

Entrega 5 GCOM

Enrique Ernesto de Alvear Doñate

20 de abril de 2022

1. Introducción:

Queremos comprobar como se deforma una esfera quitando el polo norte con una proyección. Para poder verlo mejor vamos a utilizar una curva contenida en la esfera. Usaremos la esfera unidad \mathbb{S}^2 . Lo haremos de dos formas distintas:

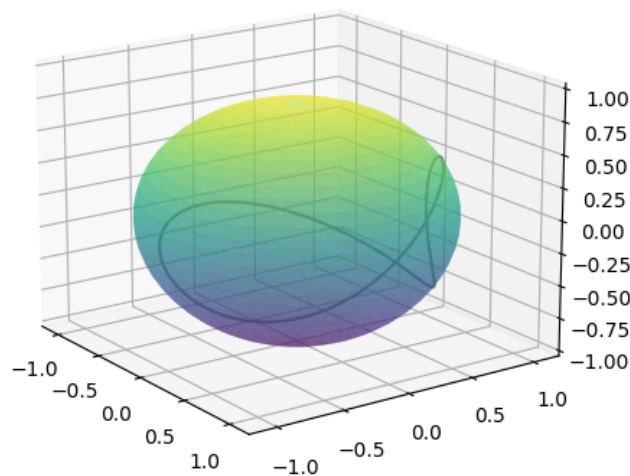
- i) Primero usaremos una proyección estereográfica con $\alpha = 0,5$, y mostraremos como queda la esfera después de la proyección.
- ii) Aquí utilizaremos una familia paramétrica para ver como se va deformando la esfera y mostraré el resultado con un gif.

2. Metodología:

Primero vamos a coger una malla regular de puntos de la esfera con $\varphi \in [0, 2\pi)$ y $\phi \in [0, \pi)$. Y para la curva voy a coger la curva de ecuaciones:

$$\begin{aligned}x &= \cos^2(t) \\ y &= \sin(t) \\ z &= \cos(t)\sin(t)\end{aligned}$$

La esfera que vamos a proyectar será esta:



- i) Para hacer esto primero vamos a coger todos los puntos de la esfera y de la curva y proyectarlos utilizando la fórmula:

$$x_i' = \frac{x_i}{(1-z)^{\frac{1}{2}}}$$

- ii) En este apartado usaremos la familia paramétrica:

$$f_t : S_1^2 \setminus e_3 \rightarrow \mathbb{R}^3$$

$$p = \begin{pmatrix} x \\ y \\ z \end{pmatrix} \mapsto \frac{2}{2(1-t) + (1-z)t} \begin{pmatrix} x \\ y \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ (-1)t + z(1-t) \end{pmatrix}$$

Para hacer el gif iremos variando el valor de t entre 0 y 1, y haremos para cada uno de esos valores la imagen de los puntos de la esfera respecto de la familia paramétrica.

3. Resultados:

- i) Como podemos ver en la gráfica siguiente:

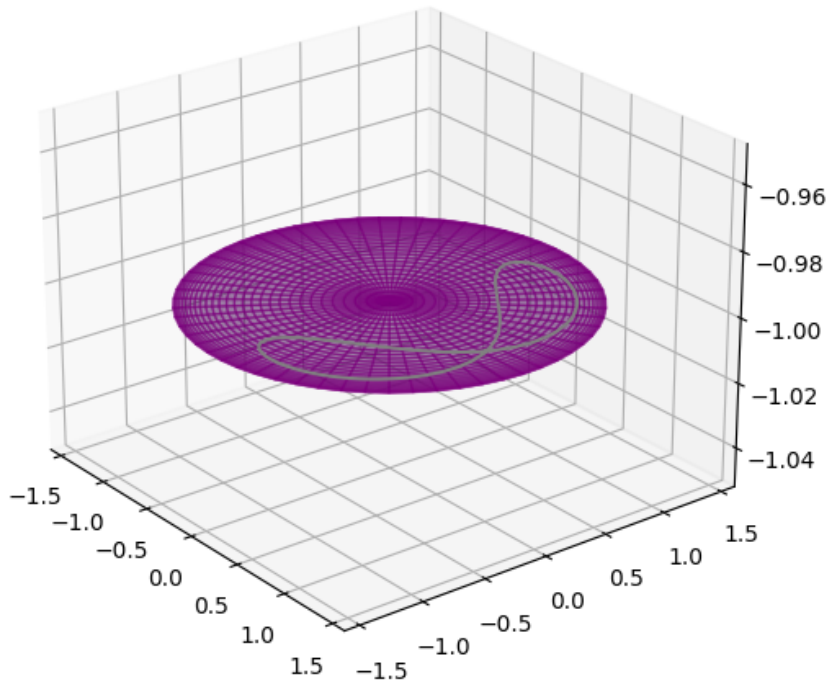


Figura 1: Proyección estereográfica de la esfera y de la curva

- ii) En el archivo .gif adjunto a este documento podemos ver la animación de nuestra familia paramétrica, a la cual hemos quitado el polo norte para que fuese bien todo.

