

Rotación de vectores

La finalidad de esta tarea es preparar el camino para la utilización de operaciones matriciales que serán elemento crucial en aplicaciones con gran cantidad de datos.

En esta tarea se refresca el uso básico de `python` y una de sus librerías fundamentales, `numpy`. Esta librería es muy eficiente (implementada en C) y debe sustituir en la medida de lo posible cualquier bucle `for` en operaciones numéricas.

Descripción

Se crearán cuatro funciones con la finalidad de rotar un punto cualesquiera en el espacio tomando como eje de rotación ya sea el eje X, Y o Z, según desee el usuario. Tres de las funciones se refieren a las rotaciones con respecto a cada eje mientras que la cuarta es una función del más alto nivel; en esta el usuario especificará las coordenadas del punto a rotar, el ángulo deseado y el eje que quiere utilizar para la rotación.

Para llevar acabo esta tarea se deben rellenar las funciones dadas en el archivo `rotacion.py`. Los pasos se describen en los siguientes problemas.

Problema 1: Documentación

Documente cada una de las funciones en `rotacion.py` de tal manera que un usuario pueda hacer uso de cada función. Para esto siga las siguientes especificaciones.

Especificaciones: Describa el propósito de la función de manera concisa en uno o dos renglones en la sección `Descripcion` ubicada en la primer línea del docstring de cada función (se conoce como docstring al texto que aparece inmediatamente después de la definición de la función).

Cuando es necesario en un siguiente párrafo se da una descripción mas detallada; nosotros no necesitaremos hacer esto para una función tan sencilla.

En la siguiente sección llamada `Parameters` documentaremos cada parámetro de manera consistente incluyendo el nombre del parámetro, el tipo de dato que se espera (`int`, `string`, `float`, `complex` o algún otro) y una breve descripción de su rol.

Ejemplo:

```
1  import numpy as np
2
3  def area_circulo(radio:float) -> float:
4      """
5          Calcula el area de un círculo utlizando su radio.
6
7          Parameters:
8              radio (float): radio del círculo.
9
10         Returns:
11             float: el area del círculo
12         """
13         area = np.pi * radio**2
14         return area
```

Problema 2: Matrices de Rotación

Implemente las tres funciones (`rot_x`, `rot_y`, `rot_z`) que llevan a cabo la rotación de un vector cualquiera en el espacio tomando como eje de rotación cada uno de los ejes cartesianos; es decir, una función donde el eje de rotación sea el eje X , otra función con Y como eje de rotación y otra tomando Z como eje de rotación.

Especificaciones:

Cada función debe tomar las tres coordenadas del punto arbitrario a rotar y el ángulo deseado. Las coordenadas y ángulo son números reales que pueden contener punto decimal. La función debe regresar un `numpy.array` correspondiente al vector rotado.

Hint:

- 1) Las coordenadas recibidas por la función deben transformarse a un vector `numpy.array`.
- 2) La matriz de rotación correspondiente debe introducirse como un `numpy array`.
- 3) El vector rotado \mathbf{p}' está dado por un producto matricial $\mathbf{p}' = \mathbf{R}\mathbf{p}$, donde \mathbf{p} es el vector inicial y \mathbf{R} es la matriz de rotación.

La creación de vectores y matrices aparece en cualquier tutorial sobre `numpy`, así como el comando para realizar la multiplicación matricial.

Problema 3: Wrapper

Implemente la función `rotar` la cual reciba las coordenadas del punto a rotar, el ángulo y el eje deseado de rotación. Esta función debe llamar alguna de las tres funciones implementadas en el paso anterior.

JESUS.ABUNDIS@INFO.UAS.EDU.MX