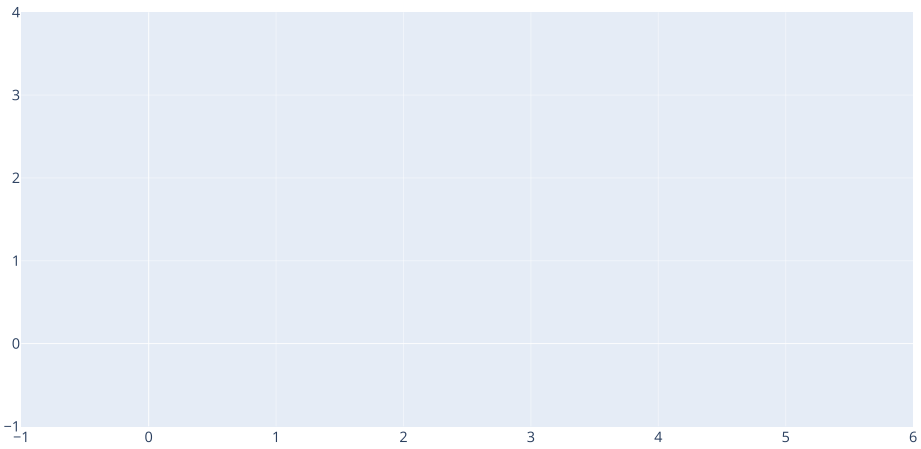
Carpet Plots in Python

How to make carpet plots in Python with Plotly.

Set X and Y Coordinates

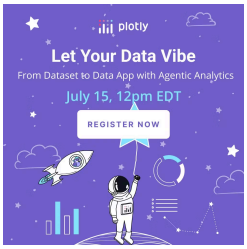
To set the `x` and `y` coordinates use `x` and `y` attributes. If `x` coordinate values are omitted a cheater plot will be created. The plot below has a `y` array specified but requires `a` and `b` parameter values before an axis may be plotted.

```
import plotly.graph_objects as go

fig = go.Figure(go.Carpet(
    y = [2, 3.5, 4, 3, 4.5, 5, 5.5, 6.5, 7.5, 8, 8.5, 10]
))

fig.show()
```

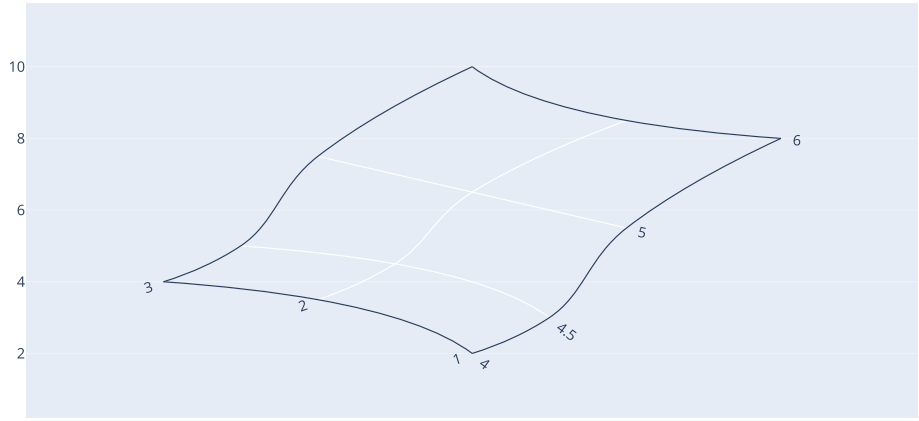


Add Parameter Values

To save parameter values use the `a` and `b` attributes.



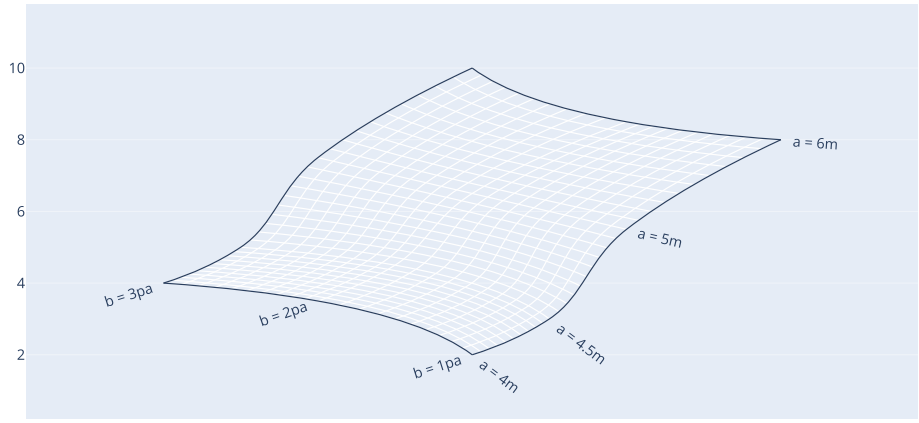
```
import plotly.graph_objects as go

fig = go.Figure(go.Carpet(
    a = [4, 4, 4, 4.5, 4.5, 4.5, 5, 5, 5, 6, 6, 6],
    b = [1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3],
    y = [2, 3.5, 4, 3, 4.5, 5, 5.5, 6.5, 7.5, 8, 8.5, 10]
))

fig.show()
```