4. Anexo: código utilizado

```
1 # -*- coding: utf-8 -*-
2 """
3 @author: Robert
4 """
5
6 from mpl_toolkits import mplot3d
7
8 import os
9 import numpy as np
10 import matplotlib.pyplot as plt
11 from mpl_toolkits.mplot3d import axes3d
12
13 vuestra_ruta = ""
14
15 os.getcwd()
16 #os.chdir(vuestra_ruta)
17
18
19 # """
20 # Ejemplo1
21 # """
22
23 # fig = plt.figure()
24 # ax = plt.axes(projection='3d')
25 # X, Y, Z = axes3d.get_test_data(0.05)
26 # cset = ax.contour(X, Y, Z, 16, extend3d=True)
27 # ax.clabel(cset, fontsize=9, inline=1)
28 # plt.show()
29
30 # """
31 # Ejemplo2
32 # """
33
34 # def g(x, y):
35 #     return np.sqrt(1-x ** 2 - y ** 2)
36
37 # x = np.linspace(-1, 1, 30)
38 # y = np.linspace(-1, 1, 30)
39
40 # X, Y = np.meshgrid(x, y)
41 # Z = g(X, Y)
42
43
44 # fig = plt.figure()
45 # ax = plt.axes(projection='3d')
46 # ax.contour3D(X, Y, Z, 10, cmap='binary')
47 # ax.contour3D(X, Y, -1*Z, 10, cmap='binary')
48 # ax.set_xlabel('x')
49 # ax.set_ylabel('y')
50 # ax.set_zlabel('z')
51
52
53 # """
54 # Ejemplo3
55 # """
56
57 # fig = plt.figure()
58 # ax = plt.axes(projection='3d')
```

```

59
60 # t2 = np.linspace(1, 0, 100)
61 # x2 = t2 * np.sin(20 * t2)
62 # y2 = t2 * np.cos(20 * t2)
63 # z2 = np.sqrt(1-x2**2-y2**2)
64
65 # c2 = x2 + y2
66
67 # ax.scatter(x2, y2, z2, c=c2)
68 # ax.plot(x2, y2, z2, '-b')
69
70
71 """
72 2-sphere
73 """
74
75 u = np.linspace(0.05, np.pi, 30)
76 v = np.linspace(0, 2 * np.pi, 60)
77
78 x = np.outer(np.sin(u), np.sin(v))
79 y = np.outer(np.sin(u), np.cos(v))
80 z = np.outer(np.cos(u), np.ones_like(v))
81
82
83
84 # t2 = np.linspace(0.01, 1, 200)
85 # x2 = abs(t2) * np.sin(80 * t2/2)
86 # z2 = abs(t2) * np.cos(80 * t2/2)
87 # y2 = np.sqrt(1-x2**2-z2**2)
88 # c2 = x2 + y2
89
90
91 t2 = np.linspace(-np.pi+0.001, np.pi-0.001, 500)
92 x2 = np.cos(t2)**2
93 z2 = np.cos(t2)*np.sin(t2)
94 y2 = np.sin(t2)
95
96 c2 = x2 + y2
97
98
99 fig = plt.figure()
100 ax = plt.axes(projection='3d')
101 #ax.plot(x2, y2, z2, '-b',c="gray")
102
103 ax.plot_surface(x, y, z, rstride=1, cstride=1,
104                cmap='viridis', edgecolor='none')
105
106 ax.plot(x2, y2, z2, '-b',c="gray",zorder=3)
107 #ax.plot_wireframe(x2, y2, z2)
108
109 ax.set_title('surface')
110
111
112
113 """
114 2-esfera proyectada
115 """
116
117 def proj(x,z,z0=1, alpha=1):
118     z0 = z*0+z0

