



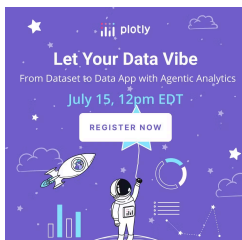
# Clustergram in Python

lths

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)

## Default Clustergram

A clustergram is a combination heatmap-dendrogram that is commonly used in gene expression data. The hierarchical clustering that is represented by the dendrograms can be used to identify groups of genes with related expression levels. The Dash Bio Clustergram component is a Python-based component that uses plotly.py to generate a figure. It takes as input a two-dimensional numpy array of floating-point values. Imputation of missing data and computation of hierarchical clustering both occur within the component itself. Clusters that meet or exceed a user-defined threshold of similarity comprise single traces in the corresponding dendrogram, and can be highlighted with annotations. The user can specify additional parameters to customize the metrics and methods used to compute parts of the clustering, such as the pairwise distance between observations and the linkage matrix.

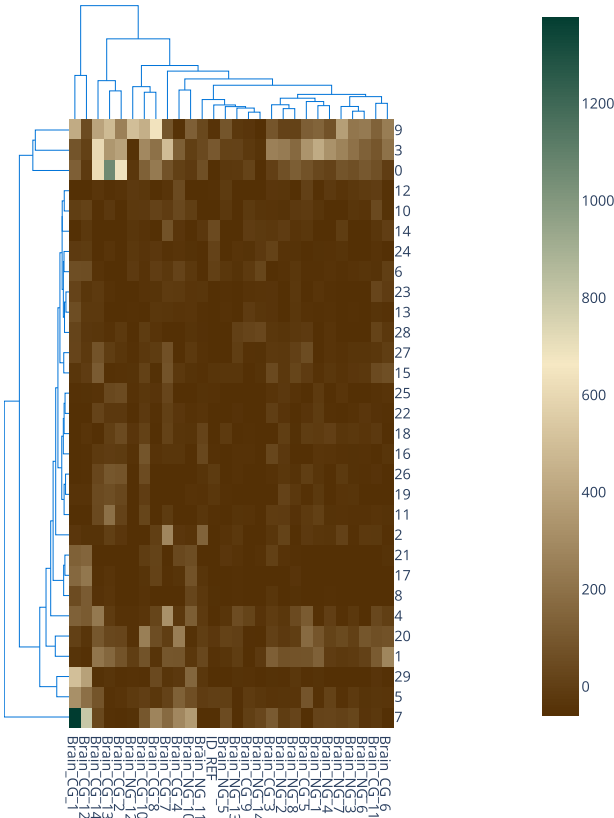


lths

```
import pandas as pd
import dash_bio

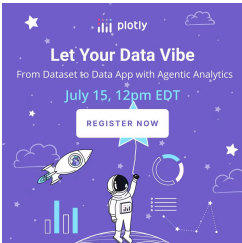
df = pd.read_csv('https://raw.githubusercontent.com/plotly/datasets/master/Dash_Bio/Chromosomal/clustergram_brain_cancer.csv')

dash_bio.Clustergram(
    data=df,
    column_labels=list(df.columns.values),
    row_labels=list(df.index),
    height=800,
    width=700
)
```



# Dendrogram Cluster Colors/Line Widths

Change the colors of the dendrogram traces that are used to represent clusters, and configure their line widths.

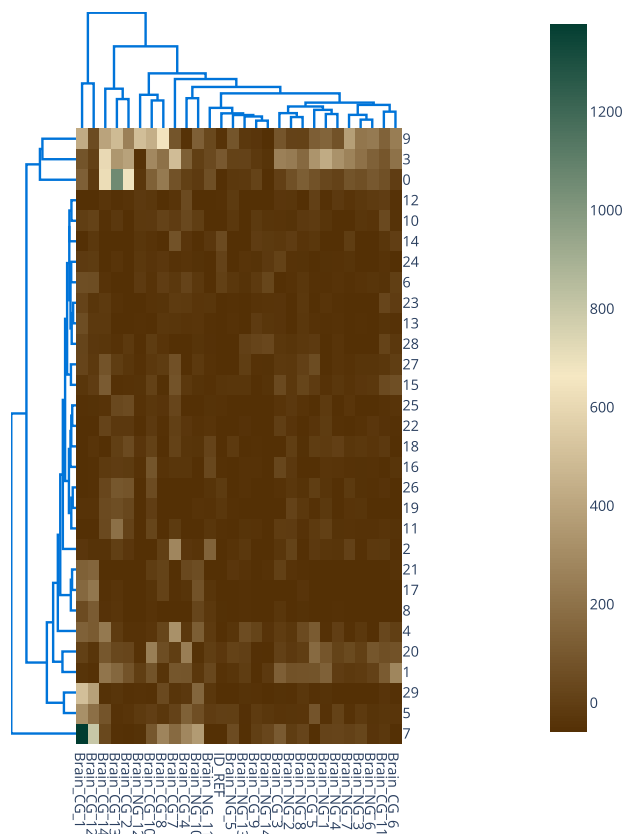


```
import pandas as pd
import dash_bio

df = pd.read_csv('https://raw.githubusercontent.com/plotly/datasets/master/Dash_Bio/Chromosomal/clustergram_brain_cancer.csv')

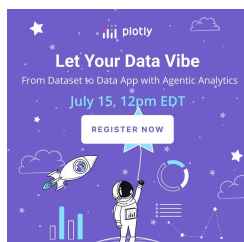
dash_bio.Clustergram(
    data=df,
    column_labels=list(df.columns.values),
    row_labels=list(df.index),
    height=800,
    width=700,
    color_list={
        'row': ['#636EFA', '#00CC96', '#19D3F3'],
        'col': ['#AB63FA', '#EF553B'],
        'bg': '#506784'
    },
    line_width=2
)
```

