



GETTING STARTED

1. Install

Download and uncompress the Plotly MATLAB library

2. Sign Up & Configure

www.plot.ly/python/matlab/getting-started

3. A Hello World Figure

```
[X, Y, Z] = peaks;
contour(X, Y, Z, 20);
```

```
fig2plotly()
```

`fig2plotly()` converts the MATLAB figure to a Plotly figure and opens the plot online in your plotly account!

For information on offline plotting options see:
<https://plot.ly/matlab/offline/>

BASIC CHARTS

Line Plots

```
x = linspace(-pi, pi);
y1 = sin(x);
y2 = cos(x);
fig = figure;
plot(x, y1, x, y2);
fig2plotly(fig);
```

Bubble Charts

```
x = 80 * randn(1, 30);
y = 80 * randn(size(x));
r = randi(1500, size(x));
c = randi(10, size(x));
scatter(x, y, r, c, 'filled',
'MarkerEdgeColor', 'k')
fig2plotly(gcf, 'strip', false);
```

Scatter Plots

```
x = rand(100, 1);
y = rand(100, 1);
fig = figure;
scatter(x, y);
fig2plotly(fig);
```

Heatmaps

```
z = zeros(50, 50);
for r = 1:50
    for c = 1:50
        z(r, c) = r + c;
    end
end
images(z);
fig2plotly();
```

Bar Charts

```
x = 1900:10:2000;
y = [1900:10:2000];
fig = figure;
bar(x, y);
fig2plotly(fig);
```

Area Plots

```
y = [1, 5, 3;
      3, 2, 7;
      1, 5, 3];
fig = figure;
area(y);
fig2plotly(fig);
```

LAYOUT

Legends

```
fig = figure;
plot([1 2 3], [1 2 3]);
hold on
plot([1 2 3], [3 2 1]);
legend(
    'blue trace', 'orange trace');
fig2plotly(fig);
```

Axes

Use Plotly syntax to edit plot axes and other layout options:

```
data = {...}
struct(...);
'x', [1, 2], ...
'y', [1, 2], ...
'type', 'scatter' ) ...
};
```

```
layout = struct(
    'xaxis', struct('autorange',
    'reversed' ));
mirror = 'all'
response = plotly(
    data, struct(
        'layout', layout,
        'filename', 'axes',
        'fileopt', 'overwrite' ) );
```

```
plot_url = response.url
```

STATISTICAL CHARTS

Histograms

```
x = randn(10000, 1);
fig = figure;
histogram(x);
fig2plotly(fig)
```

Box Plots

```
x = normrnd(5, 1, 100, 1);
fig = figure;
boxplot(x);
fig2plotly(fig)
```

2D Histogram

```
x = randn(500, 1);
y = randn(500, 1) + 1;
data = { struct('x', 'x', 'y', 'y', 'type',
'histogram2d')};
response = plotly(data);
plot_url = response.url
```

MAPS

Bubble Map

```
data = { struct('lat', [40.73, 34.05], ...
'Ion', [-73.9, -1.2], ...
'locationmode', 'USA-states', ...
'type', 'scattergeo', ...
'marker', struct('size', 10))};
layout = struct('geo', struct('scope', ...
'usa'));
plotly(data, struct('layout', layout))
```

Choropleth Map

```
data = { struct('z', [1, 2], ...
'locationmode', 'USA-states', ...
'type', 'choropleth')};
layout = struct('geo', struct('scope', ...
'usa'));
plotly(data, struct('layout', layout))
```

Scatter Map

```
data = { struct('lat', [40.73, 34.05], ...
'Ion', [-73.9, -1.2], ...
'locationmode', 'USA-states', ...
'type', 'scattergeo', ...
'marker', struct('size', 10))};
layout = struct('geo', struct('scope', ...
'usa'));
plotly(data, struct('layout', layout))
```

3D CHARTS

3D Surface Plots

```
[X, Y, Z] = peaks(25);
fig = figure;
surf(X, Y, Z);
fig2plotly(fig)
```

3D Line Plots

```
t = 0 : pi / 50 : 10 * pi;
x = sin(t);
y = cos(t);
fig = figure;
plot3(x, y, t);
fig2plotly(fig)
```

3D Scatter Plots

```
[x, y, z] = sin(16);
fig = figure;
scatter3(x, y, z);
fig2plotly(fig)
```

FIGURE HIERARCHY

Figure {}

DATA []
TRACE {}
x, y, z []
color, text, size []
colorscale ABC or []
MARKER {}
color ABC
symbol ABC
LINE {}
color ABC
width 123

LAYOUT {}
title ABC
XAXIS, YAXIS {}
SCENE {}
XAXIS, YAXIS, ZAXIS {}
GEO {}
LEGEND {}
ANNOTATIONS {}

{ } = dictionary
[] = list
ABC = string
123 = number