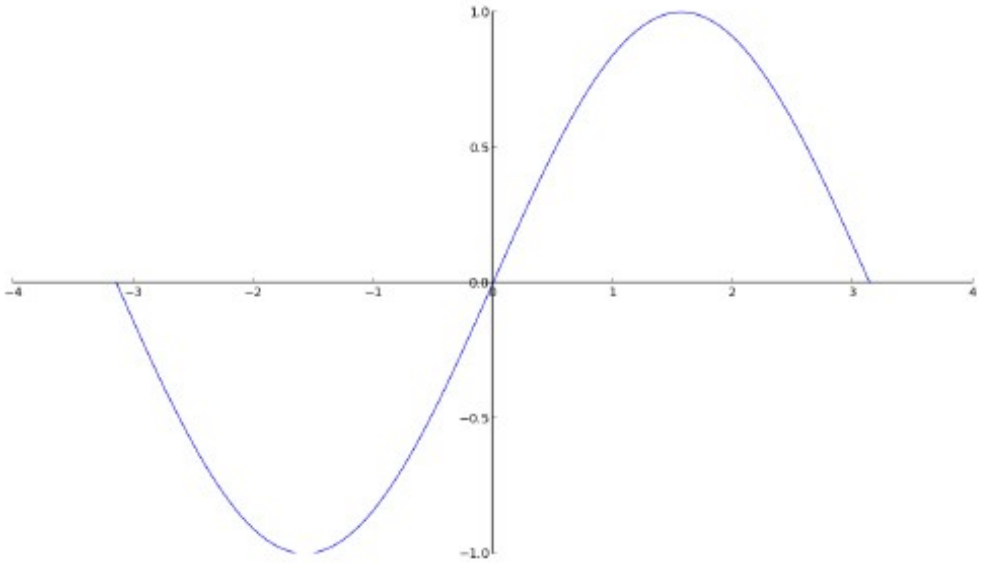


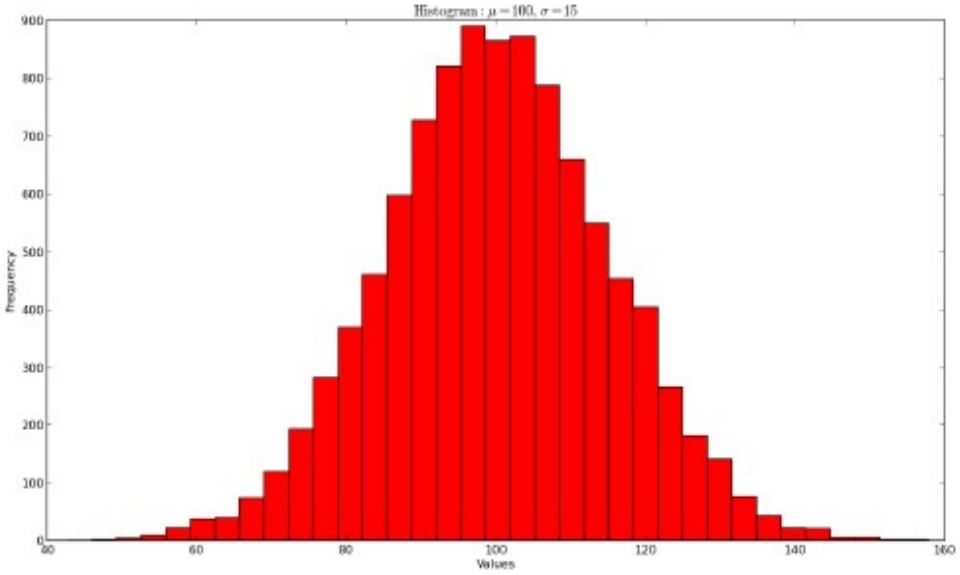
# Adding a legend and annotations



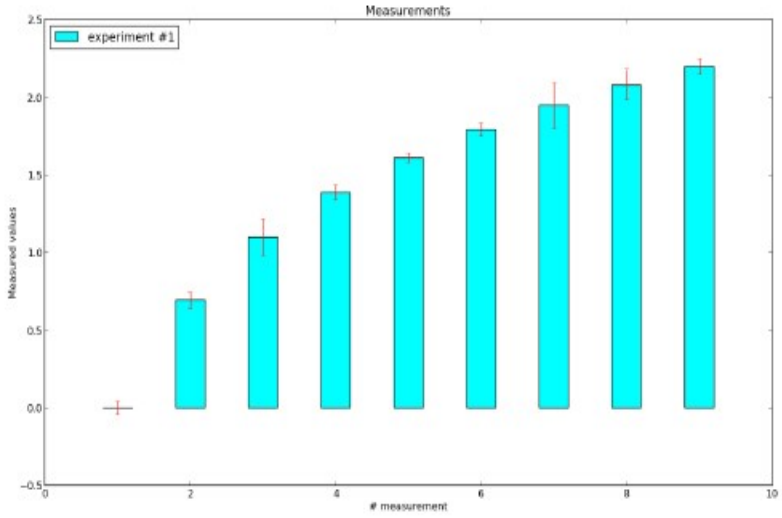
# Moving spines to the center



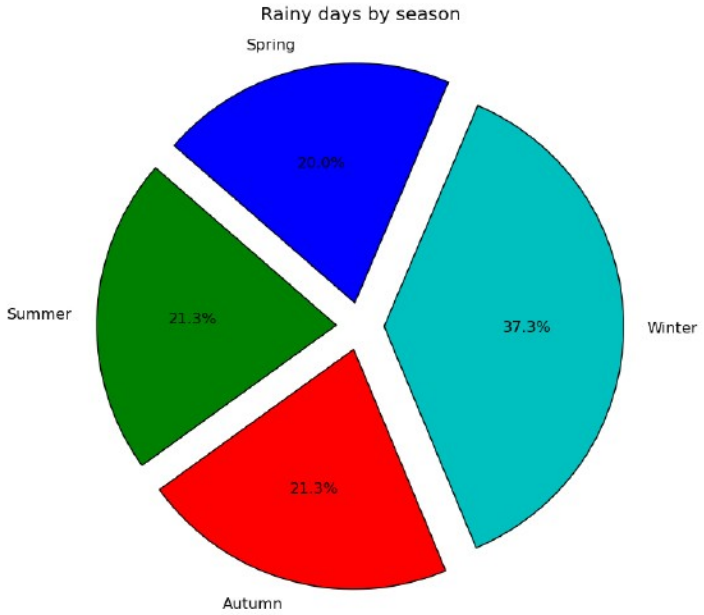
# Making histograms



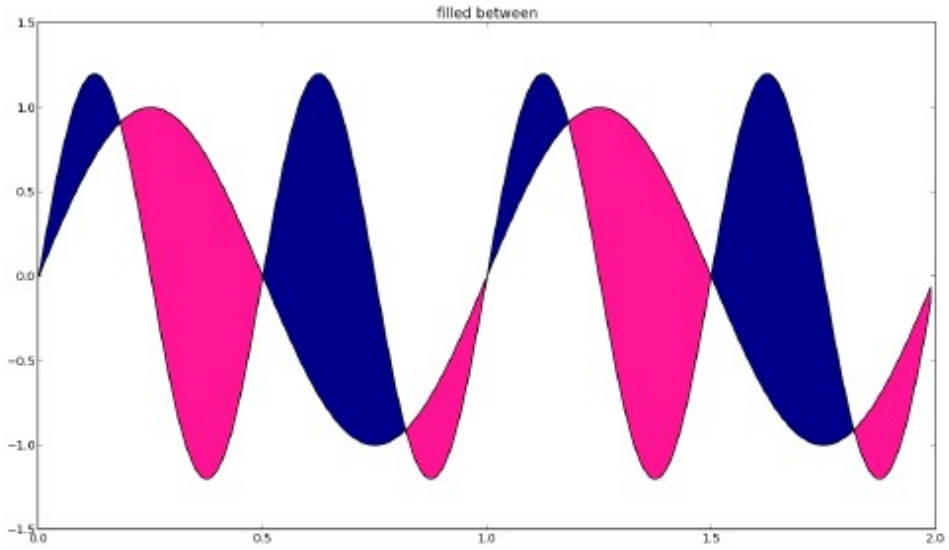
# Making bar charts with error bars



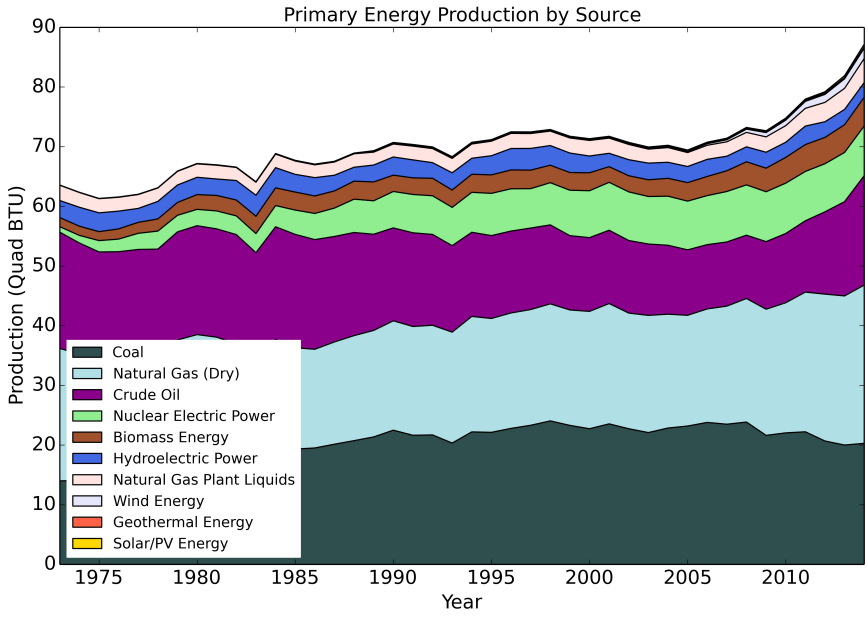
# Making pie charts count



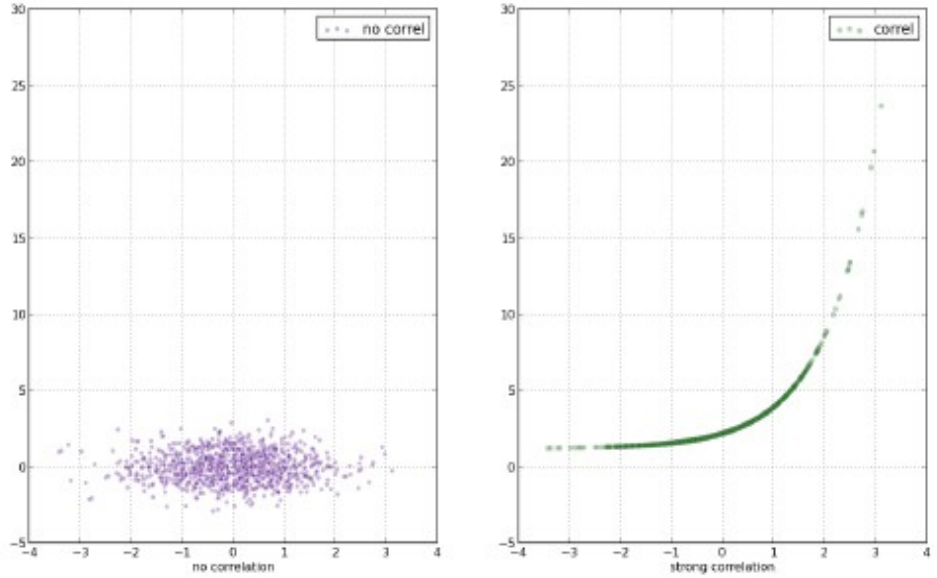
# Plotting with filled areas



# Making stacked plots



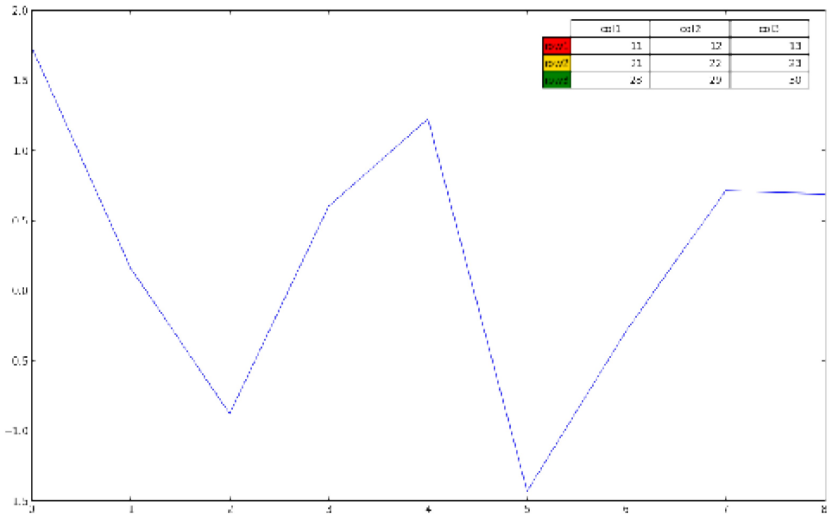
# Drawing scatter plots with colored markers



