TIPE : Etude de couvertures de réseaux de métros, application de l'homologie persistante

Elowan H

Sommaire

Résumé

Nous nous proposons ici d'étudier les différentes disparités dans les réseaux métropolitains de plusieurs grandes villes, nous allons détecter les zones spatiales les plus en déficit de transports en commun. Pour cela, nous adopterons une approche analytique utilisant la topologie : *l'homologie persistante.*

I Définitions

Intro à changer je pense

L'exemple suivant est tiré de [1].

Prenons comme exemple le style artistique du pointillisme. Lorsque l'on regarde une oeuvre d'art, comme le tableau de Seurat en Figure 1, nous voyons bien plus qu'un grand nombre de points, nous voyons des formes et des objets. D'une discrétisation, nous en faisons un *continuum* de formes.

L'homologie persistante est une *méthode de calcul* qui, à partir d'une discrétisation, cherche à fournir un descriptif des caractéristiques que l'on pourrait distinguer de cet ensemble.

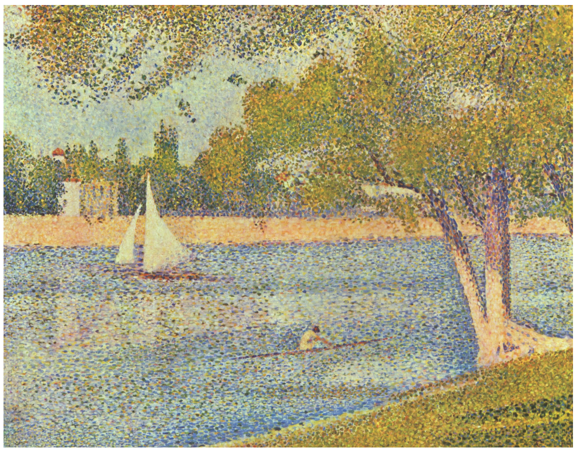

Figure 1: La scène à la Grande Jatte - Printemps (Georges Seurat, 1888)

Nous chercherons ici seulement à caractériser les "trous" dans un espace, afin de détecter les trous de couverture dans un réseau métropolitain, ici modélisé comme un nuage de points de $\mathbb{R}^2$, dont chaque élément est une station de métro.

Afin d'utiliser l'homologie persistante, nous devons définir certaines notions géométriques, nous noterons dans la suite X l'ensemble des points de $\mathbb{R}^2$ que l'on considère.

Définition :
Un simplexe σ de dimension k (ou k-*simplexe*) correspond à l'enveloppe convexe de $k+1$ points de X non inclus dans un sous-espace affine de dimension $k-1$.

On définit un simplexe de dimension 0 comme un point de X.

Par exemple, un simplexe de dimension 1 est un segment et un simplexe de dimension 3 est un trièdre.

Remarque : On dit que σ_i est une face de σ_j si et seulement si $\sigma_j \subset \sigma_i$ et la dimension de σ_i est strictement supérieure à celle de σ_j.

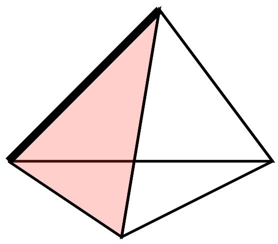

Figure 2: Ce trièdre est un simplexe σ de dimension 3, où le triangle rouge représente un simplexe τ de dimension 2 mais aussi une face de σ dans le sens de la remarque précédente. Notons que l'arête en gras est un simplexe de dimension 1 et est une face de σ et de τ.

Définition :
Un *complexe simplicial* est un ensemble de simplexes.

On note $P_n(X)$ l'ensemble des parties à $n+1$ éléments de X formant un complexe simplicial contenant tous les n-simplexes.

