Topological Persistence (a.k.a. persistent homology)

① Filtrations and persistence modules:

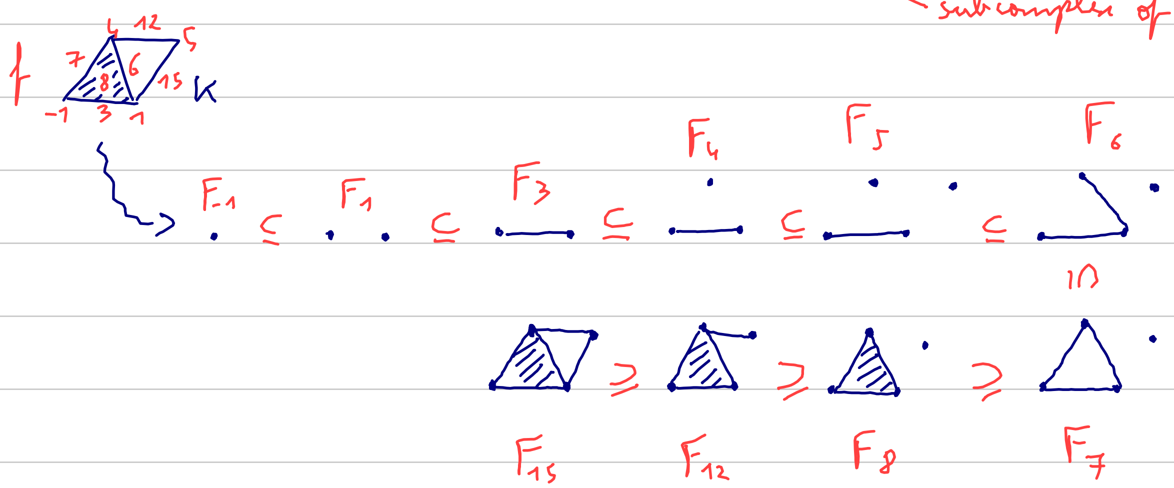
Let $T \subseteq \mathbb{R}$ be a fixed index set.

Def.: A filtration over T is a family $F = (F_t)_{t\in T}$ of nested topological spaces:

$$\forall s \leq t \in T, \quad F_s \subseteq F_t.$$

Examples:

* Sublevel sets of a function $f: X \to \mathbb{R}$:
$$\forall t \in T = \mathbb{R}, \quad F_t := f^{-1}((-\infty, t])$$
* offsets of a compact set $P \subset \mathbb{R}^d$ (i.e. sublevel sets of its distance function):
$$\forall t \in T = \mathbb{R}, \quad F_t = P^t := d_P^{-1}((-\infty, t]) = \{x \in \mathbb{R}^d \mid \min_{p\in P} \|x-p\| \leq t\}$$
* simplicial filtration: given a finite simplicial complex K, and $f: K \to \mathbb{R}$ s.t. $f(\tau) \leq f(\sigma)$ $\forall \tau \subseteq \sigma \in K$, let $T := \operatorname{Im} f$ and $F_t := f^{-1}((-\infty, t]) \subseteq K$.
subcomplex of K



↳ Goal: encode / summarize the evolution of the topology throughout F, for t ranging over T → use homology.

(filtration $\xrightarrow{H_*(\cdot\,;k)}$ persistence module)

Example: Let F be a filtration over T. Apply (singular) homology over a fixed field k to F:

$$\begin{cases} \forall t \in T, \text{ get a } k\text{-vector space } H_*(F_t). \\ \forall s \leq t \in T, \text{ get a } k\text{-linear map } H_*(F_s) \to H_*(F_t) \\ \qquad \text{induced by the inclusion } F_s \hookrightarrow F_t \end{cases}$$

↳ by functoriality of H_*, we have:

$\forall t \in T$, $H_*(F_t) \to H_*(F_t)$ is $id_{H_*(F_t)}$

$\forall s \leq t \leq u \in T$, the following triangle commutes:

$$\begin{tikzcd} H_*(F_s) \arrow[r] \arrow[dr] & H_*(F_t) \arrow[d] \\ & H_*(F_u) \end{tikzcd}$$

i.e. the map $H_*(F_s) \to H_*(F_u)$ is obtained by composing $H_*(F_s) \to H_*(F_t) \to H_*(F_u)$.

Let k be a fixed field.

