

Error Bars in Python

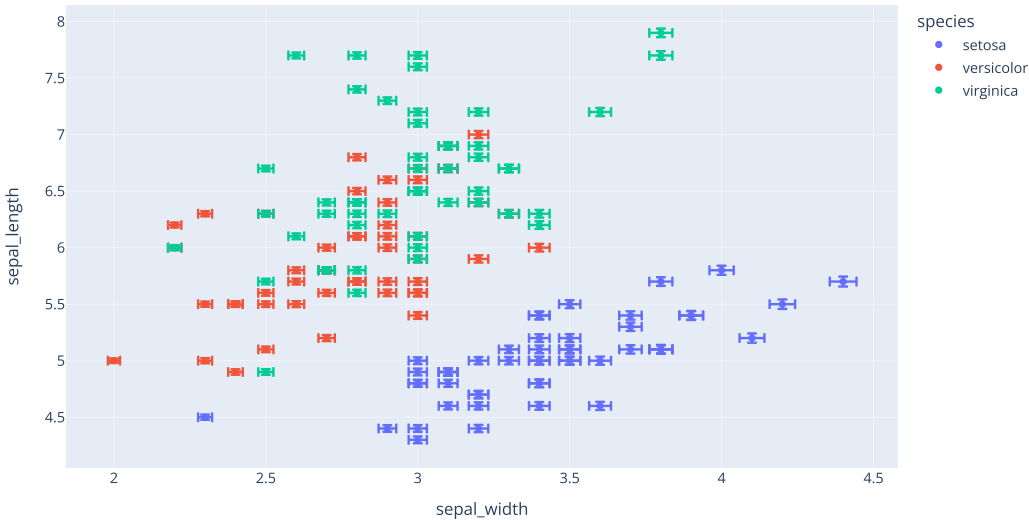
How to add error-bars to charts in Python with Plotly.

Plotly Studio: Transform any dataset into an interactive data application in minutes with AI. [Sign up for early access now.](https://plotly.com/studio/?utm_medium=graphing-libraries&utm_campaign=studio_early_access&utm_content=sidebar) (https://plotly.com/studio/?utm_medium=graphing-libraries&utm_campaign=studio_early_access&utm_content=sidebar)

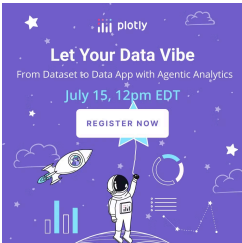
Error Bars with Plotly Express

Plotly Express (/python/plotly-express/) is the easy-to-use, high-level interface to Plotly, which operates on a variety of types of data (/python/px-arguments/) and produces easy-to-style figures (/python/styling-plotly-express/). For functions representing 2D data points such as px.scatter (https://plotly.com/python/line-and-scatter/), px.line (https://plotly.com/python/line-charts/), px.bar (https://plotly.com/python/bar-charts/) etc., error bars are given as a column name which is the value of the error_x (for the error on x position) and error_y (for the error on y position).

```
import plotly.express as px
df = px.data.iris()
df["e"] = df["sepal_width"]/100
fig = px.scatter(df, x="sepal_width", y="sepal_length", color="species",
                error_x="e", error_y="e")
fig.show()
```



Asymmetric Error Bars with Plotly Express



```
import plotly.express as px
df = px.data.iris()
df["e_plus"] = df["sepal_width"]/100
df["e_minus"] = df["sepal_width"]/40
fig = px.scatter(df, x="sepal_width", y="sepal_length", color="species",
                error_y="e_plus", error_y_minus="e_minus")
fig.show()
```

