Simplicial Complexes and Homology

Tao Hou, University of Oregon

Topological invariant

- Recall that a topological invariant is a type of characteristics for spaces that are preserved by topological equivalence (homeomorphism)
- We shall eventually look at the topological invariant called homology, which people heavily rely on in TDA
- But before looking at that, let's first we look at a simpler invariant called Euler characteristic

Euler characteristics

- Here we consider Polyhedron, which is a 3D object whose building blocks are
    - Polygonal faces (2d)
    - Edges (1d)
    - Vertices (0d)

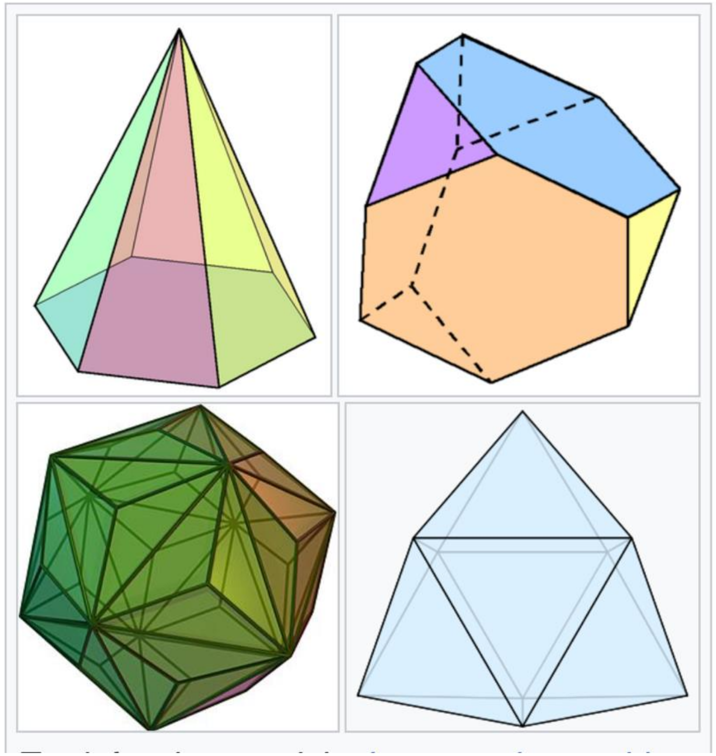
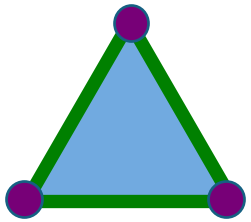


image source: Wikipedia

We wish to count:

vertex 

edge 

face 

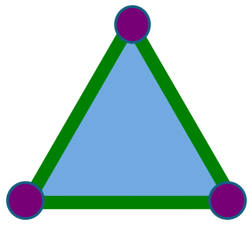
Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html

We wish to count:

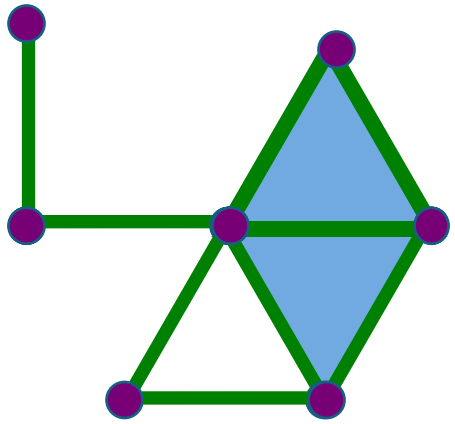
vertex

edge

face



Example:



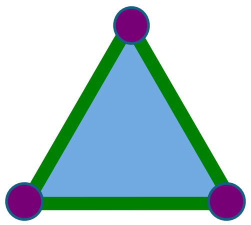
Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html

We wish to count:

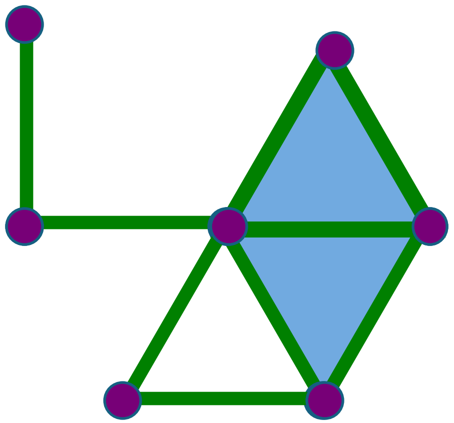
vertex

edge

face



Example:



7 vertices,
9 edges,
2 faces.

Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html

Euler characteristic (simple form):

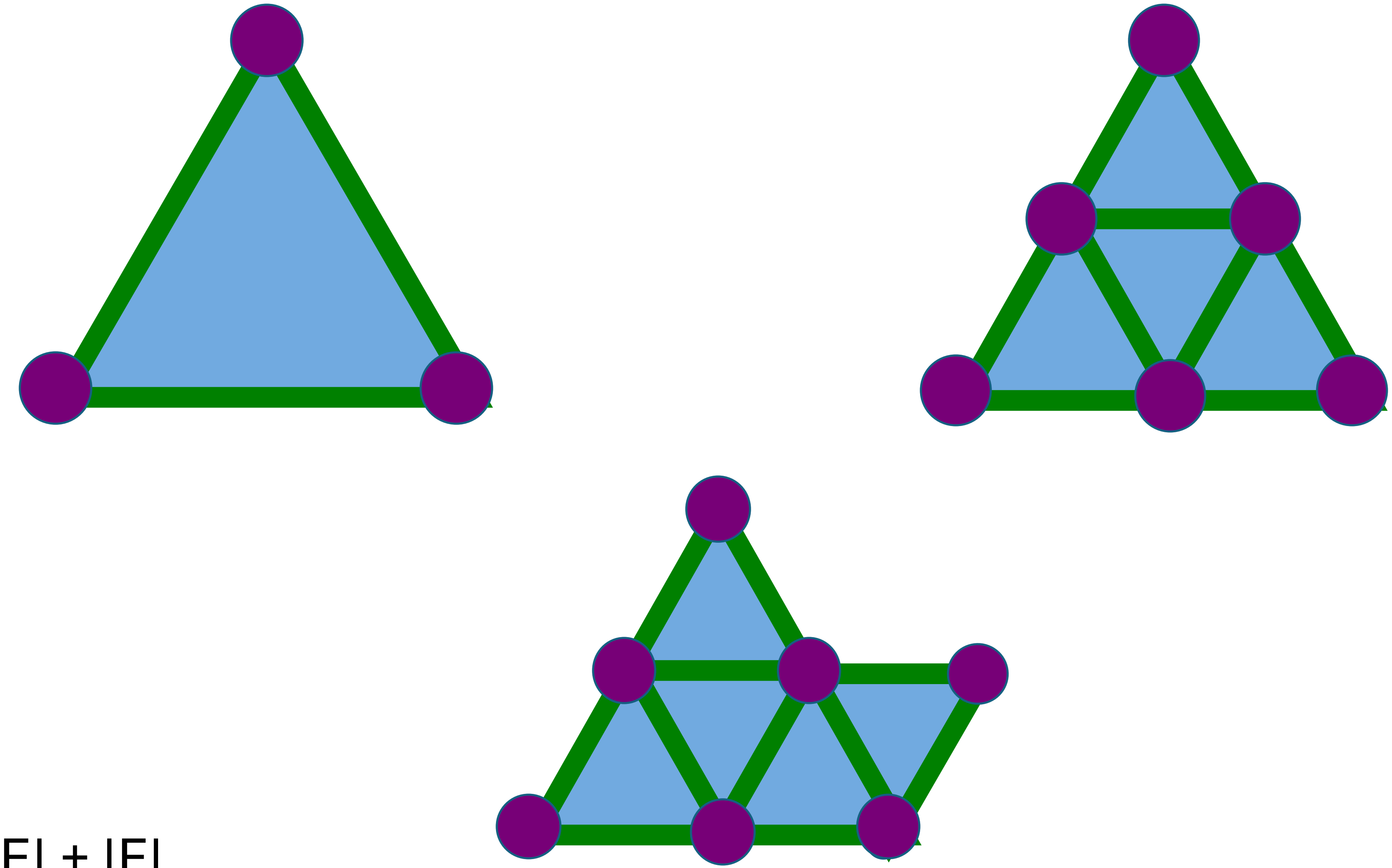
χ = number of vertices – number of edges + number of faces

Or in short-hand,

$$\chi = |V| - |E| + |F|$$

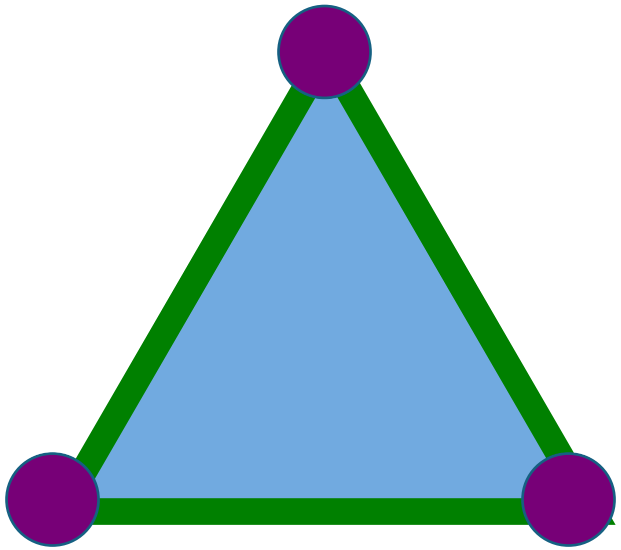
where V = set of vertices
E = set of edges
F = set of faces

Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html

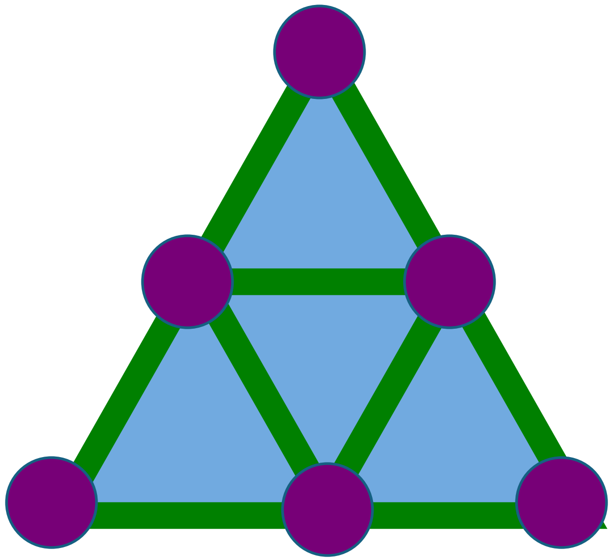


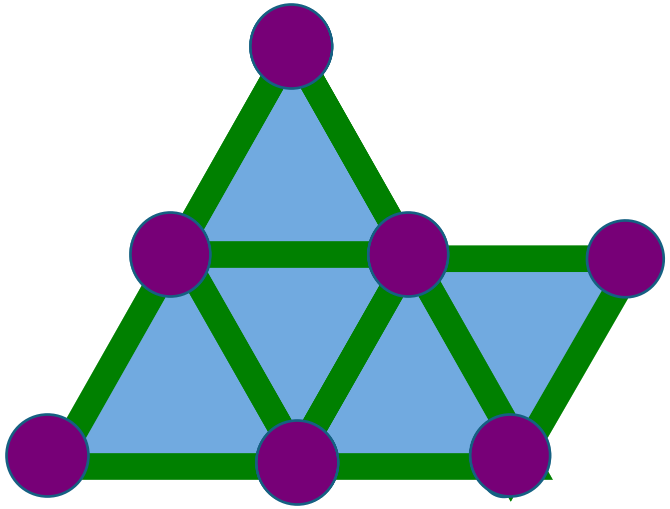
$\chi = |V| - |E| + |F|$

Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html



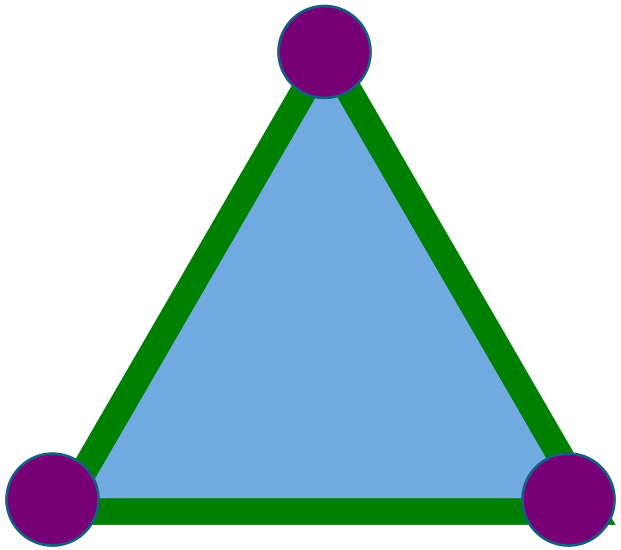
$\chi = 3 - 3 + 1 = 1$



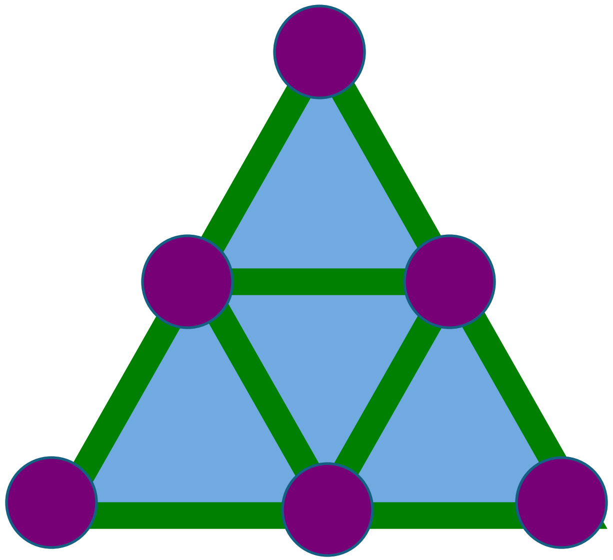


$\chi = |V| - |E| + |F|$

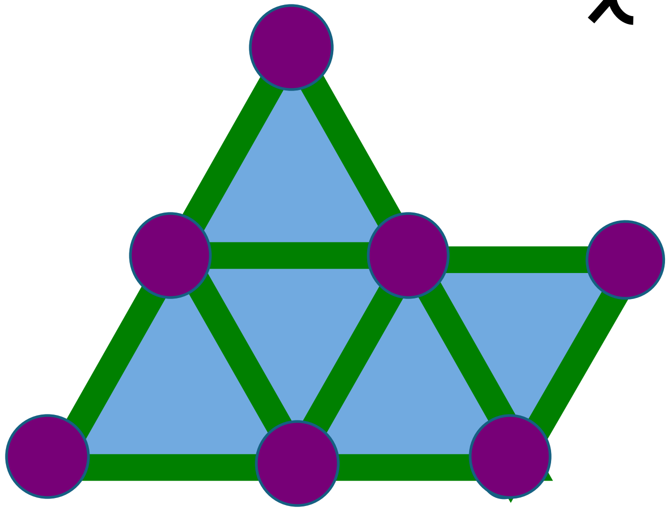
Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html



$\chi = 3 - 3 + 1 = 1$

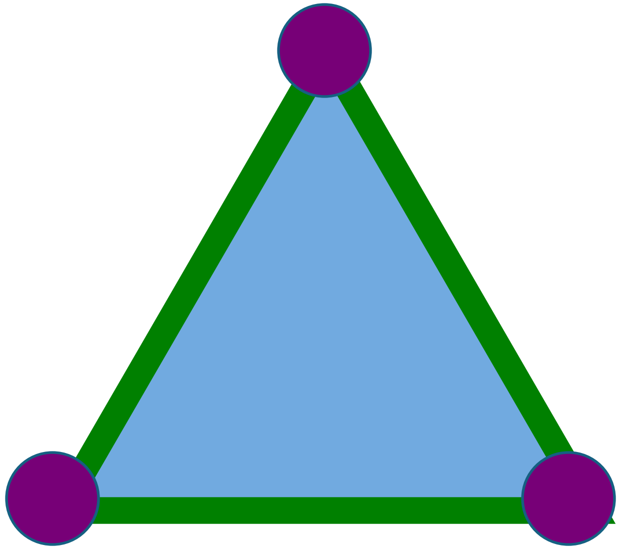


$\chi = 6 - 9 + 4 = 1$

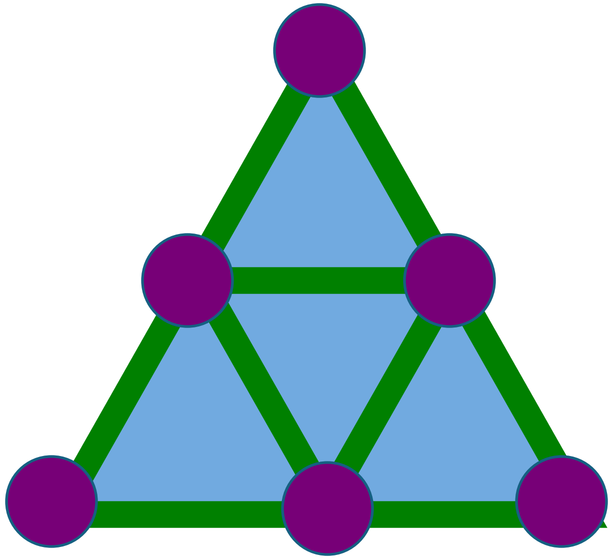


$\chi = |V| - |E| + |F|$

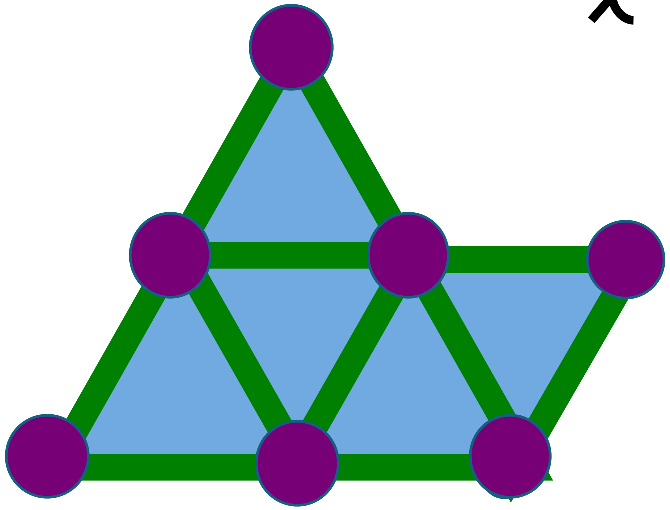
Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html



