

Rapport de projet Graphes et OpenData (GOD)

Analyse interactive des bibliothèques en France via les graphes et l'Open Data

Projet réalisé du 16 janvier 2025 au 14 avril 2025

Remerciements

Nous remercions Monsieur Delbot pour son accompagnement tout au long du projet, ainsi que les membres de notre groupe pour leur investissement.

Nos Membres



FIGURE 1 – Lydia AMROUCHE



FIGURE 2 – El Hadji Fodé KANE

Code source disponible sur : https://github.com/efkane/analyse_Bibliotheques_France.git

Table des matières

1	Introduction	4
2	Environnement de travail	4
3	Description du projet et objectifs	5
4	Analyse des données	5
4.1	Aperçu du dataset	5
4.2	Statistiques par région	5
5	Traitement des données : Nettoyage et enrichissement	6
6	Analyse par les graphes	6
6.1	Construction du graphe	6
6.2	Analyse de communautés	7
6.3	Plus court chemin	8
6.4	Minimum Dominating Set Connexe (MDS)	9
7	Travail réalisé	10
8	Difficultés rencontrées	10
9	Bilan	10
9.1	Conclusion	10
9.2	Perspectives	11
10	Bibliographie	12
11	Webographie	12
12	Annexes	13
A	Cahier des charges	13
B	Exemple d'exécution	13
C	Manuel utilisateur	14

1 Introduction

Ce projet a été réalisé dans le cadre du module "Graphes et Open Data". Il a pour objectif de croiser les apports de la visualisation de données ouvertes et de la théorie des graphes, en s'appuyant sur un jeu de données national concernant les bibliothèques universitaires françaises.

Les bibliothèques sont des infrastructures clés du service public culturel. Grâce aux données ouvertes, il est possible de mieux comprendre leur répartition, leurs caractéristiques, et leurs interactions géographiques et structurelles.

Le projet s'est structuré autour des axes suivants :

- Extraction, nettoyage et enrichissement des données.
- Développement d'une application interactive de visualisation.
- Construction et analyse de graphes représentant les relations entre bibliothèques.
- Exploration d'une problématique avancée : le Minimum Dominating Set Connexe.

