# AMAT 584 Lecture 36, 4/24/20

**Today:** The "standard" algorithm for computing persistent homology.

A version of this algorithm appeared in the paper "Topological persistence and simplification" by Edelsbrunner et al. in 2000.

A more general version appears in "Computing Persistent Homology" by Carlsson and Zomorodian in 2005.

State-of-the-art persistent homology algorithms are optimized variants of this one.

**Input:** A simplicial filtration $F$, represented as matrix (I'll explain this below).

**Output:** The barcodes $\mathrm{Barc}(H_i(F))$.

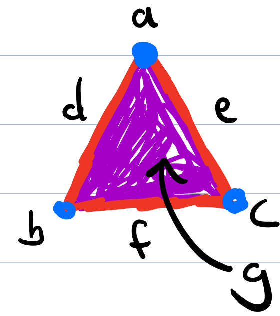
**Assumptions on $F$:**

- $F$ is indexed by $\mathbb{N}$ (but the $[0,\infty)$-indexed case works similarly)
- Each $F_z$ is a finite simplicial complex.
- There is some $y \in \mathbb{N}$ such that $F_y = F_z$ for all $z \geq y$.

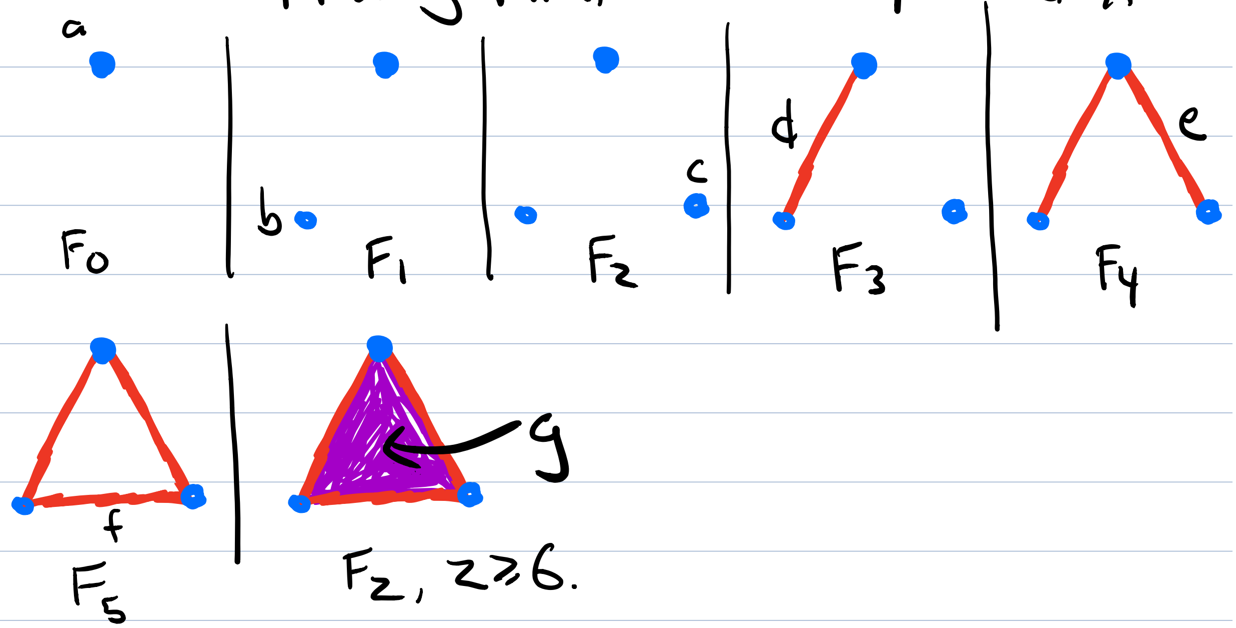
Let us write $F_4$ as $F_{max}$.

For $\sigma \in F_{max}$ let $\text{birth}(\sigma) = \min\{z \mid \sigma \in F_z\}$.

<u>Example</u>: Let $F_{max}$ be the following simplicial complex:




Consider the following filtration of subcomplexes of $X$:




$F_{max} = F_6$.

$\text{birth}(a) = 0$ $\quad$ $\text{birth}(d) = 3$ $\quad$ $\text{birth}(g) = 6$.

$\text{birth}(b) = 1$ $\quad$ $\text{birth}(e) = 4$

$\text{birth}(c) = 2$ $\quad$ $\text{birth}(f) = 5$

Recall: For $j \geq 0$, $F^j_{max}$ denotes the set of $j$-simplices of $F_{max}$.