$\chi = 3 - 3 + 1 = 1$



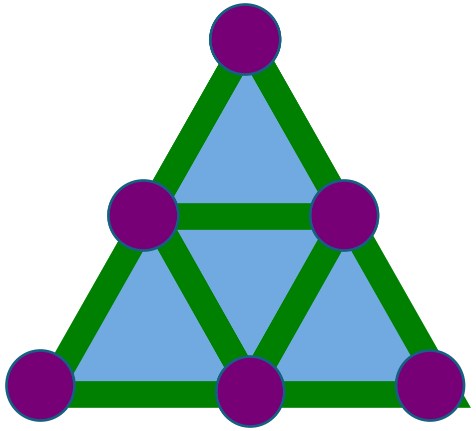
$\chi = 6 - 9 + 4 = 1$

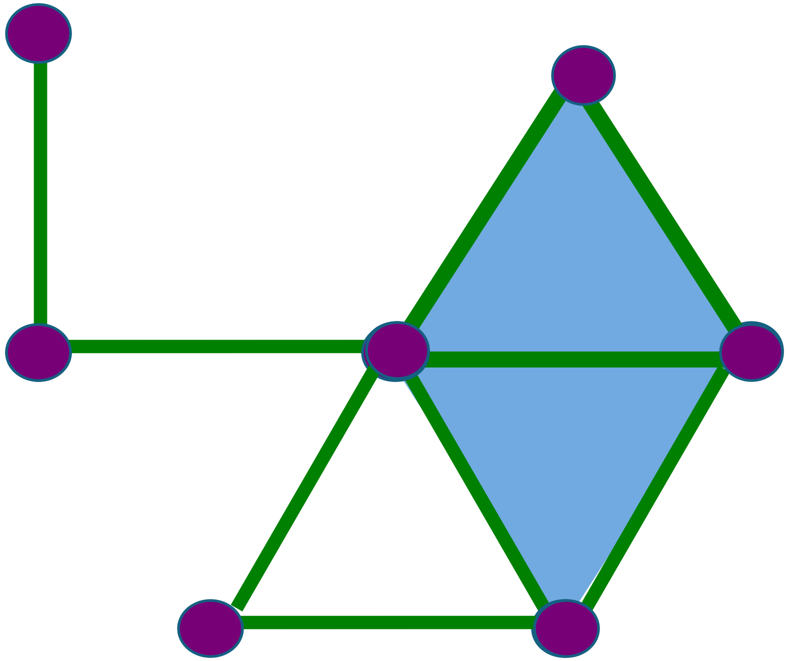


$\chi = 7 - 11 + 5 = 1$

$\chi = |V| - |E| + |F|$

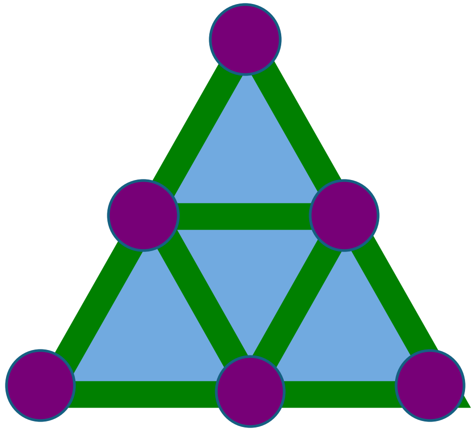
Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html



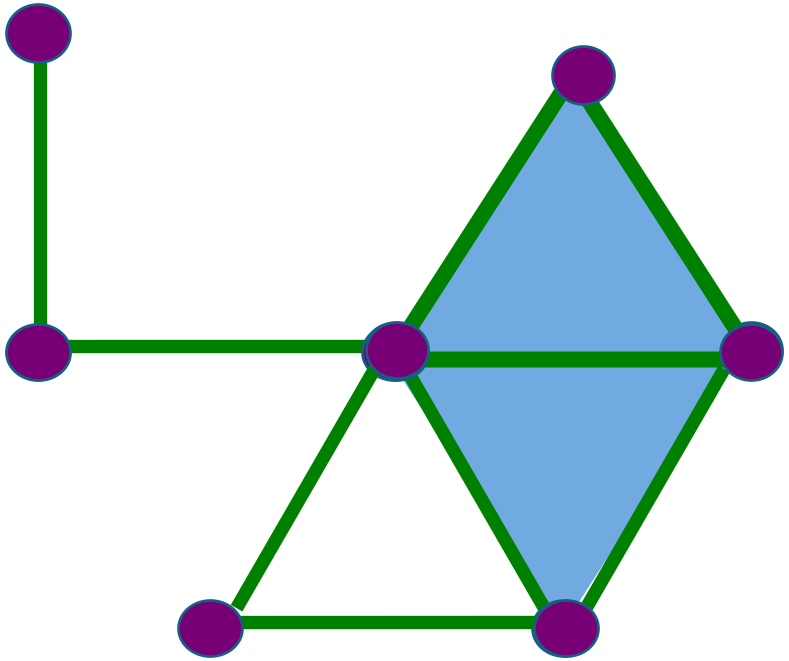


$\chi = |V| - |E| + |F|$

Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html

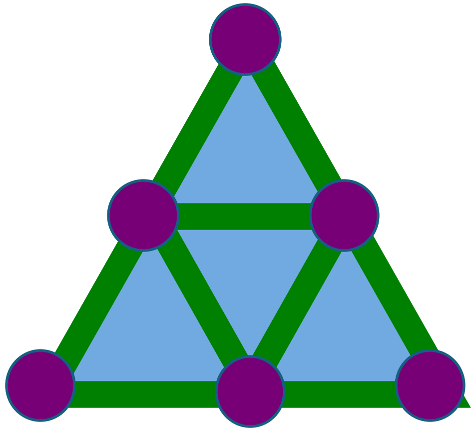


$\chi = 6 - 9 + 4 = 1$

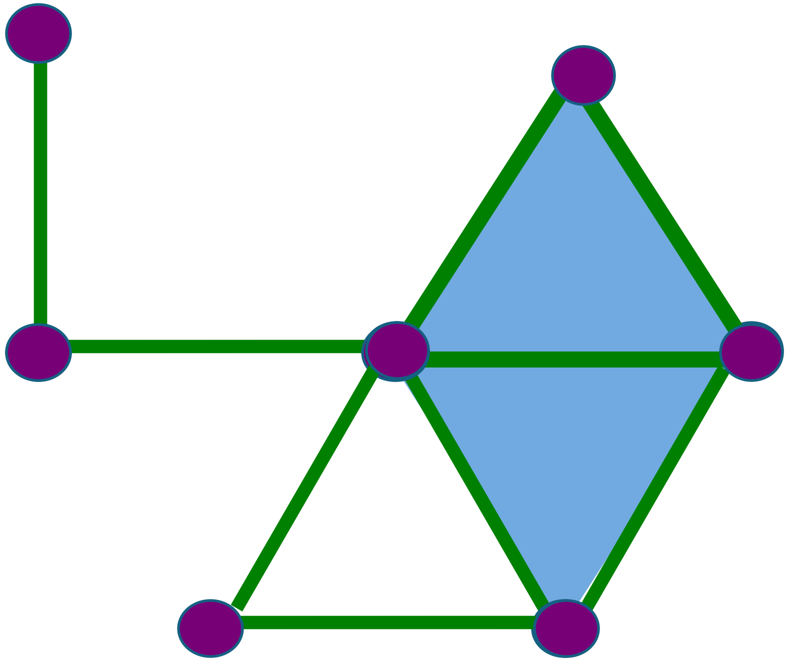


$\chi = |V| - |E| + |F|$

Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html



$\chi = 6 - 9 + 4 = 1$



$\chi = 7 - 9 + 2 = 0$

$\chi = |V| - |E| + |F|$

Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html

Conclusion

- Euler characteristics is a **topological invariant** for topological spaces,
  - meaning that for two topological spaces which are equivalent (homeomorphic), their Euler characteristics is the same.

More examples

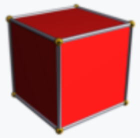
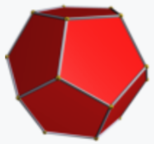
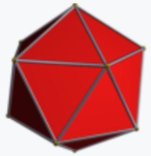
Name	Image	Vertices V	Edges E	Faces F	Euler characteristic: $\chi = V - E + F$
Tetrahedron					
Hexahedron or cube					
Octahedron					
Dodecahedron					
Icosahedron					

image source: Wikipedia

More examples

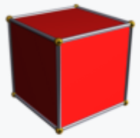
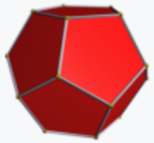
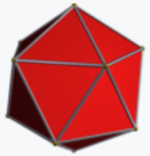
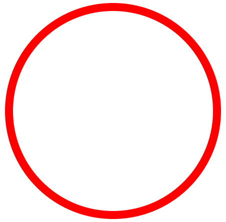
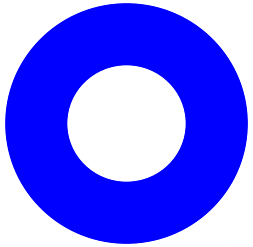
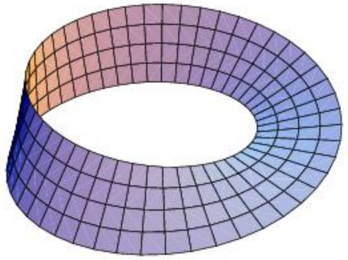
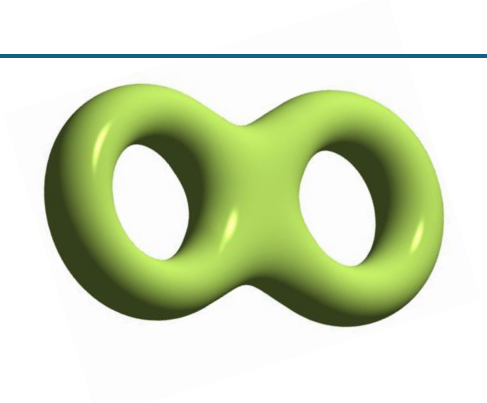
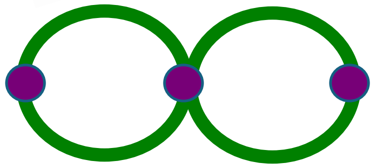
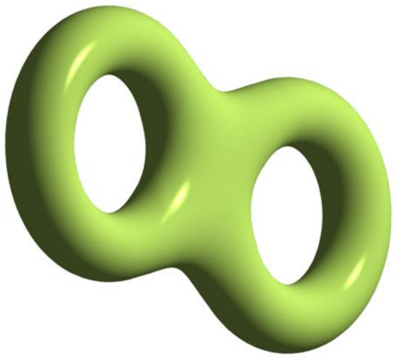
Name	Image	Vertices V	Edges E	Faces F	Euler characteristic: $\chi = V - E + F$
Tetrahedron		4	6	4	**2**
Hexahedron or cube		8	12	6	**2**
Octahedron		6	12	8	**2**
Dodecahedron		20	30	12	**2**
Icosahedron		12	30	20	**2**

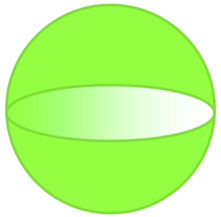
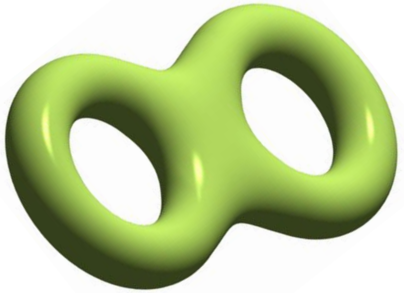
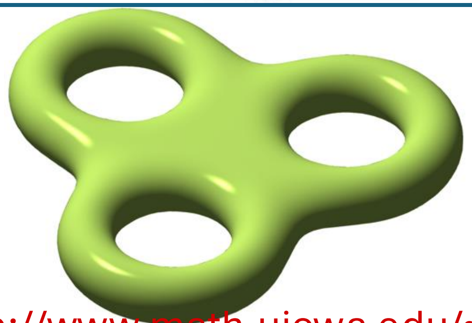
image source: Wikipedia

Euler characteristic	
0	circle 
	Annulus 
	Mobius band 
	Torus = $S^1 \times S^1$ 

Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html

Euler characteristic	
-1	Solid double torus  A graph of two cycles: 
-2	Double torus = boundary of solid double torus 

Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html

Euler characteristic	2-dimensional orientable surface without boundary
2	sphere 
0	S^1 x S^1 = torus 
-2	genus 2 torus 
-4	genus 3 torus 

Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html

Euler characteristics for graphs

- Graphs: consist of only vertices and edges
- So, Euler characteristic becomes: $V - E$

Euler characteristics for graphs

- Graphs: consist of only vertices and edges
- So, Euler characteristic becomes: $V - E$
- We can use Euler characteristic to verify whether a graph is a *tree*

Euler characteristics for graphs

- Graphs: consist of only vertices and edges
- So, Euler characteristic becomes: $V - E$
- We can use Euler characteristic to verify whether a graph is a *tree*
  - **Definition**: A *tree* is a connected graph that does not contain a cycle

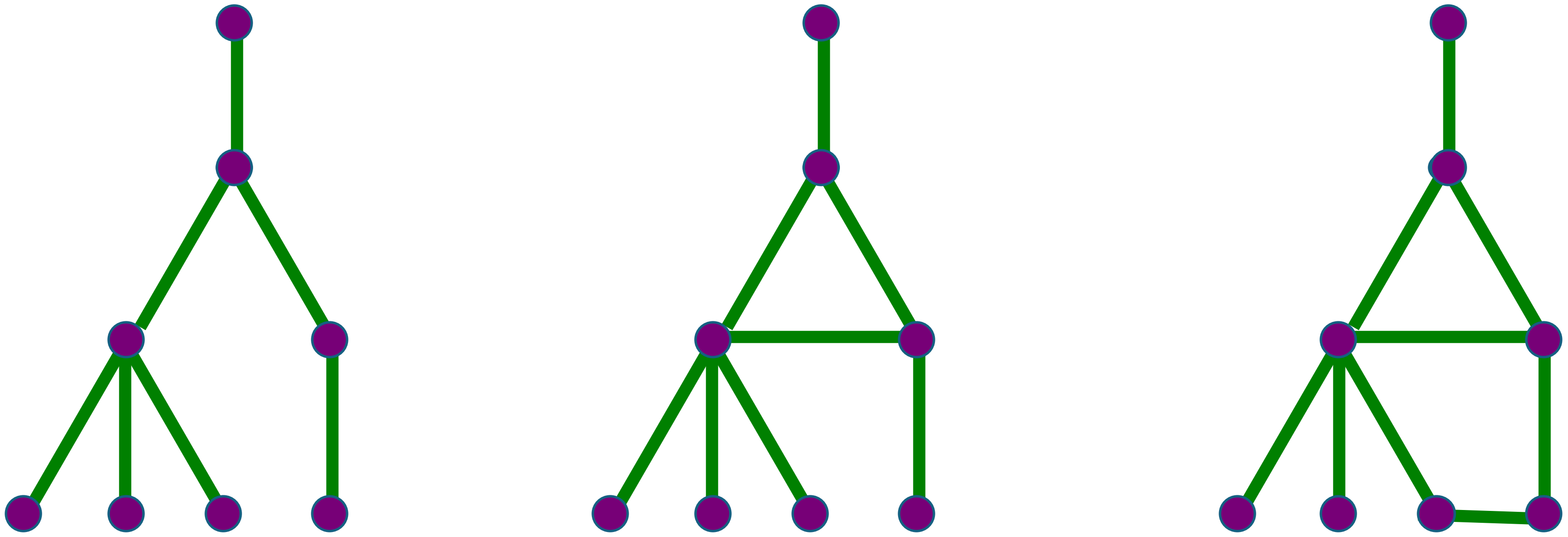
Euler characteristics for graphs

- Graphs: consist of only vertices and edges
- So, Euler characteristic becomes: $V - E$
- We can use Euler characteristic to verify whether a graph is a *tree*
  - **Definition**: A *tree* is a connected graph that does not contain a cycle
- **Theorem**: The number of edges in a tree is always equal to the number of vertices – 1. So the Euler characteristic becomes

$$V - E = V - (V - 1) = 1.$$

Graphs: Identifying Trees

Defn: A *tree* is a connected graph that does not contain a cycle



$\chi = 8 - 7 = 1$ $\chi = 8 - 8 = 0$ $\chi = 8 - 9 = -1$

Courtesy of Isabel K. Darcy: http://www.math.uiowa.edu/~idarcy/AppliedTopology.html

Representation of shapes

- Before moving on to look at the more important invariant, homology
- We need another important definition that will be utilized throughout the course
- This solves a fundamental problem we face when we try to process shapes in computer: we need a way to represent shapes (topological spaces) that is easy for computer to process
- It turns out there have been such an invention in Mathematics already, which is called **simplicial complex**.

