SM402-Basic Computational Topology

IMTECH-CSE

4th Semester

Results of course project on Topological Persistence

Authors:

Vidhish Trivedi IMT2021055
Madhav Sood IMT2021009

April-May, 2023

Contents

1 Introduction 2

2 Project description 2

3 Results 3

1 Introduction

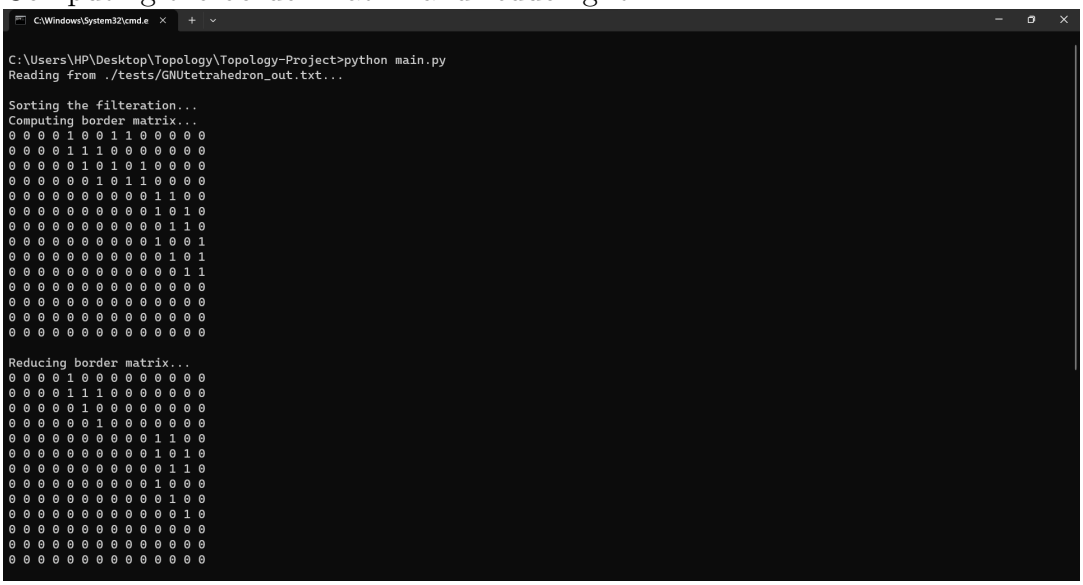
- This project was made as part of the course: Basic Computational Topology (SM402), at IIIT-Bangalore.
- **Course Instructor:** Prof. Amit Chattopadhyay

2 Project description

- This project is based on the paper: **Topological Persistence and Simplification - Herbert Edelsbrunner • David Letscher • Afra Zomorodian**
- The paper introduces several key contributions to the field of computational topology and data analysis.
- Firstly, the paper defines the concept of persistence for Betti numbers and nonbounding cycles, which provides a measure of the significance and lifetime of topological features in a data set.
- Secondly, the paper presents an efficient algorithm for computing persistence, which is applicable to a wide range of data types, including point clouds, digital images, and biomedical data. The algorithm discusses the persistence diagram, which visually represents the evolution of topological features over a range of parameter values.
- Finally, the paper describes a simplification algorithm based on persistence, which can reduce the complexity of a data set by identifying and removing topological features with low persistence. This approach allows for the extraction of the essential structure of the data while discarding irrelevant details, which can be useful for applications such as data visualization and pattern recognition.
- **Note:** The problem with the pairing algorithm is that there is no way, given in the paper, to compute whether a given simplex is positive or negative in a given filtration. Thus we move on to the next algorithm to find the persistence (*as discussed with the professor*). The following is also the algorithm implemented by us in our code.

3 Results

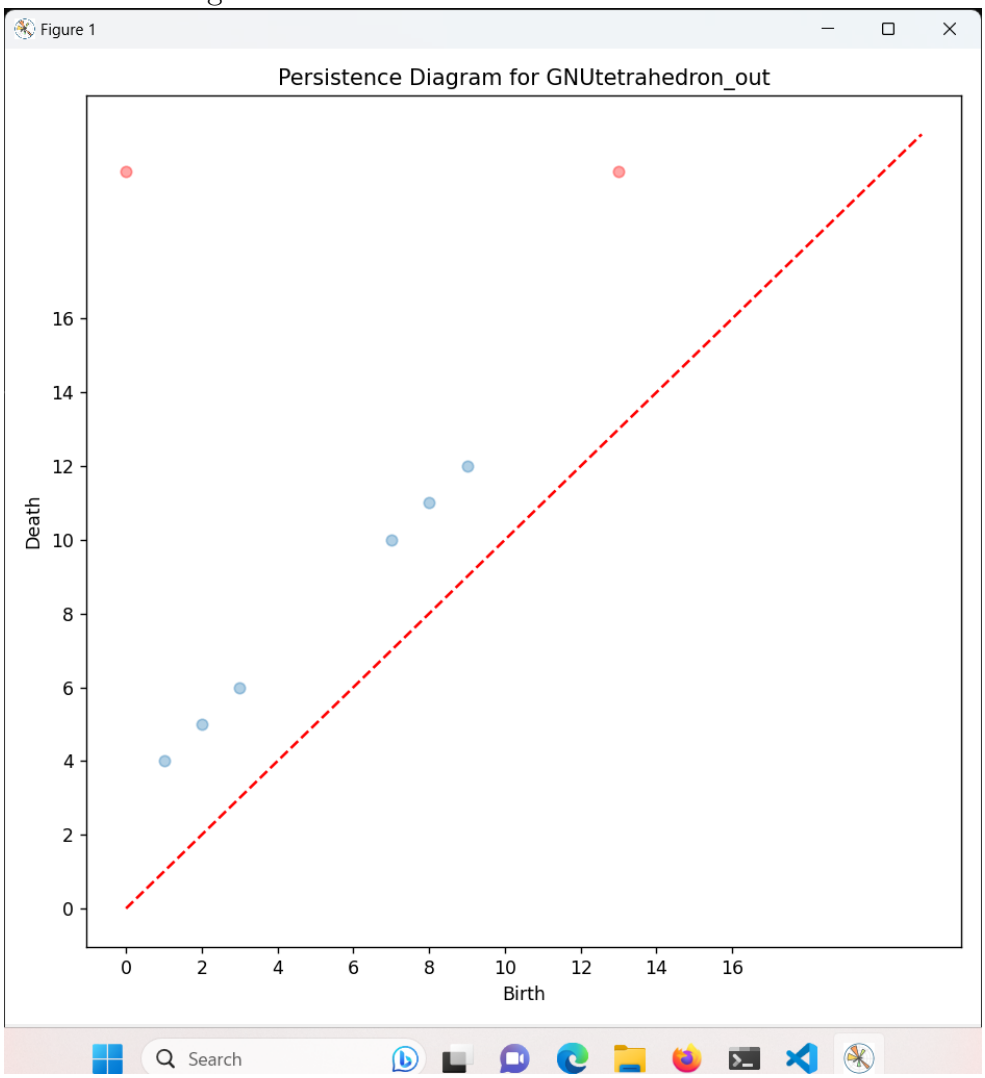
- The project can be used with sample files from *GNU triangulated surface library* (https://gts.sourceforge.net/samples.html).
- See the README.md file on GitHub (https://github.com/Vidhish-Trivedi/Topology-Project) for more details.
- The code will generate an interactive matplotlib window to display the persistence diagram, which can be zoomed into.
- The red dots on the plot denote simplices which do not die (i.e, their death time is infinite).
- **Results:**
  1. **Tetrahedron:** (https://gts.sourceforge.net/samples/tetrahedron.gts.gz)
     - 4 vertices, 6 edges, 4 triangles.
     - Computing the border matrix and reducing it



