Lecture 9: Matrix Reduction and Pairing

Topics in Computational Topology: An Algorithmic View

Scribed by: Dong Du

In Lecture 5, we introduce the concept of simplicial complex. In Lecture 6, we define the simplicial homology of simplicial complex. In Lecture 7, we provide method to compute simplicial homology by matrix. In Lecture 8, we define persistent homology of filtration of nested simplicial complexes and persistent diagram. In this lecture, we will make use of matrix to compute persistent homology.

1 Matrix Reduction

In order to construct the matrix which we will use to compute persistent homology, we need to order simplices properly. That is, we use a *compatible ordering* of the simplices, a sequence $\sigma_1, \sigma_2, \ldots, \sigma_m$ such that $i < j$ if $f(\sigma_i) < f(\sigma_j)$ or if σ_i is a face of σ_j. Such an ordering exists because f is monotonic. Note that every initial subsequence of simplices forms a subcomplex of K. We use this sequence when we set up the m-by-m boundary matrix, ∂, which stores the simplices of all dimensions in one place; that is,

$$\partial[i,j] = \begin{cases} 1 & \text{if } \sigma_i \text{ is a codimension-1 face of } \sigma_j \\ 0 & \text{otherwise.} \end{cases}$$

In words, the rows and columns are ordered like the simplices in the total ordering and the boundary of a simplex is recorded in its column. The algorithm uses column operations to reduce ∂ to another $0-1$ matrix R. Let $low(j)$ be the row index of the lowest 1 in column j. If the entire column is zero, then $low(j)$ is undefined. We call R *reduced* if $low(j) \neq low(j_0)$ whenever j and j_0, with $j \neq j_0$, specify two non-zero columns. The algorithm reduces ∂ by adding columns from left to right.

R $= \partial$;
for $j = 1 \to m$ **do**
  while there exists $j_0 < j$ with $low(j_0) = low(j)$ **do**
    add column j_0 to column j
  end while
end for

The running time is at most cubic in the number of simplices. In matrix notation, the algorithm computes the reduced matrix as $R = \partial \cdot V$; see Figure 1. Since each simplex is preceded by its proper faces, ∂ is upper triangular. The j-th column of V encodes the column in ∂ that add up to give the j-th column in R. Since we only add from left to right, V is also upper triangular and so is R.

To get the ranks of the homology groups of K, we notice that the number of zeros columns of R that correspond to p-simplices is the rank of Z_p. Similarly, the number of non-zero columns gives the rank of B_p. The difference is the p-th Betti number.

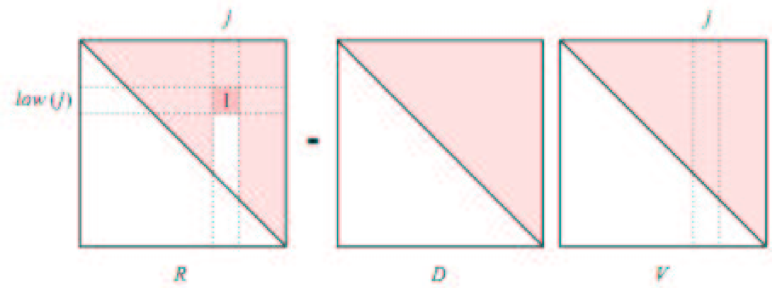

Figure 1: Reducing ∂ expressed as matrix multiplication. White areas are necessarily zero while entries in shaded areas can be either 0 or 1. (Picture courtesy of [1].)

2 Pairing

Although the reduced matrix R is not unique, the number $low(i)$ is unique for each column. To see that the lowest 1s are unique, we consider the lower left submatrix R_i^j of R whose corner element is $R[i,j]$. In other words, R_i^j is obtained from R by removing the first $i-1$ rows and the last $n-j$ columns. Since left-to-right column operations preserve the rank of every such submatrix, the rank of R_i^j is the same as that of the corresponding submatrix of ∂, the one similarly obtained by removing the first $i-1$ rows and the last $n-j$ columns. We consider the expression

$$r_R(i,j) = \text{rank} R_i^j - \text{rank} R_{i+1}^j + \text{rank} R_{i+1}^{j-1} - \text{rank} R_i^{j-1}$$

and note that $r_R(i,j) = r_\partial(i,j)$ for all i and j, where $r_\partial(i,j)$ has an analogous definition except when we take ranks of submatrices of ∂. To evaluate this expression, we observe that the linear combination of any collection of non-zero columns in R_i^j is again non-zero. It follows that the rank of R_i^j is equal to its number of non-zero columns. Now, if $R[i,j]$ is a lowest 1, then R_i^j has one more non-zeros column than the other three submatices, which implies $r_R(i,j) = 1$. If $R[i,j]$ is not a lowest 1, then we consider two subcases. If none of the columns from 1 to $j-1$ has its lowest 1 in row i, then R_i^j and R_{i+1}^j have the same number of non-zero columns and so do R_i^{j-1} and R_{i+1}^{j-1}. Second, if one of these columns has its lowest 1 in row i, then R_i^j has one more non-zero column than R_{i+1}^j and R_i^{j-1} has one more non-zero column than R_{i+1}^{j-1}. In either case, $r_R(i,j) = 0$. Since the ranks of the lower left submatrices of R are the same as those of ∂, we have a characterization of the lowest 1s that does not depend on the reduction process.

PAIRING LEMMA. We have $i = low(j)$ iff $r_\partial(i,j) = 1$. In particular, the pairing between rows and columns defined by the lowest 1s in the reduced matrix does not depend on R.

Note that column j reaches its final form at the end of the $j-$th iteration of the outer loop. At this moment, we have the reduced matrix for the complex consisting of the first j simplices in the total ordering. We distinguish the case in which column j ends up zero from the other in which it has a lowest 1.

Case 1: column j of R is zero. We call σ_j positive since its addition creates a new cycle and thus gives birth to a new homology class.

Case 2: column j of R is non-zero. It stores the boundary of the chain accumulated in column j of matrix V and is thus a cycle. We call σ_j negative because its addition gives death to a homology class.

The class that dies in Case 2 is represented by column j. We still need to verify that it is born at the time the simplex of its lowest 1, σ_i with $i = low(j)$, is added. But this is clear because the cycle in column j of R just died and all other cycles that die with it have 1s below row i; otherwise , we could further reduce the matrix and obtain $low(j) < i$, which contradicts s the algorithm. It follows that the lowest 1s indeed correspond to the points in the persistence diagrams. More precisely, (a_i, a_j) is a finite point in $Dgm_p(f)$ iff $i = low(j)$ and σ_i is a simplex of dimension p. In this case, σ_j is a simplex of dimension $p+1$. We have (a_i, ∞) in $Dgm_p(f)$ iff column i is zero but row i does not contain a lowest 1. In other words, σ_i is positive, but it does not get paired with a negative simplex.

References

[1] H. Edelsbrunner and J. Harer, Computational Topology: An Introduction, AMS, 2010.