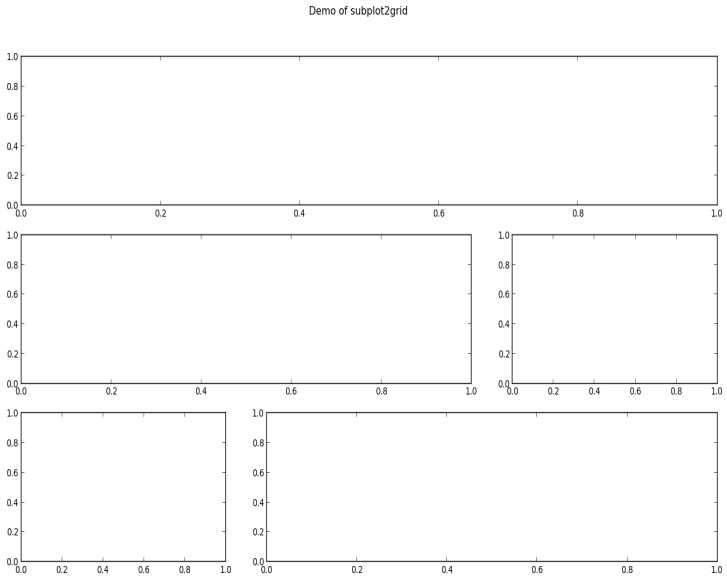
# 4

# More Plots and Customizations

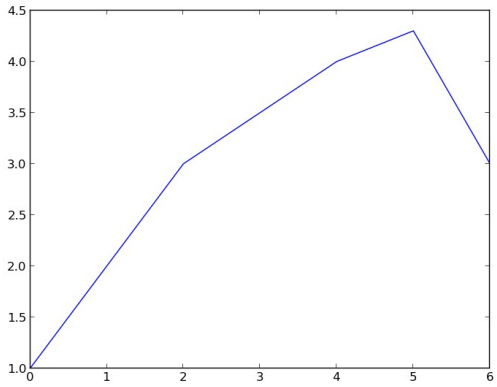
## Adding a data table to the figure

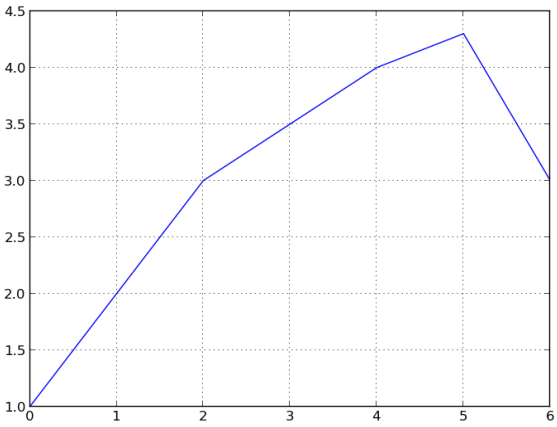


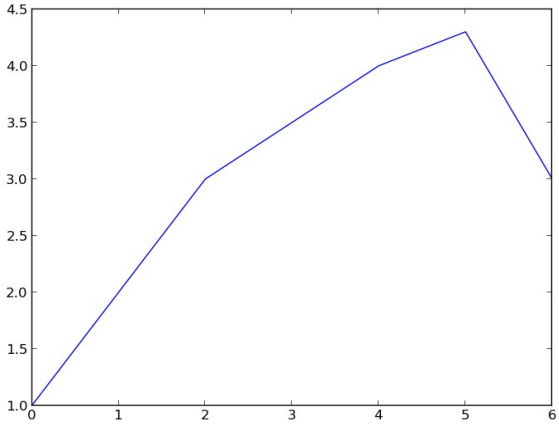
# Using subplots

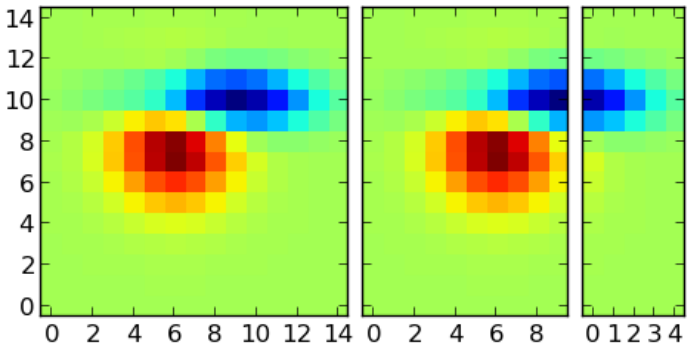


# Customizing grids

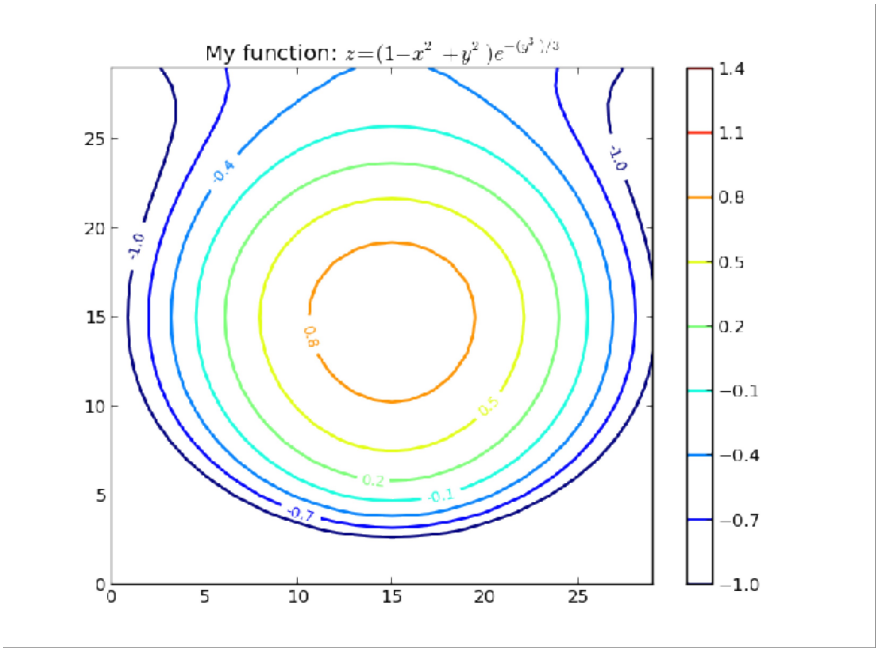


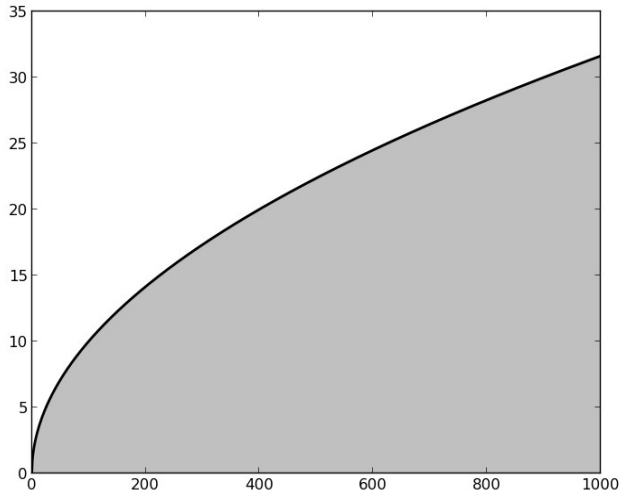

4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0
1
2
3
4
5
6

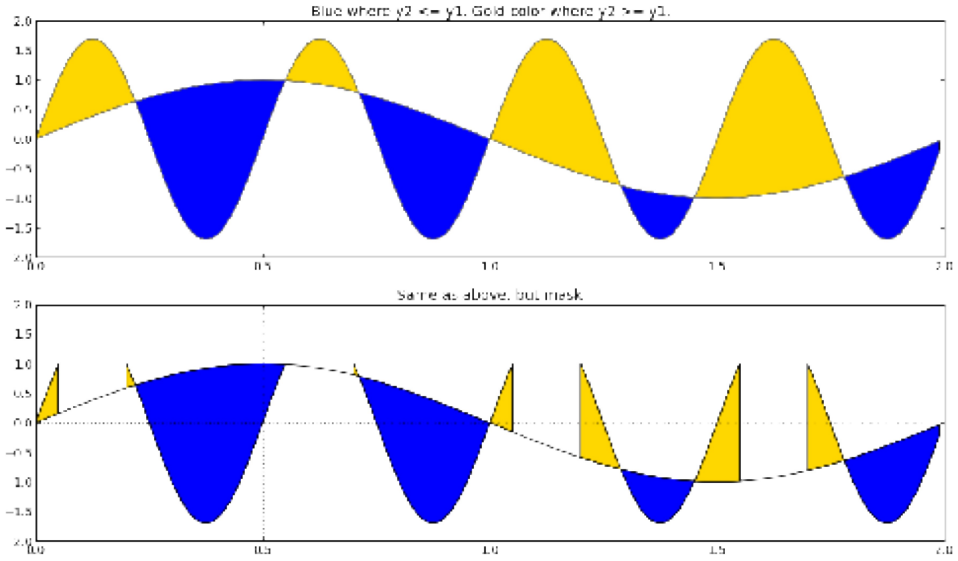

14
12
10
8
6
4
2
0
0 2 4 6 8 10 12 14
0 2 4 6 8
0 1 2 3 4

# Creating contour plots

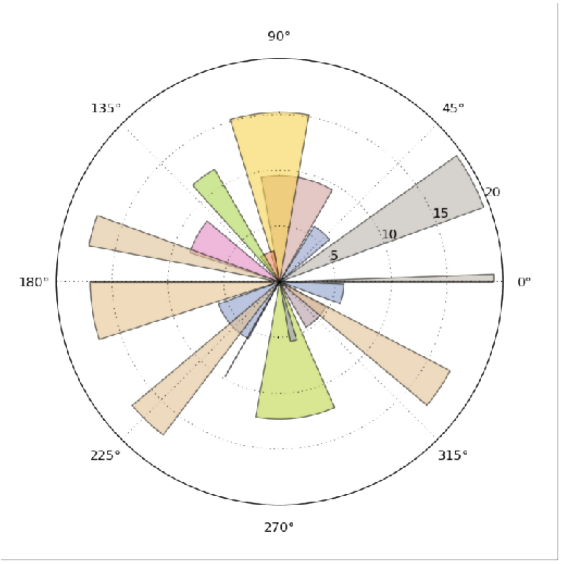


# Filling an under-plot area

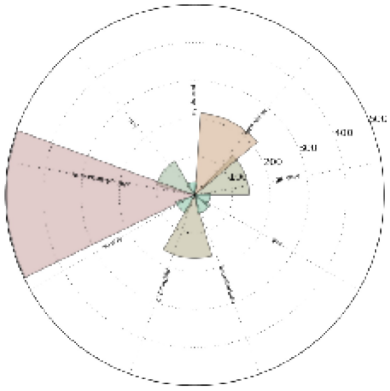


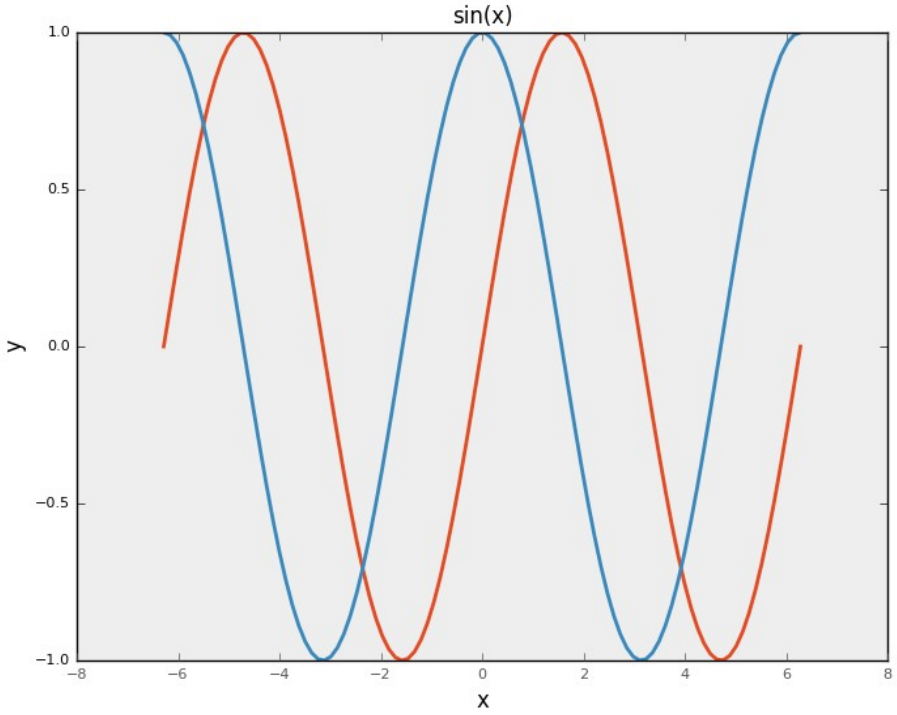
# Drawing polar plots



# Visualizing the filesystem tree using a polar bar

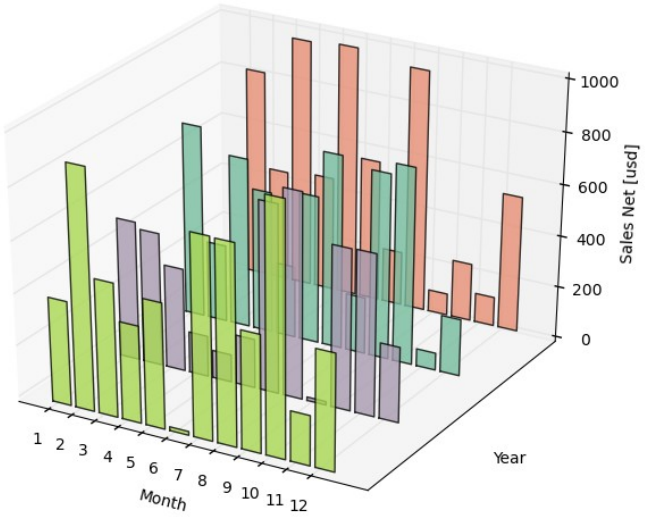


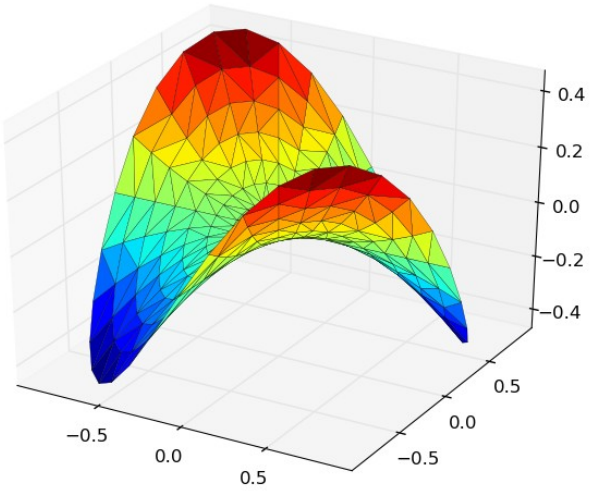
