Este código en Python utiliza **NumPy**, **Matplotlib** y **Matplotlib Animation** para generar y visualizar un paraboloide en 3D con una animación que lo rota en el eje Z. Lo desglosamos por partes:

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 se usa para manejar datos numéricos y generar las coordenadas.
- matplotlib.pyplot es la librería principal para graficar.
- mpl_toolkits.mplot3d.Axes3D habilita gráficos en 3D.
- FuncAnimation permite animar la rotación del gráfico.

2. Definición de los puntos en el espacio

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 para x y y.
- np.meshgrid(x, y): Crea una malla 2D con todas las combinaciones de x y y.

3. Definición de la superficie Z

python

CopyEdit

```
Z = X^* + Y^*
```

• Calcula $Z = X^2 + Y^2$, que es la ecuación de un **paraboloide**.

4. Creación de la figura y los ejes en 3D

python

CopyEdit

fig = plt.figure()

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

- plt.figure(): Crea la figura principal.
- fig.add_subplot(111, projection="3d"): Agrega un solo subplot en 3D (con 111 indicando una única fila y columna).

5. Dibujar la superficie

python

CopyEdit

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

- plot_surface(X, Y, Z): Dibuja la superficie en 3D.
- cmap="viridis": Usa el colormap "viridis" para los colores.
- edgecolor="k": Borde en negro para resaltar la malla.
- alpha=0.7: Da transparencia al gráfico (70% opaco).

6. Etiquetas y título

python

CopyEdit

ax.set_xlabel("X")

```
ax.set_ylabel("Y")
ax.set_zlabel("Z")
ax.set_title("Modelado de superficies, Paraboloide")
```

• Se establecen los nombres de los ejes y el título de la gráfica.

7. Función de animación

python

CopyEdit

def update(frame):

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

- frame es el ángulo en el que se encuentra la animación.
- ax.view_init(elev=30, azim=frame):
 - o elev=30: Mantiene el ángulo de elevación fijo a 30°.
 - azim=frame: Cambia el ángulo de giro en Z según el frame, rotando la vista.

8. Configurar la animación

python

CopyEdit

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

- frames=np.arange(0, 360, 2): Genera valores de 0° a 360° en pasos de 2° para girar la vista.
- interval=50: Cada frame dura 50ms, lo que hace que la animación sea fluida.

9. Mostrar la gráfica

python

CopyEdit

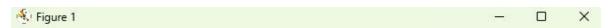
plt.show()

• Muestra la figura con la animación.

Resumen

Este código genera un **paraboloide** en 3D y lo anima rotando la vista alrededor del eje Z.

- Usa numpy para definir la malla de coordenadas.
- Usa matplotlib para graficar la superficie.
- Usa FuncAnimation para rotar la vista en la animación.



Modelado de superficies, Paraboloide