Def: A persistence module over T is a family $M = (M_t)_{t \in T}$ of k-vector spaces connected by k-linear maps $m_s^t : M_s \to M_t$ for all $s \leq t \in T$, such that:

$$\begin{cases} m_s^s = id_{M_s} & \forall s \in T \\ m_s^u = m_t^u \circ m_s^t & \forall s \leq t \leq u \in T \end{cases}$$

Def: A morphism between persistence modules $\varphi : M \to N$ is defined pointwise: $\forall t \in T$, $\varphi_t : M_t \to N_t$ such that $\forall s \leq t \in T$, the following square commutes:

$$\begin{tikzcd} M_s \arrow[r, "m_s^t"] \arrow[d, "\varphi_s"'] & M_t \arrow[d, "\varphi_t"] \\ N_s \arrow[r, "n_s^t"] & N_t \end{tikzcd}$$

$\varphi : M \to N$ isomorphism if $\varphi_t : M_t \xrightarrow{\simeq} N_t$ $\forall t \in T$.

↳ Goal: encode / summarize the algebraic structure of a persistence module → use decomposition.

② Decompositions:

Idea: decompose M as a direct sum of "simpler" modules.

Def: Given M, N persistence modules over $T \subseteq \mathbb{R}$, their <u>direct sum</u> is defined pointwise:

$$\forall t \in T, \quad (M \oplus N)_t := M_t \oplus N_t$$

$$\forall s \leq t \in T, \quad (m \oplus n)_s^t := m_s^t \oplus n_s^t \quad \left(\rightsquigarrow \text{matrix form: } \begin{bmatrix} m_s^t & 0 \\ 0 & n_s^t \end{bmatrix} \right)$$

Def: M is <u>decomposable</u> if $M \simeq M_1 \oplus M_2$ for some $M_1, M_2 \neq 0$.
Otherwise, M is <u>indecomposable</u>.

Examples:

$k^2 \xrightarrow{id} k^2$ decomposes as $k \xrightarrow{id} k \oplus k \xrightarrow{id} k$

$k \xrightarrow{id} k$ is indecomposable

$k \xrightarrow{0} k$ decomposes as $k \to 0 \oplus 0 \to k$

⚠ In contrast to vector spaces, persistence modules are not always semisimple, ie. their <u>submodules may not be summands</u> (there is no basis completion theorem).

↳ <u>example:</u> $0 \to k$ is a submodule of $k \xrightarrow{id} k$ but it is not a summand

$(\nexists N \text{ s.t. } k \xrightarrow{id} k \simeq N \oplus 0 \to k)$.

$\Rightarrow$ decomposing persistence modules is a (much) harder problem than decomposing vector spaces.

Def: An *interval* of T is a subset $I \subseteq T$ such that $\forall s \le t \le u \in T$, $\quad s, u \in I \Rightarrow t \in I$.

Def: Given an interval $I \subseteq T$, the corresponding *interval module* k_I is defined by:

$$\begin{cases} \forall t \in T, \ (k_I)_t = \begin{cases} k & \text{if } t \in I \\ 0 & \text{otherwise} \end{cases} \\ \forall s \le t \in T, \ (k_I)_s^t = \begin{cases} \mathrm{id}_k & \text{if } s, t \in I \\ 0 & \text{otherwise} \end{cases} \end{cases}$$

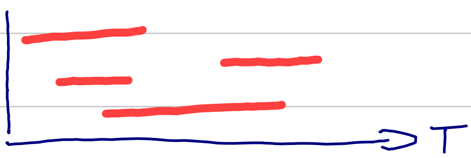
↳ interval modules are the basic building blocs for decomposing persistence modules... under some conditions

Thm: (Decomposition) [Gabriel / Auslander / Webb / Crawley-Boevey]
A persistence module M over $T \subseteq \mathbb{R}$ decomposes as a direct sum of interval modules: $M \simeq \bigoplus_{j \in J} k_{I_j}$, in either of the following (non-exclusive) cases:

- T is finite, or
- M is pointwise finite-dimensional (pfd), ie. $\dim M_t < \infty \ \forall t \in T$.

Moreover, when it exists, the decomposition is unique up to isomorphism and reordering of the terms in the direct sum.

