Catégorisation de capteurs de pollution en fonction de leur environnement

Projet - UV ODATA

6 novembre 2023

Objectifs du projet

Une société spécialisée dans l'analyse des polluants chimiques de l'air a récemment déployé un réseau de capteurs de pollution dans la ville d'Aix-en-Provence et prévoit à court terme d'étendre de tels déploiements à d'autres grandes villes en France et à l'étranger. Cette société souhaite effectuer une classification non supervisée des capteurs de pollution en fonction de leur positionnement dans la ville, donc de leur environnement. L'environnement a un impact important sur les concentrations de polluants présents dans l'atmosphère et cette catégorisation devrait permettre de regrouper des capteurs délivrant des mesures assez proches.

Cette information sera très utile pour :
- Optimiser le déploiement des capteurs dans les villes (implantation de quelques capteurs dans les différents types d'environnement)
- Optimiser la calibration des capteurs (calibration par catégorie de capteurs plutôt que par capteur)
- Analyser les mesures fournies par les capteurs et en particulier détecter les anomalies en concentrations de polluants (anomalies liées à des dysfonctionnements de capteurs ou à des événements particuliers générant des émissions anormales de polluants)

Les méthodes de clustering à considérer / tester sont les suivantes :
- K-means
- Classification ascendante hiérarchique (CAH)
- Modèle de mélange de Gaussiennes
- DBSCAN
- Partitionnement spectral (Spectral clustering)

Parmi les méthodes considérées, seule la méthode Spectral clustering n'a pas été étudiée en cours, il faudra donc faire une recherche préalable sur cette méthode et comprendre son principe.

Ensuite, vous vous intéresserez à l'analyse et au clustering des capteurs à partir d'un fichier de données regroupant un certain nombre d'informations sur leur environnement : typologie de la zone (centre ville, centre ville historique, quartiers commerciaux, quartiers périphériques, zones résidentielles), nombre de voies, hauteur des bâtiments, trafic routier (véhicules légers, poids lourds, 2 roues), émission moyenne annuelle de polluants...

Vous analyserez les partitions obtenues par les différentes méthodes à l'aide des métriques de votre choix et d'un autre fichier de données qui stocke les mesures de concentration de particules fines. Enfin vous proposerez une liste d'environnements-types avec leurs principales caractéristiques.

Livrables attendus

A la fin du projet, vous devrez fournir 2 fichiers :

1. Un rapport (format pdf) comprenant les éléments suivants :
   — un rappel des objectifs du projet,
   — une description du protocole expérimental mis en place : objectifs, métriques utilisées, ... Pour chaque méthode vous préciserez les paramètres choisis et vous justifierez vos choix,
   — les réponses aux différentes questions posées,
   — une analyse et une interprétation des résultats obtenus.
2. Le code, clair et commenté (archive au format zip).

1 Etudes préalables

1.1 Méthode de réduction de dimension t-SNE

Pour visualiser les données et les résultats de clustering sur un plan en 2 dimensions, vous pourrez utiliser la méthode de l'ACP vue en cours, mais aussi la méthode t-SNE, également très utilisée en pratique.

Expliquez de façon succincte le principe de la méthode t-SNE. Indiquez ses avantages et limitations.

1.2 Méthode de clustering Spectral Clustering

Parmi les méthodes proposées pour effectuer le clustering des capteurs, seule la méthode Spectral clustering n'a pas été étudiée en cours.

Expliquez de façon succincte le principe de la méthode Spectral Clustering. Indiquez ses avantages et limitations.

1.3 Classes et modules relatifs aux différentes méthodes

Dans le projet, vous allez considérer les classes ou modules suivants des librairies `scikit-learn` et `scipy` :

- K-means : `sklearn.cluster.KMeans`
- CAH : `scipy.cluster.hierarchy`
- Modèle de mélange de Gaussiennes : `sklearn.mixture.GaussianMixture`
- DBSCAN : `sklearn.cluster.DBSCAN`
- Spectral clustering : `sklearn.cluster.SpectralClustering`

Etudiez ces différentes classes et modules : paramètres d'appel, attributs, méthodes, fonctions.

2 Catégorisation de capteurs de pollution en fonction de leur environnement

L'objectif est maintenant d'utiliser ces 5 méthodes de clustering pour classer automatiquement les capteurs de pollution de la ville d'Aix-en-Provence à partir de données qui caractérisent leur environnement. Ces données sont en partie collectées via des questionnaires complétés par les techniciens qui installent les capteurs dans la ville.