# Customizing matplotlib with style



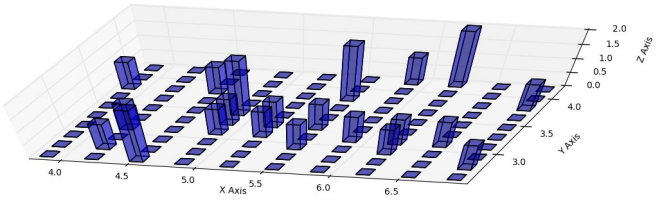
# 5
# Making 3D visualizations

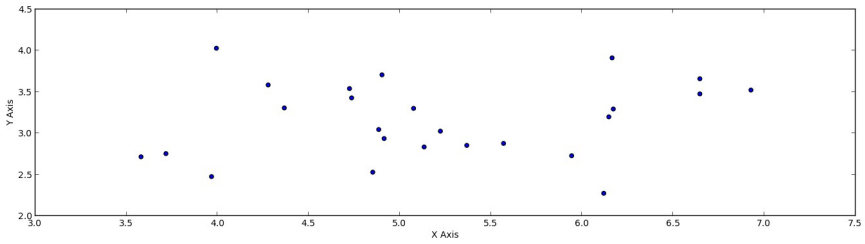
## Creating 3D bars



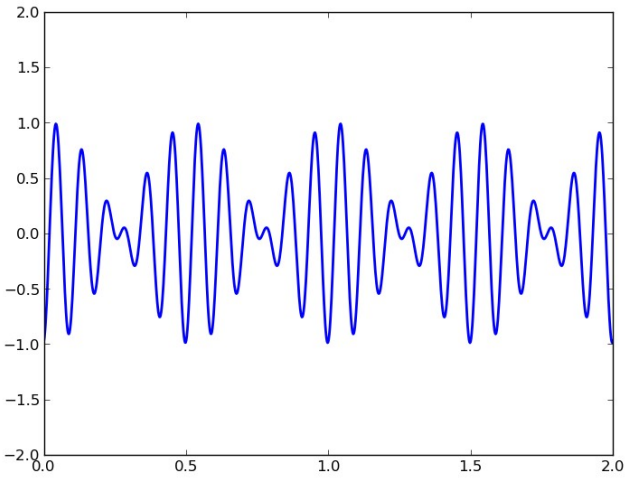

0.4
0.2
0.0
−0.2
−0.4
0.5
0.0
−0.5
−0.5
0.0
0.5

## Creating 3D histograms

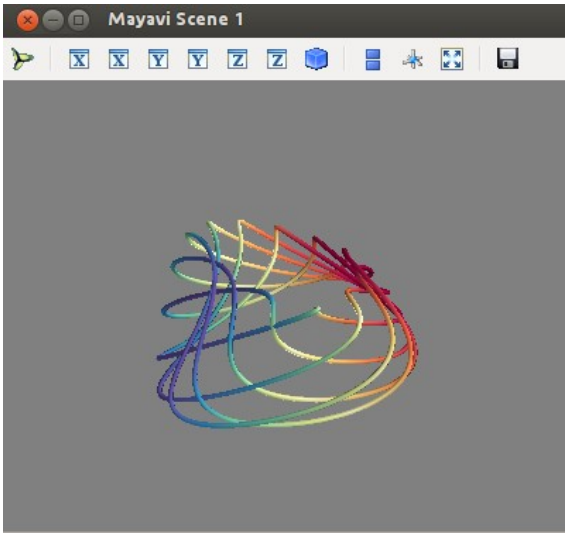



## Animating in matplotlib



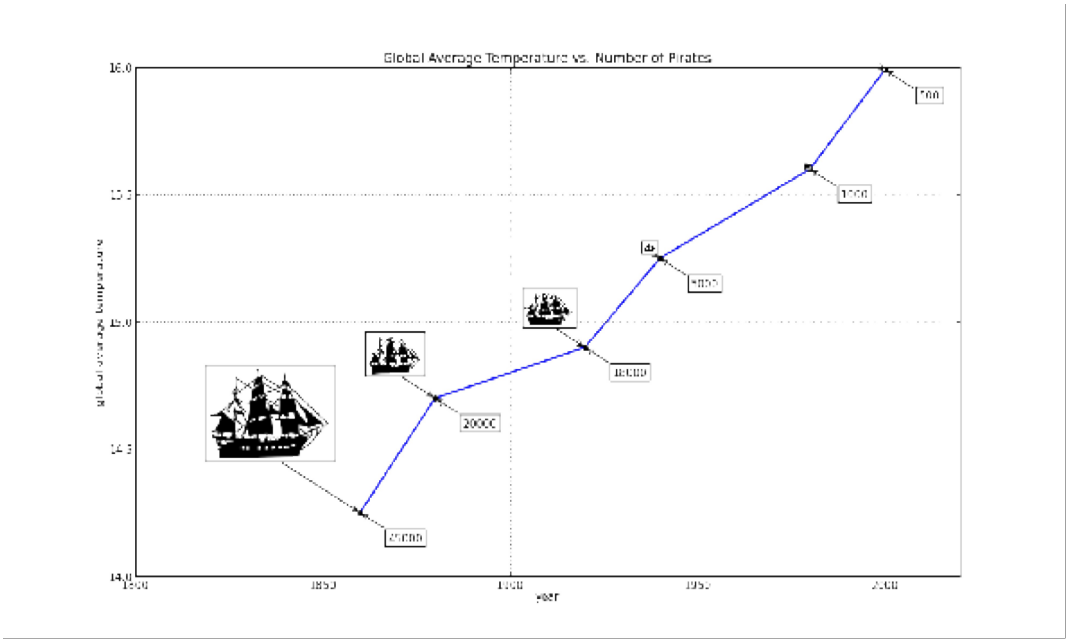
# Animating with OpenGL



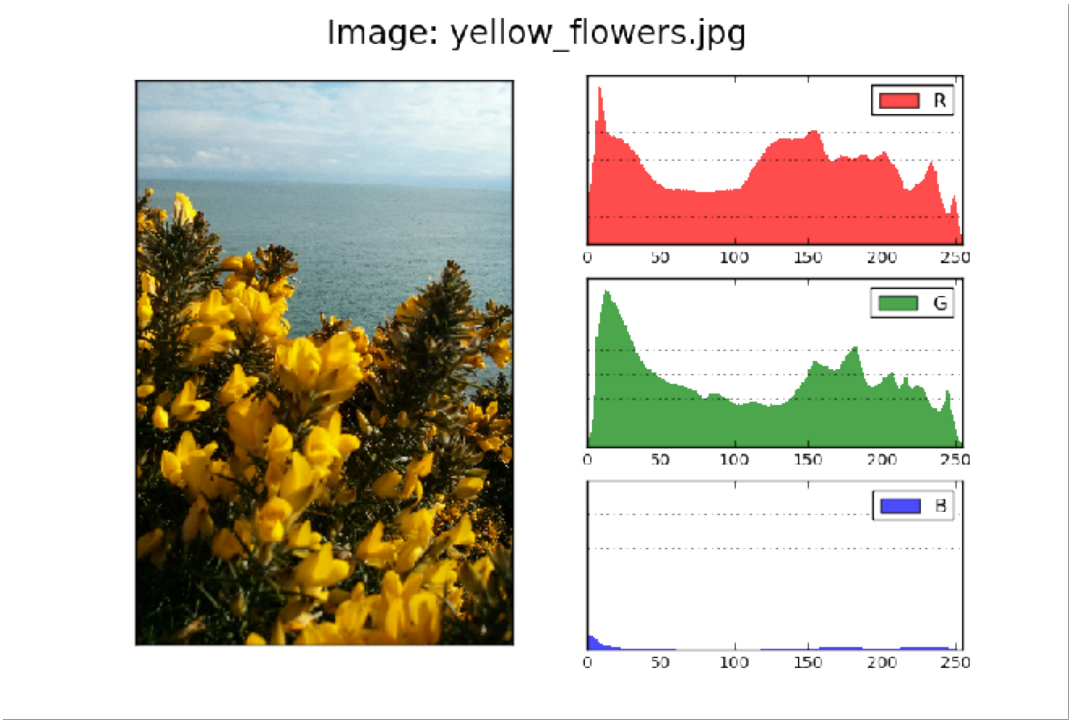
# 6

# Plotting Charts with Images and Maps

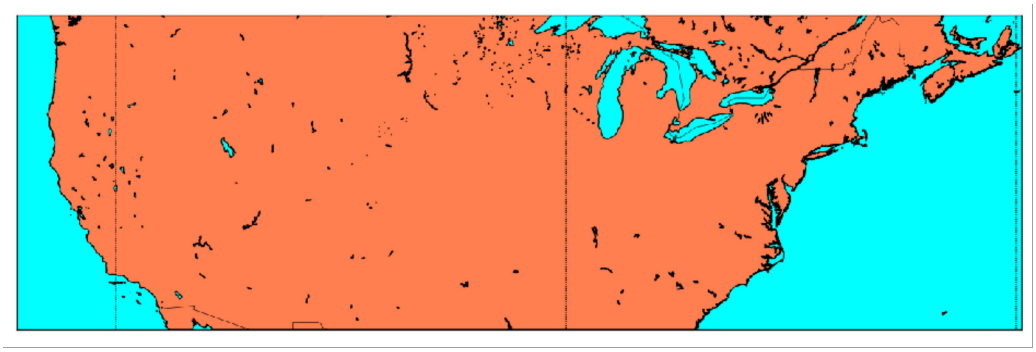
## Plotting with images



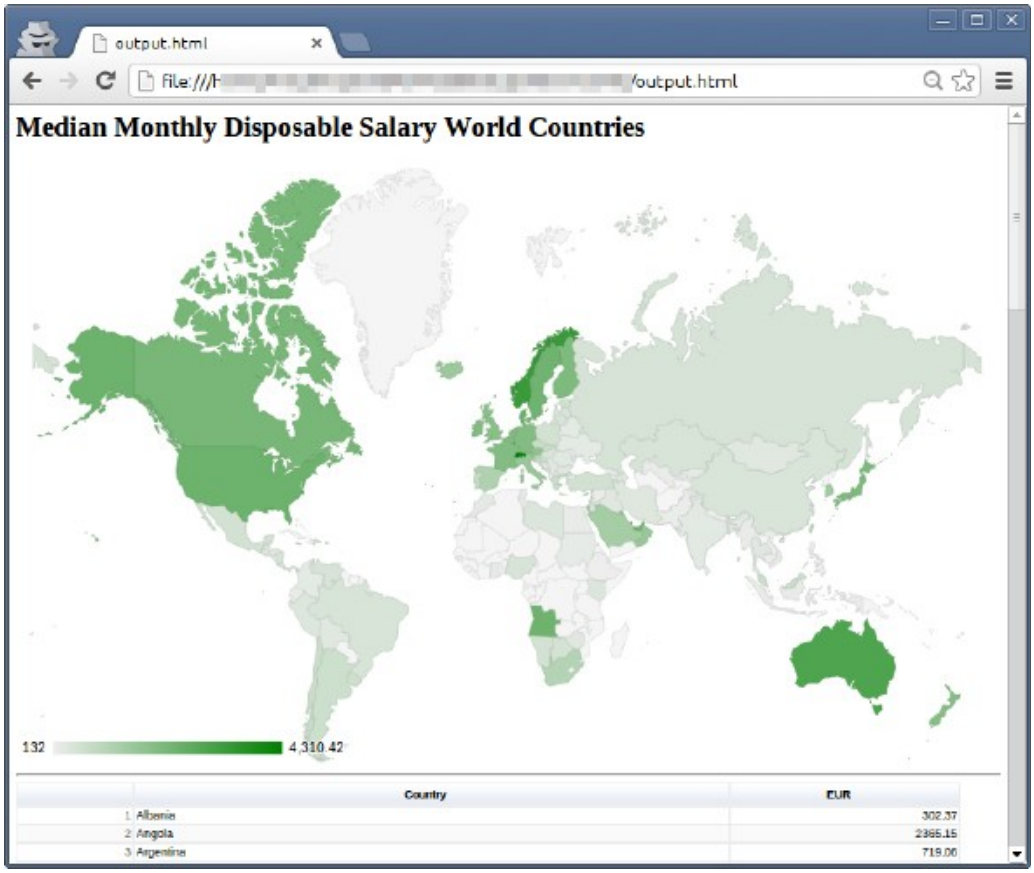
# Displaying an image with other plots in the figure



# Plotting data on a map using Basemap



# Plotting data on a map using Google Map API

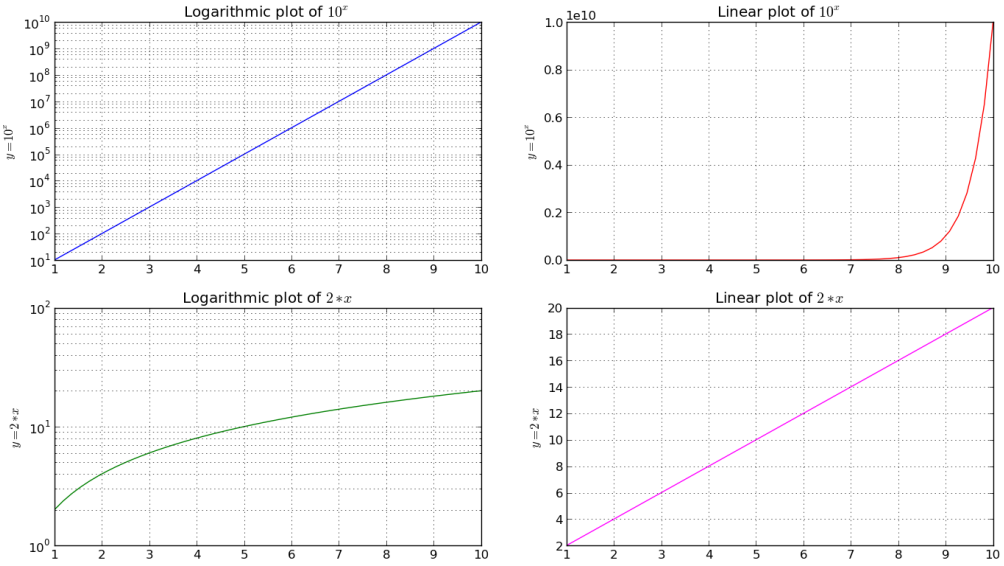


# Generating CAPTCHA images

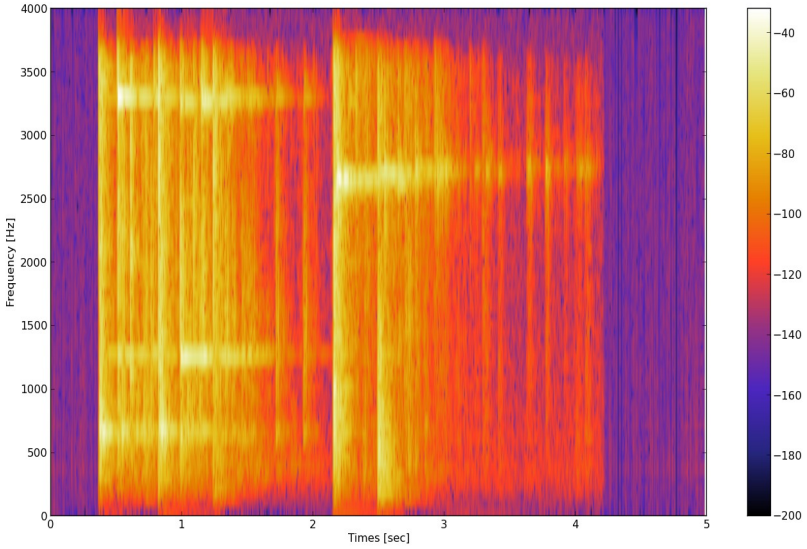
# 7

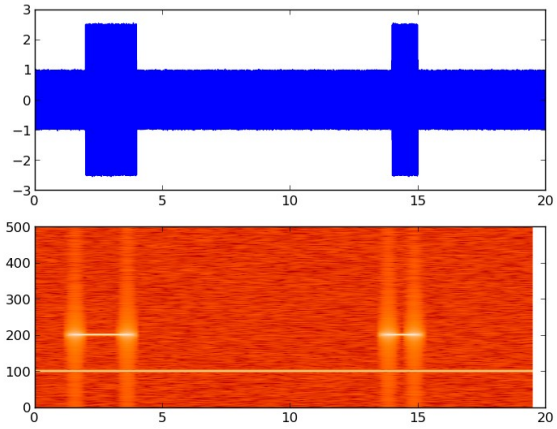
# Using the Right Plots to Understand Data

