拓扑学与拓扑数据分析

齐天博

tianboqi.github.io

目录

第一部分　数学基础

1　点集拓扑学基础

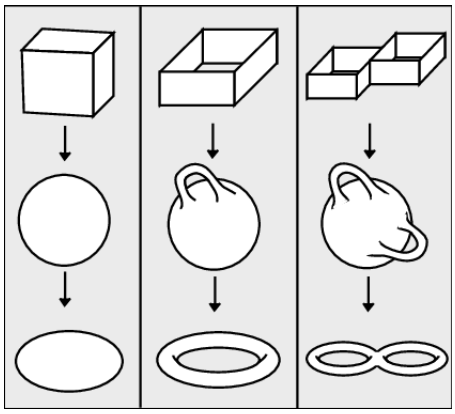
这一章，我们先来学习一些最基本的拓扑学概念和定理．这些知识都属于拓扑学中奠基的领域——点集拓扑学．

1.1　拓扑学基本概念

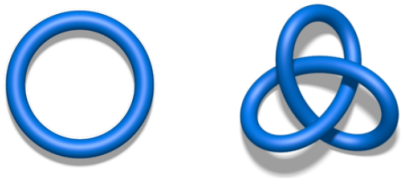
1.1.1　什么是拓扑学

拓扑学是数学中很重要的一个分支，它是一门研究几何体在连续变换下的不变量的学科．这个定义较为官方而难懂，而形象地来说，拓扑学常常被比喻为“橡皮泥的几何学”．所谓“连续变换” 就是捏橡皮泥的过程，在捏的过程中我们不能挖出新的洞，也不能将已有的洞填上．而拓扑学只关心这种过程中的“不变量”，例如橡皮泥有多少个洞、它们是“几维” 的等；而不关心捏橡皮泥时会变化的量，例如两点间的距离和由距离衍生出的面积、曲率等．

下图显示了几个连续变换的例子，可以发现这些变换过程中不会改变洞的数目．可以通过连续变换互相转化的几何图形称为拓扑等价的．因此有一个数学笑话叫做“拓扑学家分不清甜甜圈和咖啡杯”，这是因为它们在拓朴学上是等价的．



需要注意的是，这里所说的“连续变换” 并不指是一个连续形变的过程（也就是捏橡皮泥的过程）．当数学家谈到“变换” 时，正如线性代数里的线性变换，数学家指的是一个集合到集合的映射．映射只关心起点和终点，而不关心中间是怎么过去的．而所谓的“连续”，是指相邻的点也被映射到相邻的点．例如上图中左边和中间的图形在拓朴学上是不等价的，因为将左侧的图形变为中间的图形需要在中间挖一个洞．而这个洞两侧的点本来是相邻的，在映射后就不再相邻了．但下图中的环和扭结是拓扑等价的．虽然我们无法在不剪接圆环的情况下将它“连续地捏成” 右侧的扭结，但如果我们考虑二者之间的一个映射，那么相邻的点仍被映为相邻的点．因此二者可以通过一个连续变换联系起来，自然就是拓扑上等价的．



1.1.2　拓扑空间

从上一节我们已经看到，拓扑学关心的是连续变换．回顾一下我们在微积分课程中学习过 **连续函数**：函数 $f(x)$ 在 x_0 处点连续的定义为

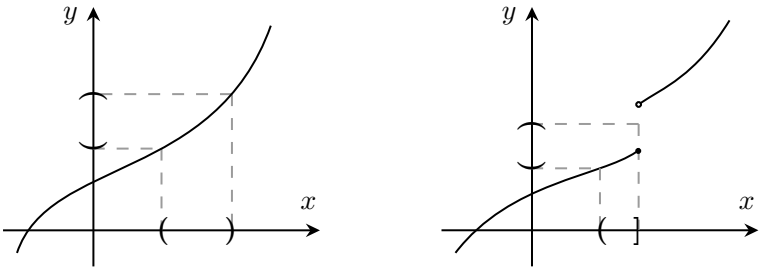
$$\lim_{x\to x_0} f(x) = f(x_0)$$

而 $f(x)$ 若在定义域的每一点处都连续，则称整个函数是连续的．这个定义非常直观，然而在拓扑学中并不适用． 我们的极限是以 ϵ-δ 语言定义的，这个定义涉及了两点之间的距离．然而拓扑学家不关心距离，因此他们压根就不想用距离的概念来定义连续性．所以，我们需要一种不涉及距离的方式来定义函数的连续性．

数学家定义数学结构时的准则是，不需要的概念就不定义，而不定义就不存在．

拓扑学家的确找到了一种等价的方式来定义连续函数，我们在此直接给出这个定义：一个函数若满足任意开区间的原像集一定是开区间或其并集，则称该函数是连续的．这个定义的出发点是开区间，而并不涉及距离的概念．关于为何开区间可以用来描述连续性，我们很难解释．不过直观上很容易验证这个定义的确和我们熟悉的连续函数定义是等价的．下图显示了连续函数和不连续函数下开区间的原像集．

给定 y 后，对应的 x 称为其 **原像**.



那么为什么一定要用开区间呢？能不能用闭区间呢？答案是不能．我们很容易想出反例：例如对于函数 $f(x) = \mathrm{e}^x$，若我们选择闭区间 $y \in [-1, 1]$，则显然对应的 $x \in (-\infty, 0]$ 不是闭区间．因此，开区间，而非闭区间，可以用来刻画连续函数．

有了以上的铺垫，我们就知道该如何定义拓扑学的研究对象了．拓扑学研究的是几何图形——也就是一个点集，上面需要定义与开区间类似的概念——开集．这就是所谓的拓扑空间．

定义 1.1: 拓扑空间

对于一个集合 X，若其一些子集被指定为 **开集**，且满足

- X 和 $\varnothing$ 都是开集，
- 有限个开集的交仍为开集，
- 任意个开集的并仍为开集，

则 X 称为一个 **拓扑空间**.

可以取无限个开集的并集，也为开集.

称所有的开集构成的集合族为 X 上的 **拓扑**，常常记作花体字母，如 $\mathcal{T}$.

定义中的这三条性质是对我们熟悉的实数上的“开区间及其并集” 的推广，当然这是在不涉及距离概念情况下的推广．我们下面仍然以实数集上的函数为例，分别来看一下为什么需要定义这三条．

- 第一条：对于一个实函数，若我们选择包含其值域的区间，则其原像集应为全集；而若我们选择其值域以外的区间，则其原像集应为空集．因此，全集和空集都应算作开集．

- 第二条：在实数集上，显然有限个开区间的交仍然是开区间，但无限个开区间的交集可能不再是开区间，例如

$$\bigcap_{n=1}^{\infty}\left(-1,\frac{1}{n}\right)=(-1,0]$$

- 第三条：对于一个实函数，开区间的原像可以是无限个开区间的并集．例如对于函数 $f(x)=\sin x$，区间 $(0,1/2)$ 的原像集为所有 $(2k\pi,2k\pi+\pi/6)$ 的并集．

看完这个定义后，我们来看几个拓扑空间的具体例子．

例 1.1. 对于一个集合 X，若只有 X 和 $\varnothing$ 为开集，则这种拓扑称为 **平凡拓扑**；若 X 的所有子集都为开集，则这种拓扑称为 **离散拓扑**．

例 1.2. 对于欧氏空间 $\mathbb{R}^n$，它常常自带度量结构，也就是两点之间具有“距离”的概念．我们把这种空间称为 **度量空间**．我们将两点 x,y 的距离记作 $d(x,y)$．则我们称集合

$$B(x,r)=\{y\,|\,d(x,y)<r\}$$

为以 x 为中心、r 为半径的 **开球**．将开球通过有限交、任意并得到的集合定义为开集，则这个空间称为一个拓扑空间，这种拓扑称为 **度量拓扑**．后面任何涉及欧氏空间或其他度量空间时，若不特殊说明，我们都默认使用度量拓扑．

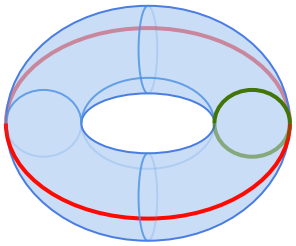
在实际应用，例如拓扑数据分析中，我们研究的几何体常常是嵌在某个欧氏空间 $\mathbb{R}^n$ 中的．我们关心的并不是整个 $\mathbb{R}^n$ 的拓扑，而是这个子空间的拓扑，例如欧氏空间里的一条曲线、一个曲面等．对于一个拓扑空间 X 和它的一个子集 Y，若我们将 X 中的开集与 Y 的交集定义为 Y 中的开集，即

$$\mathcal{T}_Y=\{U\cap Y\,|\,U\in\mathcal{T}_X\}$$

这样 Y 也成为了一个拓扑空间，称为 X 的一个 **子空间**，而 Y 上的拓扑称为 X 诱导出的子空间拓扑．下面让我们看几个 $\mathbb{R}^n$ 的子空间的例子．

例 1.3. $\mathbb{R}^n$ 的子空间 $\{x\,|\,d(x,O)=1\}$ 称为一个 n 维单位球面，记作 S^n．例如 S^1 就是单位圆，S^2 就是二维单位球面．

另一个常见而重要的拓扑空间是环面，一般 n 维环面记作 T^n．一维环面 T^1 实际上就是单位圆 S^1，而二维环面 T^2 有两个独立的“方向”，每个方向都是一个单位圆（下图中红线和绿线），因此它实际上就是 $S^1\times S^1$．n 维环面的几何意义不太好想象，但它可以被定义为 $T^n=(S^1)^n$．



