

Sujets Programmation SIG Python sous QGIS

DataViz

L'objectif est d'étendre les capacités de QGIS en permettant à l'utilisateur d'avoir accès à des outils de visualisation des données. En l'occurrence, pour une variable quantitative on peut penser à des visualisations sous la forme d'histogrammes avec les courbes de distribution correspondantes, pour plusieurs variables on peut produire des « violins plots », des boîtes à moustache ou encore des camemberts ou des histogrammes multiples (fonctions matplotlib : boxplot, violinplot, hist, pie).

Pour aller plus loin, on pourra tester des modes de représentation plus « modernes » comme des treemaps, des sankey diagrams, des diagrammes de Venn, des radars, des nuages de mots... Pour cela, il faudra sans doute utiliser des bibliothèques comme pandas, wordcloud, squarify, matplotlib-venn.

Détails : L'utilisateur doit charger une couche vectorielle contenant des variables quantitatives dans QGIS. Votre code permet à l'utilisateur de sélectionner le mode de représentation souhaité, puis vous lui permettez de sélectionner les champs à traiter (pour les modes de représentation contenant plusieurs variables, on pourra lui poser la question dans un premier temps du nombre de variables qu'il souhaite étudier). Puis le code va générer l'image produite (voir s'il est pertinent de la faire sauvegarder par l'utilisateur) et l'afficher dans une boîte de dialogue.

Classification non supervisée (clustering)

L'objectif est d'étendre les capacités de QGIS en permettant à l'utilisateur de procéder à des classifications multivariées. En l'occurrence, pour différentes variables quantitatives on peut utiliser des méthodes relativement simples comme les classifications ascendantes hiérarchiques (CAH) ou les k-means (fonctions scipy : hierarchy.linkage, hierarchy.fcluster, vq.kmeans).

Pour aller plus loin, on pourra étudier des méthodes plus récentes très utilisées en machine learning comme le DBSCAN, le clustering spectral et potentiellement très rapides car conçues pour traiter des grands jeux de données comme Birch. Pour cela, il faudra chercher à utiliser des bibliothèques comme scikit-learn.

Détails : L'utilisateur doit charger une couche vectorielle contenant plusieurs variables quantitatives dans QGIS. Votre code va demander à l'utilisateur de sélectionner la méthode de classification voulue, puis les champs concernés par la classification (dans un premier temps on pourra utiliser tous les champs du shapefile) et le nombre de classes souhaitées. Ensuite votre code effectue la classification et va créer un champ de type entier dans la couche pour stocker le résultat. Il faudra aussi produire une boîte de dialogue indiquant les principales caractéristiques des classes créées. L'utilisateur pourra alors utiliser ce champ pour cartographier le résultat du clustering. Essayez de trouver des données pertinentes à tester.

Optimisation linéaire

L'objectif est d'étendre les capacités de QGIS en permettant à l'utilisateur de résoudre de manière exacte des problèmes d'optimisation que l'on retrouve par exemple dans les modèles de localisation-allocation. En l'occurrence, on va se focaliser sur le problème de transport qui est un de ces problèmes les plus « simples » à résoudre à l'aide du simplexe ou du network simplexe (fonctions scipy ou networkx : linprog ou network_simplex).

Pour aller plus loin, il est possible de chercher à résoudre des problèmes plus complexes, car il requiert par exemple le respect de contraintes comme le fait que les variables doivent être binaires ou entières, on parle alors d'optimisation linéaire en nombre entier et les solveurs exacts sont beaucoup plus durs à coder... Pour cela, il est possible d'utiliser la bibliothèque Pulp pour résoudre le problème p-median par exemple.

Détails : L'utilisateur doit charger dans QGIS trois fichiers de type vectoriel : deux fichiers de points (contenant des identifiants et des quantités (demandes ou stocks)) et un fichier de lignes (s'appuyant sur les identifiants des fichiers de points et contenant les coûts de ces relations). Votre code permet à l'utilisateur de choisir la couche des nœuds de départ avec les stocks, la couche des nœuds d'arrivée avec les demandes, la couche de lignes avec les coûts associés (ensuite il sera possible de demander à l'utilisateur de choisir les champs à utiliser, dans un premier temps on pourra simplement obliger l'utilisateur à avoir une certaine structure de données). Votre code effectue le calcul d'optimisation à partir de ces données et renvoie dans le fichier de lignes les quantités affectées à chaque relation et un message box qui donne le coût total de la solution.

Statistiques multivariées

L'objectif est d'étendre les capacités de QGIS en permettant à l'utilisateur d'effectuer des analyses statistiques pertinentes pour des tableaux d'information géographique. En l'occurrence on va se focaliser sur les trois principaux tests statistiques en géographie : le test de corrélation, le test du chi2, le test Fischer-Snedecor (fonctions scipy : pearsonr, chisquare, f_oneway).

Pour aller plus loin, il est possible de dépasser les analyse bivariées et de passer aux analyses multivariées. Pour cela, il est possible d'utiliser des bibliothèques comme pandas, scikit-learn, statsmodels pour faire des régressions linéaires multiples, des ancova...

Détails : L'utilisateur doit charger dans QGIS un fichier vectoriel qui contient différents champs (pour effectuer vos calculs, travaillez avec un shapefile comportant au moins deux champs quantitatifs et deux champs qualitatifs). Votre code permet à l'utilisateur de choisir deux champs pour étudier leur relation (vous pouvez dans un premier temps envisager de demander à l'utilisateur les types des variables : quantitatifs ou qualitatifs ; néanmoins on peut aussi faire ce travail pour lui...). Votre code effectue le calcul du test approprié et renvoie le résultat sous la forme d'un message box qui donne notamment la p-value et indique si le résultat semble significatif pour un risque de 5%. Pour vos données qualitatives, pensez à avoir un champ « Région » qui a la propriété de regrouper plusieurs objets géographiques. Essayez de trouver des données pertinentes à tester.