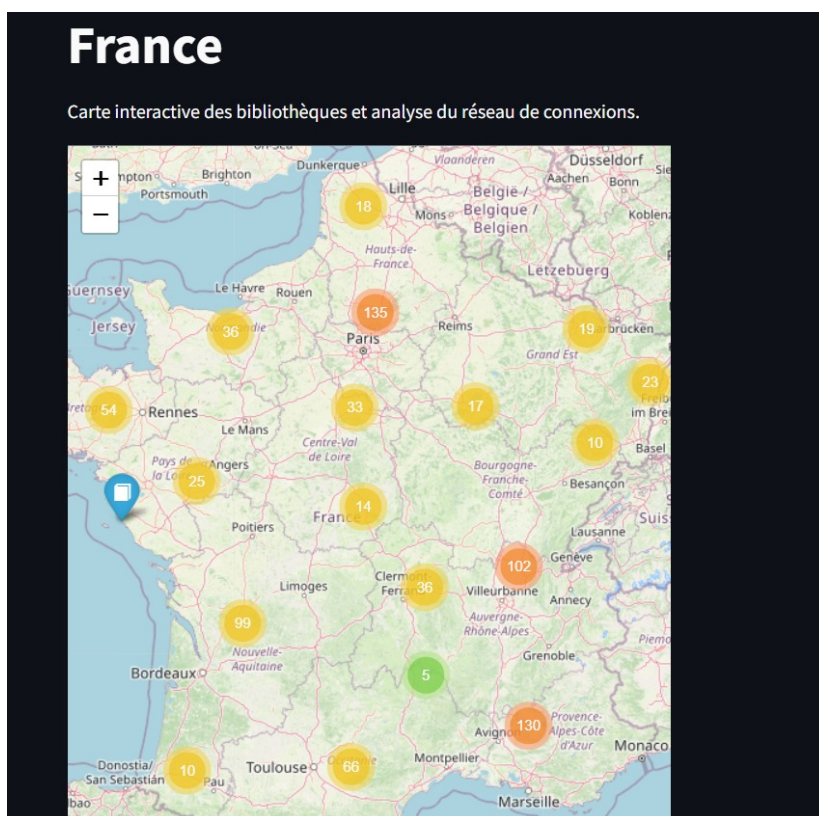


FIGURE 3 – Carte interactive des bibliothèques en France

2 Environnement de travail

- **Python 3.11** : langage principal.
- **Visual Studio Code** et **PyCharm** : IDEs pour le développement.
- **Streamlit** : création de l'interface web.
- **NetworkX**, **Folium**, **PuLP** : librairies principales.
- **Virtualenv** : gestion des environnements Python.
- **GitHub** : gestion de version et collaboration en ligne pour le dépôt du projet.

3 Description du projet et objectifs

Le projet a pour but de créer une application web qui :

- Exploite la théorie des graphes pour explorer des notions avancées.
- Visualise les bibliothèques françaises sur une carte.
- Visualise la répartition des ouvrages dans ces bibliothèques.
- Trouve la plus petite distance à parcourir pour trouver un maximum d'ouvrages qui ne doivent pas se trouver dans une seule bibliothèque.
- Propose des statistiques interactives (régions, surfaces, types).
- Analyse les relations géographiques et structurelles entre bibliothèques.

4 Analyse des données

4.1 Aperçu du dataset

A	B	C	D	E	F	G	H	I	J	K	L	N
le l'établiss	ville	region	latitude	longitude	surface	Ouvrages						
Coordinati	Caussade	Occitanie	44,15395	1,527161		JavaScript, Cloud, Réseaux, NoSQL, SQL						
Bibliothèq	Clessé	Nouvelle-A	46,71656	-0,40502	10	Java, Cloud, JavaScript						
Bibliothèq	Arnac-la-P	Nouvelle-A	46,26557	1,374143	72	Data Science, Sécurité, HTML, Cloud						
Médiathèc	Saix	Occitanie	43,58282	2,18104	120	Linux, CSS, Cloud, NoSQL, Sécurité						
Bibliothèq	Villars-le-F	Bourgogne	47,8948	5,934542	60	R, Réseaux, DataViz, JavaScript, IA						
Bibliothèq	Montpont	Bourgogne	46,5554	5,164046	150	R, C++, IA, DataViz						
MEDIATHE	Saint-Malc	Bretagne	48,64737	-2,00668	2678	Docker, Data Science, DataViz						
Médiathèc	Colombier	Occitanie	43,31336	3,141969	285	Java, JavaScript, Data Science, SQL, Docker						
Médiathèc	Martigné-f	Bretagne	47,83166	-1,32065	250	Python, IA, C++, CSS, Git						
Médiathèc	Magalas	Occitanie	43,4718	3,224881	200	DevOps, Linux, C++						
Médiathèc	Saint-Malc	Pays de la	47,35232	-2,17958	512	HTML, NoSQL, Java, Python						
Bibliothèq	Ruffey-lès	Bourgogne	47,01917	4,911014	63	SQL, DevOps, CSS, Git						
Point lectu	Rountzenh	Grand Est	48,81728	8,009476	79	C++, Data Science, Réseaux, Java, DevOps						
Bibliothèq	Torcy-le-G	Normandie	49,79142	1,175162		Cloud, DataViz, Data Science, Python, Sécurité						

