

Rencontres R 2024

12-14 juin 2024, Vannes, France



Table des matières

Comités	1
Tutoriels	1
Créez des environnements reproductibles avec rix, Rodrigues Coelho Bruno André	1
Dérivation automatique et optimisation avec la librairie torch, Mary-Huard Tristan	5
Conférences invitées : Cécile Proust Lima - Modératrice Solène Desmée	6
Modélisation conjointe de données longitudinales et de temps d'événements sous R, Proust-Lima Cécile	6
Session Biostatistique - Modératrice Julie Aubert	7
Extending code from the saemix package to fit parametric joint models in R, Lavallée-Morelle Alexandra [et al.]	7
Mise en oeuvre de méthodes semi-Bayésiennes de calcul des erreurs standards pour les données éparques dans le package saemix, Guhl Mélanie [et al.]	9
BeQut, un package R pour l'estimation bayésienne de modèles de régression quantile à effets mixtes via JAGS, Barbieri Antoine [et al.]	11
Session Reproductibilité - Modératrice Aurélie Siberchicot	13
Créez des environnements reproductibles avec rix, Rodrigues Coelho Bruno André	13
Session Courte 1 - Modérateur Vincent Brault	16
Une enquête auprès des métiers de la " data " : quelle place pour R et ses utilisateurs ?, Girard Antoine	16
Pour un namespace tout en souplesse, Floc'hlay Swann	19
ProteoBayes : un cadre bayésien pour l'analyse protéomique différentielle, Chion Marie [et al.]	21
Cadre R chez IMPACT Initiatives, Say Yann	23
Comment les communautés autour de R peuvent changer vos projets, Vaugoyeau Marie	25
Session Poster	27
hubeau : un package pour interroger les APIs du Système d'Information sur l'eau en France, Dorchies David [et al.]	27
SK8 : Un service institutionnel de gestion et d'hébergement d'applications Shiny, Rey Jean-François [et al.]	30
Créer un site pour partager sa recherche avec R, blogdown et Hugo, Ollivier Fanny	32
Les packages autour de JDemetra+ (rjd3) : une boîte à outils complète pour l'analyse des séries temporelles, Barthelemy Tanguy	33
Poster autour du package {datamods}, Goumri Samra [et al.]	35
Distance/Divergence entre distributions t multivariées, Santagostini Pierre [et al.]	36

A survey translation tool to easily migrate from Qualtrics to LimeSurvey, Straboni Camille	38
RecForest : Forêts aléatoires de survie pour l'analyse des événements récurrents en R, Murris Juliette [et al.]	41
Des tableaux et des graphiques prêts à publication avec les packages R {tabulatrise} et {chart} de la suite SciViews, Engels Guyliann [et al.]	43
Conférences invitées : Nicolas Raillard - Modératrice Audrey Poterie	45
R pour l'océano-météo et l'ingénierie marine, Raillard Nicolas	45
Session Dataviz - Modératrice Julie Lenoir	46
telraamStats : visualisation des mobilités pour la recherche et les citoyen·ne·s, Guichard-Sustowski Ketsia [et al.]	46
Collecter et cartographier les données du bilan carbone d'un congrès, Friguet Chloé [et al.]	48
Session Méthode statistique - Modérateur Vincent Brault	50
Estimation de quantiles conditionnels extrêmes : Package ExtremeFit, Durrieu Gilles [et al.]	50
Clustering sur données incomplètes avec clusterMI, Audiger Vincent	52
Créer son propre package d'extension {recipes} : retour d'expérience de {scimo}, Bi-chat Antoine [et al.]	54
Smooth testing and clustering of copulas, Ngounou Bakam Yves Ismaël [et al.]	55
Conférences invitées : Elise Maigné - Modérateur Paul Bastide	57
SK8 : Pour des applications shiny qui se déploient comment sur des roulettes, Maigné Elise	57
Session Courte 2 - Modérateur François Husson	58
Explorer et comparer des cartes de zones climatiques locales avec le paquet lczexplore, Gousseff Matthieu [et al.]	58
Des applications Shiny qui facilitent la vie, Dechaux Terence	60
Application Shiny XPBlocs - Création de blocs en expérimentation, Legris Maxime	62
Utilisation du package {flexdashboard} pour le contrôle des données de biologie dans un entrepôt de données de santé, Pierre-Jean Morgane [et al.]	64
easy16S : une application Shiny pour explorer ses données métagénomiques, Middoux Cédric [et al.]	66
Session Statistique pour l'environnement - Modérateur Baptiste Alglave	68
Human in the deep : Converting research activities pressures into ecological impact assessment., Leroux Riwan [et al.]	68
Maturation de codes scientifiques R de traitement de données liées à l'Eau au BRGM (initiative MATUREAU), Laurencelle Marc [et al.]	70
Un suivi régionalisé et actualisé des étiages des petits cours d'eau, c'est possible avec R, Hub'Eau et GitHub!, Richard Benoît [et al.]	74
Session Gestion de projet R, bonnes pratiques - Modérateur Pierre Gloaguen	76
Sécurisation des analyses statistiques avec R : retour d'expérience, Philibert Au- rone [et al.]	76
Refactoring : du code qui marche, c'est bien, mais du code maintenable, c'est mieux, Guyader Vincent	78

Une vie polyamoureuse entre R et Julia, Drouilhet Remy	81
Session Aide à la vie de tous les jours et à l'enseignement - Modératrice Audrey Lavenu	84
Un petit coup de polish - Nettoyage de fichiers Excel avec R, Vroylandt Thomas {saperlipopette}, un paquet R pour progresser en Git en toute sérénité, Salmon Maëlle	84
Génération aléatoire d'exercices de biostatistiques pour Moodle via le package SARP.Moodle de R, Curis Emmanuel [et al.]	85
.	87
Session Shiny - Modératrice Swann Floc'hlay	89
Microstructure Information from Diffusion Imaging, Stamm Aymeric	89
Esquisse, un outil de visualisation, Perrier Victor	92
webR, et le futur des apps web avec R, Fay Colin	94
Conférences invitées : Philippe Grosjean - Modératrice Chloé Friguet	96
Apprendre R et les statistiques... grâce à R, Grosjean Philippe	96
Liste des auteurs	98
Sponsors	100

Comité scientifique :

- Paul BASTIDE, IMAG-CNRS, Montpellier
- Vincent BRAULT, IUT SD, Univ Grenoble-Alpes, LJK, Grenoble (président)
- Marie CHION, University of Cambridge, UK
- Solène DESMEE, IUT GB, Univ. Tours, SPHERE-INSERM, Tours
- David GOHEL, Fondateur ARDATA, Paris
- François HUSSON, Institut Agro, IRMAR, Rennes
- Audrey POTERIE, IUT SD, Univ. Bretagne Sud, LMBA, Vannes
- Geneviève ROBIN, LaMME-CNRS Evry et Owkin, Paris

Comité d'organisation :

- Baptiste ALGLAVE IUT SD, Univ. Bretagne-Sud, Lab-sticc, Vannes
- Anne CUZOL, IUT SD, Univ. Bretagne Sud, LMBA, Vannes
- Chloé FRIGUET IUT SD, Univ. Bretagne-Sud, IRISA, Vannes (présidente)
- Pierre GLOAGUEN, UFR SSI, Univ. Bretagne Sud, LMBA, Vannes (trésorier)
- Audrey POTERIE, IUT SD, Univ. Bretagne Sud, LMBA, Vannes
- François SEPTIER, UFR SSI, Univ. Bretagne Sud, LMBA, Vannes

Supports administratifs :

- Pauline GORRE, IE appui à la Recherche/thématique IA, DRUID, Univ. Bretagne Sud, Vannes
- Antoine L'AZOU, gestionnaire, IRISA-CNRS, Rennes
- Aurélie JOUBEL, gestionnaire, LMBA, Univ. Bretagne Sud, Vannes

Comité de pilotage :

- Julie AUBERT - AgroParisTech
- Rémy DROUILHET - Université Grenoble Alpes
- Robin GENUER - Université de Bordeaux
- Francois HUSSON - Institut Agro, Rennes
- Julie JOSSE - INRIA, Montpellier
- Aurélie SIBERCHICOT - Université Lyon 1

Créez des environnements reproductibles avec rix

Bruno Rodrigues

Résumé

S’assurer que nos analyses soient reproductibles est essentiel, et il existe une multitude d’outils pour les utilisateurs de R que nous sommes pour le permettre. Néanmoins, ces outils ne gèrent qu’un seul aspect du continuum de la reproductibilité : {renv}, par exemple, permet de “figer” les paquets R pour une analyse, mais pas la version de R elle-même. Le “R Installation Manager” permet d’installer n’importe quelle version de R sur n’importe quel système ; Docker permet de containeriser le tout dans une image depuis laquelle on peut exécuter des conteneurs. Nix, développé par Dolstra et al. (2004) est un outil qui permet de gérer chacune de ces dimensions en même temps : il permet de définir un environnement complet comprenant R, les paquets R et les dépendances système sous-jacentes, et de déployer cet environnement de manière totalement reproductible. Malheureusement, Nix peut sembler très compliqué pour des novices, c’est pourquoi j’ai développé le paquet {rix}, qui permet de définir des environnements de développement pour R de manière très simple. Dans cette présentation, j’expliquerai comment Nix assure la reproductibilité d’une analyse et comment on peut l’utiliser simplement grâce à {rix}.

Mots-clefs : Package - Reproductibilité

Développement

{rix} est un paquet R qui tire parti de Nix, un puissant gestionnaire de paquets axé sur la reproductibilité. Avec Nix, il est possible de créer des environnements spécifiques à un projet contenant une version spécifique de R et des paquets R (ainsi que d’autres outils ou langages, si nécessaire). Vous pouvez utiliser {rix} et Nix pour remplacer {renv} et Docker par un seul outil. Nix est un logiciel incroyablement utile pour garantir la reproductibilité des projets. Par exemple, il permet d’exécuter des applications web telles que des applications Shiny ou des API plumber dans un environnement contrôlé.

Nix a un coût d’entrée assez élevé cependant. Nix est un logiciel complexe qui dispose de son propre langage de programmation, également appelé Nix. Son objectif est de résoudre

un problème complexe : définir des instructions sur la manière de construire des logiciels et de gérer les configurations de manière déclarative. Cela garantit que le logiciel est installé de manière entièrement reproductible, sur n'importe quel système d'exploitation ou matériel.

{rix} fournit des fonctions pour vous aider à écrire et déployer des expressions écrites dans le langage Nix. Ces expressions seront les entrées du gestionnaire de paquets Nix, pour construire des ensembles de paquets logiciels et les fournir dans un environnement de développement reproductible et cohérent. Ces environnements peuvent être utilisés pour l'analyse de données interactive, ou reproduits lors de l'exécution de pipelines dans des systèmes CI/CD. Dans la collection “nixpkgs” (l'équivalent du CRAN pour Nix), il y a actuellement plus de 80 000 logiciels disponibles via le gestionnaire de paquets Nix. Avec {rix}, vous pouvez définir et construire des environnements R isolés via le gestionnaire de paquets Nix avec facilité. Ainsi, les environnements contiennent R et tous les paquets requis dont vous avez besoin pour votre projet. Vous pouvez également ajouter n'importe quel autre logiciel nécessaire à votre analyse. L'écosystème R de Nix comprend actuellement la quasi-totalité des paquets CRAN et Bioconductor. Il est également possible d'installer des versions antérieures des paquets R, ou d'installer des paquets depuis GitHub à des commits définis.

Le gestionnaire de paquets Nix est extrêmement puissant. Non seulement il gère très bien toutes les dépendances de n'importe quel paquet de manière déterministe, mais il est également possible avec lui de reproduire des environnements contenant des versions anciennes de logiciels. Il est ainsi possible de construire des environnements contenant la version 4.0.0 de R (par exemple) pour exécuter un ancien projet qui a été développé à l'origine sur cette version de R.

Si vous avez besoin d'autres outils ou langages comme Python ou Julia, cela peut également être fait facilement. Nix est disponible pour Linux, macOS et Windows (via WSL2) et {rix} présente les fonctionnalités suivantes :

- permet d'installer n'importe quelle version de R (depuis la version 3.0.2) et des paquets R pour des projets spécifiques ;
- avoir plusieurs versions de R et des paquets R installées en même temps sur le même système ;
- définir des environnements de développement complets en code et les utiliser n'importe où ;
- exécuter des fonctions R individuelles dans un environnement différent (éventuellement avec une version différente de R et des paquets R) depuis une session R interactive, et récupérer la sortie de cette fonction en utilisant `with_nix()`;

{rix} ne nécessite pas que Nix soit installé sur votre système pour générer des expressions. Cela signifie que vous pouvez générer des expressions sur un système sur lequel vous ne pouvez pas facilement installer de logiciel, puis utiliser ces expressions sur le cloud ou dans un environnement CI/CD pour y construire le projet.

Pour définir un environnement il suffit d'utiliser la fonction `rix()`:

```
rix(r_ver = "4.3.1",
    r_pkgs = c("dplyr", "chronicler"),
    ide = "other")
```

Ceci va générer un fichier appelé `default.nix` qui sera ensuite utilisé par le gestionnaire de paquets Nix pour installer R version 4.3.1 ainsi que `{dplyr}` et `{chronicler}` (tels qu'ils étaient à la sortie de cette version de R). Toutes les autres dépendances systèmes, telles que des librairies dynamiques ou compilateurs nécessaires pour générer cette environnement seront aussi installés.

Références

Dolstra, Eelco, Merijn De Jonge, Eelco Visser, et al. 2004. “Nix: A Safe and Policy-Free System for Software Deployment.” In *LISA*, 4:79–92.

Différentiation automatique et optimisation avec la librairie torch

Tristan Mary-Huard 1*

Résumé Dans ce tutoriel, nous verrons comment utiliser le package `torch` pour calculer automatiquement le gradient d'une fonction, et comment utiliser ce gradient pour réaliser l'optimisation d'une fonction.

Mots-clefs : Statistique - Ingénierie - IA

Développement

`torch` est une implémentation R de la célèbre librairie Python PyTorch, offrant un environnement dédié à l'apprentissage des réseaux de neurones. Ce tutoriel se concentre particulièrement sur les outils de différentiation automatique et d'optimisation disponibles dans `torch`. En utilisant la “chain rule” (ou théorème de dérivation des fonctions composées), la différentiation automatique permet de calculer efficacement le gradient d'une fonction, sans recourir à la différentiation symbolique ou à des approximations numériques. Ceci simplifie le code en éliminant le besoin pour l'utilisateur de calculer et d'implémenter manuellement les dérivées. De plus, `torch` implémente des méthodes d'optimisation classiques (Adam, L-BFGS, Stochastic Gradient Descent, etc.), et fournit donc une librairie complète pour les statisticiens développant leurs propres modèles. Ce tutoriel sera illustré par un exemple d'application à la régression logistique.

Prérequis :

- Savoir réaliser un modèle glm (par exemple la régression logistique)
- Connaitre les bases de la modélisation en statistique (par exemple le modèle de régression logistique)
- Connaitre les bases de la dérivation

Durée : 2h

Avant de venir au tutoriel, il faudrait installer les packages suivants :

- `torch`
- `tidyverse` (optionnel mais recommandé)

Note : le calcul sur GPU ne sera pas considéré dans cet atelier.

*Université Paris-Saclay, INRAE, CNRS, AgroParisTech, UMR GQE-Le Moulon, 91190 Gif-sur-Yvette, France, Université Paris-Saclay, AgroParisTech, INRAE, UMR MIA-Paris, 91120 Palaiseau, France, tristan.mary-huard@agroparistech.fr

Modélisation conjointe de données longitudinales et de temps d'événements sous R

Cécile Proust-Lima*

Résumé

Les études en santé impliquent généralement la collecte de variables mesurées de manière répétée au fil du temps. Cela inclut des expositions (e.g., traitement, pression artérielle, nutrition), des marqueurs de progression (e.g., volumes cérébraux, marqueurs sanguins, taille des tumeurs, score de qualité de vie) et des délais jusqu'à des événements cliniques (e.g., décès, diagnostic, rechute). L'analyse jointe de ces données longitudinales et de ces temps d'événement se fait par des modèles dits conjoints qui prennent en compte la corrélation entre les processus en jeu¹. Les modèles conjoints sont devenus au fil des ans un outil essentiel en biostatistique car ils permettent d'aborder diverses questions prédictives, descriptives et analytiques. Cela inclut la prédiction du risque d'événement basée sur des marqueurs ou expositions mesurés de manière répétée au fil du temps, la modélisation de la progression de marqueurs tout en tenant compte d'une sortie d'étude informative, ou la description et compréhension de la structure d'interdépendance qui peut exister entre plusieurs processus.

Dans cette présentation, je vais introduire le principe de la modélisation jointe et décrire différentes approches proposées dans la littérature en m'appuyant sur les solutions R associées. Seront abordés les modèles à effets aléatoires partagés pour lesquels divers packages R existent (e.g., JM¹, JMbayes2 , INLAjoint²), les modèles à classes latentes avec le package lcmm^{3,4}, ainsi qu'une alternative aux modèles conjoints via des forêts aléatoires de survie avec le package DynForest^{5,6}.

Mots-clefs

biostatistique, modèles mixtes, classes latentes, forêts aléatoires, modèles conjoints

Références

1. Rizopoulos, D. *Joint Models for Longitudinal and Time-to-Event Data with Applications in R*. (CRC Press, Boca Raton, 2012).
2. Rustand, D., van Niekerk, J., Krainski, E. T., Rue, H. & Proust-Lima, C. Fast and flexible inference for joint models of multivariate longitudinal and survival data using integrated nested Laplace approximations. *Biostatistics* kxad019 (2023) doi:10.1093/biostatistics/kxad019.
3. Proust-Lima, C., Philipps, V. & Liquet, B. Estimation of Extended Mixed Models Using Latent Classes and Latent Processes: The R Package lcmm. *Journal of Statistical Software, Articles* **78**, 1–56 (2017).
4. Proust-Lima, C. et al. Describing complex disease progression using joint latent class models for multivariate longitudinal markers and clinical endpoints. *Stat Med* **42**, 3996–4014 (2023).
5. Devaux, A., Proust-Lima, C. & Genuer, R. Random Forests for time-fixed and time-dependent predictors: The DynForest R package. *arXiv* (2023) doi:10.48550/arXiv.2302.02670.
6. Devaux, A., Helmer, C., Genuer, R. & Proust-Lima, C. Random survival forests with multivariate longitudinal endogenous covariates. *Stat Methods Med Res* **32**, 2331–2346 (2023).

* Univ. Bordeaux, Inserm, Bordeaux Population Research Center, Bordeaux, France, cecile.proust-lima@inserm.fr

Extending code from the saemix package to fit parametric joint models in R

Alexandra Lavalley-Morelle, France Mentré, Jimmy Mullaert* Emmanuelle Comets†

Résumé

The **saemix** package ([Comets et al., 2017]) was developed to implement the Stochastic Approximation Expectation-Maximisation (SAEM) algorithm to model longitudinal data in R. Version 3 extended the types of outcome to include non-Gaussian data including count, categorical or time-to-event data, with the current version of the package limited to handling one response at at time. In this work, we showcase how to perform parameter estimation, build models and evaluate them using the functions provided in the CRAN version of the package. We also extended the code to joint models of longitudinal and time-to-event data, and implemented the stochastic approximation in [Delattre and Kuhn, 2023] to provide standard errors of estimation. A simulation study was performed to evaluate the performance on the algorithm with increasing model complexity. We present an application to data from hospitalised Covid19 patients ([Lavalley-Morelle et al., 2022]), with a joint model including three biomarkers and a competing risk framework for two events (discharge and death). The extended code, available on the github for **saemix** development, shows the flexibility and scope of **saemix** to fit longitudinal data.

Mots-clefs : Non-linear mixed effect models – Stochastic Approximation EM algorithm – Package – Model evaluation – Non Gaussian outcomes

Introduction : Non-linear mixed effect models are used in many fields, including agronomy, animal breeding, imagery and PKPD analyses. The **saemix** package in R computes the maximum likelihood estimator of the population parameters without any approximation of the model ([Comets et al., 2017]). Version 3, available on CRAN, extended the types of outcomes handled by **saemix** to non-Gaussian outcomes ([Karimi et al., 2020]). Recently, joint models have become increasingly popular to link repeated measures of biomarkers to the occurrence of terminal events. In R however, with many packages the function used to describe the biomarker dynamic is mainly linear in the parameters, and the survival submodel relies on pre-implemented functions. The objective of this work is to (i) present the **saemix** package through two examples with different outcome types and (ii) extend the code to fit parametric joint models where longitudinal submodels are not necessary linear in their parameters, evaluating the extension through a simulation study and a real data application.

Methods : **saemix** implements the SAEM algorithm for parameter estimation in (non)linear mixed effects models. It provides standard errors (SE) from the Fisher Information Matrix (FIM) and via bootstrap methods ([Comets et al., 2021]). The conditional distributions of the individual parameters are estimated using the Hastings-Metropolis algorithm. Inference and automated model building ([M Delattre, 2014]) use the log-likelihood which can be estimated by different approaches. Model diagnostics for continuous outcomes are available through the **npde** package ([Comets et al., 2008]) and some simulation-based diagnostics have been implemented for non-Gaussian outcomes. In this work, we will show how to use the package to model continuous dose-response data and repeated binary data.

In the second part of this work, we extend the main functions of **saemix** to joint model estimation (<https://github.com/saemixdevelopment/saemixextension>), keeping the flexibility for users

*Université Paris Cité, INSERM, IAME, F-75018 Paris, France

†Université Paris Cité, INSERM, IAME, F-75018 Paris, France; Université de Rennes, Inserm, EHESP, Irset - UMRS 1085, F-35000 Rennes, France emmanuelle.comets@inserm.fr (presenting author)

to define any parametric model and link function between longitudinal and survival parts. To compute standard errors (SE) of parameter estimates, we implemented a recently developed stochastic algorithm requiring only first derivatives ([Delattre and Kuhn, 2023]). We assessed in a simulation study (i) the relative bias and relative root mean square errors of the estimated parameters, (ii) the accuracy of the estimated SE and (iii) the adequacy of the type I error when testing independence between the two submodels. Four joint models were considered in the simulation study, combining a linear or nonlinear mixed-effects model for the longitudinal submodel, with a time-to-event or a competing risk model. We considered a natural link setting where the (predicted) longitudinal values are directly related to the survival process. For each joint model, we simulated 200 datasets of 100 patients. We assumed a rich design and parameters were chosen to obtain about 50% of events in single event models, and about 45% for each of the two events in competing risks models. We finally apply the **saemix** extension to fit a multivariate joint model describing biomarker dynamics (neutrophils, C-reactive protein (CRP) and arterial pH) via three linear and nonlinear mixed-effects models, with a competing risk model to describe the risk of in-hospital death and discharge from hospital, in a real case study in patients hospitalised for SARS-COV-2 infection ([Lavalley-Morelle et al., 2022]).

Results : **saemix** can be used to fit many different types of longitudinal data, giving users complete control on the model by defining the log-likelihood function. In the joint model simulation, parameters were precisely and accurately estimated with low bias and uncertainty in all scenarios. For complex joint models (with NLMEM), increasing the number of chains of the algorithm was necessary to reduce bias, but earlier censoring in the competing risk scenario still challenged the estimation. The empirical SE of parameters were very close to those computed with the stochastic algorithm. For more complex joint models (involving NLMEM), some estimates of random effects variances had higher uncertainty and their SE were moderately under-estimated. Finally, type I error was controlled in all joint models.

Conclusion : The **saemix** package, available on CRAN, uses the efficient SAEM algorithm to perform parameter estimation, build models including covariates and interindividual variability, and offers diagnostics to evaluate these models. Through our extension to fit complex parametric joint models, up till now mainly available in specialised pharmacometrics software such as Monolix or NONMEM, **saemix** is the only R package supporting estimation of nonlinear joint models with competing risks.

Références

- E Comets, K Brendel, and F Mentré. Computing normalised prediction distribution errors to evaluate nonlinear mixed-effect models : the npde add-on package for R. *Comput Meth Prog Biomed*, 90 :154–66, 2008. doi : 10.1016/j.cmpb.2007.12.002.
- E Comets, A Lavielle, and M Lavielle. Parameter estimation in nonlinear mixed effect models using saemix, an R implementation of the SAEM algorithm. *J Stat Softw*, 80 :1–41, 2017.
- E Comets, C Rodrigues, V Jullien, and M Ursino. Conditional non-parametric bootstrap for non-linear mixed effect models. *Pharm Res*, 38 :1057–66, 2021.
- M Delattre and E Kuhn. Estimating Fisher information matrix in latent variable models based on the score function. *HAL sciences*, 2023. URL <https://hal.science/hal-02285712>.
- B Karimi, M Lavielle, and E Moulines. f-SAEM : A fast stochastic approximation of the em algorithm for nonlinear mixed effects models. *Comput Stat Data Anal*, 141 :123–38, 2020.
- A Lavalley-Morelle, JF Timsit, F Mentré, J Mullaert, and The OUTCOMEREA Network. Joint modeling under competing risks : Application to survival prediction in patients admitted in intensive care unit for sepsis with daily sequential organ failure assessment score assessments. *CPT :PSP*, 11 :1472–84, 2022. doi : 10.1002/psp4.12856. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/psp4.12856>.
- MA Poursat M Delattre, M Lavielle. A note on BIC in mixed effects models. *Electr J Stat*, 8 :456–75, 2014.

Mise en oeuvre de méthodes semi-Bayésiennes de calcul des erreurs standards pour les données éparses dans le package **saemix**

Mélanie Guhl^{1,*}, Lucie Fayette¹, Julie Bertrand¹, and Emmanuelle Comets^{1,2}

* melanie.guhl@inserm.fr

¹Université Paris Cité, Inserm, IAME, F-75018 Paris, France

²Univ Rennes, Inserm, EHESP, Irset - UMR_S 1085, F-35000 Rennes, France

Résumé

Le package **saemix** implémente l'algorithme fréquentiste Stochastic Approximation Expectation Maximisation (SAEM) qui permet l'estimation de modèles non linéaires à effets mixtes. Ce type de modèle est très utilisé en biostatistiques pour la modélisation de données longitudinales, c'est-à-dire répétées dans le temps, par exemple pour étudier l'évolution clinique d'un patient, d'un biomarqueur ou la dynamique d'une réponse à un traitement. La méthode classique de calcul de l'incertitude autour de l'estimateur du maximum de vraisemblance (EMV) obtenu avec SAEM est basée sur la matrice d'information de Fisher (FIM), dont l'inverse est sa limite asymptotique. Ici, nous proposons une nouvelle approche dite semi-Bayésienne intégrée à l'algorithme SAEM pour échantillonner dans la distribution de l'EMV. Nous explorons deux algorithmes Bayésiens sur plusieurs jeux de simulations et des données issues d'un essai clinique sur des données éparses, c'est-à-dire éloignées de l'asymptotique, pour lesquelles la méthode basée sur la FIM a montré ses limites. Cette approche permet une meilleure couverture des paramètres que la méthode asymptotique, et sur le jeu de données réel, de fortes corrélations entre les paramètres donnent l'avantage à l'algorithme Approximate Bayesian Computation (ABC) n'utilisant pas les vraisemblances.

Mots-clés : Package **saemix** – Modèles non linéaires à effets mixtes – Erreurs standards – Inférence semi-Bayésienne – Données éparses

Introduction

Les modèles non linéaires à effets mixtes (NLMM) sont utilisés pour modéliser des données longitudinales en biostatistiques capturant par exemple, la progression d'une maladie et/ou la réponse à un traitement. Les paramètres sont généralement estimés par le maximum de vraisemblance (EMV) avec l'algorithme Stochastic Approximation Expectation Maximization (SAEM), et leur incertitude via la matrice d'information de Fisher (FIM). Sur des petits échantillons (petit nombre de sujets N) et des designs épars (petit nombre d'observations par sujet n), la FIM peut sous-estimer l'incertitude [Loingeville et al., 2020], et échantillonner dans la distribution de l'EMV grâce à l'inférence Bayésienne a montré de meilleurs résultats dans ce cas, en utilisant l'algorithme Hamiltonian Monte Carlo (HMC) implémenté dans le logiciel *Stan* après estimation avec SAEM [Guhl et al., 2022]. Dans ce travail, nous proposons d'intégrer une méthode semi-Bayésienne dans le package R **saemix** [Comets et al., 2017] qui implémente SAEM.

Méthodes

Nous avons implémenté deux approches semi-Bayésiennes, utilisant l'algorithme de Metropolis-Hastings (SAEM_MH) [Guhl et al., in press] et l'Approximate Bayesian Computation (SAEM_ABC),

et testé différentes variations de ces méthodes. Nous les avons comparées avec des méthodes fréquentistes plus classiques (FIM, sampling importance resampling et bootstrap) et une autre méthode semi-Bayésienne utilisant l'algorithme HMC (plus complexe à réimplémenter) (Post). Nous avons comparé les taux de couverture (c'est-à-dire la proportion de jeux de données pour lesquels l'intervalle de confiance calculé sur un paramètre recouvre sa vraie valeur) et les erreurs standards (SE) relatives obtenues sur une étude de simulation de 500 jeux de données pharmacocinétiques (PK, N=12, n=3) dans deux scénarios, l'un étant plus complexe (fortes variabilités inter-individuelles, corrélations entre les effets aléatoires). Toutes ces méthodes ont été également appliquées sur des données réelles issues d'un essai de PK clinique comparant deux formes galéniques d'un anticorps monoclonal.

Résultats

SAEM_MH a donné de bons résultats sur le premier scénario de simulations mais présenté des limites face à des structures de variabilité complexes, les taux d'acceptation très bas indiquant le manque de fiabilité de la méthode. Utiliser un échantillonnage par blocs et une marche aléatoire a permis d'améliorer les taux d'acceptation, mais les SE et les taux de couverture étaient toujours sous-estimés. Post a également été mis en difficulté sur les corrélations entre effets aléatoires, contrairement à SAEM_ABC qui a présenté de meilleurs résultats et est également efficace en temps de calcul.

Sur les données PK de l'anticorps monoclonal (N=24 sujets par bras, n=11), toutes les méthodes ont donné des résultats concordants, par exemple des RSE entre 5 et 6% pour la clairance du médicament, suggérant que les conditions asymptotiques sont atteintes dans ce cas. Sur un sous-ensemble de ces données (N=6 sujets par bras, n=11), des différences ont été observées entre les méthodes, par exemple des RSE de 1 à 12% pour la clairance, illustrant le besoin d'identifier une méthode de calcul de l'incertitude fiable sur les petits jeux de données.

Conclusion

Les approches semi-Bayésiennes que nous avons implémentées dans l'algorithme SAEM semblent prometteuses pour le calcul de l'incertitude à distance finie. Cependant, l'algorithme MH présente des limites dues à la dimensionnalité du vecteur à échantillonner. Ces difficultés sont partiellement surmontées par des variations de la méthode permettant de diminuer la dimension du vecteur de paramètres à échantillonner et ainsi de remonter les taux d'acceptation. L'algorithme ABC semble le plus prometteur, mais des travaux supplémentaires sont nécessaires pour fournir des recommandations plus complètes sur sa calibration.

Références

- E Comets, A Lavenu, and M Lavielle. Parameter estimation in nonlinear mixed effect models using saemix, an R implementation. *Journal of Statistical Software*, 80 :1–41, 2017.
- M Guhl, F Mercier, C Hofmann, S Sharan, M Donnelly, K Feng, W Sun, G Sun, S Grosser, L Zhao, L Fang, F Mentré, E Comets, and J Bertrand. Impact of model misspecification on model-based tests in PK studies with parallel design : real case and simulation studies. *Journal of Pharmacokinetics and Pharmacodynamics*, 49(5) :557–577, 2022.
- M Guhl, J Bertrand, L Fayette, F Mercier, and E Comets. Uncertainty computation at finite distance in nonlinear mixed effects models - a new method based on metropolis hastings algorithm. *The AAPS Journal*, in press.
- F Loingeville, J Bertrand, TT Nguyen, S Sharan, K Feng, W Sun, J Han, S Grosser, L Zhao, L Fang, K Möllenhoff, H Dette, and F Mentré. New Model-Based Bioequivalence Statistical Approaches for Pharmacokinetic Studies with Sparse Sampling. *The AAPS Journal*, 22(6) :141, 2020.

BeQut, un package R pour l'estimation bayésienne de modèles de régression quantile à effets mixtes via JAGS

Antoine Barbieri* Christophe Tzourio† Hélène Jacqmin-Gadda‡

Résumé

Le package **BeQut** permet l'estimation de modèles de régression quantile tels que la régression linéaire quantile simple ou à effets mixtes, ou encore les modèles conjoints pour données longitudinales et temps jusqu'à événement. Ce dernier combine un modèle de régression quantile à effets mixtes et un modèle à hasard proportionnel afin d'évaluer l'influence de l'évolution d'un quantile sur le risque de survenu d'un événement d'intérêt. Pour l'inférence statistique, il est classique de supposer un cadre paramétrique basé sur une distribution de probabilité. En régression quantile, la distribution naturelle est la distribution asymétrique de Laplace. Le package **BeQut** repose sur une procédure d'estimation bayésienne dont la vraisemblance du modèle découle d'une réécriture de la distribution asymétrique de Laplace. Chacun des modèles de régression quantile est défini comme un modèle hiérarchique bayésien et les échantillons *a posteriori* des paramètres sont obtenus via **JAGS**. La présentation de ce package permettra de : (1) discuter des avantages et inconvénients de la procédure d'estimation proposée, (2) illustrer son intérêt au travers différentes applications sur des données réelles, et (3) faire un retour d'expérience concernant la soumission au CRAN d'un package R utilisant **JAGS**.

Mots-clefs : Régression quantile – Effets mixtes – Modélisation conjointe – JAGS – Package R.

From motivations to BeQut

Le développement du package **BeQut** a été motivé par le fait qu'il n'existe pas de package permettant l'estimation de modèles conjoints pour les données longitudinales et temps jusqu'à événements basé sur la régression quantile. Les modèles conjoints pour les données longitudinales et les temps jusqu'à événements constituent un domaine de recherche très actif en biostatistique. Ces modèles sont nécessaires pour étudier les facteurs de risque variables dépendants du temps en tant que déterminants des événements de santé. Ils évitent les biais des analyses de survie standard telles que le modèle de Cox avec des variables explicatives dépendantes du temps. Les modèles conjoints combinent un modèle mixte pour l'évolution dans le temps du marqueur et un modèle de survie pour le risque d'événement incluant des fonctions des trajectoires des marqueurs comme variables explicatives [Rizopoulos, 2012]. Des packages R sont disponibles pour ajuster ces modèles (**JMbayes2**, **JointMLM**...), mais tous se focalisent sur l'évolution moyenne du marqueur et sur comment celle-ci impacte le risque de développer l'événement. Dans **BeQut**, nous proposons aux utilisateurs une alternative qui s'intéresse à l'évolution de quantile de la distribution du marqueur plutôt qu'à sa moyenne [Yang et al., 2019].

Par ailleurs, le package **BeQut** repose sur une procédure d'estimation bayésienne qui n'est pas utilisée dans les packages R traitant la régression linéaire quantile à effets mixtes (**lqmm** et **qrLMM**). Il permet également de proposer une alternative à ces packages basée sur une autre procédure d'estimation qui (dans certains cas) a montré de meilleurs résultats dans une étude de simulation. Par ailleurs, la procédure d'estimation pour la modélisation conjointe a été validée par simulation.

