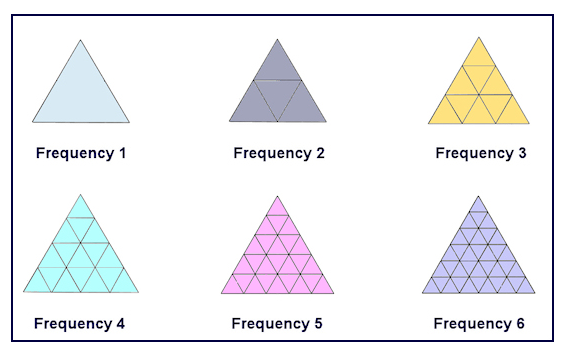
## Compare geodome libs

Requirements

* Tessellation

**geodesicDome**

* <https://pypi.org/project/geodesicDome/>
* Does not state the type of dome need to check
* Best looking
* Features
  + GeodesicDome
    - Creates a given geodesic dome with a given frequency.



* + find\_neighbours
  + get\_triangles
    - Returns: np.ndarray: the triangles of the geodesic dome
  + Tessellate
  + **License:** MIT

pyDome

* calculates vertices and chords of Class One geodesic domes of arbitrary size.
* Not clear documentation on how dome is made
* More for outputting an image of dome itself doesn’t look suitable for our purpose
* No tessellation
* <https://github.com/badassdatascience/pyDome>

**antiprism-python**

* [**https://pypi.org/project/antiprism-python/**](https://pypi.org/project/antiprism-python/)
* **Not clear documentation packaged with other programs:**
  + solve a specific problem, some to solve a general problem, some were written as prototypes
  + programs vary in quality
* functions
  + get\_ico\_coords()
    - Return icosahedron coordinate values
  + Get triangle
* **License:** MIT
* No tessellation

## Backend

* Angular
  + Single page
  + Frontend only application
* Django/flask
  + Similar in functionality more experience with Django
  + Don’t need backend for application
  + Might have to many unneeded features
* Non-web passed
  + Need to compare the native python visualisation methods to the .js visualisation

Visualisation

Web/js

* D3js
  + <https://d3js.org/>
  + Works with angular

Non-web

* Pyopengl with pygame