Add A and B axis

Use aaxis or baxis list to make changes to the axes. For a more detailed list of attributes refer to R reference (https://plotly.com/r/reference/carpet/#carpet-aaxis).

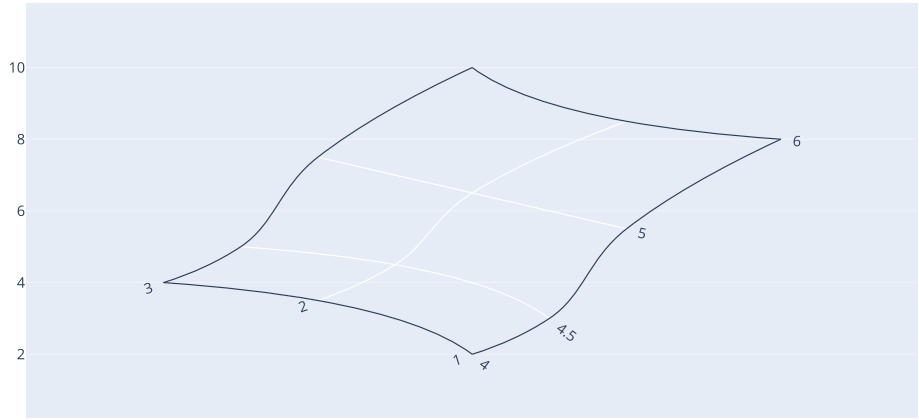
```
import plotly.graph_objects as go

fig = go.Figure(go.Carpet(
    a = [4, 4, 4, 4.5, 4.5, 4.5, 5, 5, 5, 6, 6, 6],
    b = [1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3],
    y = [2, 3.5, 4, 3, 4.5, 5, 5.5, 6.5, 7.5, 8, 8.5, 10],
    aaxis = dict(
        tickprefix = 'a = ',
        ticksuffix = 'm',
        smoothing = 1,
        minorgridcount = 9,
    ),
    baxis = dict(
        tickprefix = 'b = ',
        ticksuffix = 'pa',
        smoothing = 1,
        minorgridcount = 9,
    )
))

fig.show()
```



Alternate input format

The data arrays x, y may either be specified as one-dimensional arrays of data or as arrays of arrays. If one-dimensional, then x, y, a, and b should all be the same length. If x and y are arrays of arrays, then the length of a should match the inner dimension and the length of b the outer dimension. The plot below represents the same plot as those above.

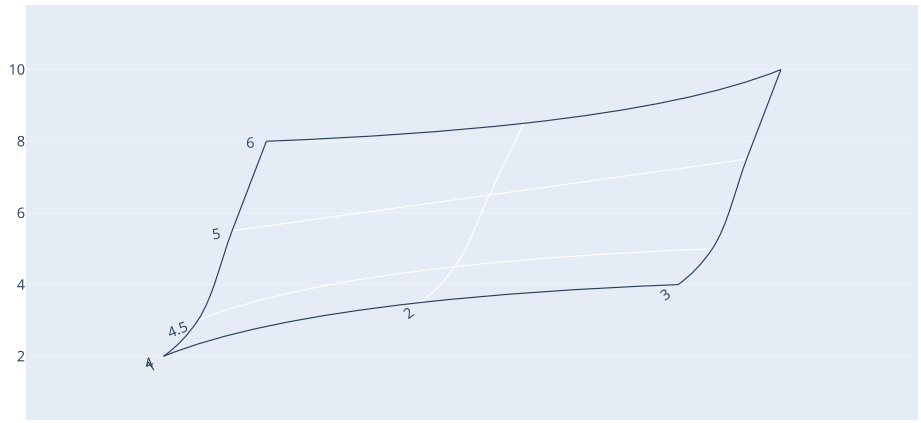
```
import plotly.graph_objects as go

fig = go.Figure(go.Carpet(
    a = [4, 4.5, 5, 6],
    b = [1, 2, 3],
    y = [[2, 3, 5.5, 8],
         [3.5, 4.5, 6.5, 8.5],
         [4, 5, 7.5, 10]]
))

fig.show()
```