*Univ. Bordeaux, Inserm U1219, Bordeaux population health center, antoine.barbieri@u-bordeaux.fr

†Univ. Bordeaux, Inserm U1219, Bordeaux population health center, christophe.tzourio@u-bordeaux.fr

‡Univ. Bordeaux, Inserm U1219, Bordeaux population health center, helene.jacqmin-gadda@u-bordeaux.fr

Implémentation

Le package **BeQut** permet l'estimation de modèles de régression quantile tels que la régression linéaire quantile simple ou à effets mixtes, ou encore les modèles conjoints pour données longitudinales et temps jusqu'à événement [Barbieri and Jacqmin-Gadda]. L'estimation des modèles repose sur une procédure d'estimation bayésienne dont la vraisemblance découle de la distribution asymétrique de Laplace. Cette distribution est naturellement considérée pour l'inférence statistique en régression quantile [Koenker and Machado, 1999]. Chacun des modèles implémentés est défini comme un modèle hiérarchique bayésien et les échantillons *a posteriori* des paramètres sont obtenus en appelant le logiciel **JAGS** [Plummer, 2003] depuis R via le package **rjags**. **JAGS** est un programme permettant d'effectuer des simulations MCMC à partir des données et de la déclaration du modèle (définition de la vraisemblance et des distributions *a priori* des paramètres). Bien que l'utilisation de ce logiciel évite la programmation de l'algorithme MCMC pour d'obtenir les échantillons *a posteriori*, il présente tout de même quelques contraintes. Par exemple, les distributions de probabilité ne sont pas toutes définies dans **JAGS**. C'est notamment le cas pour la distribution asymétrique de Laplace ou celles permettant la définition des modèles de survie. Ainsi, une réécriture est nécessaire pour définir les modèles.

Illustrations sur données réelles

L'application du package sera présentée sur deux jeux de données réelles. Le premier jeu de données¹ permettra d'illustrer et de motiver l'utilisation de la régression linéaire quantile via la modélisation de la hauteur de vagues en fonction de la vitesse du vent. Le second, issu des données de l'essai clinique PROGRESS [Mahon et al., 2001], permettra d'illustrer la régression quantile pour la modélisation de la pression artérielle au cours du temps et aussi la modélisation conjointe entre les mesures répétées de la pression artérielle et le risque d'événements cardio- et cérébro-vasculaires.

Remarque : tous les packages R cités dans ce résumé n'apparaissent pas dans les références par souci de place. Cependant, ils sont tous disponibles sur le CRAN où toutes les informations sont disponibles.

Références

- Antoine Barbieri and Hélène Jacqmin-Gadda. *BeQut : Bayesian Estimation for Quantile Regression Mixed Models*. R package version 0.1.0.
- Roger Koenker and José A. F. Machado. Goodness of fit and related inference processes for quantile regression. *Journal of the American Statistical Association*, 94(448) :1296–1310, 1999. doi : 10.1080/01621459.1999.10473882.
- S. Mac Mahon, S. Neal, C Tzourio, A. Rodgers, M. Woodward, J Cutler, C Anderson, and J Chalmers. Randomised trial of a perindopril-based blood-pressure-lowering regimen among 6105 individuals with previous stroke or transient ischaemic attack. *The Lancet*, 358(9287) :1033–1041, 2001.
- M Plummer. Jags : A program for analysis of bayesian graphical models using gibbs sampling. *Proceedings of the 3rd International Workshop on Distributed Statistical Computing*, 2003.
- Dimitris Rizopoulos. *Joint Models for Longitudinal and Time-to-Event Data : With Applications in R*. CRC Press, June 2012.
- Ming Yang, Sheng Luo, and Stacia DeSantis. Bayesian quantile regression joint models : Inference and dynamic predictions. *Statistical Methods in Medical Research*, 28(8) :2524–2537, 2019.

1. données **wave** du package **BeQut** regroupant des données issues de la bouée houle du Cap Ferret (concernant la campagne 03302 Cap Ferret), et des données météorologiques **Infoclimat2** mesurées sur le site de Bordeaux ;

Créez des environnements reproductibles avec rix

Bruno Rodrigues

Résumé

S’assurer que nos analyses soient reproductibles est essentiel, et il existe une multitude d’outils pour les utilisateurs de R que nous sommes pour le permettre. Néanmoins, ces outils ne gèrent qu’un seul aspect du continuum de la reproductibilité : {renv}, par exemple, permet de “figer” les paquets R pour une analyse, mais pas la version de R elle-même. Le “R Installation Manager” permet d’installer n’importe quelle version de R sur n’importe quel système ; Docker permet de containeriser le tout dans une image depuis laquelle on peut exécuter des conteneurs. Nix, développé par Dolstra et al. (2004) est un outil qui permet de gérer chacune de ces dimensions en même temps : il permet de définir un environnement complet comprenant R, les paquets R et les dépendances système sous-jacentes, et de déployer cet environnement de manière totalement reproductible. Malheureusement, Nix peut sembler très compliqué pour des novices, c’est pourquoi j’ai développé le paquet {rix}, qui permet de définir des environnements de développement pour R de manière très simple. Dans cette présentation, j’expliquerai comment Nix assure la reproductibilité d’une analyse et comment on peut l’utiliser simplement grâce à {rix}.

Mots-clefs : Package - Reproductibilité

Développement

{rix} est un paquet R qui tire parti de Nix, un puissant gestionnaire de paquets axé sur la reproductibilité. Avec Nix, il est possible de créer des environnements spécifiques à un projet contenant une version spécifique de R et des paquets R (ainsi que d’autres outils ou langages, si nécessaire). Vous pouvez utiliser {rix} et Nix pour remplacer {renv} et Docker par un seul outil. Nix est un logiciel incroyablement utile pour garantir la reproductibilité des projets. Par exemple, il permet d’exécuter des applications web telles que des applications Shiny ou des API plumber dans un environnement contrôlé.

Nix a un coût d’entrée assez élevé cependant. Nix est un logiciel complexe qui dispose de son propre langage de programmation, également appelé Nix. Son objectif est de résoudre

un problème complexe : définir des instructions sur la manière de construire des logiciels et de gérer les configurations de manière déclarative. Cela garantit que le logiciel est installé de manière entièrement reproductible, sur n'importe quel système d'exploitation ou matériel.

{rix} fournit des fonctions pour vous aider à écrire et déployer des expressions écrites dans le langage Nix. Ces expressions seront les entrées du gestionnaire de paquets Nix, pour construire des ensembles de paquets logiciels et les fournir dans un environnement de développement reproductible et cohérent. Ces environnements peuvent être utilisés pour l'analyse de données interactive, ou reproduits lors de l'exécution de pipelines dans des systèmes CI/CD. Dans la collection “nixpkgs” (l'équivalent du CRAN pour Nix), il y a actuellement plus de 80 000 logiciels disponibles via le gestionnaire de paquets Nix. Avec {rix}, vous pouvez définir et construire des environnements R isolés via le gestionnaire de paquets Nix avec facilité. Ainsi, les environnements contiennent R et tous les paquets requis dont vous avez besoin pour votre projet. Vous pouvez également ajouter n'importe quel autre logiciel nécessaire à votre analyse. L'écosystème R de Nix comprend actuellement la quasi-totalité des paquets CRAN et Bioconductor. Il est également possible d'installer des versions antérieures des paquets R, ou d'installer des paquets depuis GitHub à des commits définis.

Le gestionnaire de paquets Nix est extrêmement puissant. Non seulement il gère très bien toutes les dépendances de n'importe quel paquet de manière déterministe, mais il est également possible avec lui de reproduire des environnements contenant des versions anciennes de logiciels. Il est ainsi possible de construire des environnements contenant la version 4.0.0 de R (par exemple) pour exécuter un ancien projet qui a été développé à l'origine sur cette version de R.

Si vous avez besoin d'autres outils ou langages comme Python ou Julia, cela peut également être fait facilement. Nix est disponible pour Linux, macOS et Windows (via WSL2) et {rix} présente les fonctionnalités suivantes :

- permet d'installer n'importe quelle version de R (depuis la version 3.0.2) et des paquets R pour des projets spécifiques ;
- avoir plusieurs versions de R et des paquets R installées en même temps sur le même système ;
- définir des environnements de développement complets en code et les utiliser n'importe où ;
- exécuter des fonctions R individuelles dans un environnement différent (éventuellement avec une version différente de R et des paquets R) depuis une session R interactive, et récupérer la sortie de cette fonction en utilisant `with_nix()`;

{rix} ne nécessite pas que Nix soit installé sur votre système pour générer des expressions. Cela signifie que vous pouvez générer des expressions sur un système sur lequel vous ne pouvez pas facilement installer de logiciel, puis utiliser ces expressions sur le cloud ou dans un environnement CI/CD pour y construire le projet.

Pour définir un environnement il suffit d'utiliser la fonction `rix()`:

```
rix(r_ver = "4.3.1",
    r_pkgs = c("dplyr", "chronicler"),
    ide = "other")
```

Ceci va générer un fichier appelé `default.nix` qui sera ensuite utilisé par le gestionnaire de paquets Nix pour installer R version 4.3.1 ainsi que `{dplyr}` et `{chronicler}` (tels qu'ils étaient à la sortie de cette version de R). Toutes les autres dépendances systèmes, telles que des librairies dynamiques ou compilateurs nécessaires pour générer cette environnement seront aussi installés.

Références

Dolstra, Eelco, Merijn De Jonge, Eelco Visser, et al. 2004. “Nix: A Safe and Policy-Free System for Software Deployment.” In *LISA*, 4:79–92.

Une enquête auprès des métiers de la « data » : quelle place pour R et ses utilisateurs ?

Antoine GIRARD, *data analyst* indépendant

Antoine.girard@datag.fr

Quelle place pour R en 2024, dans le large univers de la « data » ? Peut-on dire que son usage soit en progression ou en déclin, parmi tous ces professionnels de la donnée ? Quelles sont les spécificités des utilisateurs de R (secteurs d'activité, missions, caractéristiques socio-démographiques) par rapport aux autres outils d'exploitation des données ? Une enquête menée en 2024 – et deuxième édition après une première réalisée en 2021 – tente d'apporter quelques éclairages quantitatifs à ces questions.

Mots-clefs (3 à 5) : Data enquête outils

Développement

Dans le large univers de la « data », les outils et missions évoluent vite : les *data analyst*, *data scientist* ou *data engineer* ont à leur disposition une large gamme d'outils, d'Excel à Python en passant par SAS et Power BI. Il n'est toutefois pas évident de savoir dans quelle mesure chacun de ces outils sont utilisés, et par qui. Des chiffres sont parfois partagés par des plateformes sur les utilisateurs (comme Kaggle ou Pylote), mais ces plateformes ont aussi de très fort biais liés à leur positionnement. Même remarque pour les classements de popularité des langages de programmation, qui concernent l'informatique au sens large mais pas forcément le créneau plus spécifique des analystes, scientifiques et ingénieurs de la donnée.

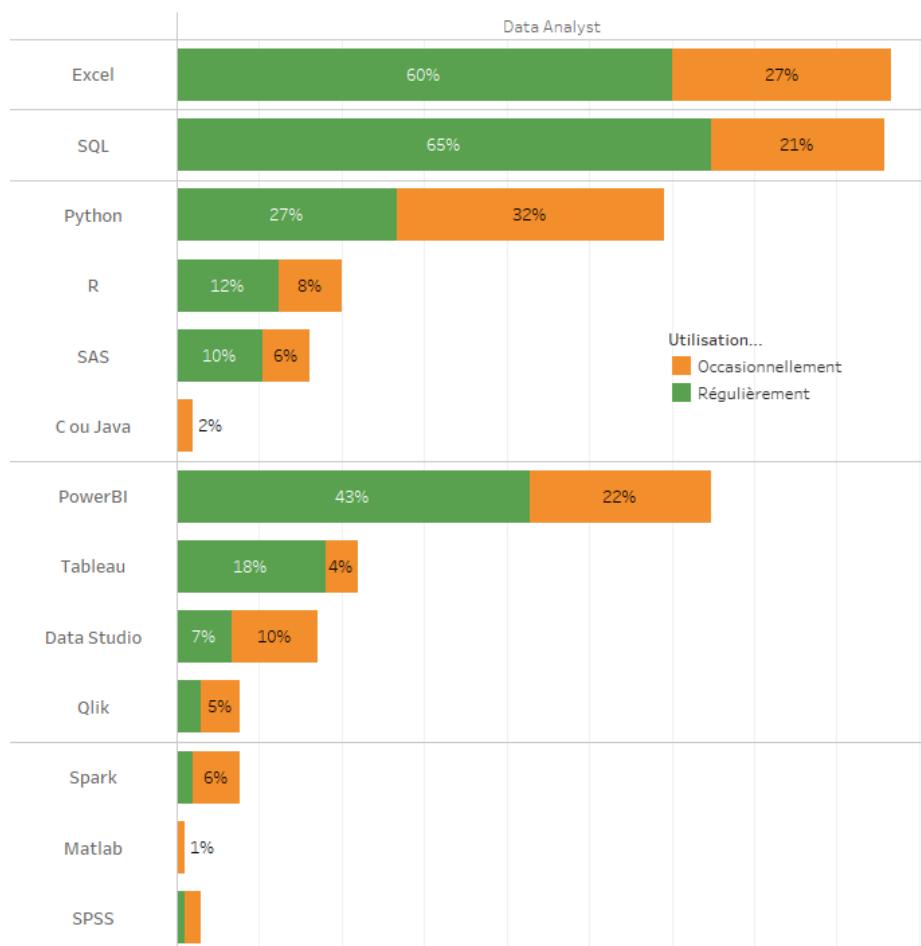
C'est dans ce but qu'une enquête quantitative a été menée par ma modeste personne. Diffusée principalement via le réseau social professionnel *Linked* en Mars 2024 (diffusion du questionnaire auprès de professionnels de la data), elle se base sur plus de 300 réponses de profils *data* en tous genres. Cet échantillon ne peut pas être considéré comme parfaitement représentatif, dans la mesure où les répondants ont été contactés de façon « pseudo aléatoire » via la plateforme *Linked*. Hors, rien ne prouve que *Linked* soit représentatif de l'ensemble de la population visée. Tout en étant conscient de ce biais inhérent à la méthodologie, cette enquête a été menée dans le souci d'une représentation la plus fidèle possible de la réalité de ces métiers : ainsi des profils choisis « aléatoirement » sur la plateforme ont été contactés, et non pas uniquement via mon réseau personnel, forcément biaisé.

Les thématiques abordées par le questionnaire se limitent à trois grands domaines : les outils (connaissance et expertise), les missions (effectives et préférées) et le profil général (secteur d'activité, intitulé du poste, âge et genre). La volonté d'avoir un questionnaire court et simple (mais avec un bon taux de retours) explique ce nombre limité de thématiques.

Dans le cadre de cette présentation, les résultats seront évidemment orientés autour des utilisateurs du logiciel R : quel « poids » représentent-ils, notamment par rapport aux autres langages de programmation que sont SAS et Python ? Avec quels autres outils est-il le plus souvent utilisé ? Quelle évolution constate-t-on par rapport à la précédente enquête menée en 2021 ? Quelles sont les particularités de leurs profils, que ce soit sur leur mission ou leurs caractéristiques
*antoine.girard@datag.fr

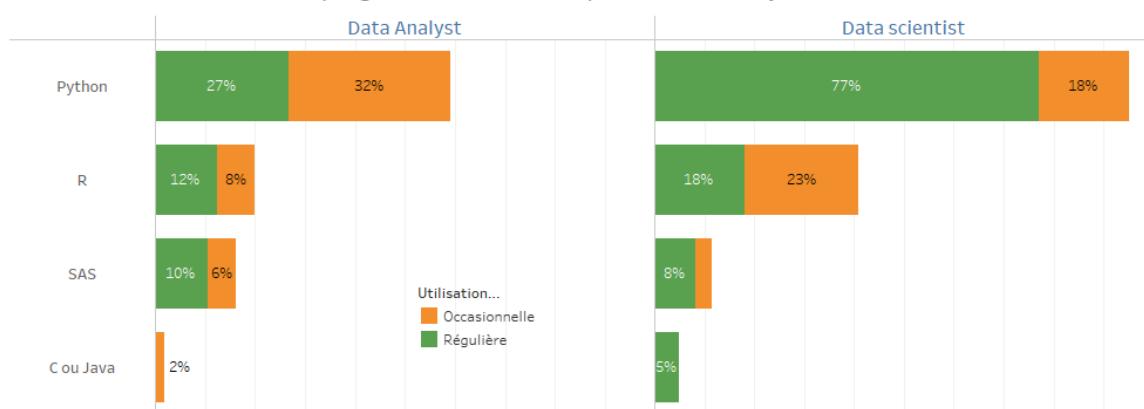
démographiques ? Autant de questions auxquelles cette enquête tâche d'apporter quelques réponses.

Les outils des *data analyst*



Enquête menée par Antoine Girard en Février 2024 auprès de 300 professionnels de la data

Outils de programmation utilisés par les *data analyst et scientist*



Enquête menée par Antoine Girard en Février 2024 auprès de 300 professionnels de la data

Références

Document de synthèse sur les principaux résultats de la précédente édition de l'enquête :
<https://public.tableau.com/app/profile/antoine.girard/viz/Synthseenquitedata2022/Synthseenquitedata2022>

Technologies utilisées par les freelances en informatique sur la plateforme *Pylote* :
<https://pylote.io/comparatif-competences-freelances-tech>

Les langages de programmation les plus populaires en 2023 selon *l'IEEE Computer society* :
<https://spectrum.ieee.org/the-top-programming-languages-2023>

Pour un namespace tout en souplesse

Swann Floc'hlay*

Résumé

Si l'on vous demande “*Comment est votre namespace?*”, que répondez-vous ?

- “*vide, je ne programme qu'en Base-R*”
- “*automatique, c'est {roxygen2} qui s'en charge*”
- “*nébuleux, j'en suis à ma 30ème dépendance*”
- “*mon namespace ?*”

Quelle que soit votre réponse, ajouter des dépendances à son package est un passage quasi obligé lorsqu'on développe. Cela permet de ne pas avoir à réinventer la roue et de bénéficier des projets open source disponibles autour de soi. Pour garder le fil, il est possible de lister ces dépendances dans un fichier **NAMESPACE** à la racine de son package.

Cependant, toutes les dépendances n'ont pas forcément le même public. Par exemple, les fonctions de **{testthat}** seront utiles aux développeurs, mais pas aux utilisateurs. Les dépendances n'auront aussi pas la même importance si elles sont systématiquement appelées, ou propres à une petite gamme de paramètres. Chaque dépendance vient aussi avec son lot plus ou moins conséquent de sous-dépendances, et ses évolutions aux fils des montées de version.

Avec tout ça en tête, comment bien ficeler son **NAMESPACE**, en s'assurant de ne rien oublier, sans trop en rajouter ?

Mots-clefs : Package - Reproductibilité - Integration Continue - Dépendances

Développement

Dans cette présentation, je vous propose de passer un **NAMESPACE** à la loupe grâce à l'intégration continue (CI), puis de réorganiser les dépendances à travers une catégorisation entre **Suggests** et **Imports**.

Le fichier **NAMESPACE** contient à la fois les fonctions à importer depuis les dépendances, et les fonctions du package à exporter dans l'environnement de l'utilisateur. Ce fichier peut être rempli manuellement, ou à l'aide de **{roxygen2}** et des balises de documentation (*e.g.* **@importFrom**, **@export**). L'absence d'une référence dans ce fichier peut causer une erreur du type **could not find function "x"**.

Le R **CMD CHECK** est un outil utile pour corriger ces erreurs. Il s'assure que toutes les dépendances sont installées et que les exemples se lancent correctement. Il y a toutefois quelques trous dans la raquette, car il ne vérifie pas si nos fonctions débordent hors du namespace, et utilisent des packages non référencés. À moins de désinstaller un à un nos packages, difficile à dire.

C'est ici que le CI entre en jeu ! L'utilisation de l'intégration continue permet de valider les besoins en packages externes à partir d'une coquille basique de R, sans aucune pré-installation. On peut y détecter les dépendances non référencées manquantes, qui cette fois seront inaccessibles lors de l'appel au R **CMD CHECK**.

*ThinkR, swann@thinkr.fr

Une fois notre liste de dépendance complétée, il faut être vigilant à ne pas tomber dans l'excès de zèle et référencer tout ça dans le même panier. La catégorisation des dépendances entre `Imports` et `Suggests` hiérarchise les besoins pour passer sous la limite de dépendances surnuméraires du R CMD CHECK. Cette refonte finale sera balisée par des appels à `requireNamespace()`, pour garantir une réutilisation du package sans retour de bâton.

ProteoBayes : un cadre bayésien pour l'analyse protéomique différentielle

Marie Chion* Arthur Leroy†

Résumé

Les méthodes statistiques actuelles dans l'analyse protéomique différentielle laissent généralement de côté plusieurs défis, tels que les valeurs manquantes, les corrélations entre les intensités des peptides et la quantification de l'incertitude. En outre, elles fournissent des estimations ponctuelles, telles que l'intensité moyenne pour un peptide ou une protéine donné(e) dans une condition donnée. La décision de considérer ou non un analyte comme différentiel est alors basée sur la comparaison d'une p-valeur avec un seuil de significativité. Nous présentons ici le package R ProteoBayes, disponible sur le CRAN. Il implémente un cadre bayésien pour l'analyse protéomique différentielle, permettant ainsi d'estimer explicitement la taille de l'effet et de quantifier l'incertitude pour une différence de moyennes entre deux conditions biologiques comparées. Une application Shiny a également été mise en place pour les utilisateurs ne codant pas en R.

Mots-cléfs : Biostatistique – Statistique Bayésienne – Package – Shiny

Développement

L'analyse protéomique différentielle consiste à mesurer des intensités de peptides (usuellement par spectrométrie de masse), puis à comparer leurs moyennes par condition pour enfin identifier celles qui sont différentiellement exprimées. Celles-ci sont alors considérées comme potentiels biomarqueurs de pathologie. Les méthodes statistiques couramment utilisées ignorent généralement plusieurs problématiques spécifiques à ces données, telles que les valeurs manquantes, les corrélations entre les intensités des peptides et la quantification de l'incertitude. En outre, elles fournissent des estimations ponctuelles, telles que l'intensité moyenne pour une protéine donnée dans une condition donnée. La décision de considérer ou non une protéine comme "différentielle" est alors basée sur la comparaison de la p-valeur avec un seuil de significativité, généralement de 5 %. Dans l'approche du test t-modéré de Smyth [2004], ainsi que dans son extension dans le cadre d'imputation multiple [Chion et al., 2022], un modèle hiérarchique bayésien est utilisé pour déduire la distribution *a posteriori* de l'estimateur de la variance pour chaque peptide. L'espérance de cette distribution est ensuite utilisée comme une estimation modérée de la variance et est injectée directement dans l'expression de la statistique de test. Cette méthode permet de prendre en compte la structure de variabilité particulière des données de protéomique et plus largement des données d'expression de gènes. Cependant, en considérant des distributions plutôt que des estimateurs ponctuels de position et de dispersion, la statistique bayésienne permet de quantifier l'incertitude. Le travail présenté dans Chion and Leroy [2023] suit une idée similaire en tirant parti des résultats standard de l'inférence bayésienne avec des distributions *a priori* conjuguées dans les modèles hiérarchiques pour développer une méthodologie adaptée au traitement des contextes d'imputation multiple.

Formellement, si l'on cherche à comparer l'intensité moyenne d'un peptide $p = 1, \dots, P$ entre différents groupes $k = 1, \dots, K$, pour lesquels nous avons observé $n = 1, \dots, N_k$ échantillons, l'inférence porte alors sur la quantité $\mu_k^p - \mu_{k'}^p$ (la différence des moyennes de groupes). Si l'on note $y_{k,n}^p$ l'observation

*MRC Biostatistics Unit, University of Cambridge, marie.chion@mrc-bsu.cam.ac.uk

†Department of Computer Science, The University of Manchester, arthur.leroy.pro@gmail.com

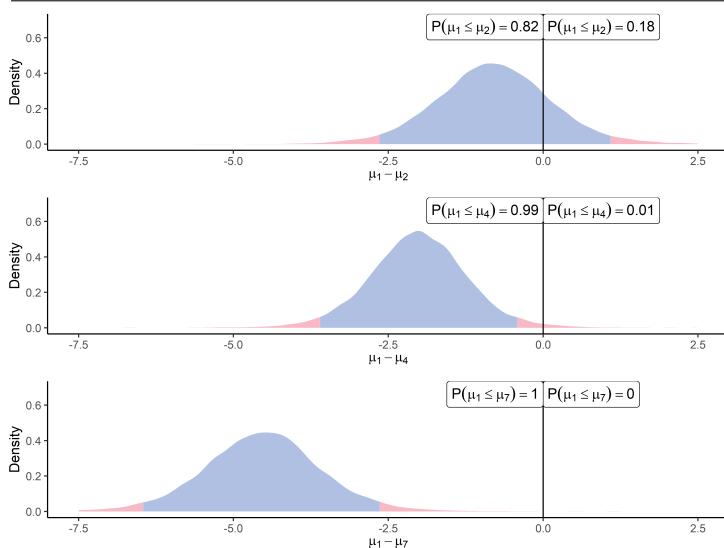


FIGURE 1 – Illustration de la loi a posteriori des moyennes de groupes pour un même peptide à travers 3 comparaisons distinctes (groupe 1 vs 2, 1 vs 4, 1 vs 7) présentant un écart croissant. La probabilité que la moyenne d'un groupe soit plus grande que l'autre est indiquée de part et d'autre de la droite d'abscisse 0. L'intervalle de crédibilité à 95% est représenté par l'aire en bleue sur la densité.

n , du groupe k , pour le peptide p , le modèle génératif est défini comme :

$$y_{k,n}^p = \mu_k^p + \varepsilon_n, \quad \forall p = 1, \dots, P, \quad \forall k = 1, \dots, K, \quad \forall n = 1, \dots, N_k,$$

avec la vraisemblance et les lois à priori suivantes : $\varepsilon \sim \mathcal{N}(0, \sigma_k^{p2})$, $\mu_k^p \mid \sigma_k^{p2} \sim \mathcal{N}\left(\mu_0, \frac{1}{\lambda_0} \sigma_k^{p2}\right)$, $\sigma_k^{p2} \sim \Gamma^{-1}(\alpha_0, \beta_0)$, où $\{\mu_0, \lambda_0, \alpha_0, \beta_0\}$ sont les hyper-paramètres associés. Ces hypothèses décrivent une loi a priori Gaussienne-inverse-gamma pour les paramètres de moyenne μ_k^p et de variance σ_k^{p2} , qui est conjuguée à la vraisemblance Gaussienne. Puisque l'objet d'intérêt pour l'inférence est le paramètre de moyenne μ_k^p pour chaque groupe, la loi a posteriori marginale $p(\mu_k^p \mid \mathbf{y}_k^p)$ résulte en une t -distribution généralisée dont l'expression analytique est connue. En échantillonnant pour chaque groupe à partir de ces distributions, il est ainsi possible de calculer la loi a posteriori de notre quantité d'intérêt $\mu_k^p - \mu_{k'}^p$, qui peut être visualisée comme sur la Figure 1, où 3 exemples de différence (plus ou moins importante) entre groupes sont représentés. Cette loi offre un bien plus large panel d'informations qu'un résultat de t -test pour conduire l'inférence. En effet, nous estimons ici explicitement la taille d'effet, ainsi que l'incertitude associée, de la différence entre groupes. Il est alors trivial de définir une procédure de décision probabiliste, se basant sur la probabilité que cette différence soit suffisamment distincte de 0.

Cette approche globale a été implémentée et rendue librement disponible à travers un package R, ProteoBayes, disponible sur le CRAN, et une application web offrant une interface graphique pour les praticiens ne codant pas en R.

Références

- Marie Chion and Arthur Leroy. A Bayesian Framework for Multivariate Differential Analysis accounting for Missing Data, July 2023.
- Marie Chion, Christine Carapito, and Frédéric Bertrand. Accounting for multiple imputation-induced variability for differential analysis in mass spectrometry-based label-free quantitative proteomics. *PLOS Computational Biology*, 18(8) :e1010420, August 2022. ISSN 1553-7358. doi : 10.1371/journal.pcbi.1010420.
- Gordon K Smyth. Linear Models and Empirical Bayes Methods for Assessing Differential Expression in Microarray Experiments. *Statistical Applications in Genetics and Molecular Biology*, 3(1) :1–25, January 2004. ISSN 1544-6115. doi : 10.2202/1544-6115.1027.

Cadre R chez IMPACT Initiatives

Yann Say*

Résumé (max 300 mots)

IMPACT Initiatives est une O.N.G. qui travaille dans plus de trente pays dans le domaine de la recherche appliquée au secteur humanitaire et de développement. Afin d'améliorer l'efficacité et la transparence du nettoyage et analyse des données, un cadre d'analyse avec R a été créé. Ce cadre permet une harmonisation des processus d'analyse et de garder une modularité et flexibilité qui permet d'adapter les processus à chaque contexte et à chaque équipe de travail. Ce cadre est composé de 4 étapes (nettoyage, composition, analyses et présentation), 4 verbes (vérifier, ajouter, créer et revoir) et 2 adjectifs (indépendant et "pipe-able"). Quatre packages R ont été créés pour accompagner le cadre et sont dans une version beta. Le déploiement du cadre et des outils est en cours au sein de l'organisation.

Mots-clefs (3 à 5) : Data - Humanitaire - Process

Développement

IMPACT Initiatives est une O.N.G. qui travaille dans plus de trente pays dans le domaine de la recherche appliquée au secteur humanitaire et de développement. Les projets de recherche sont mis en oeuvre principalement par les équipes basées sur le terrain, de l'élaboration des questions de recherche à la dissémination des résultats. Les équipes basées à Genève ont pour mission de fournir un soutien et des conseils (y compris la revue et la validation de tous les résultats pertinents) aux équipes pays sur tous les aspects de la mise en œuvre du cycle de recherche. En 2021, 80% du nettoyage et 50% des analyses utilisaient des scripts personnels aux analystes. Afin d'améliorer l'efficacité et la transparence du nettoyage et de l'analyse des données, un cadre d'analyse avec R a été créé. Le déploiement de ce cadre et ses outils a commencé au début de l'année 2024.

Le cadre d'analyse R chez IMPACT est un cadre modulaire à deux dimensions :

- une dimension horizontale qui se concentre sur le résultat d'une étape, et
- une dimension verticale qui se concentre sur le contenu d'une étape.

Le cadre est construit autour de :

- 4 étapes
 - Nettoyage : Toute manipulation pour passer des données brutes aux données propres
 - Composition : Toute manipulation avant l'analyse, par ex. ajout d'indicateurs, combiner deux bases de données (avant et après une distribution). Les indicateurs sont créés avant l'analyse, et non avec l'analyse.
 - Analyse: Toute manipulation concernant uniquement l'analyse. L'analyse s'arrête à la table des résultats. La table de résultat est format long : statistique + clé d'analyse.
 - Présentation : Toute manipulation pour formater la table des résultats.

- 4 verbes :

*IMPACT Initiatives, yann.say@impact-initiatives.org

-
- Vérifier (check_*) : Une fonction qui va signaler certaines valeurs (par exemple, valeurs autres à recoder, valeurs aberrantes, etc.). Ces valeurs sont cataloguées dans un journal. Un check_* renverra une liste : le jeu de données vérifié et le journal de nettoyage.
 - Ajouter (add_*): Une fonction qui ajoute une variable (colonne) au jeu de données. Par exemple, pour ajouter la durée d'une enquête, pour ajouter la catégorie de score de consommation alimentaire, etc.
 - Créer (create_*): Une fonction qui créé, transforme quelque chose, par exemple créer un journal de nettoyage avec des actions à faire, créer une table de résultats d'analyse, créer un produit pour une présentation.
 - Revoir (review_*): Une fonction qui examinera un objet en le comparant à des normes ou à un autre objet et signalera les différences, par exemple en examinant le nettoyage en comparant le jeu de données brut, le jeu de données propre et le journal de nettoyage, l'analyse le comparant à une autre analyse.

- 2 adjectifs :

- « pipeable » : Chaque étape, chaque famille de fonctions (verbes) peuvent être utilisée une après l'autre. Cela permet à l'utilisateur.ice de pouvoir adapter chaque étape avec les fonctions spécifiques à son contexte.
- Indépendant : A chaque étape, le cadre définit l'entrée et la sortie, le “comment” est libre, chaque utilisateur.ice peut donc choisir l'outil qu'il/elle préfère.

La présentation montrera la structure et l'articulation du cadre d'analyse grâce à quatre packages spécifiques : {cleaningtools}, {addindicators}, {analysistools} et {presentresults}. Elle présentera, à travers différentes fonctions, comment les verbes et adjectifs sont intégrés dans chacun des packages ainsi que les définitions de format à chaque étape. Un exemple pratique illustrera l'utilisation de ce cadre, en se concentrant sur les aspects suivants :

- L'étape de nettoyage présentera comment créer un journal de nettoyage avec différentes valeurs à vérifier, comment l'exporter, et créer une base de données nettoyer.
- L'étape de composition montrera comment ajouter différents indicateurs.
- L'étape d'analyse présentera comment créer une table de résultats d'analyses descriptives.
- L'étape de présentation montrera comment passer d'une table de résultats à un produit à partager.

Les packages sont disponibles sur GitHub et peut être installé dans R via la commande :

```
devtools::install_github("impact-initiatives/cleaningtools")
devtools::install_github("impact-initiatives/addindicators")
devtools::install_github("impact-initiatives/analysistools")
devtools::install_github("impact-initiatives/presentresults")
```

Le cadre d'analyse R chez IMPACT Initiatives représente une étape significative dans l'efficacité et la transparence du nettoyage et de l'analyse des données au sein de l'organisation. Ce cadre offre une approche modulaire et flexible qui permet de s'adapter aux différents contextes et analyses. Bien qu'il soit encore trop tôt pour évaluer de ce cadre, ces premiers résultats prometteurs soulignent le potentiel de ce cadre pour renforcer les pratiques de recherche et d'analyse au sein de l'organisation, en favorisant la collaboration entre les équipes.

Comment les communautés autour de R peuvent changer vos projets

Marie VAUGOYEAU 1*

Résumé (max 300 mots)

Vous **codez** tous les jours (ou pas) **en R sans faire un seul passage sur GRRR ou tout autre forum, ni aller à aucune rencontre ou congrès**, et vous êtes sûr.e.s de ne rien **louper** ?

Venez je vais vous présenter les différentes communautés qui gravitent autour de R et l'intérêt dans la réalisation de projets R.

En particulier, je détaillerai les communautés dans lesquels j'ai des interactions et comment elles ont pu m'aider dans différents projets liés à R. Plus particulièrement je mettrai l'accent sur les relations humaines.

Mon but est de vous montrer comment les **communautés autour de R** ont changé mon **parcours professionnel**, en m'amenant à :

- écrire un livre sur R
- réaliser des cours en ligne sur R
- présenter des packages R et des analyses stats en direct sur Twitch

Dans cette présentation, je rendrais hommage aux personnes qui m'ont accompagnées lors de projets R et m'ont permis de **lancer une idée**, de **réparer une erreur** ou de **trouver LA solution** qu'il fallait !