     - Generating barcode (persistence pairs)

```
Generating bar code intervals

Generating Persistence Diagram

'dimension': 0, 'Birth': 0.0, 'Death': inf
'dimension': 0, 'Birth': 1.0, 'Death': 4.0
'dimension': 0, 'Birth': 2.0, 'Death': 5.0
'dimension': 0, 'Birth': 3.0, 'Death': 6.0
'dimension': 1, 'Birth': 7.0, 'Death': 10.0
'dimension': 1, 'Birth': 8.0, 'Death': 11.0
'dimension': 1, 'Birth': 9.0, 'Death': 12.0
'dimension': 2, 'Birth': 13.0, 'Death': inf

Persistence Diagram saved.


Persistence Diagram display is being rendered.
```

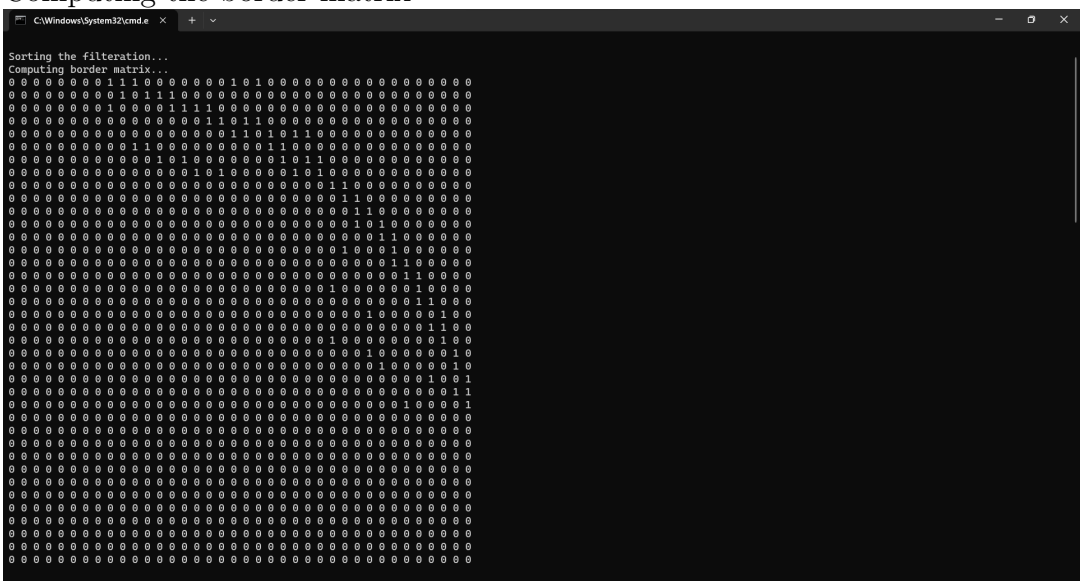
– Persistence diagram



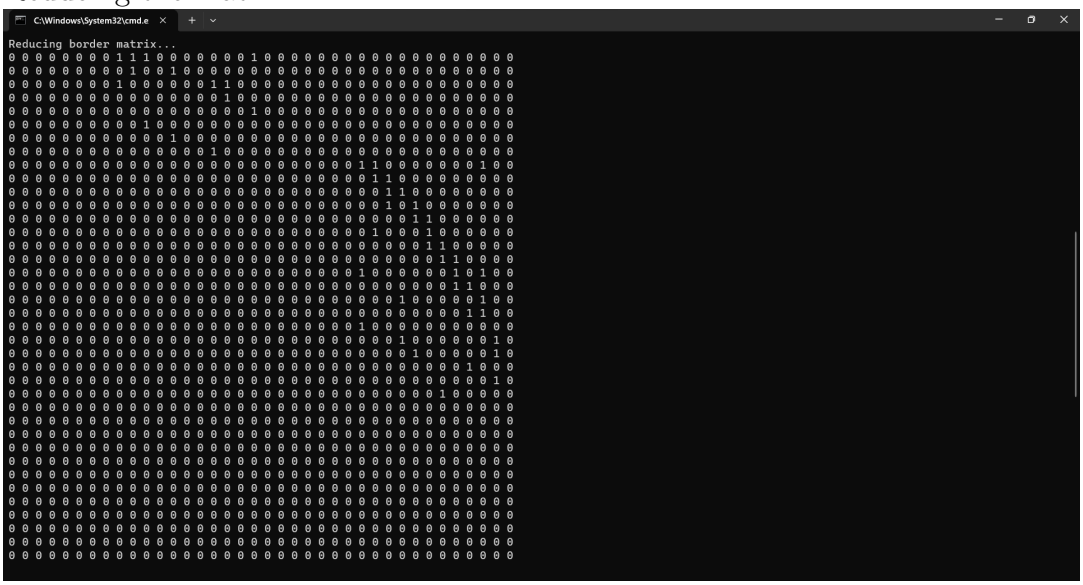
2. **Cube:** (https://gts.sourceforge.net/samples/cube.gts.gz)

– 8 vertices, 18 edges, 12 triangles.