两个集合 X 和 Y 的 **笛卡尔积** $X\times Y$ 定义为

$$\{(x,y)\,|\,x\in X,y\in Y\}$$

而其上的开集可以定义为 X 和 Y 各自的开集的笛卡尔积．

1.2　拓扑空间的等价关系

拓扑学关注的一个重要问题就是拓扑空间之间有多么相似，从而可以给拓扑空间进行分类．本节我们就来学习两种给拓扑空间分类的方式．

1.2.1　连续映射与同胚

有了前两小节的铺垫，我们已经准备好定义拓扑学中的第二个重要概念——连续映射了．这个定义正如我们前面预期的完全相同．

定义 1.2: 连续映射

设 X 和 Y 是两个拓扑空间．对于映射 $f : X \to Y$，若 Y 中任意开集的原像集一定为 X 中的开集，则称 f 是 **连续的**．

有了拓扑空间和连续映射的概念作为基础，我们就可以严格地定义我们在第一小节所直观介绍的“拓扑等价” 的概念了．经过了前面这两小节的铺垫，这个定义也已经是水到渠成了．

定义 1.3: 拓扑空间的同胚

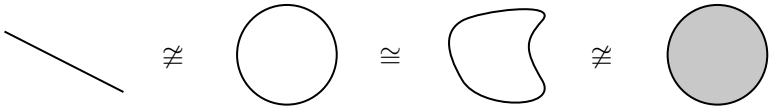
设 X 和 Y 是两个拓扑空间． 若在一个一一映射 $f : X \to Y$，使得 f 及其逆映射 f^{-1} 都是连续的，则称 X 和 Y 是 **同胚的** 或 **拓扑等价的**，记作 $X \cong Y$．称 f 为 X 和 Y 之间的一个同胚映射，简称同胚．

同胚和后面学到的同伦、同调等都没有统一的符号，这里只是挑选了一种较为常见的符号表示，但也有很多书不这么写．

我们在第一小节说过，连续变换的直观意义是“不能挖洞”．这里要求正向和逆向的映射都是连续的，就是说两个方向的变换都不能挖洞，也就相当于任意一个方向的变换既不能挖洞也不能填洞，因此两个拓扑空间的洞的数目一定是相等的．这就是同胚的直观意义．

下面，我们来看两个同胚的具体例子．

例 1.4. 在度量拓扑的意义下，任意简单闭合曲线都同胚于单位圆周 S^1．其同胚映射很容易构造出来．然而，一条线段（无论长短、开闭）不同胚与 S^1，单位圆盘（包含内部）也不同胚于 S^1．



例 1.5. 开线段 $(0,1)$ 是同胚于直线 $\mathbb{R}$ 的，例如我们可以选取映射

$$f : x \mapsto \ln \frac{x}{1-x}$$

这显然是一个一一的连续映射，且逆映射 $f^{-1} : (1+\mathrm{e}^{-y})^{-1} \mapsfrom y$ 也是连续的．因此这是一个同胚映射．

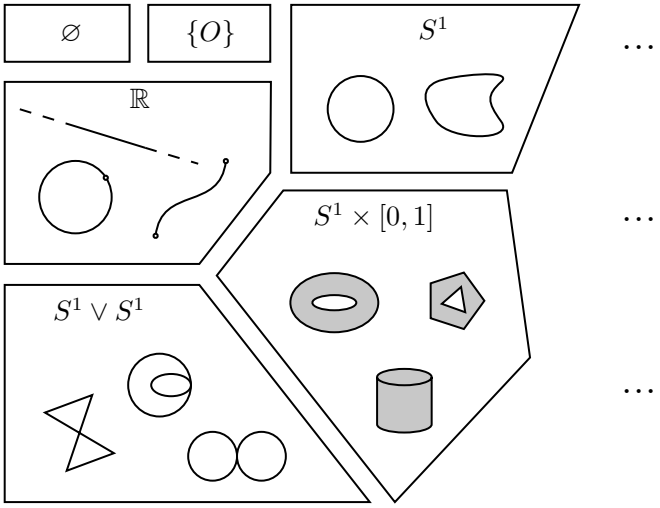
然而，闭线段 $[0,1]$ 是不同胚于直线 $\mathbb{R}$ 的．我们在这里不进行证明，但很容易发现两个端点是无法连续地映射到 $\mathbb{R}$ 上的．因此，拓扑空间的“边界” 也对其拓扑性质有着重要的影响．

证明闭线段不同胚于 $\mathbb{R}$ 需要用到 **紧致性** 的概念，这也是拓扑学中一个很基础的概念，不过我们在此不做介绍．

同胚为我们提供了对拓扑空间进行分类的依据．很容易发现，同胚是拓扑空间之间的一种 **等价关系**，也就是说它满足：

- 自反性：$X \cong X$；
- 对称性：若 $X \cong Y$ 则 $Y \cong X$；
- 传递性：若 $X \cong Y$，$Y \cong Z$ 则 $X \cong Z$．

等价关系的一个重要性质就是可以对数学对象进行分类，每个类别称为一个 **等价类**．例如在这里，我们可以依照是否相互同胚而将所有的拓扑空间分成互不相交的类别，每个类别称为一个 **同胚等价类**．例如下图显示了许多种不同的同胚等价类．拓扑学以及拓扑数据分析的最基本目的就是在给定一个拓扑空间后，找到它属于哪个等价类．



我们后面马上会看到拓扑空间之间还有其他的等价关系，从而有其他的不同分类方式．不过，同胚是其中最为严格的等价关系，它代表拓扑性质完全相同，而其他分类方式则是忽略掉了某种不同．因此，同胚等价类是最细的分类方式．

1.2.2　同伦

本节我们来介绍拓扑空间的另一种等价关系——同伦等价．同伦等价的定义稍复杂一些，让我们一点一点来看．首先，我们先介绍映射的同伦的概念．

定义 1.4: 映射的同伦

设 X 和 Y 是两个拓扑空间，连续映射 $f, g : X \to Y$．若存在一个单参连续映射 $H_t : X \to Y$（对参数 t 也连续），其中 $t \in [0,1]$，使得 $H_0 = f, H_1 = g$，则称映射 f 和 g 是 **同伦的**，记作 $f \simeq g$.

映射的同伦的定义很好理解，它是在描述两个映射可以“连续地” 由一个变为另一个（t 从 0 变为 1 的过程）．注意我们这里要求映射是连续映射，但并不要求是同胚映射，也就是说这个映射可以不是一一映射．

通过映射的同伦的概念，我们下面可以来定义拓扑空间的一种等价关系，也称为同伦．我们先给出定义．

定义 1.5: 拓扑空间的同伦

设 X 和 Y 是两个拓扑空间．若存在连续映射 $f : X \to Y$ 和 $g : Y \to X$，使得 $f \circ g$ 和 $g \circ f$ 都同伦于恒等映射，则称 X 和 Y 是 **同伦的**，记作 $X \simeq Y$.

$f \circ g$ 表示 g 和 f 的复合函数，即

$$(f \circ g)(x) = f(g(x))$$

这个定义看起来较为繁琐，但实际上它也是在描述两个拓扑空间可以“由一个连续地变为另一个”，只不过不再要求这种变化是一一映射了．为了更好地理解同伦的意义，让我们来看几个例子．

例 1.6. 开区间 $(0,1)$ 与闭区间 $[0,1]$ 同伦．例如我们可以选取如下的映射 f 和 g

$$f:(0,1)\to[0,1] \qquad f(x)=x$$
$$g:[0,1]\to(0,1) \qquad g(x)=\frac{2x+1}{4}$$

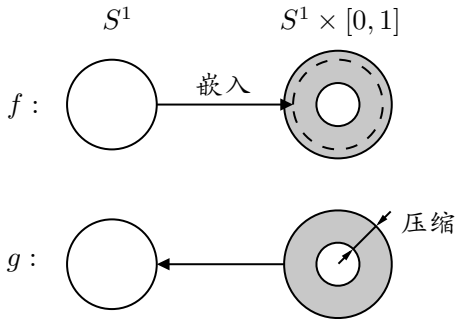
那么 $f\circ g$ 和 $g\circ f$ 都是 $x\mapsto\frac{2x+1}{4}$，只不过前者定义在 $(0,1)$ 上，后者定义在 $[0,1]$ 上．显然这两个函数都可以由恒等映射“压缩”而来，也就是说我们可以选取

$$H_t(x)=\frac{2x+t}{2t+2}$$

此时有 $H_0(x)=x$，$H_1(x)=\frac{2x+1}{4}$，因此 $f\circ g$ 和 $g\circ f$ 都同伦于恒等映射．这说明 $(0,1)$ 和 $[0,1]$ 的确是同伦的．

例 1.7. 有宽度的圆环 $S^1\times[0,1]$ 与单位圆 S^1 同伦．例如我们可以选取如下的映射 f 和 g，使得 f 将 S^1 嵌入到 $S^1\times[0,1]$ 里；而 g 则将 $S^1\times[0,1]$ 的宽度“压缩掉”，使其变为 S^1，如下图所示．

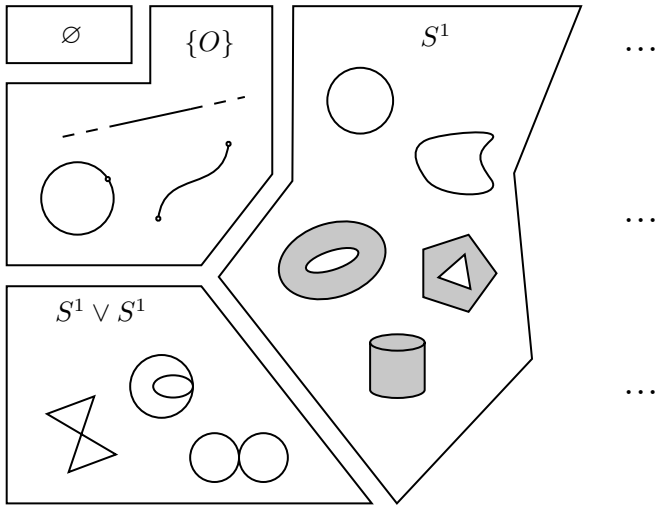
显然二者并不同胚．



那么很容易发现 $f\circ g$ 就是 S^1 上的恒等映射；而 $g\circ f$ 是将 $S^1\times[0,1]$ 的宽度“压缩掉”，像点位于嵌入在自身之中的 S^1 上，这个变换显然可以由其上的恒等变换连续“压缩”而来．因此，$S^1\times[0,1]$ 和 S^1 是同伦的．