Et si vous êtes **convaincu.e** que c'est important mais que vous **n'osez pas** ou que vous **ne savez pas comment faire**, je vais vous donner les trucs qui ont fonctionné pour moi, en **décortiquant comment les relations humaines qui m'ont aidée se sont mises en place**.

Après ma présentation, vous connaîtrez des communautés liées à R et **comment les rejoindre**.

Mots-clefs (3 à 5) : Enseignement - Retours d'expérience - Projet

Développement

Lors de cette présentation, je vous présenterais des communautés liées à R :

- Le groupe slack de GRRR, un forum francophone sur R
- Les R-Ladies : Une organisation mondiale féministe de R, incluant des événements en personne et en ligne ainsi que des forums de discussion

Mais aussi comment échanger avec des personnes :

- En présentiel lors de congrès comme les Rencontres R ou UseR!
- En présentiel lors de congrès non spécifique à R
- En distanciel via LinkedIn, le groupe slack GRRR, twitter...

Lors de cette présentation, je reviendrai sur différents projets R qui ont marqué ma vie professionnelle pour mettre en lumière l'intervention d'autres personnes que cela soit :

*MStats (microentreprise), marie.vaugoyeau@gmail.com

-
- Une recommandation qui permet d'initier un nouveau projet comme un livre ou un cours sur une plateforme. Un projet qu'on aurait pas nécessairement trouvé ou même cherché seul.e.
 - Une aide pour une erreur bloquante, en croyant qu'on est responsable alors que c'est un bug d'un pakage. Il est toujours plus facile de s'accabler en disant qu'on code mal plutôt que d'oser déranger les personnes qui conçoivent les outils.
 - Un conseil pour améliorer ou même sauver un projet quand on ne part pas dans la bonne direction ou qu'on est bloqué par une techno.

Tous cela grâce aux relations humaines.

J'aborderais aussi la façon dont ces relations au sein des communautés liées à R se sont **construites, nourries et développées**.

Je vous partagerais quelques idées pour **développer** et **entretenir les liens** à distance mais aussi lors d'évènements.

J'espère que cette présentation vous permettra de construire un **réseau d'humains qui partage la même passion que vous** et qui **dynamisera vos projets R** !

Vous connaîtrez des communautés R accueillantes et la manière de les rejoindre sans attendre.

Liens

- Groupe slack de GRRR : https://r-grrr.slack.com/join/shared_invite/zt-46utbgb9-uv0_bg5cbuxOV~H10YUX8w#/shared-invite/email
- R-Ladies : <https://rladies.org/>
- Rencontres R : <https://rr2024.sciencesconf.org/>
- UseR! : <https://events.linuxfoundation.org/user/>

hubeau: un package pour interroger les APIs du Système d'Information sur l'eau en France

David Dorchies 1* Pascal Irz 2†

Résumé

Résumé

Hub'Eau (<https://hubeau.eaufrance.fr/>) est la plateforme nationale de diffusion des données ouvertes de 12 bases de données nationales sur l'eau en France mise en place en 2018 grâce à une collaboration entre l'Office français de la biodiversité (OFB) et le BRGM. Elle met à disposition des API (Application Programming Interface) web s'appuyant sur une infrastructure et des méthodes adaptées au traitement et au stockage de données massives garantissent de hautes performances en termes de rapidité et de disponibilité (plus de 20 requêtes par seconde). L'utilisateur peut ainsi formuler une requête sur un ou plusieurs paramètres et utiliser n'importe quel outil utilisant un langage de programmation pour collecter les résultats issus de données nationales régulièrement enrichies et mises à jour quotidiennement voire en temps réel pour l'API hydrométrie.

L'utilisation de telles API pour se révéler ardue et le package R hubeau (Dorchies 2022) (<https://cran.r-project.org/package=hubeau>) implémente des fonctions qui permettent d'interroger facilement 10 des 12 API mises à disposition sur la plateforme. Le package est fourni avec une documentation complète ainsi que des vignettes proposant des exemples concrets de cartographie ou de statistiques sur l'observation des étiages, le suivi du niveau des nappes ou encore la qualité de l'eau des rivières.

Le package hubeau a notamment été adopté par les services de l'OFB pour produire des rapports régionalisés reproductibles au format Rmarkdown valorisant les données de l'Observatoire national des étiages Onde (observation de l'écoulement des petits cours d'eau en période d'étiage) (https://github.com/ofb-bzh/PRR_ONDE/).

Mots-clefs : Eau - Data - Package - Reproductibilité

*G-EAU, Univ Montpellier, AgroParisTech, BRGM, CIRAD, IRD, INRAE, Institut Agro, Montpellier, France, david.dorchies@inrae.fr

†OFB, Direction Régionale Bretagne, Cesson-Sévigné, France, pascal.irz@ofb.gouv.fr

Les 10 API actuellement implémentées dans le package sont:

- “Écoulement des cours d'eau” issue des données de l'Observatoire National Des Étiages (Onde)
- “Hydrométrie” issue de la plateforme HYDRO Centrale opérée par le Service Central d'Hydrométéorologie et d'Appui à la Prévision des Inondations (SCHAPI)
- “Indicateur des services” issue des données de l'Observatoire des services d'eau et d'assainissement
- “Piézométrie” et “Qualité des nappes d'eau souterraine” issues du portail national d'Accès aux Données sur les Eaux Souterraines (ADES)
- “Poisson” issue des données collectées lors d'opérations de pêches scientifiques réalisées par l'Office français de la biodiversité (OFB) et ses partenaires
- “Prélèvements en eau” issue de la Banque Nationale des Prélèvements quantitatifs en Eau (BNPE)
- “Qualité de l'eau potable” issue des contrôles sanitaire de l'eau distribuée à l'échelle communale gérés par le Ministère des Solidarités et de la Santé
- “Qualité des cours d'eau” et “Température des cours d'eau” issues de la base de données Naiades regroupant les données sur la qualité des eaux de surface

Le package fournit la liste des API, la liste des points de terminaison et la liste des champs utilisables pour filtrer la requêtes via les fonctions `list_apis`, `list_endpoints` et `list_params`.

La syntaxe d'interrogation employée dans le package suit une logique identique quelque soit l'API, ses points de terminaisons (endpoints), et les paramètres de filtrage de la requête :

```
get_[API]_[endpoint]([param1] = [Valeurs1], [param2] = [Valeurs2], ...)
```

Le package gère automatiquement la pagination des requêtes et retourne les données sous la forme d'une `tibble` directement utilisable pour l'analyse et la visualisation.

Par exemple, lister les stations hydrométriques sur la Seine en Île de France revient à :

```
library(hubeau)

## Warning: le package 'hubeau' a été compilé avec la version R 4.4.0

get_hydrometrie_sites(code_region = "11", libelle_site = "Seine",
                       unique_site = TRUE, fields = c("code_site", "libelle_site"))

## # A tibble: 16 x 2
##   code_site libelle_site
##   <chr>      <chr>
## 1 F2210002 La Seine à Bray-sur-Seine
## 2 F2400001 La Seine à Bazoches-lès-Bray
## 3 F4000001 La Seine à Montereau-Fault-Yonne
## 4 F4000002 La Seine à Varennes-sur-Seine
## 5 F4000003 La Seine à Saint-Mammès
## 6 F4470001 La Seine à Melun
## 7 F4470002 La Seine à Boissise-la-Bertrand
## 8 F4470003 La Seine à Saint-Fargeau-Ponthierry
## 9 F4490006 La Seine à Corbeil-Essonnes
## 10 F4900001 La Seine à Alfortville et à Villeneuve-Saint-Georges et à Vitry-su-
## 11 F7000001 La Seine à Paris
## 12 F7040001 La Seine à Suresnes
## 13 F7120001 La Seine à Bougival et à Chatou
## 14 H3000002 La Seine à Poissy
## 15 H3080001 La Seine à Limay [Mantes]
## 16 H3080002 La Seine à Méricourt
```

Et tracer tous les débits moyens mensuels de la Seine à Paris revient à :

```

start_time <- Sys.time()
df <- get_hydrometrie_obs_elab(code_entite = "F7000001", grandeur_hydro_elab = "QmM")
Sys.time() - start_time

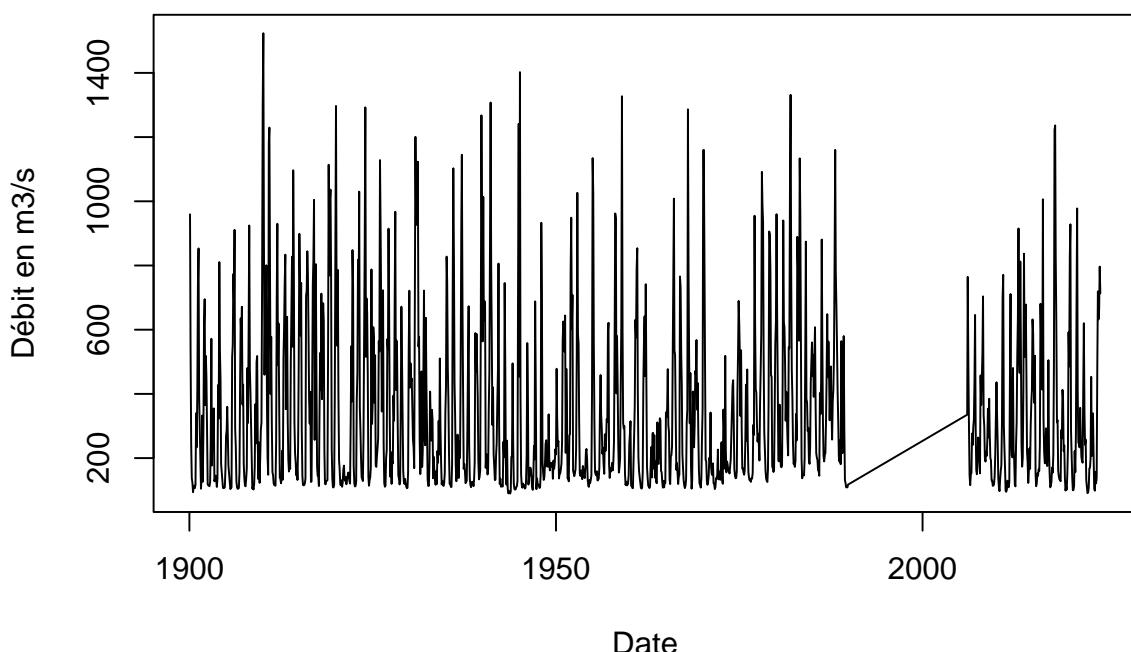
## Time difference of 12.61204 secs

df$date_obs_elab <- as.Date(df$date_obs_elab)
df <- df[order(df$date_obs_elab), ]

plot(df$date_obs_elab, df$resultat_obs_elab / 1000, type = "l",
      xlab = "Date", ylab = "Débit en m3/s",
      main = "Débit moyen mensuel de la Seine à Paris")

```

Débit moyen mensuel de la Seine à Paris



Le package hubeau est fourni avec une documentation complète ainsi que des vignettes proposant des exemples concrets de cartographie ou de statistiques sur l’observation des étiages, le suivi du niveau des nappes ou encore la qualité de l’eau des rivières.

Les contributeurs du package qui souhaitent ajouter de nouvelles API peuvent s’appuyer sur la documentation développeur qui fournit une méthode rapide comprenant un import automatique des caractéristiques des API à partir de leurs documentations en ligne au format swagger ou openapi.

Références

Dorchies, David. 2022. “hubeau: an R package for the Hub’Eau APIs.” Recherche Data Gouv. <https://doi.org/10.57745/XKN6NC>.

SK8 : Un service institutionnel de gestion et d'hébergement d'applications Shiny

Jean-François Rey 1* Elise Maigne 2† Isabelle Sanchez 3‡ David Carayon 4§
Joseph Tran 5¶

Résumé

Le projet SK8 (Shiny Kubernetes Service) est un projet qui regroupe une quinzaine d'ingénieur·es de l'institut INRAE et vise à proposer une solution de gestion et d'hébergement d'applications Shiny. Shiny a été largement adopté dans notre institut pour partager, valoriser et démocratiser les travaux scientifiques, or se pose systématiquement la question de l'hébergement de ces applications.

Partant du constat que différentes solutions isolées ont été mises en place pour répondre aux besoins des laboratoires de recherche, nous avons décidé de proposer une solution institutionnelle open-source afin de décloisonner les pratiques et fédérer la communauté R INRAE.

Le projet SK8 offre la possibilité d'héberger le code des applications Shiny sur une instance GitLab accessible à tous les agents INRAE. Des templates (Gitlab CI/CD) permettent de gérer la stabilité des applications (utilisation de{renv}), leur containerisation (Docker) et leur déploiement dans un cluster Kubernetes, le tout généré, développé et maintenu par l'équipe SK8. En terme d'utilisation, la démarche est simple puisqu'il suffit de déposer le code d'une application dans un projet Gitlab dédié. De plus l'utilisateur·rice du service reste propriétaire de son code.

La version Bêta de SK8 est accessible et utilisée depuis avril 2022. L'année 2024 marque un tournant par la reconnaissance de ce service par l'institut et son soutien par les départements et les directions INRAE, ceci afin de pérenniser ce dernier. Actuellement le service héberge plus de 60 applications et il dispose d'un catalogue public (<https://shiny.sk8.inrae.fr>).

Il s'agit d'un projet réalisé sur l'engagement personnelle d'agents INRAE regroupant des ingénieurs travaillant sur différentes thématiques scientifiques issus de différents CATIs (regroupement d'ingénieurs sur des thématiques de recherche) mais ayant une problématique commune, l'hébergement d'application R Shiny.

Dans ce poster nous présenterons le projet, le public visé et les cas d'usages, le workflow d'industrialisation d'hébergement, ainsi que l'écosystème sous-jacent.

Plus d'information sur le site web <https://sk8.inrae.fr>.

Mots-clefs (3 à 5) : Shiny - Hébergement - INRAE - Kubernetes - GitLab

*INRAE BioSP, jean-francois.rey@inrae.fr

†INRAE MIAT, elise.maigne@inrae.fr

‡INRAE MISTEA, isabelle.sanchez@inrae.fr

§INRAE INRAE ETTIS, david.carayon@inrae.fr

¶INRAE EGFV, joseph.tran@inrae.fr



Un service institutionnel de gestion et d'hébergement d'applications Shiny

INRAE
RÉPUBLIQUE FRANÇAISE
Liberté Egalité Fraternité

Élise Maigné¹, Isabelle Sanchez², David Carayon³, Joseph Tran⁴, Jean-François Rey⁵, et le reste de la team SK8⁶

¹ UR MIAT, 31326, Castanet-Tolosan – ² UMR MISTEA 34060 Montpellier – ³ UR ETTIS 33612 Cestas – ⁴ UMR EGTV 33140 Villenave-d'Ornon – ⁵ UR BioSP 84914 Avignon – ⁶ <https://sk8.inrae.fr/acteurs.html>

SK8, c'est quoi et pour qui ?

SK8 un projet proposant un service d'hébergement d'applications Shiny pour tous les agents INRAE ou tout projet financé par l'institut.

- Réspond aux besoins des laboratoires : solution pérenne et évolutrice,
- Permet une harmonisation des travaux de l'institut et une meilleure visibilité,
- S'appuie sur des infrastructures et outils open-source mis à disposition par l'institut,
- Facilite la maintenance et reproductibilité,
- Étend la facilité de développement d'applications avec Shiny en proposant la facilité de déploiement,
- Est accessible : ne nécessite aucune compétence en administration système,
- Est intégré dans une instance GitLab accessible à tous les agents INRAE.

Sous le capot de SK8

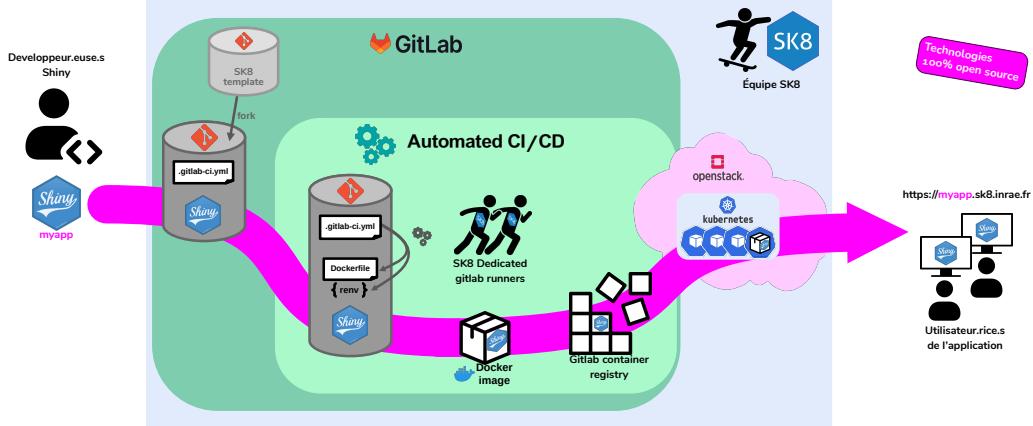
Des templates (Gitlab CI/CD) permettent d'automatiser et de gérer :

- la stabilité des applications (utilisation de {renv}),
- leur conteneurisation (Docker) et
- leur déploiement dans un cluster Kubernetes sur openstack, le tout géré, développé et maintenu par l'équipe SK8.

Pour les développeur·se·s Shiny, la démarche est simple :

1. déposer son application dans un projet Gitlab dédié à son projet,
2. cliquer sur le bouton de publication pour lancer une mise à jour,
3. rester propriétaire de son code.

SK8, comment ça roule ?



The diagram illustrates the SK8 deployment process. A developer (Developer.euse Shiny) creates a Shiny app ('myapp') and pushes it to a GitLab repository. This triggers an Automated CI/CD pipeline. The pipeline involves a 'fork' step where a template ('SK8 template') is copied into the repository. The developer then adds a '.gitlab-ci.yml' file containing a 'Dockerfile' and '{renv}' configuration. 'SK8 Dedicated gitlab runners' execute the pipeline, building a 'Docker image' and pushing it to a 'Gitlab container registry'. Finally, the application is deployed to an 'openstack' environment using 'kubernetes', and the user ('Utilisateur.rice.s de l'application') can access it via the URL <https://myapp.sk8.inrae.fr>.

SK8, c'est qui ?

- Deux groupes, une quinzaine d'ingénieur·e·s aux compétences diverses (devops, administration système, statistique, bioinformatique...) provenant de multiples unités INRAE.
- Le projet est rythmé en réunions mensuelles pour chaque groupe, ponctué par des sessions de travail communes (hackathon, AG).



Groupe user

- Développeur·e·s R & Shiny
- Expressent les besoins et retours utilisateur·rice·s
- Accompagnent les utilisateur·rice·s
- Rédigent la documentation

Groupe devops

- Administrateur·rice·s systèmes
- Gèrent l'infrastructure SK8 (Gitlab, CI/CD, K8S, ...)
- Veillent sur l'infrastructure (monitoring et logs)
- Mettent en place les solutions

Envie de participer ? Rejoignez nous !

SK8, quand et comment ?

Mars 2021 : lancement du projet
Avril 2022 : ouverture du service
Mai 2023 : ~50 applications hébergées

<https://sk8.inrae.fr>

Le poster SK8  Le site SK8 

Remerciements

CATIs INRAE actifs	Pour les cas d'usage
<ul style="list-style-type: none"> • IMOTEP • GEDEOP • CODEX • IUMAN • Bios4Biol • SoNet • Baric • PROSODIE • CITISES 	<ul style="list-style-type: none"> • Plateformes d'Épidémosurveillance 
Soutien technique et financier	
<ul style="list-style-type: none"> • Infrastructures INRAE  	

Figure 1: Le poster SK8

Créer un site pour partager sa recherche avec R, blogdown et Hugo

Fanny Ollivier 1*

Résumé (max 300 mots)

Communiquer sur sa recherche permet de participer à rendre la science plus ouverte, d'échanger et d'ouvrir de nouvelles collaborations. Le site académique répond à ce besoin de communication en offrant une interface accessible avec des contenus et des possibilités de contact. Il apparaît également comme un outil particulièrement utile pour les doctorants et post-doctorants, qui pourraient y être formés. Développer un site académique avec R offre des possibilités peu contraignantes et ouvertes. Dans ce poster une synthèse de la création de ce type de site sera proposée, de la conception et du développement avec Hugo et blogdown (voir Xie, Dervieux, and Hill 2024) au déploiement via Netlify et versionnage par Github.

Mots-clefs (3 à 5) : Site internet - Communication - Education - Blogdown - Hugo

Développement

Ce poster sera composé de 4 parties. Des images seront utilisées pour illustrer le propos, rendre sa lecture attrayante et ouvrir la discussion.

Dans une première partie introductory, l'intérêt de communiquer de façon ouverte sur ses objets de recherche sera souligné, notamment pour les chercheurs et chercheuses qui débutent et qui participent aux campagnes de recrutement. Cette partie visera également à rassurer le lecteur sur l'accessibilité de la démarche de création à toutes et à tous grâce aux outils proposés par la communauté R.

La seconde partie traitera de l'utilisation du package blogdown et du modèle Hugo pour la construction d'un site académique (Academic Hugo). Le modèle Hugo académique propose une structure comprenant une présentation, des résumés ainsi que des liens vers des publications, présentations, projets et une page de blog. Il est simple à compléter et peut être personnalisé dans son apparence, la langue utilisée ou les préférences de contact. Des exemples seront imaginés afin que les lecteurs puissent voir ce qu'il est possible de faire, et comment le faire.

La troisième partie s'intéressera au versionnage et au déploiement du site. L'exemple du versionnage sur github et du déploiement par Netlify sera proposé. De façon assez simple, faire communiquer Rstudio et github permet de garder des versions de travail et des versions à jour du site. Les versions à jour peuvent ensuite être publiées sur le site via Netlify, qui propose une solution entièrement gratuite. Cela sera montré à partir de captures d'écran de l'interface Rstudio, de github et de Netlify.

La dernière partie comportera des ressources, sous forme de liens et de références.

Parmi les captures d'images réalisées se trouveront des images issues du site que j'ai créé grâce aux conseils de membres de la communauté R. (<https://fannyllivier.netlify.app/>)

Références

Xie, Yihui, Christophe Dervieux, and Alison Presmanes Hill. 2024. *Blogdown: Create Blogs and Websites with r Markdown*. <https://github.com/rstudio/blogdown>.

*Université d'Angers 1, fanny.ollivier@univ-angers.fr

Les packages autour de JDemetra+ (rjd3) : une boîte à outils complète pour l'analyse des séries temporelles

Tanguy BARTHELEMY*

Résumé

JDemetra+ (Smyk et al. (2024)) est un logiciel open source d'analyse des séries temporelles dont les algorithmes sont accessibles via une interface graphique ou un écosystème de packages R (rjd3 Palate et al. (2024a)). Cet abstract a pour objectif de donner un aperçu des possibilités de ces packages, couvrant tous les domaines clef de l'analyse des séries temporelles (ajustement saisonnier, détection d'outliers, nowcasting, analyse des révisions, benchmarking et désagrégation temporelle). Nous détaillerons plus particulièrement l'ajustement saisonnier des séries hautes fréquences (infra-mensuelles) qui requiert une adaptation des algorithmes classiques.

Mots-clefs (3 à 5) : Statistique - Package

Développement

Introduction

JDemetra+ (Smyk et al. (2024)) est un logiciel open source d'analyse des séries temporelles (ajustement saisonnier, détection d'outliers, nowcasting...). Ce logiciel est recommandé par Eurostat et à tous les instituts du système statistique européen (ESS) depuis 2015. Le développement de JDemetra+ a été motivé par l'envie de proposer aux utilisateurs une interface et un support stable de production et d'études des séries temporelles. Une galerie de packages rjd3 (Palate et al. (2024a)) basé sur la version 3 de JDemetra+ a été développé. Ces packages permettent de faire le lien entre R et JDemetra+. Nous allons présenter ces outils, leur implémentation en R et leur avantage comparatif par rapport aux packages et outils disponibles sous R. L'intérêt essentiel des packages R est l'intégration des fonctionnalités de JDemetra+ dans l'environnement R (à travers la variété des traitements, des objets et des visualisations). Aujourd'hui la diversité des fonctionnalités proposées par les nouveaux packages R en version 3 rendent cet outil polyvalent permettant à la fois du travail d'étude one shot et de test mais aussi la mise en place de structure complexe de production, stable et reproductibles.

Méthode

Les méthodes d'ajustement saisonnier disponibles dans JDemetra+ relèvent des algorithmes classiques X13-Arima (Reg-Arima et décomposition à l'aide de moyennes mobiles) et TramoSeats (décomposition en modèles Arima). Ces algorithmes ont été étendus à des portées plus larges comme l'étude de fréquences non classiques (non mensuelle trimestrielle ou semestrielles) et des hautes fréquences (infra mensuelles). Pour cela deux packages dédiés (**rjd3x11plus** (Palate et al. (2024c)) et **rjd3highfreq** (Palate et al. (2024b))) implémentent des extensions de ces algorithmes.

Ces algorithmes sont écrits en Java. C'est pourquoi les packages R relatifs à JDemetra+ ont été développés à partir de cette structure et de ces routines de base pour ajouter une surcouche de connexion de JDemetra+ à l'environnement R. Cette connexion est effectuée grâce au Protocol buffer (le protocole de sérialisation de Google). Le Protobuf permet de faire le lien entre les programmes Java et l'environnement R. Il est nécessaire de décrire la structure de classe des programmes Java et d'expliquer un moyen de convertir chaque élément

*Insee, tanguy.barthelemy@insee.fr

en objet R. Les objets R seront nativement écrits en S4 mais pourront être réécrits en S3 pour être plus facilement manipulables.

JDemetra+ propose des outils d'estimations performants (modélisation Arima) et des nombreux tests statistiques. Le package **rjd3sts** est une boîte à outils polyvalente pour les modélisations espace état. Elle permet notamment de créer des modèles structurels de base, offrant une procédure d'ajustement saisonnier avec une décomposition explicite et une correction des jours ouvrables variables dans le temps.

Approche

Nous présentons un cas d'application : l'ajustement saisonnier de données haute fréquence. Comme expliqué précédemment, nous utiliserons les packages **rjd3x11plus** et **rjd3highfreq**. Les données haute fréquence sont désormais largement disponibles et de plus en plus utilisées dans la production statistique : données de cartes bancaires, validation des titres de transport, données journalières de suivi d'une épidémie. Ces données sont souvent saisonnières et leur analyse nécessite donc une désaisonnalisation préalable.

Conclusion

Les packages rjd3 offrent au datascientist de nouveaux outils pour l'analyse des séries temporelles en R. Ces packages sont en cours de développement et en cours de consolidation notamment en ce qui concerne les algorithmes les plus novateurs comme ceux consacrés à l'ajustement saisonnier des données haute fréquence.

Références

- Palate, Jean, Tanguy Barthelemy, Alain Quartier-la-Tente, Philippe Charles, Anna Smyk, and Corentin Lemasson. 2024a. “Organisation rjdemetra.” GitHub. <https://github.com/rjdemetra>.
- . 2024b. “Package rjd3highfreq.” GitHub. <https://github.com/rjdemetra/rjd3highfreq>.
- . 2024c. “Package rjd3x11plus.” GitHub. <https://github.com/rjdemetra/rjd3x11plus>.
- Smyk, Anna, Tanguy Barthelemy, Alain Quartier-la-Tente, and Karsten Webel. 2024. *JDemetra+ Documentation*. jdemetra-new-documentation.netlify.app. <https://jdemetra-new-documentation.netlify.app/>.

Poster autour du package {datamods}

Samra GOUMRI 1*

Victor PERRIER 2†

Résumé (max 300 mots)

Le poster s'articulera autour des fonctionnalités du package {datamods}. Pour explications, le package {datamods} propose différents modules Shiny pour importer et manipuler des données pouvant être utilisés dans n'importe quelle application Shiny standard ou addins RStudio, comme notamment le package {esquisse}. En effet, il fournit des modules Shiny personnalisés pour importer des données à partir de diverses sources, sélectionner, renommer et convertir des variables dans un ensemble de données et valider le contenu avec le package {validate}.

Mots-clefs (3 à 5) : Package - Shiny - Data

Développement

La présentation du poster comprendra une présentation des principaux modules du package, comme par exemple:

- module d'import, pour importer des données depuis des fichiers (txt, csv, xlsx, ...) ou simplement en copiant/collant des données,
- module d'édition, pour ajouter / modifier / supprimer des lignes dans un `data.frame`,
- module de filtre, pour filtrer un `data.frame` quelque soit le type des colonnes,

et encore plus : échantillonner des données, créer des colonnes, réordonner les `levels` d'un `factor` ...

La plupart des modules retournent le code permettant de répéter l'opération dans un script (travail en cours).

(Perrier and al (2024)).

Références

Perrier, Victor, and al. 2024. “GitHub Du Package Datamods.” [Https://Github.com/dreamRs/Datamods](https://Github.com/dreamRs/Datamods).

*DreamRs, samra.goumri@dreamrs.fr

†DreamRs, victor.perrier@dreamrs.fr

Distance/Divergence entre distributions t multivariées

Pierre Santagostini 1* Nizar Bouhlel 2†

Résumé

Différentes mesures peuvent être utilisées pour le calcul de la distance entre deux distributions de probabilité. Parmi celles-ci, on trouve la divergence de Rényi, de Bhattacharyya, de Hellinger ou de Kullback Leibler. Les applications sont nombreuses en image, en signal et dans d'autres domaines connexes en informatique telles que la reconnaissance de forme, l'apprentissage automatique, la détection de changements, la sélection de modèles, la recherche d'image par le contenu, etc. Le package **mstudentd** fournit des fonctions pour le calcul de ces divergences entre lois t multivariées (lois de Student multivariées) en utilisant des expressions analytiques récemment introduites (Bouhlel and Rousseau 2023).

Mots-clefs : Package - Divergence de Renyi - Divergence de Kullback-Leibler - Distibution t multivariée

Package mstudentd : divergence entre deux distributions t multivariées centrées

Le package **mstudentd** (Santagostini and Bouhlel 2024) fournit des fonctions pour le calcul de la divergence entre une distribution t multivariée centrée à ν_1 degrés de liberté, de matrice de corrélation Σ_1 et une distribution à ν_2 degrés de liberté, de matrice de corrélation Σ_2 :

- Divergence de Rényi d'ordre $\beta = 0.25$:

```
diststudent(nu1, Sigma1, nu2, Sigma2, dist = "renyi", bet = 0.25)
```

- Divergence de Bhattacharyya :

```
diststudent(nu1, Sigma1, nu2, Sigma2, dist = "bhattacharyya")
```

- Divergence de Hellinger :

```
diststudent(nu1, Sigma1, nu2, Sigma2, dist = "hellinger")
```

- Divergence de Kullback-Leibler :

```
kldstudent(nu1, Sigma1, nu2, Sigma2)
```

Des expressions analytiques des divergences de Rényi et Kullback-Leibler, exposées ci-après, ont récemment été introduites par Bouhlel and Rousseau (2023). Les calculs implémentés dans ces fonctions **diststudent** et **kldstudent** sont ainsi basés sur des expressions exactes des divergences. Cela permet un calcul plus précis, comparé à un calcul approché par la méthode de Monte Carlo, ou à un calcul de la divergence entre des distributions de probabilité discrétisées (différents packages R fournissent des fonctions pour la divergence entre des distributions discrètes, par exemple **rd** du package **divo** ou **kld** du package **dlookr**).

Calcul de la divergence entre distributions t multivariées

Soient \mathbf{X}_1 et \mathbf{X}_2 deux vecteurs aléatoires distribuées selon des lois t multivariées centrées (à p variables) à ν_i degrés de liberté, de vecteurs moyennes $\boldsymbol{\mu}_i = \mathbf{0}$ et de matrices de corrélation Σ_i , de densités de probabilité :

$$f(\mathbf{x}|\nu_i, \boldsymbol{\mu}_i = \mathbf{0}, \Sigma_i) = f(\mathbf{x}|\nu_i, \Sigma_i) = \frac{\Gamma\left(\frac{\nu_i+p}{2}\right)}{\Gamma\left(\frac{\nu_i}{2}\right)(\nu_i\pi)^{p/2}} |\Sigma_i|^{-1/2} \left(1 + \frac{1}{\nu_i} \mathbf{x}^T \Sigma_i^{-1} \mathbf{x}\right)^{-\frac{\nu_i+p}{2}}, \quad i = 1, 2 \quad (1)$$

*Institut Agro, Univ Angers, INRAE, IRHS, SFR QUASAV, F-49000 Angers, France, pierre.santagostini@agrocampus-ouest.fr

†Institut Agro, Univ Angers, INRAE, IRHS, SFR QUASAV, F-49000 Angers, France, nizar.bouhlel@agrocampus-ouest.fr

Divergence de Rényi

Une expression de la divergence de Rényi d'ordre β ($\beta > 0, \beta \neq 1$) entre ces deux distributions est donnée par Bouhlel and Rousseau (2023) :

$$D_R^\beta(\mathbf{X}_1\|\mathbf{X}_2) = \frac{1}{\beta-1} \left[\beta \ln \left(\frac{\Gamma(\frac{\nu_1+p}{2}) \Gamma(\frac{\nu_2}{2}) \nu_2^{\frac{p}{2}}}{\Gamma(\frac{\nu_2+p}{2}) \Gamma(\frac{\nu_1}{2}) \nu_1^{\frac{p}{2}}} \right) + \ln \left(\frac{\Gamma(\frac{\nu_2+p}{2})}{\Gamma(\frac{\nu_2}{2})} \right) + \ln \left(\frac{\Gamma(\delta_1 + \delta_2 - \frac{p}{2})}{\Gamma(\delta_1 + \delta_2)} \right) - \frac{\beta}{2} \sum_{i=1}^p \ln \lambda_i + \ln F_D^{(p)} \left(\underbrace{\delta_1, \frac{1}{2}, \dots, \frac{1}{2}}_p; \delta_1 + \delta_2; 1 - \frac{\nu_2}{\nu_1} \frac{1}{\lambda_1}, \dots, 1 - \frac{\nu_2}{\nu_1} \frac{1}{\lambda_p} \right) \right] \quad (2)$$

où $\delta_1 = \frac{\nu_1+p}{2}\beta$, $\delta_2 = \frac{\nu_2+p}{2}(1-\beta)$, $\lambda_1 < \dots < \lambda_{p-1} < \lambda_p$ sont les valeurs propres de la matrice carrée $\Sigma_1 \Sigma_2^{-1}$ et $F_D^{(p)}$ est la fonction de Lauricella D -hypergéométrique :

$$F_D^{(p)}(a; b_1, \dots, b_p; g; x_1, \dots, x_p) = \sum_{m_1 \geq 0} \dots \sum_{m_p \geq 0} \frac{(a)_{m_1+\dots+m_p} (b_1)_{m_1} \dots (b_p)_{m_p}}{(g)_{m_1+\dots+m_p}} \frac{x_1^{m_1}}{m_1!} \dots \frac{x_p^{m_p}}{m_p!}$$

$(q)_i$ étant le symbole de Pochhammer : $(q)_i = \frac{\Gamma(q+i)}{\Gamma(q)} (i = 1, 2, \dots)$.

La condition $\left|1 - \frac{\nu_2}{\nu_1} \frac{1}{\lambda_i}\right| < 1$, $i = 1, \dots, p$ assure la convergence de la série. Quand cette condition n'est pas vérifiée, des transformations de la fonction de Lauricella permettent d'obtenir la convergence.