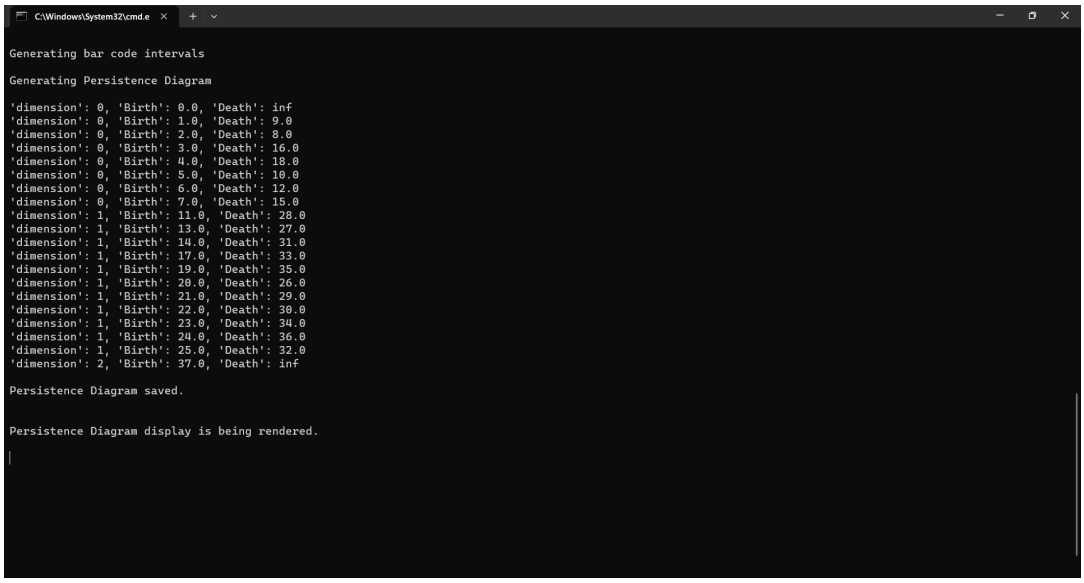
– Computing the border matrix



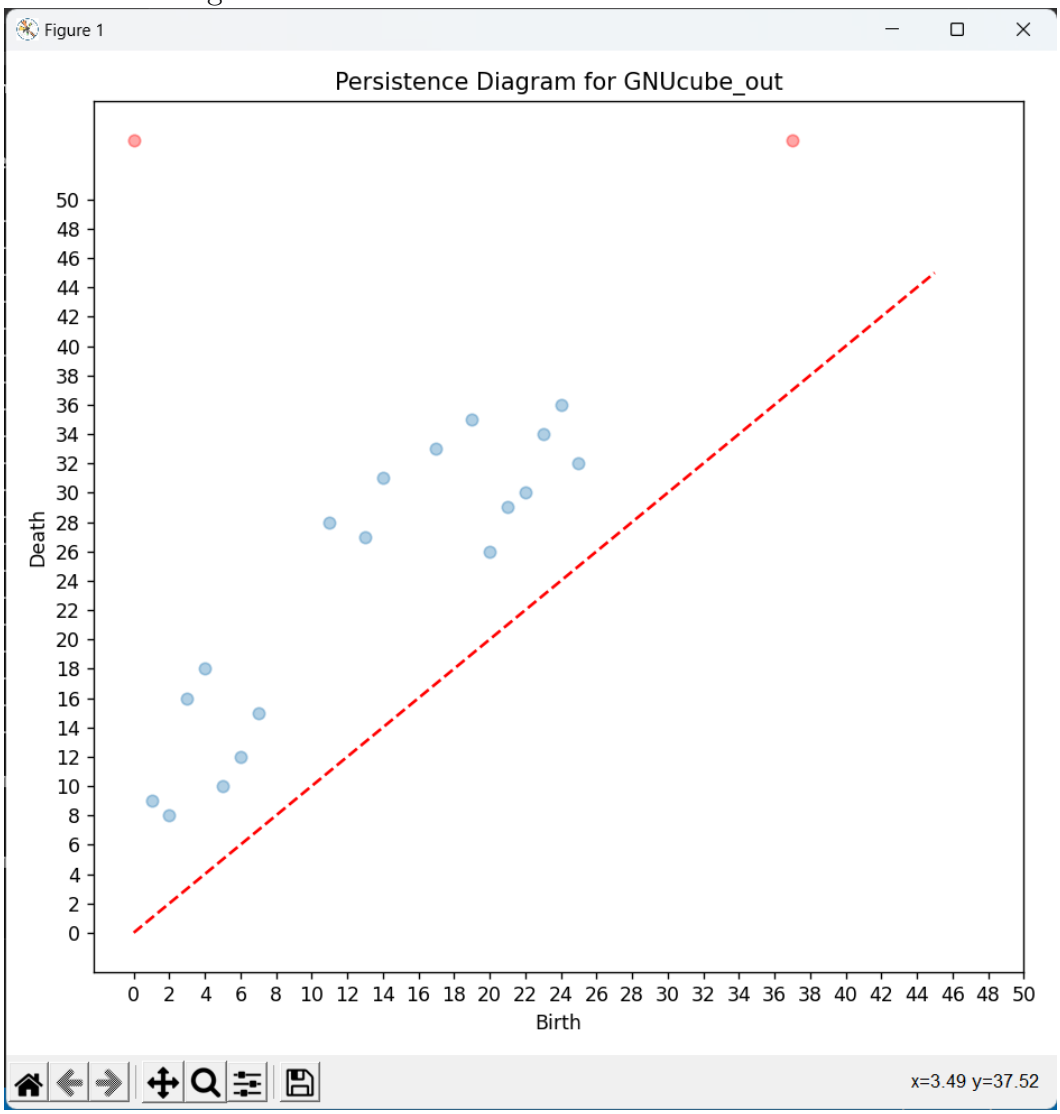
– Reducing the matrix



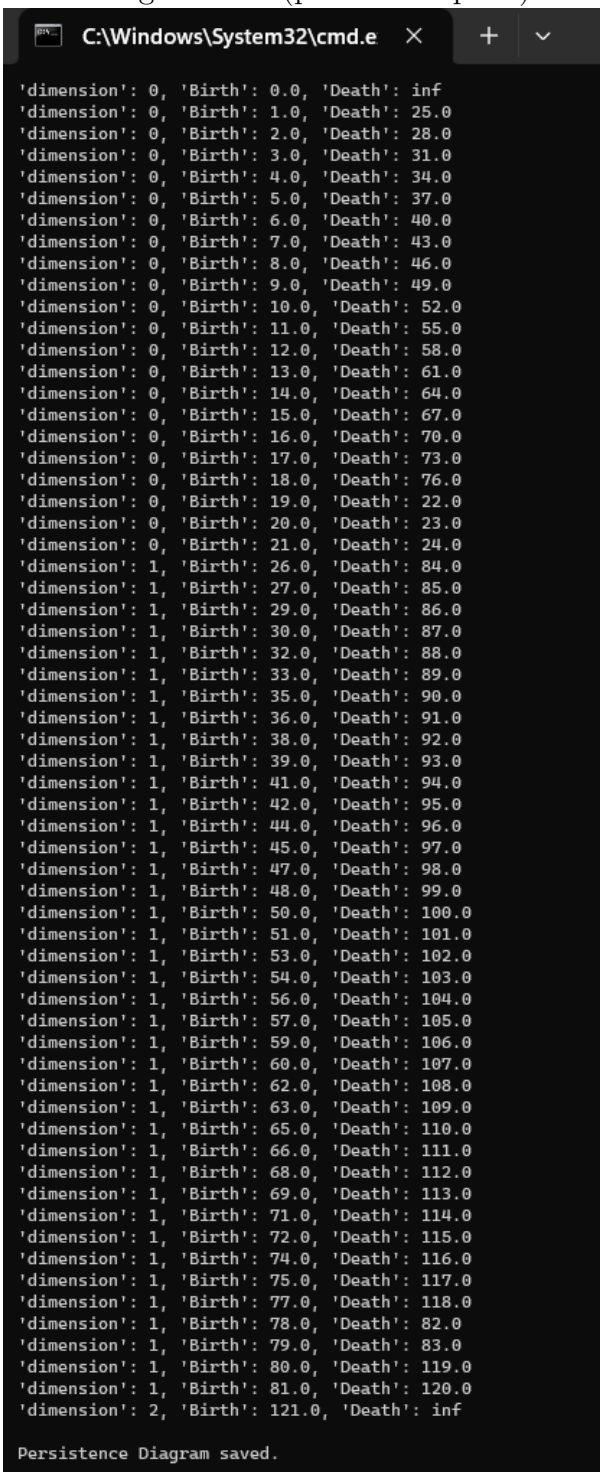
– Generating barcode (persistence pairs)