这几个例子体现了同伦等价和同胚等价的不同．第二个例子体现的差异更为重要——同伦允许在不打洞或者挖洞的情况下，把一些连续的“维度”给“压缩掉”．因此如果我们利用同伦对拓扑空间划分等价类，则这种分类比同胚等价类要更加粗糙一些．也就是说许多种不同的同胚等价类会被归为同一个同伦等价类中，如下图所示．

在这里其实并没有维度的概念，这里说维度只是一个形象的比喻．



2　代数拓扑学基础

这一章，我们来利用代数学知识来研究拓扑空间，这个分支称为代数拓扑学．具体来说，我们只讨论代数拓扑学中最为简单的一个理论，称为 **单纯同调论**．

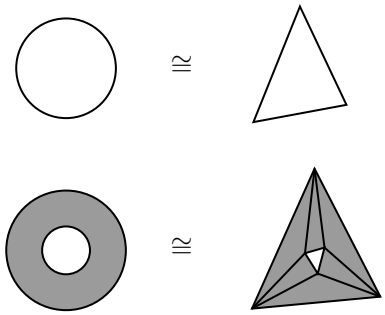
2.1　单纯形与复形

2.1.1　拓扑空间的三角剖分

我们之前说过，拓扑学的一个基本目的是给拓扑空间分类——根据等价关系，例如同胚和同伦将拓扑空间归为相应的等价类．

不过任意一个拓扑空间可以十分的复杂，因此一些拓扑学家选择从一些简单的几何图形入手．我们熟悉的比较简单的几何图形包括多边形、多面体等．这些几何图形虽然简单，但却在拓扑意义下却同胚于许多拓扑空间．因此，它们不失为很好的研究对象．

不过拓扑学家还是嫌多边形和多面体有点麻烦．于是，拓扑学家们把这些图形继续分割，直到分成了无法更简单的三角形和四面体等．于是，许多拓扑空间都可以同胚为由这些最基本的图形拼接而成的图形．这称为拓扑空间的 **三角剖分** 或 **单纯剖分**．

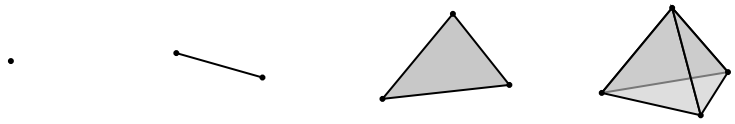


这样，我们就大大简化了研究的对象．现在，我们只需要研究这些最最基本的图形和它们拼接而成的图形的拓扑性质就行了．这样构造出的理论就是单纯同调论．

2.1.2　单纯形与单纯复形

下面，我们来严格化我们前面所说的这些几何对象．我们先从最基本的图形开始．0 维图形只能是点，1 维图形便是线段，2 维和 3 维就是我们上面提到的三角形和四面体了 … 这些简单的图形我们称为 **单纯形**，简称单形． 具体来说，一个 q 维单形是处于一般位置的 $q+1$ 个点构成的对象 $(a_0, a_1, \cdots, a_q)$．我们下面用填充灰色的几何体代表单形．

处于一般位置是指图形不退化，例如三角形的三个顶点不共线、四面体的四个顶点不共面等．

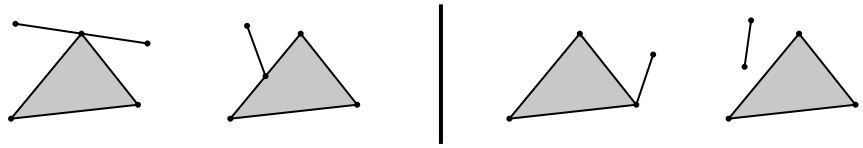


0 维单形　　1 维单形　　2 维单形　　3 维单形

为了后续计算的方便，我们在这里默认讨论 **定向单形**．也就是说，我们讨论的单形是区分正反方向的，如下图所示．在记号上，它体现在 $(a_0, a_1, \cdots, a_q)$ 中点的排列的奇偶顺序．例如 $(a_0, a_1, a_2) = -(a_0, a_2, a_1)$．

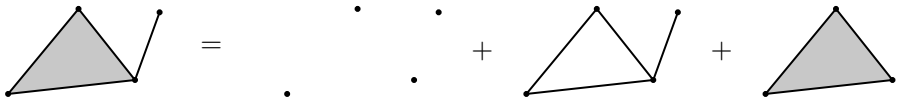
由单形可以构建更加复杂的几何体，称为 **单纯复形**，简称复形．复形就是我们研究的主要对象了．如上一小节所说，它可以代表许多复杂的拓扑空间．

不过，并不是任意摆放的单形都可以构成复形的，我们对它们的组合方式有一定的要求：如果两个单形相交，那么它们只能在共有的“面” 上相交，这里的面可以是任意维的．也就是说，两个单形要不然不相交，要不然只能共享顶点、边或者面等，而不能交于中间．如下图所示．



不是单纯复形　　单纯复形

构成复形的所有 q 维单形称为其 q 维部分．需要指出的是，一个复形的 q 维部分包括其中所有的 q 维单形，这也必须包括其所有子复形的 q 维部分．例如下图的复形中，其 1 维部分包含了 4 条边，其中包括了其子复形（三角形）的三条边．由于我们下面的定义会一直涉及复形的 q 维部分，因此这个概念必须严格．



0 维部分　　1 维部分　　2 维部分

2.2　单纯同调论

2.2.1　单纯链群

下面，我们就要开始使用线性代数的工具来描述单纯复形的拓扑结构了．这部分的定义的概念比较多，而且这些概念最初看似都很莫名其妙，等到最后单纯同调的几何直观会才显现出来．我们到那时再进行解释．

我们首先来定义第一个概念，称为单纯链与单纯链群．

定义 2.1: 单纯链群

设 K 是一个单纯复形，其 q 维部分的实系数线性组合称为一个 ***q*-链**． 所有 q-链构成一个线性空间，称为 K 的 ***q* 维单纯链群**，记作 $C_q(K,\mathbb{R})$．

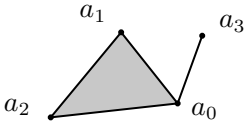
在没有歧义时也可以简写为 $C_q(K)$ 或 C_q．

需要指出的是， 这里面的线性组合只是形式上的，意思是我们假装单形是某种可以做加法和数乘的东西，然后按照运算规则进行计算，而不去纠结这种运算到底代表什么．如果你愿意，你也可以把它理解为“几个单形放在一起构成一个链”，不过它也没有什么更深的意思了．这里构造这种链是为了使它们构成一个线性空间，从而使用线性代数的工具．

单纯同调论中更常见用 $\mathbb{Z}$ 作为系数，此时 $C_q(K,\mathbb{Z})$ 不再构成线性空间，因此我们把它称为群而非线性空间．不过在我们这里没有太大区别，而用 $\mathbb{R}$ 做系数更容易讲．

下面我们来看一个具体的例子．

例 2.1. 考虑如下图所示的复形 K（其定向被略去了，因为不重要）



其各个维链群为

这里 $\mathbb{R}a+\mathbb{R}b$ 是指 $\{k_1a+k_2b \mid k_1,k_2\in\mathbb{R}\}$

$$
\begin{aligned}
C_0 &= \mathbb{R}a_0+\mathbb{R}a_1+\mathbb{R}a_2+\mathbb{R}a_3\\
C_1 &= \mathbb{R}(a_0,a_1)+\mathbb{R}(a_1,a_2)+\mathbb{R}(a_2,a_0)+\mathbb{R}(a_0,a_3)\\
C_2 &= \mathbb{R}(a_0,a_1,a_2)
\end{aligned}
$$

由于复形 K 中没有更高维的单形，因此其更高维的链群只能为 $\{0\}$，常常也直接被记作 0.

现在我们有了复形的一系列链群，那么这些链群之间有什么关系呢？将线性空间联系起来的自然是线性映射．那么链群之间有什么有用的线性映射呢？我们知道，高维单形是由低维单形包围成的，或者说低维单形构成了高维单形的边缘．这种关系就可以构造出一种线性映射，称为边缘同态．

同态是群论里的概念，在线性代数中它就是指线性映射.

定义 2.2: 边缘同态

相邻维度的单纯链群之间的 **边缘同态** $\partial_q : C_q(K,\mathbb{R}) \to C_{q-1}(K,\mathbb{R})$ 是一个线性映射，满足对于单形 $(a_0,\cdots,a_q)$ 有

有时也可以简写为 ∂.

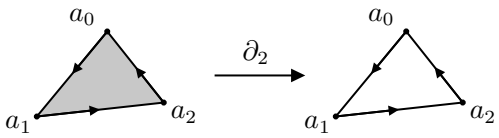
$$
\partial_q(a_0,\cdots,a_q)=\sum_{i=0}^{q}(-1)^i(a_0,\cdots,a_{i-1},a_{i+1},\cdots,a_q)
$$

这个定义看似吓人，但其实非常简单．上式的右侧的每一项 $(a_0,\cdots,a_{i-1},a_{i+1},\cdots,a_q)$ 就是单形中顶点 a_i 相对的那个面，而因子 $(-1)^i$ 只是增加了符号，也就是在调整这个面的定向．所以这个定义的的确确就代表了带定向的边缘．下面我们来看一个非常简单的例子．

例 2.2. 考虑 2 维单形 (a_0,a_1,a_2)，其边缘按照定义可以计算得

$$
\begin{aligned}
\partial_2(a_0,a_1,a_2) &= (a_1,a_2)-(a_0,a_2)+(a_0,a_1)\\
&= (a_0,a_1)+(a_1,a_2)+(a_2,a_0)
\end{aligned}
$$

很明显这就是这个单形的定向边缘.