Divergence de Bhattacharyya, divergence de Hellinger

Les divergences de Bhattacharyya et de Hellinger sont calculées à partir de la divergence de Rényi :

- Divergence de Bhattacharyya entre \mathbf{X}_1 et \mathbf{X}_2 : $D_B(\mathbf{X}_1\|\mathbf{X}_2) = \frac{1}{2} D_R^{1/2}(\mathbf{X}_1\|\mathbf{X}_2)$
- Divergence de Hellinger : $D_H(\mathbf{X}_1\|\mathbf{X}_2) = 1 - \exp \left(-\frac{1}{2} D_R^{1/2}(\mathbf{X}_1\|\mathbf{X}_2) \right)$

Divergence de Kullback-Leibler

La divergence de Kullback-Leibler entre \mathbf{X}_1 et \mathbf{X}_2 est donnée par : $D_{KL}(\mathbf{X}_1\|\mathbf{X}_2) = \lim_{\beta \rightarrow 1} D_R^\beta(\mathbf{X}_1\|\mathbf{X}_2)$. Soit ψ la fonction digamma : $\psi(x) = \frac{d}{dx} \ln \Gamma(x) = \frac{\Gamma'(x)}{\Gamma(x)}$. D'après (Bouhlel and Rousseau 2023) :

$$D_{KL}(\mathbf{X}_1\|\mathbf{X}_2) = \ln \left(\frac{\Gamma(\frac{\nu_1+p}{2}) \Gamma(\frac{\nu_2}{2}) \nu_2^{\frac{p}{2}}}{\Gamma(\frac{\nu_2+p}{2}) \Gamma(\frac{\nu_1}{2}) \nu_1^{\frac{p}{2}}} \right) + \frac{\nu_2 - \nu_1}{2} \left[\psi \left(\frac{\nu_1+p}{2} \right) - \psi \left(\frac{\nu_1}{2} \right) \right] - \frac{1}{2} \sum_{i=1}^p \ln \lambda_i - \frac{\nu_2 + p}{2} \times \frac{\partial}{\partial a} \left\{ F_D^{(p)} \left(a, \underbrace{\frac{1}{2}, \dots, \frac{1}{2}}_p; a + \frac{\nu_1+p}{2}; 1 - \frac{\nu_1}{\nu_2} \lambda_1, \dots, 1 - \frac{\nu_1}{\nu_2} \lambda_p \right) \right\} \Big|_{a=0} \quad (3)$$

La convergence est assurée quand $|1 - \frac{\nu_1}{\nu_2} \lambda_i| < 1$, $i = 1, \dots, p$.

Références

- Bouhlel, N., and D. Rousseau. 2023. “Exact Rényi and Kullback-Leibler Divergences Between Multivariate t -Distributions.” *IEEE Signal Processing Letter*. <https://doi.org/10.1109/LSP.2023.3324594>.
- Santagostini, P., and N. Bouhlel. 2024. *mstudentd: Multivariate t Distribution*. <https://CRAN.R-project.org/package=mstudentd>.

A survey translation tool to easily migrate from Qualtrics to LimeSurvey

Camille STRABONI*

Résumé

A lot of labs in various fields such as Social Sciences, Cognitive Sciences, Humanities have been using the proprietary survey tool Qualtrics for years to run online experiments. Despite the high price of the Qualtrics license and the equally high efficiency of other software such as popular open-source LimeSurvey, they have a huge amount of survey templates stored on their account and manually migrating from a platform to another is time-consuming.

To tackle this issue, I developed an easy-to-use R package that allows to automatically translate Qualtrics surveys into LimeSurvey. The translated survey retains the structure, question types and contents such as subquestions, choices and language versions of the former Qualtrics survey. Importantly, this solution also has the advantage to save the data locally when a LimeSurvey instance is installed in a local server that helps comply with French and UE data protection regulations.

Mots-clefs : package – open-source – Qualtrics – survey - online experiment - LimeSurvey

Development

Qualtrics is an online survey platform widely used in various fields of academia⁽¹⁾ such as in medical research⁽²⁾, cognitive studies⁽³⁾ or psychology⁽⁴⁾. It is a proprietary tool owned and sold by Silver Lake Technology Management that allows to create surveys online and record data. The R package I developed automates the migration of surveys from a Proprietary to an Open-Source Solution, in this case LimeSurvey. LimeSurvey is a software with even more complex surveys than Qualtrics by making use of custom Javascript. LimeSurvey can be installed on a local server that allows to save the collected data locally and comply with French and UE data protection regulations.

This package builds on other packages^{(5), (6)} to translate Qualtrics surveys to LimeSurvey retaining its structure, languages versions, questions (types, subquestions, answers, choices).

* Département d'Etudes Cognitives, Ecole normale supérieure – Paris Sciences et Lettres,
camille.straboni@ens.psl.eu

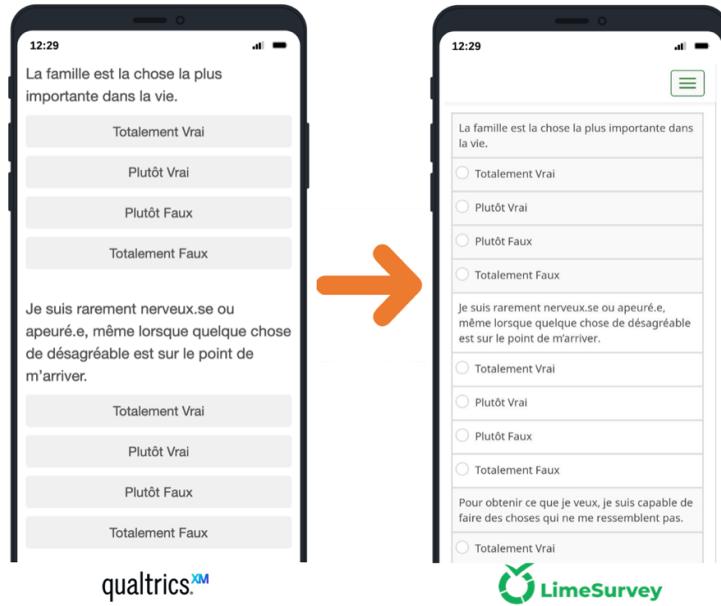


Figure 1 : Render a survey in Qualtrics (left) and LimeSurvey (left) are very similar

Although the renders in Qualtrics and LimeSurvey are very similar (Figure 1), multi-choice questions with the same answer choices can appear redundant in LimeSurvey after translation (figure 2a). In order to improve the design of the migrated surveys, options have been implemented in the package so that questions with redundant choice answers (figure 2a) can be merged into a table (figure 2b).

*La famille est la chose la plus importante dans la vie.
Choose one of the following answers

- Totalement Vrai
- Plutôt Vrai
- Plutôt Faux
- Totalement Faux

	Totalement Vrai	Plutôt Vrai	Plutôt Faux	Totalement Faux
La famille est la chose la plus importante dans la vie.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Je suis rarement nerveux.se ou apeuré.e, même lorsque quelque chose de désagréable est sur le point de m'arriver.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Pour obtenir ce que je veux, je suis capable de faire des choses qui ne me ressemblent pas.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
J'aime faire quelque chose pour laquelle je suis performante.	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

*Je suis rarement nerveux.se ou apeuré.e, même lorsque quelque chose de désagréable est sur le point de m'arriver.
Choose one of the following answers

- Totalement Vrai
- Plutôt Vrai
- Plutôt Faux
- Totalement Faux

Figure 2a : After translation from Qualtrics to LimeSurvey, two different questions with the same redundant 4 choices (Totalement Vrai, Plutôt Vrai, Plutôt Faux, Totalement Faux)

Figure 2b : Questions with previously redundant choice answers are merged into a single table with answer choices as column headers and questions as rows

In conclusion, this R package that automates the translation of Qualtrics surveys into LimeSurvey, makes it easy to use LimeSurvey, an inexpensive open-source software package, and will have a significant impact for researchers and survey developers looking to streamline their workflow.

Références

-
- (1) Douglas BD, Ewell PJ, Brauer M. Data quality in online human-subjects research: Comparisons between MTurk, Prolific, CloudResearch, Qualtrics, and SONA. *PLoS One*. 18(3), (2023).
 - (2) Belliveau, J., et al. The validity of qualtrics panel data for research on video gaming and gaming disorder. *Experimental and Clinical Psychopharmacology*, 30(4), 424–431, (2022).
 - (3) Altay, S. et al. Scaling up interactive argumentation by providing counterarguments with a chatbot. *Nat Hum Behav* 6, 579–592 (2022). <https://doi.org/10.1038/s41562-021-01271-w>
 - (4) McBride O et al. Monitoring the psychological, social, and economic impact of the COVID-19 pandemic in the population: Context, design and conduct of the longitudinal COVID-19 psychological research consortium (C19PRC) study. *Int J Methods Psychiatr Res.* (2021).
 - (5) Moritz Marbach (2020), QSF Package : <https://github.com/sumtxt/qsf>
 - (6) Christian Testa's (2017), Quickstart Guide to understanding the Qualtrics Survey File : <https://gist.github.com/ctestaa01/d4255959dace01431fb90618d1e8c241>

RecForest : Forêts aléatoires de survie pour l'analyse des événements récurrents en R

Juliette Murris*

Sandrine Katsahian†

Audrey Lavenu‡

Résumé

Les forêts aléatoires de survie permettent de modéliser les effets de prédicteurs sur des données de survie en s'affranchissant d'hypothèses, comme la linéarité ou la faible dimension (le nombre d'individus supérieur au nombre de prédicteurs). Leur utilisation est ainsi accrue en recherche médicale. Nous avons récemment adapté ces forêts aux données de survie avec événements récurrents avec **RecForest**, en présence ou non d'un événement terminal pour prédire le nombre attendu d'événements. L'objectif de ce travail est d'introduire l'utilisation de **RecForest** en programmation R. Notre approche est en 5 étapes, pour i) discerner la pertinence des événements récurrents et terminaux, ii) développer des arbres et construire la forêt, iii) évaluer minutieusement la performance, iv) fournir l'importance des variables, et v) permettre des prédictions sur de nouvelles données. Pour l'illustration, le jeu de données `readmission` du package `frailtypack` de R a été utilisé.

Mots-clefs : Biostatistique – Analyse de survie – Santé

1 Introduction

Les modèles d'apprentissage automatique, tels que les forêts aléatoires, sont de plus en plus appliqués à l'analyse de données de survie, notamment avec l'utilisation courante des forêts aléatoires de survie (RSF) [Huang et al., 2023, Ishwaran et al., 2008]. Toutefois, face à des événements récurrents tels que les hospitalisations, les rechutes, ou les crises d'asthme, l'analyse de survie classique, et ainsi les RSF, ne considère que la première occurrence. Pour pallier cela, nous avons étendu les RSF aux événements récurrents pour prédire le nombre attendu d'événements pour chaque individu au cours du temps avec **RecForest**. L'objectif ici est d'introduire l'utilisation de **RecForest** en programmation R.

2 Vue d'ensemble de l'algorithme

Les données en entrée sont constituées de plusieurs observations pour chaque individu, avec des variables indiquant la présence (ou non) d'événements récurrents, la présence (ou non) d'un événement terminal, le temps associé à la survenue de chaque événement, et d'éventuelles variables explicatives, dépendantes ou indépendantes du temps.

RecForest fonctionne de manière similaire aux RSF de Ishwaran et al. [2008]. Pour chaque échantillon bootstrap, des arbres de survie adaptés aux événements récurrents sont construits. La forêt agrège les résultats obtenus pour chaque arbre afin d'obtenir une estimation globale.

Règle de division. A chaque noeud, l'objectif est de discriminer au mieux les données à partir de *mtry* variables sélectionnées aléatoirement. Cette discrimination est faite suivant la présence ou non d'un événement terminal en utilisant la statistique du pseudo-score test ou la statistique du test de Wald d'un modèle marginal de Gosh-Lin [Lawless and Nadeau, 1995, Ghosh and Lin, 2002].

Estimation et agrégation. L'estimation est le nombre attendu d'événements pour chaque individu jusqu'au temps t , notée $\hat{M}(t | x)$, qui est l'aggrégation des estimations $\hat{\mu}_b(t | x)$ pour chaque arbre de la forêt. Suivant la présence ou non d'un événement terminal,

$$\hat{M}(t | x) = \frac{1}{B} \sum_{b=1}^B \hat{\mu}_b(t | x) = \begin{cases} = \frac{1}{B} \sum_{b=1}^B \hat{R}_b(t | x) = \frac{1}{B} \sum_{b=1}^B \int_0^t \frac{N_b(du | x)}{Y_b(du | x)} & \text{sans événement terminal,} \\ = \frac{1}{B} \sum_{b=1}^B \int_0^t \hat{S}_b(u | x) d\hat{R}_b(u | x) & \text{avec un événement terminal.} \end{cases}$$

*HeKA, Inria, Inserm, Université Paris Cité, Paris, France, juliette.murris@inria.fr

†Centre d'Investigation Clinique 1418 Épidémiologie Clinique, Paris, France, sandrine.katsahian@aphp.fr

‡Institut de Recherche Mathématique de Rennes (IRMAR), Rennes, France, audrey.lavenu@univ-rennes.fr

avec B le nombre d'arbres, \mathbf{x} un vecteur de variables explicatives, dépendantes ou indépendantes du temps, N le nombre d'événements noeud-spécifique, Y le nombre d'individus à risque, \hat{S} l'estimateur Kaplan-Meier de la fonction de survie.

Elagage de chaque arbre. De même que Devaux et al. [2023], nous proposons deux règles d'arrêt pour chaque noeud terminal : (i) un nombre minimal d'événements appelé *minsplit*, et (ii) un nombre minimal d'individus appelé *nodesize*.

Les sorties de l'algorithme correspondent aux prédictions du nombre attendu d'événements récurrents au cours du temps pour chaque individu.

3 Mise en pratique

Données d'entrée. Les données `readmission` sont fréquemment utilisées pour des principes méthodologiques [Rondeau et al., 2012]. Cette base contient les réhospitalisations de 403 patients atteint d'un cancer colorectal.

```
1 data(readmission, package = "frailtypack")
2 X <- readmission |> dplyr::select(id, chemo, sex, dukes) |> dplyr::group_by(id) |>
  as.data.frame()
3 Y <- readmission |> dplyr::pull(id, t.start, t.stop, event, death)
```

Création d'un objet RecForest. Les hyper-paramètres à fixer (ou à optimiser) sont le nombre de variables tirées aléatoirement à chaque noeud est *mtry*, le nombre minimal d'événements est *minsplit*, le nombre minimal d'individus est *nodesize* et le nombre d'arbres *ntrees*.

```
1 mtry <- 5 # number of candidate variables randomly drawn at each node
2 minsplit <- 5 # minimal number of events required to split the node
3 nodesize <- 5 # minimal number of subjects required in both child nodes to split
4 n_trees <- 100 # number of trees
5 method <- "GL" # Ghosh-Lin for recurrent events, with a terminal event
6 params <- list(seed = seed, mtry = mtry, minsplit = minsplit, nodesize = nodesize,
  method = method, n_trees = n_trees)
7 my_recforest <- RecForest(X = X, Y = Y, params = params)
```

Evaluation. Pour l'évaluation des performances, nous utilisons des métriques adaptées aux événements récurrents, soient des versions étendues du C-index et de l'erreur quadratique moyenne.

```
1 c_index(my_recforest, X_new = NULL) # X_new = NULL refers to OOB samples
2 mse(my_recforest, X_new = NULL) # X_new = NULL refers to OOB samples
```

Importance des variables. L'importance d'une variable est évaluée par permutation, correspondant à l'impact de perturbations aléatoires dans l'échantillon sur l'erreur OOB.

```
1 vimp(my_recforest, n_permutations = 10)
```

Prédictions. A partir de nouvelles données en entrée, `RecForest` est utilisée pour prédire les nombres attendus d'événements pour chaque nouvel individu.

```
1 predictions = predict(my_recforest, X_new = X_new)
```

Références

- Anthony Devaux, Catherine Helmer, Robin Genuer, and Cécile Proust-Lima. Random survival forests with multivariate longitudinal endogenous covariates. *Statistical Methods in Medical Research*, 32(12) :2331–2346, 2023.
- Debashis Ghosh and Danyu Y Lin. Marginal regression models for recurrent and terminal events. *Statistica Sinica*, pages 663–688, 2002.
- Yinan Huang, Jieni Li, Mai Li, and Rajender R Aparasu. Application of machine learning in predicting survival outcomes involving real-world data : a scoping review. *BMC Medical Research Methodology*, 23(1) :268, 2023.
- Hemant Ishwaran, Udaya B Kogalur, Eugene H Blackstone, and Michael S Lauer. Random survival forests. 2008.
- Jerald F Lawless and Claude Nadeau. Some simple robust methods for the analysis of recurrent events. *Technometrics*, 37(2) :158–168, 1995.
- Virginie Rondeau, Yassin Marzroui, and Juan R Gonzalez. frailtypack : an r package for the analysis of correlated survival data with frailty models using penalized likelihood estimation or parametrical estimation. *Journal of Statistical Software*, 47 :1–28, 2012.

Des tableaux et des graphiques prêts à publication avec les packages R `{tabularise}` et `{chart}` de la suite SciViews

Guyliann Engels 1*

Philippe Grosjean 2†

Résumé (max 300 mots)

Au sein de la suite de packages SciViews (Grosjean 2023), voir aussi <https://sciviews.r-universe.dev/>, nous présentons `{tabularise}` pour les tableaux et `{chart}` pour les graphiques. Ces packages s'appuient respectivement sur `{flextable}` et `{ggplot2}` pour offrir des tableaux et des graphiques d'objets R dans une forme pratiquement prête à publication.

Pour simplifier la création de ces tableaux et graphiques, `{tabularise}` et `{chart}` supportent l'anglais et le français et prennent en compte les labels des variables automatiquement (pour les axes des graphiques ou les noms de colonnes des tableaux). Les équations, au format LaTeX, sont également automatiquement générées pour une large gamme de modèles statistiques grâce à `{equatiomatic}` et intégrés dans les tableaux.

Ces tableaux et graphiques étant compatibles avec `{flextable}` et `{ggplot2}`, ils restent entièrement personnalisables par la suite. Pour l'utilisateur avancé ou plus créatif, ils seront plutôt considérés comme de premières étapes de production de contenu plus spécifique.

Mots-clefs : Prêt à publication – Tableau – Graphique – tabularise – chart – SciViews

Développement

Quarto (Allaire 2023) est couramment utilisé pour générer facilement des documents publiables de manière reproductible. Les tableaux et graphiques demandent, par contre, un peu plus de travail pour être conformes aux normes de publication. Dans l'univers SciViews (<https://sciviews.r-universe.dev/>), le package `{tabularise}` accélère le processus de création de tableaux en une seule instruction pour une série d'objets R et `{chart}` en fait de même pour les graphiques. Ces premières versions peuvent ensuite être remaniées avec `{flextable}` et `{ggplot2}`, respectivement. `{tabularise}` et `{chart}` gèrent l'anglais et le français et prennent en compte automatiquement les labels et unités des variables (attributs "label" et "units"). Ces labels et unités peuvent être rajoutés directement dans le data.frame, ou en utilisant `data.io::labellise()` (`{data.io}` est un autre package de l'univers SciViews).

Ces packages ne sont pas encore sur CRAN car ils sont en cours de développement et susceptibles d'évoluer de manière difficilement compatible avec CRAN. L'installation depuis R-universe est donc recommandée.

```
# Installation des packages nécessaires
#install.packages(c('tabularise', 'chart', 'data.io', 'equatags', 'modelit'),
#  repos = c('https://sciviews.r-universe.dev', 'https://cloud.r-project.org'))
library(tabularise); library(chart); library(modelit)
flextable::set_flextable_defaults(font.family = "Arial", font.size = 9)
```

De nombreuses méthodes sont disponibles pour `tabularise()` et `chart()` au travers d'autres packages dans l'univers SciViews comme `{modelit}`, `{inferit}` ou `{exploreit}`. Pour les objets `lm`, `glm`, `nls...`, les tableaux `tabularise()` de `{modelit}` intègrent également des équations en s'appuyant sur `{equatiomatic}` (Anderson, Heiss, and Sumners 2024). Voici un exemple :

*Service d'écologie numérique, Institut Complexys & Infortech, Université de Mons, Belgique, guyliann.engels@umons.ac.be
†Service d'écologie numérique, Institut Complexys & Infortech, Université de Mons, Belgique, philippe.grosjean@umons.ac.be

```

data(Loblolly, package = "datasets")
Loblolly$height <- round(Loblolly$height * 0.3048, 2) # pieds -> m
Loblolly <- data.io::labelise(Loblolly, # Labelisation des variables
  label = list(height = "Hauteur", age = "Age", Seed = "Semence"),
  units = list(height = "m", age = "ans"))
set.seed(3652)
pine <- dplyr::slice_sample(Loblolly, n = 1, by = Seed) # une mesure/arbre
pine_lm <- lm(data = pine, height ~ age + I(age^2)) # modèle quadratique
pine_lm |> tabularise$tidy(lang = "fr") # tableau formaté

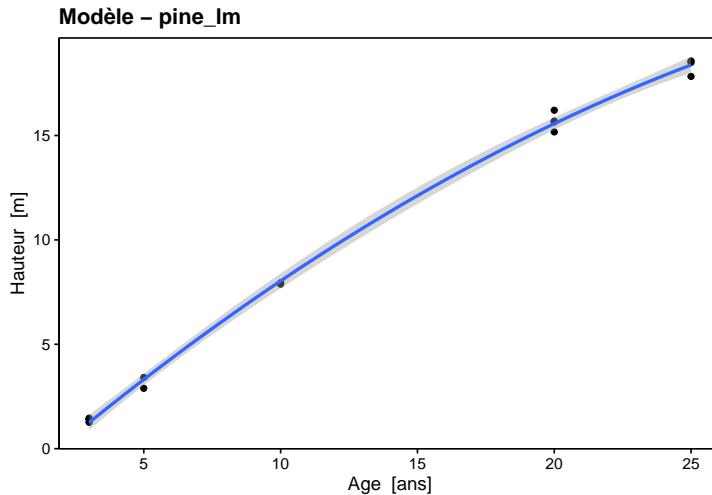
```

Modèle linéaire				
$\text{Hauteur [m]} = \alpha + \beta_1(\text{Age [ans]}) + \beta_2(\text{age}^2) + \epsilon$				
Terme	Valeur estimée	Ecart type	Valeur de t	Valeur de p
α	-2.0393	0.2895	-7.04	$2.14 \cdot 10^{-5}***$
β_1	1.1344	0.0584	19.41	$7.37 \cdot 10^{-10}***$
β_2	-0.0127	0.0021	-6.06	$8.14 \cdot 10^{-5}***$

$0 \leq *** < 0.001 < ** < 0.01 < * < 0.05$

Plusieurs graphiques sont disponibles avec {chart} (nous n'en montrons qu'un seul ici par manque de place, pour plus de détails, voyez <https://sciviews.r-universe.dev/articles/modelit/modelit.html> et <https://wp.sciviews.org/sdd-umons2/?iframe=wp.sciviews.org/sdd-umons2-2023/outils-de-diagnostic-suite.html>).

```
chart$model(pine_lm, lang = "fr") # graphique du modèle
```



Références

- Allaire, JJ. 2023. *Quarto: R Interface to 'Quarto' Markdown Publishing System*. <https://CRAN.R-project.org/package=quarto>.
- Anderson, Daniel, Andrew Heiss, and Jay Sumners. 2024. *Equatiomatic: Transform Models into 'LaTeX' Equations*. <https://github.com/datalorax/equatiomatic>.
- Grosjean, Philippe. 2023. *SciViews::r*. MONS, Belgique: UMONS. <https://sciviews.r-universe.dev/>.

R POUR L'OCÉANO-MÉTÉO ET L'INGÉNIERIE MARINE

Nicolas Raillard

Abstract

Ces dernières années, les zones côtières ont fait l'objet d'une attention accrue, avec le développement accéléré des activités humaines d'une part, et le réchauffement climatique d'autre part, qui expose les populations vivant dans les zones côtières à des risques côtiers plus fréquents et plus intenses. Ces deux problématiques ont en commun la nécessité de bien connaître les conditions de mer à proximité des côtes, afin de qualifier précisément les risques auxquels les personnes et les structures sont exposées, un objectif pour lequel la modélisation statistique est particulièrement bien adaptée, et donc l'utilisation de R est très répandue. Dans cette présentation, nous nous concentrerons sur les structures marines, qui sont soumises aux effets combinés du vent, des vagues et des courants. Dans un premier temps, nous présenterons les données disponibles et les travaux menés à l'IFREMER pour acquérir de nouvelles données et connaissances. Dans un deuxième temps, nous présenterons les apports de la modélisation statistique pour préciser les conditions opérationnelles de ces structures, ainsi que les conditions extrêmes impactant leur conception en présentant les packages R que j'utilise au quotidien.

telraamStats : visualisation des mobilités pour la recherche et les citoyen·ne·s

Ioana Gavra* Ketsia Guichard-Sustowski† Pascal Irz‡ Loïc Le Marrec§
Véronique Thelen¶

Résumé

Le package telraamStats facilite la collecte, l'analyse et la visualisation des données de mobilité des capteurs Telraam. Il les rend accessibles à tou·te·s, offrant aux utilisateur·trice·s novices un accès simplifié aux données et aux chercheurs et chercheuses des outils standardisés. Des exemples concrets seront présentés, notamment l'application Mov'Around [1] à Châteaubourg (Ille-et-Vilaine), illustrant le passage d'une initiative citoyenne à une recherche universitaire, soulignant la place de R dans cette transition.

Mots-clefs : Package - Données ouvertes - Shiny - Sciences citoyennes - Données spatio-temporelles.

Développement

Le package telraamStats est dédié à la collecte, à l'analyse et à la visualisation de données de mobilité issues de capteurs Telraam. D'une part, il rend ces données accessibles aux utilisateur·ice·s les moins familier·e·s avec les requêtes API, d'autre part, il fournit au monde académique des outils de traitement standardisés pour une étude plus approfondie.

Les capteurs Telraam [2] sont des dispositifs de comptage de trafic pouvant être installés à moindre coût par divers acteurs (municipalités, citoyens, associations), pour enregistrer le flux horaire de différents modes de transports (poids lourds, voitures, deux-roues, piétons) sur un segment de route.

Les comptages issus des capteurs ont l'intérêt majeur d'être accessibles sous forme de données ouvertes [3] et d'offrir à chaque citoyen·ne la possibilité d'avoir en temps réel des informations sur le trafic local. Le package proposé s'inscrit dans cette philosophie du logiciel libre comme outil d'alimentation du débat public.

Des exemples concrets de visualisation des données de circulation seront présentés sur différents territoires afin de montrer les possibilités du package. Une des illustrations portera sur une utilisation de telraamStats par l'application R-Shiny Mov'Around [1] qui est un visualisateur interactif des données de circulation de la ville de Châteaubourg (Ille-et-Vilaine).

Ce territoire n'a pas été choisi par hasard : le projet a été initié par une association locale de protection de l'environnement, Agis-Ta-Terre, à l'origine de l'installation de nombreux capteurs dans la ville. Châteaubourg est ainsi la petite ville française la plus densément couverte avec un historique de 3 ans de données de circulation sur une vingtaine de capteurs. Le dispositif capteurs - package - application a déjà largement irrigué le débat public sur un projet de contournement routier de Châteaubourg [4] [5] [6].

*Univ Rennes 2, CNRS, IRMAR - UMR 6625, ioana.gavra@univ-rennes2.fr

†Univ Rennes, IRMAR - UMR 6625, CREM - UMR 6211, ketsia.guichard@univ-rennes.fr

‡OFB, Direction régionale Bretagne, pascal.irz@ofb.gouv.fr

§Univ Rennes, CNRS, IRMAR - UMR 6625, loic.lemarrec@univ-rennes.fr

¶Univ Rennes, CNRS, CREM - UMR 6211, veronique.thelen@univ-rennes.fr

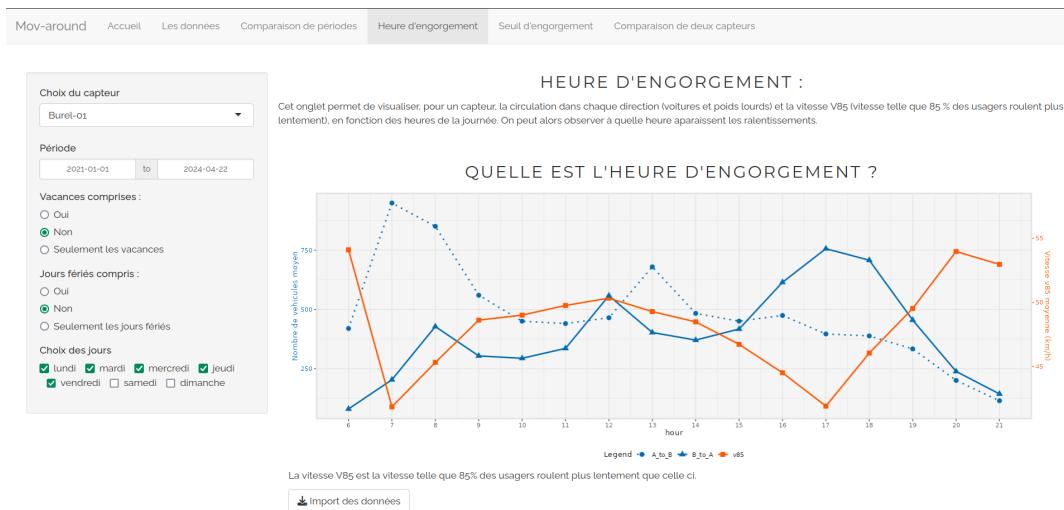


FIGURE 1 – Application R Shiny Mov'Around [1]

Ce projet résulte donc de la transition d'une initiative citoyenne vers un sujet de recherche universitaire. Notre présentation mettra également en évidence la place de R dans ce processus, soulignant le potentiel de ce type d'outils pour favoriser le débat public et les sciences citoyennes.

Remerciements

Ce travail a bénéficié d'une aide de l'État gérée par l'Agence Nationale de la Recherche au titre du programme d'investissements d'avenir intégré à France 2030, portant la référence ANR-11-LABX-0020-01. Il a bénéficié également d'un Défi Scientifique de l'Université de Rennes et le projet est soutenu par la plateforme de recherches participatives de TISSAGE - Science avec et pour la société.

Références

1. AGIS-TA-TERRRE. *Mov'Around : l'interface d'étude des mobilités à Châteaubourg* <https://www.agistaterre.org/movaround-linterface-detude-des-mobilites-a-chateaubourg/>.
2. TELRAAM. <https://telraam.net/>.
3. Telraam API <https://telraam-api.net/>.
4. Comptages routiers : une association réalise ses propres mesures. *Maison de la Consommation et de l'Environnement*. <https://www.mce-info.org/comptages-routiers-une-association-realise-ses-propres-mesures/> (jan. 2022).
5. CHOLEZ, L.-A. & GUEDJ, L. Les projets routiers, contrevérités et carnage écologique. *Reporterre*. <https://reporterre.net/Les-projets-routiers-un-carnage-ecologique> (mai 2022).
6. VIDARD, M., CHOLEZ, L.-A., LEDENVIC, P. & CUBAUD, S. *Les projets routiers contestés : une aberration financière et écologique* France Inter, mai 2022. <https://www.radiofrance.fr/franceinter/podcasts/la-terre-au-carre/la-terre-au-carre-du-mercredi-18-mai-2022-7380402>.

Collecter et cartographier les données du bilan carbone d'un congrès

Chloé Friguet*

François Husson†

Résumé

Les manifestations scientifiques, comme de nombreux événements, suivent une tendance croissante à la prise de conscience et à l'action en faveur des enjeux environnementaux. Quantifier l'impact carbone participe à la prise de conscience mais nécessite de recueillir les habitudes de chaque participant ainsi que les trajets pour venir à ce rassemblement. Si de nombreux *packages R* permettent la construction et la mise en ligne d'un questionnaire grâce à une application R-shiny, d'autres permettent de visualiser les trajets sur une carte selon leur bilan carbone. Mais est-ce pour autant facile de faire tout cela ?

Pour ceux expérimentés avec R, la réponse est certainement "oui". Mais choisir les *packages* adaptés s'avère en pratique moins évident.

Cet exposé présente le bilan carbone des Rencontres R'24 ainsi que la mise en place du questionnaire, des traitements et de la cartographie.

On s'intéresse ici essentiellement aux impacts des trajets des participants des Rencontres R (200 personnes attendues), ainsi qu'à leurs habitudes alimentaires. Nous présentons les résultats des données récoltées, ce qui permet de sensibiliser les organisateurs du colloque ainsi que les participants à l'impact environnemental des activités scientifiques, et d'inclure des statistiques dans les bilans scientifiques à destination des partenaires institutionnels et sponsors de l'événement.

Mots-clefs (3 à 5) : Questionnaire – Dataviz – Cartographie – R-Shiny

Contexte : Le bilan carbone d'un événement scientifique Établir le bilan carbone d'un événement, et notamment d'une manifestation scientifique, s'inscrit dans une démarche globale de responsabilité environnementale. Il s'agit essentiellement de quantifier les émissions de gaz à effet de serre (notamment de CO_2), associées à l'organisation et à la tenue de l'événement, ainsi que d'autres paramètres comme la production de déchets ou la consommation alimentaire.

Dans le monde académique comme dans certaines entreprises, des actions sont mises en place pour sensibiliser leurs personnels aux impacts environnementaux de leurs pratiques personnelles et professionnelles. Des subventions sont attribués à ceux qui utilisent les transports en commun ou des mobilités douces pour aller au travail. Du point de vue professionnel dans le monde académique, les laboratoires de recherche et les universités sensibilisent à travers des procédures plus ou moins contraignantes : subventions uniquement pour des événements vertueux, quantification de l'impact carbone d'un déplacement professionnel, incitation à limiter ces déplacements, à ne pas prendre l'avion pour des trajets en France, à privilégier le train, etc.

Il existe ainsi plusieurs outils en ligne permettant d'évaluer l'impact carbone d'un déplacement professionnel, citons par exemple cette application pour calculer le bilan carbone d'un voyage <http://carbone.polytech.umontpellier.fr/simulation> et le Labo 1.5¹ qui propose des simulateurs pour les trajets du quotidien ou pour les missions, et les habitudes alimentaires.

De même, depuis 2023, la Société Française de Statistique² (SFdS) dispose d'une charte³ à laquelle est tenu de se conformer tout comité d'organisation d'événement porté par cette société savante, incluant

*Université de Bretagne sud, chloe.friguet@univ-ubs.fr

†Institut Agro Rennes, francois.husson@institut-agro.fr

1. **Labo 1point5** : collectif de membres du monde académique, de toutes disciplines et sur tout le territoire, partageant un objectif commun : mieux comprendre et réduire l'impact des activités de recherche scientifique sur l'environnement, en particulier sur le climat. Site web : <https://apps.labos1point5.org>

2. <https://www.sfds.asso.fr/>

3. https://www.sfds.asso.fr/d/doc-12005-d7bc5c51310f25ef149924163d8a4339-2023_1121_sfds_charte_d_ecoresponsabilite_vf.pdf

des recommandations comme privilégier le train pour les transports ou la mise en place de procédure contre le gaspillage alimentaire et la production de déchet. Les Rencontres R'24⁴ s'inscrivent pleinement dans cette initiative.