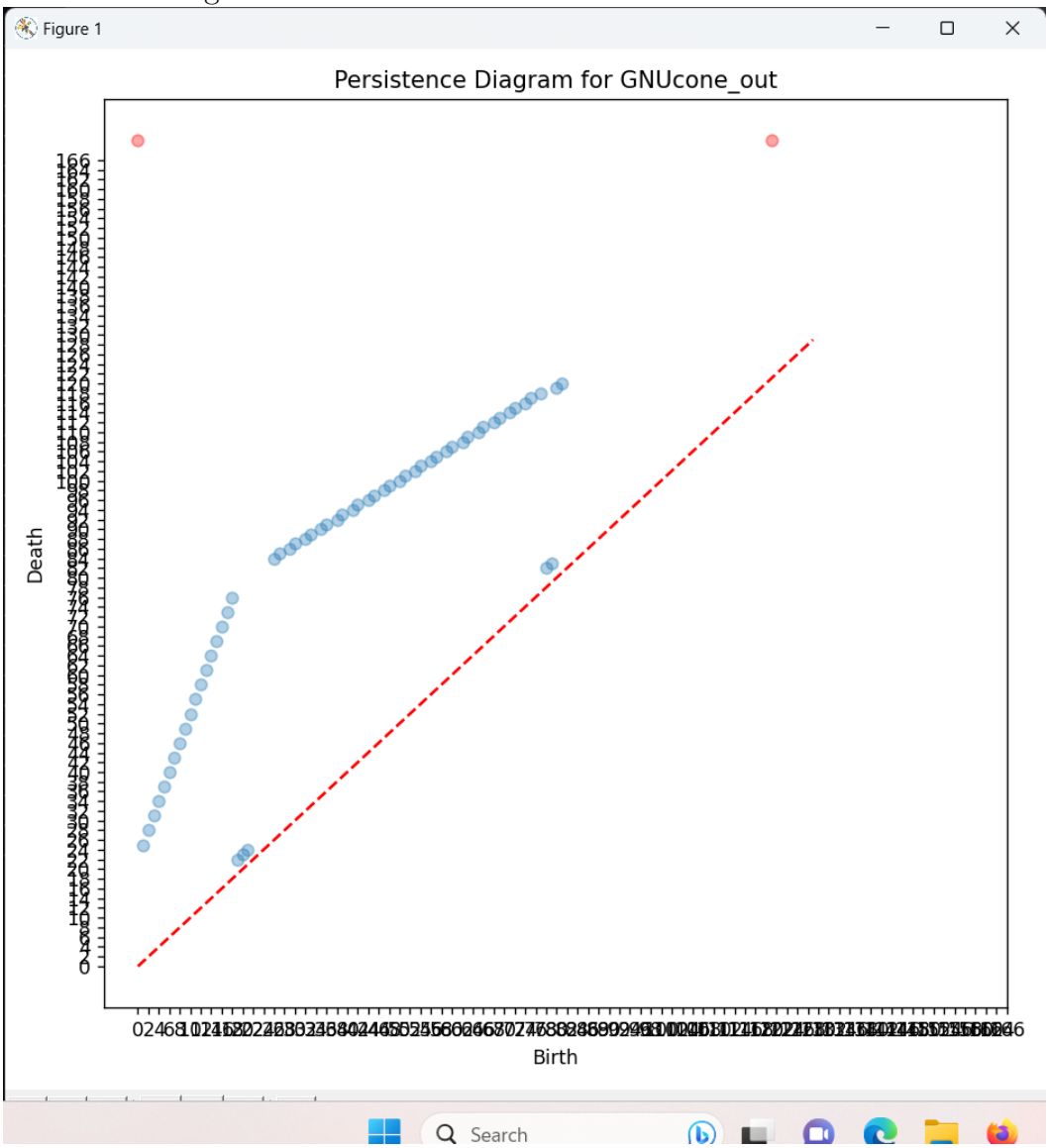
– Persistence diagram



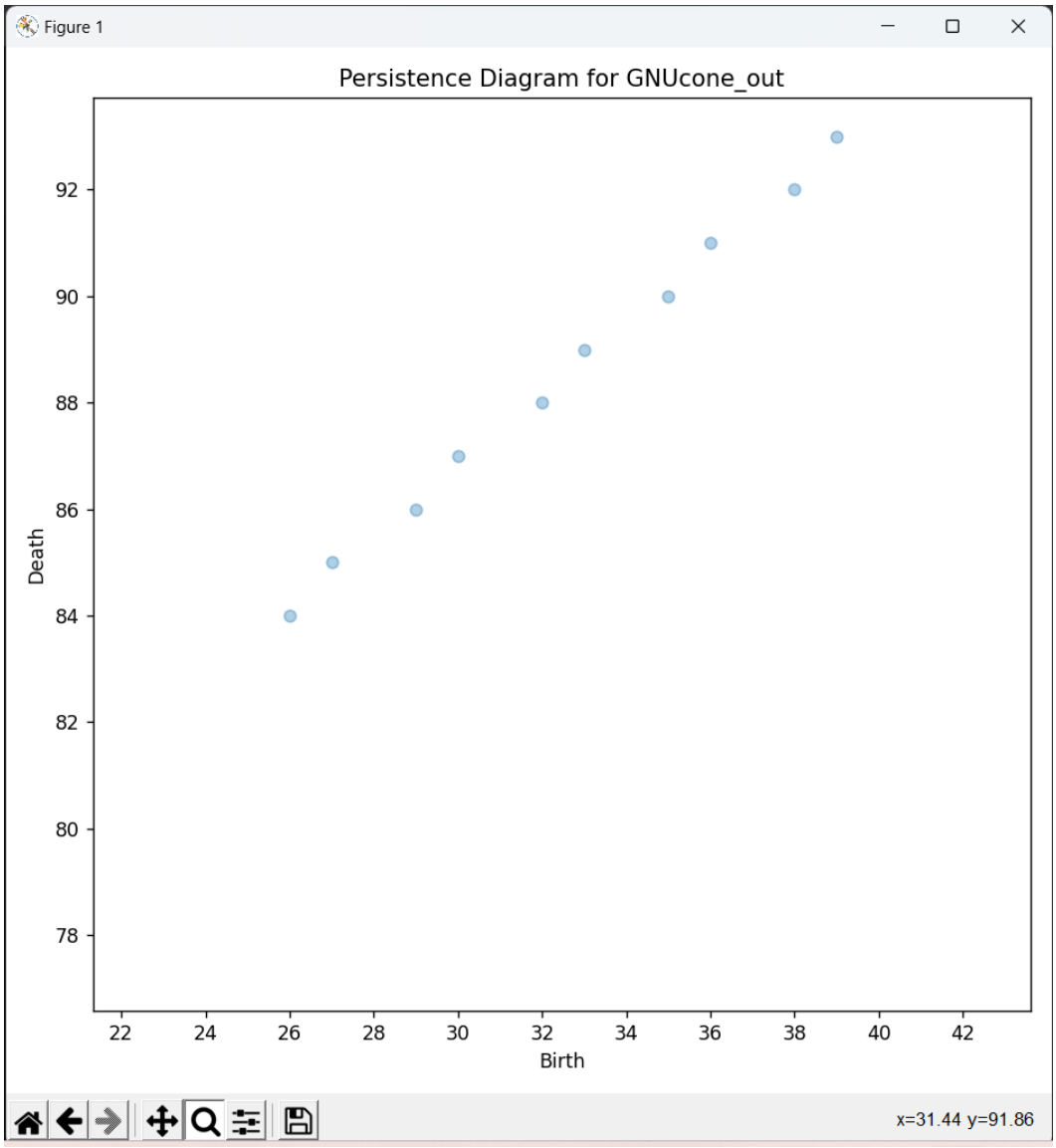
3. Results (both the persistence pair logs and persistence diagrams) for various test data have been submitted in the zip file (as well as on GitHub). As the border matrices can get quite large, we shall display only the persistence diagrams and persistence pair logs from now on.
4. **Cone:** (https://gts.sourceforge.net/samples/cone.gts.gz)
   - 22 vertices, 60 edges, 40 triangles.
   - Generating barcode (persistence pairs)