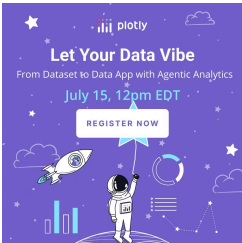
xpress

value
ant



Error Bars with graph_objects

Basic Symmetric Error Bars



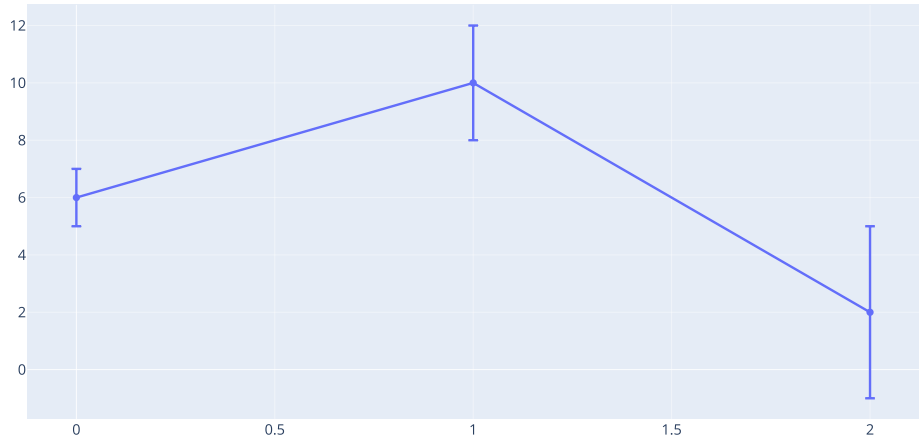
xpress

```
import plotly.graph_objects as go

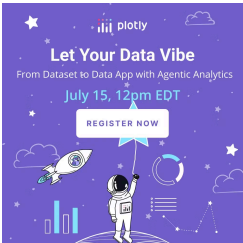
fig = go.Figure(data=go.Scatter(
    x=[0, 1, 2],
    y=[6, 10, 2],
    error_y=dict(
        type='data', # value of error bar given in data coordinates
        array=[1, 2, 3],
        visible=True)
    ))

fig.show()
```

value
ant



Asymmetric Error Bars

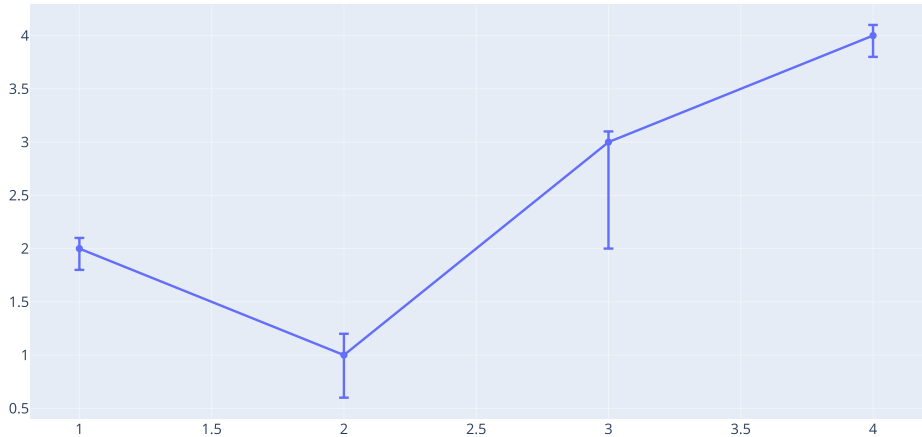


xpress

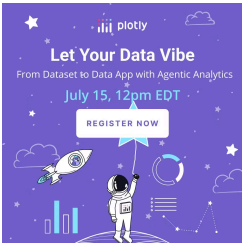
```
import plotly.graph_objects as go

fig = go.Figure(data=go.Scatter(
    x=[1, 2, 3, 4],
    y=[2, 1, 3, 4],
    error_y=dict(
        type='data',
        symmetric=False,
        array=[0.1, 0.2, 0.1, 0.1],
        arrayminus=[0.2, 0.4, 1, 0.2])
    ))
fig.show()
```