lths



## Relative Dendrogram Size

Change the relative width and height of, respectively, the row and column dendrograms compared to the width and height of the heatmap.

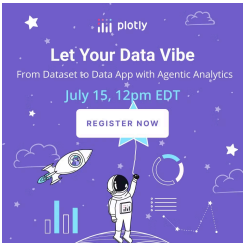
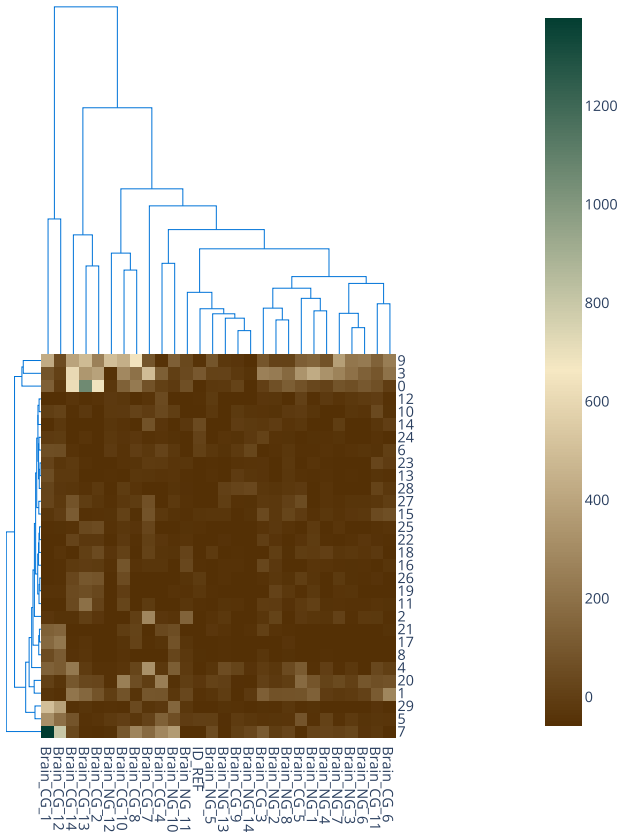


lths

```
import pandas as pd
import dash_bio

df = pd.read_csv('https://raw.githubusercontent.com/plotly/datasets/master/Dash_Bio/Chromosomal/clustergram_brain_cancer.csv')

dash_bio.Clustergram(
    data=df,
    column_labels=list(df.columns.values),
    row_labels=list(df.index),
    height=800,
    width=700,
    display_ratio=[0.1, 0.7]
)
```



# Clustergram with Dash

lths

```
from dash import Dash, dcc, html, Input, Output
import dash_bio as dashbio
import pandas as pd

app = Dash(__name__)

df = pd.read_csv('https://git.io/clustergram_brain_cancer.csv').set_index('ID_REF')

columns = list(df.columns.values)
rows = list(df.index)

app.layout = html.Div([
    "Rows to display",
    dcc.Dropdown(
        id='dropdown',
        options=[
            {'label': row, 'value': row} for row in list(df.index)
        ],
        value=rows[:10],
        multi=True
    ),
    html.Div(id='graph')
])
```

DOWNLOAD

Rows to display

×

1005\_at

×

1039\_s\_at

×

103\_at

×

1052\_s\_at

×

1058\_at

×

1063\_s\_at

×

1072\_g\_at

×

1091\_at

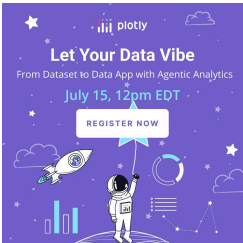
×

1101\_at

×

1104\_s\_at

×



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:


lths

```
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	9139352	Americas	65.554	3821.137884



([https://dash.plotly.com/tutorial?utm\\_medium=graphing\\_libraries&utm\\_content=python\\_footer](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/>)