Définition :
Une *filtration* est une application qui à un entier i associe un complexe simplicial K_i de sorte que :

$$\forall j \in [\![0, i]\!] \ , \ \ K_j \subset K_i$$

Observons sur la Figure 3 les notions précédemment définies. Chaque K_i est un complexe simplicial, la suite $(K_i)_{i=0}^{3}$ est une filtration et tous les points, segments et faces (ici en rouge) sont des simplexes de dimension 0, 1 et 2 respectivement.

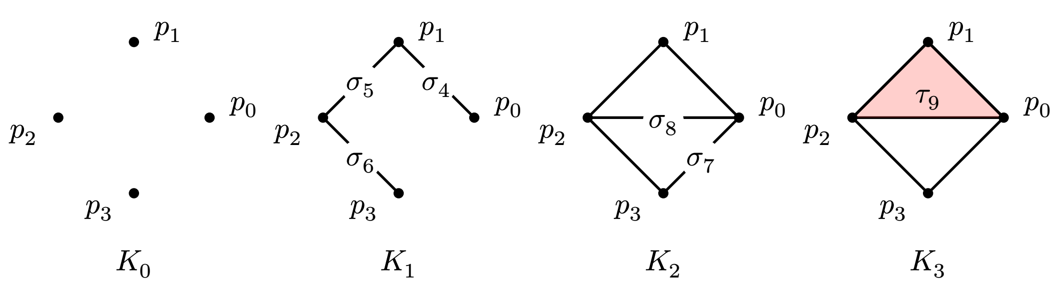

Figure 3: Représentation d'une filtration où $K_0 \subset K_1 \subset K_2 \subset K_3$ et où chaque simplexe a été nommé : p_i pour les simplexes de dimension 0; σ_j pour la dimension 1 et τ_k pour la dimension 2.

Nous observons qu'une filtration permet d'ajouter une notion de "temporalité" dans un ensemble de points. Nous sommes capable de noter quels événements surviennent entre deux complexes à la suite, par exemple l'apparition d'un cycle entre K_1 et K_2 (le cycle $(\sigma_4, \sigma_5, \sigma_6, \sigma_7)$) ou l'apparition d'un simplexe de dimension 2 entre K_2 et K_3 (la face rouge).

On introduit de plus un ordre total $\preccurlyeq$ sur l'ensemble des simplexes d'une filtration.

Définition :
Soient une filtration $K_0 \subset K_1 \subset ... \subset K_p$ et l'ensemble S de tous les simplexes apparaissant dans la filtration. On indice S de sorte que pour tout σ_i et σ_j de S:

$$\begin{cases} \text{Si } \sigma_i \in K_{k_i} \text{ et } \sigma_j \in K_{k_j} \text{ avec } k_i < k_j \\ \text{Sinon si } \sigma_i \text{ est une face de } \sigma_j \end{cases} \Rightarrow i < j$$

Si aucun des deux cas n'est réalisé alors le choix de l'ordre entre les deux simplexes est arbitraire.

On définit donc l'ordre total $\preccurlyeq$ telle que $\sigma_i \preccurlyeq \sigma_j \Leftrightarrow i \leq j$

Observons sur la Figure 3 l'indexation des simplexes suivant l'ordre précédemment défini : les simplexes σ_8 et σ_7 sont plus grand au sens de $\preccurlyeq$ que tous les autres σ_i et p_i parce qu'ils apparaissent plus tard dans la filtration. De plus, si σ_8 était apparu dans K_3, l'ordre aurait toujours été respecté puisque σ_8 est une face de τ_9.