Simplicial Complex

- A simplicial complex is a generalization of a polyhedron, with building blocks called ***simplices*** in different dimensions:
  - 0-simplex: vertex
  - 1-simplex: edge
  - 2-simplex: triangle
  - 3-simplex: tetrahedron
  - ...
  - d-simplex (generalizations)

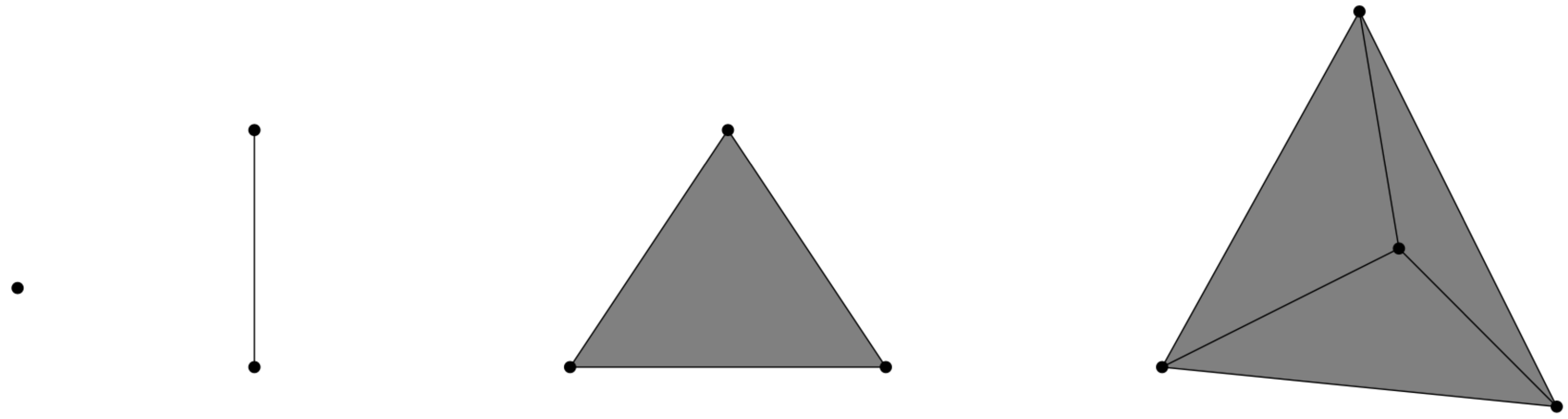


Image source: Patrick Schnider. Introduction to Topological Data Analysis Lecture Notes FS 2023

Simplicial Complex

- The following is a simplicial complex with simplices up to dimension 3:
  - 0-simplices (vertices): 18
  - 1-simplices (edges): 23
  - 2-simplices (triangles): 8
  - 3-simplices (tetrahedra): 1

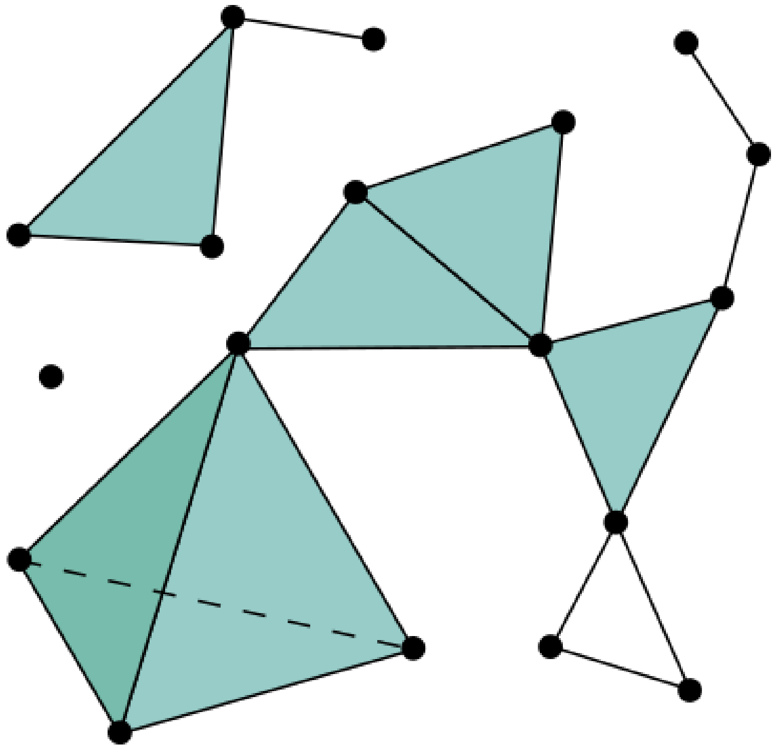


Image source: Wikipedia

Simplicial Complex

- The following is a simplicial complex with simplices up to dimension 3:
  - 0-simplices (vertices): 18
  - 1-simplices (edges): 23
  - 2-simplices (triangles): 8
  - 3-simplices (tetrahedra): 1
- **Definition**: The *dimension* of a simplicial complex is the maximum dimension of its simplices
- So the dimension of the left complex is 3
- Note: A simplicial complex is sometimes simply called a *complex*
- A d-dimensional simplicial complex is sometimes simply called a *simplicial d-complex* or *d-complex*

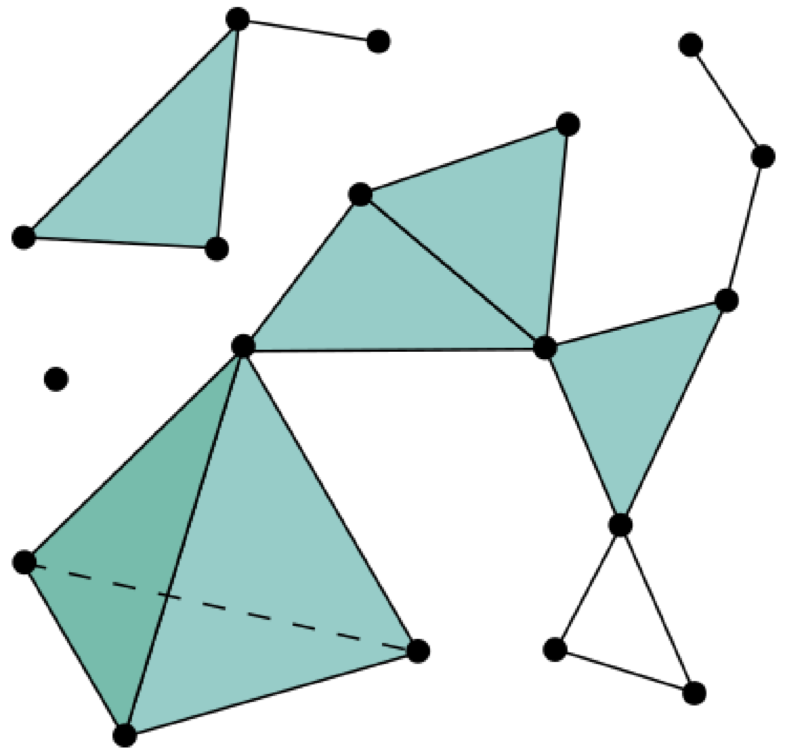
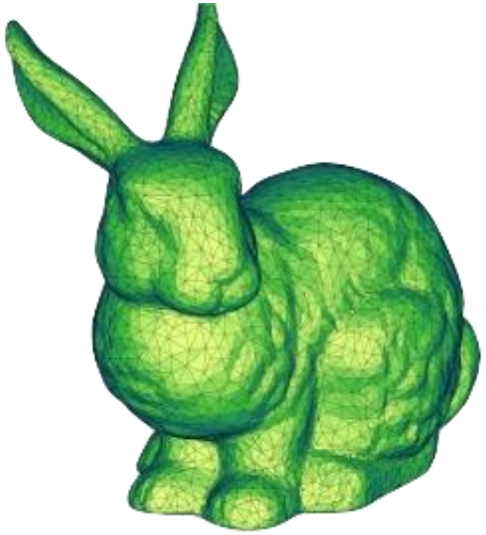


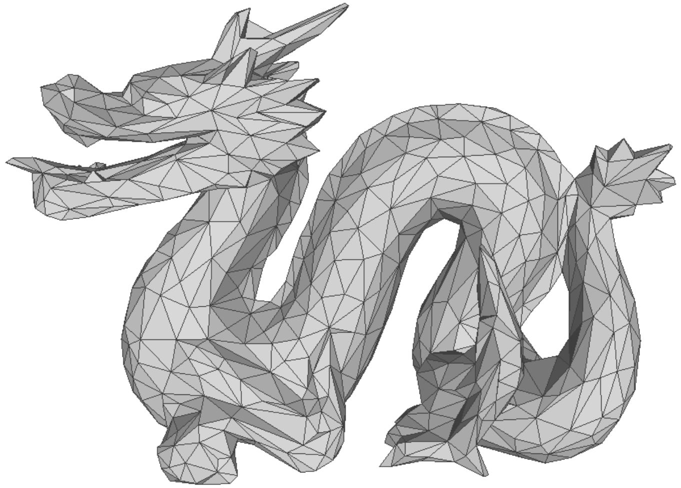
Image source: Wikipedia

Triangular meshes

- A very common type of simplicial complexes used in computer graphics are triangular meshes (a 3D object whose surface is made up of glueing small triangle patches)
- From a topological point of view, they are nothing but 2-dimensional simplicial complexes



(figure from favpng.com)



Graphs

- Another more common type of simplicial complexes in CS are graphs.
- A graph is a tuple $G = (V, E)$, where V is the set of 0-simplices and E is the set of 1-simplices. So it's a 1-complex.

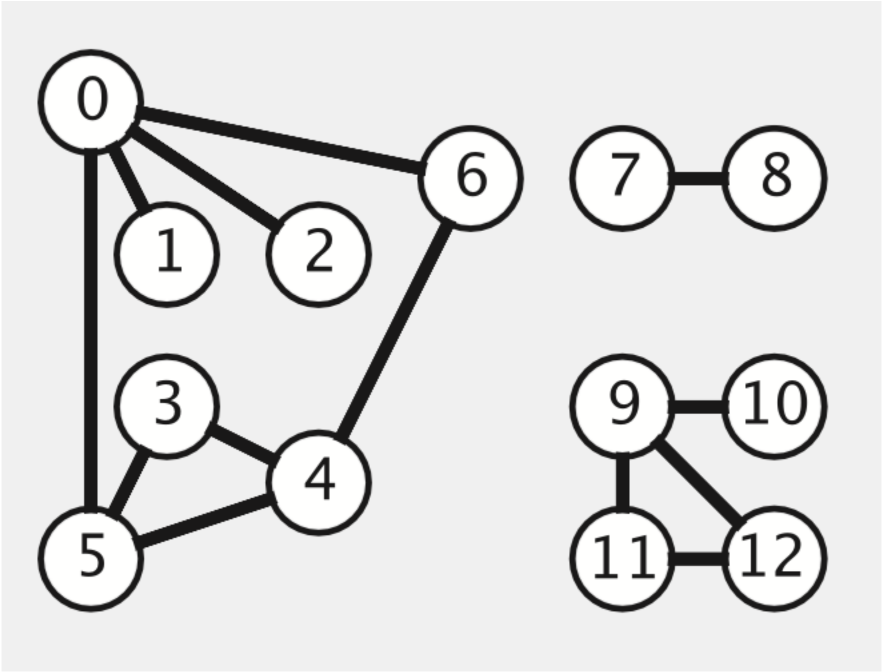


Image source: Sedgewick and Wayne. Algorithms (4th ed)

Faces of a Simplex

- For a simplex σ, we notice that there are other simplices on its boundary, which are called the **faces** of σ.
- If a face τ of σ is a d-dimensional simplex, then we also call it a d-face of σ.
- A convention is that σ is always a face of itself.

Faces of a Simplex

- For a simplex σ, we notice that there are other simplices on its boundary, which are called the **faces** of σ.
- If a face τ of σ is a d-dimensional simplex, then we also call it a d-face of σ.
- A convention is that σ is always a face of itself.
- Ex: A vertex has only one face which is itself

Faces of a Simplex

- For a simplex σ, we notice that there are other simplices on its boundary, which are called the **faces** of σ.
- If a face τ of σ is a d-dimensional simplex, then we also call it a d-face of σ.
- A convention is that σ is always a face of itself.
- Ex: An edge ab has:
  - Two 0-faces: a and b
  - One 1-face: ab

Faces of a Simplex

- For a simplex σ, we notice that there are other simplices on its boundary, which are called the **faces** of σ.
- If a face τ of σ is a d-dimensional simplex, then we also call it a d-face of σ.
- A convention is that σ is always a face of itself.
- Ex: A triangle abc has:
  - Three 0-faces: a, b, and c
  - Three 1-faces: ab, ac, and bc
  - One 2-face: abc

Faces of a Simplex

- For a simplex σ, we notice that there are other simplices on its boundary, which are called the **faces** of σ.
- If a face τ of σ is a d-dimensional simplex, then we also call it a d-face of σ.
- A convention is that σ is always a face of itself.
- Ex: A tetrahedron $abcd$ has:
  - Four 0-faces: a, b, c, and d
  - Six 1-faces: ab, ac, ad, bc, bd, cd
  - Four 2-faces: abc, abd, acd, bcd
  - One 3-face: $abcd$

Simplicial Complex (Formal Definition)

- **Definition:** a simplicial complex $\mathcal{K}$ is a set of simplices such that, if a simplex σ is in $\mathcal{K}$, then all the faces of σ are also in $\mathcal{K}$.

Simplicial Complex (Formal Definition)

- **Definition**: a simplicial complex $\mathcal{K}$ is a set of simplices such that, if a simplex σ is in $\mathcal{K}$, then all the faces of σ are also in $\mathcal{K}$.

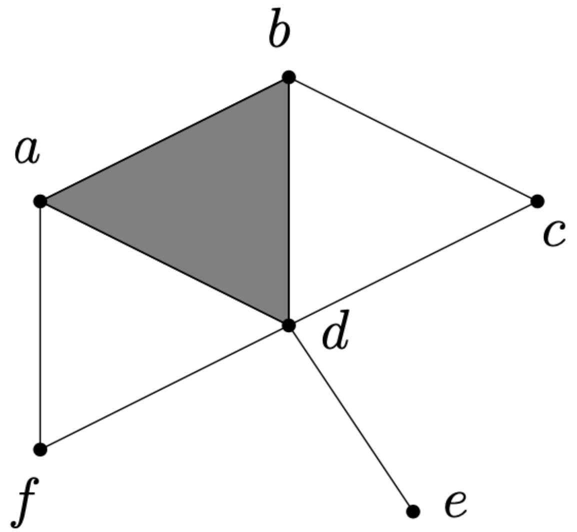
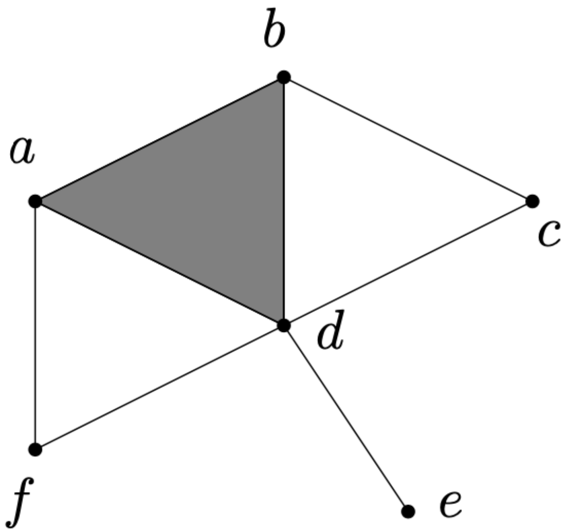


Image source: T.K.Dey and Y.Wang. Computational Topology for Data Analysis

Simplicial Complex (Formal Definition)

- **Definition:** a simplicial complex $\mathcal{K}$ is a set of simplices such that, if a simplex σ is in $\mathcal{K}$, then all the faces of σ are also in $\mathcal{K}$.

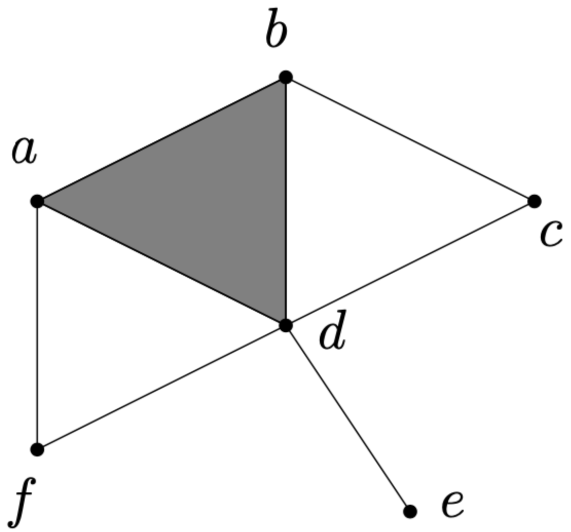


✔: {abd, ab, bd, ad, bc, cd, af, dd, de, a, b, c, d, e, f}

Image source: T.K.Dey and Y.Wang. Computational Topology for Data Analysis

Simplicial Complex (Formal Definition)

- **Definition:** a simplicial complex $\mathcal{K}$ is a set of simplices such that, if a simplex σ is in $\mathcal{K}$, then all the faces of σ are also in $\mathcal{K}$.



✔: {abd, ab, bd, ad, bc, cd, af, dd, de, a, b, c, d, e, f}

✘: {abd, ~~ab~~, bd, ad, bc, cd, af, dd, de, ~~a~~, b, c, d, e, f}

Image source: T.K.Dey and Y.Wang. Computational Topology for Data Analysis

Simplicial Complex (Formal Definition)

- **Definition**: a simplicial complex $\mathcal{K}$ is a set of simplices such that, if a simplex σ is in $\mathcal{K}$, then all the faces of σ are also in $\mathcal{K}$.
- Another example: For a graph (1-complex), each edge must join two vertices in the vertex set (a vertex that is a face of an edge must also be in the complex)

Image source: T.K.Dey and Y.Wang. Computational Topology for Data Analysis

Simplicial Complex (Formal Definition)

- **Definition**: a simplicial complex $\mathcal{K}$ is a set of simplices such that, if a simplex σ is in $\mathcal{K}$, then all the faces of σ are also in $\mathcal{K}$.
- Another example: For a graph (1-complex), each edge must join two vertices in the vertex set (a vertex that is a face of an edge must also be in the complex)

The condition that faces of any simplex in a complex should also be in the complex is very important part in the definition making it mathematically sound

Image source: T.K.Dey and Y.Wang. Computational Topology for Data Analysis

Some remarks

- For denoting a simplicial complex, we typically first assign labels to the vertices of the complex.
    - In class, the labels could be letters but could also be other things
    - In computer programs, the labels are almost always integers $0, \cdots, l-1$.
- Then, each simplex is represented as a set of the vertices on its corner.
- Note: Each d-simplex σ is represented by a set of $d+1$ vertices
- All the faces of σ are nothing but all subsets of σ, excluding empty set

Some remarks

- For denoting a simplicial complex, we typically first assign labels to the vertices of the complex.
  - In class, the labels could be letters but could also be other things
  - In computer programs, the labels are almost always integers $0, \cdots, l-1$.
- Then, each simplex is represented as a set of the vertices on its corner.
- Note: Each d-simplex σ is represented by a set of $d+1$ vertices
- All the faces of σ are nothing but all subsets of σ, excluding empty set
- Also note: A d-simplex is typically represented by a **sorted** array of $d+1$ vertices (integers) in computer programs, this makes checking the equality of two simplices easier

Endowing Algebraic Structures to Complexes

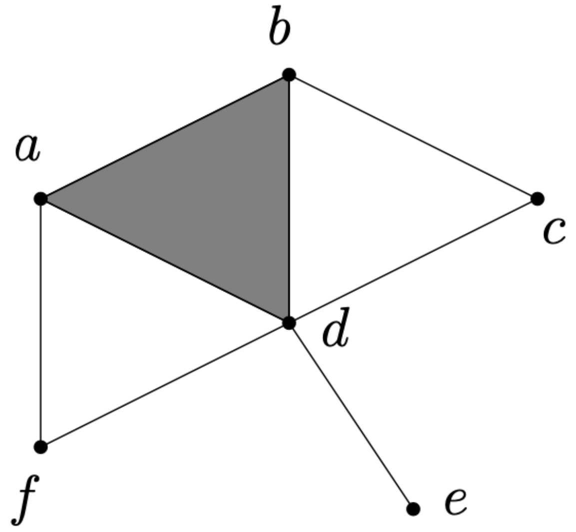
- Now we continue towards our goal of defining homology
- There are still *a few steps* before that
- Recall that homology is a "numeric" invariant that computer can handle
- More formally, it's an "algebraic" invariant.
- So, let's give a simplicial complex and its simplices an algebraic structure, so that we could do algebra on it (just like what we used to do 1+1=2 in primary school).

