Compte rendu de projet

Université Grenoble Alpes

Informatique, Mathématiques et Mathématiques Appliquées de Grenoble

Logiciel Spécialisé R

Auteur:
Sébastien Calvignac
Kassim Kone

Enseignant :
M. Rémy Drouilhet

January 9, 2022

Contents

1 Résumé

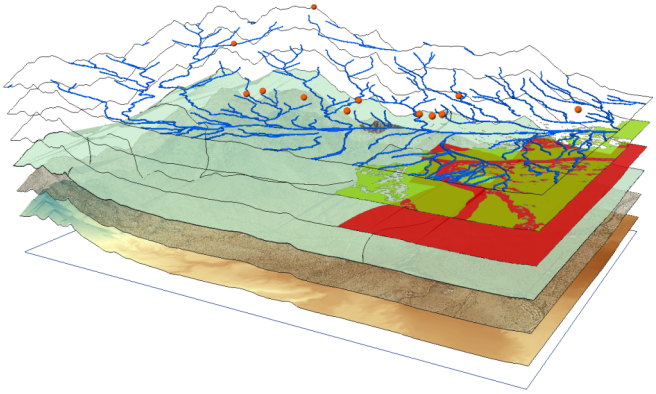
Dans le cadre du Master de Statistique et Sciences des Données, nous nous formons à la génération de rapport et de dévéloppement en R avec M. Drouilhet. Nous allons nous intéresser dans ce rapport au sujet de l'analyse spatiale de données géolinguistiques et géographiques. Et dans un second temps à l'experience de développement d'un package R pour une application Shiny.

2 Géomatique

2.1 Définition

La géomatique regroupe l'ensemble des outils mathématiques permettant d'acquérir, représenter et analyser des données géographiques.

Figure 1: Visualisation de couches de données géographiques



2.2 Occitanie

Occitanie est une zone géographique au sud de la France qui a sa propre langue appelé l'occitan avec différent dialectes.

Figure 2: La région d'Occitanie

2.3 Données

Un sondage qui date du début du 20ème siècle, a enregistré quel mots était utilisé dans chaque village pour exprimer une même idée. Avec ces données nous pouvons visualiser quels mots étaient utilisé dans l'espace géographique. Chaque village est associé à des coordonnées géographiques.

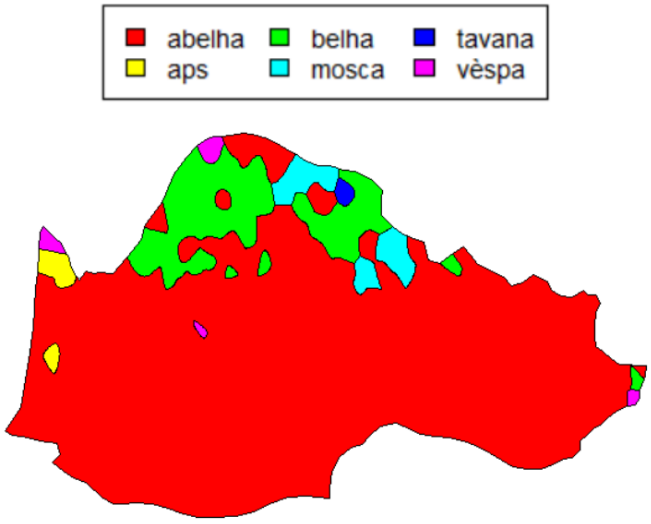
Figure 3: Représentation des données

	notion 1	notion 2	notion 3
village 1	lemme	lemme	lemme
village 2	lemme	lemme	lemme
village 3	lemme	lemme	lemme

2.4 Cartographie

Nous avons donc des villages, des lemmes et des coordonnées de villages. On trace ensuite des frontières avec les coordonnées des villages utilisant un même lemme pour une notion donnée et en fesant attention aux enclaves.

Figure 4: Localisation des 6 lemmes de la notion "abeille"



2.5 Statistiques

On connais donc pour chaque m2 de l'occitanie quel est la façon d'exprimer une notion. On souhaite maitenant mesurer mathématiquement la concordance avec des frontières géographiques. Cela nous permet de dire si tel ou tel frontière épousent les mêmes frontières. Pour ce faire il faut calculer la somme des aires qui intersectionne (en commun) de chaque lemme avec chaque section (exemple de section peut être le departement de l'aveyron). Ce calcule retourne une matrice de contigence. Tout nos calculs statistiques sont basées sur cette matrice. Le calcul de Khi2, entropie, indice de localisation etc. sont des façon différentes de mesurer la dépendance et dispersion entre deux variables.

