

Práctica parte 1

12 de marzo de 2021

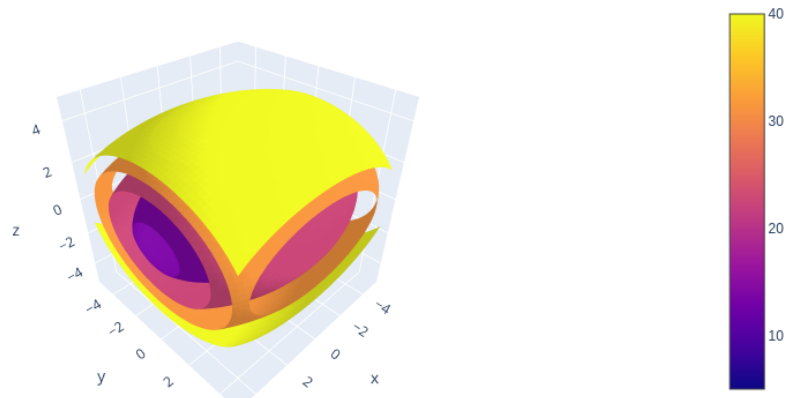
```
[1]: import plotly.graph_objects as go
import numpy as np
import math
```

```
[2]: # se puede crear una rejilla de dos formas diferentes
#X, Y, Z = np.mgrid[-5:5:40j, -5:5:40j, -5:5:40j]
X, Y, Z = np.meshgrid(np.arange(-5,5,.25),
                      np.arange(-5,5,.25),
                      np.arange(-5,5,.25))

# un valor para cada punto en ese espacio
values = X * X * 0.5 + Y * Y + Z * Z * 2

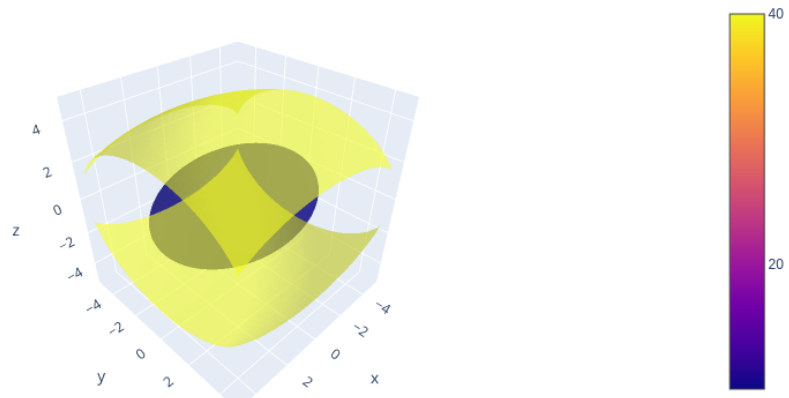
# se crean isosuperficies
data=go.Isosurface(
    x=X.flatten(),
    y=Y.flatten(),
    z=Z.flatten(),
    value=values.flatten(),
    isomin=5,
    isomax=40,
    surface_count=5, # number of isosurfaces, 2 by default: only min and max
    colorbar_nticks=5, # colorbar ticks correspond to isosurface values
    caps=dict(x_show=False, y_show=False)
)

fig = go.Figure(data)
fig.show()
```



```
[3]: # se crean isosuperficies
data=go.Isosurface(
    x=X.flatten(),
    y=Y.flatten(),
    z=Z.flatten(),
    value=values.flatten(),
    opacity=.6,
    isomin=10,
    isomax=40,
    surface_count=2, # number of isosurfaces, 2 by default: only min and max
    colorbar_nticks=2, # colorbar ticks correspond to isosurface values
    caps=dict(x_show=False, y_show=False)
)

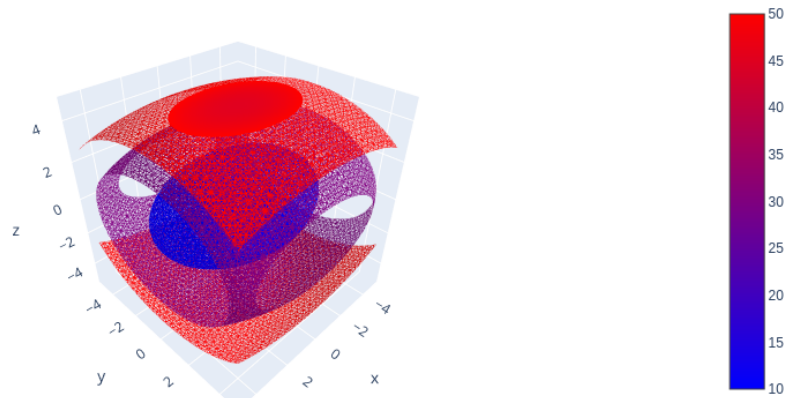
fig = go.Figure(data)
fig.show()
```



```
[4]: # se pueden cambiar las escalas de colores
fig = go.Figure(data=go.Isosurface(
    x=X.flatten(),
    y=Y.flatten(),
    z=Z.flatten(),
    value=values.flatten(),
    colorscale='BlueRed',
    isomin=10,
    isomax=50,
    surface_count=3,
    caps=dict(x_show=False, y_show=False)
))
fig.show()
```



```
[5]: # se pueden cambiar las escalas de colores y mostrar las mallas
fig = go.Figure(data=go.Isosurface(
    x=X.flatten(),
    y=Y.flatten(),
    z=Z.flatten(),
    value=values.flatten(),
    colorscale='BlueRed',
    isomin=10,
    isomax=50,
    surface_count=3,
    surface=dict(show=True,count=1, fill=0.6),
    caps=dict(x_show=False, y_show=False)
))
fig.show()
```