Nous présentons ci-après un outil permettant de collecter les données nécessaire pour établir le bilan carbone des participants des Rencontres R'24 sous forme de questionnaire, puis une application permettant de visualiser les résultats de façon dynamique. Ces 2 ont été développés avec R-shiny.

Un questionnaire pour récupérer les données des participants Nous avons développé une application R-shiny qui permet de recueillir de façon interactive des éléments sur le profil du participant, ses habitudes alimentaires et de trajet domicile-travail, et sur le déplacement effectué pour se rendre sur le lieu des Rencontres R'24. Les données sont enregistrées via un fichier *googlesheet*. Il existe plusieurs *packages* permettant la création de questionnaires avec R-shiny (par ex. *shinysurveys*, *quetzio*). Néanmoins, nous avons dû développer notre propre questionnaire afin de pouvoir prendre en compte la description du trajet du participant qui peut inclure plusieurs étapes et nécessite une liste dynamique de villes proposées pour éviter la gestion fastidieuse de ce champ textuel ensuite.

Une application R-shiny pour visualiser et restituer les résultats Comme de beaux graphes valent mieux qu'un long discours, nous avons choisi de restituer les résultats via une application R-shiny. La construction de cartes interactives permettant de visualiser les trajets et les impacts carbones de ces trajets aura nécessité l'utilisation de plusieurs packages de cartographie. Nous présenterons les choix que nous avons faits et les difficultés que nous avons pu rencontrer : les cartes permettent une restitution visuelle très informative et très facilement compréhensible même par des non experts en statistique ... mais la construction de celles-ci nécessite un peu de technicité !

4. Les **Rencontres R** ont pour objectif d'offrir à la communauté francophone un lieu d'échange et de partage d'idées sur l'usage du langage R. Elles s'adressent aussi bien aux débutants qu'aux utilisateurs confirmés et expérimentés issus de tous les secteurs d'activités. L'édition 2024 organisée à Vannes (site web : <https://rr2024.sciencesconf.org/>)

Estimation de quantiles conditionnels extrêmes : Package *Extremefit*

G. Durrieu * I. Gramma †

Résumé

La modélisation et l'estimation des valeurs extrêmes jouent un rôle important dans de nombreux domaines tel que la finance, la biologie, l'écologie, etc. Nous introduisons ici un package, *extremefit*, pour la détermination de quantiles extrêmes conditionnels. Nous présentons brièvement la méthode d'estimation des quantiles conditionnels extrêmes et nous expliquons les concepts statistiques afférents. Nous donnons un exemple d'utilisation du package *extremefit* pour la surveillance de la qualité des eaux littorales et la détection des effets du changement climatique.

Mots-clefs : Valeurs extrêmes – Modèle semi-paramétrique – Statistique – Écologie – Package R

1 Développement

1.1 Modèle et estimation

Le modèle utilisé dans le package est décrit dans [3], [4], [5] et [6]. Nous donnons ici un bref résumé. Soit $F_t(x) = P(X \leq x | T = t)$ la distribution conditionnelle d'une variable aléatoire X sachant que la covariable T prend la valeur $t \in [0, T_{max}]$ où $T_{max} > 0$. La distribution conditionnelle F_t est supposée avoir son support sur $[x_0, \infty)$, $x_0 > 0$ avec une densité f_t strictement positive. Le but principal est d'estimer ponctuellement la fonction de survie $S_t(x) = 1 - F_t(x)$ et les p -quantiles extrêmes $F_t^{-1}(p)$ pour $t \in [0, T_{max}]$.

L'approche utilisée est celle du “Peak-Over-Threshold” (voir section 5.3 dans [1]). Nous ajustons pour un certain seuil $\tau > x_0$ la fonction de répartition d'excès $F_{t,\tau}(x) = 1 - \frac{1-F_t(x)}{1-F_t(\tau)}$, $x \in [\tau, \infty)$ par la fonction de répartition de Pareto $G_{\tau,\theta}(x) = 1 - (x/\tau)^{-1/\theta}$, $x \in [\tau, \infty)$. Nous adoptons donc le modèle semi-paramétrique défini par :

$$F_{t,\tau,\theta}(x) = \begin{cases} F_t(x) & x \in [x_0, \tau], \\ 1 - (1 - F_t(\tau))(1 - G_{\tau,\theta}(x)) & x > \tau. \end{cases} \quad (1)$$

Ce modèle couvre essentiellement des observations dont la distribution est dans le domaine d'attraction de la loi de Fréchet. Selon la partie (i) du théorème 2.1 dans [1], une condition nécessaire et suffisante est $1 - F_{t,\tau}(\tau x) \rightarrow x^{-\frac{1}{\theta}}$ quand $\tau \rightarrow \infty$ ce qui implique que $G_{\tau,\theta}(x)$ peut être utilisée pour estimer $F_{t,\tau}(x)$ pour $x \in [\tau, \infty)$. Ceci justifie notre approche pour l'estimation de la fonction de répartition F_t sur l'intervalle $[x_0, \tau]$. Nous estimons F_t par la fonction de répartition empirique \hat{F}_t , alors qu'au-delà du seuil τ , nous utilisons la probabilité ajustée $(1 - \hat{F}_t(\tau)) (1 - G_{\tau,\hat{\theta}_t}(x))$ où $\hat{\theta}_t$ est l'estimateur de Hill pondéré décrit dans [1]. Le choix du seuil τ est un problème pour la qualité des estimations dans les modèles des valeurs extrêmes. Nous donnons un choix automatique du seuil en fonction des observations basé sur une suite de tests d'ajustement du modèle paramétrique proposé (voir [3] et [4]).

Dans l'article [5], nous visons à estimer les distributions de survie conditionnellement à une covariable dans le modèle de risques proportionnels de Cox, dans le cas où les probabilités estimées se

*LMBA Université Bretagne Sud, gilles.durrieu@univ-ubs.fr

†LMBA Université Bretagne Sud, ion.gramma@univ-ubs.fr

situent en dehors de la plage des données observées en utilisant la modélisation des valeurs extrêmes. Notre analyse est également basée sur la méthode “Peak-Over-Threshold” qui permet d'estimer la queue d'une distribution au-delà d'un seuil. Les applications de cette méthode peuvent être observées dans divers domaines tels que l'assurance, la biologie, les prévisions météorologiques et écologie [4].

1.2 Package *extremefit*

Le package *extremefit* [2] implémente la méthode ci-dessus pour estimer ponctuellement les quantiles extrêmes ainsi que les probabilités de survie.

Tout d'abord, les valeurs critiques associées aux tests d'ajustement pour le choix du seuil τ sont déterminées à l'aide d'une fonction contenue dans le package, pour un noyau choisi parmi plusieurs noyaux proposés ou définis par l'utilisateur. Ensuite, une fonction permet de choisir automatiquement le seuil τ ainsi que d'estimer le paramètre θ_t en fonction de la covariable $T = t$ dans le cas d'estimation des quantiles et probabilités conditionnelles. Les prédictions des quantiles et/ou des probabilités de survie sont disponibles via la fonction *predict* qui agit sur les objets résultant de la fonction précédente.

Certaines fonctions contenues dans le package permettent de calculer des intervalles de confiance ponctuels par bootstrap ou encore de visualiser les statistiques du test d'ajustement. Les méthodes présentées ci-dessus sont disponibles dans une version non-conditionnelle, i.e. pour des observations issues d'une fonction de répartition. Nous donnerons une application sur des données écologiques en précisant les apports de ce package.

Références

- [1] Jan Beirlant, Yuri Goegebeur, Johan Segers, and Jozef L Teugels. *Statistics of extremes : theory and applications*. John Wiley & Sons, 2006.
- [2] Gilles Durrieu, Ion Gramă, Kevin Jaunatre, Quang-Khoai Pham, and Jean-Marie Tricot. Extremefit : a package for extreme quantiles. *Journal of Statistical Software*, 87 :1–20, 2018.
- [3] Gilles Durrieu, Ion Gramă, Quang-Khoai Pham, and Jean-Marie Tricot. Nonparametric adaptive estimator of extreme conditional tail probabilities quantiles. *Extremes*, 18 :437–478, 2015.
- [4] Gilles Durrieu, Quang-Khoai Pham, Anne-Sophie Foltete, Valérie Maxime, Ion Gramă, Véronique Le Tilly, Helene Duval, Jean-Marie Tricot, Chiraz Ben Naceur, and Olivier Sire. Dynamic extreme values modeling and monitoring by means of sea shores water quality biomarkers and valvometry. *Environmental monitoring and assessment*, 188 :1–8, 2016.
- [5] Ion Gramă and Kévin Jaunâtre. Estimation of extreme survival probabilities with cox model. *Statistics*, 53(4) :807–838, 2019.
- [6] Ion Gramă, Jean-Marie Tricot, and Jean-François Petiot. Estimation of the extreme survival probabilities from censored data. *Buletinul Academiei de Științe a Moldovei. Matematica*, 74(1) :33–62, 2014.

Clustering sur données incomplètes avec clusterMI

Vincent Audigier *

26 mars 2024

Résumé

Nous nous intéressons à la classification non-supervisée d'observations incomplètes. Pour cela une nouvelle méthodologie basée sur l'imputation multiple est proposée. Celle-ci consiste en trois grandes étapes : l'imputation selon des modèles dédiés, la classification sur les données imputées, avec l'estimation de l'instabilité associée, et l'agrégation des résultats. Nous revenons sur ces différentes étapes et présentons comment les mettre en œuvre via le package clusterMI disponible sur le CRAN.

Mots-clefs : Clustering – Données manquantes – Imputation multiple

L'imputation multiple fait partie des stratégies classiques pour gérer le problème des données manquantes [Little and Rubin, 2002]. Telle que proposée historiquement, celle-ci consiste à imputer M fois le jeu de données selon un modèle, dit *modèle d'imputation*, à ajuster un modèle dit *d'analyse* sur chacun des tableaux imputés, puis à agréger les résultats selon les règles de Rubin. Ces règles consistent en l'agrégation à la fois en termes d'estimation ponctuelle des paramètres du modèle d'analyse, mais aussi en termes d'estimation des variances des estimateurs associés.

Toutefois, dans le contexte du clustering, une telle méthodologie ne peut pas être appliquée directement. En effet, l'imputation des données nécessite la prise en compte de la structure de groupes sur les individus. Les modèles d'imputation classiquement utilisés, tel que le modèle gaussien multivarié [Schafer, 1997] ne sont alors plus adaptés [Audigier et al., 2021]. Par ailleurs, les règles de Rubin ont été établies pour l'agrégation de coefficients numériques et ne permettent pas directement l'agrégation des partitions issues d'un clustering.

Cette présentation a pour objet de présenter différentes méthodes d'imputation pertinentes pour le clustering, ainsi qu'une façon d'agréger les résultats obtenus suite à l'analyse de chacun des tableaux, tant en termes de partition que d'instabilité associée. Concernant les modèles d'imputation, nous présenterons des approches *par modèle joint*, où une distribution explicite à l'ensemble des variables est effectuée, et des approches *séquentielles*, où seule la distribution conditionnelle de chaque variable est spécifiée. Ces dernières sont connues pour permettre un meilleur ajustement des données. Concernant l'agrégation des résultats, nous présenterons dans un premier temps comment évaluer une instabilité en clustering par bootstrap, comme proposé dans Fang and Wang [2012]. À partir de là, nous proposerons une façon d'agréger les différentes partitions obtenues, par factorisation matricielle non-négative, ainsi que les différentes mesures d'instabilité selon les règles détaillées dans Audigier and Niang [2022].

En résumé, la procédure proposée se présente ainsi :

Imputation Etant donné un jeu de données incomplet, générer M tableaux imputés selon une méthode d'imputation multiple pré-définie (par exemple Kim et al. [2014])

Analyse Pour m dans $\{1 \dots M\}$,

1. construire une partition Ψ_m à partir du $m^{\text{ème}}$ jeu imputé selon la méthode de clustering choisie
2. estimer V_m^{boot} l'instabilité associée par bootstrap

*CEDRIC-MSMDA, CNAM, Paris, vincent.audigier@cnam.fr

Agrégation

1. L'ensemble de partitions $(\Psi_m)_{1 \leq m \leq M}$ est agrégé en utilisant une factorisation matricielle non-négative [Li et al., 2007] consistant à rechercher la partition $\bar{\Psi}$ en K groupes telle que

$$\bar{\Psi} = \operatorname{argmin}_{\Psi} \sum_{m=1}^M \delta(\Psi, \Psi_m)$$

où $\delta(\Psi, \Psi_m)$ indique le nombre de désaccords entre les partitions Ψ and Ψ_m ¹.

2. Les mesures d'instabilité $(V_m^{boot})_{1 \leq m \leq M}$ sont agrégées selon :

$$T = \frac{1}{M} \sum_{m=1}^M V_m^{boot} + \frac{1}{M^2} \sum_{m=1}^M \sum_{m'=1}^M \delta(\Psi_m, \Psi_{m'}) / n^2$$

Cette méthodologie permet de gérer le problème de données manquantes en classification non-supervisée, que ce soit pour des approches probabilistes ou géométriques. Elle présente l'avantage de ne pas être sensible au problème de *label switching* et permet également d'avoir des nombres de groupes différents lors de la phase d'analyse. Enfin, elle offre la possibilité d'estimer le nombre de groupes dans le cadre d'un jeu de données incomplet, ceci en comparant les mesures d'instabilité des partitions agrégées pour différentes valeurs de K .

Le package associé **clusterMI** offre quatre méthodes d'imputation différentes, dont deux par modèle joint, et deux par approche séquentielle. Différentes méthodes de clustering sont pré-implémentées (k-means, pam, clara, agnes, mélange gaussien, fuzzy c-means) mais il est possible d'envisager d'autres méthodes. Le package offre aussi plusieurs outils de diagnostic afin d'apprecier l'ajustement du modèle d'imputation, la convergence et le choix des modèles d'imputation des approches séquentielles, le nombre de tableaux imputés M , ou encore le nombre de groupes K .

Références

- V. Audigier and N. Niang. Clustering with missing data : which equivalent for Rubin's rules ? *Advances in Data Analysis and Classification*, September 2022. doi : 10.1007/s11634-022-00519-1. URL <https://hal.science/hal-03766733>.
- V. Audigier, N. Niang, and M. Resche-Rigon. Clustering with missing data : which imputation model for which cluster analysis method ?, 2021.
- Y. Fang and J. Wang. Selection of the number of clusters via the bootstrap method. *Comput. Stat. Data Anal.*, 56(3) :468–477, 2012. ISSN 0167-9473. doi : 10.1016/j.csda.2011.09.003. URL <https://doi.org/10.1016/j.csda.2011.09.003>.
- H. J. Kim, J. P. Reiter, Q. Wang, L. H. Cox, and A.F. Karr. Multiple imputation of missing or faulty values under linear constraints. *Journal of Business & Economic Statistics*, 32(3) :375–386, 2014. doi : 10.1080/07350015.2014.885435. URL <https://doi.org/10.1080/07350015.2014.885435>.
- T. Li, C. Ding, and M. I. Jordan. Solving consensus and semi-supervised clustering problems using nonnegative matrix factorization. In *Proceedings of the 2007 Seventh IEEE International Conference on Data Mining*, ICDM '07, page 577–582, USA, 2007. IEEE Computer Society. ISBN 0769530184. doi : 10.1109/ICDM.2007.98.
- R. Little and D. Rubin. *Statistical Analysis with Missing Data*. Wiley series in probability and statistics, New-York, 2002.
- J. Schafer. *Analysis of Incomplete Multivariate Data*. Chapman & Hall/CRC, London, 1997.

1. $\delta(\Psi, \Psi') = \sum_{(i,i')} \delta_{ii'}$ avec $\delta_{ii'} = 1$ si les individus i et i' appartiennent au même cluster dans une partition et non dans l'autre et $\delta_{ii'} = 0$ sinon

Créer son propre package d'extension `{recipes}`: retour d'expérience de `{scimo}`

Antoine Bichat* Julie Aubert†

Résumé

L'avènement des technologies de séquençage à haut débit a entraîné une augmentation massive de la production de données omiques : génomiques, transcriptomiques, protéomiques, métagénomiques, et bien plus encore.

Pour explorer et analyser efficacement les données omiques, des étapes de prétraitement adaptées sont requises, comme la normalisation, la sélection et l'agrégation de variables (Perez-Riverol et al. (2017)). Cependant, ces méthodes spécifiques ne sont pas disponibles nativement dans le package `{recipes}` (Kuhn, Wickham, and Hvitfeldt (2024)). Nous avons alors développé un package d'extension, `{scimo}`, conçu pour intégrer ces étapes dans l'écosystème `{tidymodels}` (Kuhn and Wickham (2020)).

`{scimo}` propose une série d'étapes de prétraitement adaptées à l'analyse des données omiques, tout en restant adaptable à d'autres types de données. Il est disponible à cette adresse : <https://github.com/abichat/scimo>

Mots-clés : Tidymodels - Package - Statistique - Biostatistique - Données omiques

Développement

Au cours de cette présentation, nous introduirons les fonctions principales du package `{scimo}` et expliquerons comment créer son propre package d'extension `{recipes}`. Nous insisterons sur les difficultés rencontrées au cours du développement.

Références

- Kuhn, Max, and Hadley Wickham. 2020. *Tidymodels: A Collection of Packages for Modeling and Machine Learning Using Tidyverse Principles*. <https://www.tidymodels.org>.
- Kuhn, Max, Hadley Wickham, and Emil Hvitfeldt. 2024. *Recipes: Preprocessing and Feature Engineering Steps for Modeling*. <https://CRAN.R-project.org/package=recipes>.
- Perez-Riverol, Yasset, Max Kuhn, Juan Antonio Vizcaino, Marc-Phillip Hitz, and Enrique Audain. 2017. “Accurate and Fast Feature Selection Workflow for High-Dimensional Omics Data.” *PLoS One* 12 (12): e0189875.

*Les Laboratoires Servier, antoine.bichat@servier.com

†MIA Paris-Saclay, julie.aubert@agroparistech.fr

Smooth testing and clustering of copulas

Yves Ismaël Ngounou Bakam* and Denys Pommeret†

Résumé

A smooth test to simultaneously compare K copulas, where $K \geq 2$, is proposed. The K observed populations can be paired. The test statistic is based on the differences between moment sequences, called copula coefficients. These coefficients characterize the copulas, even in cases where the copula densities may not exist. The procedure involves a two-step data-driven procedure. In the initial step, the most significantly different coefficients are selected for all pairs of populations. The subsequent step utilizes these coefficients to identify populations that exhibit significant differences.

We then use this test to automatically construct clusters consisting of populations with significantly similar dependence structures. The procedure is data-driven and relies on the asymptotic level of the test.

We apply our test methodology and clustering algorithm, implemented in the **Kcop** R package [Bakam and Pommeret, 2022a] to real datasets.

Mots-clefs (3 à 5) : Copula coefficients – Data-driven – Smooth test – Legendre polynomials – Nonparametric clustering – **Kcop** R package.

Context

Investigating dependence structures between multidimensional variables in various domains can provide valuable insights and inform decision-making processes. Below are a few applications that can be considered :

- Finance : recognize and managing financial risks, enhancing portfolio diversification, evaluating derivative products, and evaluating financial contagion. It provides valuable insights for investors, portfolio managers, and financial regulators in a constantly evolving market environment.
- Marketing : understanding the dependence between product sales in different retail stores can help companies optimize their inventory management, pricing strategies, and marketing efforts. For instance, if certain products tend to sell together in one store but not in another, it may suggest differences in consumer preferences or market demand, leading to tailored marketing campaigns or adjustments in product placement.
- Economics : analyzing the dependence between income and consumption in different countries can offer insights into consumer behavior, economic trends, and policy effectiveness. It can help policymakers design targeted interventions to stimulate economic growth, reduce income inequality, or encourage responsible consumption habits.
- Accounting : investigating the dependence between reported items on corporate balance sheets across different countries can aid in assessing financial risk, compliance with accounting standards, and potential exposure to economic shocks. It can inform investors, regulators, and corporate decision-makers about the financial health and resilience of multinational corporations operating in diverse environments.

Moreover, non-parametric classification of these multidimensional variables offers a structured framework for investigating and interpreting various dependence structures within the data. This facilitates deeper insights into the interrelationships among variables, enhances model refinement, and

*Univ Rennes, Ensai, CNRS, CREST - UMR 9194, F-35000 Rennes, France , yves.ngounou@ensai.fr

†Aix-Marseille University, CNRS, Centrale Marseille, I2M, Campus de Luminy, 13288 Marseille cedex 9, France, denys.pommeret@univ-amu.fr

enables more informed decision-making. It underscores the importance of robust statistical methods, such as copula-based analysis, in extracting meaningful insights from complex datasets across diverse fields.

The purpose of this presentation (Ngounou et al., 2024) is to develop, initially a new method for simultaneously comparing copulas based on data Ngounou Bakam and Pommeret [2024]. We consider K continuous random vectors, namely

$$\mathbf{X}^{(1)} = (X_1^{(1)}, \dots, X_p^{(1)}), \dots, \mathbf{X}^{(K)} = (X_1^{(K)}, \dots, X_p^{(K)}),$$

with joint cdf $\mathbf{F}^{(1)}, \dots, \mathbf{F}^{(K)}$, and with associated copulas C_1, \dots, C_K , respectively. Assume that we observe K iid samples from $X^{(1)}, \dots, X^{(K)}$, possibly paired, denoted by

$$(X_{i,1}^{(1)}, \dots, X_{i,p}^{(1)})_{i=1,\dots,n_1}, \dots, (X_{i,1}^{(K)}, \dots, X_{i,p}^{(K)})_{i=1,\dots,n_K}.$$

where we assume that for all $1 \leq \ell < m \leq K$, $\min(n_\ell, n_m) \rightarrow \infty$, and

$$n_\ell / (n_\ell + n_m) \rightarrow a_{\ell,m}, \text{ with } 0 < a_{\ell,m} < \infty.$$

We consider the problem of testing the equality

$$H_0 : C_1 = \dots = C_K,$$

against H_1 : there exist $1 \leq k \neq k' \leq K$ such that $C_k \neq C_{k'}$.

The test statistic is based on the differences between moment sequences, called copula coefficients. These coefficients characterize the copulas, even in cases where the copula densities may not exist [Ngounou Bakam and Pommeret, 2024].

In the second part, we present an adapted version of the previous testing procedure to establish a data-driven method for clustering K populations into N subgroups, each characterized by a common dependence structure. The unknown number N of clusters will be automatically determined by the procedure and validated through our testing method. The full details is proposed in Bakam and Pommeret [2022b].

Package Kcop

The `KcopTest` function conducts a nonparametric smooth test to simultaneously compare K (where $K > 1$) copulas, while the `KcopClust` function executes a data-driven clustering procedure to group K multivariate populations of varying sizes into N subgroups distinguished by a shared dependence structure. The number N of clusters is unknown and determined automatically by our approach.

For more details, examples, and additional information, please refer to Bakam and Pommeret [2022a].

A comprehensive application involving the financial returns of companies listed in the S&P 500 index will be provided.

Références

Yves Ismaël Ngounou Bakam and Denys Pommeret. *Kcop : Smooth Test for Equality of Copula and Clustering Multivariate*, 2022a. URL <https://CRAN.R-project.org/package=Kcop>. R package version 1.0.0.

Yves Ismaël Ngounou Bakam and Denys Pommeret. Non-parametric clustering of multivariate populations with arbitrary sizes. *arXiv preprint arXiv:2211.06338*, 2022b.

Yves Ismaël Ngounou Bakam and Denys Pommeret. Smooth test for equality of copulas. *Electronic Journal of Statistics*, 18(1) :895–941, 2024.

SK8 : Pour des applications shiny qui se déploient comment sur des roulettes

Elise Maigné*

Résumé (max 300 mots)

Là où le développement d'applications est rendu aujourd'hui accessible à toute personne faisant du R via Shiny, leur hébergement reste quant à lui souvent plus compliqué. Selon la solution envisagée et l'équipe qui l'entoure il faudra sûrement au développeur Shiny des compétences supplémentaires, un peu d'admin sys par ci, un peu de dev ops par là et un ensemble de bonnes pratiques de développement qui ne sont pas toujours connues, rendant des applications difficilement maintenables à long terme.

A INRAE nous avons ouvert un service d'hébergement d'applications Shiny qui se veut facile d'utilisation dans lequel nous proposons une prise en charge des aspects reproductibilité et gestion de l'environnement logiciel. Après un rapide tour d'horizon des solutions d'hébergement possibles et une présentation du service SK8, je vous expliquerai dans cet exposé ce que nous avons mis en place - et qui peut être réutilisé plus largement - pour accompagner nos utilisateurs pour rendre leurs applications Shiny robustes, reproductibles... destinées à se déployer comme sur des roulettes !

Mots-clefs (3 à 5) : R – Shiny – CI/CD – Docker - Kubernetes - Reproductibilité – Applications – Déploiement

* INRAE, MIAT, elise.maigne@inrae.fr

Explorer et comparer des cartes de zones climatiques locales avec le paquet lczexplore

Matthieu Gousseff* Jérémie Bernard † Erwan Bocher* Elisabeth Le Saux
Wiederhold+ Baptiste Alglave+ François Leconteø

Résumé (max 300 mots)

Le changement climatique est un enjeu majeur, toujours plus important pour les aménageurs, les urbanistes, les élus, notamment en raison des phénomènes de surchauffe qu'il induit sur les territoires et qui impactent la mortalité et la santé en général. La classification en zones climatiques permet d'appréhender ce risque en établissant une relation entre la topographie d'un territoire, l'organisation de ses éléments structurants (bâtiments, végétation....) et l'intensité de cet îlot de chaleur. Les Zones Climatiques Locales (LCZ) proposées par [@stewart2012] sont devenues, au cours des dernières décennies, une référence pour faire ce lien entre géographie urbaine et risque de surchauffe. Il existe plusieurs méthodes pour produire des cartes de LCZ qui, selon l'algorithme ou les données utilisées ne donnent pas exactement le même résultat (par exemple GeoClimate ou Wudapt).

Le paquet lczexplore permet de comparer facilement des paires de cartes produites sur un même territoire. Il fournit une carte d'accord des classifications, une statistique d'accord dérivée du Kappa de Cohen, et une matrice de confusion pour quantifier et qualifier l'accord entre les classifications. Une analyse de sensibilité de l'accord entre deux cartes en fonction d'un indice de confiance sur l'attribution des classes est également proposée. Enfin, des facilités de regroupement de classes permettent d'explorer et de généraliser les comparaisons à toute paire de classifications sur le même territoire. Cette application est disponible sous forme d'un paquet R ouvert.

Mots-clefs (3 à 5) : Information Géographique - Statistique Spatiale - Package - sf - Dataviz

Développement

L'existence de plusieurs méthodes de zones climatiques locales fait émerger le besoin d'un outil permettant d'obtenir, rapidement et facilement :

- 1) Des cartes depuis les formats et conventions largement partagées (raster et vecteur).
- 2) Une mesure statistique d'accord entre deux cartes produites sur le même territoire avec deux méthodes différentes (pseudo Kappa de Cohen).
- 3) Une représentation spatiale pour une évaluation visuelle rapide des zones en désaccord.
- 4) Une matrice de confusion pour voir comment les niveaux d'une classification se ventilent dans ceux de l'autre classification.
- 5) Une analyse de sensibilité de l'accord entre les classifications selon la confiance accordée à la classe affectée à chaque unité spatiale.

* CNRS Lab-STICC, matthieu.gousseff@univ-ubs.fr, † CNRM, jeremy.bernard@zaclys.net

+ Université Bretagne-Sud, elisabeth.le-saux@univ-ubs.fr, ø Université de Lorraine, francois.leconte@univ-lorraine.fr

- 6) Le regroupement de plusieurs types de LCZ pour faire varier la granularité de l'analyse (par exemple regrouper en zones urbaines et zones végétalisées)
- 7) La comparaison de toute paire de classifications sur des polygones géoréférencés.

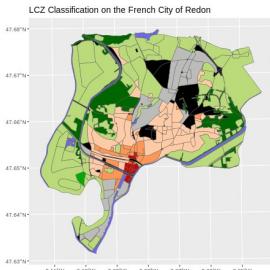


Figure 1: Zones Climatiques Locales sur le territoire de Redon

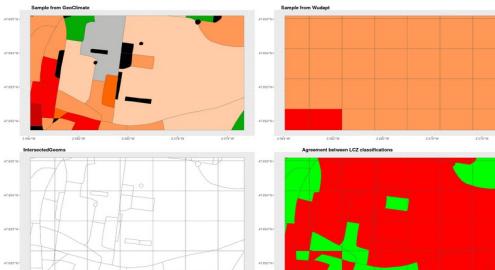


Figure 2: Intersection des géométries issues d'une carte vectoriel et d'une carte raster, représentation de l'accord / désaccord

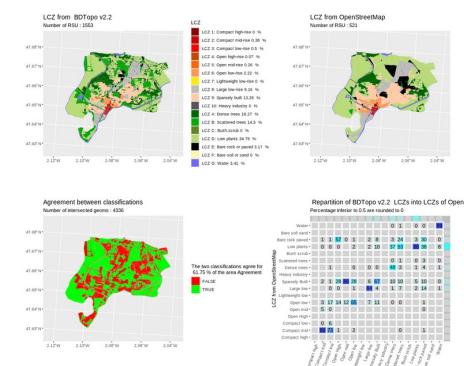


Figure 3: Comparaison des LCZ produites en se basant sur des données IGN et sur des données OpenStreetMap

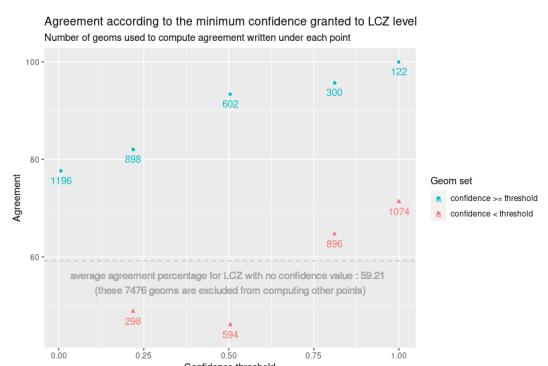


Figure 4: Analyse de sensibilité : les désaccords entre les cartes proviennent-ils des géométries dont la classe est douteuse ?

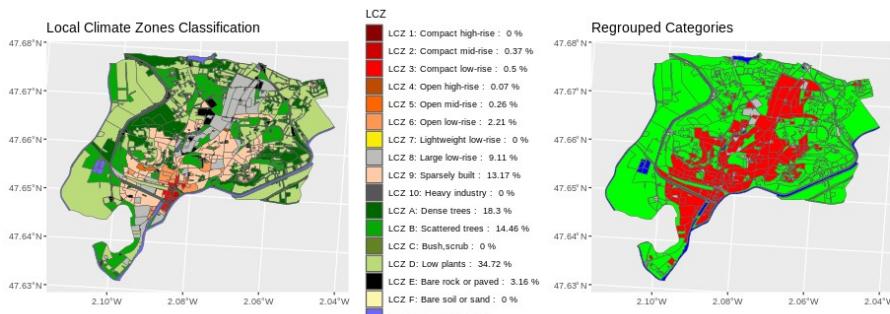


Figure 5: Un exemple de regroupement de LCZ

Le paquet lczexplore s'appuie sur les paquets de référence sf et terra [@bivand2022] et peut être installé depuis le dépôt github de notre équipe :
`devtools::install_github("orbisgis/lczexplore")`.

Références

- Stewart, I.D., Oke, T.R., Local Climate Zones for Urban Temperature Studies, 2012, Bulletin of the American Meteorological Society, DOI : 10.1175/BAMS-D-11-00019.1
- Bivand, Roger, R Packages for Analyzing Spatial Data: A Comparative Case Study with Areal Data, 2022, Geographical Analysis, DOI : <https://doi.org/10.1111/gean.12319>

Des applications Shiny qui facilitent la vie !

Dechaux T.* , Jean-Louis U.* , Legris M.*

Résumé (max 300 mots)

Les applications Shiny offrent une interface intuitive, permettant à un utilisateur de tirer pleinement parti des fonctionnalités avancées offertes par R sans avoir à maîtriser la programmation. Une aubaine pour nos collègues qui traitent de la donnée et qui sont confrontés à la barrière de l'apprentissage de ce langage. DATA'STAT a développé pour eux des applications Shiny qui répondent à leurs besoins de traitement de données. De la simple mise en forme des données, à leur exploration, en passant par des sujets plus spécifiques comme l'élaboration de dispositifs expérimentaux, ces outils couvrent un large éventail de besoins pour faciliter des tâches quotidiennes.

Mots-clefs (3 à 5) : Shiny – Ingénierie – Data

Développement

R est un logiciel largement utilisé dans le domaine des statistiques, de l'analyse de données et de la visualisation graphique. Cependant, sa syntaxe peut s'avérer complexe à maîtriser nécessitant un investissement en temps important. Cela peut entraîner certaines difficultés au niveau de l'apprentissage pour nos collègues qui traitent de la donnée avec Excel. Néanmoins, ils sont confrontés aux limites de ce logiciel et n'ont pas d'alternative pour réaliser leurs analyses. Heureusement pour eux, il existe des solutions pour rendre R plus accessible

Depuis 2 ans, DATA'STAT développe des applications RShiny dont le but est de simplifier l'exploration, la manipulation et l'analyse de données. Elles reposent sur la combinaison de packages existants en R comme *{dplyr}* pour la manipulation de données, ou bien *{FactoMineR}* pour l'ACP, et de la création d'interface web avec Rshiny. Elles permettent ainsi de répondre aux besoins de nos collègues en traitement de données sans avoir à maîtriser R et de rendre toutes ces fonctionnalités plus accessibles.

Ces applications permettent de couvrir un large éventail de besoins en matière de manipulation et d'analyse de données. A ce jour, six applications sont disponibles en interne :

- ✓ **Weibull** pour ajuster de courbes d'acidification du lait par loi de Weibull à 3 paramètres
- ✓ **Visualisation images 3D** pour visualiser et trier des images 3D au format obj
- ✓ **Jointure** pour réaliser tout type de jointure entre deux fichiers
- ✓ **ACP** pour réaliser une Analyse en Composantes Principales

* Institut de l'élevage, 75595 Paris

- ✓ **XPBloc** pour créer des blocs d'animaux dans un essai expérimental
- ✓ **CalculEffectif** pour estimer le nombre d'individus à inclure dans un essai expérimental

Par ailleurs, cinq applications supplémentaires sont en développement permettant entre autres d'empiler des fichiers compris dans un répertoire ou de réaliser des statistiques descriptives et des visualisations graphiques des données.

Grâce à ces applications, nos collègues peuvent réaliser des traitements de données simples en toute autonomie, sans avoir à manipuler R. De plus, certains de ces outils peuvent avoir des vertus pédagogiques en permettant à l'utilisateur de comprendre la nécessité de bien préparer ses données avant une analyse (jointure, pivot) ou d'assimiler l'importance de préparer rigoureusement un essai expérimental (Blocs et Calculs effectif).