Chains

- We first introduce an algebraic notion called chains.
- A chain is a summation (formal sum) of a bunch of simplices of the same dimension d, and we also call it a d-chain.
- i.e., it is of the general form: $\sum_{i=1}^{k} \sigma_i$, where each σ_i is a d-simplex.

Chains

- We first introduce an algebraic notion called chains.
- A chain is a summation (formal sum) of a bunch of simplices of the same dimension d, and we also call it a d-chain.
- i.e., it is of the general form: $\sum_{i=1}^{k} \sigma_i$, where each σ_i is a d-simplex.

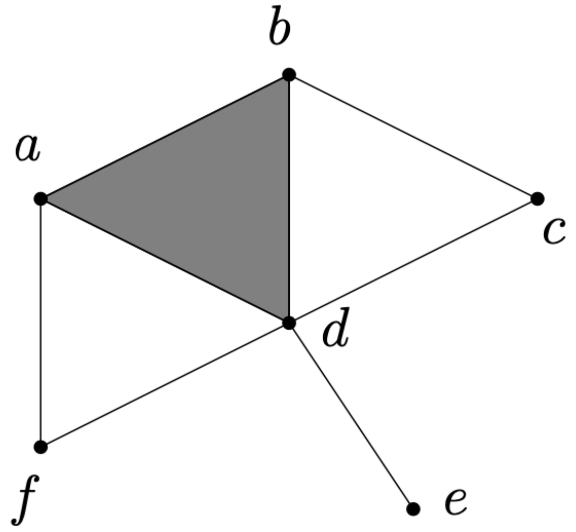


- 0-chain: $a + d + f$
- 1-chain: $ab + cd + de$
- 2-chain: abd

Image source: T.K.Dey and Y.Wang. Computational Topology for Data Analysis

Chains

- We first introduce an algebraic notion called chains.
- A chain is a summation (formal sum) of a bunch of simplices of the same dimension d, and we also call it a d-chain.
- i.e., it is of the general form: $\sum_{i=1}^{k} \sigma_i$, where each σ_i is a d-simplex.



- 0-chain: $a + d + f$
- 1-chain: $ab + cd + de$
- 2-chain: abd
- Note: we have a special chain ‘0’ which contains no simplices

Image source: T.K.Dey and Y.Wang. Computational Topology for Data Analysis

Summation of Chains

- The summation of two chains is called the "symmetric difference", i.e.,
  - Keep simplices that occurs in exactly one of the chains
  - A simplex occurring in both chains will be cancelled out

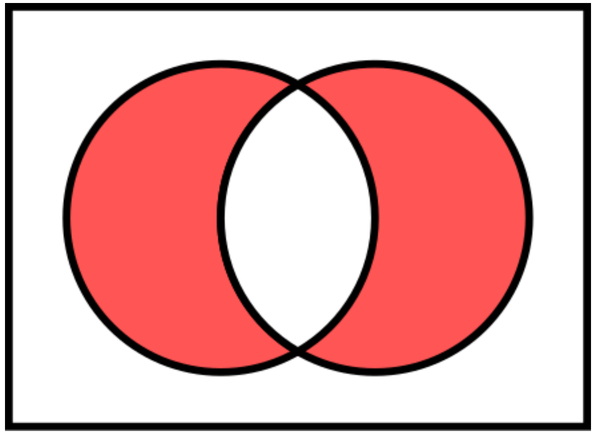
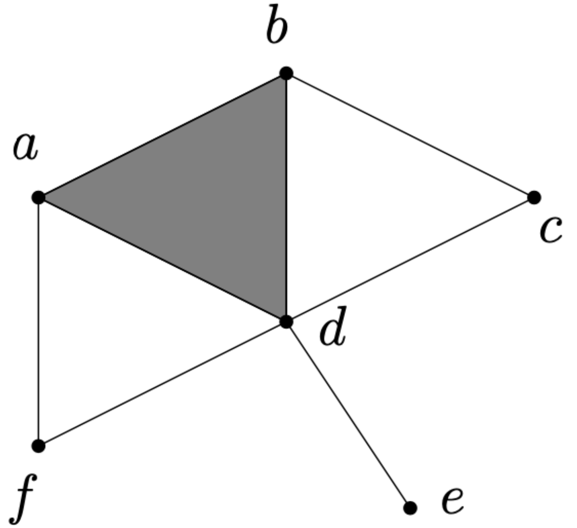


image source: Wikipedia

Summation of Chains

- The summation of two chains is called the "symmetric difference", i.e.,
  - Keep simplices that occurs in exactly one of the chains
  - A simplex occurring in both chains will be cancelled out

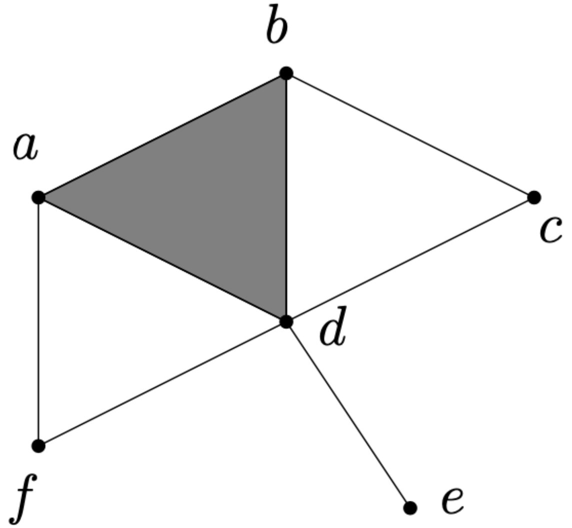


- $(a + d + f) + (a + c + e) = c + d + e + f$

Image source: T.K.Dey and Y.Wang. Computational Topology for Data Analysis

Summation of Chains

- The summation of two chains is called the "symmetric difference", i.e.,
  - Keep simplices that occurs in exactly one of the chains
  - A simplex occurring in both chains will be cancelled out



- $(a + d + f) + (a + c + e) = c + d + e + f$
- $(ab + cd + de) + (bd + cd) = ab + bd + de$

Image source: T.K.Dey and Y.Wang. Computational Topology for Data Analysis

Boundaries of Chains

- Next thing we want to define are boundaries for chains
- But before doing that let's try to define boundary for a region in general
- For a two-dimensional region, the boundary is just the "border" of the region

Image source: https://www.geoapify.com/tutorial/getting-administrative-divisions-boundaries/

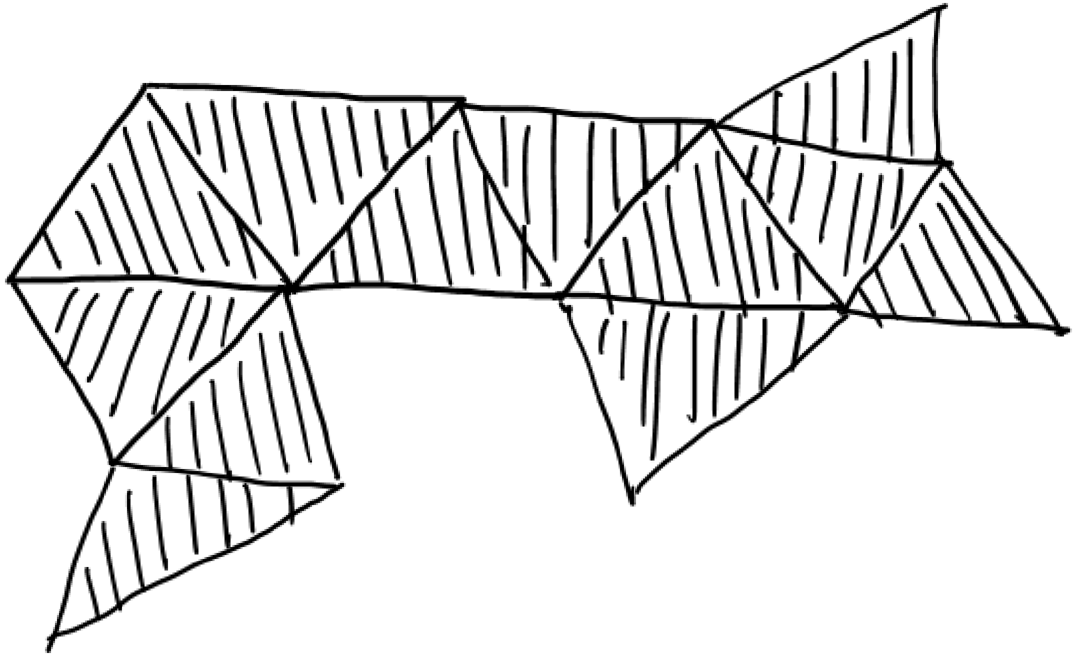
Boundaries of Chains

- Question: what is the boundary for a 1-dimensional line segment?

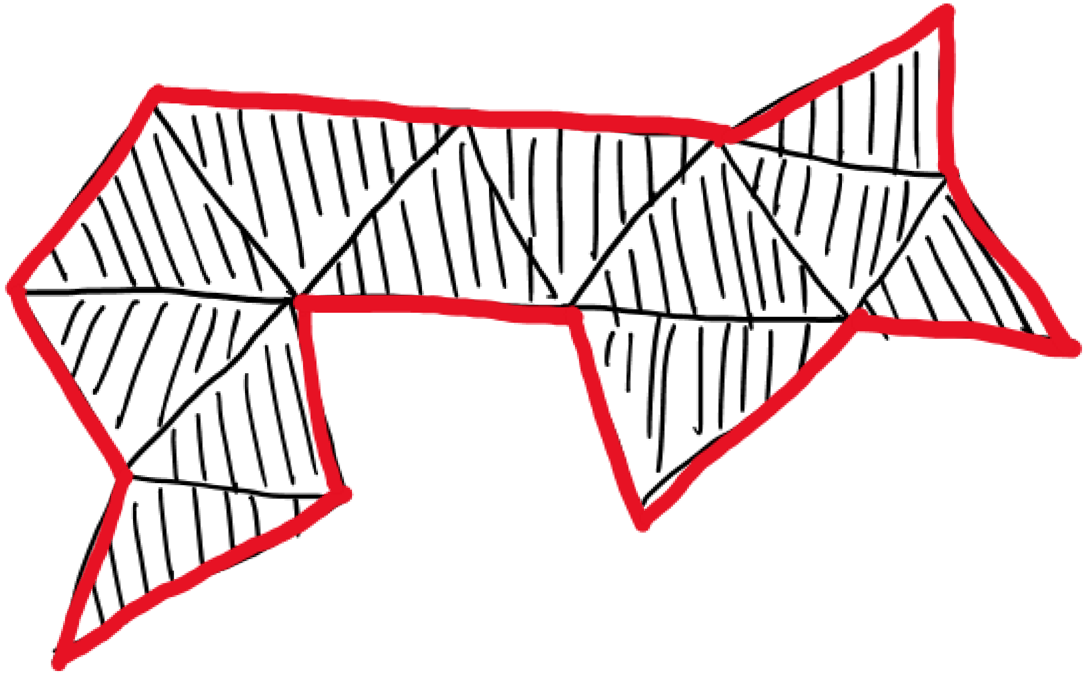
Boundaries of Chains

- Question: what is the boundary for a 1-dimensional line segment?
- Answer: the two end points (because they are the places where we cannot travel any further within the 1-dimensional region)

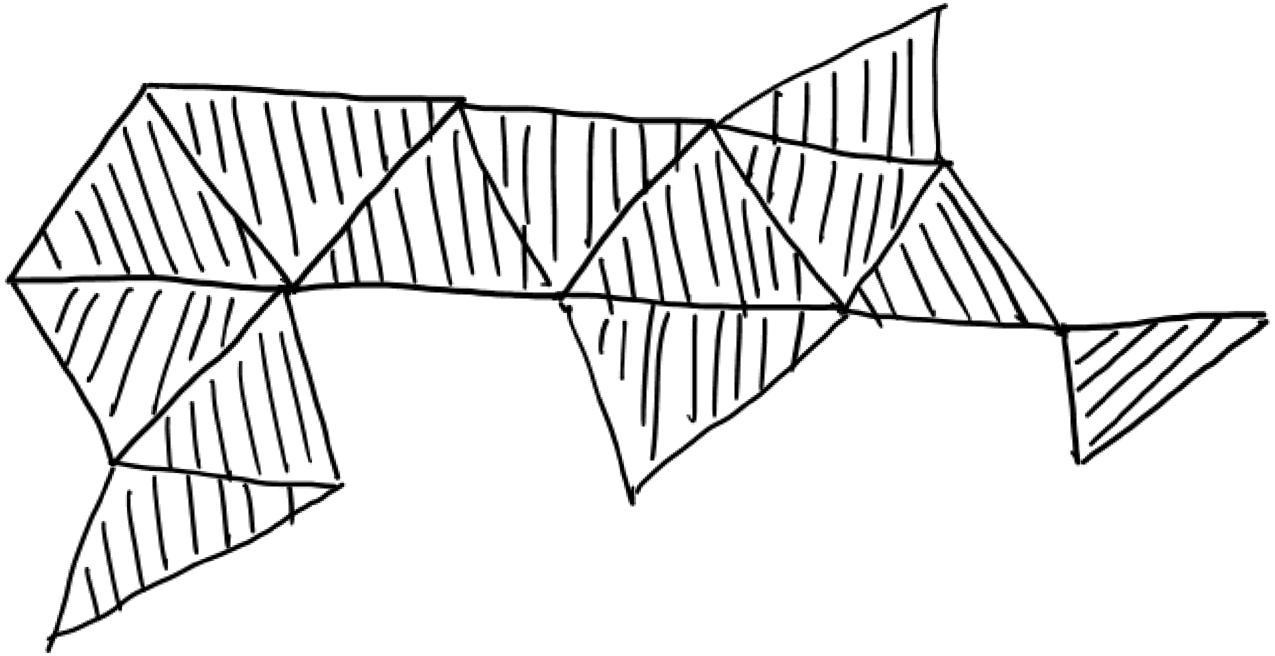
Boundaries of Chains



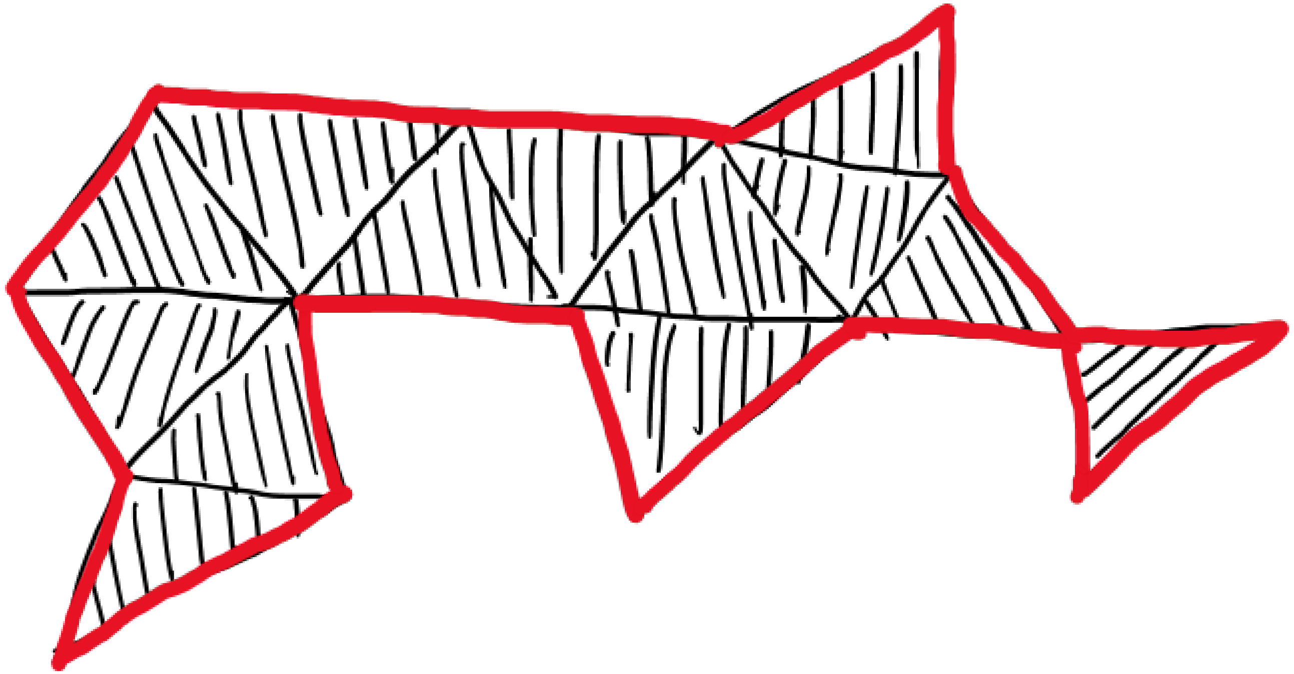
Boundaries of Chains



Boundaries of Chains



Boundaries of Chains



Boundaries of Chains

- **Observation**: the boundary of a d-chain c is also a chain, which is of one dimension lower, so it's a $(d+1)$-chain. We denote it as $\partial(c)$.

Boundary of a Simplex

- Since a single simplex is also a chain, we look at the **boundary of a simplex**:
  - The boundary of a vertex is 0 (empty).

Boundary of a Simplex

- Since a single simplex is also a chain, we look at the **boundary of a simplex**:
  - The boundary of a vertex is 0 (empty).
  - The boundary of an edge ab is its two endian vertices:

$$\partial(ab) = a + b$$

Boundary of a Simplex

- Since a single simplex is also a chain, we look at the **boundary of a simplex**:
  - The boundary of a vertex is 0 (empty).
  - The boundary of an edge ab is its two endian vertices:

$$\partial(ab) = a + b$$

  - The boundary of a triangle abc is the three edges it contains:

$$\partial(abc) = ab + ac + bc$$

Boundary of a Simplex

- Since a single simplex is also a chain, we look at the **boundary of a simplex**:
  - The boundary of a vertex is 0 (empty).
  - The boundary of an edge ab is its two endian vertices:
$$\partial(ab) = a + b$$
  - The boundary of a triangle abc is the three edges it contains:
$$\partial(abc) = ab + ac + bc$$
  - The boundary of a tetrahedron $abcd$ is the four triangles it contains:
$$\partial(abcd) = abc + abd + acd + bcd$$

Boundaries of Chains

- **Observation**: The boundary of a chain equals the summation of the boundaries of its simplices, i.e.,

$$\partial(\sigma_1 + \sigma_2 + \cdots + \sigma_k) = \partial(\sigma_1) + \partial(\sigma_2) + \cdots + \partial(\sigma_k)$$

Boundaries of Chains

- **Observation**: The boundary of a chain equals the summation of the boundaries of its simplices, i.e.,

$$\partial(\sigma_1 + \sigma_2 + \cdots + \sigma_k) = \partial(\sigma_1) + \partial(\sigma_2) + \cdots + \partial(\sigma_k)$$

- This helps us very easily calculate boundaries for chains, because the boundary of a simplex is immediately available.