不过，定义中只直接给出了单形的边缘，那么对于由单形组合而成的链呢？实际上任何链的边缘也被这个定义唯一确定了．这是因为，单形构成了单纯链群的一组基．知道了基在线性映射下的像，我们就可以组合出任意一个元素的像．我们再来看一个例子．

例 2.3. 考虑 1-链 $c=2(a_0,a_1)-(a_1,a_2)$，其边缘为

$$
\begin{aligned}
\partial_1 c &= 2\partial_1(a_0,a_1)-\partial_1(a_1,a_2)\\
&= 2(-a_0+a_1)-(-a_1+a_2)\\
&= -2a_0+a_1-a_2
\end{aligned}
$$

这样，我们就用边缘同态将这一系列单纯链群之间连接起来．我们可以把这种关系画成一个图，称为 **链复形**

$$\cdots \xrightarrow{\partial_{q+2}} C_{q+1} \xrightarrow{\partial_{q+1}} C_q \xrightarrow{\partial_q} C_{q-1} \xrightarrow{\partial_{q-1}} \cdots$$

2.2.2　单纯同调群

边缘同态可以帮助我们来分析链群的结构．首先，我们通过边缘同态可以定义出链群 $C_q(K,\mathbb{R})$ 的两个子空间．我们定义 $\boldsymbol{q}$ **维闭链群** 为线性映射 ∂_q 的核

$$Z_q(K,\mathbb{R}) = \operatorname{Ker}\,\partial_q$$

我们在线代里学过，所谓线性映射的核，是指所有被它映为单位元 0 的元素的集合，它是 $C_q(K,\mathbb{R})$ 的一个子空间．

另一个重要的子空间称为 $\boldsymbol{q}$ **维边缘链群**，它定义为 ∂_{q+1} 的值域

$$B_q(K,\mathbb{R}) = \operatorname{Im}\,\partial_{q+1}$$

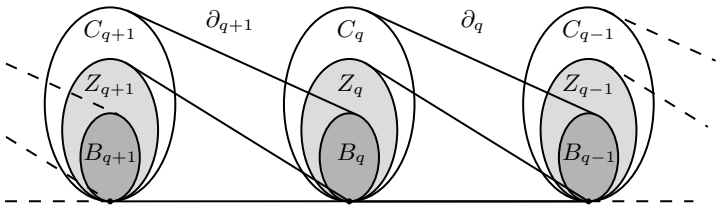
值域的概念不必多说．∂_{q+1} 的像位于 $C_q(K,\mathbb{R})$ 中，因此其值域也是 $C_q(K,\mathbb{R})$ 的一个子空间．

那么 q 维闭链群和边缘链群是什么关系呢？我们回顾一下上面的例 3.2. 和 3.3.，如果对这两个例子中求得的边缘再求一次边缘，都会得到 0 的结果．也就是说有

$$\partial_{q-1} \circ \partial_q = 0$$

这可以被形象地称为“边缘没有边缘”． 这说明任何边缘链也一定是一个闭链，也就是说 q 维边缘链群是 q 维闭链群的一个子空间．这几个链群的关系如下图所示．

也可以简记作 $\partial^2 = 0$



有了上面的基础，我们终于可以定义单纯同调论中最核心的概念——单纯同调群．

定义 2.3: 单纯同调群

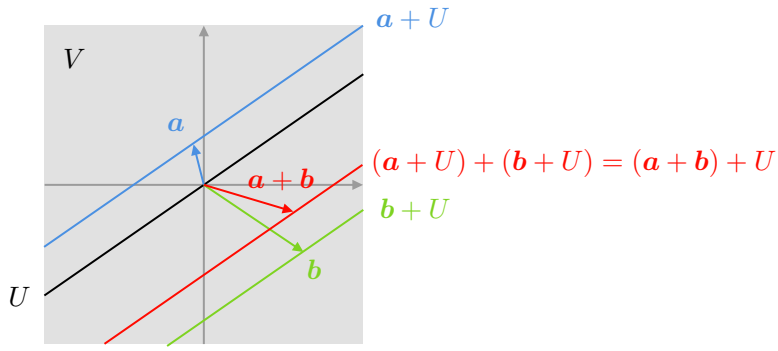
设 K 是一个单纯复形，其 q 维闭链群与 q 维边缘链群的商称为 $\boldsymbol{q}$ **维单纯同调群**，记作 $H_q(K,\mathbb{R})$．即

$$H_q(K,\mathbb{R}) = \operatorname{Ker}\,\partial_q / \operatorname{Im}\,\partial_{q+1}$$

由于许多线代课本中不会介绍商空间，因此我们在这里讲一下这个概念．熟悉这一概念的读者可以跳过这几段．对于一个线性空间 V 和它的一个子空间 U，我们定义它们的商空间 V/U 为所有“平行于 U 的超平面” 所构成的线性空间，即所有形如 $\boldsymbol{a}+U = \{\boldsymbol{a}+\boldsymbol{u} \mid \boldsymbol{u} \in U\}$ 的集合构成的空间，其中向量 $\boldsymbol{a}$ 代表这个超平面距离原点的偏离向量．而这些超平面之间的线性组合即定义为它们的偏离向量的线性组合，即

$$c_1(\boldsymbol{a}+U) + c_2(\boldsymbol{b}+U) = (c_1\boldsymbol{a} + c_2\boldsymbol{b}) + U$$

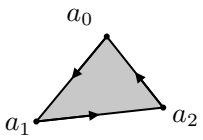
用几何直观表示出来如下图所示



商空间 V/U 把一个个平行于子空间 U 的超平面看作一个个元素，也就相当于将 U 所在的维度“压缩” 没了，只剩下了与 U 线性无关的方向．换句话说，对一个子空间求商的过程就是剔除掉这个子空间所含有的信息、保留其他信息的过程．我们后面会看到，单纯同调群的定义就是遵循这个思路．

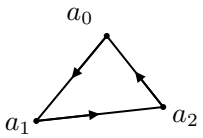
现在，我们终于可以回过头来看一看我们定义的同调群有什么直观的意义了．我们先来通过一个例子来解释一下闭链和边缘链都是什么意思．

例 2.4. 考虑下面的 2 维复形



我们在例 3.2. 中已经推过了，它的 1 维链 $(a_0,a_1)+(a_1,a_2)+(a_2,a_0)$ 是 (a_0,a_1,a_2) 的边缘，因此它是一个边缘链；同时它的边缘又为零，所以它也是一个闭链．

例 2.5. 而如果我们考虑下面的这个 1 维复形



这个复形的 1 维链 $(a_0,a_1)+(a_1,a_2)+(a_2,a_0)$ 显然仍然是一个闭链，因为它的边缘仍为 0，这一点没有变化．然而它不再是一个边缘链了，因为这个复形不包含 (a_0,a_1,a_2)．

上面这个例子说明，边缘链就是复形中某个高维链（也就是某个实心区域）的边缘，而闭链则是一个“可以作为边缘” 的链，只不过“以它作为边缘的那块区域” 还可能是空着的，也就是说可能是一个洞．这也是闭链的名字的来源——它是一个闭合的链，包裹着一些区域．因此闭链也被称为 **环**．

这样，用闭链群对边缘链群求商，就是在找不是边缘链的闭链，也就是在找“洞”．为了证实这一点，我们来求一下上面两个例子中两个复形的同调群．

例 2.6. 对于上面的第一个例子，即 2 维复形 (a_0,a_1,a_2)，很容易求出其闭链群和边缘链群

$$Z_0=\mathbb{R}a_0+\mathbb{R}a_1+\mathbb{R}a_2 \quad B_0=\mathbb{R}(a_0-a_1)+\mathbb{R}(a_1-a_2)+\mathbb{R}(a_2-a_0)$$
$$Z_1=B_1=\mathbb{R}[(a_0,a_1)+(a_1,a_2)+(a_2,a_3)]$$
$$Z_2=B_2=0$$

这样很容易得到

$$H_2 = H_1 = 0$$

而 H_0 稍稍难计算一些，但我们可以通过把 Z_0 和 B_0 同构为更简单的群，来计算商群的结构．由于 Z_0 有三个自由的整系数，因此有 $Z_0 \cong \mathbb{R}^3$．而 B_0 若重新整理一下，可以发现有效的系数只有两个，即 $B_0 \cong \mathbb{R}^2$．因此，可以发现

$$H_0 \cong \mathbb{R}^3/\mathbb{R}^2 = \mathbb{R}$$

在同构的意义下我们也可以直接写 $H_0 = \mathbb{R}$．

例 2.7. 对于上面的第二个例子，即 1 维复形 $(a_0, a_1) + (a_1, a_2) + (a_2, a_3)$，其非零的闭链群和边缘链群最高只能到 1 维，且 1 维边缘链群变为 0．即

$$Z_0 = \mathbb{R}a_0 + \mathbb{R}a_1 + \mathbb{R}a_2 \quad B_0 = \mathbb{R}(a_0 - a_1) + \mathbb{R}(a_1 - a_2) + \mathbb{R}(a_2 - a_0)$$
$$Z_1 = \mathbb{R}[(a_0, a_1) + (a_1, a_2) + (a_2, a_3)] \quad B_1 = 0$$