Pour remplir ces objectifs, il est indispensable de rendre ces applications accessibles. Pour faciliter leur utilisation, le développement s'articule autour d'une interface commune composée de 5 rubriques :

- Une page d'accueil qui présente les objectifs de l'application
- Le contenu de l'application
- Une FAQ qui recense les réponses aux principales questions des utilisateurs
- Une page de contact permettant de contacter le développeur en cas de besoin
- Une page d'informations sur les dernières mises à jour de l'application et les références et bibliographies

Ces applications sont déployées sur un serveur Shiny interne. Une page d'accueil (Figure 1) recensant les applications disponibles et celles à venir a été développée et permet un accès simplifié à ces outils. Pour faciliter la diffusion des informations autour de ces applications, une newsletter interne a été mise en place cette année. Accessible sur inscription, elle permet à nos collègues de s'informer sur les mises à jour et des nouveautés à venir.

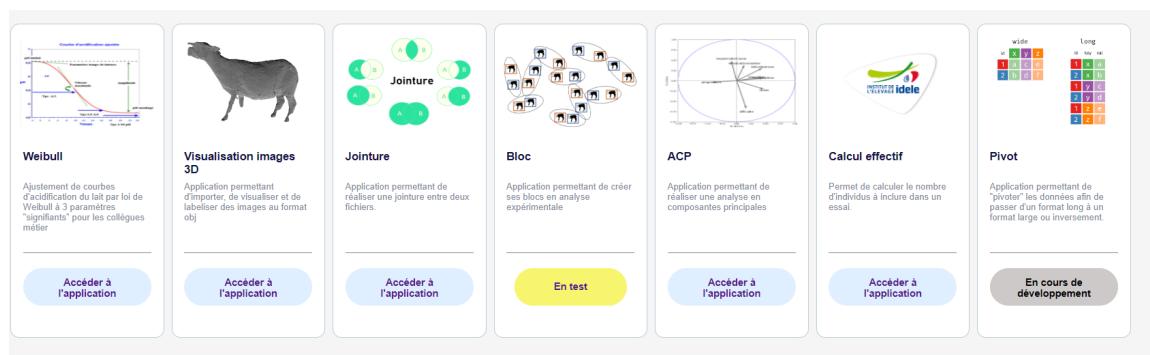


Figure 1- Extrait de la page d'accueil des applications

Par conséquent, en plus des applications, c'est tout une organisation qui se construit autour de Rshiny pour simplifier la vie de nos collègues dans le traitement de leurs données.

Application Shiny XPBloc - Création de blocs en expérimentation

M. Legris * T. Dechaux †

Résumé (max 300 mots)

Dans les dispositifs expérimentaux, on souhaite étudier l'effet d'un traitement sur une variable d'intérêt. L'objectif : mettre en évidence une relation de causalité. Pour ce faire, le montage d'une expérimentation demande une approche minutieuse comprenant plusieurs étapes. Au même titre que le choix du dispositif expérimental ou du calcul d'effectif, la mise en lot (i.e. l'affectation des animaux à un lot de l'essai) est une étape fondamentale en expérimentation animale. C'est elle qui permet de s'assurer de la comparabilité des lots au démarrage de l'essai en prenant en compte les facteurs de confusion.

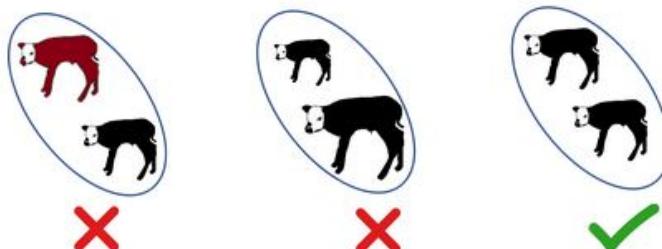
L'application RShiny XPBloc, développée par Data'Stat, permet d'accompagner les salariés de l'Institut de l'Elevage et les techniciens de ferme expérimentale dans la constitution de leurs lots grâce à la méthode des blocs expérimentaux. De la sélection des critères de mise en lot à la randomisation, en passant par la création pas à pas de leur blocs, l'utilisateur navigue dans une interface graphique et interactive au service de son expérimentation.

Mots-clefs (3 à 5) : Biostatistique - Expérimentation - Shiny

Les blocs en expérimentation

Pour s'assurer que les différences observées à la fin de l'essai sont bien la conséquence des traitements, il faut que les lots d'animaux recevant les traitements se ressemblent le plus possible au début de l'essai, de sorte que seuls les traitements puissent expliquer ces différences de fin d'essai. La mise en lot est donc une étape fondamentale en expérimentation animale. La méthode des blocs permet de répondre à cet objectif (Burger et al. 2021, Kernan et al. 1999, Fisher Box 1980).

Un bloc est un ensemble d'unités expérimentales (ex : animaux, cases, parcelles...) qui se ressemblent sur un ensemble de critères définis, les critères de mise en lot. Les unités expérimentales du même bloc peuvent donc être considérées comme des « jumeaux » ou des « clones » sur ces critères.



Au sein de chaque bloc, les animaux recevront un traitement différent, sélectionné de manière aléatoire. Ainsi, la seule chose qui différenciera les animaux d'un bloc c'est le traitement.

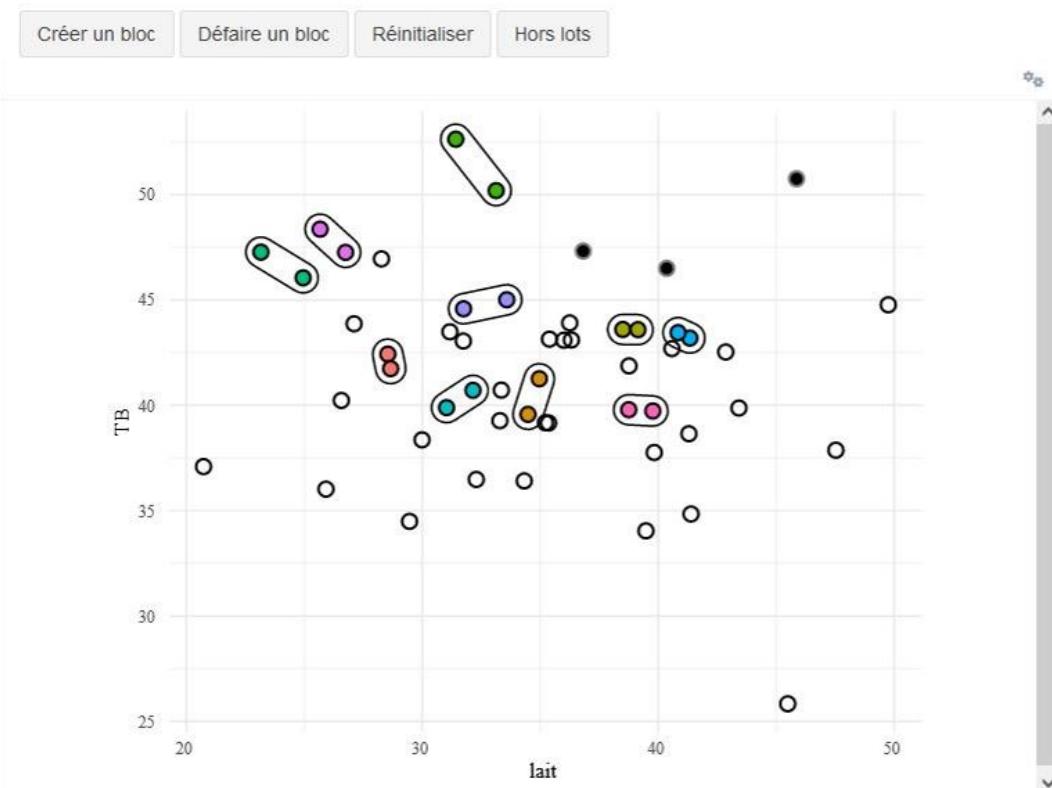
*Institut de l'élevage, maxime.legris@idele.fr

†Institut de l'élevage, terence.dechaux@idele.fr

L'application interactive XPBloc

L'application Shiny XPBloc a été développée par Maxime Legris (service Data'Stat) pour accompagner les expérimentateurs dans la création de blocs expérimentaux grâce à des méthodes graphiques et interactives de représentation des données pré-expérimentales. Les packages utilisés ont permis un fort niveau d'interactivité et ont contribué à l'ergonomie de l'application. Après avoir importé les données simplement (module d'import du package *datamods*), puis sélectionné les critères de mises en lot grâce (*bucket_list* du package *sortable*), l'utilisateur pourra associer des animaux en fonction de leur proximité sur les données pré-expérimentales grâce à des nuages de points interactifs (package *ggiraph*) et à des tableaux interactifs (package *DT*).

Constitution des blocs



L'application permet également de vérifier la qualité des blocs constitués et de procéder à la randomisation, étape indispensable à toute bonne expérimentation. La randomisation affecte un traitement à chaque animal de manière aléatoire, ce qui permet d'équilibrer les facteurs de confusion non contrôlés (Burger et al. 2021, Kernan et al. 1999).

Références

- J. Fisher Box, “R. A. Fisher and the Design of Experiments, 1922-1926”. *The American Statistician.*, 34 (1980), 1–7
B. Burger, M. Vaudel, and H. Barsnes, Importance of Block Randomization When Designing Proteomics Experiments, *J. Proteome Res.* 20 (2021), 1, 122–128
W.N. Kernan, C.M. Viscoli, R.W. Makuch, L.M. Brass, R.I. Horwitz, Stratified randomization for clinical trials, *J Clin Epidemiol.* 52 (1999), 19–26

Utilisation du package {flexdashboard} pour le contrôle des données de biologie dans un entrepôt de données de santé

Morgane Pierre-Jean 1* Guillaume Bouzillé 2†

Résumé

Les résultats des laboratoires hospitaliers constituent une source de données importante dans les entrepôts de données de santé. Pour garantir la comparabilité entre les établissements et pour pouvoir être utilisés dans des études de recherche, les résultats doivent être interopérables. Des événements pouvant perturber la distribution des valeurs pour un même dosage biologique peuvent se produire au cours du temps. Par exemple, de nouveaux équipements peuvent être ajoutés au pipeline d'analyse, un automate peut être remplacé, les formules peuvent évoluer en raison de nouvelles connaissances scientifiques et des terminologies existantes peuvent être adoptées. Nous avons automatisé la production de tableaux de bord pour surveiller ces événements et la qualité des données.

Nous avons utilisé le package flexdashboard pour produire des pages html pour chacun des dosages considérés, des méthodes de détection de ruptures automatiques telles que PELT et KernSeg ont été utilisées pour la détection d'événements. Pour un code LOINC donné, nous créons un tableau de bord qui synthétise le nombre de codes locaux, et le nombre de patients (par sexe, âge et service hospitalier) associés au code. Enfin, le tableau de bord permet de visualiser les événements temporels qui perturbent la distribution du signal. Malgré la quantité de données dans l'entrepôt du CHU de Rennes, la génération d'un dashboard est simple et rapide. Les dashboards permettent de vérifier que les valeurs d'un dosage biologique au cours du temps sont stables. Ils permettent également de vérifier si les données de l'hôpital ont bien été remontées et de détecter facilement les problèmes techniques. Un exemple de dashboard produit pour un dosage est disponible à cette adresse : <https://mpierrejean.github.io/MIE2024/>.

Mots-clefs : Data – Santé – Reporting – Monitorage – **flexdahboard**

1 Contexte

L'entrepôt de données de santé Madec et al. [2019] du CHU de Rennes permet de réaliser un grand nombre d'études de recherche à l'aide des données produites dans les unités de soin. Les données de biologie constituent une grande source de données pour les entrepôts de données de santé (EDS). Cependant, les données sont dépendantes des technologies de production comme les automates, les nouvelles formules, les changements de réactifs... Ces changements n'affectent pas les soins délivrés aux patients, mais lorsque ces données sont réutilisées en recherche, ils peuvent avoir un impact significatif. Le but de ce projet est de produire très rapidement un monitoring de la qualité des données biologiques disponibles dans l'entrepôt.

2 Méthodes

Nous avons utilisé le package flexdashboard Aden-Buie et al. [2023] qui nous a permis de générer des pages html interactives pour parcourir différents onglets qui correspondent à différents types d'indicateurs de qualité. La génération est automatisée à l'aide de paramètres : paramètre du code biologique à explorer et connexion à notre base de données. Nous avons également utilisé la librairie plotly Sievert [2020] afin de rendre certains graphiques interactifs.

*Univ Rennes, CHU Rennes, INSERM, LTSI-UM R 1099, morgane.pierre-jean@chu-rennes.fr

†Univ Rennes, CHU Rennes, INSERM, LTSI-UM R 1099, guillaume.bouzille@chu-rennes.fr

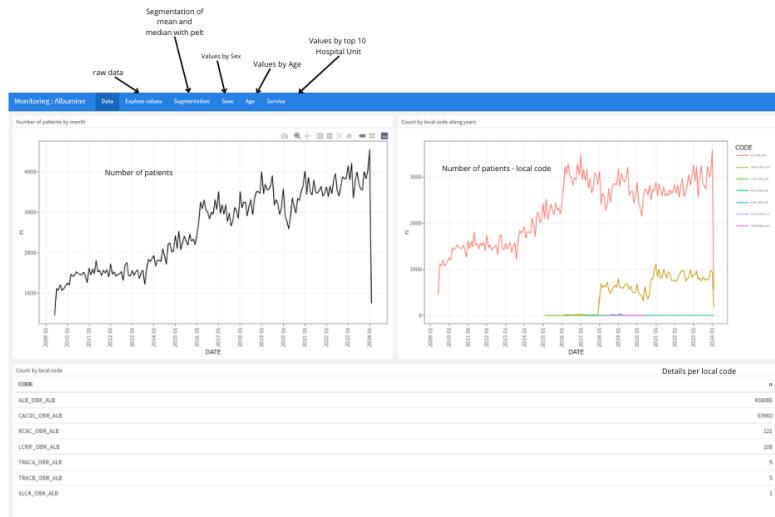


FIGURE 1 – Dashboard annoté

3 Résultats

Le workflow de génération contient, un fichier template Rmd pour le dashboard et un fichier R qui permet de générer autant de dashboards que de dosages biologiques que l'on veut explorer.

Le dashboard contient donc plusieurs onglets afin de pouvoir explorer le dosage à différents niveaux et détecter de potentielles anomalies. On peut donc explorer le nombre de valeurs produites au cours du temps, les valeurs brutes, la distribution des valeurs par année, l'évolution de la moyenne et de la médiane, une exploration est possible par sexe, par âge et par unité de soin 1. Un exemple de dashboard produit pour un dosage est disponible à cette adresse : <https://mpierrejean.github.io/MIE2024/>.

Grâce à ces dashboards, nous avons pu facilement détecter plusieurs types d'anomalies. Les anomalies au niveau des données brutes générées sont souvent des changements de technique de dosage ou des changements de machine. Les anomalies au niveau du nombre de données générées peuvent être dues à un problème technique ou un changement de pratique (le dosage n'est plus prescrit automatiquement).

4 Conclusion

La génération de dashboards avec les libraries R disponibles est simple et rapide. Nous avons pour projet de déployer ces dashboards en production afin de détecter des problèmes de qualité dans les données au fil de l'eau.

Références

Gerrick Aden-Buie, Carson Sievert, Richard Iannone, JJ Allaire, and Barbara Borges. *flexdashboard : R Markdown Format for Flexible Dashboards*, 2023. <https://pkgs.rstudio.com/flexdashboard/>, <https://github.com/rstudio/flexdashboard/>.

Julia Madec, Guillaume Bouzillé, Christine Riou, Pascal Van Hille, Christian Merour, Marie-Lisen Artigny, Denis Delamarre, Veronique Raimbert, Pierre Lemordant, and Marc Cuggia. ehop clinical data warehouse : From a prototype to the creation of an inter-regional clinical data centers network. *Studies in health technology and informatics*, 264 :1536–1537, 2019.

Carson Sievert. *Interactive Web-Based Data Visualization with R, plotly, and shiny*. Chapman and Hall/CRC, 2020. ISBN 9781138331457. URL <https://plotly-r.com>.

easy16S : une application Shiny pour explorer ses données métagénomiques

Cédric Midoux 1*, 2†, 3‡, §

Mahendra Mariadassou 2, 3¶

Résumé

L'analyse des données du microbiome est devenue un atout majeur pour l'étude de la diversité et de la dynamique microbienne dans divers domaines de la biologie. Après une étape bioinformatique permettant le passage des données de séquences (FASTQ) à une matrice d'abondance des espèces microbiennes (OTU : Operational Taxonomic Unit) et une table d'annotation taxonomique, une étape d'analyse statistique est nécessaire pour répondre aux questions biologiques (*étude de la diversité, comparaison d'abondance en fonction des conditions expérimentales, évolution d'un écosystème, ...*). Il existe une demande croissante d'outils interactifs conviviaux permettant aux chercheurs d'analyser et de visualiser leurs données de manière autonome, sans dépendre d'un biostatisticien ou sans avoir besoin d'acquérir des compétences en programmation R.

Nous présentons ici **easy16S**, un package R et une application Shiny visant à faciliter l'analyse des données du microbiome. Cette application s'appuie sur un objet phyloseq (McMurdie and Holmes (2013)) constitué d'une matrice d'abondance des OTU, d'un data.frame de métadonnées des échantillons et d'une matrice d'affiliation taxonomique des OTU. Cette application est intuitive et est orientée pour la visualisation des variables d'intérêt sur la structuration des communautés microbiennes.

Après le chargement de ses données brutes, l'utilisateur peut facilement les prétraiiter. Cela inclut des options telles que le filtrage des échantillons et des taxons, la modification du tableau d'affiliations, la raréfaction ou la transformation de la matrice de comptage.

L'utilisateur peut ensuite réaliser diverses analyses, telles que :

- Tableaux constituant l'objet phyloseq
- Visualisation des métadonnées grâce à **esquisse** (Meyer and Perrier (2024))
- Barplot de composition
- Courbes de raréfaction
- Heatmap d'abondance
- Richesse au sein d'un échantillon
- Dissimilarité entre les échantillons
- *MultiDimensional Scaling*
- Analyse d'abondance différentielle

L'application s'adresse à des utilisateurs débutants souhaitant mener leurs analyses sans compétence technique, à des utilisateurs experts souhaitant visualiser rapidement les patterns et tendances présents au sein de ses données, ainsi qu'aux apprenants lors de sessions de formation.

Mots-clefs : Bioinformatique - Métagénomique - Visualisation - Shiny

Développement

Le package est versionné sur la ForgeMIA d'INRAE (<https://forgemia.inra.fr/migale/easy16s>). Le framework **golem** (Fay et al. (2023)) a été utilisé pour faciliter le développement. Une documentation est disponible

*Université Paris-Saclay, INRAE, PROSE, 92761, Antony, France

†Université Paris-Saclay, INRAE, MaIAGE, 78350, Jouy-en-Josas, France

‡Université Paris-Saclay, INRAE, BioinfOmics, MIGALE bioinformatics facility, 78350, Jouy-en-Josas, France

§cedric.midoux@inrae.fr

¶mahendra.mariadassou@inrae.fr

grâce à `pkgdown` et via intégration continue : <https://easy16s.migale.inrae.fr/>. De plus, une image docker de l’application est déployée grâce à l’intégration continue de GitLab. Enfin, une instance de l’application est disponible en accès libre et est hébergée grâce à shinyproxy sur <https://shiny.migale.inrae.fr/app/easy16S>.

L’application a été soumise à *Journal of Open Source Software* (<https://joss.theoj.org/papers/e03bb9530fd2e1c0621e35352b71691e>).

Lors de cette courte présentation, seront présentés succinctement les besoins ayant motivé le développement de cette application, les méthodologies de développement mises en place, y compris la rédaction de la documentation et l’utilisation en sessions de formation, ainsi que les stratégies de déploiement et de valorisation adoptées.

Références

- Fay, Colin, Vincent Guyader, Sébastien Rochette, and Cervan Girard. 2023. *Golem: A Framework for Robust Shiny Applications*. <https://CRAN.R-project.org/package=golem>.
- McMurdie, Paul J., and Susan Holmes. 2013. “Phyloseq: An r Package for Reproducible Interactive Analysis and Graphics of Microbiome Census Data.” *PLoS ONE* 8 (4): e61217. <https://doi.org/10.1371/journal.pone.0061217>.
- Meyer, Fanny, and Victor Perrier. 2024. *Esquisse: Explore and Visualize Your Data Interactively*. <https://CRAN.R-project.org/package=esquisse>.

Human in the deep: Converting research activities pressures into ecological impact assessment.

Riwan Leroux* Jozée Sarrazin⁺ Marjolaine Matabos[~]

Résumé

Les écosystèmes hydrothermaux profonds abritent une biodiversité unique mais sont de plus en plus exposés aux activités anthropiques et suscitent un intérêt croissant pour leurs ressources minérales. Le champ hydrothermal Lucky Strike (LS) est localisé le long de la dorsale médio-Atlantique au cœur de l'aire marine protégée (AMP) des Açores. Découvert en 1993, LS est une des zones hydrothermales la plus étudiée au monde avec des campagnes annuelles de maintenance de l'observatoire EMSO-Açores depuis 2010. Alors que le gouvernement des Açores revoit la législation de l'AMP, ce projet a pour objectif de caractériser les activités de recherche, évaluer les pressions associées et déterminer leur impact. Pour cela nous avons construit une base de données spatio-temporelle sur R compilant toutes les activités de recherche (trajectoires des submersibles, échantillonnages, mouillages ou déchets) des 31 dernières années. La cartographie de ces activités a été faite avec les packages Leaflet et Terra pour fournir un catalogue interactif de cartes facile à utiliser pour les décideurs politiques responsables de l'AMP de LS. Ensuite, la mise en œuvre de la méthode Delphi pour l'élicitation d'experts a permis de pondérer les différentes activités et construire des cartes intégrées de pressions. Il en résulte une série de cartes décrivant l'évolution temporelle des pressions exercées par les activités de recherche au cours des dernières décennies. Une classification par clustering a permis d'analyser les variations spatio-temporelles des pressions exercées afin de grouper les sites en fonction de l'indice de pression. La prochaine étape consiste à utiliser les données historiques et nouvelles de communauté issues de l'échantillonnage et de l'imagerie pour évaluer l'impact potentiel de ces pressions. Des défis découlent des effets confondant aléatoires tels que la variabilité inter et intra-site ou la présence de domaines chimiques et devront être pris en compte pour les analyses multivariées.

Mots-clefs (3 à 5) : Statistique – Ecologie – Statistique Spatiale – Environnements profonds

Développement

Découverts en 1977, les écosystèmes hydrothermaux profonds sont de plus en plus ciblés pour leurs ressources minérales. Des années d'exploration sont encore nécessaires pour mieux comprendre leur fonctionnement et évaluer leur résilience. Ces connaissances sont essentielles pour comprendre et prévoir comment les activités anthropiques peuvent altérer ces écosystèmes. A ce jour, la seule pression directe d'origine humaine sur les champs hydrothermaux profonds provient des activités de recherches scientifiques. Le projet PROTECT vise à caractériser ces pressions de recherches sur le champ hydrothermal Lucky Strike (LS), étudié depuis 30 ans. Une fois ces pressions caractérisées, il sera possible d'estimer pour la première fois leur impact sur les écosystèmes et leur fonctionnement.

Premièrement, je vais montrer comment avec R il est possible de construire une base de données complexe et explicite dans le temps et l'espace. En récupérant les métadonnées des activités de recherches des 30 dernières années à LS, on a obtenu plusieurs bases de données pour chaque type d'activité. Par exemple, les coordonnées spatiales, dates et natures des échantillonnages ou les trajectoires des submersibles utilisés. Les carnets de bords qui décrivent les opérations scientifiques

à chaque plongée, minute par minute, ont été analysés sur R pour extraire toutes les mentions de déchets laissés au fond par les scientifiques. Grâce aux packages Leaflet et Terra (raster), on a compilé toutes ces données sur une seule carte interactive où l'on peut afficher ou non les différentes activités et autres caractéristiques topographiques du site d'étude (voir Annexe 1).

Une fois ce travail d'archive fait et que l'Atlas est disponible, nous avons cherché à intégrer toutes les pressions de natures différentes en un seul indice pondéré par l'impact des différentes activités, la lumière d'un submersible étant à priori moins dommageable pour l'écosystème qu'un prélèvement important de faune. Pour éviter une pondération arbitraire, nous avons utilisé la méthode Delphi (Mukherjee et al., 2015 ; Grime & Wright 2016) pour prédir les impacts des différentes activités de recherche. Un questionnaire (<https://forms.ifremer.fr/eep-protect/delphi/>) a été envoyé à l'international à des chercheurs en écologie des environnements profonds pour qu'ils attribuent un niveau d'impact à chaque activité. Leurs réponses ont été analysées sur R (moyenne, écart-type, distribution) et un deuxième questionnaire a été renvoyé pour demander aux participants de renoter les différentes activités à la vue des réponses pour tenter de s'approcher d'un consensus et ainsi renforcer la précision de la prédition des impacts associés aux activités de recherche. Les notes issues du deuxième questionnaire ont été utilisée pour construire notre indice intégré : le Integrated Research Pressure Index (IRPI). Pour chaque pixel (x,y) de la carte de LS, l'IRPI est la somme des pression, avec l'intensité de chaque pression (A_i) normalisée par son maximum (A_{max}) puis pondérée par la note Delphi associée (P_{Ai}) :

$$IRPI_{x,y} = \sum_i \frac{A_{x,y,i}}{A_{max, i}} \times P_{Ai}$$

Cet indice unique nous a permis ensuite de définir un gradient de pression. En le calculant année après année, on a pu caractériser le niveau de pression pixel par pixel au cours du temps. Une analyse de clustering spatial qui prend en compte les variations temporelles de l'IRPI définit les sites fortement pressurisés, modérément pressurisés et préservés (voir Annexe 2). Ce résultat a été transmis au gouvernement des Açores qui gère l'aire marine protégée dans laquelle se situe LS et qui est sujette à une actualisation de sa législation.

Finalement, ce travail de recherche d'archive, de cartographie et d'analyse permet l'élaboration d'un plan d'échantillonnage pour évaluer l'impact de ces pressions sur l'écosystème et son fonctionnement. D'abord, un échantillonnage sera effectué le long du gradient de pressions défini pour analyser les variations des communautés. En parallèle, des analyses d'images compareront les communautés présentes dans les années 90 avec celles que l'on retrouve dans les années 2020.

En conclusion, ce projet est un cas d'étude soulignant la versatilité de R. De la création de base de données, la cartographie, la communication, le conseil aux décideurs politiques, les analyses statistiques ; R est un outil qui permet de construire tout un projet dans le même environnement de travail. A cela s'ajoute le pouvoir reproductible avec les documents R markdown qui standardisent le workflow et facilitent la communication. Par exemple, la transmission des résultats du Delphi au panel d'experts via le markdown a permis un ajustement rapide et efficace des analyses et questionnaires (voir Annexe 3).

Références

- Grime, M. M., & Wright, G. (2016). Delphi method. Wiley StatsRef: Statistics Reference Online, 1, 16.
Mukherjee, N., Huge, J., Sutherland, W. J., McNeill, J., Van Opstal, M., Dahdouh-Guebas, F., & Koedam, N. (2015). The Delphi technique in ecology and biological conservation: applications and guidelines. Methods in Ecology and Evolution, 6(9), 1097-1109.

* IFREMER, Riwan.Leroux@ifremer.fr

[†] IFREMER, Jozee.Sarrazin@ifremer.fr

~ IFREMER, Marjolaine.Matabos@ifremer.fr

Maturation de codes scientifiques R de traitement de données liées à l'Eau au BRGM (initiative MATUREAU)

Marc LAURENCELLE* Théophile LOHIER†

Résumé

Plusieurs outils et fonctions de traitement de données scientifiques de la thématique Eau ont été développés au cours des dernières années au Bureau de recherches géologiques et minières (BRGM). La démarche MATUREAU initiée au BRGM en 2024 vise à finaliser une sélection d'outils et packages internes existants, en travaillant principalement sur : i) la modularisation des codes (le plus possible en fonctions) ; ii) la création de packages ; iii) la documentation, incluant la proposition d'exemples, de vignettes et de workflows exécutables ; iv) la simplification du mode opératoire. Le tout afin de rendre ces outils plus accessibles, d'encourager la poursuite de leur développement et maintenance d'une manière mieux structurée, et de rendre possible leur éventuelle diffusion à l'externe (en open source lorsqu'approprié). Cette présentation vise à décrire la démarche en cours, en l'illustrant par quelques outils en cours de maturation. Un outil de plus en plus utilisé au BRGM pour explorer la variabilité spatiale et temporelle de la dynamique des nappes d'eau souterraine par le clustering des chroniques piézométriques (séries temporelles de niveaux d'eau) en fonction de la similarité de leur signal, sera présenté plus en détails. Des éléments liés aux algorithmes et packages utilisés dans les outils développés seront également mentionnés afin d'alimenter la discussion et les échanges. Nous souhaitons ainsi ressortir de ces Rencontres R avec plein d'idées et pistes d'améliorations de la part des participants (aspects calculatoires, gestion des données, rendu graphique, création de packages, etc.) tout en intégrant un réseau de scientifiques particulièrement passionnés par le langage R.

Mots-clefs : Maturation – Traitement de données – Outil – Package – Eau

Développement

De nombreux outils et fonctions de traitement de données scientifiques de la thématique Eau ont été développés au cours des dernières années au BRGM. Le Bureau de recherches géologiques et minières ([BRGM](#)) est l'établissement public de référence dans les applications des sciences de la Terre pour gérer les ressources et les risques du sol et du sous-sol dans une perspective de développement durable. Ces codes, écrits principalement en langage R, constituent tantôt une collection de fonctions (ex. dédiées à l'analyse ou au prétraitement de séries temporelles ou plus spécifiquement de chroniques de niveaux d'eau souterraine ou de débits de cours d'eau) ; ou tantôt un outil pouvant être exécuté en chargeant un script principal dans l'environnement R, en modifiant manuellement les configurations d'options inscrites dans le script. **Le problème actuel** : peu de ces outils ont atteint un degré de maturité suffisant pour que l'outil soit facilement maniable par un utilisateur autre que le développeur lui-même, ou pour une diffusion de l'outil à l'externe.

L'objectif de la démarche MATUREAU, initiée au BRGM en 2024, est double : il s'agit d'une part de simplifier la mise en œuvre de chaînes de traitements complexes et éprouvées, sur de nouvelles

* BRGM, DEPA/EVE, Orléans, m.laurencelle@brgm.fr

† BRGM, DNG/TIA, Orléans, t.lohier@brgm.fr

sources de données ; et d'autre part de faciliter l'évolution des fonctionnalités existantes, le développement de nouvelles fonctionnalités, et leur intégration dans ces chaînes de traitements. La mise en œuvre est simplifiée à travers le packaging des principales fonctionnalités, l'implémentation de chaînes de traitements de référence, la standardisation des entrées et sorties, et la documentation du tout notamment à l'aide de vignettes. Un travail de structuration, d'harmonisation et de documentation des codes décrivant les principales fonctionnalités facilite l'implémentation de ces chaînes de traitements de référence. Une attention particulière est portée à la modularité afin de permettre aux développeurs actuels et futurs d'intégrer des méthodes alternatives dans les workflows. Enfin, des tests fonctionnels sont mis en place en s'appuyant sur ces workflows de référence afin de faciliter la maintenance et la mise à niveau des packages. **Plusieurs outils ou traitements** ont été identifiés pour une maturation en 2024. En voici les principaux :

(1) Tout d'abord, un outil d'analyse de « chroniques piézométriques » (séries temporelles de niveaux d'eau souterraine) incluant un module de **clustering de ces séries temporelles**. Cet outil est régulièrement sollicité par les collègues en interne, et intéresserait sans doute aussi plusieurs hydrogéologues en dehors du BRGM. Cet outil permet d'abord d'extraire et prétraiter les données de niveaux d'eau de N piézomètres issues d'ADES (la base de données nationale française sur les eaux souterraines, gérée par le BRGM) et/ou d'autres sources de données (ex. fichiers locaux) au besoin. Ensuite, le regroupement des chroniques de niveaux d'eau (individus) se fait avec la fonction pam de la librairie cluster qui met en œuvre l'algorithme des k-médoïdes, préféré par rapport aux k-means car jugé plus robuste face aux bruits et aux valeurs aberrantes. La dissimilarité entre individus est évaluée via deux mesures de distances : i) une distance « statistique » basée sur la corrélation entre séries A et B : $d = 1 - r(A,B)$; ii) une distance « géographique » (optionnelle) entre les points A et B. L'outil permet de tester différentes combinaisons de paramètres, dont une prise en compte ou non de la distance géographique (en plus de la distance statistique toujours considérée). L'utilisateur peut choisir le nombre de clusters, le type de corrélation calculée (linéaire classique de Pearson ; de rang, non paramétrique, de Kendall), etc. L'outil exporte un tableau de résultats (avec le numéro de cluster attribué à chaque individu et plusieurs indicateurs renseignant sur la qualité du clustering) ainsi que des résultats visuels détaillés (par cluster) et globaux (carte, matrice de graphiques). Les **FIGURES 1 ET 2** apportent un exemple de résultats visuels pouvant être générés par l'outil. En termes de maturation, les aspects à améliorer en priorité dans la structure de cet outil sont sa capacité à s'adapter à différents types de sources de données (API en ligne, base de données locale interne, fichiers spécifiques) sans codage complexe, et un accès simplifié à la configuration des options.

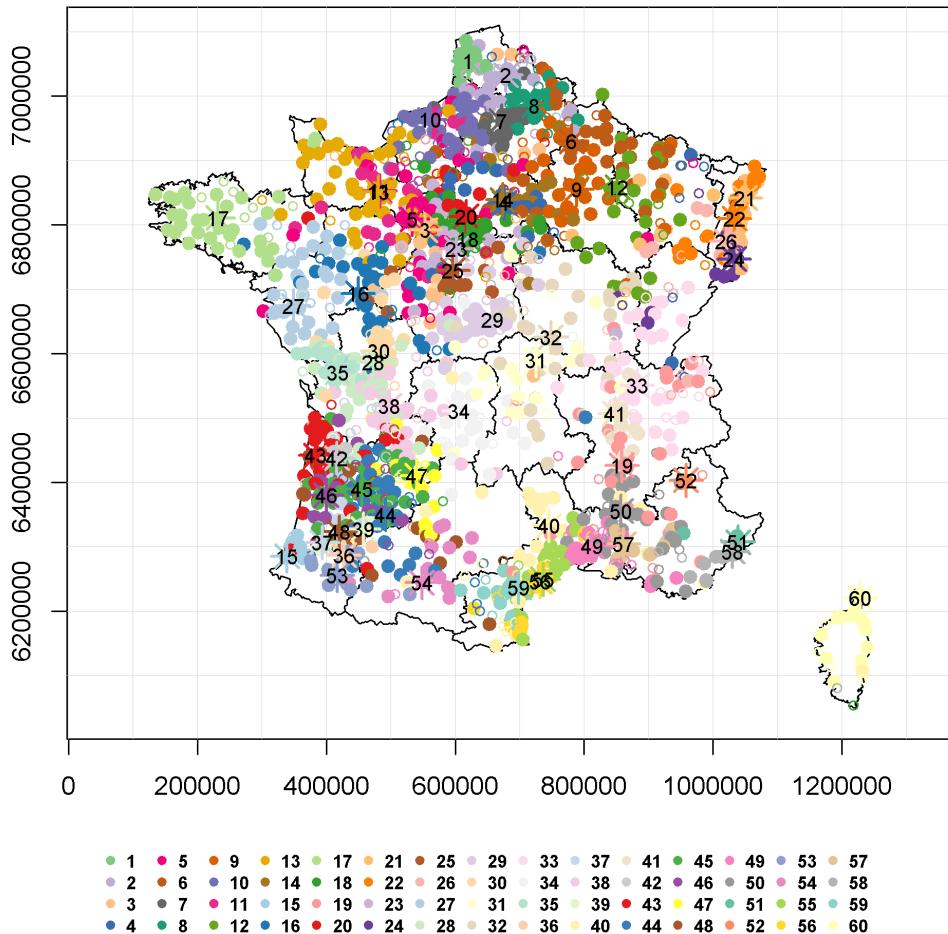
(2) Un autre outil permet de traiter en lot N séries piézométriques afin de calculer divers indicateurs de « Bilans Annuels » sur la nappe (ex. sa recharge apparente) tout en servant aussi à la détection du début d'une Recharge notable de la nappe par analyse de sa Piézométrie (d'où son nom « BARP »). Cet outil inclut un algorithme maison très efficace pour **repérer les Hautes Eaux et Basses Eaux** par année hydrologique adaptative dans des chroniques piézométriques mêmes complexes : des informations clés pour dériver quantité d'indicateurs sur l'évolution historique de la ressource en eau souterraine et révéler les situations les plus anormales, extrêmes (**FIGURE 3**). Un des principaux enjeux pour cet outil est de le rendre plus modulaire afin de permettre aux utilisateurs de calculer les indicateurs à la demande et de faciliter l'intégration de nouveaux indicateurs dans le workflow.