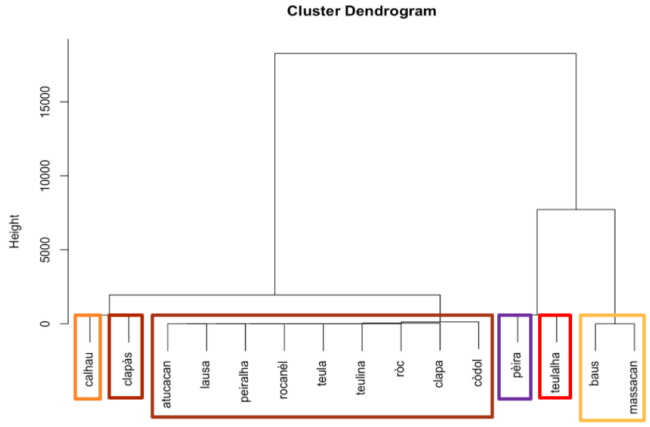
Figure 5: Matrice de l'intersection des aires (km^2) "Abeille" X régions dialectales

	Lemosin	Auvernhat	Vivaroaupenc	Provencau	Lengadocian	Gascon
abelha	9737.8949	9174.7675	17985.7083	28980.1016	57176.34413	43716.357
aps	0.0000	0.0000	0.0000	0.0000	0.00000	1758.919
belha	13685.3297	5592.1411	342.0969	289.3196	74.15037	0.000
mosca	571.5372	3377.9414	1228.9462	0.0000	0.00000	0.000
tavana	0.0000	499.3834	0.0000	0.0000	0.00000	0.000
vèspa	645.2929	0.0000	0.0000	233.8120	154.39037	516.222

2.6 Dendrogramme

Cette partie n'est pas terminé mais nous pouvons parler de son objectif. On souhaite regrouper les lemmes qui sont le plus similaire, c'est à dire, ceux ont une grande aire d'intersection dans les mêmes sections. On pourrai donc passer de 20 lemmes par exemple à 6 lemmes, puis recontruire la carte avec que 6 couleurs. D'autre part, le clustering pourai theoriquement être utilisé pour recontruire des frontières géographiques en se basant uniquement sur les donnée geolinguistiques.

Figure 6: Clustering des lemmes les plus similaires dans leurs dispersion géographique



3 Dévéloppement

3.1 Shiny

Shiny permet de créer des dashboards en R. Il separe l'interface graphique et le backend avec une partie ui et une partie server. Le deux parties sont connectés par des que vous avez a definir pour chaque element en entrée et ou sortie. L'idée générale et de contruire dans le UI la partie saisie par le client. Puis dans le serveur on fait appel aux fonction définie dans /R qui font les calculs et les graphiques etc. Enfin l'UI définié la disposition des ces sorties sur la page.

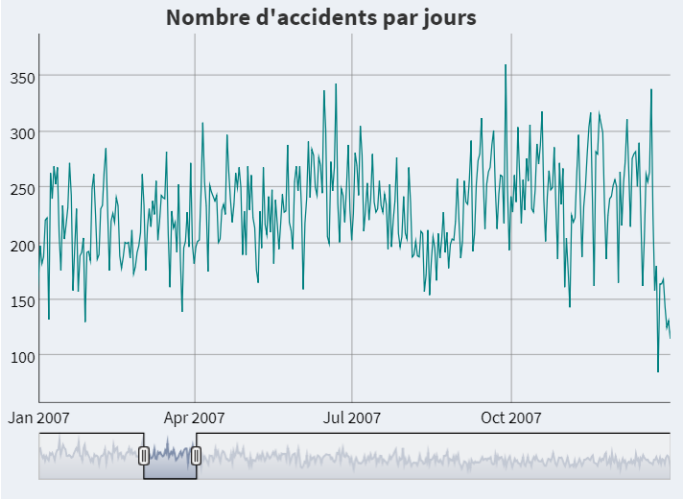
3.2 Golem

golem est un package qui permet la création d'une application "Shiny" prête pour la production. golem, facilite le travail en équipe grâce a la séparation de l'application en modules, qui contiennent chaqu'un la partie UI et serveur. Ces modules peuvent être dévéloppés indépendament et intégrés rapidement. Golem a plein de fonctionnalités tel que pour le deploiement qui facilite le travail du développeur. Et aussi pour gérér les dépendances en écrivant le NAMESPACE et DESCRIPTION.