同样，在同构的意义下，我们很容易得到

$$H_0 = H_1 = \mathbb{R}$$

由于这个复形同胚于单位圆 S^1，因此单位圆的同调群与此相同．实际上可以证明，对于 n 维球面 S^n，其同调群为

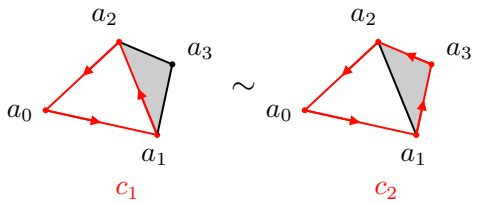
$$H_q(S^n, \mathbb{R}) = \begin{cases} \mathbb{R}, & q = 0, n \\ 0, & \text{其他} \end{cases}$$

这两个例子说明，同调群的确找拓扑空间中的“洞”．当有 q 维的洞时，这个维数的同调群 H_q 就会非零．我们来进一步看一下其中的数学逻辑．对于一个同调群 $H_q = Z_q/B_q$，其中的元素应该是这样的超平面

$$[z] = z + B_q$$

其中 z 是某个闭链，而 $[z]$ 即代表含有 z 的超平面，我们在这里也称之为 z 的 **等价类**．很容易发现，如果两条闭链之间相差的是边缘链的线性组合（也就是 B_q 中的元素），那么它们属于同一个等价类．而边缘链是某个是实心区域的边缘．因此，等价的两条链所包围的区域只能相差一些实心区域，而不能相差洞．也就是说，等价的两条链代表了复形中同一个洞．我们来看一个例子．

例 2.8. 下图显示了一个复形中的两条闭链



这两条闭链的差

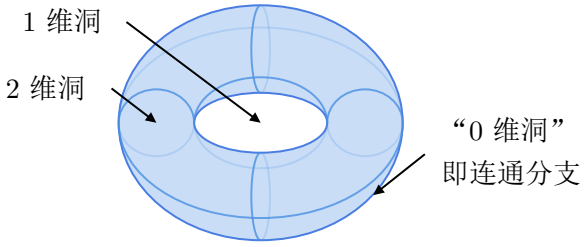
$$c_2 - c_1 = (a_1, a_3) + (a_3, a_2) + (a_2, a_1)$$

> 是一条边缘链，因此 c_1 和 c_2 是属于同一个等价类的，或者说它们是等价的．而可以发现，c_1 和 c_2 实际上包围的是同一个洞，只不过 c_2 还包围了一些实心的部分．从这个例子我们可以看出，一个等价类——也就是 H_q 中的一个元素——代表的是复形中的一个 q 维洞．不过更准确地说，应该是 H_q 里的每一个方向代表复形中的一个 q 维洞，因为 $2c_1$ 和 $2c_2$ 等等构成了一个不同的等价类，但是它们实际上仍然代表的是图中同一个洞．

上面的例子说明，同调群 H_q 里的各个不同方向——用线性代数的语言说，就是一组基——代表的是复形里所有的 q 维洞．这样，H_q 的维数就是 q 维洞的总数目，我们称这个数为这个拓扑空间的第 q 个 **Betti 数**．

$$\beta_q = \dim H_q$$

这里的“q 维洞”是指一个边缘是 q 维的洞，例如一个环面中心的洞是 1 维洞，而一个被二维曲面包裹的区域是一个 2 维洞．不过“0 维洞”的含义比较特殊，它指的是拓扑空间中连通分支，也就是代表了这个拓扑空间“分为多少个不相连的部分”．



最后，既然单纯同调群可以用来显示拓扑空间的拓扑结构，那么我们自然就会问，它能否告诉我们两个拓扑空间是否同胚？或者退一步，能否告诉我们拓扑空间是否同伦？很遗憾，答案都是否定的．我们有下面的定理．

定理 2.1: 同调群的同伦不变性

若两个拓扑空间同伦，则它们的同调群相同．

注意，这是一个充分条件而非充要条件，也就是说同调群相同是一个比同伦还弱的等价关系．因此，单纯同调论并不能完整地揭示拓扑空间的拓扑结构．不过，单纯同调群较为简单、方便计算，因此在实际应用中仍然非常常用．

第二部分　拓扑数据分析

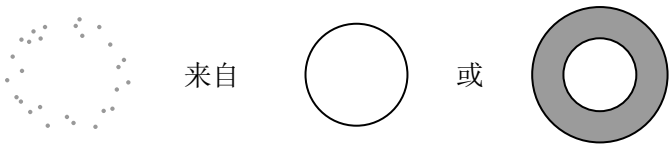
3　持续同调方法

3.1　数据的拓扑结构

从本章开始，我们将把关注点从纯数学的拓扑学转向它的一个重要应用——**拓扑数据分析 (TDA)**，即应用前面学习的拓扑学知识，来分析数据的拓扑结构．

我们在各类实验中收集到的数据，是来自某个未知 **总体** 的带有噪声的 **样本**．而由于变量的各个维度之间的非独立性，这个总体可能具有各种各样的结构．数据分析的目标则是通过样本数据推测总体的这些结构，例如聚类分析、降维等．而拓扑数据分析，就是在通过数据来推测总体的拓扑结构．例如，下面的这个数据可能来着一个具有环状拓扑结构的总体．

不过，由于数据通常在每个维度的方向都存在噪声，我们往往不能从其中得到总体的完整拓扑信息，哪怕我们有任意多的数据点．例如，我们上面提过的环形点云可能来自一个拓扑结构为一维圆 S^1 的总体，但也可能来自一个“有厚度的” 二维甚至高维的环形总体．这几种可能性是难以区分的．因此我们无法得知数据总体的同胚型，只能得到一些更弱的拓扑性质，例如同调型．这一章，我们就会使用前面学习过的单纯同调论来分析数据总体的同调型．



最后，我们来明确一下拓扑数据分析中的“数据” 是什么．我们的数据点往往都是实验采集得到的，它们以一组有限的数字——即坐标的形式记录．那么，数据就可以看作是高维空间的一个 **点云**．因此，在拓扑数据分析中我们接触的不再是抽象的拓扑空间，而是回到了某种类似于线性空间的、更为直观的空间中．这使得我们比基础的拓扑学有了更好的工具来分析数据的结构．

那么在拓扑数据分析中，数据所存在的空间具有什么结构呢？对于数据的拓扑结构，重要的并不是数据点的坐标，而是点之间的位置关系——我们可以把点云整体平移或旋转，这并不会影响其拓扑结构．具体来说，我们其实只需要点之间的距离信息即可．这样，数据点实际上是位于一个 **度量空间** 中．我们最为熟悉的度量空间就是定义了欧氏距离的欧氏空间．不过，我们完全可以以其他方式定义距离．我们在这里就不讨论其他种类的距离的定义了．

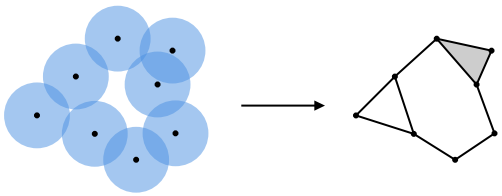
3.2　持续同调的理论基础

下面，我们来学习拓扑数据分析中一个重要的方法，称为 **持续同调**．持续同调及其变体现在成为了拓扑数据中最为主流的方法，几乎成为了拓扑数据分析的同义词．

3.2.1　从数据构建复形

持续同调方法使用单纯同调论来分析数据的拓扑结构．而单纯同调论分析的对象是单纯复形，因此我们首先需要把数据转化为一个复形．

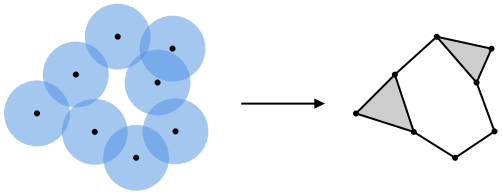
对于一个点云，我们需要以这些数据点为顶点，添加上各个维度的单纯形，从而构造出一个复形．很自然地想到，我们可以将临近的点连接在一起构成单形．若我们以某个给定的半径 ϵ 在每个点处做开球，并考虑它们之间的相交关系：对于选定的 k 个点，若这 k 个开球的总交集非空，则我们就把这 k 个点连成一个 $k-1$ 维单形．这样构造出的复形称为 **Čech 复形**．我们后面会专门讨论半径 ϵ 的问题．下面显示了从一个点云构造出的 Čech 复形．对于开球相交的两个点，我们会将它们连成一个 1 维单形．而对于 2 维单形，注意到左边的三个点处的开球的交集为空，因此它们不构成一个 2 维单形；而右上的三个点的开球有交集，因此它们构成一个 2 维单形．



我们如此精确地定义 Čech 复形，是因为它对点云的拓扑结构的保持有着严格的理论基础．**Čech 定理**，又称为 **神经定理** 指出，Čech 复形与所有开球的交集同伦等价．因此，在同伦的意义下，我们认为 Čech 复形可以在一定程度上代表数据的拓扑结构．

不过，Čech 复形的计算和存储过于复杂，其 k 维骨架的数目达到了 $\mathcal{O}(n^{k+1})$ 数量级．因此，在实际应用中我们常常使用一个简化的版本，称为 **Vietoris-Rips 复形**．它的构造方式与 Čech 复形类似，也是在每个点处构造以给定的 ϵ 为半径的开球，若对于选定的 k 个点，若这 k 个开球两两的交集都非空，则我们就把这 k 个点连成一个 $k-1$ 维单形．仍然以同样的点云和半径为例，这次左边的三个点两两相交，因此它们此时构成一个单形，如下图所示．

Vietoris-Rips 复形是拓扑数据分析里最常见的复形．除了它和 Čech 复形外，还有其他的构建复形的方式，例如 alpha 复形、witness 复形等，我们在此就不具体讨论了．