FIGURE 4 – Extrait du dataset brut

4.2 Statistiques par région

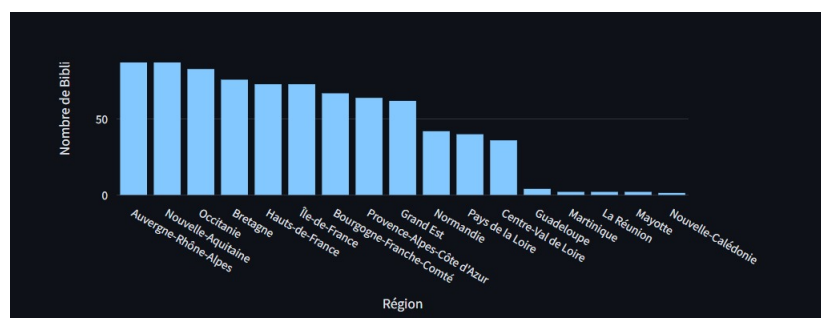


FIGURE 5 – Nombre de bibliothèques par région

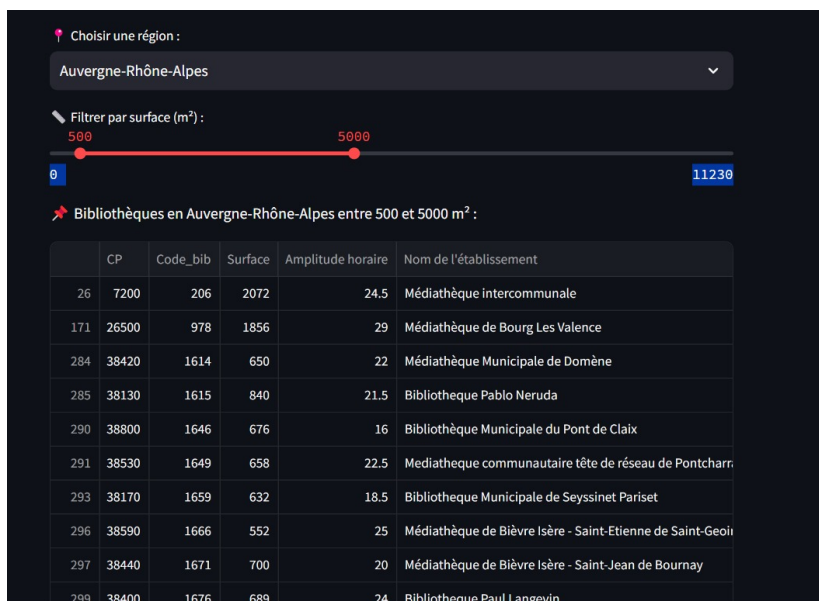


FIGURE 6 – Zoom sur les bibliothèques d’une région

5 Traitement des données : Nettoyage et enrichissement

- Suppression des doublons, les régions d’outre-mer par souci de connexité pour les graphes.
- Transformation des coordonnées GPS.
- Unification des noms de villes.
- Formatage des surfaces (m²) en valeurs numériques.

6 Analyse par les graphes

6.1 Construction du graphe

Chaque bibliothèque est un nœud. Deux nœuds sont connectés si les bibliothèques sont proches géographiquement ou partagent une même ville.



FIGURE 7 – Graphe géographique des bibliothèques



FIGURE 8 – Zoom sur une partie connectée du graphe

6.2 Analyse de communautés

Nous avons appliqué l'algorithme de Louvain pour identifier les communautés de bibliothèques étroitement reliées.