Boundaries of Chains

- **Observation**: The boundary of a chain equals the summation of the boundaries of its simplices, i.e.,

$$\partial(\sigma_1 + \sigma_2 + \cdots + \sigma_k) = \partial(\sigma_1) + \partial(\sigma_2) + \cdots + \partial(\sigma_k)$$

- This helps us very easily calculate boundaries for chains, because the boundary of a simplex is immediately available.
- Ex:

$$\partial(ab + bc + cd) = a + d$$

Boundaries of Chains

- **Observation**: The boundary of a chain equals the summation of the boundaries of its simplices, i.e.,

$$\partial(\sigma_1 + \sigma_2 + \cdots + \sigma_k) = \partial(\sigma_1) + \partial(\sigma_2) + \cdots + \partial(\sigma_k)$$

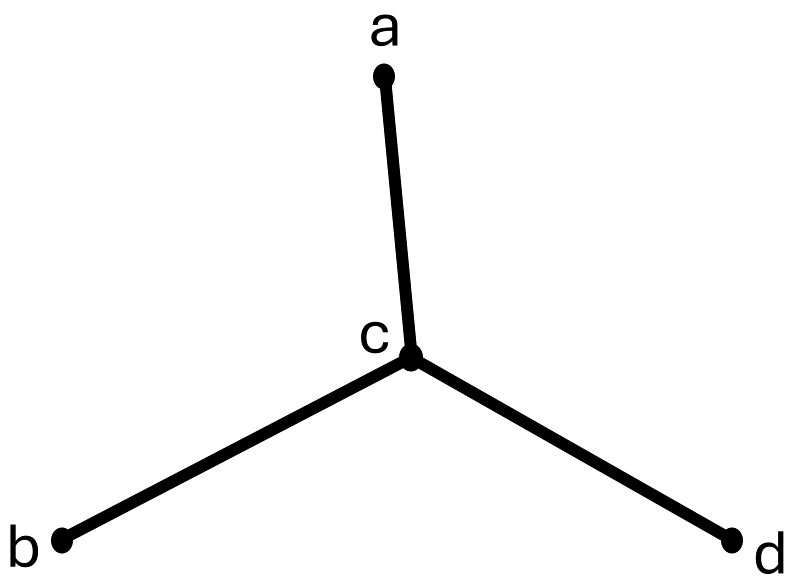
- This helps us very easily calculate boundaries for chains, because the boundary of a simplex is immediately available.
- Ex:

$$\partial(ab + bc + cd) = a + d$$

$$\partial(abc + bcd) = ab + ac + bd + cd$$

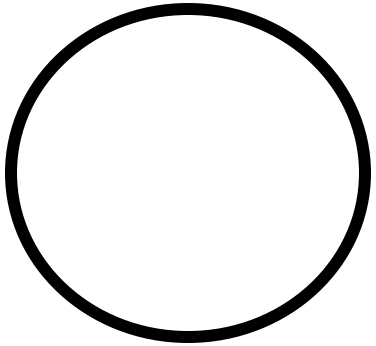
Quick Question

- What is the boundary of the following 1-chain?



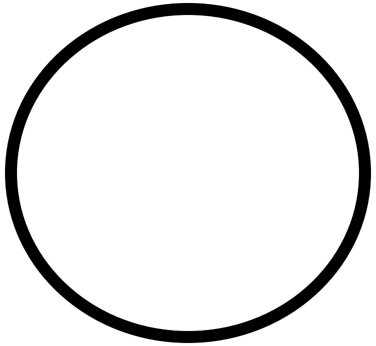
Boundaries of Some Special Chains

- What is the boundary of a circle?



Boundaries of Some Special Chains

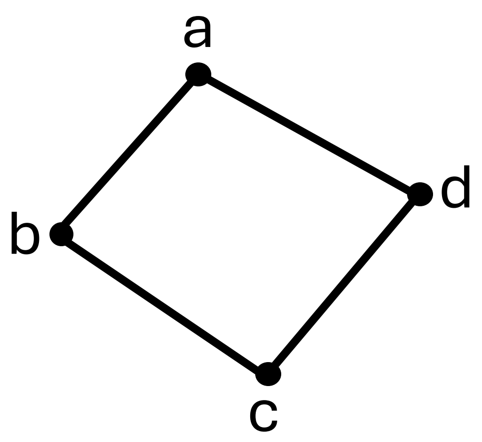
- What is the boundary of a circle?



- Answer: 0 (empty)
- **Definition**: We generalize a circle and define a ***d*-cycle** as a d-chain whose boundary is 0.

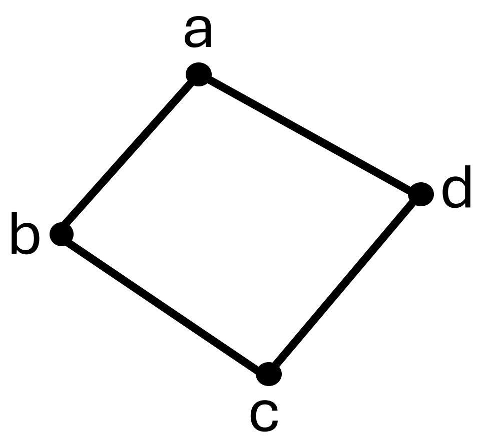
Cycles

- Example of a 1-cycle (the same as a cycle on graphs):



Cycles

- Example of a 1-cycle (the same as a cycle on graphs):



- Example of a 2-cycle (triangulated sphere):

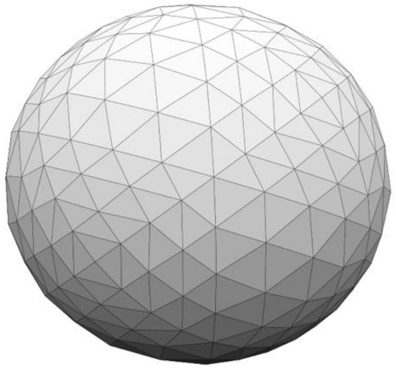


Image source: https://eugene-eeo.github.io/blog/sphere-triangles.html

Fundamental theorem of homology

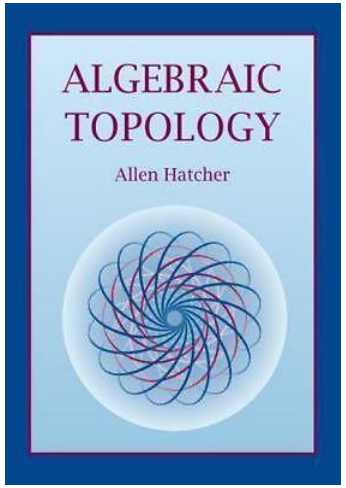
- Recall that the boundary of a d-chain c is also a chain, which is a $(d+1)$-chain, and we denote it as $\partial(c)$.
- Since $\partial(c)$ is also a chain, what is its boundary?

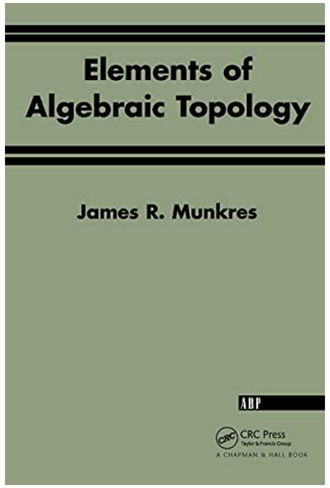
Fundamental theorem of homology

- Recall that the boundary of a d-chain c is also a chain, which is a $(d+1)$-chain, and we denote it as $\partial(c)$.
- Since $\partial(c)$ is also a chain, what is its boundary?
- **Theorem**: the boundary of a chain is always a cycle, aka. boundaries have empty boundaries, or simply
    - For any chain c, $\partial\big(\partial(c)\big) = 0$.

Fundamental theorem of homology

- Recall that the boundary of a d-chain c is also a chain, which is a $(d+1)$-chain, and we denote it as $\partial(c)$.
- Since $\partial(c)$ is also a chain, what is its boundary?
- **Theorem**: the boundary of a chain is always a cycle, aka. boundaries have empty boundaries, or simply
  - For any chain c, $\partial\big(\partial(c)\big) = 0$.
- The above theorem is a fundamental fact making homology theory possible
- The proof of it and any further algebraic interpretations of it are beyond the scope of this course.





Fundamental theorem of homology

- A simple exercise: calculate the boundary of $\partial(abc + acd + ade)$.

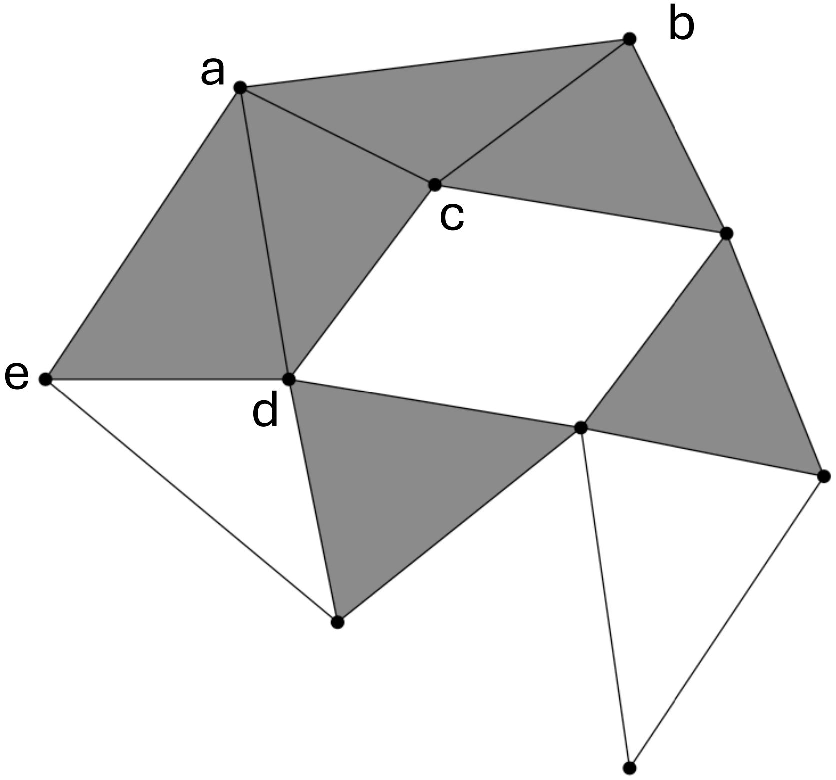


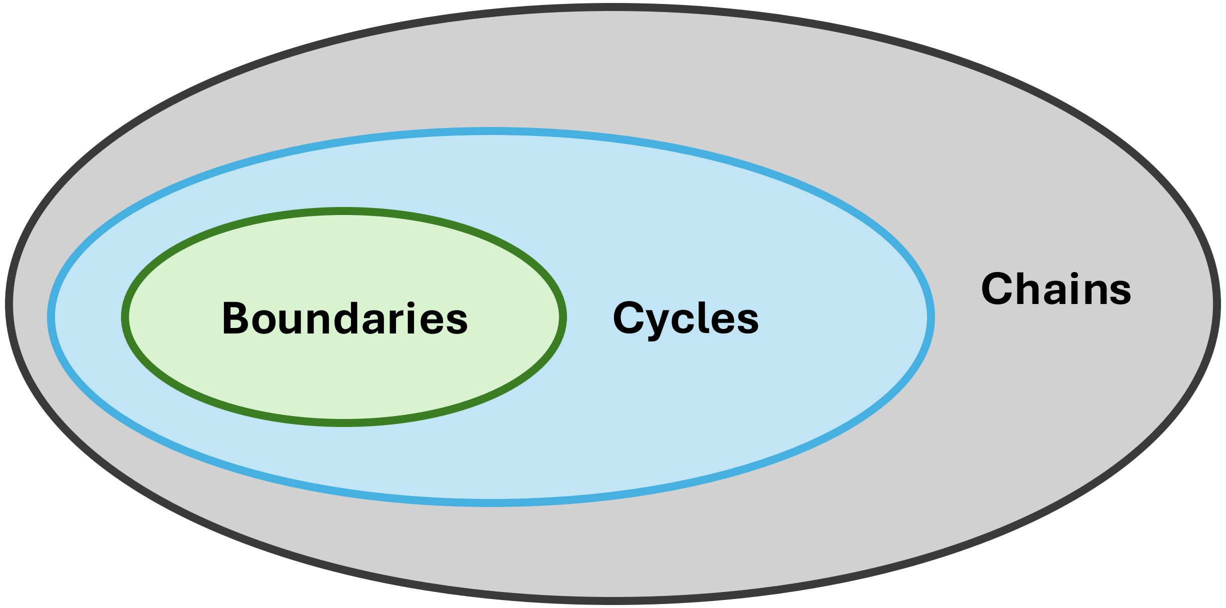
Image source: Patrick Schnider. Introduction to Topological Data Analysis Lecture Notes FS 2023

Classification for the set of all cycles

- Terminology:
  - Boundaries: **Trivial** cycles
  - Cycles that are not boundaries: **Non-trivial** cycles

Classification for the set of all cycles

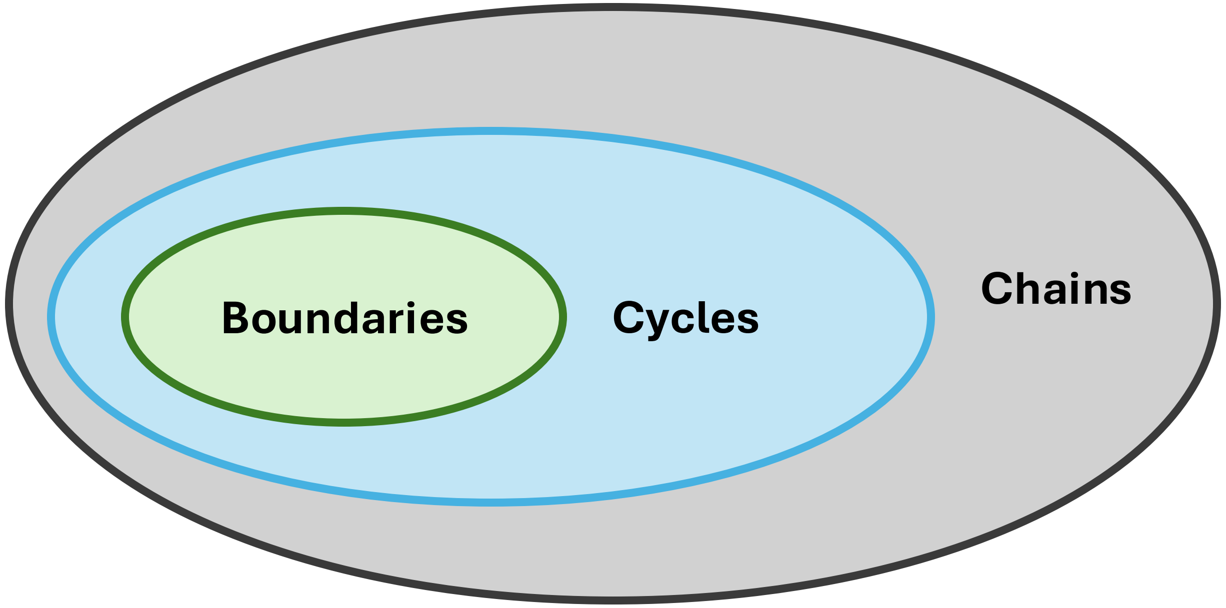
- Terminology:
  - Boundaries: **Trivial** cycles
  - Cycles that are not boundaries: **Non-trivial** cycles
- We can classify **the set of all chains for a simplicial complex** into:



Boudnaries $\subseteq$ Cycles $\subseteq$ Chains

Classification for the set of all cycles

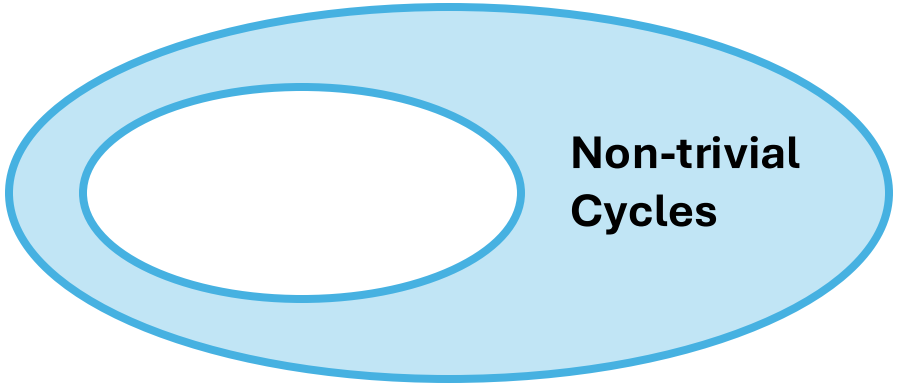
- Terminology:
  - Boundaries: **Trivial** cycles
  - Cycles that are not boundaries: **Non-trivial** cycles
- We can classify **the set of all chains for a simplicial complex** into:
- Homology is a theory studying those ***cycles that are not boundaries*** for a simplicial complex.



Boudnaries ⊆ Cycles ⊆ Chains

Classification for the set of all cycles

- Terminology:
  - Boundaries: **Trivial** cycles
  - Cycles that are not boundaries: **Non-trivial** cycles
- We can classify **the set of all chains for a simplicial complex** into:
- Homology is a theory studying those ***cycles that are not boundaries*** for a simplicial complex.



$$\text{Boudnaries} \subseteq \text{Cycles} \subseteq \text{Chains}$$

Homology Studies: Cycles That Are NOT Boundaries

- Homology is a theory studying those ***cycles that are not boundaries*** for a simplicial complex.