value
ant



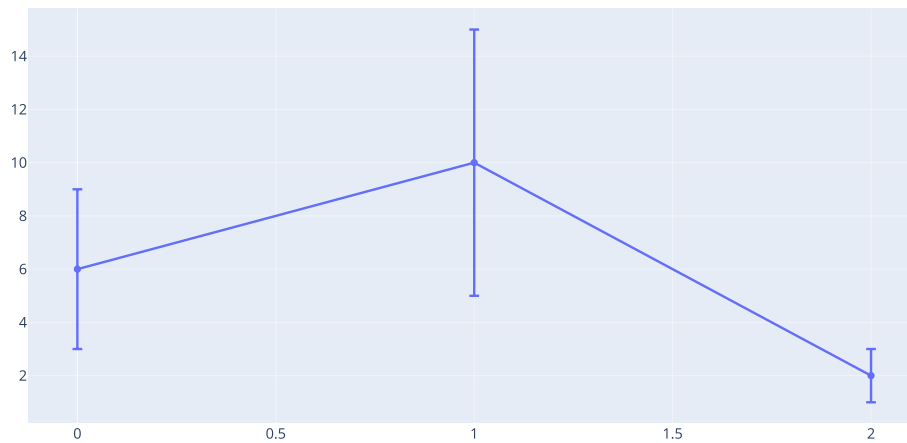
Error Bars as a Percentage of the y Value



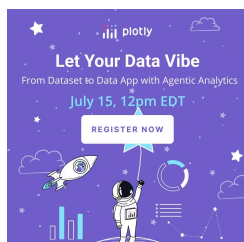
```
import plotly.graph_objects as go

fig = go.Figure(data=go.Scatter(
    x=[0, 1, 2],
    y=[6, 10, 2],
    error_y=dict(
        type='percent', # value of error bar given as percentage of y value
        value=50,
        visible=True)
    ))
fig.show()
```

xpress

value
ant

Asymmetric Error Bars with a Constant Offset

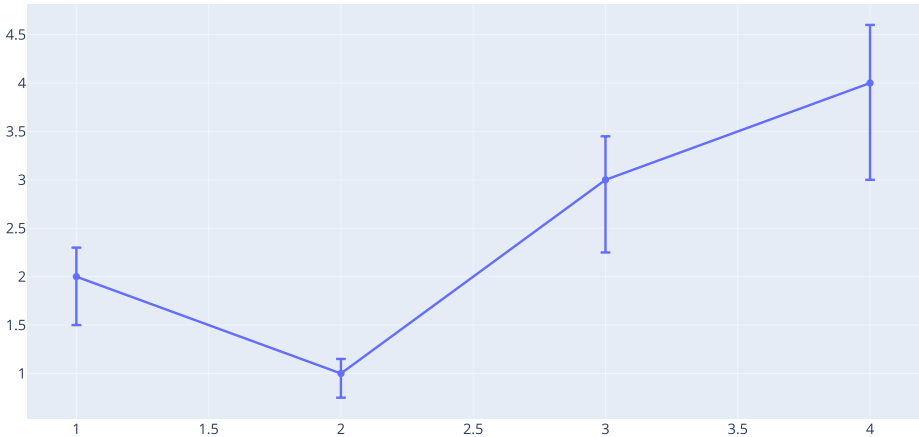


xpress

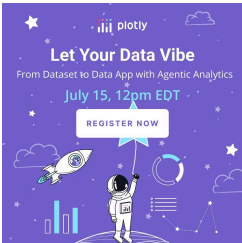
```
import plotly.graph_objects as go

fig = go.Figure(data=go.Scatter(
    x=[1, 2, 3, 4],
    y=[2, 1, 3, 4],
    error_y=dict(
        type='percent',
        symmetric=False,
        value=15,
        valueminus=25)
    ))
fig.show()
```