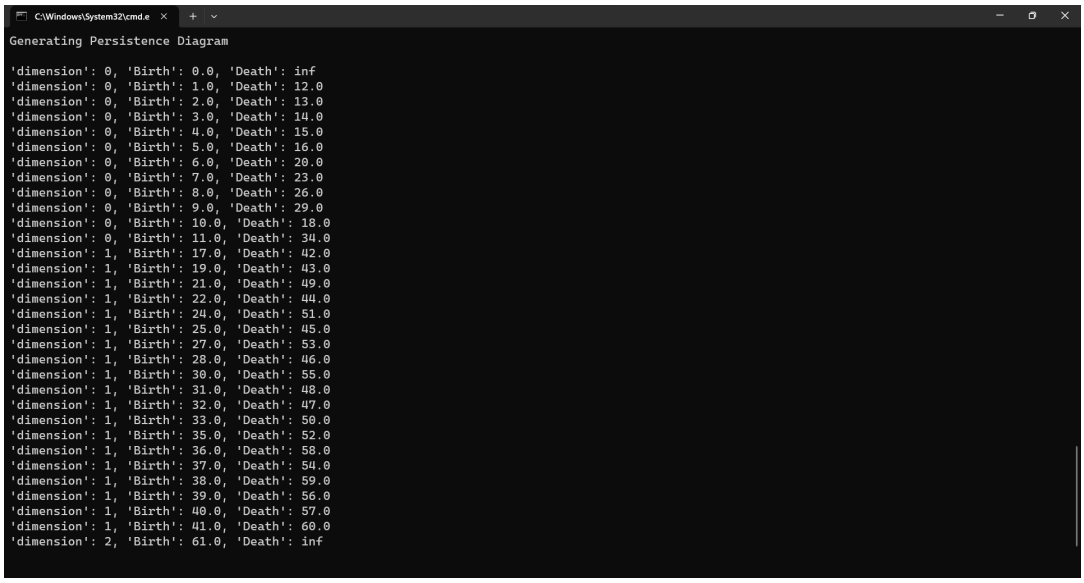
– Persistence diagram



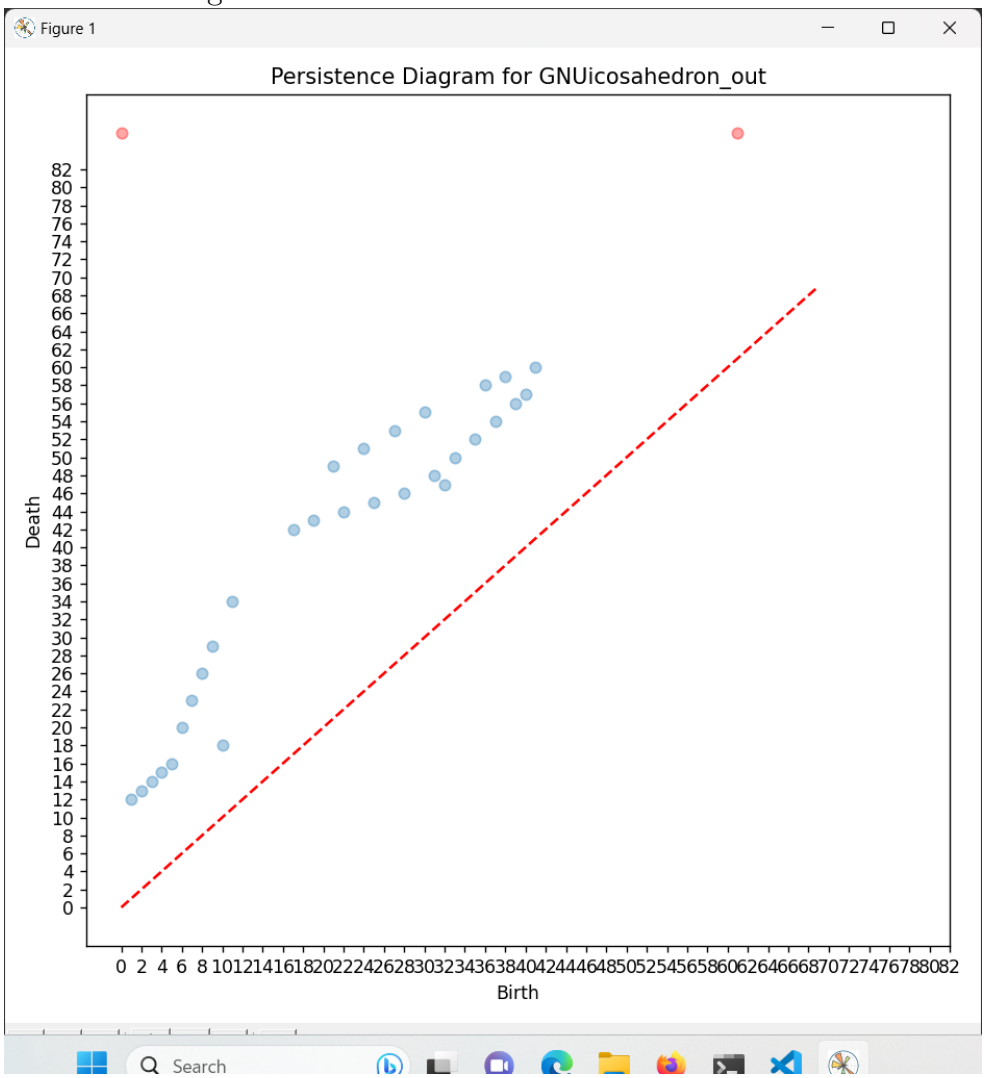
– Persistence diagram (zoomed)