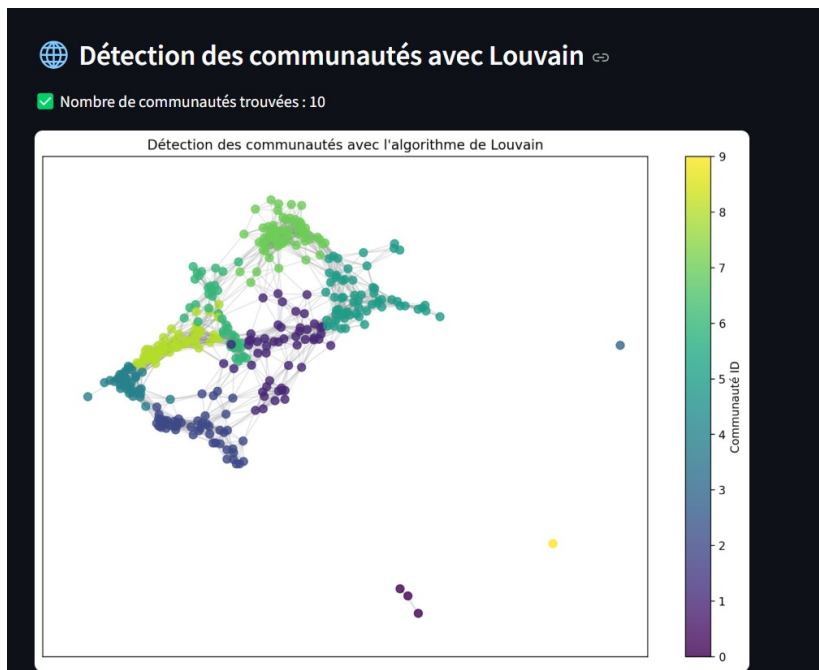


FIGURE 9 – Détection de communautés (Louvain)

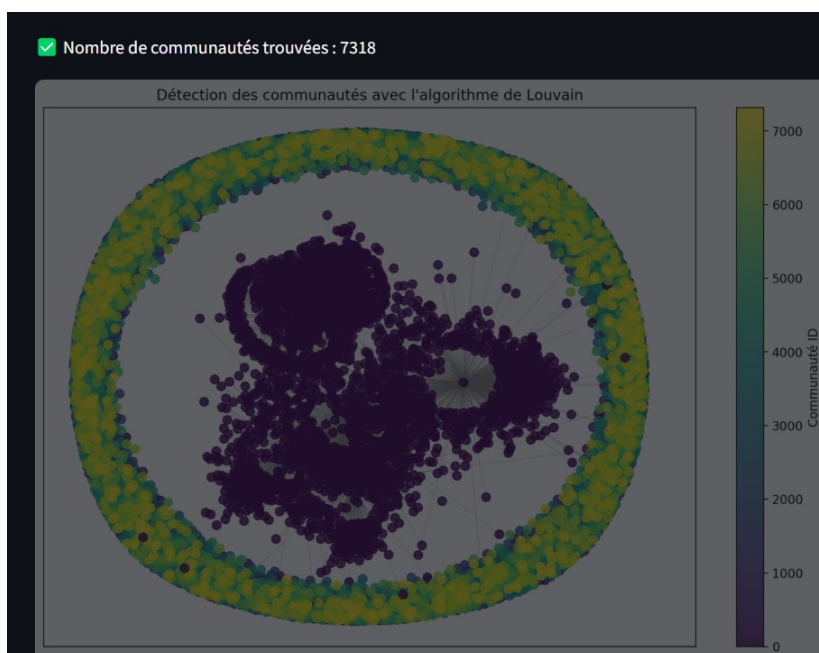


FIGURE 10 – Vue complète des communautés dans le graphe

6.3 Plus court chemin

L'utilisateur peut simuler des trajets entre bibliothèques pour obtenir le chemin optimal.

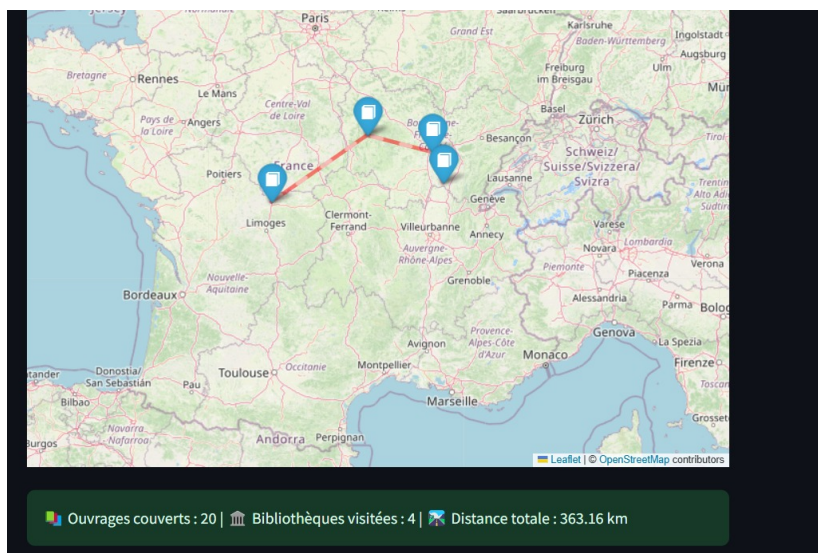


FIGURE 11 – Calcul du plus court chemin entre deux bibliothèques

6.4 Minimum Dominating Set Connexe (MDS)

Objectif : identifier un sous-ensemble minimal de bibliothèques couvrant tout le graphe, tout en étant connexe. Cela a été modélisé comme un problème d'optimisation via PuLP.