Vietoris-Rips 复形并不具有如Čech 复形一样的良好性质，它和开球的并集并不一定同伦等价．事实上，上面的例子所构造的Čech 复形和 Vietoris-Rips 复形显然就不等价．不过 Vietoris-Rips 复形代来的计算复杂度提升非常巨大，因此比Čech 复形常用得多．

3.2.2　持续同调

我们前面讨论了数据点云在给定半径 ϵ 后可以构建出一个复形，进而可以通过单纯同调论来分析复形的拓扑结构．然而，我们并不知道哪个半径所构建出的复形最能体现数据总体的拓扑结构．持续同调便是解决这个问题的一个方法．所谓持续同调，就是随着我们逐渐增加半径 ϵ，我们来看哪些拓扑结构可以持续存在．

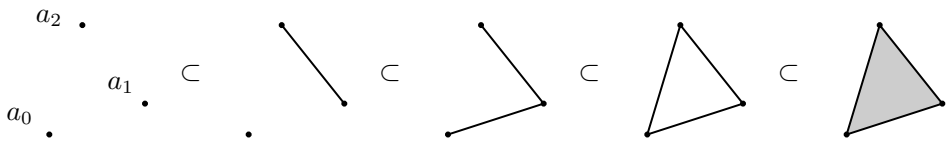
下面我们来看一看持续同调的数学原理．对于一个数据点云，我们可以在给定半径 ϵ 后构建出一个 Vietoris-Rips 复形．随着半径的增大，更多的点被连接起来，这些复形也会

增长，也就是会逐渐往里添加新的单形．我们称这一系列复形按照半径由小到大的顺序排列成一个 **滤子**．

$$K^0 \subset K^1 \subset \cdots \subset K^{n-1} \subset K^n$$

其中 K^0 就是数据点云，而 K^n 则是所有点连接成的一个最大的单形．不过注意我们需要让单形一个一个地进入复形，也就是说相邻的两个复形之间之应该相差一个单形．如果复形后加入某个单形后会自动导致新的单形的加入，那么我们也需要把这两个单形按顺序分开加入复形．

例 3.1. 考虑下面的三个点在半径 ϵ 逐渐增加时构成的滤子



当我们把 (a_0, a_2) 加入复形时，(a_0, a_1, a_2) 也会自动被加入复形，否则无法构成 Vietoris-Rips 复形．不过，在滤子中，我们仍然要求将这两个单形依次加入．

将复形连接成滤子后，这些每个复形又都有自己的各个维链群，同维度的链群之间自然也有类似的包含关系．

C_q^i 是 $C_q(K^i)$ 的简写，下面的 H_q^i 同理．

$$C_q^0 \subset C_q^1 \subset \cdots \subset C_q^{n-1} \subset C_q^n$$

这样，我们可以很自然地构造出链群之间的一个嵌入映射 $C_q^i \hookrightarrow C_q^j, (i \leq j)$，它把 K^i 的一个 q 维链映射到 K^j 的完全相同的 q 维链．这个映射会自然地诱导出同调群之间的一个映射

带勾的箭头 $\hookrightarrow$ 表示嵌入映射．

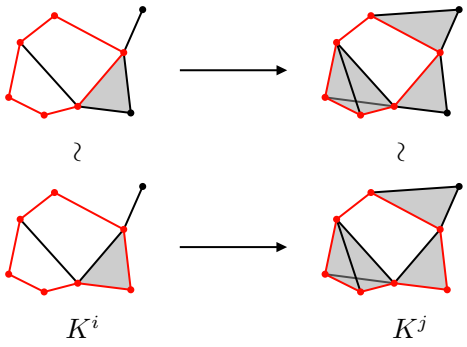
回顾一下 2.2.2. 中讲的，两条链（同调）等价的意思是它们之间相差的是一个边缘链．

$$\begin{aligned} \phi_q^{i,j}: \quad & H_q^i \to H_q^j \\ & c + B_q^i \mapsto c + B_q^j \end{aligned}$$

我们来解释一下这个映射．如果 C_q^i 中的两条 q-链是等价的，则它们被映到 C_q^j 以后仍然是等价的，这是因为它们相差的是一条边缘链，而这条边缘链在 C_q^j 中仍然是边缘链，因为从 C_q^i 到 C_q^j 只往里添加了新的链，而没有“挖洞”．这说明我们可以把 C_q^i 中的等价类（即 H_q^i 中的元素）整个映射为 C_q^j 中的等价类（H_q^j 中的元素）．这就是我们定义的 $\phi_q^{i,j}$．

例 3.2. 我们下面画出了一个滤子中的两个复形，并标出了其中等价的两个单形．可以发现，它们在 C_q^i 和 C_q^j 中都是等价的，因为它们相差的部分（右下的三角形的边缘）一直是一个边缘链，这是因为右下的三角形从 K^i 到 K^j 不会消失．

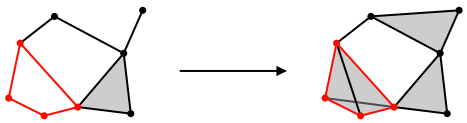
我们用 $\sim$ 表示链之间的等价关系．



这样，我们就可以把 C_q^i 中的整个等价类映射为 C_q^j 中的整个等价类，这就是 $\phi_q^{i,j}$．这

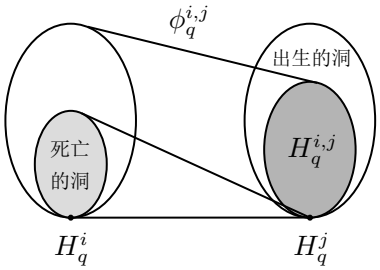
个映射可以体现出同调群的变化．例如下面这个单形在 C_q^i 中不等价为 0，但是在 C_q^j 中却等价为 0 了． 或者说，它在 H_q^i 中所在的等价类在 H_q^j 中被映为了 $[0]$．这就体现出它内部的洞被填上了．

注意到 $[0]$ 就是所有边缘链构成的等价类．



K^i　　K^j

上面的例子可以看出，$\phi_q^{i,j}$ 可以记录洞被填上（称为洞的 **死亡**）的过程．而实际上，洞产生的过程（称为洞的 **出生**）也体现在了同调群 H_q^i 随着 i 变化的过程中．如果 H_q^j 中出现了某个不存在于 H_q^i 的等价类，那么这个等价类应该代表一个刚出生的洞．让我们来看下面这个示意图．注意这里的出生和死亡都是发生在 i 到 j 之间的．



这样，我们就可以定义 **持续同调群**．从上面的示意图可以看出，它代表的是所有在 i 时已经出生、且到 j 时还未死亡的 q 维洞．

$$H_q^{i,j} = \operatorname{Im} \phi_q^{i,j}$$

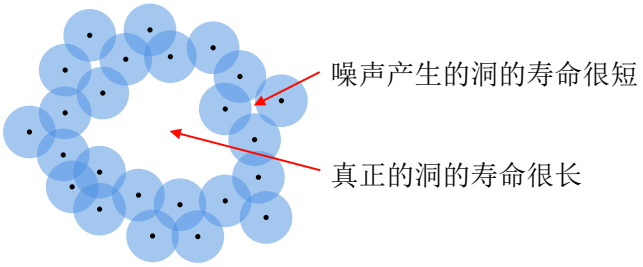
接下来，我们很自然地可以数出这些洞的数目，即“持续 Betti 数” $\beta_q^{i,j} = \dim H_q^{i,j}$．通过它，我们进而可以很容易地求得在 i 时出生、在 j 时死亡的 q 维洞的数目

$$\mu_q^{i,j} = (\beta_q^{i,j-1} - \beta_q^{i,j}) - (\beta_q^{i-1,j-1} - \beta_q^{i-1,j})$$

注意在半径 ϵ 逐渐增大的过程中，在同一时刻只可能连接起来一对数据点．而这至多会围出一个洞，或它产生的高维复形至多会填上一个洞． 所以当 i 每增加 1 时，至多有一个洞出生、一个洞死亡．这进一步意味着至多有一个在 i 时出生，在 j 时死亡的 q 维洞，也就是说 $\mu_q^{i,j}$ 只可能为 0 或 1．而 使得 $\mu_q^{i,j} = 1$ 的 (i,j) 记录了所有洞的生卒时间．

我们不考虑两对或多对数据点的距离完全相同的情况，这很难碰到．即使碰到了，我们也可以随便给它们排一个顺序．

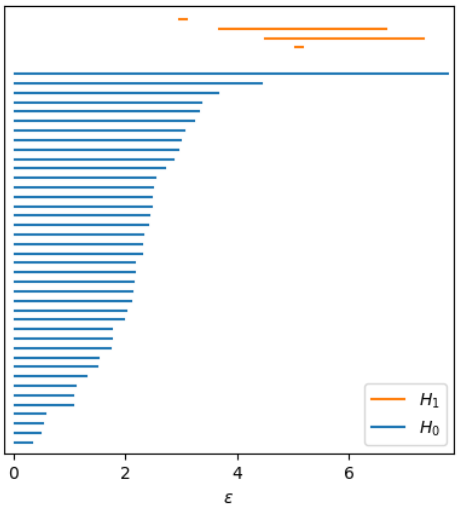
那么各个洞的生卒时间如何体现总体的拓扑结构呢？很容易想象，当我们从零逐渐增加半径 ϵ 而将临近的点连在一起时，它们首先会围成一些洞，然后这些洞又会逐渐被填上．对于由随机噪声产生的小的洞，它们会很快被填上，因此寿命会比较短．而数据总体拓扑结构中的洞则比较大，因而寿命会更长．因此，寿命较长的洞更有可能是总体结构的中的洞，而这些洞的数目体现了总体真正的拓扑结构．不过，持续同调中目前没有很好的统计方法来处理洞的寿命，也就是说我们无法说寿命多长才算长，只能比较主观地判断．这也是持续同调目前的一大弱点．