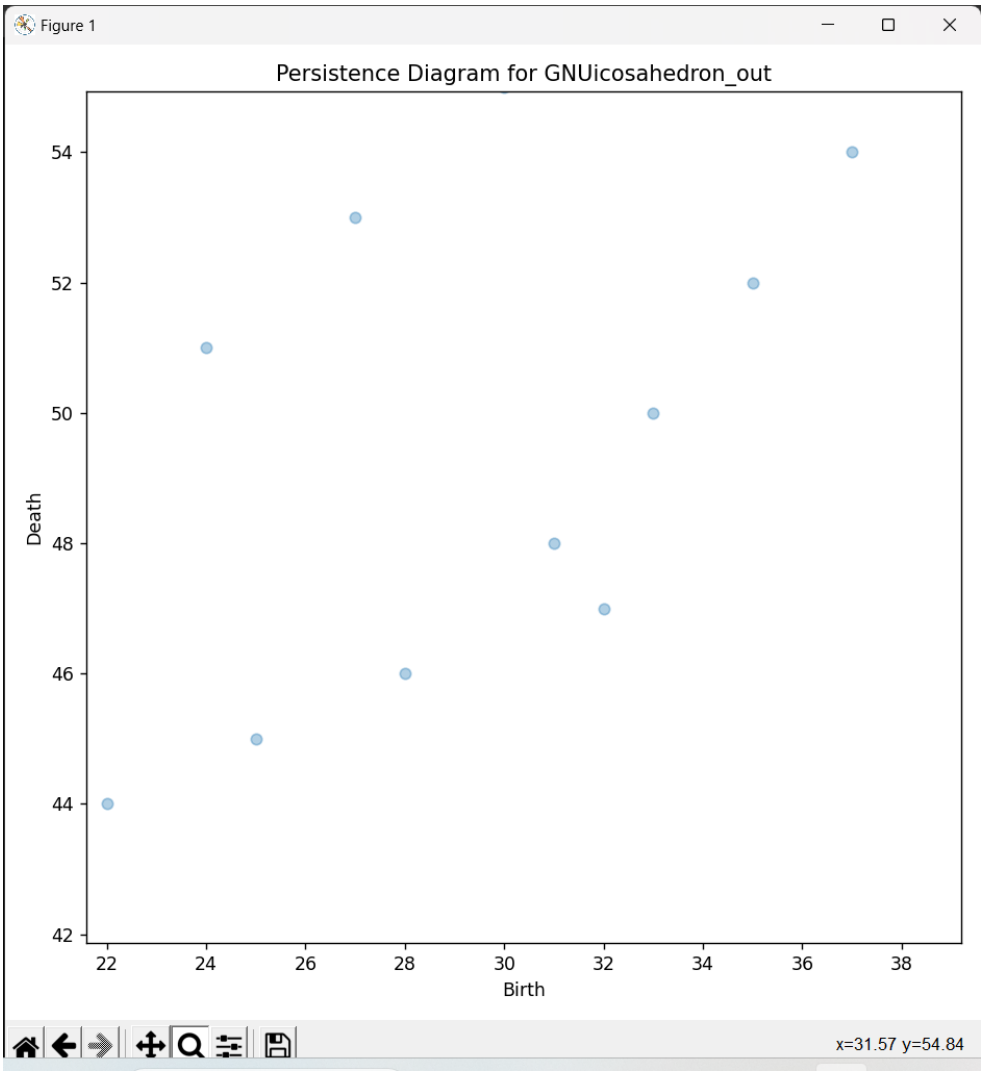
5. **Icosahedron:** (https://gts.sourceforge.net/samples/icosa.gts.gz)
   - 12 vertices, 30 edges, 20 triangles.
   - Generating barcode (persistence pairs)



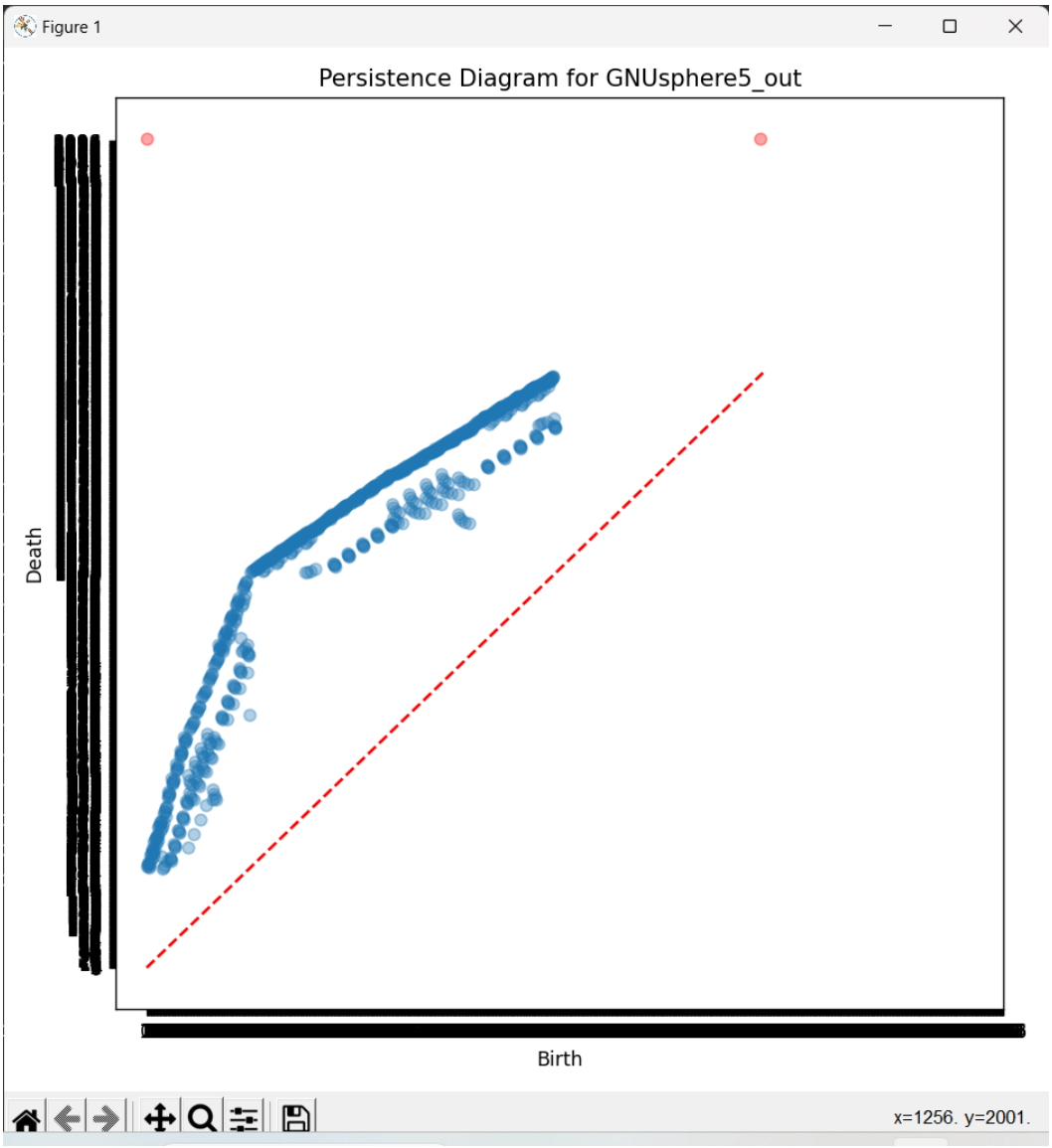
– Persistence diagram



– Persistence diagram (zoomed)