value
ant



Horizontal Error Bars

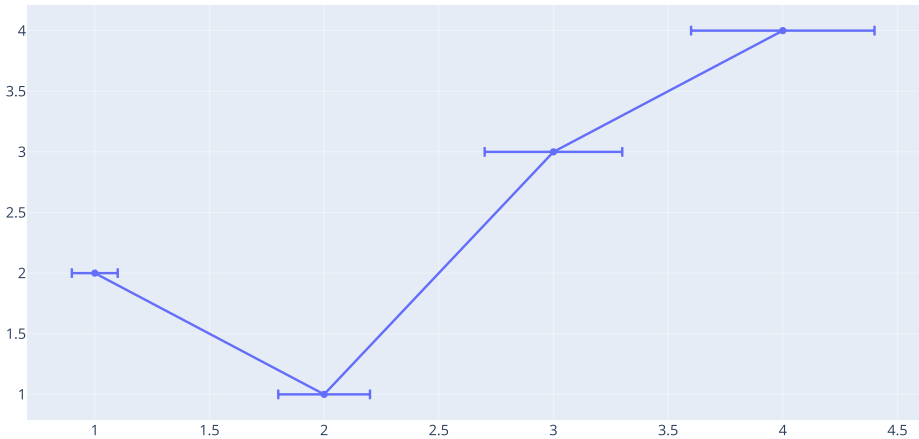


xpress

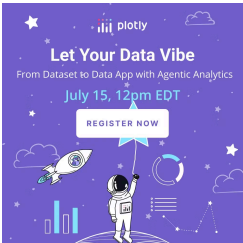
```
import plotly.graph_objects as go

fig = go.Figure(data=go.Scatter(
    x=[1, 2, 3, 4],
    y=[2, 1, 3, 4],
    error_x=dict(
        type='percent',
        value=10)
))
fig.show()
```

value
ant



Bar Chart with Error Bars

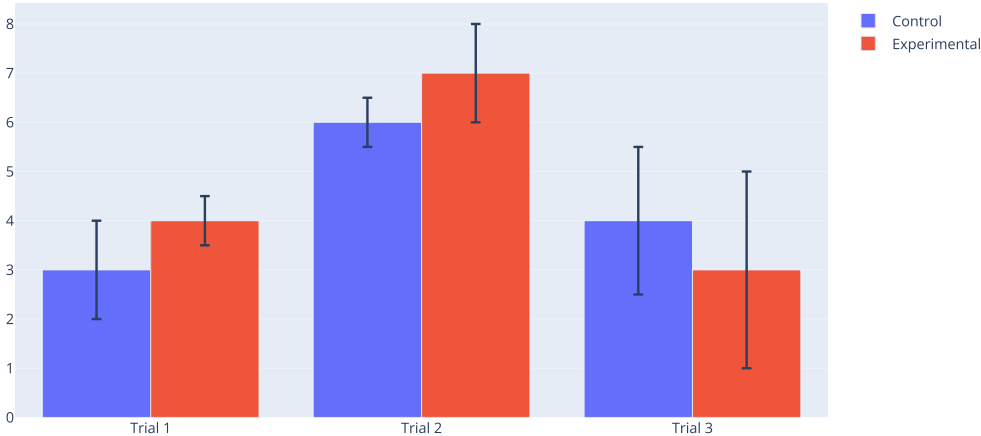


xpress

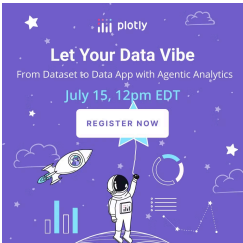
value
ant

```
import plotly.graph_objects as go

fig = go.Figure()
fig.add_trace(go.Bar(
    name='Control',
    x=['Trial 1', 'Trial 2', 'Trial 3'], y=[3, 6, 4],
    error_y=dict(type='data', array=[1, 0.5, 1.5])
))
fig.add_trace(go.Bar(
    name='Experimental',
    x=['Trial 1', 'Trial 2', 'Trial 3'], y=[4, 7, 3],
    error_y=dict(type='data', array=[0.5, 1, 2])
))
fig.update_layout(barmode='group')
fig.show()
```