```

```

119     eps = 1e-16
120     #x_trans = x/(abs(z0-z)**alpha+eps)
121     x_trans = x/((abs(1-z)**.5) + eps)
122     return(x_trans)
123     #Nótese que añadimos un épsilon para evitar dividi entre 0!!
124
125
126 z0 = 1
127
128 fig = plt.figure(figsize=(12,12))
129 fig.subplots_adjust(hspace=0.4, wspace=0.2)
130
131 c2 = np.sqrt(x2**2+y2**2)
132 col = plt.get_cmap("hot")(c2/np.max(c2))
133
134 ax = fig.add_subplot(2, 2, 1, projection='3d')
135 ax.plot_surface(x, y, z, rstride=1, cstride=1,
136               cmap='viridis', edgecolor='none')
137 #ax.plot(x2, y2, z2, '-b',c="gray",zorder=3)
138 ax.scatter(x2, y2, z2, '-b',c=col,zorder= 3,s=0.1)
139 ax.set_title('2-sphere')
140 #ax.text(0.5, 90, 'PCA-'+str(i), fontsize=18, ha='center')
141
142 ax = fig.add_subplot(2, 2, 2, projection='3d')
143 #ax.set_xlim3d(-8,8)
144 #ax.set_ylim3d(-8,8)
145 #ax.set_zlim3d(0,1000)
146 ax.plot_surface(proj(x,z,z0=z0), proj(y,z,z0=z0), z*0-1, rstride=1, cstride=1,
147               cmap='viridis', alpha=0.5, edgecolor='purple')
148
149 ax.plot(proj(x2,z2,z0=z0), proj(y2,z2,z0=z0), 1, '-b',c="gray",zorder=1)
150 #ax.scatter(proj(x2,z2,z0), proj(y2,z2,z0), -1, '-b',c="gray",zorder= 3,s=0.01)
151 ax.set_title('Stereographic projection')
152 plt.show()
153 fig.savefig('stereo1.png', dpi=250)    # save the figure to file
154 plt.close(fig)
155
156
157
158 """
159 2-esfera proyectada - familia paramétrica - FORMA INCORRECTA
160 """
161
162 t = 1
163 z0 = 1
164
165 xt = proj(x,z,z0)*(t)+x*(1-t)
166 yt = proj(y,z,z0)*t+y*(1-t)
167 zt = (z*0-z0)*t + z*(1-t)
168 #Esto es para la curva
169 x2t = proj(x2,z2,z0)*t+x2*(1-t)
170 y2t = proj(y2,z2,z0)*t+y2*(1-t)
171 z2t = -z0*t+z2*(1-t)
172
173 fig = plt.figure(figsize=(6,6))
174 #fig.subplots_adjust(hspace=0.4, wspace=0.2)
175 ax = plt.axes(projection='3d')
176
177
178 # ax.set_xlim3d(-8,8)

```

```

179 # ax.set_ylim3d(-8,8)
180 ax.plot_surface(xt, yt, zt, rstride=1, cstride=1,
181                cmap='viridis',alpha=0.5 ,edgecolor='purple')
182 ax.plot(x2t,y2t, -1, '-b',c="gray",zorder=3)
183
184 plt.show()
185 fig.savefig('stereo2.png') # save the figure to file
186 plt.close(fig)
187
188
189
190 """
191 HACEMOS LA ANIMACIÓN
192 """
193
194 from matplotlib import animation
195 #from mpl_toolkits.mplot3d.axes3d import Axes3D
196
197 def proj2(x,z,z0=1):
198     z0 = z*0+z0
199     eps = 1e-16
200     x_trans = 2*x/(abs(z0-z)+eps)
201     return(x_trans)
202
203 def animate(t):
204
205     #Esto primero es para la esfera
206     xt = (2/(2*(1-t)+(1-z)*t))*x
207     yt = (2/(2*(1-t)+(1-z)*t))*y
208     zt = - t +z*(1-t)
209     #Esto es para la curva
210     x2t = (2/(2*(1-t)+(1-z2)*t))*x2
211     y2t = (2/(2*(1-t)+(1-z2)*t))*y2
212     z2t = - t +z2*(1-t)
213     # xt = proj2(x,z,z0)*t+x*(1-t)
214     # yt = proj2(y,z,z0)*t+y*(1-t)
215     # zt = (-1)*t + z*(1-t)
216     # x2t = proj2(x2,z2,z0)*t+x2*(1-t)
217     # y2t = proj2(y2,z2,z0)*t+y2*(1-t)
218     # z2t = (-1)*t+z2*(1-t)
219     ax = plt.axes(projection='3d')
220     ax.set_zlim3d(-1,1)
221     ax.plot_surface(xt, yt, zt, rstride=1, cstride=1, alpha=0.5,
222                    cmap='viridis', edgecolor='none')
223     ax.plot(x2t,y2t, z2t, '-b',c="gray")
224     return ax
225
226 def init():
227     return animate(0),
228
229 animate(np.arange(0, 1,0.1)[1])
230 plt.show()
231
232
233 fig = plt.figure(figsize=(6,6))
234 ani = animation.FuncAnimation(fig, animate, np.arange(0,1,0.01), init_func=init,
235                               interval=60)
236 ani.save("familiaparamEEAD.gif", fps = 10)
237 #ani.save("ejemplo.gif", fps = 5)
238

```

```
239 """
240 Alternativas para hacer animación
241 #celluloid
242 pypi.org/project/celluloid
243
244 #APNG
245 from apng import APNG
246 APNG.from_files(['atleta-01.jpg',
247                 'atleta-02.jpg',
248                 'atleta-03.jpg',
249                 'atleta-04.jpg',
250                 'atleta-05.jpg'],
251                 delay=100).save('animatleta1.png')
252
253 """
254 #Alternativas:
255 #
```