Cette classification non supervisée devrait permettre de regrouper des capteurs délivrant des mesures de concentration de polluants assez proches et constituer une aide à :

- l'optimisation du déploiement des capteurs dans les villes
- l'optimisation de la calibration des capteurs
- l'analyse des mesures délivrées par les capteurs et en particulier à la détection des anomalies en concentrations de polluants

Le fichier `donnees_environnement_capteurs.xlsx` stocke pour les 97 capteurs déployés dans la ville d'Aix-en-Provence les données suivantes :

- Typologie de la zone
- Type de voie
- Nombre de voies
- Distance capteur / voie
- Position capteurs
- Présence d'arbres
- Feuilles d'arbres
- Morphologie urbaine
- Hauteur des bâtiments
- Distance capteur / bâtiment
- Particularité
- Trafic routier/ véhicules légers (TMJA : trafic moyen journalier annuel)

— Trafic routier/ poids lourds (TMJA)
— Trafic routier/ 2 roues (TMJA)
— Emission moyenne annuelle (kg/maille/an) : données d'inventaire de 2017 fournies par ATMOSUD pour la quantité de PM2.5 sur la région d'Aix, chaque maille est de 1 km par 1 km.
— Données cartographie : données de modélisation de 2018 fournies par ATMOSUD pour la concentration de dioxyde d'azote (NO2) à une résolution de 25 m sur la région d'Aix (le NO2 est caractéristique du niveau de pollution en général)

Les données du trafic routier sont des données de comptage issues de la base routière modélisée de l'année 2020 à Aix-en-Provence.

Visualisez le tableau de données, puis importez les données du fichier à l'aide de la fonction `read_excel()` de `pandas`.

Les sections suivantes proposent une ligne directrice pour l'analyse et le clustering des données. Mais **le projet est ouvert, donc soyez curieux, n'hésitez pas à proposer, tester, expérimenter... Votre démarche, vos idées, votre analyse comptent plus que le résultat de clustering lui-même. Donnez toutes les informations qui vous paraissent pertinentes pour enrichir la connaissance sur les capteurs et leur environnement.**

2.1 Examen des données

Après l'importation, il est important d'examiner plus en détail les données, en particulier :
— la taille du jeu de données,
— le type des données (numérique : int, float ou qualitatif/catégoriel : object),
— la qualité des données (est-ce qu'il y a des données manquantes ?),
— la distribution des données (est-ce qu'il y a des données aberrantes ?).

En utilisant les méthodes de la classe `DataFrame`, procédez à l'examen des données et notez les informations qui vous paraissent pertinentes.

En particulier, il est important d'identifier les données qualitatives, les données manquantes (représentées par le symbole 'NA' : Not Available) et les données aberrantes. Si le fichier contient ce type de données, il faudra les pré-traiter (voir section suivante). En utilisant la méthode `isna()` de la classe `DataFrame` et la fonction `sum()`, vous pouvez obtenir le nombre de valeurs manquantes pour chacune des variables.

Il est aussi intéressant de connaître les statistiques des données à traiter. Pour cela, vous pouvez utiliser la méthode `describe()` de la classe `DataFrame` et construire une visualisation de type histogramme pour chaque variable numérique avec la méthode `hist()` de la classe `DataFrame`. Vous pouvez ainsi

identifier les éventuelles données aberrantes, c'est-à-dire en dehors de l'échelle de valeurs prises habituellement par une variable.

2.2 Pré-traitement des données

Pour faire fonctionner correctement les algorithmes de clustering, il est nécessaire d'avoir des données numériques de bonne qualité. Si besoin, il faut effectuer les opérations suivantes.

Données manquantes et aberrantes : Pour résoudre le problème des valeurs manquantes et des valeurs aberrantes, plusieurs solutions sont possibles :

- rechercher la vraie valeur via d'autres sources d'information,
- attribuer une valeur conforme à la distribution de la variable : moyenne, médiane, valeur la plus probable... (avec la méthode `fillna()` de la classe `DataFrame` pour les valeurs manquantes),
- supprimer la variable correspondante, si le nombre de valeurs manquantes ou aberrantes est très important (plus d'un tiers des données environ).