Colored and Styled Error Bars



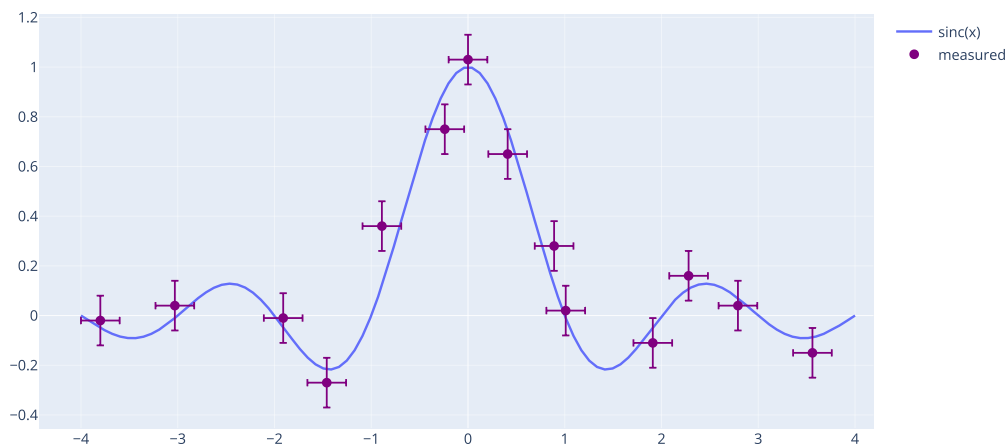

```

import plotly.graph_objects as go
import numpy as np

x_theo = np.linspace(-4, 4, 100)
sincx = np.sinc(x_theo)
x = [-3.8, -3.03, -1.91, -1.46, -0.89, -0.24, -0.0, 0.41, 0.89, 1.01, 1.91, 2.28, 2.79, 3.56]
y = [-0.02, 0.04, -0.01, -0.27, 0.36, 0.75, 1.03, 0.65, 0.28, 0.02, -0.11, 0.16, 0.04, -0.15]

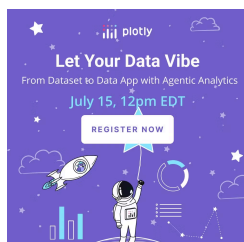
fig = go.Figure()
fig.add_trace(go.Scatter(
    x=x_theo, y=sincx,
    name='sinc(x)'
))
fig.add_trace(go.Scatter(
    x=x, y=y,
    mode='markers',
    name='measured',
    error_y=dict(
        type='constant',
        value=0.1,
        color='purple',
        thickness=1.5,
        width=3,
    ),
    error_x=dict(
        type='constant',
        value=0.2,
        color='purple',
        thickness=1.5,
        width=3,
    ),
    marker=dict(color='purple', size=8)
))
fig.show()

```



Reference

See <https://plotly.com/python/reference/scatter/> (<https://plotly.com/python/reference/scatter/>) 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:

xpress

value


ant

```
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
```



Dash your way to interactive web apps.

No JavaScript required!

GET STARTED NOW


My First App with Data, Graph, and Controls

pop

lifeExp

gdpPerCap

country	pop	continent	lifeExp	gdpPerCap
Afghanistan	31889923	Asia	43.828	974.5883384
Albania	3600523	Europe	76.423	5937.829525999999
Algeria	33333216	Africa	72.381	6223.367465
Angola	12420476	Africa	42.731	4707.231267
Argentina	40301927	Americas	75.32	12779.37964
Australia	20434176	Oceania	81.235	34435.367439999995
Austria	8199783	Europe	79.829	36326.4927
Bahrain	706573	Asia	75.635	29796.04834
Bangladesh	150448339	Asia	64.062	1701.253792
Belgium	10391226	Europe	79.441	33062.04908
Benin	8878314	Africa	56.728	1441.284873
Bolivia	9119152	Americas	65.554	3821.137884



(https://dash.plotly.com/tutorial?utm_medium=graphing_libraries&utm_content=python_footer)

JOIN OUR MAILING LIST

Sign up to stay in the loop with all things Plotly — from Dash Club to product updates, webinars, and more!

SUBSCRIBE
(<https://go.plot.ly/subscription>)

About Us

Careers (<https://plotly.com/careers>)
Resources (<https://plotly.com/resources/>)
Blog (<https://medium.com/@plotlygraphs>)

Products

Dash (<https://plotly.com/dash/>)
Consulting and Training
(<https://plotly.com/consulting-and-oem/>)

Support

Community Support (<https://community.plot.ly/>)
Documentation (<https://plotly.com/graphing-libraries>)

Pricing

Enterprise Pricing (<https://plotly.com/get-pricing/>)

