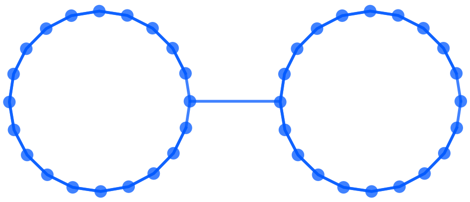
project 17: back to self-organizing maps (SOMs)

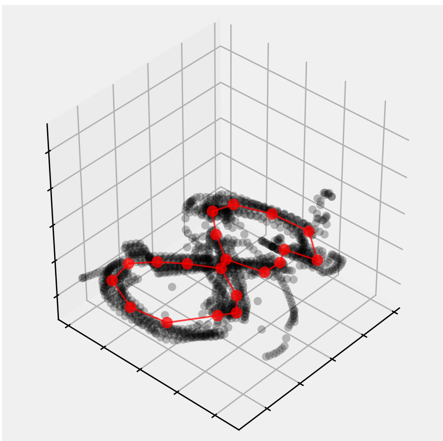
task 17.1: SOMs of peculiar topology

Recall that the file `q3dm1-path2.csv` contains human trajectory data, i.e. a sequence of 3D locations $\boldsymbol{x}_t$ the avatar of a human player was seen at at time t while moving around the Quake III map *q3dm1*.

Load the content of `q3dm1-path2.csv` into a data matrix $\boldsymbol{X}$ and fit a self organizing map (SOM) of k neurons to this data. Your SOM should have the following topological structure



That is, the neurons of the SOM should form two rings of $\frac{k}{2}$ neurons each and both rings should be connected by a "bridge" as shown.
Experiment with different choices of k and plot your results. For $k = 20$, for instance, you result should look something like this



Obviously, this plot shows SOM weights and their connections in red. However, when you implement routines for plotting results like this, make sure your implementation can plot the data points in black and the SOM weights in blue.

task 17.2: SOM batch training

In the supplementary material for lecture 14, we discussed the idea of SOM batch training. Letting

$$X = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\} \subset \mathbb{R}^m$$

denote a given data set and

$$W = \{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_k\} \subset \mathbb{R}^m$$

denote the weights of a SOM (of whatever topology), batch training works like this

for $t = 1, \ldots, t_{\max}$
  for $j = 1, \ldots, n$
    $b_j = \underset{i}{\operatorname{argmin}} \big\|\boldsymbol{w}_i(t) - \boldsymbol{x}_j\big\|^2$
  for $i = 1, \ldots, k$
    $\boldsymbol{w}_i(t) = \frac{\sum_j \boldsymbol{x}_j \cdot h(b_j, i, t)}{\sum_j h(b_j, i, t)}$

Implement this algorithm and use it to train a SOM of the above two-ring topology on the data in `q3dm1-path2.csv`. Visualize your results.