Il y a un problème : nous voulons analyser un ensemble de points discrets, et non une filtration déjà existante, il nous faut alors créer une filtration depuis un ensemble de points. Nous faisons cela via une construction incrémentale de complexes simpliciaux avec les complexes de Vietoris-Rips pondérés. Ainsi d'après [2] :

Définition :
Soient un ensemble $X = (x_i)_{i=0}^n$ de points associés à des poids $(w_i)_{i=0}^n$ et une distance d, on définit le complexe simplicial pondéré de Vietoris-Rips au rang r, noté $V_r(X, d)$, comme l'ensemble des simplexes $\left(x_{i_0}, ..., x_{i_k}\right)$ tels que :

$$\begin{cases} \forall j \in [\![0, k]\!], w_{i_j} < r \\ \forall (j, l) \in [\![0, k]\!]^2, d\left(x_{i_j}, x_{i_k}\right) + w_{i_j} + w_{i_k} < 2r \end{cases}$$

Ainsi plus on augmente r, plus le complexe possède des simplexes, on en donne une représentation Figure 4. Pour chaque r qui augmente le nombre de simplexes du complexe simplicial, nous ajoutons $V_r(X, d)$ à la filtration que l'on est en train de créer.

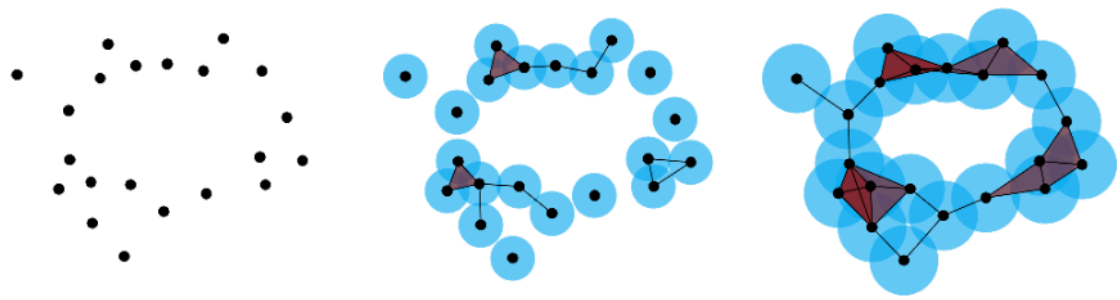

Figure 4: Construction d'un complexe simplicial avec un r grandissant de gauche à droite

Ici, r est le rayon des boules bleues, et un simplexe est considéré dès lors que les boules associées à ces sommets se rencontrent.

Pour définir formellement des “trous”, nous devons définir les opérateurs de bords. Ainsi selon [3] :

Définition :
On définit un *complexe de chaînes* comme la donnée d'une suite

$$\ldots \stackrel{\delta_{k+2}}{\to} C_{k+1} \stackrel{\delta_{k+1}}{\to} C_k \stackrel{\delta_k}{\to} C_{k-1} \stackrel{\delta_{k-1}}{\to} \ldots$$

Où chaque C_k est un groupe abélien libre qui a pour base les k-simplexes de X et δ_k est une morphisme de groupes tel que $\delta_k \circ \delta_{k+1} = 0$

On appelle δ_k un *opérateur de bords.*

Définition :
On définit alors les *classes d'homologie de dimension* k comme le groupe de $\text{Ker}(\delta_k)$ quotienté par $\text{Im}(\delta_{k+1})$:

$$H_k = \text{Ker}(\delta_k) / \text{Im}(\delta_{k+1})$$

Celle-ci représente les “trous” en dimension k.

On peut voir que H_0 représente les composantes connexes de X, H_1 représente un trou qui est entouré par un chemin fermé de points connectés (comme le cycle $(\sigma_4, \sigma_5, \sigma_6, \sigma_7)$ dans K_2 dans Figure 3 par exemple) et H_2 représenterait par exemple un trièdre d'intérieur vide.

Exemple à faire

Pour notre usage, nous voulons calculer H_0, les temps moyens pour se rendre à une station de métros, et H_1 qui représente les zones critiques de couverture du réseau.

II Les données

II.1 Sources

On choisit de se baser uniquement sur des vraies villes, que l'on nommera Ville A et Ville B par la suite, pour tester notre approche. De plus, toutes les informations relatives aux stations de metros ainsi que les temps de passages sont trouvables sur le site du gouvernement.