## Understanding logarithmic plots



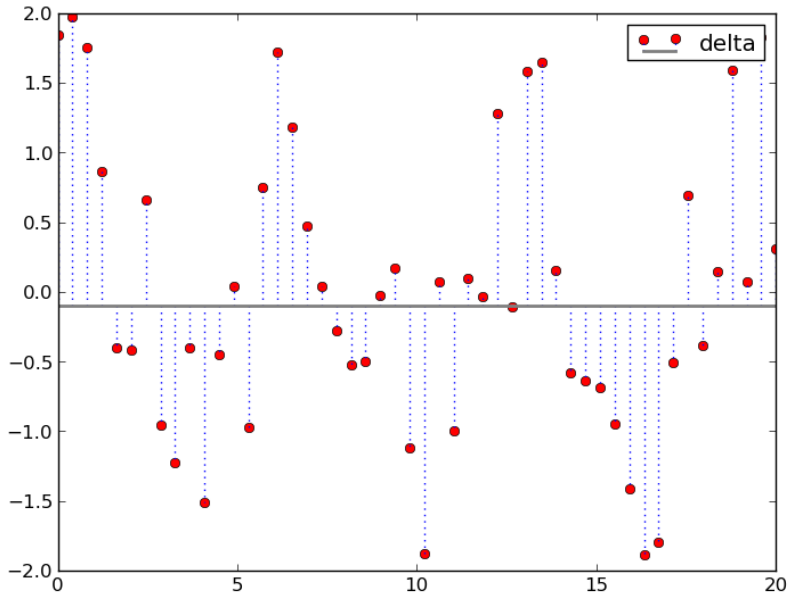
# Understanding spectrograms



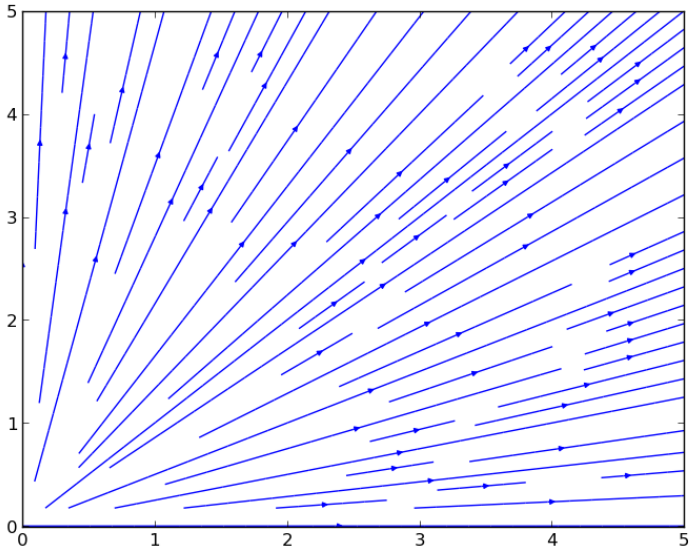



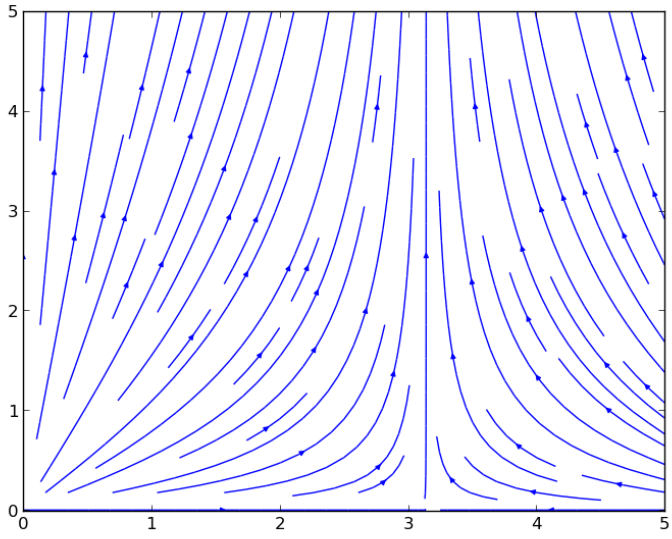
# Creating stem plot

```
steam | leaf
==========================
    0 | 6 7 8
    1 | 0 2 3 4 7 7 7 8 9
    2 | 1 3 4 4 5 7
    3 | 3 1 1 2 6 6 9
    4 | 1 5 5 6 9
    5 | 0
```