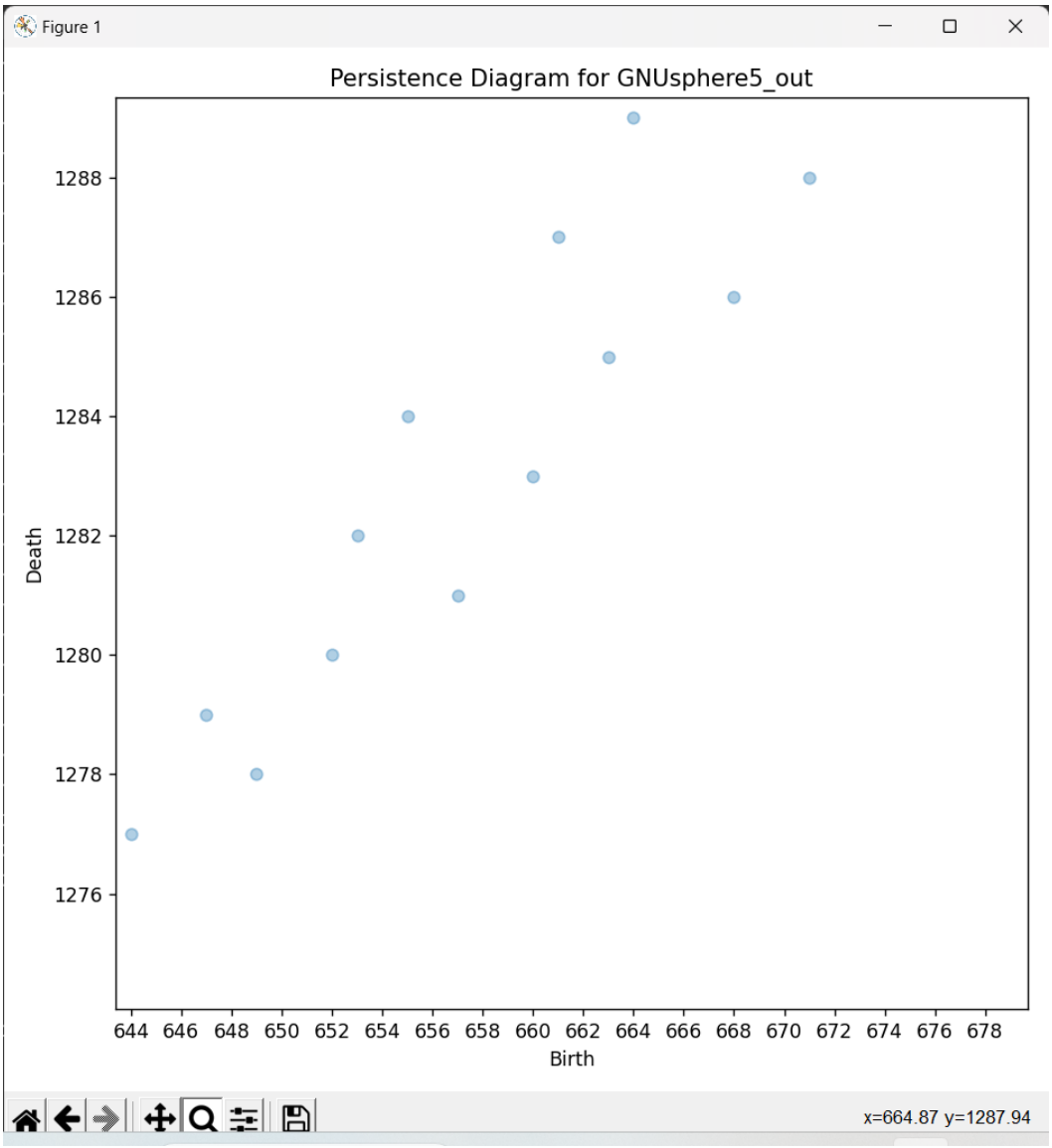


6. **Sphere5:** (https://gts.sourceforge.net/samples/sphere5.gts.gz)
   - 252 vertices, 750 edges, 500 triangles.
   - Barcode (persistence pairs) are omitted here as the output is over 1000 lines. It can be found attached with the submission.
   - Persistence diagram

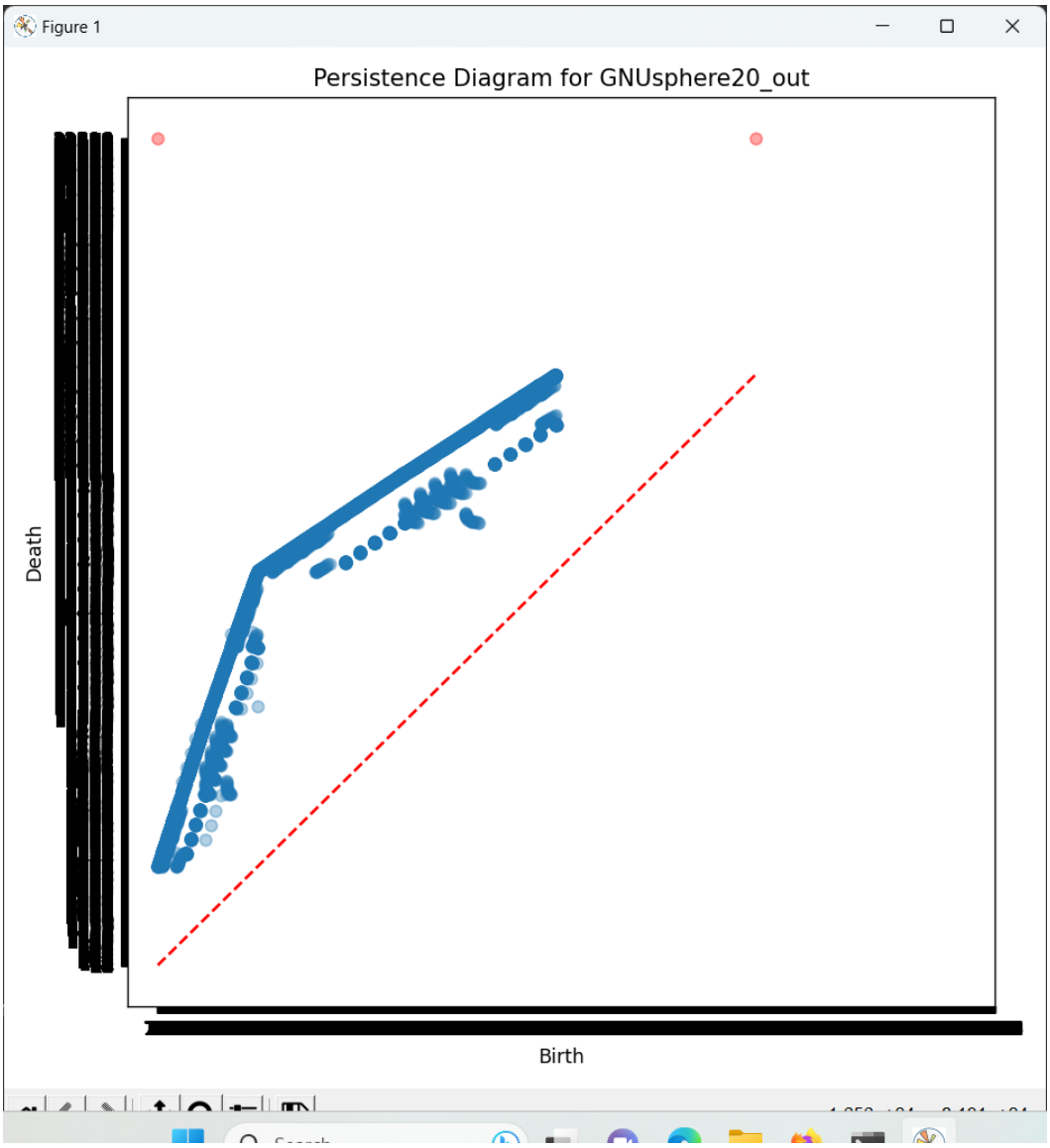


– Persistence diagram (zoomed)