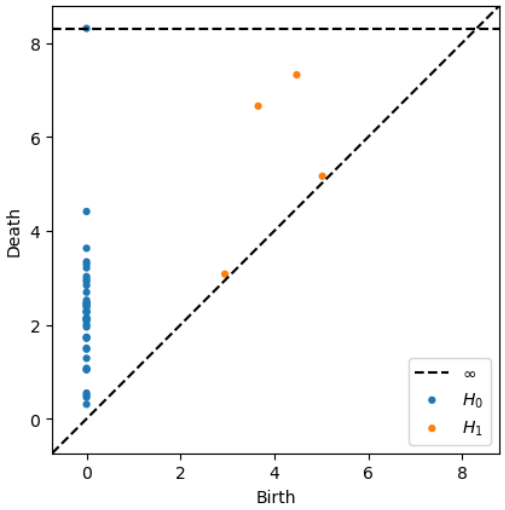
最后，我们来看一下拓扑数据分析常用的可视化方法．通常，我们在意的是构建复形时的逐渐增加的半径 ϵ，而不是复形的序号．因此我们会从 (i,j) 逆回去找到每个洞的出生和死亡时的半径 (ϵ_i,ϵ_j)．我们可以以 ϵ 为横坐标，把每个洞存活的时间段画成一条横线．这样画出来的图叫做 **条形码**．在条形码中横线的长度表示洞的寿命．或者我们可以以出生时间为横坐标，死亡时间为纵坐标，将每个洞表示成一个点．这样的图称为 **持续图**．在持续图中，所有的点都位于从原点向右上的 45° 直线左上方，而洞的寿命体现在点到这条直线的距离，越靠左上的点代表寿命越长的洞．

当 ϵ 足够大时，整个数据会连成一整个连通分支，因此一定会有一个寿命无限长的 0 维洞．我们只需要选择一个足够大的数作为它的死亡时间进行作图即可．

例如，下图显示了同一个点云的条形码和持续图．从图中可以读出，它有一个寿命较长的 0 维洞（连通分支）和两个寿命较长的一维洞，这说明整个数据应类似一个“8” 的形状．



条形码



持续图

3.2.3　$\mathbb{Z}_2$ 上的持续同调及其计算

我们前面以很数学的方式建立了持续同调论．不过对于拓扑数据分析，我们关注的并不是同调群或者持续同调群本身，而是从它的维数定义出来的各个维度的洞的寿命．计算持续同调群再计算它的维数的过程太复杂，因此下面我们来看一个直接计算持续 Betti 数的方法．这也是计算机计算持续同调的算法．

首先，由于计算机计算上的一些便捷性，最常见的持续同调算法使用的是以 $\mathbb{Z}_2$ 而非 $\mathbb{R}$ 或 $\mathbb{Z}$ 为系数的同调群．它们在我们介绍的知识范围内区别不大，不过我们下面还是来看一下 $\mathbb{Z}_2$ 上的同调群．$\mathbb{Z}_2$ 和 $\mathbb{R}$ 类似，也是一个数域，不过它里面只有两个数 $0,1$，它们之间的加法和乘法定义为

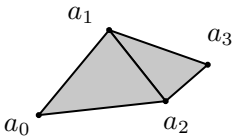
一些地方也会把 $\mathbb{Z}_2$ 写作 $\mathbb{Z}/2\mathbb{Z}$ 或 $\mathbb{F}_2$．

$$0+0=1+1=0,\qquad 0+1=1+0=1$$
$$0\times 0=0\times 1=1\times 0=0,\qquad 1\times 1=1$$

而接下来我们可以按照前面单纯同调论和持续同调论中完全相同的思路定义各个概念，只是将 $\mathbb{R}$ 替换成 $\mathbb{Z}_2$．也就是说，我们定义以 0 和 1 为系数的所有单形的线性组合构成单纯链群 $C_q(K,\mathbb{Z}_2)$，以同样的方式定义单纯同调群 $H_q(K,\mathbb{Z}_2)$．对于一个滤子，我们也以同样的方式定义持续同调群 $H_q^{i,j}$，最终求得各个洞的寿命．这和以 $\mathbb{R}$ 为系数的同调论最终求得的结果是完全相同的．

接下来我们回过头来看一下 Betti 数的计算．我们首先来看简单的单纯同调论．注意到，链群是线性空间，而它们之间的边缘同态 ∂_q 是线性映射．那么线性代数的知识告诉我们，如果我们给各个链群选择一组基——很自然地可以选择该维度的所有单形作为一组基——我们就可以把边缘同态写成矩阵的形式．

例 3.3. *考虑下面的复形*



对于它的链群 C_1 和 C_2，我们可以分别选择基 $[(a_0,a_1),(a_0,a_2),(a_1,a_2),(a_1,a_3),(a_2,a_3)]$ 和 $[(a_0,a_1,a_2),(a_1,a_2,a_3)]$．那么边缘同态 $\partial_2 : C_2 \to C_1$ 在这组基下可以写成矩阵

真正的矩阵只包含数的部分，我们在这里只是把对应的基写成了表头而已．

$$\partial_2 = \begin{array}{c|cc} & (a_0,a_1,a_2) & (a_1,a_2,a_3) \\ \hline (a_0,a_1) & 1 & 0 \\ (a_0,a_2) & 1 & 0 \\ (a_1,a_2) & 1 & 1 \\ (a_1,a_3) & 0 & 1 \\ (a_2,a_3) & 0 & 1 \end{array}$$

把边缘同态写成矩阵以后，我们就可以用矩阵理论来计算 Betti 数了． 由 Betti 数的定义很容易得到

由线代知识很容易知道

$\dim U/V = \dim U - \dim V$

$$\beta_q = \dim Z_q - \dim B_q$$

而 rank 表示矩阵的秩，定义为

$\operatorname{rank} M = \dim \operatorname{Im} M$

其中 $\dim B_q = \dim \operatorname{Im} \partial_{q+1}$ 就是 $\operatorname{rank} \partial_{q+1}$，而 $\dim Z_q = \dim \operatorname{Ker} \partial_q = \dim C_q - \dim \operatorname{Im} \partial_q$ 也可以用同样方式计算．这样，计算 Betti 数就化为了计算矩阵的秩．

计算机计算矩阵的秩通常是通过初等行和列变换将矩阵化为一个只有在从左上角向右下 45° 的线上非零的矩阵，称为 **Smith 标准型**． 由线代的知识可以很容易发现，Smith 标准型的非零元素数就是其秩，进而计算出我们想要的 Betti 数．

初等行/列变换是指交换两行/列，或把一行/列加到另一行/列上．注意做加法时是在 $\mathbb{Z}_2$ 上进行计算的．

$$\partial_q = \begin{array}{c} \overbrace{}^{\dim B_{q-1}}\ \overbrace{}^{\dim Z_q} \\ \left[\begin{array}{ccc|cc} * & & & & \\ & \ddots & & & \\ & & * & & \\ \hline & & & & \\ & & & & \end{array}\right] \end{array}$$

下面我们进一步来看一看持续同调的计算．持续同调中的持续 Betti 数也可以用矩阵化简的方式进行计算，并且不同维度的持续 Betti 数可以通过对一个矩阵的一次化简直接得到．我们来看一下这个算法．这个算法的原理不是很好理解，我们在此就不费力解释了．

首先，我们可以把不同维度的边缘同态合并成一个大的同态．然后我们可以把整个滤子中所有的单形作为一组基，把这个大的同态写成一个大的矩阵．而单形作为基的顺序就是它们进入复形的顺序．持续同调的算法是通过列变换将这个矩阵化简为“列上阶梯矩阵”．不过此处的列变换并不是任意的，而是只能用比自己靠前的列进行组合，也就是说

只能和在自己之前已经出现了的单形进行组合，并且也不能交换两列．具体来说，这个算法如下所示，其时间复杂度不超过 $\mathcal{O}(n^3)$.

Algorithm Matrix Reduction

for $j = 1 \cdots n$ **do**
  while $\exists j_0 < j$ 满足 $\mathrm{low}(j_0) = \mathrm{low}(j)$ **do**
    第 j 列 $\leftarrow$ 第 j 列 + 第 j_0 列
  end while
end for

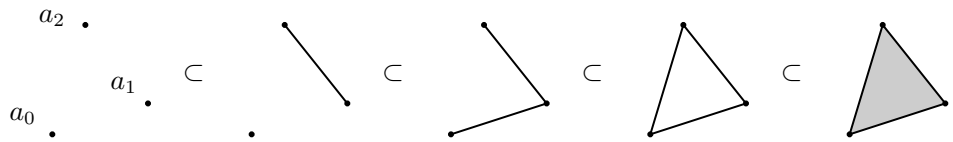
$\mathrm{low}(j)$ 表示第 j 列的最下非零元素所在的行数.

这个算法化简得到的矩阵称为“列上阶梯矩阵”，它的特点是没有任何两列（非零的两列）的最下元素所在的行相同．并且，每列最下元素的行数从左到右会越来越靠下，但其行数一定不会超过列数.

得到了这个矩阵后，所有的洞的生卒时间都可以从它的每列的最下元素读出来——对于每一个非零列的最下元素，若它位于第 i 行、第 j 列，则它代表着一个从 i 活到 j 的洞．而每个洞的维数可以通过它的列所对应的单形的维数读出—— n 维单形的列对应着 $n-1$ 维的洞．最后，我们还要再补上一个从 1 活到 ∞ 的零维洞．这样就完成了持续同调的计算.

这段文字解释或许有些绕．下面让我们看一个简单的例子.