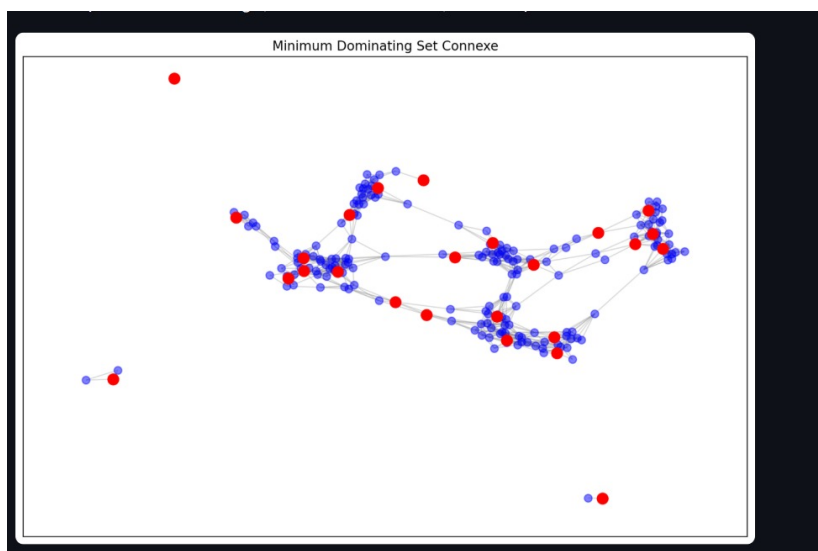


FIGURE 12 – Sous-ensemble dominant connexe (MDS)

Ce graphique montre le réseau des bibliothèques sélectionnées, reliées par leur proximité géographique. Les points rouges sont les bibliothèques stratégiques assurant la couverture du réseau, et chaque point bleu est connecté à une bibliothèque rouge.

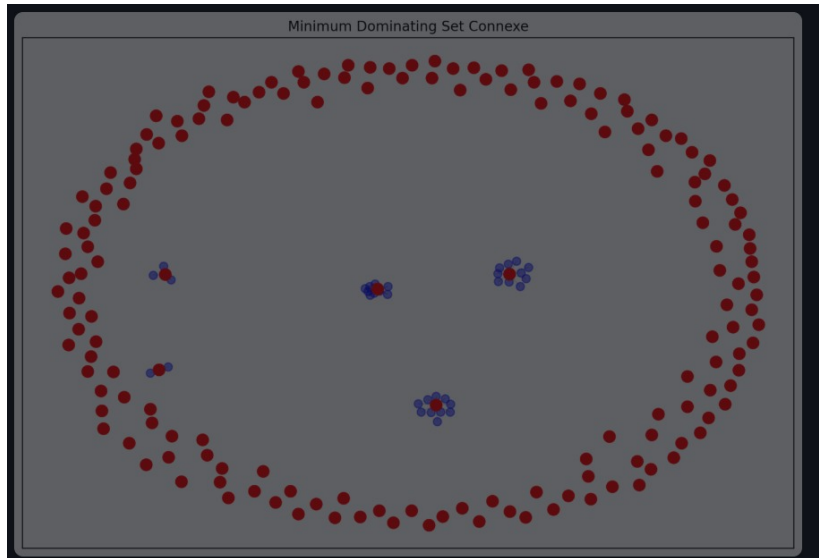


FIGURE 13 – Vue complète du MDS appliqué au graphe

7 Travail réalisé

- Création d’une application Streamlit complète.
- Intégration d’une carte interactive (Folium).
- Statistiques dynamiques par région.
- Graphe interactif entre bibliothèques.
- Algorithmes : Louvain, plus court chemin, MDS.