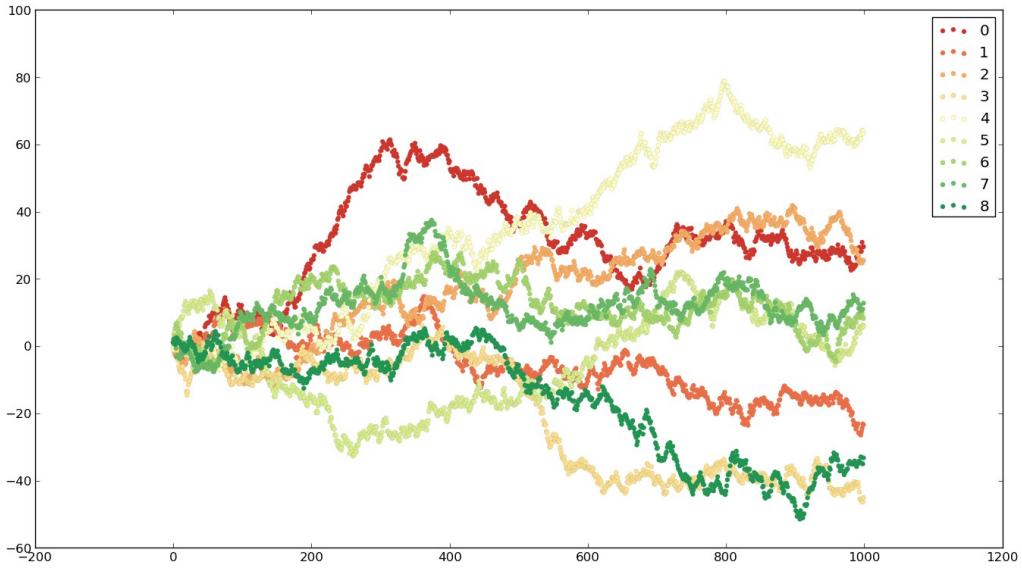


# Drawing streamlines of vector flow

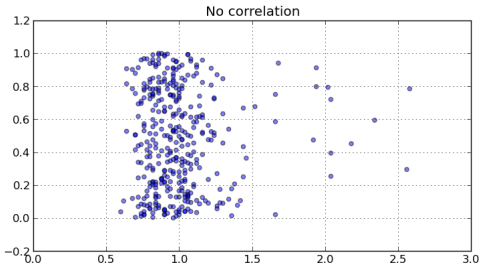


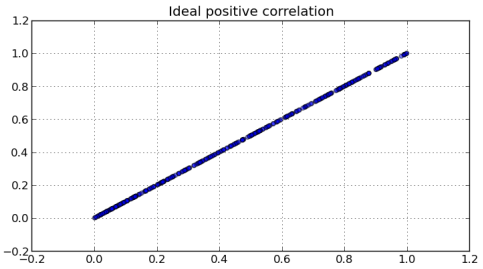



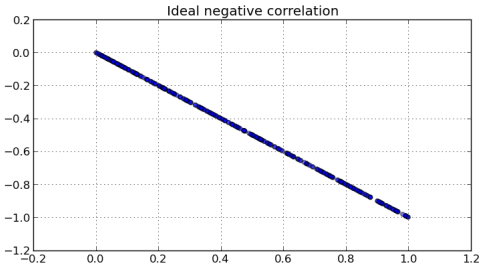
# Using colormaps

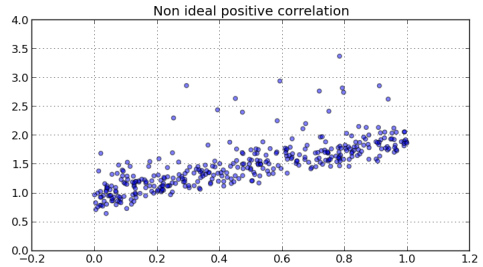


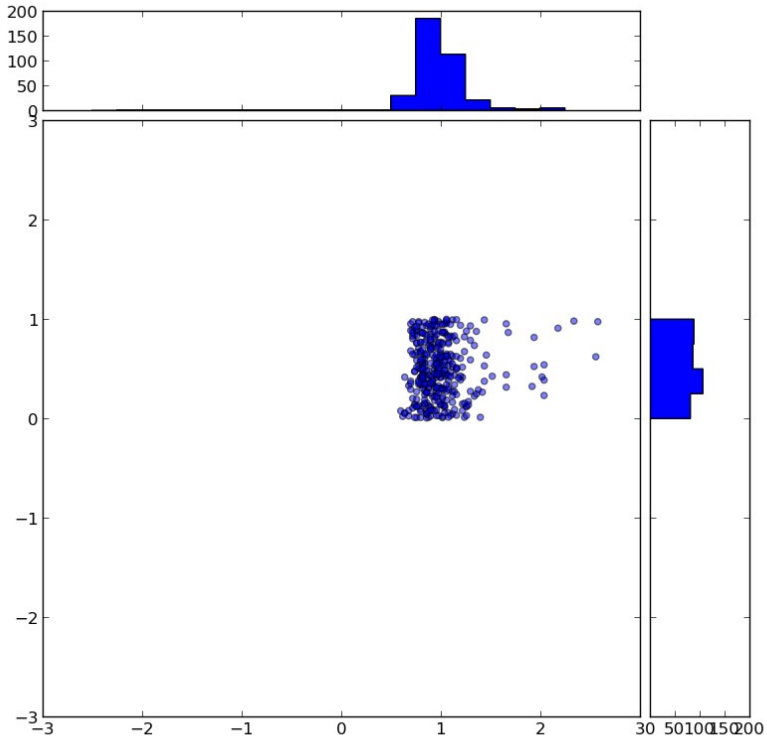
# Using scatter plots and histograms



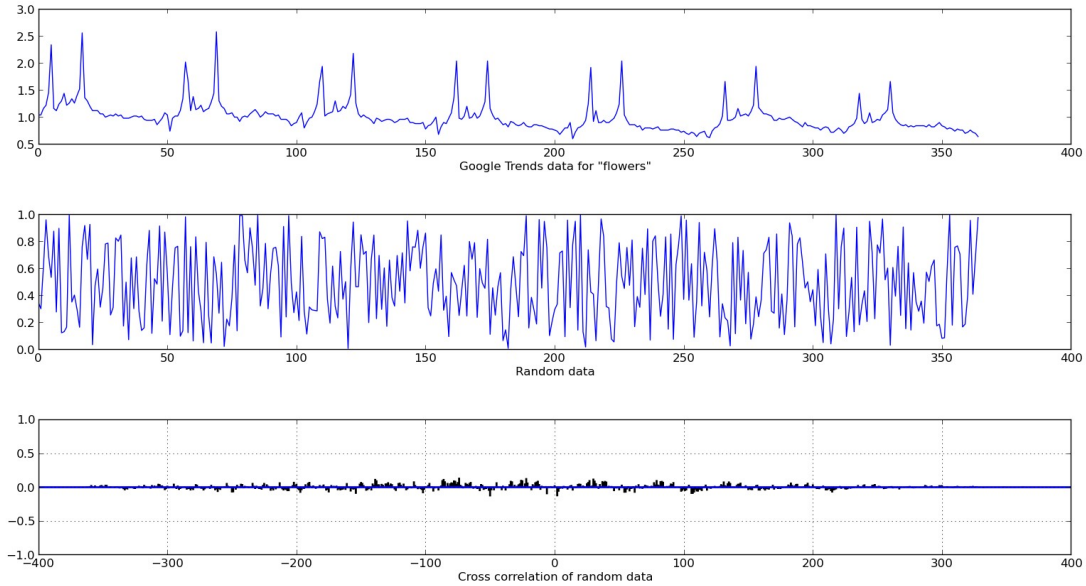



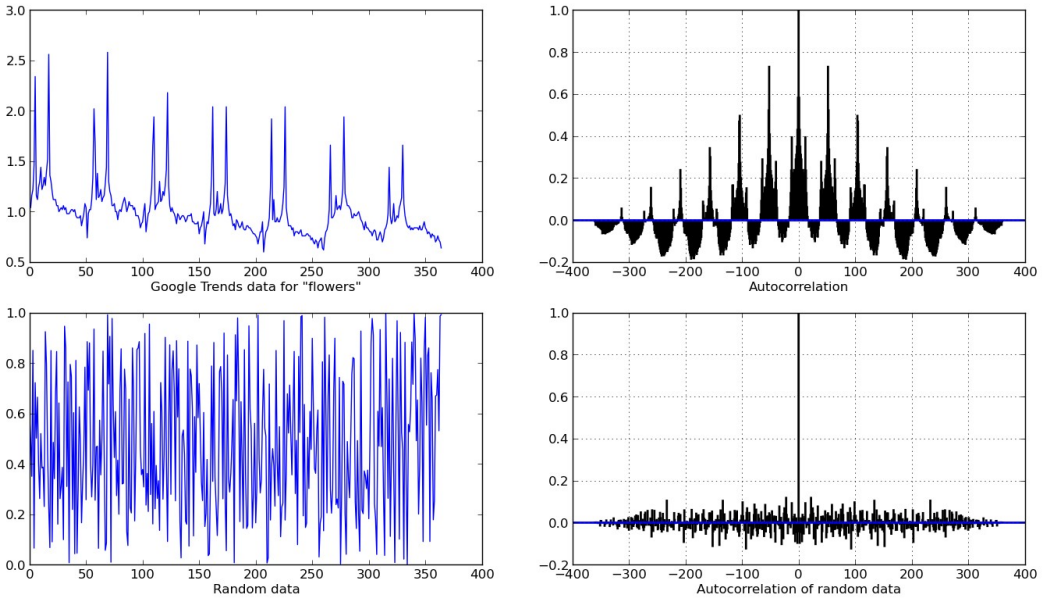




200
150
100
50
0
3
2
1
0
−1
−2
−3
−3
−2
−1
0
1
2
30
50
100
150
200

# Plotting the cross correlation between two variables



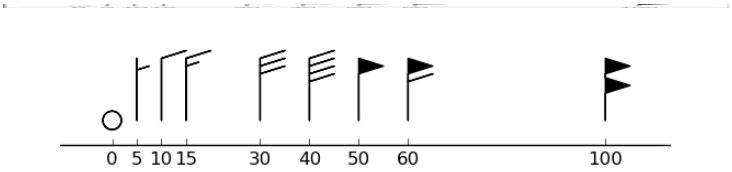
## Importance of autocorrelation

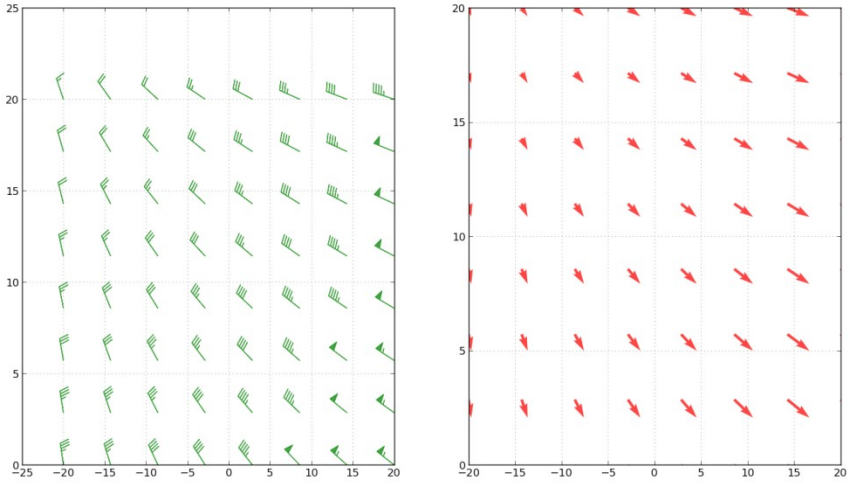



# 8

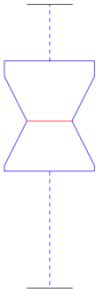
# More on matplotlib Gems

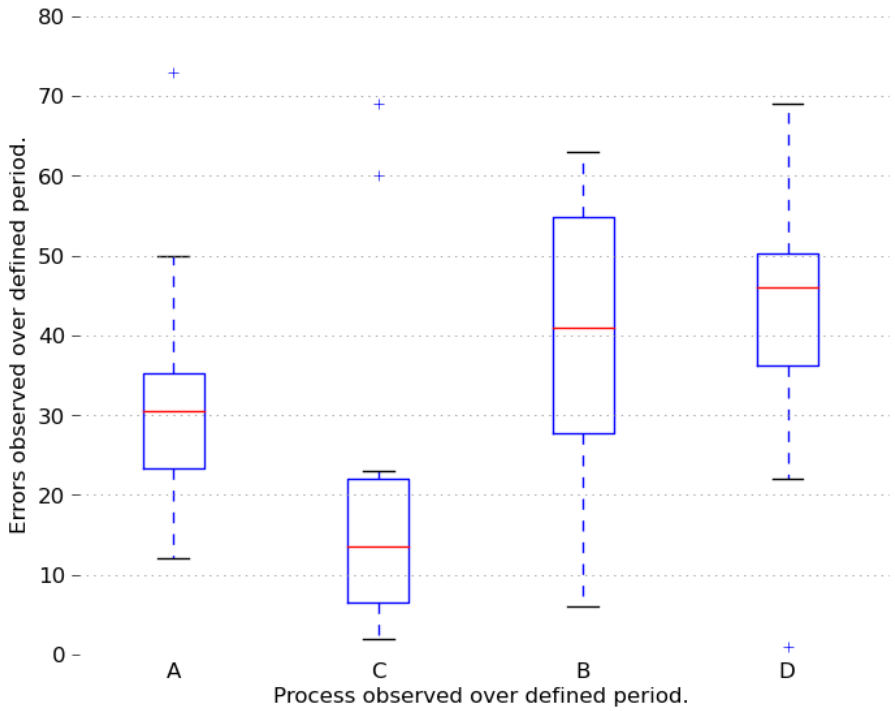
## Drawing barbs



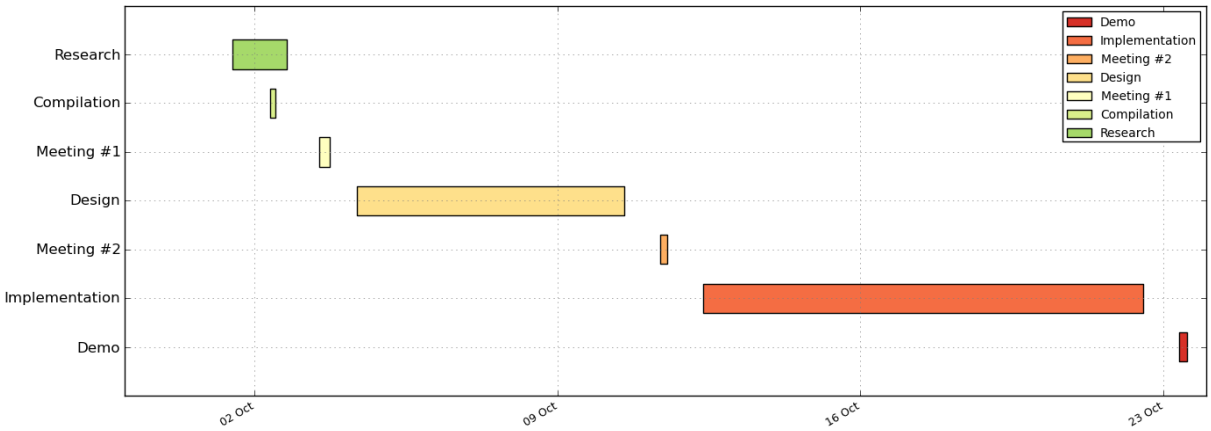




## Making a box and a whisker plot

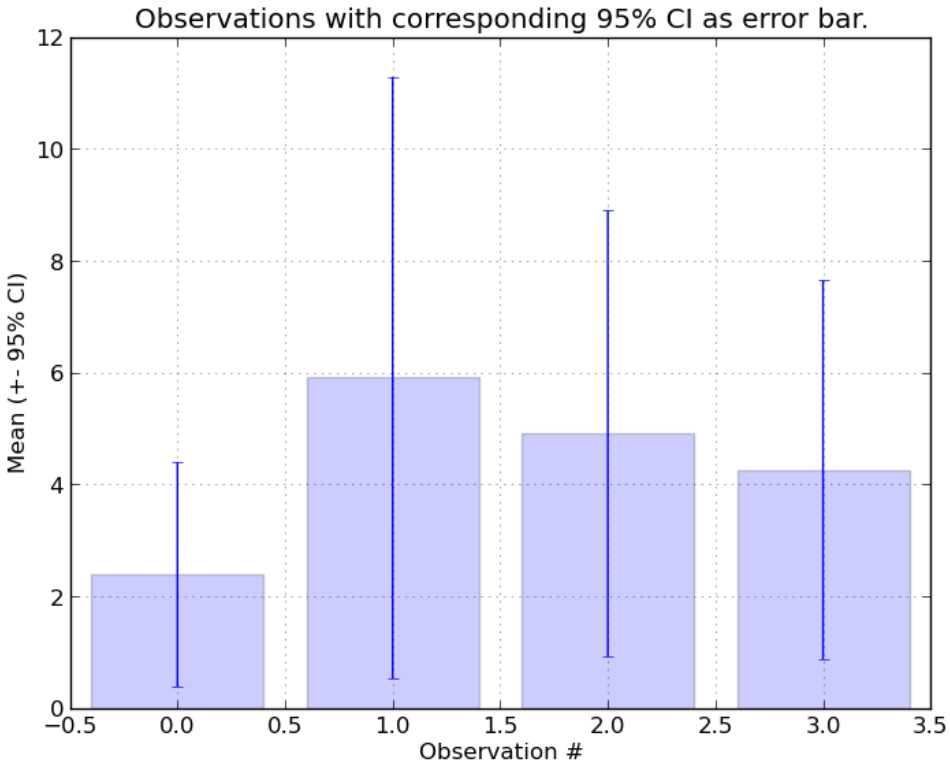