We assume each $F^j_{max}$ is ordered so that if $\sigma, \tau \in F^j_{max}$ and $\text{birth}(\sigma) < \text{birth}(\tau)$, then $\sigma < \tau$.

In the example above, the alphabetical order on each $F^j_{max}$ satisfies this property.

Given these orderings, we can represent each boundary map $\delta_j : C_j(F_{max}) \to C_{j-1}(F_{max})$ as a matrix $[\delta_j]$ of dimensions $|F^{j-1}_{max}| \times |F^j_{max}|$, as we saw in recent lectures.

In our example

$$[\delta_1] = \begin{array}{c} \\ a \\ b \\ c \end{array} \begin{array}{c} \begin{array}{ccc} d & e & f \end{array} \\ \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \end{array}$$

$$[\delta_2] = \begin{array}{c} \\ d \\ e \\ f \end{array} \begin{array}{c} g \\ \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \end{array}$$

We place the matrices $[\delta_1], [\delta_2], \ldots, [\delta_{\dim(F_{max})}]$ into a block matrix $D$, given as follows:

$$D = \begin{pmatrix} 0 & [\delta_1] & 0 & 0 & & & 0 \\ 0 & 0 & [\delta_2] & 0 & \cdots & & 0 \\ 0 & 0 & 0 & [\delta_3] & & & 0 \\ & & \vdots & & \ddots & & \vdots \\ 0 & 0 & 0 & 0 & \cdots & & [\delta_{\dim(F_{max})}] \\ 0 & 0 & 0 & 0 & \cdots & & 0 \end{pmatrix}$$

Note that the non-zero blocks are all just above the diagonal

In our example, $\dim(F_{max}) = 2$ and

$$D = \begin{pmatrix} 0 & [\delta_1] & 0 \\ 0 & 0 & [\delta_2] \\ 0 & 0 & 0 \end{pmatrix} = \begin{array}{c|ccc|ccc|c}
 & a & b & c & d & e & f & g \\
\hline
a & & & & 1 & 1 & 0 & \\
b & & 0 & & 1 & 0 & 1 & 0 \\
c & & & & 0 & 1 & 1 & \\
\hline
d & & & & & & & 1 \\
e & & 0 & & & 0 & & 1 \\
f & & & & & & & 1 \\
\hline
g & & 0 & & & 0 & & 0
\end{array}$$

Note that in $D$,

- each column corresponds to a simplex in $F_{max}$
- each row corresponds to a simplex in $F_{max}$.

So we can think of the columns and rows as being labeled by simplices.

To compute persistent homology, we do a variant of Gaussian elimination on the columns of $D$. [It is also possible to give a version which does row operations, but the column version is the standard one.]

<u>Definition:</u> The pivot of a non-zero column vector $\vec{v}$ is the largest index of a non-zero entry.
We denote this $\mathrm{piv}(\vec{v})$. We write $\mathrm{piv}(\vec{0}) = \mathrm{null}$

<u>Example:</u> $\mathrm{piv}\begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} = 2$. $\quad \mathrm{piv}\begin{pmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} = 4$

We say a matrix is reduced if no two non-zero columns have the same pivot.

The following algorithm converts any matrix into a reduced one, via left-to-right column operations.

## The standard reduction algorithm

Input: $m \times n$ matrix $D$, with $\mathbb{F}_2$ coefficients

Output: Reduced $m \times n$ matrix $R$, obtained from $D$ by $L \to R$ column additions

$R \leftarrow D$.

For $j = 1$ to $n$:

  while $\exists\ k < j$ such that $\text{null} \neq \text{piv}(R_{*,k}) = \text{piv}(R_{*,j})$

    add column $k$ to column $j$.

After reducing $D$ to obtain a matrix $R$, we read the barcodes $\text{Barc}(H_i(F))$ off of $R$, as follows:

$\text{Barc}(H_i(F)) =$

$\{[\text{birth}(\sigma), \text{birth}(\tau)) \mid \text{pivot of column } \tau \text{ in } R \text{ is } \sigma \text{ and } \dim(\sigma) = i\}$

$\{[\text{birth}(\sigma), \infty) \mid \text{col } \sigma = 0, \sigma \text{ is not the pivot of any column in } R, \dim(\sigma) = i\}$.

Note: Any interval the form $[z, z)$ is ignored.