↳ the intervals $\{I_j \mid j \in J\}$ involved in the decomposition (when it exists) form the *barcode* of M. It is a *multiset* of intervals as one may have $I_j = I_{j'}$ for some $j \ne j' \in J$ (example: $k^2 \xrightarrow{\mathrm{id}} k^2 \simeq k \xrightarrow{\mathrm{id}} k \oplus k \xrightarrow{\mathrm{id}} k$).



Def.: The *persistence diagram* of M, denoted by $\mathrm{Dgm}\, M$, is the multiset of points $\{(b_j, d_j) \mid j \in J\}$ where $b_j \leq d_j$ denote the endpoints of interval I_j in the decomposition of M.

Proof of decomposition (out of scope)

Assume (for simplicity) that T is finite (e.g. $\{0, 1, \dots, n\}$) and M is pfd.

$$M_0 \xrightarrow{m_0^1} M_1 \xrightarrow{m_1^2} \cdots \longrightarrow M_{n-1} \xrightarrow{m_{n-1}^n} M_n$$

Extend M over $\mathbb{N}$ by adding copies of M_n:

$$M_0 \xrightarrow{m_0^1} M_1 \xrightarrow{m_1^2} \cdots \longrightarrow M_{n-1} \xrightarrow{m_{n-1}^n} M_n \xrightarrow{id} M_n \xrightarrow{id} \cdots$$

M then has the structure of a module over $k[t]$:

$$\begin{cases} M \simeq \bigoplus_{t=0}^{\infty} M_t & \text{equipped with multiplication by } t: \\ t \cdot (x_0, x_1, \dots) = (0, m_0^1(x_0), m_1^2(x_1), \dots) \end{cases}$$

M is then a finitely generated module over $k[t]$, which is a PID, hence we have the well-known decomposition:

$$M \simeq \underbrace{\bigoplus_{i \in J_{free}} t^{b_i} \cdot k[t]}_{\text{free (infinite cyclic) part}} \oplus \underbrace{\bigoplus_{j \in J_{tor.}} \cancel{t^{b_j} \cdot k[t]}\, t^{(d_j - b_j)} \cdot k[t]}_{\text{torsion (finite cyclic) part}}.$$

infinite bars $[b_j, \infty)$

finite bars $[b_j, d_j)$

---

③ Computing persistence barcodes / diagrams:

Input: a simplicial filtration F such that:

any finite simplicial filtration can be adapted to satisfy these conditions

$T = \{0, 1, \dots, n\}$
$F_0 = \emptyset$, $F_n = K$ (finite simplicial complex)
$\forall t$, F_t is a subcomplex of F_{t+1}
$\forall t$, $F_{t+1} \setminus F_t = \{\sigma_t\}$ (only one simplex inserted at each iteration)

↳ simplices of K are in bijection with $\{1, \dots, n\}$ $(\sigma_t \Leftrightarrow t)$
↳ total order on the simplices of K, compatible with incidence relations: $\sigma_s \subseteq \sigma_t \Rightarrow s \leq t$.

Algo: same as for computing homology, with order on simplices fixed to be the one given by the filtration:

a) Compute matrix of boundary operator:

$$A = \begin{array}{c} \\ \sigma_1 \\ \vdots \\ \sigma_n \end{array} \begin{array}{c} \sigma_1 \ \dots \ \sigma_n \\ \left[\begin{array}{ccc} & & \\ & & \\ & & \end{array} \right] \end{array}$$

$A_{ij} \neq 0$ iff σ_i is a face of σ_j of codimension 1.

b) $$\mathrm{low}(j) := \begin{cases} 0 \text{ if } A_{ij} = 0 \ \forall i \\ \max\{i \mid A_{ij} \neq 0\} \text{ otherwise} \end{cases}$$

c) Reduce A to column-echelon form from left to right:

for $j = 1$ to n do:
 while $\exists \ell < j$ s.t. $\mathrm{low}(\ell) = \mathrm{low}(j) \neq 0$ do:
  change column c_j into $c_j - \dfrac{A_{\mathrm{low}(j)j}}{A_{\mathrm{low}(j)\ell}} \cdot c_\ell$

d) Interpret the reduced matrix:

$$
\begin{array}{c} \\ \sigma_1 \\ \vdots \\ \sigma_i \\ \vdots \\ \sigma_n \end{array}
\begin{array}{c}
\begin{array}{ccccccc} \hat\sigma_1 & \cdots & \hat\sigma_j & \cdots & \hat\sigma_{j'} & \cdots & \hat\sigma_n \end{array} \\
\left[\begin{array}{ccccccc}
 & & 0 & & * & & \\
 & & | & & \vdots & & \\
\cdots & \cdots & | & \cdots & \boxed{*} & & \\
 & & | & & 0 & & \\
 & & | & & | & & \\
 & & 0 & & 0 & &
\end{array}\right]
\end{array}
$$