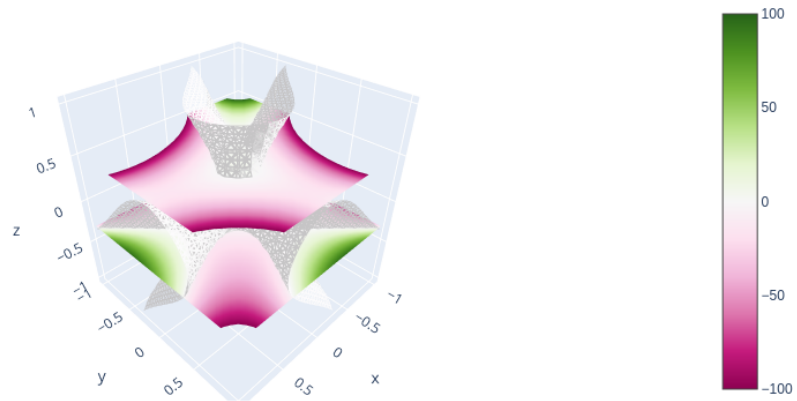


```
[6]: # otra forma de evaluar los valores del campo escalar y planos intermedios
f = lambda x, y, z: 81*(x**3 + y**3 + z**3) - 189*(x**2*y + x**2*z + y**2*x
→ +y**2*z + z**2*x + z**2*y) + \
                    54*(x*y*z) + 126*(x*y + x*z + y*z) - 9*(x**2 + y**2 + z**2)
→ - 9*(x + y + z) + 1

a = 1
X, Y, Z = np.mgrid[-a:a:25j, -a:a:25j, -a:a:25j]

data = [go.Isosurface(
    x=X.flatten(),
    y=Y.flatten(),
    z=Z.flatten(),
    value=f(X, Y, Z).flatten(),
    isomin=-100,
    isomax=100,
    surface=dict(show=True, count=1, fill=0.8),
    slices=go.isosurface.Slices(
        z=go.isosurface.slices.Z(
            show = True,
            locations=[-0.3, 0.5])
    ),
    caps=dict(x_show=False, y_show=False, z_show=False)
)]

fig = go.Figure(data)
fig.show()
```



1. Práctica viernes 12

El objetivo de la práctica es visualizar dos campos escalares usando isosuperficies. El valor en cada punto estará dado por:

■

$$v_1 = \frac{\sin((x^2 + y^2 + z^2)^{1/2})}{(x^2 + y^2 + z^2)^{1/2}} \quad (1)$$

■

$$v_2 = x e^{(-x^2 - y^2 - z^2)} \quad (2)$$

A continuación, algunos ejemplos de el contenido de los campos escalares