(3) Les fonctions de calcul de l'**Indicateur Piézométriques Standardisé (IPS)** utilisé par le BRGM depuis plus de 10 ans pour décrire l'état quantitatif des nappes d'eau souterraine en France dans les bulletins de situation hydrologique (**BSH**) publiés chaque mois (**FIGURE 4**) font elles aussi l'objet d'un

important travail de maturation et mise en package afin de centraliser et ainsi faciliter le déploiement et la maintenance de ces codes de calcul et d'assurer la cohérence entre services. (4) Des outils de traitement de données liées non pas à la quantité mais à la **qualité de l'eau** sont également considérés : génération de diagrammes de Piper, caractérisation des tendances temporelles par régression multi-segments, ... (5) Enfin, d'**autres fonctions** de portée plus générale seront packagées dans le cadre de la démarche MATUREAU.

Ces outils reposent sur l'utilisation de **nombreux packages R** publics et pour la plupart bien connus : zoo et lubridate (pour les séries temporelles) ; cluster, fpc, amap, usedist, psych et igraph (distances et clustering) ; segmented (régression multi-segments) ; circular (stat. directionnelles) ; RPostgreSQL ; data.table ; httr (requêtes web) ; etc. D'autres packages seraient peut-être plus intéressants parfois ?

Cette participation aux Rencontres R sera l'occasion d'exposer la démarche MATUREAU en cours, mais surtout de montrer les principaux outils et packages en cours de maturation, pour susciter des échanges inspirants et gagnants-gagnants sur les aspects techniques liés à la démarche et aux outils.



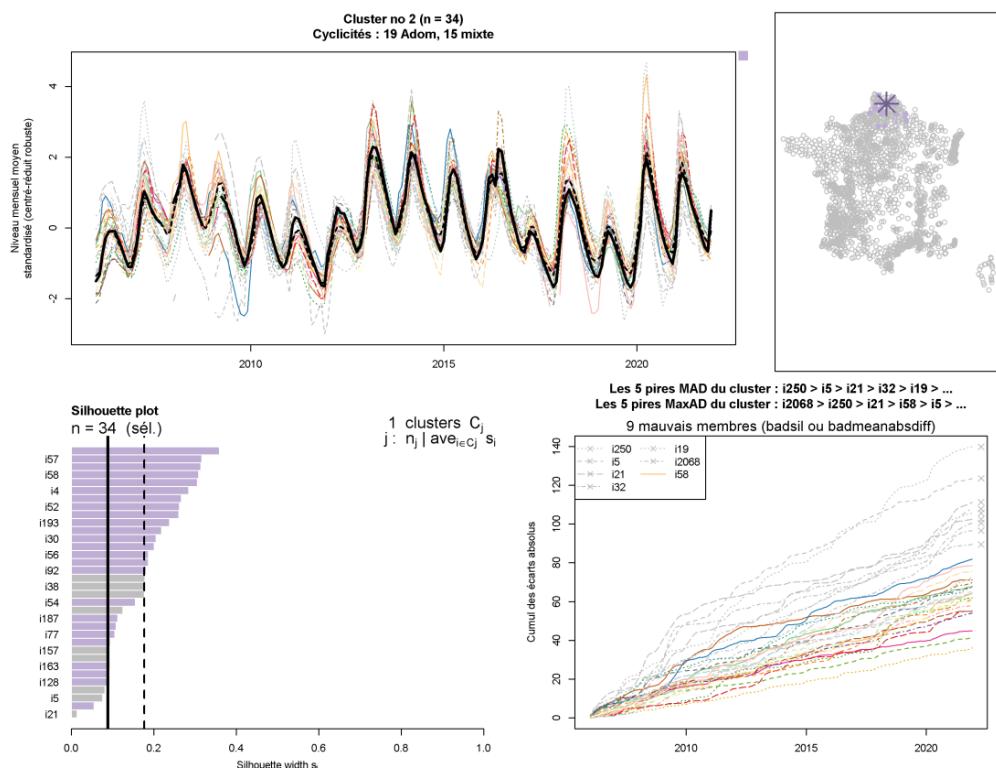


Figure 2 : Exemple d'un clustering... (suite) : b) résultats détaillés du cluster n°2 concentré dans le nord du pays

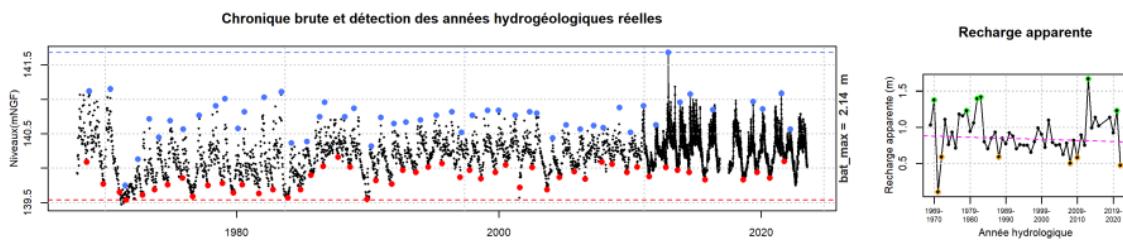


Figure 3 : Exemple de résultats produits par l'outil « BARP » : hautes eaux (en bleu) et basses eaux (en rouge) détectées par année flexible et indicateur annuel de recharge apparente calculé à partir de celles-ci

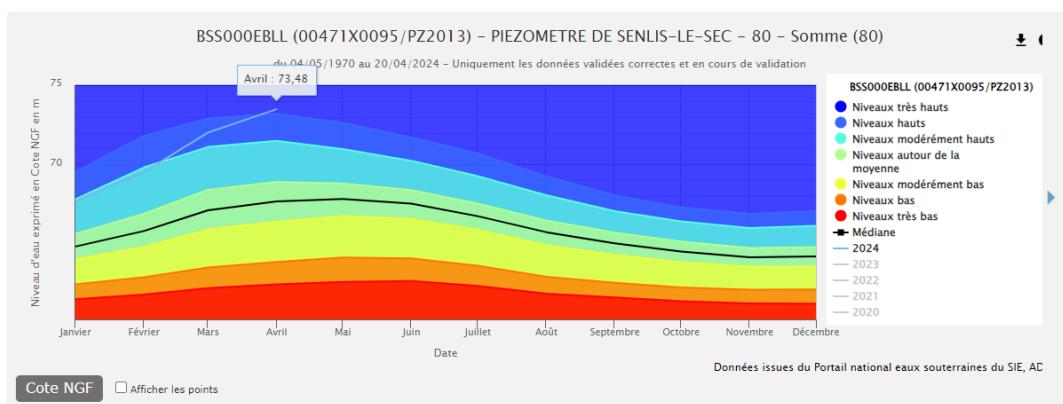


Figure 4 : Exemple de présentation visuelle de l'Indicateur Piézométrique Standardisé (IPS) sur le site web ADES : niveaux seuils mensuels calculés par la méthode (pour un piézomètre donné) afin de délimiter les classes d'IPS et courbe d'évolution récente du niveau piézométrique moyen mensuel au cours des mois de l'année 2024

Un suivi régionalisé et actualisé des étiages des petits cours d'eau, c'est possible avec R, Hub'Eau et GitHub !

Benoit Richard 1* Cédric Mondy 2† Pascal Irz 3‡ Antonio Andrade 4§
Céline Nowak 5¶

Résumé (max 300 mots)

Depuis 2012, l'Observatoire national des étiages (ONDE) porté par l'Office français de la biodiversité (OFB) permet de suivre en période estivale la situation d'écoulement des petits cours d'eau sur le territoire, et contribue à la gestion équilibrée de la ressource en eau lors de situations de crise (e.g. sécheresse). Dans ce contexte, la visualisation des données fréquemment actualisées doit être facilitée. Plusieurs directions régionales « pilotes » de l'OFB ont initié le développement d'un projet de valorisation régionalisée autour de ces données (https://github.com/richaben/PRR_ONDE). Ce projet s'est construit dans un cadre reproductible et automatisé en capitalisant sur plusieurs outils : la plateforme Hub'Eau pour la diffusion des données via API, le package R hubeau (Dorchies 2022) pour leur récupération, le format Rmarkdown pour la création de rapports, les services GitHub pour les mises à jour périodiques (GH Actions) et le déploiement au format HTML (GH Pages). L'organisation du projet a été pensée pour faciliter son déploiement et son exploitation sur l'ensemble des régions, sans besoin de compétences fortes en traitement de données.

Le développement s'appuie sur un workflow de plusieurs scripts R permettant la récupération des données, leur préparation, la création des visuels et de la page HTML, le tout déposé sur une branche principale d'un dépôt de référence. Les différentes directions régionales peuvent ainsi cloner ce dépôt et personnaliser le fichier de configuration pour adapter le traitement des données à leur territoire. Une seconde branche de déploiement a été mise en place où une GH Action récupère les modifications du code de la branche principale. Une autre GH Action y est exécutée régulièrement pour consulter l'API à l'aide du package hubeau. En cas de nouvelles données disponibles, les scripts sont exécutés dans un conteneur Docker, et la page HTML hébergée sur GH Pages est mise à jour.

Mots-clefs (3 à 5) : Reproductibilité - Visualisation - Data - Eau

*Office français de la biodiversité, Direction Régionale Normandie, Seulline, France, benoit.richard@ofb.gouv.fr

†Office français de la biodiversité, Direction Régionale, Île-de-France, Vincennes, France, cedric.mondy@ofb.gouv.fr

‡Office français de la biodiversité, Direction Régionale Bretagne, Cesson-Sévigné, pascal.irz@ofb.gouv.fr

§Office français de la biodiversité, Direction Surveillance Évaluation et Données, Vincennes, France, antonio.andrade@ofb.gouv.fr

¶Office français de la biodiversité, Direction Surveillance Évaluation et Données, Vincennes, France, celine.nowak@ofb.gouv.fr

Développement

La présentation abordera plusieurs parties :

- une introduction présentant le réseau ONDE et ses objectifs, les données collectées et les besoins de valorisations associées ;
- le cadre dans lequel ce projet de valorisation régionalisé reproductible a été développé (outils, méthodes, organisation, documentation, ...) ;
- une présentation du rendu final (page HTML) avec les datavisualisations produites ;
- et une conclusion avec retours d'expérience et perspectives éventuelles.

Références

Dorchies, David. 2022. *Hubeau: An r Package for the Hub'eau APIs.* <https://doi.org/10.57745/XKN6NC>.

Sécurisation des analyses statistiques avec R : retour d'expérience

Julien Dugas* Aurore Philibert†

Résumé (max 300 mots)

Comment maintenir dans le temps des scripts R et applications Shiny ?

C'est la question que nous nous sommes posés au GEVES (Groupe d'Étude et de contrôle des Variétés Et des Semences) car cette maintenance crée de nombreuses contraintes au quotidien :

- chaque script R ou application Shiny doit être lancé avec la bonne version de R et de packages (et toujours la même pour chaque utilisateur) via des applications métiers spécifiques développées dans d'autres technologies
- une mise à jour des versions de R et de packages doit se faire au fur et à mesure, par analyse, en éliminant les effets de bords pour garantir la stabilité des résultats
- plusieurs utilisateurs doivent pouvoir utiliser la même application Shiny simultanément
- le choix des méthodes statistiques et de la manière de les retranscrire en R doit être facilement transmissible d'un statisticien à un autre
- les scripts R doivent être centralisés et gérés de la même manière
- les évolutions des analyses et les nouveaux développements doivent être tracés et suivis

Dans cette présentation vous apprendrez comment créer un environnement technique et fonctionnel (Docker, Shiny Proxy, Git, tests unitaires, ...) permettant de palier à toutes ces contraintes. Nous détaillerons les choix techniques et leurs raisons, les différentes étapes suivies, les compétences nécessaires, les problématiques rencontrées et les solutions choisies.

Mots-clefs (3 à 5) : Maintenance - Reproductibilité - Plateforme - Biostatistique

Développement

En tant qu'organisme réglementaire, le GEVES doit garantir la sécurité de ses systèmes d'information, de ses données mais aussi de ses analyses statistiques. Au sein du pôle Biostatistique du GEVES nous avons travaillé sur la sécurisation de toutes les analyses statistiques que nous avons développées en R. Ses scripts ayant la particularité de devoir être maintenus dans le temps.

Pour cela nous avions comme cible les états suivants :

- Quand un script R est lancé, tous les utilisateurs doivent utiliser la même version du script
- Quand un script R est lancé, c'est bien la version de R et des packages avec lesquels ce script fonctionne qui doit être utilisée
- Il doit être possible de retrouver la version de script R utilisé pour produire un précédent résultat

*GEVES, Pôle Biostatistique, julien.dugas@geves.fr

†GEVES, Pôle Biostatistique, aurore.philibert@geves.fr

-
- Une montée de version de R et de packages doit être faite régulièrement pour chaque script R, indépendamment
 - La maintenance d'un script R doit pouvoir être repris en charge rapidement par un nouveau statisticien ou un statisticien ne travaillant pas sur ce script en temps normal.

En ce qui concerne la centralisation des scripts, le sujet de les stocker sur gitlab et de pouvoir les lancer via un seul et même serveur est rapidement apparu. Mais la difficulté de pouvoir lancer chaque script R avec SA propre version de R et de packages nous a amené à nous intéresser à la technologie Docker qui permet ce cloisonnement entre script. Malheureusement aucune personne dans les pôles informatiques n'étaient connaisseurs de cette technologie. Nous avons donc travaillé de concert avec le pôle Systèmes et Réseaux du GEVES pour comprendre et installer cette technologie.

Nos scripts étant, en partie, mis à disposition aux utilisateurs via des applicatifs métiers (développés en .net par le pôle Bases et Développement du GEVES), nous avons collaboré avec ce pôle pour trouver la manière la plus appropriée de faire les lancements de scripts R. Pour cela une API a été développée, interagissant avec l'API Docker. Les échanges de données ont du être retravaillés pour que l'application métier puisse envoyer les données d'entrées au script R et ensuite récupérer les données de sorties. Le format des fichiers d'entrées et de sorties a été défini comme du json pour faciliter ces échanges avec un protocole standard (ou normalisé). Cette solution permet de garantir que si deux utilisateurs lancent le même traitement (mêmes paramètres et mêmes données d'entrées) ils obtiendront exactement le même résultat, quel que soit leur équipement informatique, leur lieu géographique, etc.

Une base de données spécifique a été créée pour permettre le suivi des versions de scripts au fil du temps avec à un temps donné un seul script d'"actif". Tous les lancements d'analyse sont ainsi tracés ce qui permet le débogage et la récupération de la version de script R utilisée sur un précédent lancement.

En ce qui concerne les scripts R non mis à disposition via des applicatifs métiers nous avons décidé de les mettre à disposition des utilisateurs via des applications web développées en Shiny. Afin de permettre à plusieurs utilisateurs d'utiliser la même application en parallèle nous avons opté pour la solution Shiny Proxy. De même que pour Docker il nous a fallu apprendre à installer et gérer cette nouvelle technologie.

La maintenance dans le temps des scripts R nous a amené à chercher comment procéder aux "montées de versions" de R et des packages. Nous nous sommes intéressés aux tests unitaires, très répandus dans le monde du développement informatique. Ces tests unitaires permettent, par une étape préalable de création de ces tests, de lancer automatiquement le script avec une version de R et de packages plus récentes. Les tests vérifient automatiquement que les résultats sont identiques sur un jeu de données tests. Et si un test ne fonctionne pas alors il est clairement indiqué où une modification est nécessaire. Cette solution demande du temps lors de l'implémentation mais rend très rapide ces changements de version de R par la suite.

Enfin, nous devons être capable de garantir la connaissance des analyses de chaque script R développé. Une vignette est maintenant associée à chaque analyse décrivant les modifications/sélections de données éventuelles ainsi que les analyses statistiques utilisées. Dans cette vignette un lien est fait entre les analyses et la structuration du script R permettant à une nouvelle personne de facilement prendre en main le suivi de ce script.

En conclusion, nous avons réussi à développer une plateforme de biostatistique permettant de sécuriser les analyses statistiques et de les maintenir au cours du temps. Ce développement nous a pris une année entière entre le démarrage et la mise en production. Nous avons eu besoin de nous former à de nombreuses nouvelles technologies (Docker, Shiny Proxy, API Docker, ...) et de solliciter les compétences informatiques de nos collègues. Avec nos collègues il a fallu apprendre à communiquer, les vocabulaires et notions n'étant pas les mêmes entre un statisticien et un informaticien, et à s'adapter au fur et à mesure des tests et des avancées que nous avons faites. L'avantage d'avoir pu le développer en interne est que cela nous a permis de monter en compétences et d'être capable de maintenir dans le temps cette plateforme que nous avons appelé le Black Pearl.

Références

Refactoring : du code qui marche, c'est bien, mais du code maintenable, c'est mieux

Vincent Guyader 1*

Résumé

“N’importe qui peut écrire du code qu’un ordinateur peut comprendre. Les bons programmeurs écrivent du code que les humains peuvent comprendre.” *Martin Fowler. (Fowler (1999))*

Formalisé dans les années 90, le refactoring est défini comme un processus systématique de restructuration du code sans en modifier le comportement externe. Il émerge comme un outil essentiel pour améliorer la lisibilité, la maintenabilité et l’efficacité des programmes informatiques. Cette intervention vise à présenter les techniques et bonnes pratiques de refactoring spécifiquement adaptées à l’environnement R.

Mots-clefs : Refactoring, Bonnes pratiques, Maintenabilité, Développement logiciel.

Développement

Nous commencerons par définir le concept de refactoring et son importance dans le développement logiciel. Ensuite, nous explorerons les motifs courants de mauvaise conception du code R et discuterons de la manière dont le refactoring peut les résoudre. Nous mettrons en lumière les principaux avantages du refactoring, notamment l’amélioration de la clarté du code, la réduction de la duplication, et par conséquent, l’amélioration de sa maintenabilité.

Renommage de Variables

Le renommage de variables est une technique de refactoring essentielle pour rendre le code plus compréhensible et cohérent. En R, il est fréquent de rencontrer des noms de variables peu descriptifs ou mal choisis, ce qui peut rendre le code difficile à suivre. En utilisant des noms de variables significatifs et explicites, nous pouvons améliorer la clarté du code et faciliter sa compréhension par les autres développeurs. Par exemple, en remplaçant des noms de variables génériques comme “x” ou “temp” par des noms descriptifs tels que “revenu_annuel” ou “donnees_utilisateurs”, nous rendons le code plus explicite et plus facile à maintenir.

```
# Avant
f <- function(p) {
  s <- 0
  for (i in p) {
    s <- s + i
  }
  return(s / length(p))
}

# Après
calcul_moyenne <- function(numbers) {
  moyenne <- sum(numbers) / length(numbers)
  return(moyenne)
}
```

Extraction de Fonctions

L’extraction de fonctions consiste à regrouper des blocs de code répétitifs ou complexes dans des fonctions distinctes, ce qui permet de réduire la duplication et d’améliorer la modularité du code. En identifiant les sections de code récurrentes dans notre programme, nous pouvons les extraire dans des fonctions réutilisables, ce qui facilite la mise à jour et la maintenance du code. Par exemple, si nous avons un bloc de code qui

*ThinkR 1, vincent@thinkr.fr

effectue un calcul spécifique sur plusieurs jeux de données, nous pouvons le regrouper dans une fonction dédiée et l'appeler chaque fois que nécessaire, au lieu de le répéter à plusieurs endroits dans le code.

```
# Avant
calculate_and_print_average <- function(a, b) {
  average <- (a + b) / 2
  cat("La moyenne de", a, "et",
      b, "est", average, "\n")
}

# Après
calculate_average <- function(a, b) {
  return((a + b) / 2)
}

calculate_and_print_average <- function(a, b) {
  average <- calculate_average(a, b)
  cat("La moyenne de", a, "et",
      b, "est", average, "\n")
}
```

Simplification des Expressions Conditionnelles

Les expressions conditionnelles complexes peuvent rendre le code difficile à comprendre et à déboguer. En simplifiant les expressions conditionnelles, nous pouvons rendre le code plus lisible et réduire le risque d'erreurs. En utilisant des opérateurs logiques clairs et en évitant les constructions excessivement complexes, nous pouvons améliorer la clarté et la concision du code. Les expressions conditionnelles complexes cachent généralement des règles métiers importante qu'il convient d'extraire et d'expliquer.

```
# Avant
if ( !(jour != "Saturday" & jour != "Sunday") | rtt ) {
  # Instructions
} else {
  # Instructions
}
```

Deviendra :

```
# Après
if ( en_vacances(jour,rtt) ) {
  # Instructions
} else {
  # Instructions
}

# on extrait et explicite la règle métier
en_vacances <- function(jour,rtt) {
  week_end <- c("Saturday", "Sunday")
  jour %in% week_end | rtt
}
```

Ces techniques de refactoring, parmi d'autres qui seront abordées, contribuent à améliorer la qualité et la maintenabilité du code R, en rendant le code plus clair, plus concis et plus facile à gérer pour les développeurs.

Cette intervention présentera des cas d'usage de la "vraie vie" et s'adressera aussi bien aux débutants qu'aux programmeurs expérimentés en R, en offrant des conseils pratiques pour améliorer la qualité et l'efficacité de leur code tout au long du cycle de développement. Elle proposera de faire un détour sur le "golden master" préalable nécessaire à tout refactoring et ouvrira sur le concept de Test Driven Development (TDD).

Références

Fowler, Martin. 1999. *Refactoring: Improving the Design of Existing Code*. Addison-Wesley Professional.

Une vie polyamoureuse entre R et Julia

Rémy Drouilhet*

Résumé

Le langage **julia** partage avec le langage **R** les caractéristiques comme l'indexation des tableaux commençant à 1, le (*multiple*) *dispatching*, la *metaprogramming* et son système unique de gestion des librairies (paquets). A la différence de **R**, **julia** proposant une compilation JIT (*Just In Time*) est intrinsèquement plus performant que le langage **R**, rétablissant au passage l'utilisation des boucles **for** comme c'est le cas pour les langages compilés. De par sa jeunesse (un peu plus de 10 ans), **julia** reste toutefois un langage en devenir surtout au niveau du développement de son écosystème de paquets. Pour toutes ces raisons, le langage **julia** peut être vu comme un digne successeur du langage **R**. Dans cette présentation, nous proposons le paquet **R**, nommé **j14R**, dont l'objectif avoué est de téléguider depuis **R** des paquets **julia**. L'esprit du paquet est principalement d'imaginer le **julia** comme un remplacement de **Rcpp** et ainsi de proposer des paquets **R** de type *wrapper* de paquet **julia**.

Mots-clefs : Julia – *Multiple Dispatching* – Paquet **R** – *wrapper* de paquet **julia**

Démarrage rapide

Rien de mieux pour présenter le paquet **j14R** [1] que de commencer par une utilisation basique.

```
1 > require(j14R)
2 > ## conversion R vers julia
3 > v_jl <- jl(c(1,3,2))
4 > v_jl
5 3-element Vector{Float64}:
6  1.0
7  3.0
8  2.0
9 > jltypeof(v_jl)
10 Vector{Float64} (alias for Array{Float64, 1})
11 > length(v_jl)
12 [1] 3
13 > v_jl[2]
14 3.0
15 > jltypeof(v_jl[2])
16 Float64
17 > R(v_jl) # ou toR(v_jl)
18 [1] 1 3 2
19
20 > ## exécution directe code julia
21 > v2_jl <- jl('[1,3,2]')
22 > v2_jl
23 3-element Vector{Int64}:
24   1
```

*Laboratoire Jean Kuntzmann, Rémy.Drouilhet@univ-grenoble-alpes.fr

```

25  3
26  2
27 > jltypeof(v2_jl)
28 Vector{Int64} (alias for Array{Int64, 1})
29 > jltypeof(v2_jl[2])
30 Int64
31 > toR(v2_jl)
32 [1] 1 3 2

```

Après avoir chargé le paquet, nous créons une instance `julia` obtenue par autoconversion d'un vecteur R dont la sortie est celle proposée par `julia`. La fonction `jltypeof()` retourne le type `julia`, ici un `Array{Float64}`. Après avoir obtenu sa longueur, on extrait son deuxième élément dont on fournit le type `julia`. La fonction `jl()` permet aussi d'exécuter directement du code `julia` comme illustrée avec la variable `v2_jl`. Remarquons au passage que le type des éléments de `v2_jl` sont des `Int64`. La fonction `R()` (ou `toR()`) retourne le résultat d'une instance `julia` converti en objet R si possible.

En interne, une instance `julia` est représentée par un pointeur externe (`externalptr`) dont la classe est dans l'ordre le type `julia` puis `jlvalue`, comme le montre la sortie ci-dessous.

```

1 > typeof(v_jl)
2 [1] "externalptr"
3 > class(v_jl)
4 [1] "Array"    "jlvalue"

```

Fonctionnalités principales du paquet

N'ayant pas la place nécessaire dans ce résumé d'illustrer par des exemples les fonctionnalités principales du paquet (cela fera plutôt l'objet de la présentation), nous en proposons une liste non exhaustive :

1. la fonction principale `jl()` permettant :
 - (a) lorsque l'argument est une expression `julia` entre deux caractères ` (*backtick* en anglais) : d'exécuter du code `julia` et de créer une instance `julia`.
 - (b) lorsque l'argument est un objet R dont la conversion est gérée par le paquet : créer l'instance `julia` correspondante.
2. un unique objet `jl` défini dans l'environnement global de R qui permet de créer et extraire des variables dans le module `Main` de `julia`
3. une conversion de R vers `julia` des vecteurs, matrices de données, facteurs.
4. une conversion de `julia` vers R des `Array`, `Tuple`, `DataFrame` et `CategoricalArray`.
5. synchronisation entre les ramasses-miettes (*garbage collectors*) de `julia` et R.

Commentaires

Le paquet `j14R` est en phase de développement et est loin d'être considéré comme stable. Le paquet pourrait notamment dans l'avenir changer de nom. Comme le but avoué du paquet est de remplacer l'utilisation de `Rcpp`, il n'en dépend surtout pas. Un cas d'usage serait notamment le développement de *wrappers* de paquets `julia` et notamment un pour remplacer le paquet `VAM` (*Virtual Age Models*) qui repose actuellement sur `Rcpp`. Le nom du paquet R serait `VirtualAgeModels_jl` et permettrait de rendre accessible dans le système R le paquet `VirtualAgeModels.jl` en cours de développement et déjà installable dans l'environnement `julia` via le paquet `Pkg.jl`.

Notons aussi qu'il existe déjà un paquet R du même type **JuliaCall** [2] qui permet déjà de proposer des paquets R de type *wrapper* de paquet julia. L'exemple le plus notable est le paquet **diffeqr** [3] qui permet d'utiliser le paquet **DifferentialEquations.jl** directement depuis R. L'auteur de **diffeqr** [3] fournit dans le même esprit **diffeqpy** [4] un paquet python permettant d'utiliser le paquet **DifferentialEquations.jl** en python.

Cependant, à la différence de notre paquet, le paquet **JuliaCall** [5] :

1. dépend de **Rcpp** quand notre paquet repose uniquement sur l'API C de R
2. nécessite l'installation du paquet **RCall.jl**

Remerciements

Ce travail a été partiellement financé par l'Agence National Française pour la Recherche dans le cadre du programme France 2030 (ANR-15-IDEX-0002) et par le LabEx PERSYVAL-Lab (ANR-11-LABX-0025-01).

Références

- [1] **j14R**, paquet R, <https://github.com/rcqls/j14R>
- [5] **JuliaCall**, paquet R, <https://github.com/Non-Contradiction/JuliaCall>
- [3] **diffeqr**, paquet R, <https://github.com/SciML/diffeqr>
- [4] **diffeqpy**, paquet python, <https://github.com/SciML/diffeqpy>

Un petit coup de polish - Nettoyage de fichiers Excel avec R

Thomas Vroylandt*

Résumé

Les données qui nourrissent nos analyses sont souvent issues de fichiers Excel (xls ouxlsx), envoyées par d'autres services, disponibles en *open data* ou simplement issues d'outils de collecte de données. La première étape de toute bonne analyse est alors de parvenir à importer ces données correctement. Pourtant ces fichiers Excel ne sont pas tout le temps pensés pour être importés par un ordinateur et un logiciel statistique. Ils peuvent être truffés de formatages - texte en gras, cellules colorées, indentations, cellules fusionnées, pour ne donner que quelques exemples - qui complexifie leur import. Bien qu'il s'agisse d'une étape classique et cruciale, il est courant de buter sur un problème de ce type. Cette présentation se propose d'aborder plusieurs niveaux pour importer ces fichiers et les transformer en un format *tidy* (Wickham (2014)), facilement exploitable par la suite, à l'aide de packages comme {readxl} (Wickham and Bryan (2023)), {tidyxl} (Garmonsway (2023a)) ou encore {unpivotr} (Garmonsway (2023b)). Elle s'appuie sur des jeux de données réels disponibles en accès libre.

Mots-clefs : Excel - Data Cleaning - Nettoyage de données

Développement

Cette présentation s'articulera principalement autour de jeux de données réels et disponibles en ligne plus ou moins complexes à importer et nettoyer, issus d'administrations françaises productrices de données (Insee, Urssaf CN, Drees, etc.).

Elle s'attache à proposer des solutions pratiques, nourries par l'expérience et de complexité croissante pour quelques cas courants compliquant l'import des données :

- tableaux mal positionnés sur la feuille
- onglets contenant chacun une année/région/etc.
- formatages avancés (cellules fusionnées, indentations)
- information sous une autre forme que le texte (gras, couleur)
- fichiers différents d'une année à l'autre
- plusieurs tableaux à la suite dans un même fichier

Références

- Garmonsway, Duncan. 2023a. *Tidyxl: Read Untidy Excel Files*. <https://CRAN.R-project.org/package=tidyxl>.
- . 2023b. *Unpivotr: Unpivot Complex and Irregular Data Layouts*. <https://CRAN.R-project.org/package=unpivotr>.
- Wickham, Hadley. 2014. “Tidy Data.” *The Journal of Statistical Software* 59.
- Wickham, Hadley, and Jennifer Bryan. 2023. *Readxl: Read Excel Files*. <https://CRAN.R-project.org/package=readxl>.

*Kantiles, thomas@kantiles.com

{saperlipopette}, un paquet R pour progresser en Git en toute sérénité

Maëlle Salmon*

Résumé (max 300 mots)

Une pratique de Git confiante peut changer la vie de tout développeur·se de paquets R : un historique utile, une capacité à travailler en parallèle sur différents aspects dans différentes branches, etc. Cependant, il faut en arriver là à la sueur de son front, car Git, ce n'est pas simple ! Notamment, un obstacle peut être de ne pas savoir où et comment s'entraîner à utiliser les commandes moins basiques telles que `git commit --amend` pour changer son dernier commit, `git rebase -i` pour retravailler l'historique d'une branche, `git bisect` pour trouver quel commit a introduit ce fichu bug... Le paquet R saperlipopette est là pour vous aider ! Il offre pour le moment 12 exercices, inspirés du célèbre site "Dangit, Git!?!?", ou de la pratique Git de l'autrice. Chaque exercice est commencé en appelant une fonction telle que `saperlipopette::exo_committed_to_wrong()` qui concerne le scénario "zut, j'ai fait mon commit sur la mauvaise branche". La fonction crée l'exercice dans un dossier qu'on lui indique, temporaire par exemple. L'utilisateur·rice ouvre R dans ce dossier, et lit les instructions qui apparaissent. Si cela ne suffit pas, il·elle peut appeler la fonction `tip()` qui lui donne des indices en plus. Ainsi, on s'entraîne à Git sur un scénario utile et réaliste, avec ses outils habituels, depuis notre chère console R, dans un dossier où il n'y a rien à casser. Dans cette présentation, je vous expliquerai pourquoi j'ai créé ce paquet, comment il s'utilise, et pourquoi les fichiers `.Rprofile` sont les rois de son implémentation.

Mots-clefs (3 à 5) : Git - Package - Enseignement - Bonnes pratiques - Développement

Git, un outil crucial mais difficile à prendre en main

L'utilisation compétente d'un logiciel de contrôle de version améliore la productivité du développement logiciel des individus et des équipes en leur permettant de travailler simultanément sur différentes fonctionnalités, tout en enregistrant l'historique du projet. Améliorer ses compétences en contrôle de version au-delà des basiques est un investissement utile, qui se traduit rapidement par des gains de temps : par exemple on peut utiliser des commandes spécifiques pour annuler les modifications au lieu d'effacer et de copier-coller manuellement. Cela permet également de réduire les risques de perte de travail, car on sait comment éviter les erreurs et comment les réparer. Parmi les outils de contrôle de version, Git est le plus populaire ; en outre, une grande partie du développement de logiciels libres dans le monde se fait sur GitHub et, dans une moindre mesure, sur GitLab, qui utilisent tous deux Git. Cependant, Git est notoirement déroutant et difficile à apprendre, ce qui rend le perfectionnement potentiellement décourageant.

saperlipopette, un astucieux paquet R pour s'entraîner à Git en confiance

Pour tenter de résoudre ce problème, nous avons développé un paquet R, saperlipopette <https://maelle.github.io/saperlipopette/>. Le paquet saperlipopette permet de pratiquer localement les commandes Git avancées, qui ne sont généralement pas enseignées dans les ateliers pour débutant·e·s. Le public cible de saperlipopette est constitué de professionnel·le·s

- qui développent des scripts ou paquets R,
- qui travaillent habituellement avec R et connaissent les commandes Git de base telles que add, commit, push, et pull, et qui comprennent un peu les branches,

*rOpenSci, msmaellesalmon@gmail.com

-
- mais ne se sentent pas encore à l'aise avec Git, et donc sous-utilisent la palette de fonctionnalités de Git.