* every column j corresponds to a chain

$$\hat\sigma_j = \sigma_j - \sum_{\ell < j} \lambda_\ell \sigma_\ell$$

appearing when σ_j is inserted

* if $\hat\sigma_j = 0$ then σ_j is positive (its insertion creates a cycle)

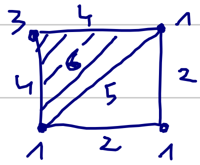
* if $\hat\sigma_j \neq 0$ then σ_j is negative (its insertion trivializes a cycle)

(the cycle created when σ_i was inserted where $i = \mathrm{low}(j)$)

Thus: the summands of $H_*(F)$ are in bijection with the zeroed out columns in the reduced matrix:

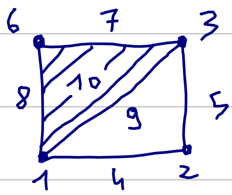
- each pair $(\sigma_i, \hat\sigma_j)$ with $i = \mathrm{low}(j)$ generates a finite interval summand $k_{[i,j)}$ in $H_{\dim \sigma_j}(F)$,
- each unpaired $\hat\sigma_j$ generates an infinite interval summand $k_{[j,\infty)}$ in $H_{\dim \sigma_j}(F)$.

Example: $k = \mathbb{Z}/2\mathbb{Z}$, $K =$



$\hookrightarrow$ map simplex set bijectively and monotonously to $\{1, \ldots, n\}$:

$F =$



$\hookrightarrow$ boundary matrix:

$$
\begin{array}{c}
\\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10
\end{array}
\begin{array}{c}
\begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{array} \\
\left[\begin{array}{cccccccccc}
 & & & 1 & & & & 1 & 1 & \\
 & & & 1 & 1 & & & & & \\
 & & & & 1 & & 1 & & 1 & \\
 & & & & & & & & & \\
 & & & & & & & & & \\
 & & & & & & 1 & 1 & & \\
 & & & & & & & & & 1 \\
 & & & & & & & & & 1 \\
 & & & & & & & & & 1 \\
 & & & & & & & & &
\end{array}\right]
\end{array}
\xrightarrow{\text{(reduction)}}
\left[\begin{array}{cccccccccc}
 & & & 1 & & & & & & \\
 & & & \underset{(2,4)}{\boxed{1}} & 1 & & & & & \\
 & & & & \underset{(3,5)}{\boxed{1}} & & 1 & & & \\
 & & & & & & & & & \\
 & & & & & & & & & \\
 & & & & & & \underset{(6,7)}{\boxed{1}} & & & \\
 & & & & & & & & & 1 \\
 & & & & & & & & & 1 \\
 & & & & & & & & & \underset{(9,10)}{\boxed{1}} \\
 & & & & & & & & &
\end{array}\right]
$$

Hence the barcode:

$$\begin{cases} H_0 : [1,\infty)\,;\,[2,4)\,;\,[3,5)\,;\,[6,7) \\ H_1 : [8,\infty)\,;\,[9,10) \end{cases}$$

↳ after mapping to the original indices:

$$\begin{cases} H_0 : [1,\infty)\,;\,[1,2)\,;\,[1,2)\,;\,[3,4) \\ H_1 : [4,\infty)\,;\,[5,6) \end{cases}$$

$\boxed{\text{Note:}}$ For computation by hand, the same approach as with homology can be taken, with a little extra book keeping:

a) fix a total order on simplices that is compatible with filtration and incidence orders.

b) iterate over the simplices in this order, and for each simplex σ_j:

- either it creates a cycle $\Rightarrow$ creates a summand $[j, ?)$ in $H_{\dim \sigma_j}(F)$
- or it kills a cycle (find σ_i that created it) $\Rightarrow$ $[i, ?)$ is completed as $[i, j)$ in $H_{\dim \sigma_i}(F)$

c) Upon termination, replace the remaining "?" in the barcode by "∞".