Ces informations servent à définir nos points et notre pondération, en revanche elles ne permettent pas d'obtenir les distances entre les stations, pour cela nous utiliserons alors geoapify qui nous permet d'estimer des temps de trajet en voiture et à pied.

II.2 Points et distances

Définissons dès lors nos objets :

Définition :
Un point x_i, représentant une station de métro, est défini par la données de sa position géographique (latitude/longitude) ainsi que son poids w_i. Le poids w_i est égal à la moyenne du temps d'attente entre deux métros en station x_i sur une semaine entière.

Les temps de passage des metros en station étant plus ou moins constant sur la semaine, il est cohérent d'utiliser une moyenne.

On définit la distance similairement à [2]:

Définition :
On définit la distance entre deux stations de métros x et y comme :

$$d(x,y) = \frac{1}{2}(\min(t_{\text{marche}}(x,y), t_{\text{voiture}}(x,y)) + \min(t_{\text{marche}}(y,x), t_{\text{voiture}}(y,x)))$$

Ainsi en revenant aux boules des complexes simplicaux de Vietoris-Rips, elle modélise le coût temporel d'un trajet "porte à porte" en utilisant le métro.

III Méthode

Pour trouver les zones critiques, nous utiliserons la méthode de l'homologie persistante décrite dans [2] (dans le cas de notre réseau de métros). Celle ci se décompose en 3 étapes :

- Transformation de l'ensemble de $\mathbb{R}^2$ des stations de métros x_i de poids w_i en une filtration;
- Création et réduction de la matrice de bordure (définie dans la suite);
- Récupération des simplexes "tueurs" de classes d'homologies

On suppose que l'étape une, malgré la difficulté technique qu'elle pose à implémenter, est déjà réalisée suivant la section I.

Ainsi à partir de cette filtration, nous pouvons obtenir les classes d'homologie grâce au théorème qui suit :

Théorème des facteurs invariants :
D'après [1] et [4], il existe un unique ensemble $\{d_1, ..., d_p\}$ d'éléments de H_k définis à des inversibles près, tel que :

$$H_k(X) \simeq \bigoplus_{i=1}^{p} P_k(X)/d_i P_k(X)$$

en notant $P_k(X)$ l'ensemble des parties à $k+1$ éléments de X, formant donc un complexe simplicial constitué uniquement de k-simplexes. Cet ensemble est appelé *code barre* de H_k

A reprendre

Informatiquement, selon [5], on calcule ce code barre en créant une matrice de bordure B après avoir défini un ordre total sur les simplexes respectant les propriétés énoncées section I.

Définition :
On définit la matrice de bordure, associée à un ordre total $\sigma_0 \preccurlyeq ... \preccurlyeq \sigma_{n-1}$ sur tous les simplexes $(\sigma_i)_{i=0}^{n-1}$ de la filtration, suivant :

$$\forall (i,j) \in [\![0, n-1]\!]^2, B[i][j] = \begin{cases} 1 \text{ si } \sigma_i \text{ est une face de } \sigma_j \\ 0 \text{ sinon} \end{cases}$$

Note : D'après [4], B peut être vu comme la matrice de δ_1 dans la base associée à C_1.

Un exemple d'une telle matrice est donnée en Table 1.

Après avoir calculé B, nous voulons la *réduire* à un code barre, dans le sens où l'on peut interpréter correctement les valeurs de cette matrice avec la filtration (grâce au théorème précédent). Le terme de *réduction* fait ici référence à la réduction de B en forme normale de Smith. Dans notre cas, ce résultat s'interprète comme l'attribution à chaque simplexe de la naissance d'*au plus* une classe d'homologie. Nous pouvons observer ce résultat en Table 2.