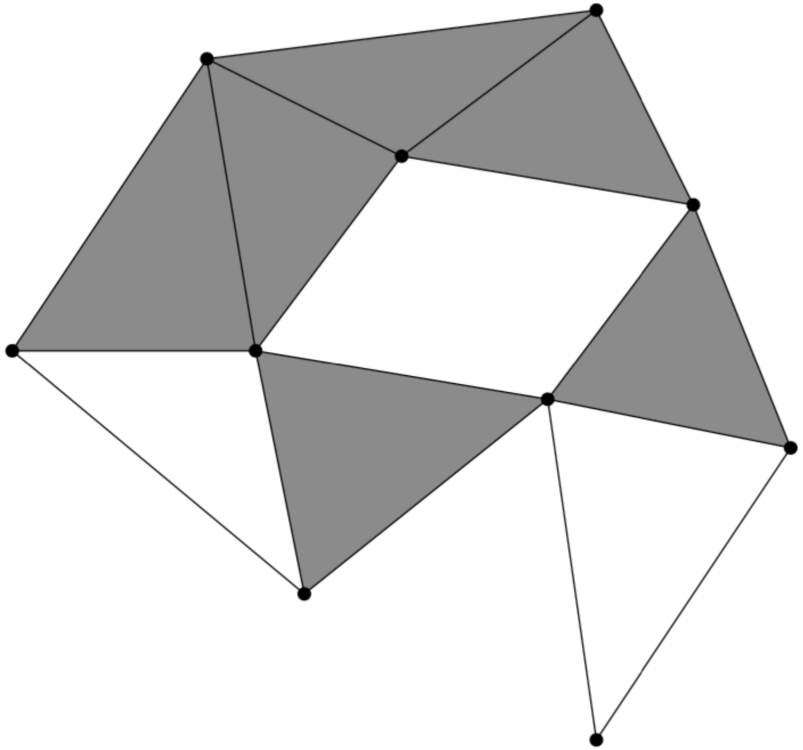


Image source: Patrick Schnider. Introduction to Topological Data Analysis Lecture Notes FS 2023

Homology Studies: Cycles That Are NOT Boundaries

- Homology is a theory studying those ***cycles that are not boundaries*** for a simplicial complex.

Boundary

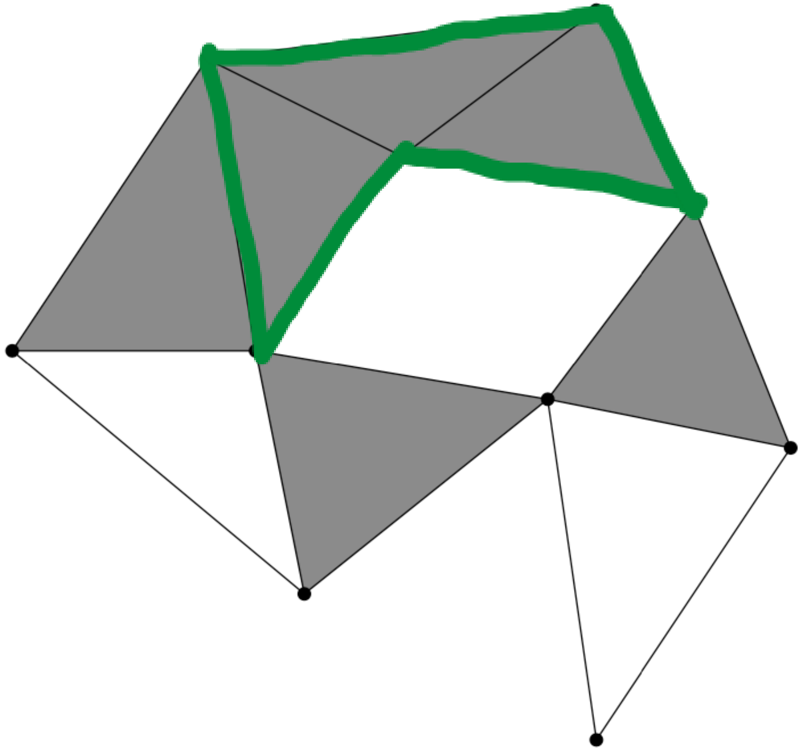


Image source: Patrick Schnider. Introduction to Topological Data Analysis Lecture Notes FS 2023

Homology Studies: Cycles That Are NOT Boundaries

- Homology is a theory studying those ***cycles that are not boundaries*** for a simplicial complex.

Boundary

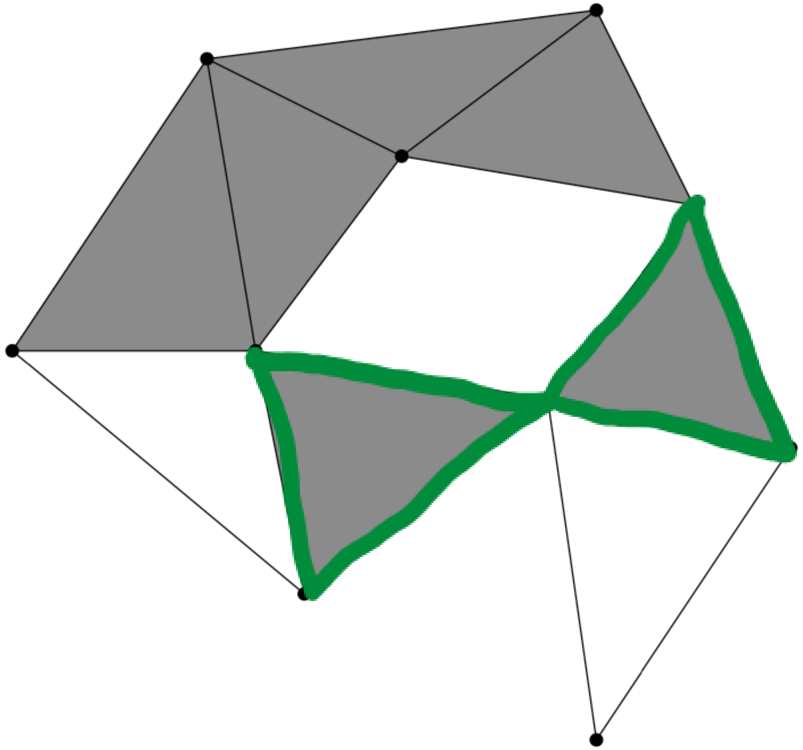


Image source: Patrick Schnider. Introduction to Topological Data Analysis Lecture Notes FS 2023

Homology Studies: Cycles That Are NOT Boundaries

- Homology is a theory studying those ***cycles that are not boundaries*** for a simplicial complex.

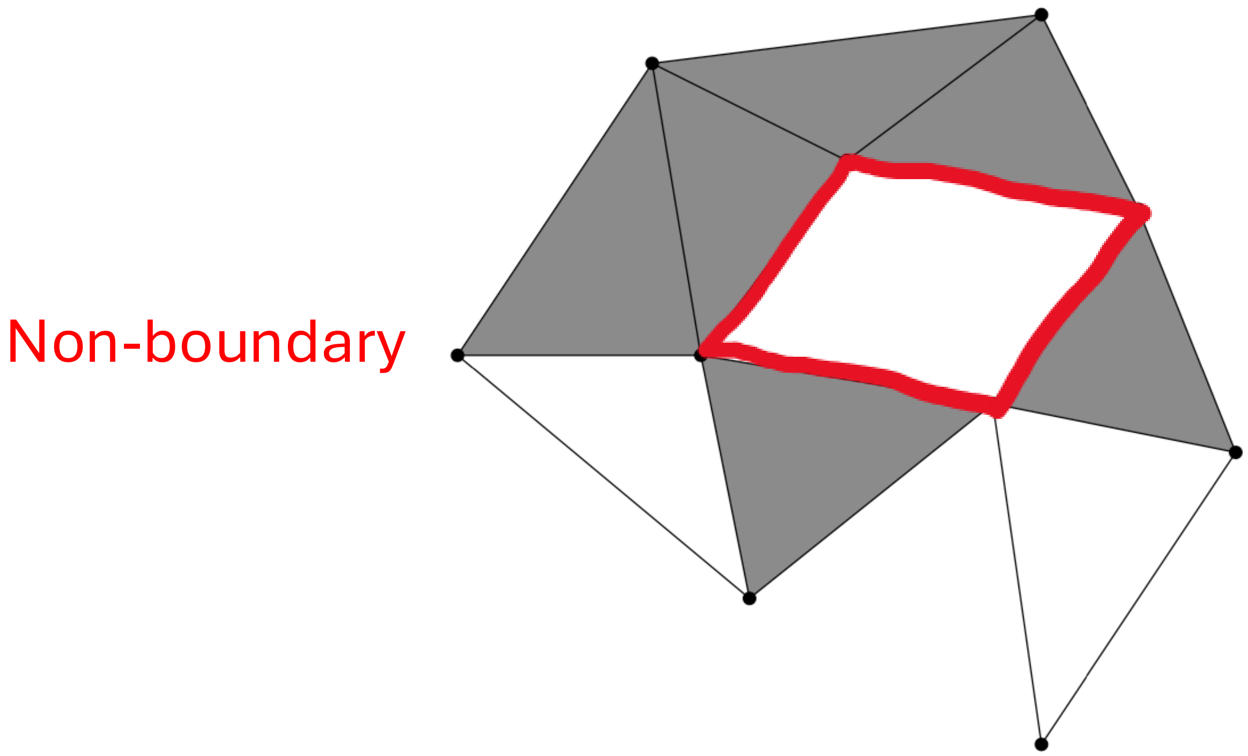


Image source: Patrick Schnider. Introduction to Topological Data Analysis Lecture Notes FS 2023

Homology Studies: Cycles That Are NOT Boundaries

- Homology is a theory studying those ***cycles that are not boundaries*** for a simplicial complex.

Non-boundary

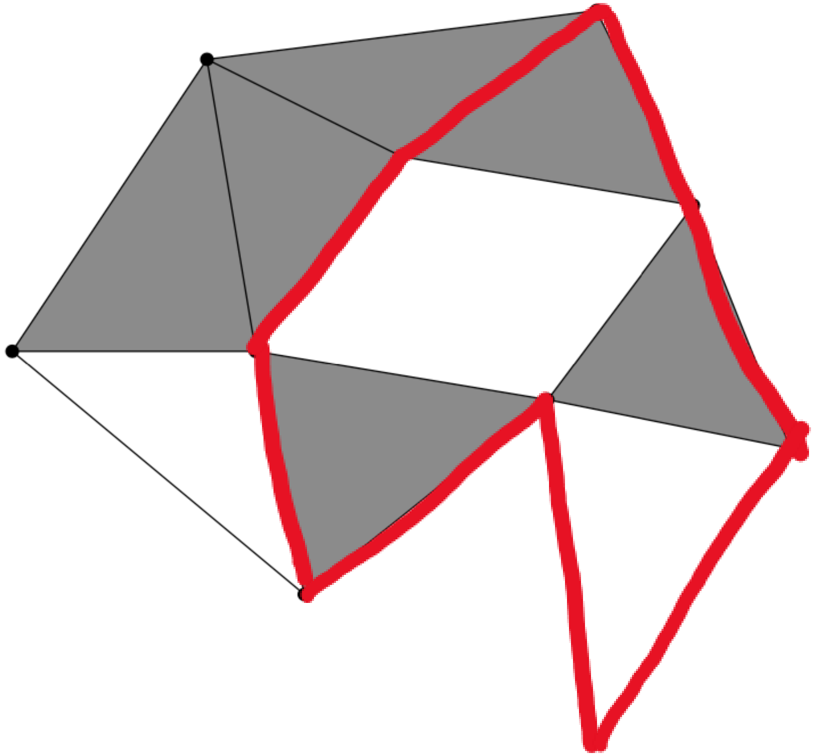


Image source: Patrick Schnider. Introduction to Topological Data Analysis Lecture Notes FS 2023

Homology Studies: Cycles That Are NOT Boundaries

- The reason we focus on non-boundary cycles is that:
  - We can “squeeze” a boundary cycle into a single point within the simplicial complex, so it doesn’t seem to represent an interesting “feature” of the data

Image source: Yan et al. Persistence Landscape based Topological Data Analysis for Personalized Arrhythmia Classification

Homology Studies: Cycles That Are NOT Boundaries

- The reason we focus on non-boundary cycles is that:
  - We can "squeeze" a boundary cycle into a single point within the simplicial complex, so it doesn't seem to represent an interesting "feature" of the data
  - On the other hand, a non-boundary cycle typically represents a "***hole***" in the complex, which represents some "essential feature" of the data

Image source: Yan et al. Persistence Landscape based Topological Data Analysis for Personalized Arrhythmia Classification

Homology Studies: Cycles That Are NOT Boundaries

- The reason we focus on non-boundary cycles is that:
  - We can "squeeze" a boundary cycle into a single point within the simplicial complex, so it doesn't seem to represent an interesting "feature" of the data
  - On the other hand, a non-boundary cycle typically represents a "***hole***" in the complex, which represents some "essential feature" of the data
- Ex: For the right simplicial complex, do you think the green boundary cycle represent the shape, or the red non-boundary one?

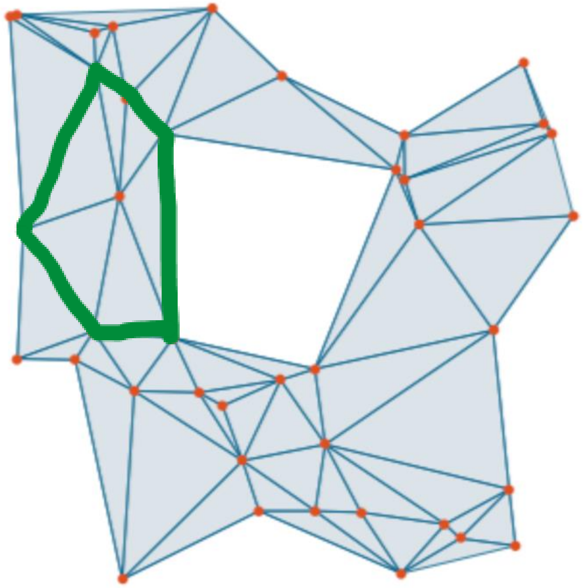


Image source: Yan et al. Persistence Landscape based Topological Data Analysis for Personalized Arrhythmia Classification

Homology Studies: Cycles That Are NOT Boundaries

- The reason we focus on non-boundary cycles is that:
  - We can "squeeze" a boundary cycle into a single point within the simplicial complex, so it doesn't seem to represent an interesting "feature" of the data
  - On the other hand, a non-boundary cycle typically represents a "***hole***" in the complex, which represents some "essential feature" of the data

- Ex: For the right simplicial complex, do you think the green boundary cycle represent the shape, or the red non-boundary one?

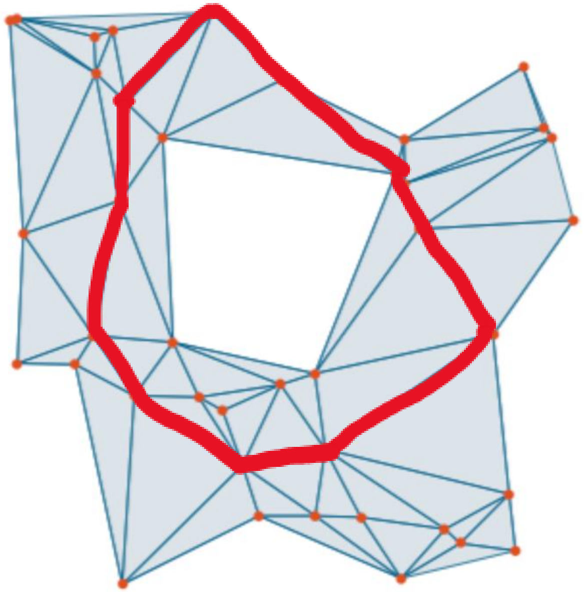


Image source: Yan et al. Persistence Landscape based Topological Data Analysis for Personalized Arrhythmia Classification

Homology Studies: Cycles That Are NOT Boundaries

- The reason we focus on non-boundary cycles is that:
  - We can "squeeze" a boundary cycle into a single point within the simplicial complex, so it doesn't seem to represent an interesting "feature" of the data
  - On the other hand, a non-boundary cycle typically represents a "***hole***" in the complex, which represents some "essential feature" of the data

- Ex: For the right simplicial complex, do you think the red boundary cycle represent the shape, or the large blue non-boundary one?

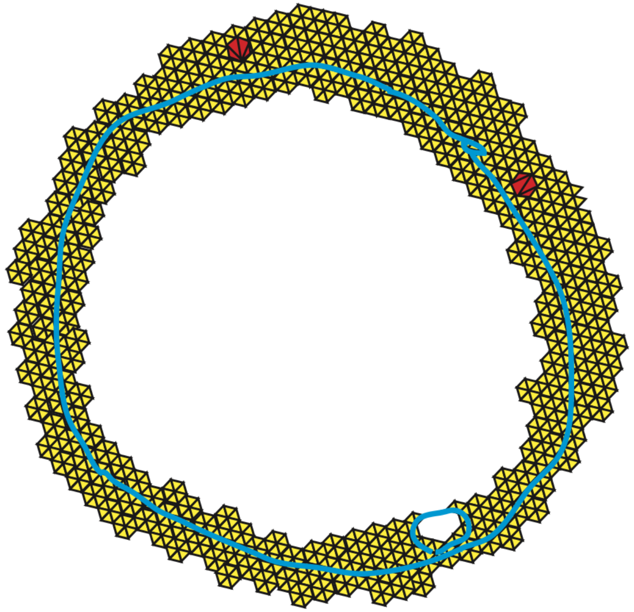


Image source: Gunnar Carlsson, Topology and Data

Homology Studies: Cycles That Are NOT Boundaries

- The reason we focus on non-boundary cycles is that:
  - We can "squeeze" a boundary cycle into a single point within the simplicial complex, so it doesn't seem to represent an interesting "feature" of the data
  - On the other hand, a non-boundary cycle typically represents a "***hole***" in the complex, which represents some "essential feature" of the data

- Ex: For the right simplicial complex, do you think the red boundary cycle represent the shape, or the large blue non-boundary one?
- **Trying to capture the something like the major blue cycle to represent the shape of data is an aim of TDA and the course!!**

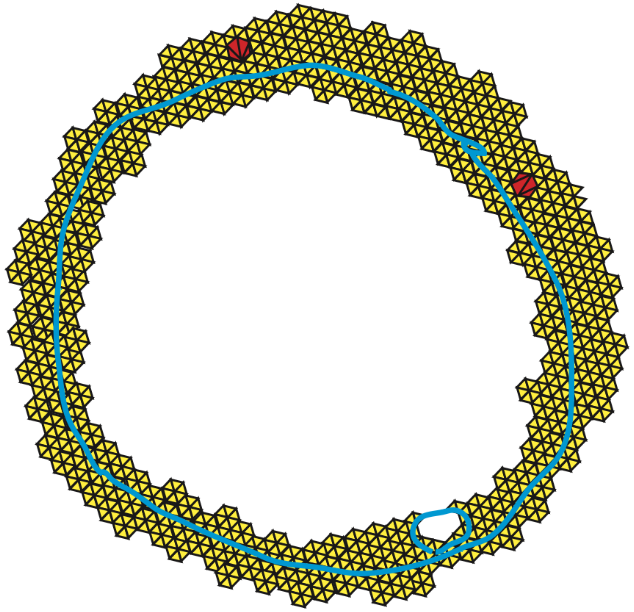


Image source: Gunnar Carlsson, Topology and Data

Homology Studies: Cycles That Are NOT Boundaries

- So far we have been showing **1-dimensional non-boundary cycles** (aka. holes) captured by homology theory, partially because it's easy to visualize
- But notice that homology can capture holes in any dimension (≥ 0).
- What about holes in other dimensions?

