

LTCC - INTRODUCTION TO APPLIED TOPOLOGY PROBLEMS

The idea behind these problems is to explore some of the concepts in the lecture more deeply or fill in the details of various proofs.

1.) Torsion in homology – In lecture, we use \mathbb{Z}_2 -coefficients. In this problem we will investigate other coefficients and see the effect of torsion. We will compute the homology of the projective plane over \mathbb{Z}_2 and \mathbb{Z}_3 . Recall \mathbb{Z}_3 is the integers modulo 3 (so for example, $2 = -1$).

First, we define the boundary operator. Put a complete ordering on the vertices, $v_i < v_j$ if $i < j$. This ordering is completely arbitrary. Then if σ is a k -simplex,

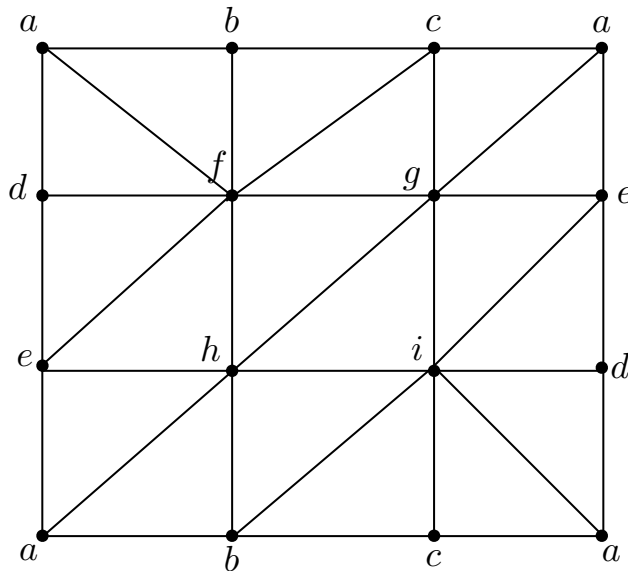
$$\sigma = [v_0, v_1, \dots, v_k]$$

then

$$\partial(\sigma) = \sum_{i=0}^k (-1)^i [v_0, \dots, \hat{v}_i, \dots, v_k]$$

where \hat{v}_i is omitted. Our space will be the Klein bottle (a non-orientable surface). A simplicial model is shown below:

Compute β_1 using \mathbb{Z}_2 and \mathbb{Z}_3 - what do you get? (Note: this is quite a large example - it has 54 simplicies - you can use the fact that it is connected to only have to do Gaussian elimination on the boundary operator for the triangles). Note that this can also be done by a program – as it is a long computation. Alternatively, it can be done by a software package – examples include Dionysus, GAP, GUDHi, among others.



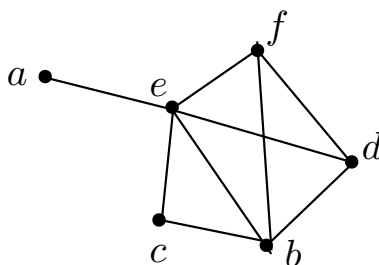
2.) Expected Euler characteristic – here we will warm up for the applications we will discuss in the later weeks. We will define two simple models of random simplicial complexes and compute their expected Euler characteristic. We can do this by using linearity of expectation and taking the alternating sum of the expected number of k -simplices.

First we describe two models.

- (1) Model 1: Linial-Meshulam model. Say we have n points and all possible $k - 1$ -simplices, that is $\binom{n}{k}$ possible simplices. This is called the complete $k - 1$ -skeleton. Now we place a k -simplex in with probability p (for each possible k -simplex, we do a coin toss to decide to place it into the complex). What is the expected Euler characteristic? First compute the Euler characteristic of the $k - 1$ -skeleton, then compute the Euler characteristic of the resulting complex.
- (2) Model 2: Now we will compute the clique complex. Again, assuming n points, we consider all $\binom{n}{2}$ edges and place them into the complex with probability p as above. This is also known as an Erdos-Renyi graph. Now we build the complex up iteratively. If edges of a triangle are in the graph, we place it in the complex. If all the triangles on the boundary of a tetrahedron are in the complex, we add it to the complex. More generally, if we have a k -clique (a complete graph on k -vertices), we add the $(k - 1)$ -simplex (and all faces). Notice that the number of k -simplices is still a function of independent random variables. Try and compute the expected Euler characteristic.

Interestingly, trying to anything other than asymptotic results for Betti numbers is difficult to impossible.

3.) Computing persistent homology – Here we will compute some (small) example diagrams through the different methods. Consider the simplicial complex below (triangles are not drawn).



The filtration values are as follows:

- $t = 1$ b, c, e
- $t = 2$ a, f, d, bc, ae
- $t = 3$ bf, ce, ef
- $t = 4$ be, bd, cbe, df
- $t = 5$ de, bed, bef

Now we will compute the persistence barcode/diagram using two different methods.

- (1) Standard algorithm: write out the boundary matrix as we did in lecture, ordering the simplices from small filtration value to large values (left-to-right, top-to-bottom). When there is a tie, break the tie arbitrarily (Note: Here you are free to choose \mathbb{Z}_2 or general coefficients). Process the columns left to right. Look at the lowest non-zero entry. If lowest row does not contain a pivot, set the lowest entry as a pivot. If the lowest row contains a pivot in the previous columns, then add the pivot column to the current column and repeat. Either the column gets a pivot or is completely zeroed out (in which case it is a cycle). When going to the next dimension, mark the simplices which are cycles and erase the other rows. The pivots represent the bars (row filtration value is the birth time and the column filtration value is the death time). To see the algorithm see page 9 of Computing Persistent Homology.

- (2) Draw out all the possible bars. Here assume that we have closed interval (if something dies at time x , is is still alive at x). Use the above computed barcode to compute the rank function (rank of the image). Verify that inclusion-exclusion returns the correct result.

4.) Experimentation: This is just a warmup to see how more general persistence diagrams/barcodes look like. Go to <https://live.ripser.org/> (you can also download it - there is a standalone version <https://github.com/Ripser/ripser> and python version <https://pypi.org/project/ripser/>).

Do not set the maximum parameter in the live version as this may crash the site. Both compiling the code or using python has been tested. In the latter case you can use `loadtxt` from NumPy to read in the files.

Compute the diagrams on three text files which are on website - `circle.txt`, `noisycircle.txt`, and `noise.txt`. What are the differences you notice?