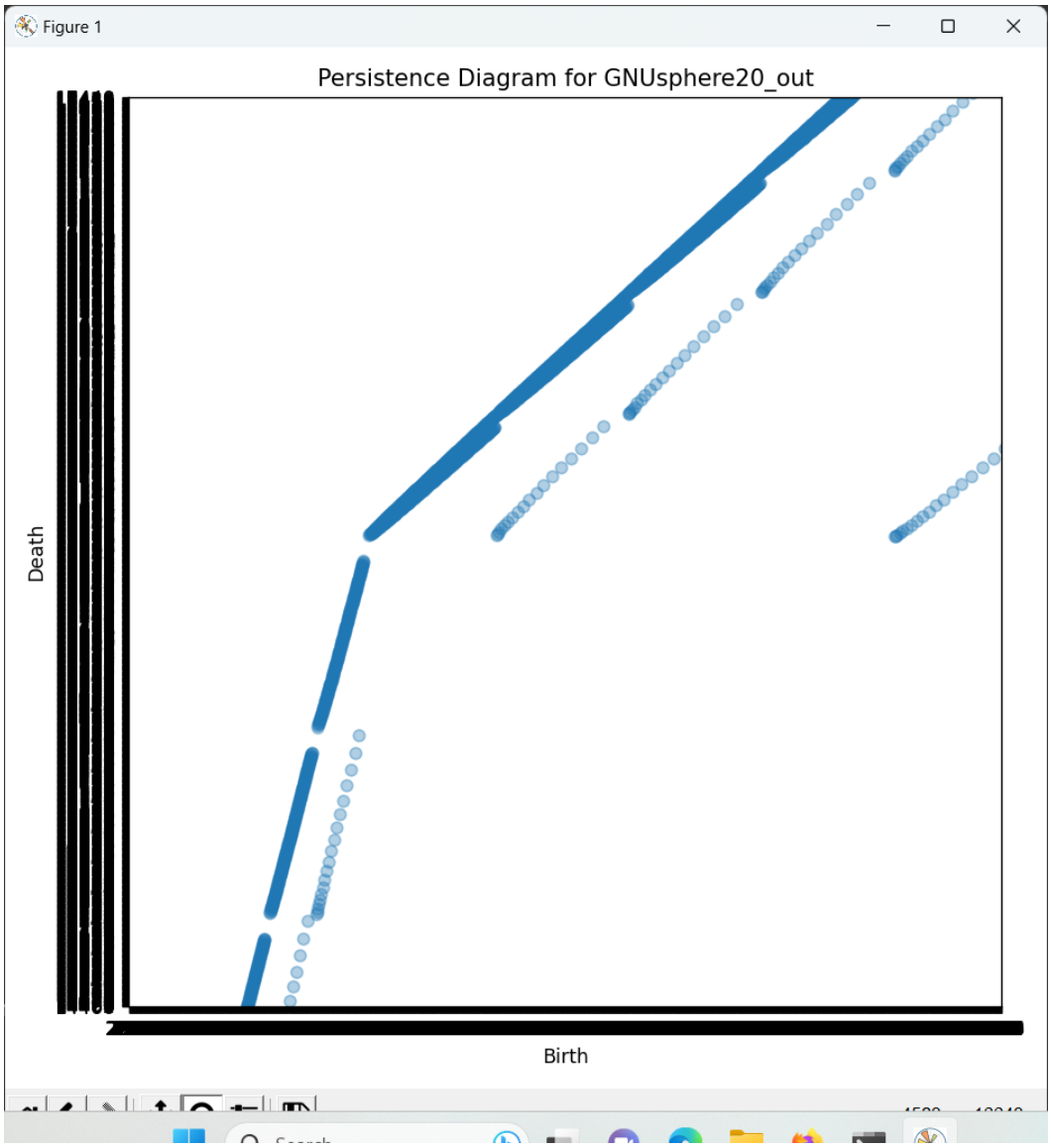
– Persistence diagram (zoomed)



7. **Sphere20:** (https://gts.sourceforge.net/samples/sphere20.gts.gz)
   - 4002 vertices, 12000 edges, 8000 triangles.
   - Barcode (persistence pairs) are omitted here as the output is over 24000 lines. It can be found attached with the submission.
   - Persistence diagram



– Persistence diagram (zoomed)



– Persistence diagram (zoomed)

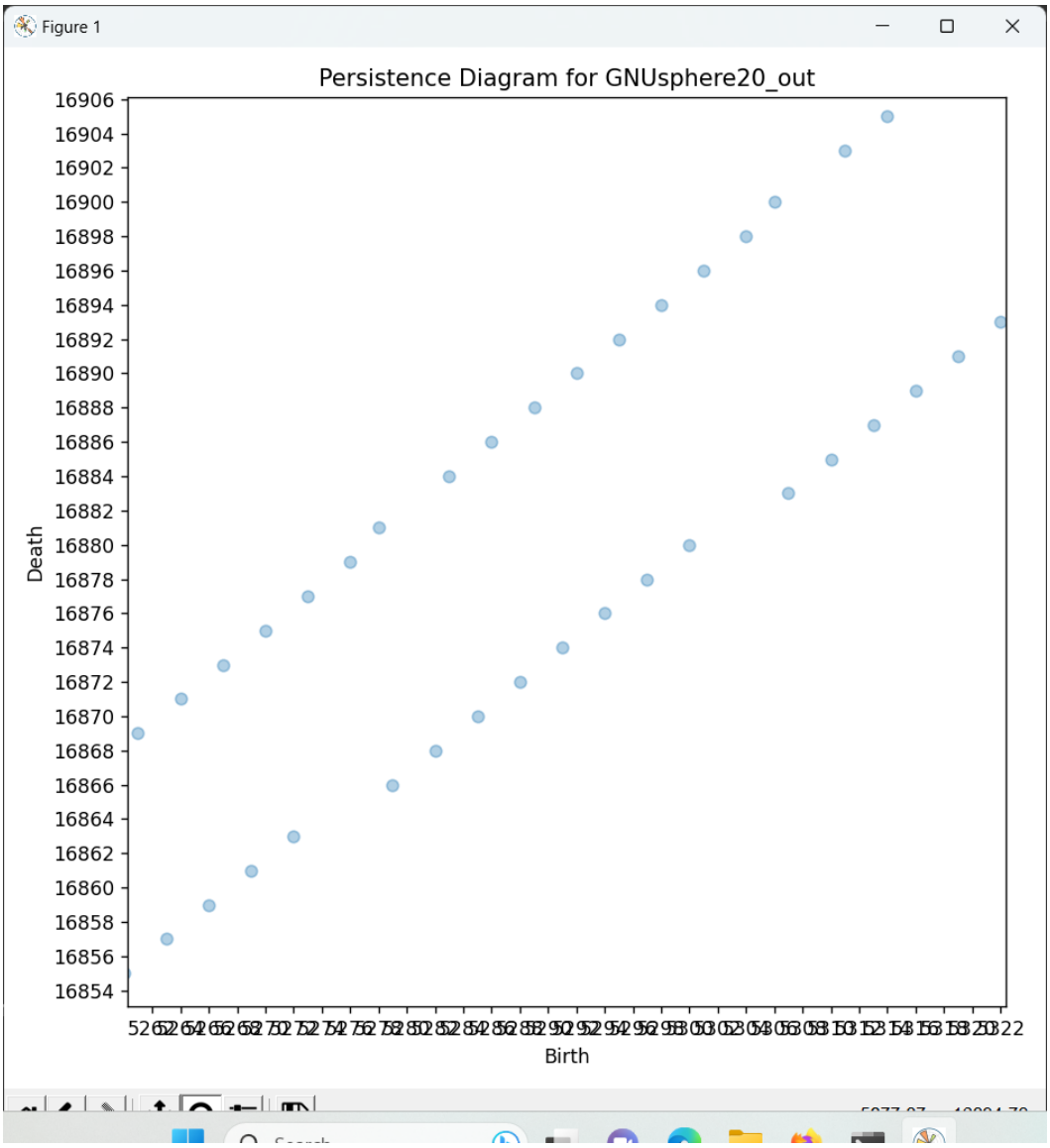


Figure 1
Persistence Diagram for GNUsphere20_out
Death
Birth
16906
16904
16902
16900
16898
16896
16894
16892
16890
16888
16886
16884
16882
16880
16878
16876
16874
16872
16870
16868
16866
16864
16862
16860
16858
16856
16854