例 3.4. 考虑一个最简单的由三个点连接成的滤子



它的单形作为基的顺序应该是（我们这里把给它们做了编号，方便下面对应上矩阵的行列数）

1	2	3	4	5	6	7
(a_0)	(a_1)	(a_2)	(a_1,a_2)	(a_0,a_1)	(a_0,a_2)	(a_1,a_2,a_3)

这样，整个边缘同态在这组基下可以被写作下面左侧的矩阵，而这个矩阵经过上面的算法后可以得到右侧的矩阵.

我们在此标出了行和列数，并且矩阵内只写出了非零元素.

$$\partial = \begin{array}{c|ccc:ccc:c} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 1 & & & & 1 & & 1 & \\ 2 & & & & 1 & 1 & & \\ 3 & & & & & 1 & 1 & \\ \hdashline 4 & & & & & & & 1 \\ 5 & & & & & & & 1 \\ 6 & & & & & & & 1 \\ \hdashline 7 & & & & & & & \end{array} \quad \rightarrow \quad \begin{array}{c|ccc:ccc:c} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 1 & & & & 1 & & & \\ 2 & & & & \mathbf{1} & 1 & & \\ 3 & & & & & \mathbf{1} & & \\ \hdashline 4 & & & & & & & 1 \\ 5 & & & & & & & 1 \\ 6 & & & & & & & \mathbf{1} \\ \hdashline 7 & & & & & & & \end{array}$$

我们加粗标出了化简的矩阵中每个非零列的最下元素．从这些位置可以读出来：有两个零维洞的生卒时间是 $(2,4)$ 和 $(3,1)$，再补上一个 $(1,\infty)$ 的零维洞；还有一个一维洞的生卒时间是 $(6,7)$.

3.3　持续同调算法实践

以上都是持续同调方法的数学理论．下面，我们使用 Python 来实现一个数据的持续同调分析．

3.3.1　数据准备

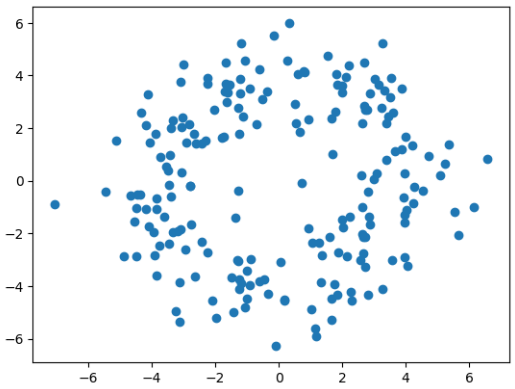
首先，我们来生成一个构成 1 维圆 S^1 的二维数据．我们将使用这个数据来试验持续同调算法．

```
import numpy as np

n = 200      # number of data points
a, b, r = 0.0, 0.0, 4.0      # parameter of the circle

np.random.seed(2023)      # set random seed
theta = np.linspace(0, 2.0*np.pi, n)

x = a + r*np.cos(theta) + np.random.normal(0,0.5,n)
y = b + r*np.sin(theta) + np.random.normal(0,0.5,n)

data = np.hstack([x.reshape((-1,1)), y.reshape((-1,1))])
```

这样我们就生成了我们的试验数据 `data`．这是一个 $n \times 2$ 的 2 维数组，每一行代表一个数据点．我们可以把它画出来．

```
import matplotlib.pyplot as plt

plt.scatter(data[:,0], data[:,1])
plt.show()
```

输出应如下图所示．点云很明显构成一个环形．我们下面就来试验一下持续同调算法能否给出环形所具有的拓扑结构．



3.3.2　调包实现

下面，我们使用现成的库来实现一个简单的持续同调分析． 我们使用的是 scikit-tda 下的 Ripser 库，在 conda 环境下可以通过以下代码来安装．

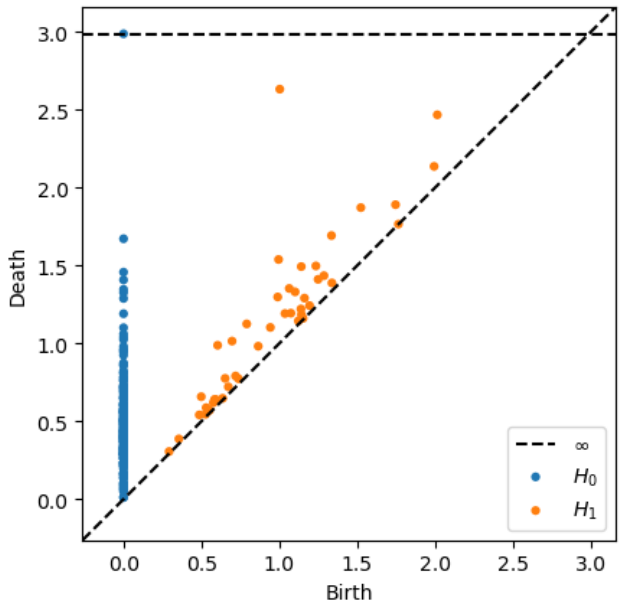
```
conda install -c conda-forge ripser
```

Ripser 的官网文档见本笔记的最后一页．另外，Ripser 使用的是持续同调的一种变种，称为 **持续上同调**，我们在此就不介绍它的数学原理了．

安装完成后，我们就可以调用 Ripser 来分析我们的试验数据了．Ripser 支持与 scikit-learn 类似的语法，我们可以用下面这几行简单的代码实现对数据的持续同调分析．

```
from ripser import Rips

rips = Rips()
persistence = rips.fit_transform(data)
rips.plot(persistence)
```

函数 `rips.plot()` 的输出是持续图．上面代码输出的持续图如下图所示．



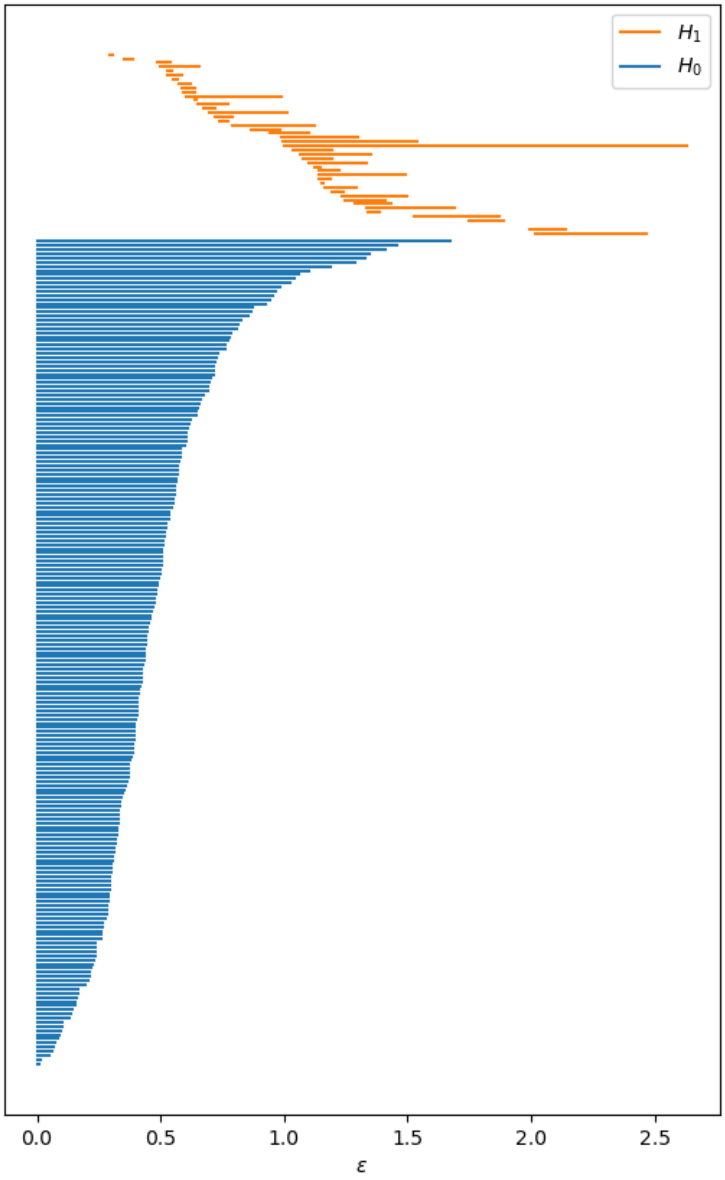
可以看到，我们的数据中有一个寿命较长的 0 维洞（即一个连通分支）和一个寿命较长的 1 维洞．这就代表其拓扑结构同调等价于一个一维圆 S^1，与我们的数据相符．

另外，Ripser 库不自带画条形码的函数．不过上面输出的 `persistence` 是一个由数组构成的列表，它存储了各个维度的洞的出生与死亡时的 ϵ．我们可以自己写一个函数画出数据的条形码．

```
def plot_barcode(persistence, figuresize=(6.4,4.8)):
  from matplotlib.lines import Line2D
  dim = len(persistence)
  cmap = plt.rcParams['axes.prop_cycle'].by_key()['color']
  start = 0
  plt.figure(figsize=figuresize)
  for n,pers in enumerate(persistence):  # each dimension
    for i,row in enumerate(pers):        # each hole of this dim
      plt.plot(row, [start+i,start+i], color=cmap[n])
    start += pers.shape[0]
  plt.tick_params(left=False, right=False, labelleft=False)
  plt.xlabel('$\epsilon$')
  plt.legend(
    [Line2D([0],[0],color=cmap[n]) for n in reversed(range(dim))],
    ['$H_{%i}$' % n for n in reversed(range(dim))])
  plt.show()
```

这样，通过调用函数 `plot_barcode(persistence, figuresize=(6.4,10))`，我们可以画出如下的条形码.

最长的 0 维洞的死亡时间是 inf，我们的画图函数中没有处理这个点，因此无限长的那根 0 维条码是没有画出来的.



从这个条形码中同样可以看到一个寿命较长的 0 维洞和一个寿命较长的 1 维洞，与上面的结论相同.