3.3 Dygraph

dygraph est une bibliothèque de graphiques JavaScript. Nous l'utilisons pour un graphique de données temporelles interactif. Le jeux de données est à propos d'accidents de voiture en France. L'intégration avec l'application est s'est fait rapidement puisque nous lui avons attribué un module à part entière. Avec rétrospection, j'aurai plutôt préféré utilisé le package plotly car cela fait partie d'une famille d'outils d'analyse de données plus vaste que dygraph. Nous avons perdu beaucoup de temp sur l'analyse de données des données accidents mais la grande mojorité n'a pas été inclu car il s'agissait d'analyse descriptives sans aucune composante intéractive et donc n'a pas d'intérêt pour une application shiny.

Figure 7: Graphique de données chronologique et intéractif avec dygraph



3.4 Git

C'est le premier projet où j'utilise vraiment les fonctionnalités de gestionnaire de version et je suis agréablement impréssionné.

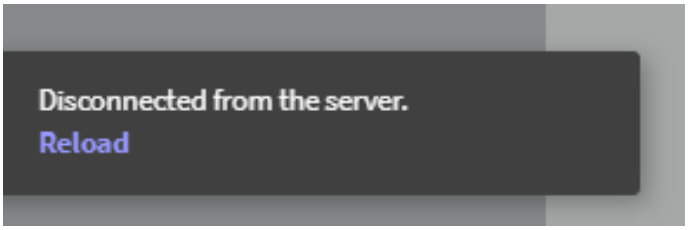
3.5 Shinyapps io

Le déploiment de l'application se fait facilement grâce a shinyapps.io et golem. shinyapps.io propose des services de déploiement d'application shiny en suivant un modèle freemium. Par conséquence leur service gratuit est bridé et peut créer des dysfonctionnements lorsque de la puissance de calcul est requis.

3.6 Rcpp

Nous obtenons en effet des problèmes d'optimisation quand l'application est déployé sur le service gratuit de shinyapps.io. Notament pour notre graphique interactif et lorsque on modifie le découpage géographie de la carte. Ces deux zones de dysfonctionnement sont notées par "1min de chargement" et provoquent une déconnexion du client.

Figure 8: Message d'erreure sur shinyapps.io lors du lancement d'un processus trop lourd



L'optimisation de l'application est fortement conseillé si l'on passe en production car les temps de chargement sont parfois trop longs (même sur une machine puissante). Bien que le bridage soit partiellement responsable, il est possible de contourner ce problème par l'optimisation de l'application avec rcpp par exemple. Rcpp permet d'intégrer des fonctions écrites en C++ plûtot que R pour augmenter la vitesse du programme. Nous avons tenté d'utiliser Rcpp pour faire des calculs en relation avec 'collatz conjecture' qui dit que pour tout entier naturel supérieure à 1, si on applique récusivement l'operation 3n+1 quand n est est impaire, et n/2 quand n est pair alors on fini toujours par tomber sur 1. Ça n'a jamais été démontré mathématiquement. Nous n'avons pas terminé cette partie mais nous aurions souhaité reproduire une visualisation comme celle ci dessous par exemple.

Figure 9: Message d'erreure sur shinyapps.io lors du lancement d'un processus trop lourd

3.7 CSS

dans inst->app->www->costum.css vous trouverez quelques styles pour la table statistque et du text. Grâce a golem, il n'y a pas de concurrence de balise html classe ou id.

3.8 Roxygen2

Pour la documentation des fonctions et de données de notre package nous utilisons roxygen2. De manière générale il suffit d'ajouter des commentaires avant votre fonction dans un format bien particulier mais facile a utiliser. Dans ces commentaires, vous pouvez décrirer la fonction, ses paramètres et ses sorties etc. Une fois que vous avez fait le travail de correctement doccumenter votre package, roxygen2 s'occupe de créer le ficher de documentation r (.Rd). C'est des fichier avec du latex.