Répartition des tâches

- **Lydia** : Implémentation des algorithmes, l’application Streamlit, Intégration des visualisation
- **El Hadji Fodé** : Scraping, Traitement des données, Création des visualisations.

8 Difficultés rencontrées

- Données incomplètes ou non homogènes.
- Compréhension des formats géographiques.
- Prise en main des librairies NetworkX et PuLP.
- Gestion de la mémoire lors du traitement de graphes denses.
- Lenteur de la visualisation des graphes dans le site créé avec Streamlit, en raison du traitement de graphes denses et de la consommation importante de ressources.

9 Bilan

9.1 Conclusion

Le projet nous a permis de combiner plusieurs disciplines : traitement de données, géovisualisation, théorie des graphes, optimisation. Il a également donné lieu à une application complète et utilisable.

9.2 Perspectives

- Déploiement de l'application en ligne.
- Ajout de données supplémentaires (fréquentation, budget).
- Automatisation du nettoyage de données.
- Amélioration des performances sur des graphes plus larges.

10 Bibliographie

Références

- [1] M. L. S. T. Thomas, *Introduction aux graphes et leurs applications*, Dunod, 2018.
- [2] A. F. L. Jean-Pierre, *Open Data et transformation numérique*, Editions Eyrolles, 2020.

11 Webographie

Références

- [1] data.gouv.fr : *Jeu de données des bibliothèques françaises*, <https://www.data.gouv.fr/fr/datasets>
- [2] Bibliothèques.fr : *Site officiel des bibliothèques publiques de France*, <https://www.bibliotheques.fr/>.

12 Annexes

Annexe A : Cahier des charges

- Objectif : analyser les bibliothèques françaises via les données publiques.
- Outils : Python, Streamlit, NetworkX, PuLP.
- Résultats : carte interactive, statistiques dynamiques, analyse de graphes.

Annexe B : Exemple d'exécution

Cette section présente les étapes pour exécuter le projet localement. Toute personne disposant de Python 3.10 ou supérieur peut le faire en suivant les étapes ci-dessous. L'application se lance en local dans un navigateur via Streamlit.

1. Créer et activer un environnement virtuel

Cela permet d'isoler les dépendances du projet (donc fortement conseillé) :

Sous macOS / Linux :

```
python3 -m venv venv  
source venv/bin/activate
```

Sous Windows :

```
python -m venv venv  
venv\Scripts\activate.bat
```

Remarque : sur PyCharm, l'environnement virtuel est souvent activé automatiquement lors de l'ouverture du projet.

2. Cloner le projet depuis GitHub

```
git clone https://github.com/efkane/analyse_Bibliotheques_France.git
```

3. Installer les dépendances

Assurez-vous que le fichier `requirements.txt` est à jour avec toutes les bibliothèques nécessaires, puis exécutez :

```
pip install -r requirements.txt
```

4. Lancer l'application Streamlit

Démarrez l'interface interactive en local :

```
streamlit run app/analyse_bibliotheque.py
```

5. Accéder à l'application dans le navigateur

L'application s'ouvre automatiquement dans l'onglet d'un navigateur à l'adresse suivante :

`http://localhost:8501`

Si cela ne se lance pas automatiquement, ouvrez manuellement un navigateur et collez l'adresse ci-dessus.

6. Problèmes fréquents (FAQ rapide)

- Si vous obtenez une erreur liée à `streamlit`, vérifiez qu'il est bien installé dans l'environnement virtuel.
- En cas de problème de port déjà utilisé : utilisez `streamlit run app/analyse_bibliotheque.py -server.port 8502`.
- Sur certaines machines, Python doit être appelé avec `python3` au lieu de `python`.

Annexe C : Manuel utilisateur

L'utilisateur peut :

- Filtrer par région ou taille.
- Voir les bibliothèques sur la carte.
- Afficher des graphes.
- Lire les statistiques clés.
- Voir les ouvrages sur la carte.
- Parcourir le maximum d'ouvrages pour le plus court chemin.