

## 1. Importación de librerías

python

CopyEdit

```
import numpy as np
```

```
import matplotlib.pyplot as plt
```

```
from mpl_toolkits.mplot3d import Axes3D
```

```
from matplotlib.animation import FuncAnimation
```

- `numpy (np)`: Se usa para trabajar con arreglos numéricos y funciones matemáticas.
  - `matplotlib.pyplot (plt)`: Se usa para graficar datos.
  - `mpl_toolkits.mplot3d.Axes3D`: Permite crear gráficos en 3D con matplotlib.
  - `FuncAnimation`: Permite crear animaciones en matplotlib.
- 

## 2. Definición de los datos de la superficie

python

CopyEdit

```
x = np.linspace(-2, 2, 50)
```

```
y = np.linspace(-2, 2, 50)
```

```
X, Y = np.meshgrid(x, y)
```

- `np.linspace(-2, 2, 50)`: Genera 50 valores equidistantes entre -2 y 2 en los ejes x y y.
  - `np.meshgrid(x, y)`: Crea una malla 2D con las coordenadas X y Y.
- 

## 3. Cálculo de la función Z

python

CopyEdit

```
Z = np.sin(X**2 + Y**2) / (X**2 + Y**2)
```

- $X^2 + Y^2$ : Calcula la distancia al origen en coordenadas cartesianas.
  - $\text{np.sin}(X^2 + Y^2) / (X^2 + Y^2)$ : Aplica la función matemática, que es una forma suavizada de una función sinc en 2D.
- 

#### 4. Creación de la figura y el gráfico en 3D

python

CopyEdit

```
fig = plt.figure()
```

```
ax = fig.add_subplot(111, projection="3d")
```

- `plt.figure()`: Crea una figura donde se dibujará la gráfica.
  - `fig.add_subplot(111, projection="3d")`: Agrega un subplot en 3D a la figura.
- 

#### 5. Graficado de la superficie

python

CopyEdit

```
ax.plot_surface(X, Y, Z, cmap="viridis", edgecolor="k", alpha=0.7)
```

```
ax.set_xlabel("X")
```

```
ax.set_ylabel("Y")
```

```
ax.set_zlabel("Z")
```

```
ax.set_title("Modelado de superficies, Paraboloide")
```

- `ax.plot_surface(X, Y, Z, cmap="viridis", edgecolor="k", alpha=0.7)`:
    - Grafica la superficie con el colormap "viridis", bordes negros (`edgecolor="k"`) y transparencia (`alpha=0.7`).
  - Se configuran las etiquetas de los ejes (X, Y, Z) y el título.
- 

#### 6. Función de animación

python

CopyEdit

```
def update(frame):
```

```
    ax.view_init(elev=30, azim=frame)
```

- `update(frame)`: Modifica la vista de la gráfica en cada frame.
    - `ax.view_init(elev=30, azim=frame)`: Mantiene una elevación fija de 30° y rota en el eje Z con `azim=frame`.
- 

## 7. Creación de la animación

python

CopyEdit

```
ani = FuncAnimation(fig, update, frames=np.arange(0, 360, 2), interval=50)
```

- `FuncAnimation(fig, update, frames=np.arange(0, 360, 2), interval=50)`:
    - Anima la rotación de la vista desde 0° hasta 360° en pasos de 2°.
    - `interval=50`: Controla la velocidad de la animación (50 ms por frame).
- 

## 8. Mostrar la gráfica

python

CopyEdit

```
plt.show()
```

- Muestra la figura y la animación en pantalla.
- 

## Resumen

Este código genera una **superficie en 3D** basada en una función tipo  $\text{sinc}(x,y)$ , la grafica con colores y bordes resaltados, y luego la **rota automáticamente** para visualizarla desde diferentes ángulos. 🚀

## Modelado de superficies, Paraboloide

