CS 6170 Computational Topology: Topological Data Analysis **University of Utah**
Spring 2017 **School of Computing**

Lecture 15: Feb 28, 2017

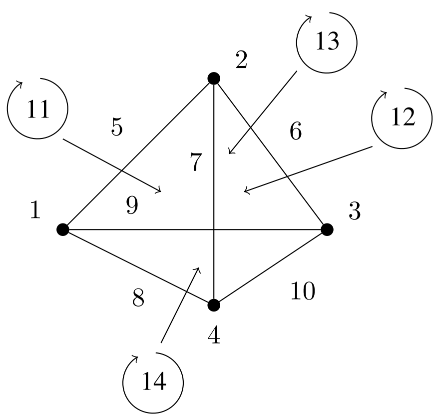
Lecturer: Prof. Bei Wang `<beiwang@sci.utah.edu>` *Scribe: Paul English*

15.1 TDA Software Links

- TDA R Package: `https://cran.r-project.org/web/packages/TDA/index.html`
- PHAT (Persistent Homology Algorithm Toolbox): `https://bitbucket.org/phat-code/phat`
- DIPHA (A Distributed Persistent Homology Algorithm): `https://github.com/DIPHA/dipha`

15.2 Example Persistent Homology Reduction

For the following simplicial complex,



we have the boundary matrix ∂ as follows,

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1					1			1	1					
2					1	1	1							
3						1			1	1				
4							1	1		1				
5											1		1	
6												1	1	
7											1	1		
8											1			1
9													1	1
10												1		1
11														
12														
13														
14														

Algorithm 1 Persistent Homology Reduction

procedure PERSISTEND HOMOLOGY REDUCTION
    $R \leftarrow \partial$
    for $j = 1$ to m **do**
        while there exists $j_0 < j$ with $\text{low}(j_0) = \text{low}(j)$ **do**
            add column j_0 to j

This is reduced using the following steps from the persistent homology routine in algorithm 1,

1. From left to right, we find the low values for column 5, 6, and 7.
2. At column 8, we add columns 7 and then 5, to arrive at a zero column.
3. At column 9, we add columns 6 and 5, to arrive at a zero column.
4. At column 10, we add columns 7 and 6, to arrive at a zero column.
5. For columns 11, 12, and 13 we find unique low values.
6. Lastly column 14, is set to zero through the column operations adding 12, 13, and 11.

The end result when traced looks something like the following,

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1					1			~~1~~	~~1~~					
2					**1**	1	1	~~1~~	~~1~~	~~1~~				
3						**1**			~~1~~	~~1~~				
4							**1**	~~1~~		~~1~~				
5											1		1	~~1~~
6												1	1	~~1~~
7											1	1		~~1~~
8											**1**			~~1~~
9													**1**	~~1~~
10												**1**		~~1~~
11														
12														
13														
14														

From this reduced form we have the following birth and death times:

$$\{(1,\infty),(2,5),(3,6),(4,7),(8,11),(10,12),(9,13),(14,\infty)\}.$$

- Note: Some papers refere to low(j) as pivot(j).
- The general worst case bounds for this algorithm is $O(n^3)$, for n the number of simplices (e.g. n = 14)

15.3 Positive and Negative Columns

When adding a simplex 1 of 2 things happen. For dimension p and simplex σ_i,

$$(k_{i-1})_d + \sigma_i \begin{cases} \to \beta_{p-1} - 1 & \text{(negative simplex)} \\ \to \beta_p + 1 & \text{(positive sample)} \end{cases}$$

For example in our boundary matrix, we have the following positive and negative columns,

5	6	7	8	9	10	11	12	13	14
-	-	-	+	+	+	-	-	-	+

The column operations here represent the cycles created.

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ \vdots \end{array} \begin{array}{c} \begin{array}{cccccc} \dots & 5 & 6 & 7 & 8 & \dots \end{array} \\ \begin{bmatrix} & 1 & & & 1 & \\ & 1 & 1 & 1 & & \\ & & 1 & & & \\ & & & 1 & 1 & \\ & & & & & \ddots \end{bmatrix} \end{array}$$

The pivot columns before 8 represent a basis for all combinations of cycles with 8

In general we have two cases. For ∂ a boundary matrix,

1. let ∂_j be a non-zero column with i = low(∂_j) then ∂_i is positive and inessential,
2. let ∂_i be a negative column then i is not low for any column in ∂.

15.4 Clearing Operation

We can improve the runtime of the algorithm using a clearing technique and the concept of positive and negative columns. Following the initial reduction algorithm, if you find a pivot, zero out the following row values, left to right.

- Paper: Persistent homology computation with a twist
- δ_j: Simplex dimension, e.g. $0 \to \{1,2,3,4\}$, $1 \to \{5,6,7,8,9,10\}$, $2 \to \{11,12,13\}$, $4 \to \{14\}$.
- σ_i: Columns, $1,\dots,14$

Algorithm 2 Decreasing Dimension Persistence Computation

procedure TWIST
  $R \leftarrow \sigma_j$
  $L \leftarrow [0, \ldots, 0]_{1\times n}$
  for $\delta = d, d-1, \ldots, 1$ **do**
    for $j = 1, 2, \ldots, n$ **do**
      [reduce R_j]
    if $R_j \neq 0$ **then**
      $L[\text{low}(R_j)] \leftarrow j$
      $R[\text{low}(R_j)] \leftarrow 0$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1					1			~~1~~	~~1~~					
2					1	1	1							
3						1			~~1~~	~~1~~				
4							1	~~1~~		~~1~~				
5											1		1	~~1~~
6												1	1	~~1~~
7											1	1		~~1~~
8											1			~~1~~
9													1	~~1~~
10												1		~~1~~
11														
12														
13														
14														

In general tracing this algorithm on this matrix results in the following,

1. From right to left, we add 12, 13, 11 to column 14, since it has low values to the left after each operation. This zero's out this column.
2. Now in columns 13, 12, and 11 we have the unique low values.
3. We cancel columns 9, 10, 8 since there are births shown by column 13, 12, 11.
4. We then have unique low values for 7, 6, and 5.

- Note: we can store ∂ as a sparse matrix, saving space.
- Next: Spectral Decomposition, which works with the diagonals to reduce the matrix.

References

[1] Herbert Edelsbrunner and John Harer. COMPUTATIONAL TOPOLOGY AN INTRODUCTION.

[2] Chao Chen and Michael Kerber. Persistent Homology Computation with a Twist. 2011.