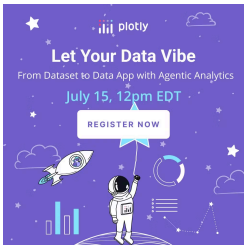


Cheater plot layout

The layout of cheater plots is not unique and depends upon the cheaterslope and axis cheatertype parameters. If x is not specified, each row of the x array is constructed based on the the formula a + cheaterslope * b, where a and b are either the value or the integer index of a and b respectively, depending on the corresponding axis cheatertype. Although the layout of the axis below is different than the plots above, it represents the same data as the axes above.

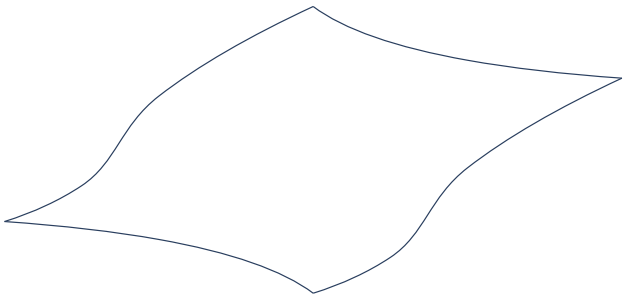
```
import plotly.graph_objects as go

fig = go.Figure(go.Carpet(
    a = [4, 4.5, 5, 6],
    b = [1, 2, 3],
    y = [[2, 3, 5.5, 8],
         [3.5, 4.5, 6.5, 8.5],
         [4, 5, 7.5, 10]],
    cheaterslope = -5,
    aaxis = dict(cheatertype = 'index'),
    baxis = dict(cheatertype = 'value')
))

fig.show()
```





Style A and B axis

```
import plotly.graph_objects as go

fig = go.Figure(go.Carpet(
    a = [4, 4, 4, 4.5, 4.5, 4.5, 5, 5, 5, 6, 6, 6],
    b = [1, 2, 3, 1, 2, 3, 1, 2, 3, 1, 2, 3],
    y = [2, 3.5, 4, 3, 4.5, 5, 5.5, 6.5, 7.5, 8, 8.5, 10],
    aaxis = dict(
        tickprefix = 'a = ',
        ticksuffix = 'm',
        smoothing = 1,
        minorgridcount = 9,
        minorgridwidth = 0.6,
        minorgridcolor = 'white',
        gridcolor = 'white',
        color = 'white'
    ),
    baxis = dict(
        ticksuffix = 'Pa',
        smoothing = 1,
        minorgridcount = 9,
        minorgridwidth = 0.6,
        gridcolor = 'white',
        minorgridcolor = 'white',
        color = 'white'
    )
))

fig.update_layout(
    plot_bgcolor = 'black',
    paper_bgcolor = 'black',
    xaxis = dict(
        showgrid = False,
        showticklabels = False
    ),
    yaxis = dict(
        showgrid = False,
        showticklabels = False
    )
)

fig.show()
```



Add Points and Contours

To add points and lines see Carpet Scatter Plots (https://plotly.com/python/carpet-scatter) or to add contours see Carpet Contour Plots (https://plotly.com/python/carpet-contour)

Reference

See https://plotly.com/python/reference/carpet/ (https://plotly.com/python/reference/carpet/) for more information and chart attribute options!

What About Dash?

Dash (https://dash.plot.ly/) is an open-source framework for building analytical applications, with no Javascript required, and it is tightly integrated with the Plotly graphing library.

Learn about how to install Dash at https://dash.plot.ly/installation (https://dash.plot.ly/installation).

Everywhere in this page that you see fig.show(), you can display the same figure in a Dash application by passing it to the figure argument of the Graph component (https://dash.plot.ly/dash-core-components/graph) from the built-in dash_core_components package like this:

```
import plotly.graph_objects as go # or plotly.express as px
fig = go.Figure() # or any Plotly Express function e.g. px.bar(...)
# fig.add_trace( ... )
# fig.update_layout( ... )

from dash import Dash, dcc, html

app = Dash()
app.layout = html.Div([
    dcc.Graph(figure=fig)
])

app.run(debug=True, use_reloader=False)  # Turn off reloader if inside Jupyter
```