Cet algorithme de réduction est nommé *standard algorithm* et est décrit dans [5] par, en posant $\text{low}_B(j) = \max(\{i \in [\![0, n-1]\!], B[i][j] \neq 0\}) \in \mathbb{N} \cup \{-1\}$:

```
StandardAlgorithm(B)
    for j in [|0, n-1|]:
        while (il existe i < j avec low_B(i) = low_B(j)):
            ajouter colonne i de B à colonne j
```

Comparons alors nos deux matrices, sur l'exemple de la filtration de Figure 3 (Les cases vident remplacent les zeros pour plus de lisibilité et les colonnes/lignes vides ont été omises), avec ici notre ordre total sur les simplexes :

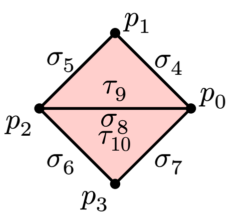

Figure 5: Rappel du nommage des simplexes (p_i pour la dimension 0, σ_j pour la dimension 1, τ_k pour la dimension 2)

En regardant la matrice $\overline{B}$, nous remarquons que l'opération de réduction à permis d'avoir la ligne low sans répétion de nombre positifs, autrement dit, on accorde la naissance d'un simplexe à un unique autre simplexe. Ainsi le simplexe 1 donne naissance au simplexe 4, 8 donne naissance à 9 et 7 donne naissance à 10.

Enfants (colonnes) / **Parents** (lignes)

	4	5	6	7	8	9	10
0	1			1	1		
1	1	1					
2		1	1		1		
3			1	1			
4						1	
5						1	
6							1
7							1
8						1	1
low	**1**	**2**	**3**	**3**	**2**	**8**	**8**

Table 1: Matrice B

Enfants (colonnes) / **Parents** (lignes)

	4	5	6	7	8	9	10
0	1						
1	1	1					
2		1	1				
3			1				
4						1	1
5						1	1
6							1
7							1
8						1	
low	**1**	**2**	**3**	−1	−1	**8**	**7**

Table 2: Matrice $\overline{B}$, la matrice B après reduction

Par exemple, si $\text{low}_{\overline{B}}(j) = i \neq -1$ alors on a une paire de simplexe (σ_i, σ_j) telle que l'apparition de σ_i fait apparaitre une nouvelle classe d'homologie. Et au contraire, σ_j va la *tuer* en apparaissant. Prenons comme exemple la filtration Figure 3 : dans K_0, p_1 cause l'apparition d'une classe dans H_0 cependant l'apparition du simplexe σ_7 dans K_2 tue la classe de p_1 dans H_0 mais créer une nouvelle classe dans H_1 (car elle crée un cycle).

En revanche si $\mathrm{low}_{\overline{B}}(j) = -1$ alors l'apparition de σ_j crée une classe d'homologie : s'il existe k tel que $\mathrm{low}_{\overline{B}}(k) = j$ on est dans le cas précédent, sinon la classe d'homologie n'est jamais tuée.

C'est depuis cette matrice que nous sommes capables de déterminer H_0 et H_1, et donc de générer des représentations graphiques comme montré en Figure 6

IV Résultats et conclusion

Ville	Dimension	Médiane	Variance
Ville A	0D Homologie	184.00s	23.57s
	1D Homologie	223.50s	10.5s
Ville B	0D Homologie	211.00s	22.27s
	1D Homologie	318.00s	58.62s

Table 3: Tableau récapitulant les médianes ainsi que la variance des temps de mort des classes holomogiques pour chaque ville.

On comprend que globalement il faut 200s (soit 3m20s) pour quelqu'un de se rendre d'une station à une autre (le minimum en temps entre la voiture et la marche) ce qui est effectivement cohérent avec la réalité. Les temps des classes pour la dimension 1 montrent le temps moyen de trajet entre les deux stations les plus éloignées d'un même cycle. Donc par exemple pour Ville B, il faudra en moyenne 318s (5min20s) pour rejoindre une station depuis les zones les moins bien deservies.