## Making Gantt charts



# Making error bars



# Making use of text and font properties

monospace xx-large
monospace x-large
monospace large
monospace medium
monospace small
monospace x-small
monospace xx-small

fantasy xx-large
fantasy x-large
fantasy large
fantasy medium
fantasy small
fantasy x-small
fantasy xx-small

cursive xx-large
cursive x-large
cursive large
cursive medium
cursive small
cursive x-small
cursive xx-small

sans-serif xx-large
sans-serif x-large
sans-serif large
sans-serif medium
sans-serif small
sans-serif x-small
sans-serif xx-small

serif xx-large
serif x-large
serif large
serif medium
serif small
serif x-small
serif xx-small

***black oblique***
***black italic***
**black normal**

***heavy oblique***
***heavy italic***
**heavy normal**

***bold oblique***
***bold italic***
**bold normal**

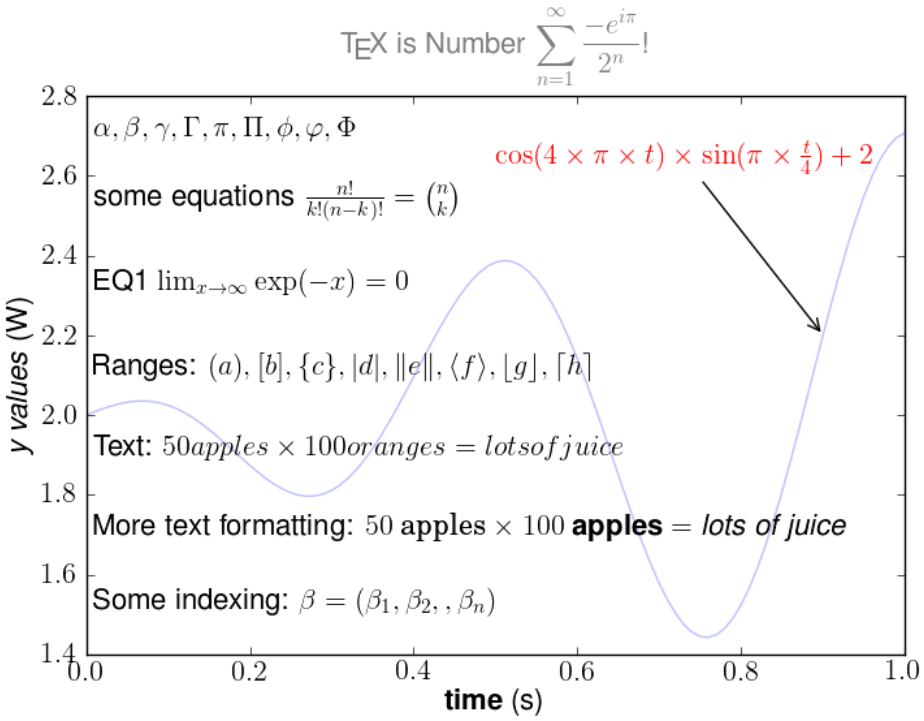
***semibold oblique***
***semibold italic***
**semibold normal**