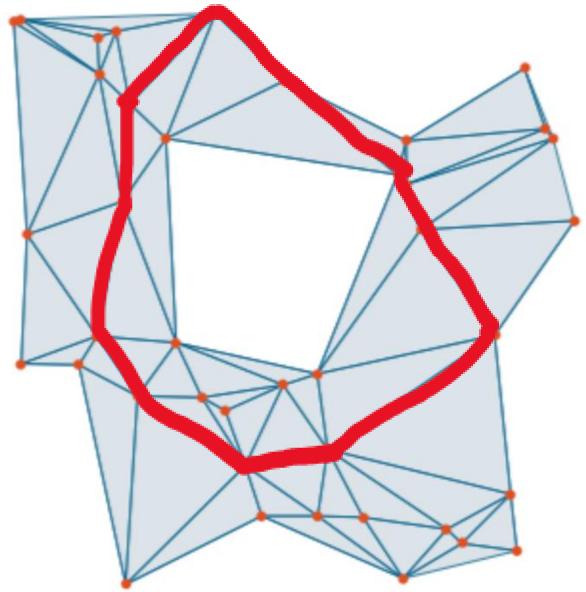
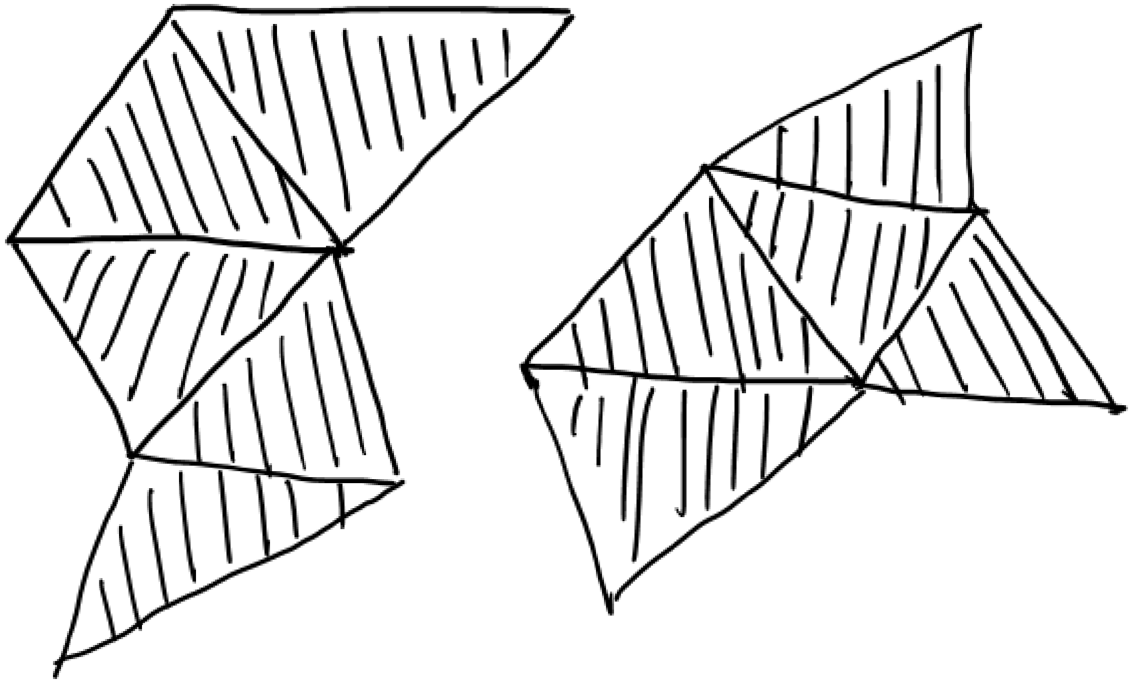


Image source: Yan et al. Persistence Landscape based Topological Data Analysis for Personalized Arrhythmia Classification

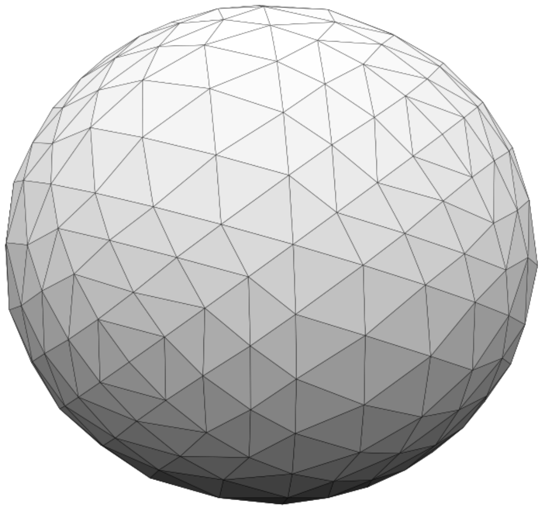
Homology Studies: Cycles That Are NOT Boundaries

- **0-dimensional holes** capture "gaps" between different connected components:



Homology Studies: Cycles That Are NOT Boundaries

- **2-dimensional holes** capture "cavity" or "hollowness" inside:



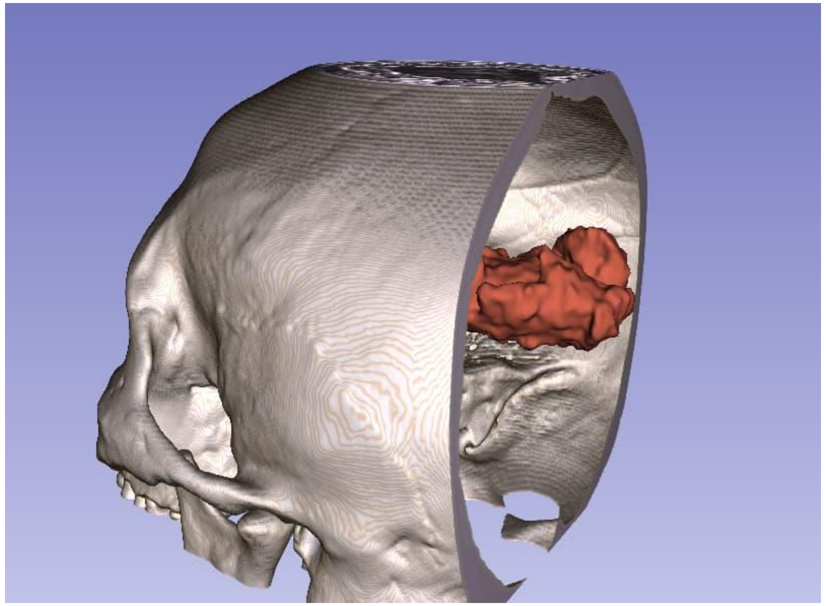


Image source: https://eugene-eeo.github.io/blog/sphere-triangles.html

Homology Basis: Additional Structures on Cycle Space

- Okay, now we know that homology studies cycles that are not boundaries
- But we can observe more algebraic structures on the cycles in a simplicial complex
- **Fact**: Each d-cycle of a simplicial complex is "generated by" a set of non-trivial d-cycles called the ***homology basis***.

Homology Basis: Additional Structures on Cycle Space

- Okay, now we know that homology studies cycles that are not boundaries
- But we can observe more algebraic structures on the cycles in a simplicial complex
- **Fact**: Each d-cycle of a simplicial complex is "generated by" a set of non-trivial d-cycles called the ***homology basis***.
- Formally, a d-cycle z being "generated by" cycles in the homology basis means that z **can be written as:**
  - A sum of cycles in the basis + a boundary (which is "trivial").

Homology Basis: Additional Structures on Cycle Space

- **Ex**: A homology basis for the 1-cycles in the right simplicial complex contains the **single red 1-cycle**

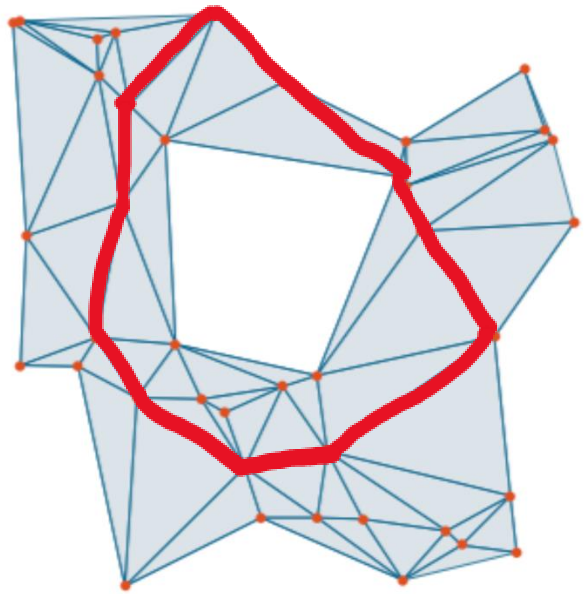


Image source: Yan et al. Persistence Landscape based Topological Data Analysis for Personalized Arrhythmia Classification

Homology Basis: Additional Structures on Cycle Space

- **Ex**: A homology basis for the 1-cycles in the right simplicial complex contains the **single red 1-cycle**
- Suppose we have another green 1-cycle

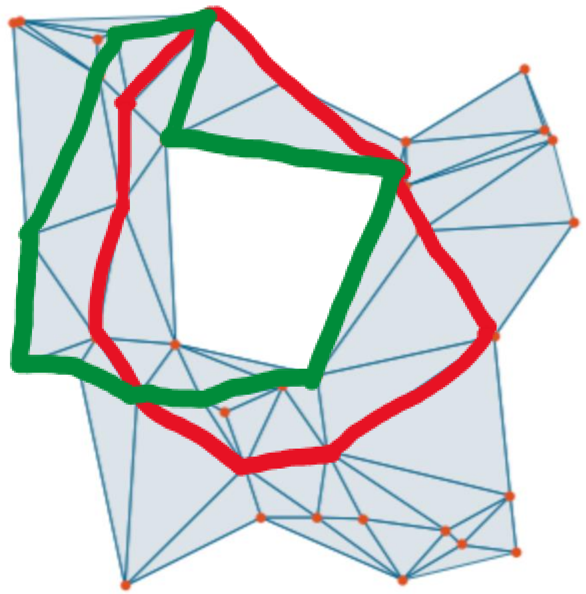


Image source: Yan et al. Persistence Landscape based Topological Data Analysis for Personalized Arrhythmia Classification

Homology Basis: Additional Structures on Cycle Space

- **Ex**: A homology basis for the 1-cycles in the right simplicial complex contains the **single red 1-cycle**
- Suppose we have another green 1-cycle
- The green 1-cycle equals:
  - red 1-cycle + boundary of shaded 2-chain

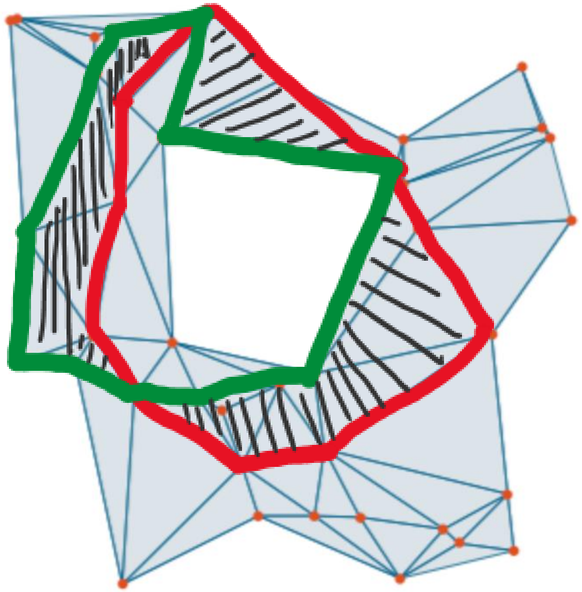


Image source: Yan et al. Persistence Landscape based Topological Data Analysis for Personalized Arrhythmia Classification

Homology Basis: Additional Structures on Cycle Space

- **Ex**: A homology basis for the 1-cycles in the right simplicial complex contains the **single red 1-cycle**
- Suppose we have another green 1-cycle
- The green 1-cycle equals:
  - red 1-cycle + boundary of shaded 2-chain
- In a sense, the red cycle "generates" the green cycle because you can continuously stretch the red one to the green one

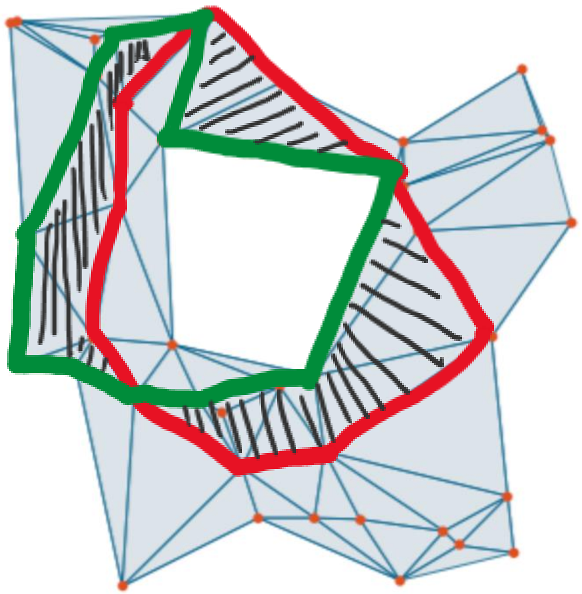
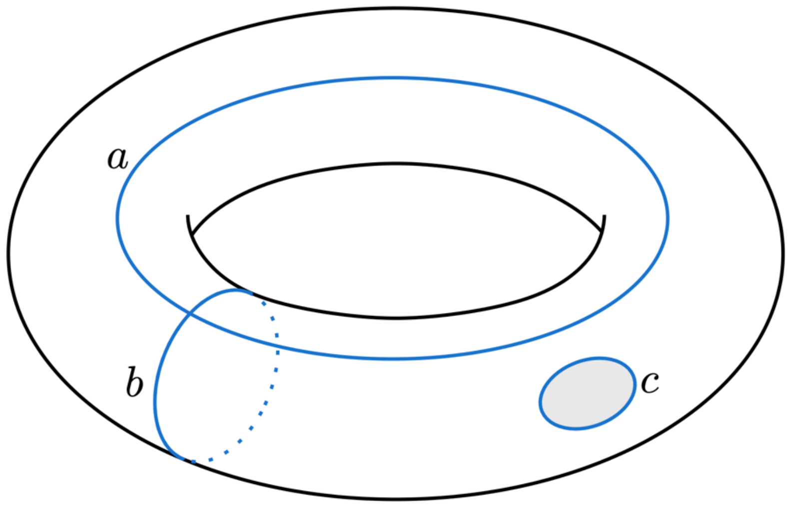


Image source: Yan et al. Persistence Landscape based Topological Data Analysis for Personalized Arrhythmia Classification

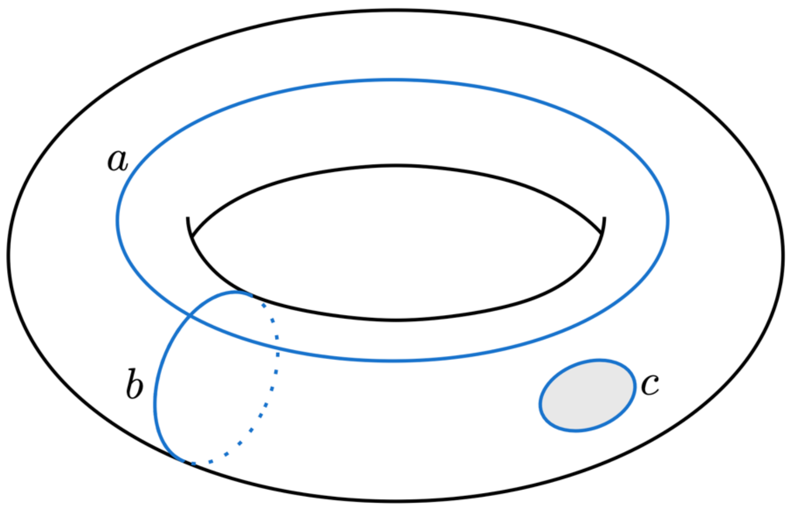
Homology Basis: Additional Structures on Cycle Space

- **Ex**: A homology basis for the 1-cycles in a torus (the surface of a donut) contains two 1-cycles:
    - $\boldsymbol{a}$ **(longitude) and** $\boldsymbol{b}$ **(meridian)**



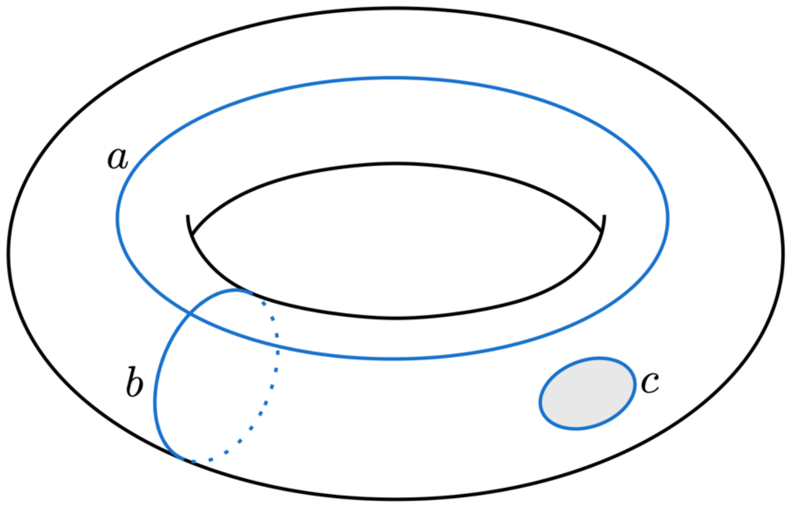
Homology Basis: Additional Structures on Cycle Space

- **Ex**: A homology basis for the 1-cycles in a torus (the surface of a donut) contains two 1-cycles:
  - $\boldsymbol{a}$ **(longitude) and** $\boldsymbol{b}$ **(meridian)**
- One way to differentiate trivial and non-trivial cycles on a torus:
  - a non-trivial cycle does not cut the torus into two pieces