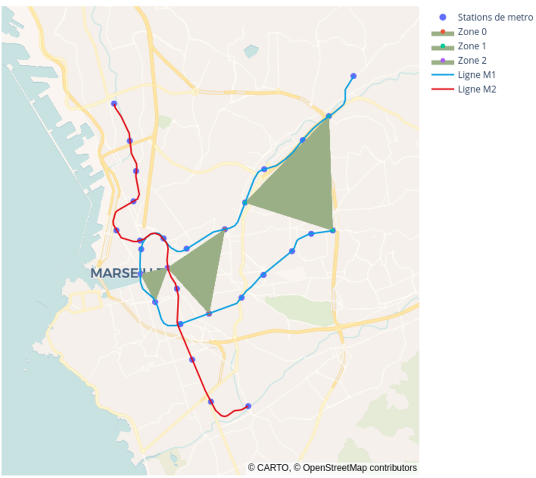

Figure 6: Carte de Ville A

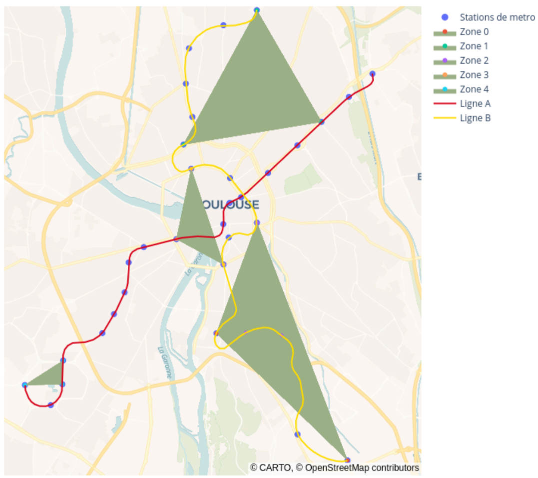

Figure 7: Carte de Ville B

Visuels à changer pour supprimer le nom des villes et rendre plus joli les triangles

Les triangles ici représentés montrent les zones où il est le plus difficile de rejoindre une station de métro. Pour les plus gros triangles, il peut être cohérent de croire qu'il est difficile de se rendre à ces stations de métros. En revanche, l'interprétation est plus dure pour les plus petits triangles.

Nous devons revenir à la définition de notre distance : celle ci prend en compte le temps minimal entre un trajet en voiture et le même trajet à pied. Les plus petites zones, comme à gauche sur la ligne bleue dans la Ville A ou en bout de ligne rouge dans la Ville B, correspondent en fait à des espaces uniquement piétons dont le temps de trajet est plus court à pied qu'en voiture. Ainsi, les plus petites zones indiquent donc la même information (difficulté d'accès à ces stations) que les grandes mais à une échelle différente.

L'homologie persistante est donc une méthode nous permettant de mettre en lumière des zones mal desservies en prenant en compte des realités plus complexes que seul le temps de trajets. Par exemple, nous prenons en compte les temps d'attente en station mais nous aurions pu aussi prendre en compte la densité de population autour de ces stations. Cette caractéristique peut être une possibilité d'ouverture de ce sujet car celle ci joue intuitivement un rôle dans le temps d'attente en station et donc dans la difficulté de prendre un métro.

Bibliographie

[1] Henri Paul de Saint-Gervais, Une invitation à l'homologie persistante, (n.d.).

[2] ABIGAIL HICKOK, BENJAMIN JARMAN, MICHAEL JOHNSON, JIAJIE LUO, and MASON A. PORTER, *Persitent Homology for Resource Coverage: A Case Study of Access to Polling Sites.*

[3] Jean-Yves Welschinger, Introduction aux théories homologiques, (n.d.).

[4] Afra Zomorodian and Gunnar Carlsson, *Computing Persistent Homology.*

[5] Nina Otter, Mason A Porter, Ulrike Tillmann, Peter Grindrod, and Heather A Harrington, *A Roadmap for the Computation of Persistent Homology.*