*medium oblique*
*medium italic*
medium normal

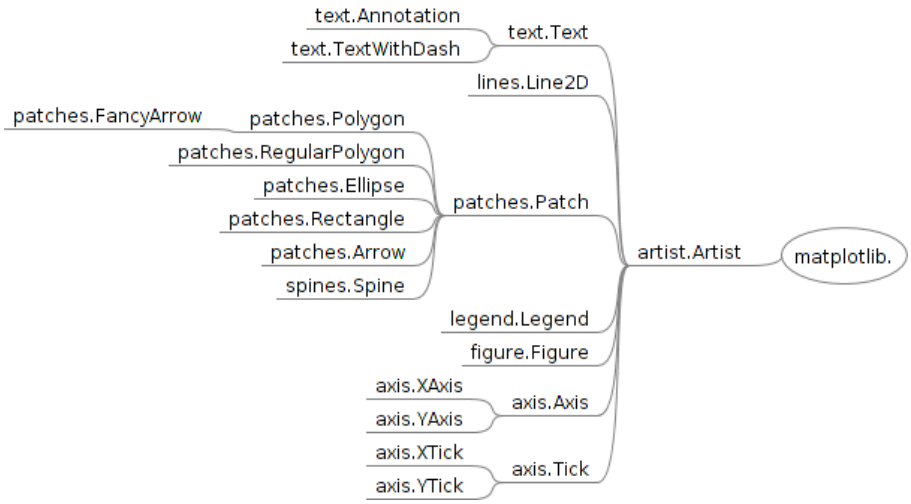
*normal oblique*
*normal italic*
normal normal

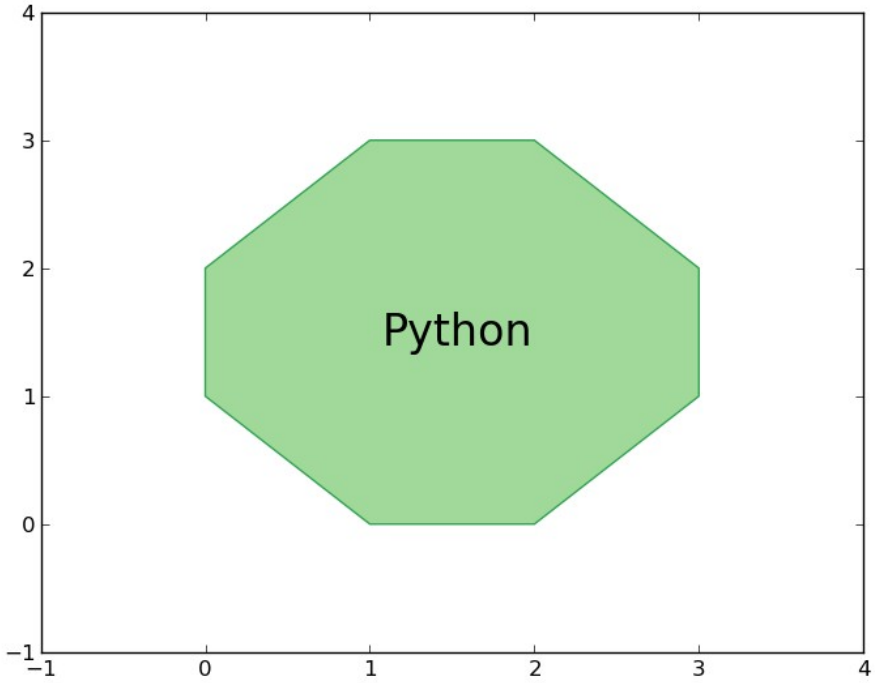
*light oblique*
*light italic*
light normal

# Rendering text with LaTeX



# Understanding the difference between pyplot and OO API

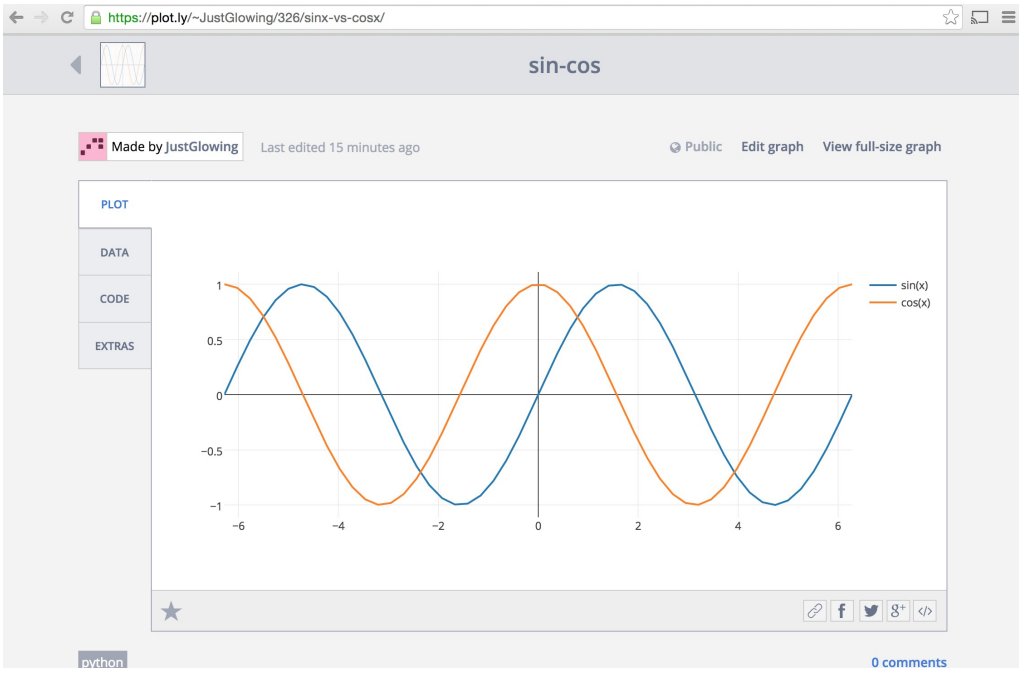


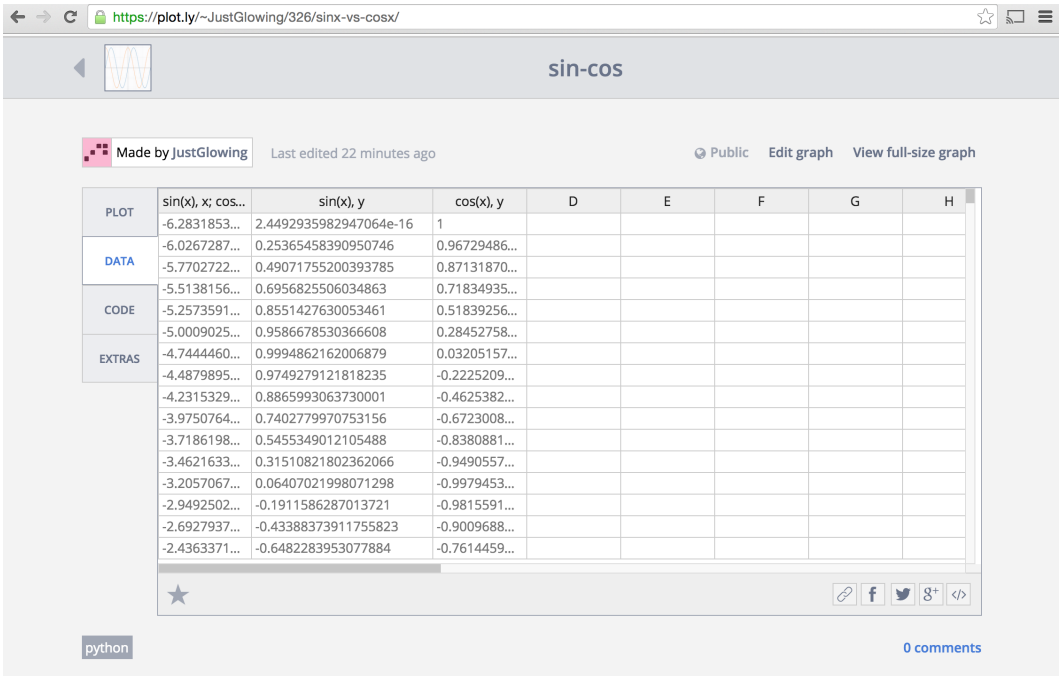

Python
4
3
2
1
0
−1
−1
0
1
2
3
4

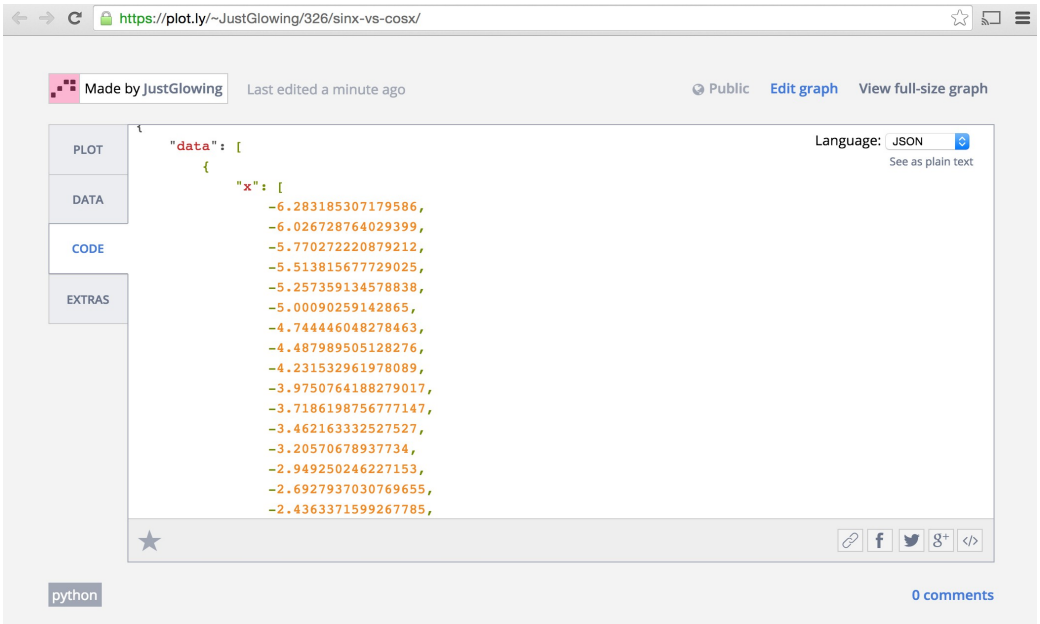
# 9

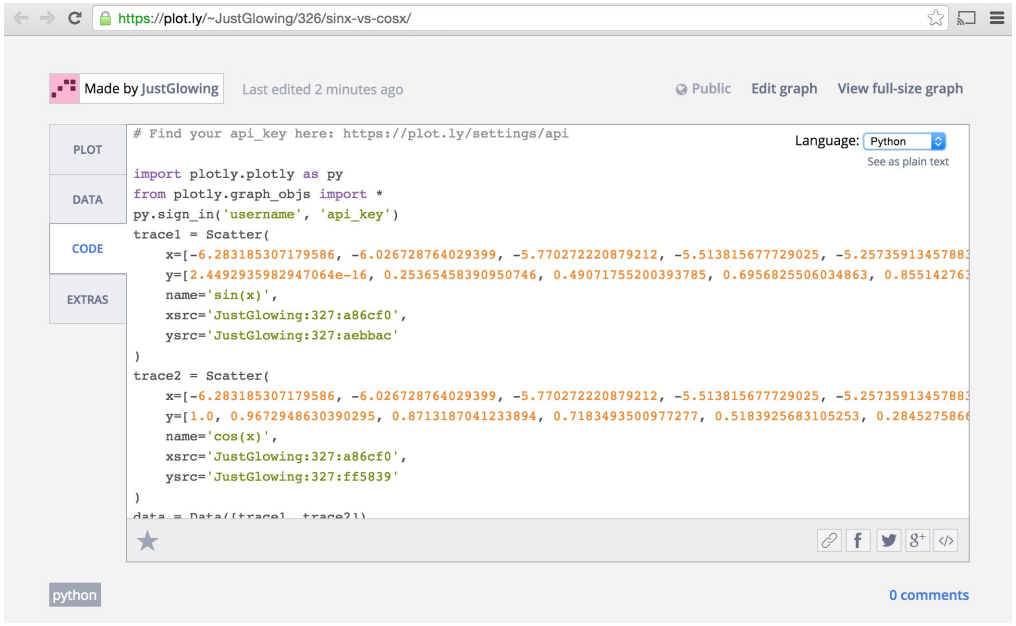
# Visualizations in the clouds with Plot.ly