Transformation des variables qualitatives en variables numériques : Les algorithmes d'apprentissage ne traitent que des grandeurs numériques. Il faut donc transformer les variables qualitatives en variables numériques. Dans le cas de variables booléennes, le remplacement peut se faire directement avec la méthode `replace()`. Dans les autres cas, il est possible de faire appel aux classes suivantes : `LabelEncoder`, `OrdinalEncoder` ou `OneHotEncoder`. Il peut être intéressant de comparer les résultats obtenus avec les 2 types d'encodage : d'une part `LabelEncoder` ou `OrdinalEncoder` et d'autre part `OneHotEncoder`.

Normalisation des données : Un dernier point concernant la préparation des données est le recalibrage des variables. Lorsque les variables numériques ont des échelles différentes, il est nécessaire de centrer et réduire les données, en utilisant par exemple la classe `StandardScaler`.

Après avoir réalisé toutes ces transformations, il est intéressant d'examiner à nouveau toutes les variables qui seront utilisées pour le clustering.

2.3 Visualisation des données

Visualisez graphiquement les données sous la forme d'un nuage de points en 2 dimensions en utilisant l'ACP et la méthode t-SNE.

Commentez les représentations graphiques obtenues. Est-ce que visuellement des clusters apparaissent ?

2.4 Recherche de corrélations

Pour mieux comprendre les données, il faut s'intéresser aux relations qui existent entre les variables. Pour cela, il faut calculer le coefficient de corrélation entre chaque couple de variables numériques, par exemple avec la méthode `corr()` de la classe `DataFrame`.

Comme le nombre de variables est assez grand, la matrice de corrélation peut être entièrement représentée avec la fonction `heatmap()` de la librairie `seaborn`. De façon générale, cette librairie propose des fonctions intéressantes pour la visualisation des données.

Commentez les résultats obtenus.

2.5 Analyse exploratoire des données

Avant d'appliquer les méthodes de clustering, il est intéressant d'effectuer une ACP pour aller plus loin dans l'analyse des données. L'ACP va permettre de mieux comprendre les relations (corrélations) entre les variables, de faire des regroupements de capteurs aux environnements similaires et éventuellement de réduire la dimension des données.

Effectuez une ACP sur les données. Combien d'axes proposez-vous de conserver dans le cas d'une réduction de dimension ?
Représentez la projection des capteurs (la totalité ou une partie pour plus de lisibilité) dans le(s) premier(s) plan(s) principal(aux) ainsi que la projection des variables dans le(s) cercle(s) des corrélations. Donnez une interprétation des axes conservés.

2.6 Clustering des données

Effectuez le clustering proprement dit sur les données décrivant l'environnement des 97 capteurs avec les 5 méthodes proposées.

Pour chaque méthode :
- Précisez les valeurs choisies pour les paramètres.
- Evaluez la qualité des partitions obtenues pour différentes valeurs de K, le nombre de clusters, en utilisant les métriques de votre choix disponibles dans le module `sklearn.metrics`.
- Proposez une valeur de K, justifiez votre choix.
- Pour la valeur de K choisie, analysez et visualisez les clusters obtenus en 2 dimensions en utilisant l'ACP ou la méthode t-SNE.

Vous pouvez appliquer les algorithmes sur les données complètes (sans ACP) et/ou les données réduites (obtenues après réduction de dimension par ACP).

Comparez les résultats obtenus avec les 5 méthodes et **proposez une catégorisation des capteurs de pollution ainsi qu'une liste d'environnements-types avec leurs principales caractéristiques.**

Pour vérifier la pertinence de la catégorisation proposée, vous pouvez utiliser les données d'un autre fichier `donnees_mesures_PM2_5.xlsx`**.** Ce fichier stocke les mesures de concentration de particules fines PM2.5, c'est-à-dire les particules fines (particulate matter) dont le diamètre est inférieur à 2.5 microns. Ces particules sont dangereuses pour la santé parce qu'elles peuvent facilement pénétrer au plus profond des voies respiratoires. Il s'agit des mesures fournies tous les quarts d'heure par les 28 capteurs fonctionnant sur la période allant du 1er janvier au 26 septembre 2023. Le fichier ne contient pas les mesures de tous les capteurs parce qu'un certain nombre de capteurs ont été mis en fonctionnement plus tard au cours de l'année 2023. Vous pouvez utiliser les données de ce fichier comme bon vous semble.