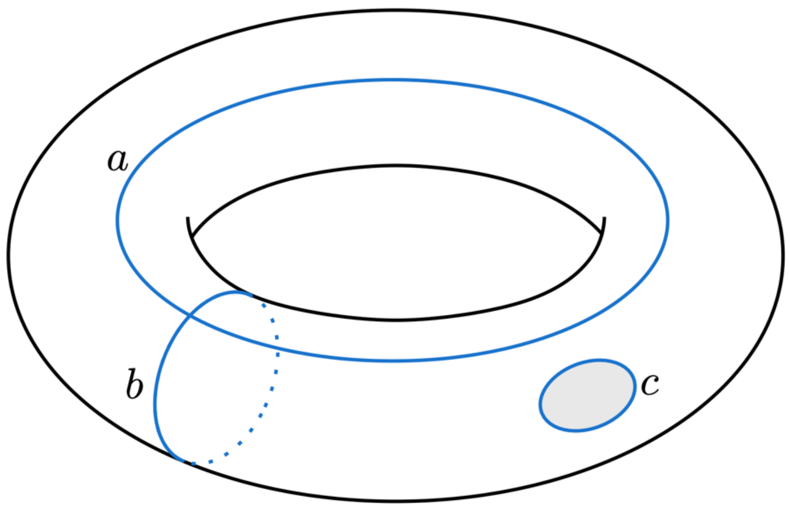
Homology Basis: Additional Structures on Cycle Space

- **Ex**: A homology basis for the 1-cycles in a torus (the surface of a donut) contains two 1-cycles:
  - $\boldsymbol{a}$ **(longitude) and** $\boldsymbol{b}$ **(meridian)**
- One way to differentiate trivial and non-trivial cycles on a torus:
  - a non-trivial cycle does not cut the torus into two pieces
  - while a trivial cycle (such as c) does



Homology Basis: Additional Structures on Cycle Space

- **Ex**: A homology basis for the 1-cycles in a torus (the surface of a donut) contains two 1-cycles:
  - **a (longitude) and b (meridian)**
- One way to differentiate trivial and non-trivial cycles on a torus:
  - a non-trivial cycle does not cut the torus into two pieces
  - while a trivial cycle (such as c) does
- Example of how a and b generate cycles in torus: the cycle in the lower torus equals



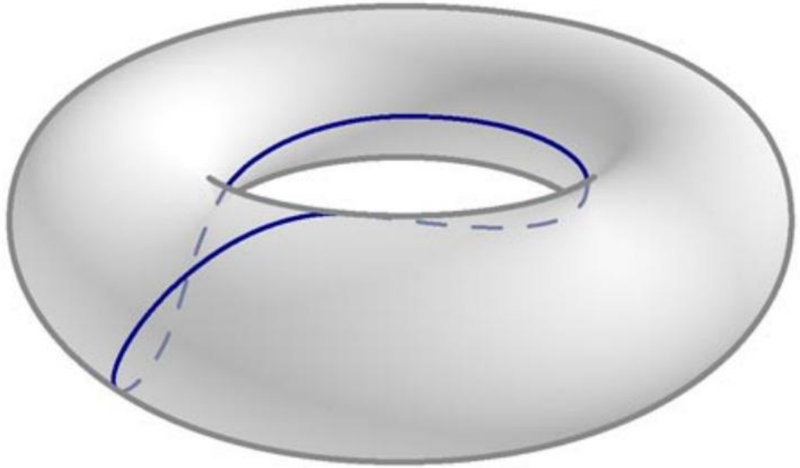
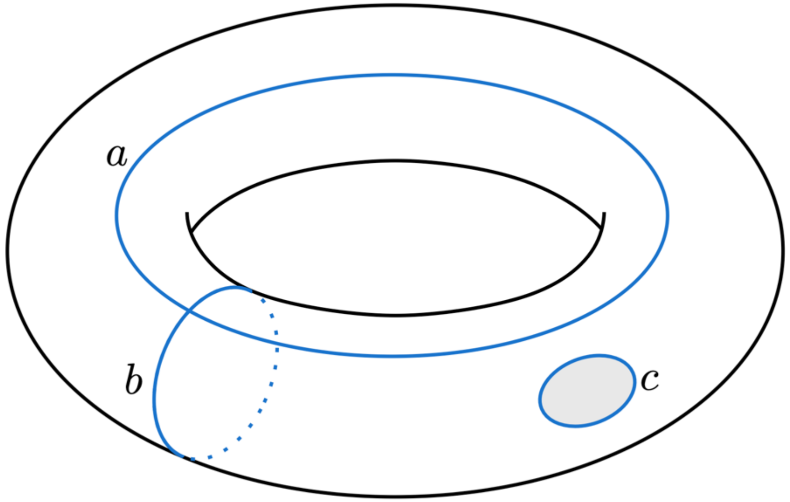


Image source: Toroidal Hitomezashi patterns - by Richard Green

Homology Basis: Additional Structures on Cycle Space

- **Ex**: A homology basis for the 1-cycles in a torus (the surface of a donut) contains two 1-cycles:
  - ***a* (longitude) and *b* (meridian)**
- One way to differentiate trivial and non-trivial cycles on a torus:
  - a non-trivial cycle does not cut the torus into two pieces
  - while a trivial cycle (such as c) does
- Example of how a and b generate cycles in torus: the cycle in the lower torus equals
  - $a + b +$ a boundary



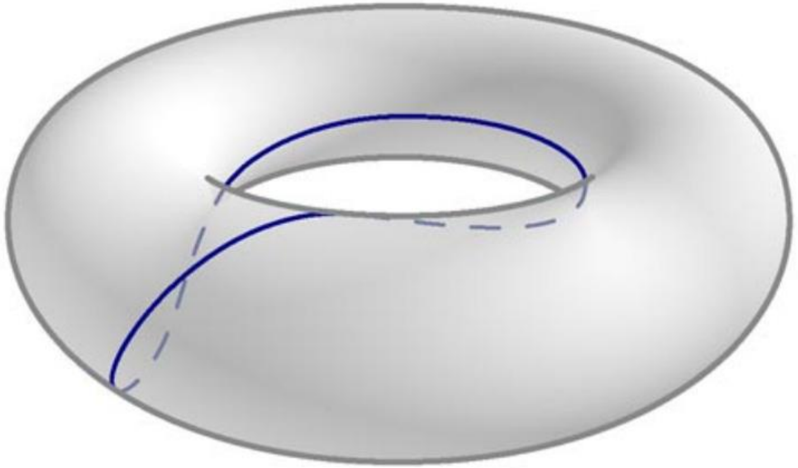
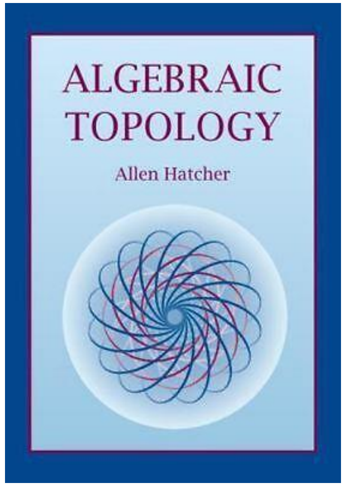
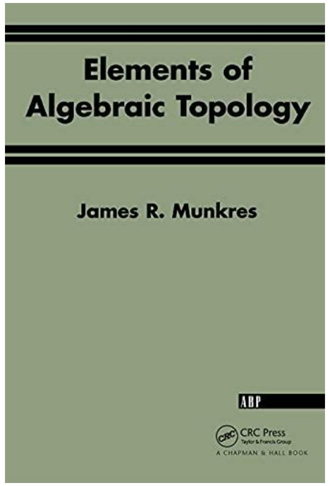


Image source: Toroidal Hitomezashi patterns - by Richard Green

Full Algebraic Structures on Cycle Space

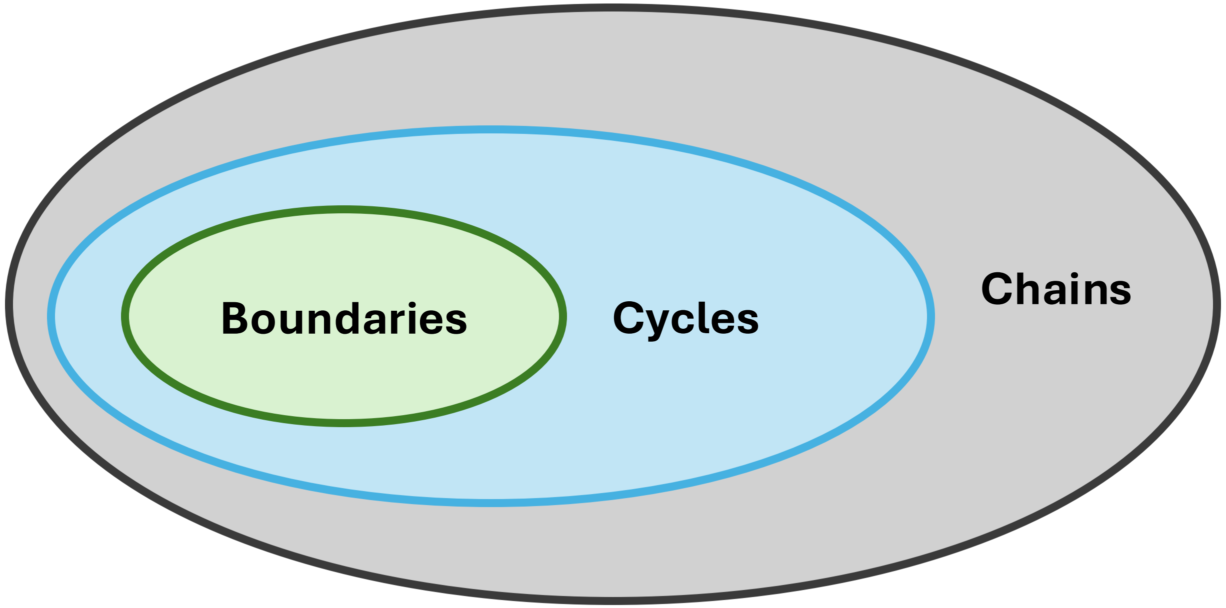
- We will briefly explain the full algebraic structures for the cycles and boundaries we described so far; see the textbooks for detailed formulation





Full Algebraic Structures on Cycle Space

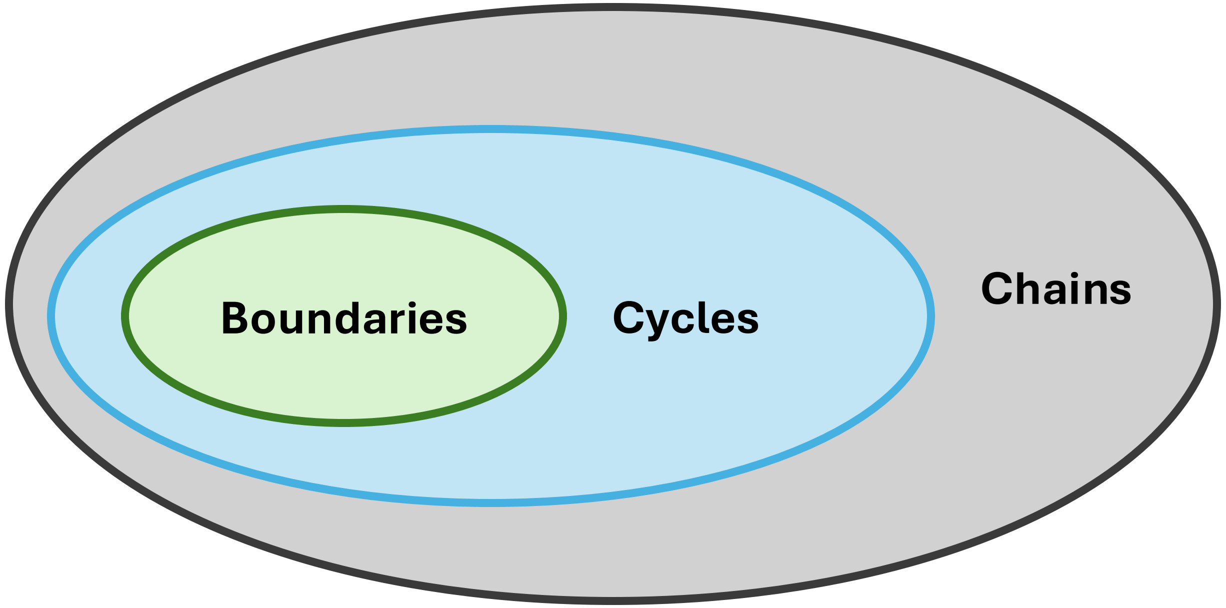
- Recall the following picture.
- We have that all p-chains for simplicial complex K not only form a set, but also form a **vector space** (object studied by linear algebra; see: https://en.wikipedia.org/wiki/Vector_space), denoted by $C_p(K)$



$$\text{Boudnaries} \subseteq \text{Cycles} \subseteq \text{Chains}$$

Full Algebraic Structures on Cycle Space

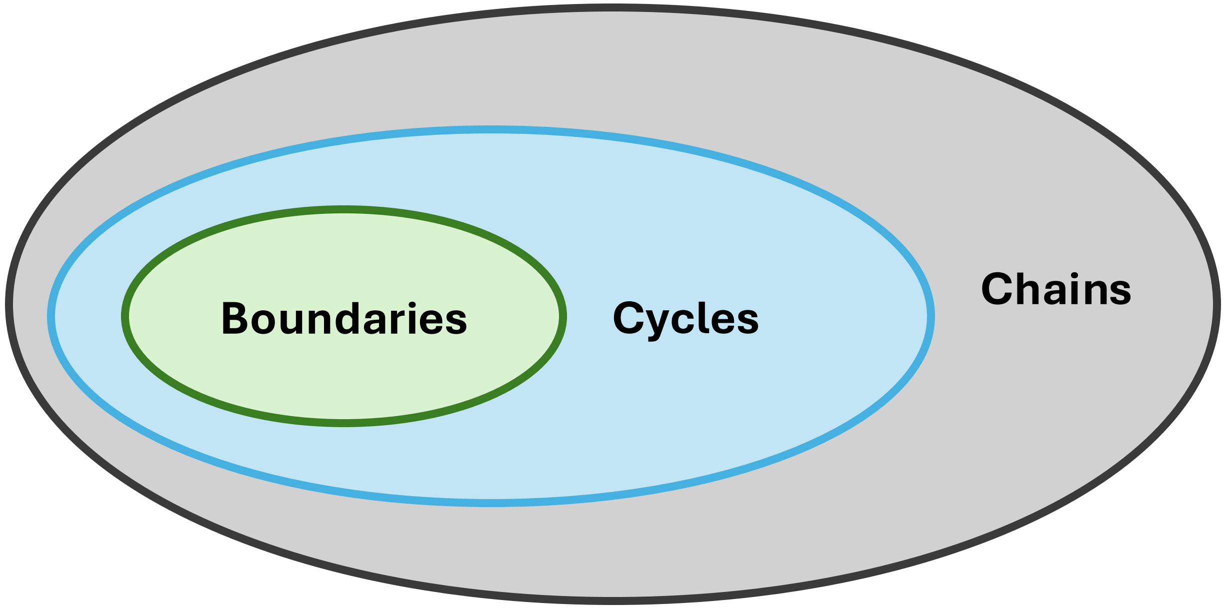
- Recall the following picture.
- We have that all p-chains for simplicial complex K not only form a set, but also form a **vector space** (object studied by linear algebra; see: https://en.wikipedia.org/wiki/Vector_space), denoted by $C_p(K)$



$$\text{Boudnaries} \subseteq \text{Cycles} \subseteq \text{Chains}$$

Full Algebraic Structures on Cycle Space

- Recall the following picture.
- We have that all p-chains for simplicial complex K not only form a set, but also form a **vector space** (object studied by linear algebra; see: https://en.wikipedia.org/wiki/Vector_space), denoted by $C_p(K)$
- Furthermore, boundaries and cycles are not only subsets but also **vector subspaces** of $Z_p(K)$, denoted $B_p(K)$ and $Z_p(K)$ respectively.



$$B_p(K) \subseteq Z_p(K) \subseteq C_p(K)$$

Homology group

- The p-dimensional homology of a simplicial complex can then be described by the following **quotient group**:

$$H_p(K) = Z_p(K)/B_p(K)$$

Homology group

- The p-dimensional homology of a simplicial complex can then be described by the following **quotient group**:

$$H_p(K) = Z_p(K)/B_p(K)$$

- Note: a quotient group is a notion in abstract algebra that is beyond the scope of this course.
- But in a nutshell, by letting the boundaries $B_p(K)$ be denominator, we are discarding the effect of boundaries among the cycles, so that we only focus on the non-trivial cycles.

Homology group

- The p-dimensional homology of a simplicial complex can then be described by the following **quotient group**:

$$H_p(K) = Z_p(K)/B_p(K)$$

- Note: a quotient group is a notion in abstract algebra that is beyond the scope of this course.
- But in a nutshell, by letting the boundaries $B_p(K)$ be denominator, we are discarding the effect of boundaries among the cycles, so that we only focus on the non-trivial cycles.
- Also notice that $H_p(K)$ is also a vector space. And the homology basis we mentioned previously is indeed a **basis for the vector space** $H_p(K)$.
  - See: https://en.wikipedia.org/wiki/Basis_(linear_algebra)

Homology group

- The p-dimensional homology of a simplicial complex can then be described by the following **quotient group**:

$$H_p(K) = Z_p(K)/B_p(K)$$

- Note: a quotient group is a notion in abstract algebra that is beyond the scope of this course.
- But in a nutshell, by letting the boundaries $B_p(K)$ be denominator, we are discarding the effect of boundaries among the cycles, so that we only focus on the non-trivial cycles.
- Also notice that $H_p(K)$ is also a vector space. And the homology basis we mentioned previously is indeed a **basis for the vector space** $H_p(K)$.
  - See: https://en.wikipedia.org/wiki/Basis_(linear_algebra)
- BTW, the cardinality (number of elements) of the homology basis for the p-dimensional cycles is called the **p-th Betti number**, denoted β_p.

Betti number

- $\beta_1 = 1$

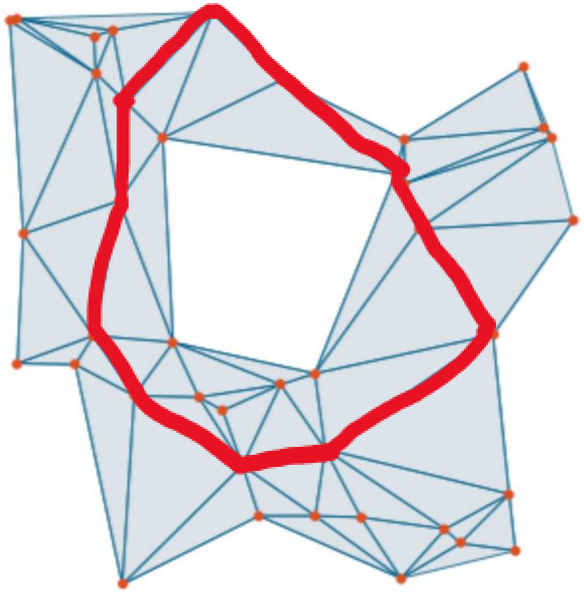


Image source: Yan et al. Persistence Landscape based Topological Data Analysis for Personalized Arrhythmia Classification

Betti number

- $\beta_1 = 2$