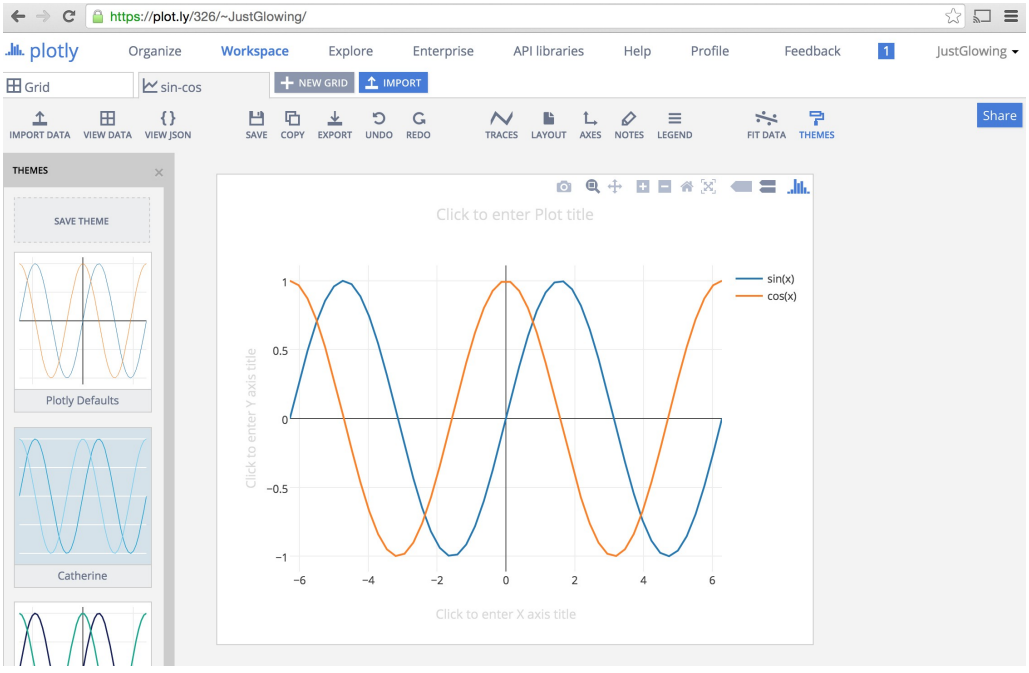
## Creating line charts



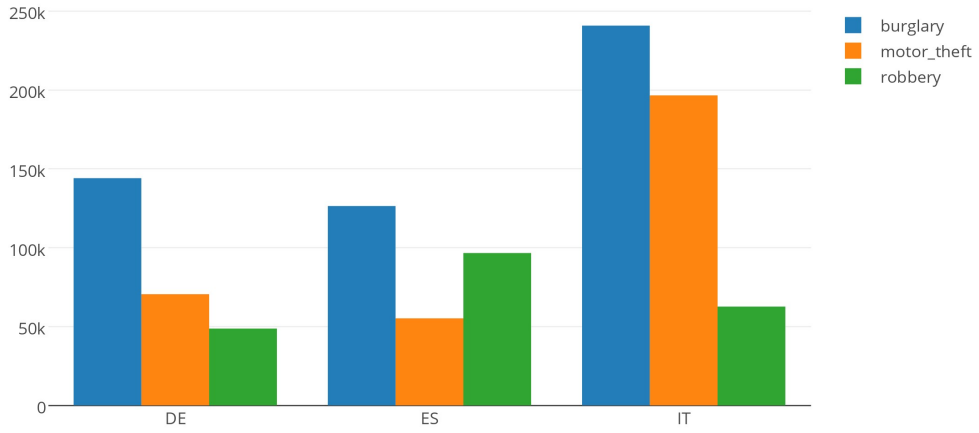

https://plot.ly/~JustGlowing/326/sinx-vs-cosx/
sin-cos
Made by JustGlowing
Last edited 22 minutes ago
Public
Edit graph
View full-size graph
PLOT
DATA
CODE
EXTRAS
sin(x), x; cos...
sin(x), y
cos(x), y
D
E
F
G
H
-6.2831853...
2.4492935982947064e-16
1
-6.0267287...
0.25365458390950746
0.96729486...
-5.7702722...
0.49071755200393785
0.87131870...
-5.5138156...
0.6956825506034863
0.71834935...
-5.2573591...
0.8551427630053461
0.51839256...
-5.0009025...
0.9586678530366608
0.28452758...
-4.7444460...
0.9994862162006879
0.03205157...
-4.4879895...
0.9749279121818235
-0.2225209...
-4.2315329...
0.8865993063730001
-0.4625382...
-3.9750764...
0.7402779970753156
-0.6723008...
-3.7186198...
0.5455349012105488
-0.8380881...
-3.4621633...
0.31510821802362066
-0.9490557...
-3.2057067...
0.06407021998071298
-0.9979453...
-2.9492502...
-0.1911586287013721
-0.9815591...
-2.6927937...
-0.43388373911755823
-0.9009688...
-2.4363371...
-0.6482283953077884
-0.7614459...
python
0 comments

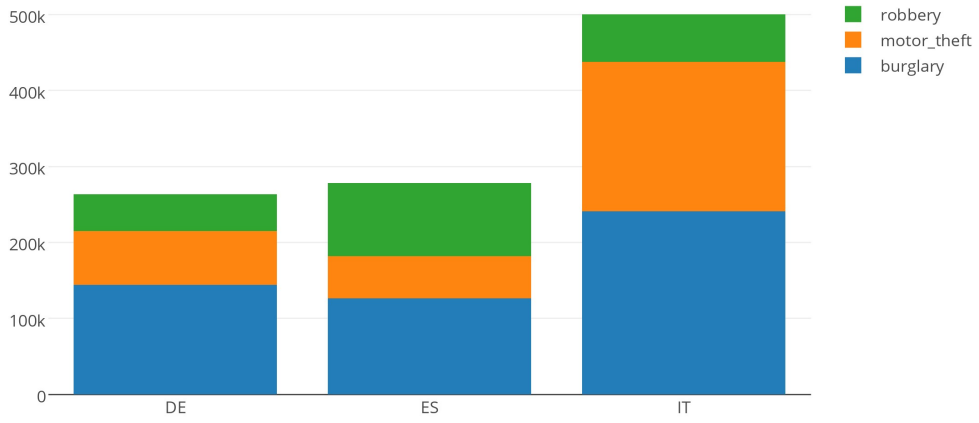

https://plot.ly/~JustGlowing/326/sinx-vs-cosx/
Made by JustGlowing
Last edited a minute ago
Public
Edit graph
View full-size graph
PLOT
DATA
CODE
EXTRAS
Language:
JSON
See as plain text
{
"data": [
{
"x": [
-6.283185307179586,
-6.026728764029399,
-5.770272220879212,
-5.513815677729025,
-5.257359134578838,
-5.00090259142865,
-4.744446048278463,
-4.487989505128276,
-4.231532961978089,
-3.9750764188279017,
-3.7186198756777147,
-3.462163332527527,
-3.20570678937734,
-2.949250246227153,
-2.6927937030769655,
-2.4363371599267785,
python
0 comments


https://plot.ly/~JustGlowing/326/sinx-vs-cosx/
Made by JustGlowing
Last edited 2 minutes ago
Public
Edit graph
View full-size graph
PLOT
DATA
CODE
EXTRAS
Language: Python
See as plain text
# Find your api_key here: https://plot.ly/settings/api
import plotly.plotly as py
from plotly.graph_objs import *
py.sign_in('username', 'api_key')
trace1 = Scatter(
x=[-6.283185307179586, -6.026728764029399, -5.770272220879212, -5.513815677729025, -5.25735913457883
y=[2.4492935982947064e-16, 0.25365458390950746, 0.49071755200393785, 0.6956825506034863, 0.855142763
name='sin(x)',
xsrc='JustGlowing:327:a86cf0',
ysrc='JustGlowing:327:aebbac'
)
trace2 = Scatter(
x=[-6.283185307179586, -6.026728764029399, -5.770272220879212, -5.513815677729025, -5.25735913457883
y=[1.0, 0.9672948630390295, 0.8713187041233894, 0.7183493500977277, 0.5183925683105253, 0.284527586
name='cos(x)',
xsrc='JustGlowing:327:a86cf0',
ysrc='JustGlowing:327:ff5839'
)
python
0 comments

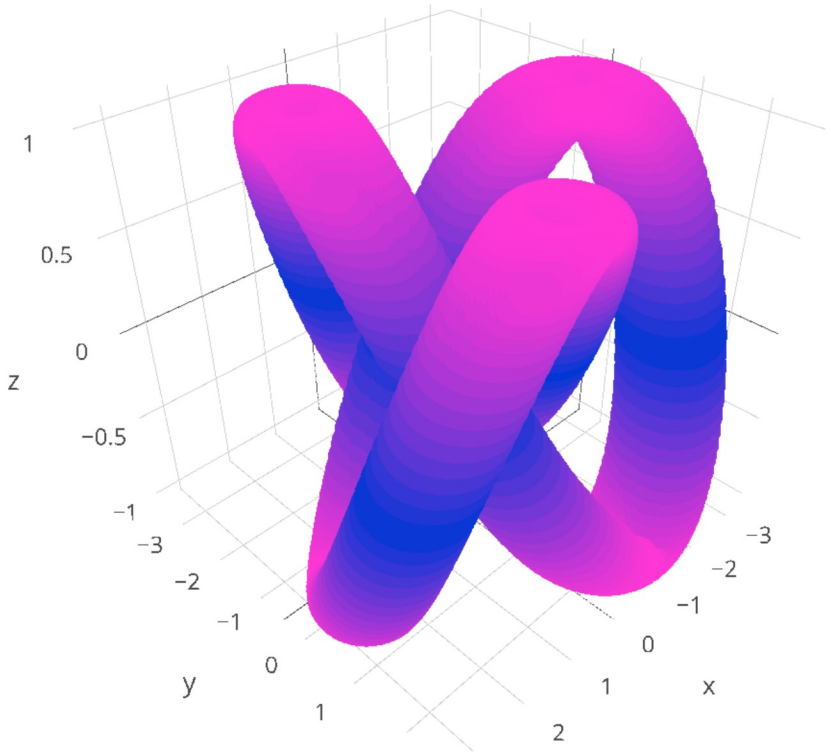

https://plot.ly/326/~JustGlowing/
plotly
Organize
Workspace
Explore
Enterprise
API libraries
Help
Profile
Feedback
JustGlowing
Grid
sin-cos
NEW GRID
IMPORT
IMPORT DATA
VIEW DATA
VIEW JSON
SAVE
COPY
EXPORT
UNDO
REDO
TRACES
LAYOUT
AXES
NOTES
LEGEND
FIT DATA
THEMES
Share
THEMES
SAVE THEME
Plotly Defaults
Catherine
Click to enter Plot title
sin(x)
cos(x)
Click to enter Y axis title
Click to enter X axis title

# Creating bar charts



# Plotting a 3D trefoil knot



# Visualizing maps and bubbles