Par le paquet saperlipopette, nous fournissons aux apprenant·e·s 12 exercices (12 fonctions), inspirés du célèbre site “Dangit, Git!?!”, ou de la pratique Git de l'autrice. Les exercices sont centrés sur un scénario réaliste et utile, comme l'annulation d'un changement ou l'utilisation de la machine à remonter le temps de Git pour réparer un projet, et qui sont tous contenus dans un petit répertoire temporaire donc jetable : le code de saperlipopette crée le dossier, recrée l'historique et la situation Git nécessaires à l'exercice, stocke les instructions dans le `.Rprofile` du dossier. Par conséquent, les utilisateur·rice·s de saperlipopette peuvent s'exercer de manière réaliste à l'utilisation de nouvelles commandes Git plus avancées sans mettre en danger leur travail ni avoir à réfléchir à la manière de créer artificiellement un scénario d'exercice pour eux-mêmes.

Un outil d'apprentissage actif

Le principe de fonctionnement de saperlipopette est basé sur des stratégies pédagogiques d'apprentissage actif qui consistent à travailler sur des tâches authentiques, avec une pratique guidée. De plus, les exercices sont locaux, ce qui signifie que les utilisateurs peuvent s'exercer sur leur machine, avec les outils qu'ils connaissent déjà ou avec lesquels ils essaient de se familiariser : Git en ligne de commande, Git avec une interface telle que RStudio IDE, ou d'autres clients Git à usage général. La recréation d'un exercice se fait simplement par un appel de fonction R, de sorte que les utilisateurs peuvent s'exercer à tout moment, et même se rafraîchir la mémoire sur une commande avant de l'appliquer à leur projet de travail réel. Cette approche permet de réduire la charge cognitive des apprenant·e·s. Il est important de noter que notre logiciel fournit deux niveaux de guidage pour chaque exercice :

- une description générale du problème à résoudre à l'ouverture de la session R,
- la possibilité de demander plus d'indications si nécessaire en appelant la fonction `tip()`.

Enfin, du point de vue du formateur ou de la formatrice, les exercices prêts à l'emploi de saperlipopette permettent de gagner du temps dans la démonstration des flux de travail Git.

Présentation de saperlipopette

Dans cette présentation, nous couvrirons

1. L'utilité des commandes Git au-delà des bases ;
2. Le fonctionnement de saperlipopette pour un exercice simple ;
3. La liste d'exercices et les raisons de leurs choix ;
4. Le fonctionnement de saperlipopette pour un exercice plus compliqué ;
5. L'infrastructure interne de saperlipopette comme son utilisation de fichiers `.Rprofile` ;
6. Des pistes d'amélioration du paquet.

L'objectif sera de rendre l'audience curieuse de saperlipopette et Git, et aussi de récolter des retours pour améliorer le paquet.

Références

- saperlipopette <https://maelle.github.io/saperlipopette/>
- Dangit, Git!?! <https://dangitgit.com/fr>

GÉNÉRATION ALÉATOIRE D'EXERCICES DE BIOSTATISTIQUE POUR MOODLE VIA LE PACKAGE SARP.MOODLE DE R

Emmanuel CURIS¹ & Virginie LASSEUR²

Résumé

Moodle est un système d'aide à l'apprentissage, utilisé par de nombreux établissements d'enseignement (lycées, universités, grandes écoles...) qui permet, entre autres fonctionnalités, de réaliser des exercices avec correction automatique. Ces exercices, ou « questions » dans le langage Moodle, permettent en particulier à un apprenant de s'entraîner et de s'évaluer. Pour des raisons pédagogiques, il est nécessaire qu'à chaque utilisation, les questions auxquelles il doit répondre changent, afin d'éviter de trouver les réponses par tâtonnement. Cela peut se faire en tirant les questions au sort dans la base de questions, mais à condition que cette base soit bien organisée et riche en questions.

Dans ce contexte, nous proposons une présentation du package SARP.moodle, qui permet de générer simplement, à partir de R, des questions en grand nombre. Cette génération peut se faire soit à l'aide d'un fichier de questions créé dans un tableur, soit directement à partir de fonctions R. Cette seconde possibilité, bien plus riche, sera celle présentée ici. Elle permet en effet de faire « simplement » des questions de types complexes — questions qui se suivent, questions demandant de légendier un graphique... — et d'en préparer facilement de très nombreuses variantes, comme le montreront les exemples présentés, appliqués au cas de l'enseignement des statistiques.

Ce package et sa documentation sont disponibles sur le CRAN.

Mots-clés : R, SARP.moodle, moodle, questions aléatoires, génération d'exercices.

Développement

Pour un public apprenant, disposer d'exercices d'application est indispensable pour acquérir les compétences souhaitées, tant en termes de calculs que de méthodologie et d'interprétation. Pouvoir recommencer ces exercices est essentiel à l'apprentissage, ainsi que disposer d'un retour, même succinct, sur les étapes erronées de la résolution. Néanmoins, recommencer exactement le même exercice peut conduire à trouver la bonne solution davantage par combinatoire et essais-erreurs, ou par apprentissage « par cœur » des bonnes réponses, et non par l'application d'une démarche appropriée. L'exercice, tout en restant fondamentalement le même, doit donc varier à chaque tentative dans son énoncé. Une illustration archétypale serait, par exemple, pour la compétence « calculer la moyenne arithmétique d'une série de valeurs », qu'à chaque tentative, les valeurs doivent changer, mais aussi la taille de l'échantillon. Par ailleurs, dans un contexte de contrôle continu ou d'examen à distance, il est nécessaire de disposer d'exercices à la fois très voisins (tous les étudiants doivent passer une certification d'identique difficulté) et différents pour limiter les tentatives de fraudes.

Dans ces contextes, envisager un exercice comme une trame couplée à une génération aléatoire de ses variantes est une piste efficace pour répondre à ces besoins pédagogiques. Moodle [1] est une plateforme d'apprentissage, très répandue, adaptée à cette double démarche pédagogique, grâce à son système d'exercices avec correction automatique et commentaires aux étudiants. De plus, la possibilité de tirer une question au sort dans une catégorie donnée permet d'assurer ces variantes d'un même exercice. Cependant, la création de questions, en particulier complexes, est rapidement fastidieuse surtout lorsqu'il s'agit de générer de très nombreuses variantes d'une même question.

Pour pallier cette difficulté, nous avons développé pour R le *package* SARP.moodle [2] permettant d'utiliser les possibilités de R pour générer aléatoirement des séries de valeurs. Il permet ainsi de générer un nombre illimité de variantes d'un même exercice avant import dans Moodle tout en s'affranchissant du format XML complexe décrivant les questions pour Moodle. Cette présentation illustrera plusieurs utilisations du *package* pour générer de telles collections d'exercices avec différents niveaux de complexité :

¹ UR 7537 BioSTM, faculté de pharmacie de Paris, Université Paris Cité, 4 avenue de l'Observatoire, 75006 Paris, France – emmanuel.curis@u-paris.fr

² UR 7537 BioSTM, faculté de pharmacie de Paris, Université Paris Cité, 4 avenue de l'Observatoire, 75006 Paris, France – virginie.lasserre@u-paris.fr

Une expérience a conduit à l'échantillon ci-dessous.
14,4 **n** valeurs tirées au sort – **n** aléatoire 12,48

Quelle est la moyenne arithmétique de cet échantillon ?
Vous donnerez la réponse avec 5 chiffres après la virgule.

Réponse : **Réponse adaptée au tirage aléatoire**

Figure 1 : Trame d'exercice (à gauche) et ses variantes en nombre illimité (à droite) pour la compétence « calcul d'une moyenne »

- des questions calculatoires avec, par exemple, une trame pour la compétence « calcul de moyenne » et la génération de variantes (figure 1) ;
- des tests statistiques avec, par exemple, la compétence « construire un tableau d'ANVA à un facteur » dont les variantes peuvent se décliner à l'infini (figure 2) ;
- des interprétations graphiques avec, par exemple, une trame pour la compétence « interprétation d'une boîte à moustaches » et la génération de variantes sous la forme d'une figure où identifier la bonne représentation (figure 3) ;
- des sorties du logiciel R à interpréter (figure 4)...

Ces différents exemples, et bien d'autres, sont mis à disposition de nos étudiants lors de nos enseignements de statistique du L2 au M2 pour une auto-évaluation de leurs compétences. Le *package* a aussi été utilisé avec succès pour les évaluations de nos étudiants lors du confinement.

Bibliographie

- [1] Moodle : https://moodle.org/?lang=fr_fr
- [2] E. CURIS & V. LASERRE, SARP.moodle, <https://cran.r-project.org/package=SARP.moodle>

Le calcul des sommes de carrés pour 5 échantillons a conduit aux valeurs ci-dessous.

Échantillon	1	2	3	4	5
n_i	4	4	4	4	4
Somme des x²	44,934	15,176	4,525	49,403	10,902
Somme des x²	507,646	69,056	14,93	631,015	41,165

Construisez la table d'analyse de la variance permettant de comparer les espérances de ces échantillons.

Source des variations	Somme des carrés ddl	Moyenne des carrés	Critère de test
Factoriel			$f_{\text{obs}} =$
Résiduel			
Total			

Vous donnerez les valeurs avec 4 décimales

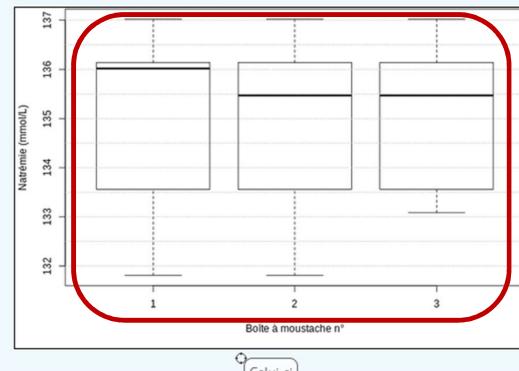
Figure 2 : Variante d'exercice pour la compétence « tableau d'ANVA à un facteur » avec génération aléatoire des échantillons

Les éléments variables sont cercles de rouge
Les réponses attendues dans les champs s'adaptent aux valeurs tirées pour générer l'exercice

Au cours d'une étude, la natrémie de 11 patients a été mesurée. Les quartiles de l'échantillon obtenu sont donnés dans le tableau ci-dessous.

min	Q ₁	Q ₂	Q ₃	max
131,81	133,56	135,47	136,14	137,02

Identifiez la boîte à moustache (boxplot) qui correspond à cet échantillon.
Faites glisser l'étiquette au centre de la bonne boîte à moustache.



Celui-ci

Figure 3 : Variante d'exercice pour la compétence « boîte à moustaches » avec génération aléatoire de l'échantillon

Les éléments variables sont cercles de rouge
Le nombre de boîtes peut varier

On soupçonne que l'espérance d'une mesure diffère entre deux populations.
Un premier échantillon de 6 individus a été réalisé dans la première population.
Un second échantillon de 15 individus a été réalisé dans la seconde population.
Un test statistique a été réalisé avec un logiciel, conduisant aux résultats ci-dessous.

> t.test(x1, x2, var.equal = TRUE)

Two Sample t-test

```
data: x1 and x2
t = 71.98, df = ?, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
```

On admettra que les valeurs proviennent d'une distribution gaussienne.

On a ici supposé les variances égales ou différentes

À la place de ?, il devrait y avoir

égales ou différentes

significatif ou non-significatif

Au risque $\alpha = 1\%$, le test est

Figure 4 : Variante d'exercice pour la compétence « interprétation d'une sortie de logiciel » avec génération aléatoire des deux échantillons

Les éléments variables sont cercles de rouge

La réponse attendue dans le champ et le choix de la bonne réponse dans les menus s'adaptent aux valeurs tirées pour générer l'exercice

Microstructure Information from Diffusion Imaging

Aymeric Stamm*

Abstract

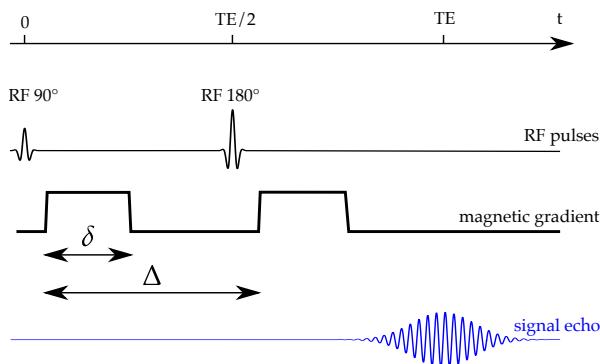
Diffusion magnetic resonance imaging is an *in-vivo* and non-invasive imaging modality that can be used to probe the microstructure of biological tissues. Magnetization is made sensitive to the diffusion of water molecules in the brain. Water trapped in tissues with impermeable membranes is subject to restricted diffusion, which depends on the geometry of the tissue. Brain white matter is essentially composed of axons and glial cells which form the connections between neurons. Axons can be modeled as cylindrically-shaped tissue compartments while glial cells can be modeled as spherically-shaped tissue compartments. A number of models have been devised in the literature to predict the magnetization decay induced by such compartments. These models are either based on the approximations of the solution of the Bloch-Torrey equation, which describes the evolution of the magnetization in the presence of diffusion or on the Gaussian phase approximation of the magnetization decay or on the restricted diffusion between two parallel planes. These models are often fitted to the diffusion signal to provide so-called microstructural parameters such as the axon diameter or the axon density. Yet, there is currently no user-friendly way of understanding whether the acquired data is sensitive enough to the microstructural parameters of interest. The present work is an attempt to fill in this gap, by providing the Shiny application **midi** and its companion eponymous **R package** that allow to simulate the MR signal for a given set of microstructural parameters. The application features theoretical summaries of the existing models along with key visualizations of the signal they predict to help understanding the sensitivity of the MR signal to a parameter of interest, for a given set of experimental conditions.

Keywords : brain imaging - diffusion MRI - biophysical tissue models - Shiny - R6.

Theory

Diffusion MRI

Images of diffusion MRI are often acquired using the pulse-gradient spin-echo (PGSE) sequence. The sequence is based on the application of two diffusion-sensitizing gradient pulses in direction \mathbf{n} with intensity G during a pulse duration δ and separated by a diffusion time Δ . The following figure illustrates the sequence:



*Department of Mathematics Jean Leray, UMR CNRS 6629, Nantes Université, France, aymeric.stamm@cnrs.fr

Models of restricted diffusion

We focus here on the signal attenuation induced by diffusion in the presence of tissue compartments with cylindrical geometry which is a fair approximation of the axon geometry. Let $\mathbf{u} \in \mathbb{S}^2$ and $R > 0$ be the axis and the radius of the cylinder, respectively. Let also D_0 be the coefficient associated with free diffusion in the cylinder. The signal attenuation $S(\delta, \Delta, G)$ induced by diffusion in the presence of tissue compartments with cylindrical geometry is assumed to be the product of the signal attenuation $S_{\parallel}(\delta, \Delta, G)$ induced by diffusion parallel to the cylinder axis and the signal attenuation $S_{\perp}(\delta, \Delta, G)$ induced by diffusion perpendicular to the cylinder axis (Assaf et al. 2004).

Diffusion parallel to the cylinder axis is assumed to be free and therefore usually modeled as a 1-dimensional Gaussian process inducing a mono-exponential signal decay, which reads:

$$S_{\parallel}(\delta, \Delta, G, \mathbf{n}; \mathbf{u}, D_0) = e^{-b(\delta, \Delta, G)D_0 \langle \mathbf{n}, \mathbf{u} \rangle^2},$$

where $b(\delta, \Delta, G) = \gamma^2 \delta^2 G^2 (\Delta - \delta/3)$ is the b-value, with γ the gyromagnetic ratio of the proton.

Diffusion perpendicular to the cylinder axis is assumed to be restricted and a number of models have been proposed to describe the signal attenuation induced by this type of diffusion. They are either based on approximations of the solution of the Bloch-Torrey equation under the narrow-pulse approximation (NPA), which assumes that no-diffusion occurs during the pulse duration δ (Söderman and Jönsson 1995; Callaghan 1995), or on the Gaussian phase approximation (GPA) of the magnetization decay (Neuman 1974; Vangelderden et al. 1994) or on a geometry simplification pertaining to studying restricted diffusion between two parallel planes (Stanisz et al. 1997).

A common feature of these models is that they predict the signal attenuation to be a function of the experimental parameters δ , Δ and G and the tissue properties such as the axon axis \mathbf{u} , the axon radius R , the axon density or the free diffusion coefficient D_0 . However, there is currently no user-friendly way of understanding whether the acquired data is sensitive enough to the tissue properties of interest. The following sections describe the structure of the Shiny application **midi** and its companion eponymous R package that allow to simulate the MR signal for a given set of tissue properties and experimental conditions.

The **midi** application

Theory

A first tab of the application provides a tabset of focused one-page summaries on

1. *Diffusion imaging* to introduce the user to the basics of diffusion MRI and the notation of the experimental parameters;
2. *Models of restricted diffusion* to introduce the user to the existing models of restricted diffusion and the notation of the tissue properties;
3. *CHARMED* to introduce the user to the CHARMED model and optimized acquisition schemes for estimating the axon diameter and the axon density (Assaf et al. 2004);
4. *References values* to provide the user with reference or range values of the tissue properties and experimental parameters.

Axon diameter

A second tab of the application allows the user, through an embedded Shiny application, to simulate the MR signal for a given set of tissue properties and experimental conditions and to visualize the signal decay as a function of axon diameter. This is meant to understand the sensitivity of the MR signal to the axon diameter. In particular, if the signal decay remains approximately constant in the range of biologically plausible axon diameters, then one can conclude that the MR signal produced by a given set of experimental conditions and other tissue properties is not sensitive to the axon diameter.

Specifically, the user can control the experimental parameters δ , Δ and G and the angle between the axon axis \mathbf{u} and the direction \mathbf{n} of the applied diffusion-sensitizing gradient but the other tissue properties are set to biologically plausible values internally.

Axon density

A third tab of the application allows the user, through an embedded Shiny application, to simulate the MR signal for a given set of tissue properties and experimental conditions and to visualize the signal decay as a function of axon density. This is meant to understand the sensitivity of the MR signal to the axon density. In particular, if the signal decay remains approximately constant in the range of biologically plausible axon densities, then one can conclude that the MR signal produced by a given set of experimental conditions and other tissue properties is not sensitive to the axon density.

Specifically, the user can control the experimental parameters δ , Δ and G , the angle between the axon axis \mathbf{u} and the direction \mathbf{n} of the applied diffusion-sensitizing gradient, the axon diameter and the free diffusion coefficient D_0 . The user can also apply a tortuosity model to account for the effect of the extra-axonal space on the MR signal.

The midi R package

The package provides the user with a set of functions to simulate the MR signal. The package relies on the R6 class system from the R package **R6** to define classes for each model of restricted diffusion. The package also relies on the R package **ggplot2** to visualize the MR signal decay. The package is available on [GitHub](#) and can be installed using the R package **devtools** as follows:

```
# install.packages("devtools")
devtools::install_github("astamm/midi")
```

Classes are designed to separate the tissue model parameters from the experimental parameters. In particular, each class has a constructor that takes as arguments the geometrical properties of the tissue in which diffusion is being modeled. There is then a single method `get_signal()` which takes as arguments the experimental parameters δ , Δ and G and returns the MR signal.

References

- Assaf, Yaniv, Raisa Z Freidlin, Gustavo K Rohde, and Peter J Basser. 2004. “New Modeling and Experimental Framework to Characterize Hindered and Restricted Water Diffusion in Brain White Matter.” *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 52 (5): 965–78. <https://doi.org/10.1002/mrm.20274>.
- Callaghan, Paul T. 1995. “Pulsed-Gradient Spin-Echo NMR for Planar, Cylindrical, and Spherical Pores Under Conditions of Wall Relaxation.” *Journal of Magnetic Resonance, Series A* 113 (1): 53–59.
- Neuman, CH. 1974. “Spin Echo of Spins Diffusing in a Bounded Medium.” *The Journal of Chemical Physics* 60 (11): 4508–11. <https://doi.org/10.1063/1.1680931>.
- Söderman, Olle, and Bengt Jönsson. 1995. “Restricted Diffusion in Cylindrical Geometry.” *Journal of Magnetic Resonance, Series A* 117 (1): 94–97.
- Stanisz, Greg J, Graham A Wright, R Mark Henkelman, and Aaron Szafer. 1997. “An Analytical Model of Restricted Diffusion in Bovine Optic Nerve.” *Magnetic Resonance in Medicine* 37 (1): 103–11. <https://doi.org/10.1002/mrm.1910370115>.
- Vangelderden, P, D DesPres, PCM Vanzijl, and CTW Moonen. 1994. “Evaluation of Restricted Diffusion in Cylinders. Phosphocreatine in Rabbit Leg Muscle.” *Journal of Magnetic Resonance, Series B* 103 (3): 255–60. <https://doi.org/10.1006/jmrb.1994.1038>.

Esquisse, un outil de visualisation

Victor Perrier*

Résumé (max 300 mots)

Le paquet {esquisse} permet via une application web de construire des graphiques avec {ggplot2}, tout en générant le code permettant de reproduire ces graphiques. Esquisse est disponible à la fois dans une application disponible en ligne, dans une extension RStudio, ou bien via des modules {shiny} permettant sa réutilisation dans n'importe quelle application.

Mots-clefs : Visualisation – ggplot2 – shiny – application – apprentissage – métaprogrammation

Développement

Esquisse permet aux utilisateurs de créer des graphiques avec le paquet {ggplot2} via une interface graphique tout en générant le code pour recréer ces graphiques dans un script, mais aussi via l'utilisation des modules {shiny} d'intégrer esquisse dans une application {shiny}.

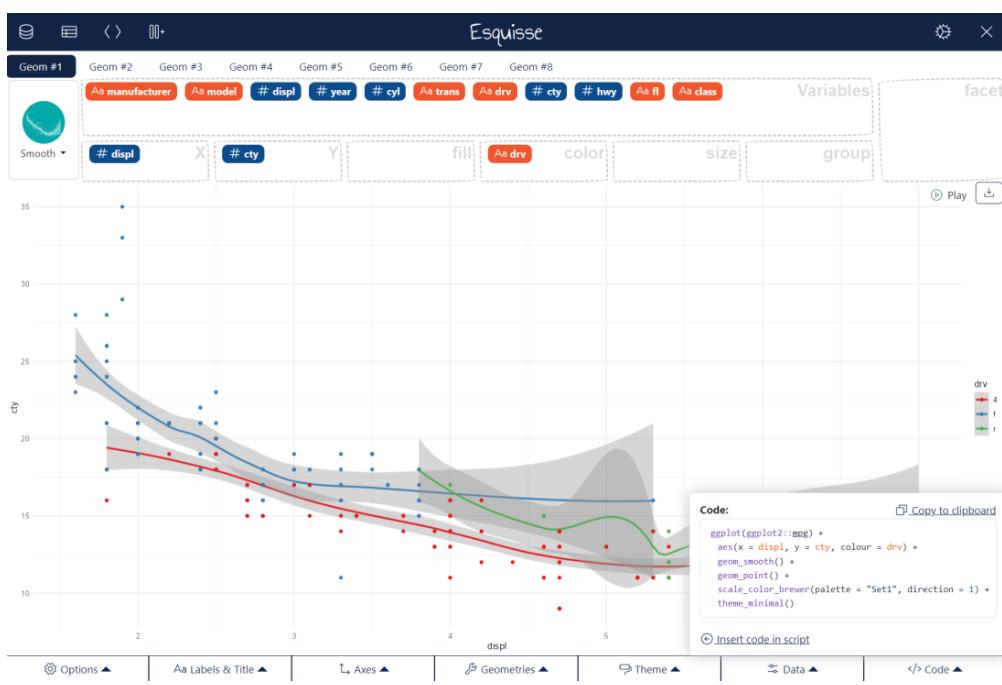


Fig1. Interface graphique d'{esquisse}

* dreamRs, victor.perrier@dreamRs.fr

Esquisse est disponible sur le CRAN depuis 2018 sous la forme de paquet R, le code source est quant à lui disponible sur GitHub (<https://github.com/dreamRs/esquisse>), mais depuis cette année une application web est également disponible en ligne, permettant de l'utiliser sans avoir besoin d'installer R, nous présenterons cette application lors de cette présentation.

Nous présenterons également les dernières nouveautés intégrées à {esquisse} récemment :

- La possibilité de rendre les graphiques interactifs grâce au paquet {plotly} pour afficher une étiquette avec les valeurs des points survolés par la souris ou encore zoomer sur le graphique.
- Le mode “multi-geom” permettant de cumuler plusieurs `geom_*`, ayant chacun leur propre mapping.
- L’intégration de nouveaux outils pour manipuler les données directement dans l’application venant du paquet {datamods}, tels que :
 - o Un module pour créer de nouvelles variables à partir d’une expression R saisie par l’utilisateur
 - o Un module pour découper une variable numérique en plusieurs intervalles
 - o Un module pour réordonner les niveaux d’une variable de type ‘factor’.

Nous évoquerons également deux composants majeurs constituant {esquisse} :

- La génération de code avec le paquet {rlang} et la création de `call` et d’`expression` pouvant être plus tard évaluées.
- Le mode international d’{esquisse} permettant l’utilisation de l’application dans 12 langues différentes, grâce à des contributions de la communauté.

Nous conclurons cette présentation sur les évolutions futures prévues pour le paquet {esquisse} et celles souhaitées par la communauté.

Références

Meyer F, Perrier V (2024). *_esquisse: Explore and Visualize Your Data Interactively_*. R package version 1.2.0, <<https://CRAN.R-project.org/package=esquisse>>.

Perrier V, Meyer F, Goumri S, Abeer Z (2024). *_datamods: Modules to Import and Manipulate Data in 'Shiny'_*. R package version 1.5.0, <<https://CRAN.R-project.org/package=datamods>>.

H. Wickham. *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York, 2016.

Henry L, Wickham H (2024). *_rlang: Functions for Base Types and Core R and 'Tidyverse' Features_*. R package version 1.1.3, <<https://CRAN.R-project.org/package=rlang>>.

webR, et le futur des apps web avec R

Colin Fay, ThinkR
colin@thinkr.fr

Résumé (max 300 mots)

L'un des grands plaisirs de l'ingénierie logicielle est que les choses bougent sans arrêt. De nouvelles technologies, de nouveaux langages, de nouveaux frameworks... De temps en temps, parmi ces nouveautés, de nouvelles choses émergent et changent la manière dont nous construisons des produits.

Au cours des dernières années dans le monde de R, nous avons construit et déployé des applications web et des API de manière assez stable : en construisant des applications {shiny} avec des frameworks comme {golem} ou {rhino}, des API avec {plumber}, pour ensuite les envoyer sur un serveur qui peut lancer R et rendre notre code disponible au monde. Ces derniers mois, une nouvelle technologie est apparue dans le "R World" : webR, une version de R compilée pour WebAssembly (WASM), permettant d'exécuter R directement dans le navigateur ou à l'intérieur de NodeJS, sans avoir besoin d'une installation R. Ce qui ouvre un champs des possibles incroyable pour le développement web, JavaScript étant l'outil de choix pour la construction d'applications et d'API.

Dans cette présentation, Colin commencera par expliquer ce qu'est webR, projet développé par l'équipe Posit, et comment cela changera notre façon de penser la construction et le déploiement de code R pour le web.

Il présentera ensuite deux packages NodeJS expérimentaux sur lequel il travaille : webrcli, interface en ligne de commande offrant un "kit de démarrage" pour des projets avec NodeJS / webR, et spidyr, une boîte à outils pour interagir avec webR depuis une application NodeJS.

Et enfin, Colin se concentrera également sur les défis qui se poseront avec webR, et comment nous construirons des applications web avec R à l'avenir.

Mots-clefs (3 à 5) : webR, API, shiny, développement web

Développement

L'un des aspects les plus exaltants du métier d'ingénieur logiciel réside dans sa nature perpétuellement changeante. Dans cet univers en constante évolution, de nouvelles technologies, de nouveaux langages et de nouveaux frameworks émergent régulièrement, apportant avec eux des perspectives nouvelles et des possibilités inédites pour la conception et le développement de produits et de service. R n'est pas étranger à cette constante évolution, et s'aventure d'année en année sur des terrains nouveaux, tant pour l'analyse de données que pour la construction d'outils rendant accessible au monde extérieur les résultats de nos travaux : on pense ici à {shiny} bien sûr, mais aussi à {plumber}, {markdown}, quarto... tant de technologies appelées depuis R et qui permettent de servir sur le web du code directement sur le web, plutôt que dans un terminal.

Sur le web, l'information se consomme de deux grandes manières : via des applications web, ou via des systèmes d'API REST. Ces dernières années, nous avons pu constater une certaine stabilité dans la manière dont nous construisons et déployons ces applications et services :

-
- Pour des apps web, avec des outils bien établis tels que {shiny}, en association avec des frameworks comme {golem} ou {rhino}. Avec eux, nous avons pu créer des applications interactives et des interfaces utilisateur dynamiques
 - Des solutions telles que {plumber} nous ont permis de développer des API robustes et performantes.

Ces applications étaient ensuite déployées sur des serveurs capables de lancer du code R, rendant ainsi le code R accessible à un large public, via des services comme Posit Connect, des technologies comme Docker, et d'autres.

Et si nous étions aujourd'hui à l'aube d'un nouveau bouleversement ?

Car oui, même dans ce paysage relativement stable, des bouleversements se produisent. Ces derniers mois ont vu l'émergence d'une technologie qui devrait révolutionner la façon dont nous faisons du web avec R : webR, outil développé par l'équipe open source de chez Posit. Un terme mystérieux qui désigne la compilation de R pour WebAssembly (WASM), un outil destiné à rendre possible l'exécution de code qui était jusqu'ici traditionnellement exécuté sur le serveur, directement dans les navigateurs web modernes. Une exécution également possible au sein de l'environnement NodeJS, le runtime JavaScript sur serveur.

Cette innovation ouvre des perspectives incroyables pour le développement web, en permettant l'exécution de code R directement dans le navigateur, sans nécessiter une installation préalable de R sur l'appareil de l'utilisateur. Avec JavaScript qui continue d'être le langage de programmation dominant pour la construction d'applications web et d'API, cette convergence entre R et WebAssembly promet de révolutionner la façon dont nous pensons et concevons les applications web basées sur R.

Dans le cadre de cette présentation, Colin nous guidera à travers les méandres de webR, nous offrant un aperçu de cette technologie émergente et de son impact sur le développement web avec R. Il commencera par expliquer en quoi consiste exactement webR, en mettant en lumière ses avantages et ses implications pour notre manière de créer et de déployer des applications web.

Colin nous présentera deux outils expérimentaux sur lesquels il est en train de travailler : webrcli, interface en ligne de commande permettant de faciliter la création de projets avec NodeJS / webR, et spidyr, une boîte à outils pour interagir avec webR depuis une application écrite en NodeJS. Ces outils permettent de créer des applications NodeJS capables d'appeler du code R via webR, offrant ainsi une solution élégante pour intégrer R dans des environnements JavaScript et tirer parti des capacités avancées de cette combinaison de technologies.

Imaginez un instant : la puissance et la simplicité de la manipulation de données dans R, combinée à l'incommensurable paysage d'outils pour le web de NodeJS — le tout, dans le but inavoué de réconcilier les équipes de développeurs web et les équipes de développeurs R, et de leur donner un cadre de travail commun.

Enfin, Colin abordera les défis potentiels associés à l'adoption de webR, notamment en ce qui concerne la gestion des performances, la sécurité et la maintenance à long terme. Il partagera sa vision sur la manière dont nous pourrons surmonter ces défis et construire des applications web robustes et évolutives avec R et JavaScript à l'avenir.

Références :

- <https://webr.r-wasm.org/>
- <https://github.com/ColinFay/webrcli>
- <https://github.com/ColinFay/spidyr>
- <https://colinfay.me/webrcli-and-spidyr/>
- <https://github.com/ColinFay/webrspongebob>

Apprendre R et les statistiques... grâce à R

Philippe Grosjean 1* Guyliann Engels 2†

Résumé

Un apprenant est confronté à plusieurs difficultés lorsqu'il débute dans l'analyse de ses données avec R. Il doit maîtriser un environnement logiciel, un langage de programmation, en même temps que les concepts statistiques. Il doit également apprendre à bien formuler ses questions et à interpréter les résultats obtenus.

Dans le cadre du cours de Science des Données biologiques à l'Université de Mons en Belgique, nous avons développé une approche graduelle qui met en œuvre des outils pédagogiques variés existants (comme `{learnr}`, `{gradethis}`, `{quarto}` ou `{ghclass}`/GitHub Classroom), mais aussi originaux : `{learnitdown}` et `{learnitgrid}`. Nous verrons comment ces différents packages R contribuent à créer un matériel pédagogique interactif et évolutif. Il permet un voyage initiatique qui débute dans un cours en ligne (<https://wp.sciviews.org>), se poursuit avec des tutoriels `{learnr}`, pour finalement aborder des projets GitHub/Quarto cadrés avec `{learnitgrid}`. Au bout du voyage, les étudiants sont capables d'analyser leurs données de manière autonome et de bien communiquer leurs résultats.

Mots-clés : Apprentissage hybride - Statistique - Science des données - Outils pédagogiques - Cours en ligne interactif

Développement

La pédagogie universitaire classique sous forme de cours en amphithéâtre où les étudiants écoutent passivement suivis de séances d'exercices pratiques a montré ses limites lors des périodes de confinement imposées par le Covid-19. Un enseignement à distance et des classes inversées ou hybrides (présentiel et distanciel à part à peu près égale) en utilisant du matériel pédagogique préenregistré et des exercices interactifs ont alors été développés. Ces approches différentes ont montré leur efficacité et leur intérêt pour les étudiants. Elles ont également permis de réfléchir à une autre pédagogie, notamment dans l'enseignement de la science des données biologiques à l'Université de Mons en Belgique (UMONS).

Nous détaillons dans cette présentation notre approche pédagogique en mode hybride dont l'originalité tient en la combinaison d'exercices en quatre niveaux de difficulté croissante. Ces exercices se basent sur des packages R qui fournissent les ressources nécessaires pour les mettre en œuvre tels `{learnr}`, `{gradethis}` et `{shiny}`. Nous en avons écrit d'autres pour les compléter : `{learnitdown}`, `{learnitdashboard}`, `{learnitgrid}` et `{BioDataScience|1|2|3}`.

Les quatre cours de science des données à l'UMONS se distribuent de la seconde année universitaire à la cinquième et dernière année du cursus de biologie, donc à cheval sur le Bachelier (la licence en France) et le Master pour un total de 17 crédits ECTS. La matière est découpée en 30 modules qui correspondent à un travail s'étalant sur deux semaines à chaque fois. L'apprentissage est actif et progressif selon quatre niveaux de difficulté croissante.

- **Niveau 1 :** Les étudiants préparent la matière du cours en ligne à leur rythme chez eux et disposent d'exercices H5P (<https://h5p.org>) et d'applications Shiny pour vérifier leur compréhension des concepts abordés.
- **Niveau 2 :** Ils testent ensuite leurs acquis à l'aide de tutoriels `{learnr}` et commencent à coder en R dans ces tutoriels interactifs, toujours avant les séances en présentiel. La correction est automatisée à l'aide de `{gradethis}` de sorte qu'ils ont un retour immédiat et des suggestions pour corriger par eux-mêmes leur code et leurs réponses.

*Service d'écologie numérique, Institut Complexys & Infortech, Université de Mons, Belgique, philippe.grosjean@umons.ac.be
†Service d'écologie numérique, Institut Complexys & Infortech, Université de Mons, Belgique, guyliann.engels@umons.ac.be

Pour les exercices de niveau 1 et 2, l'activité des étudiants est enregistrée dans une base de données de “learning analytics” MongoDB grâce au package R {learnitdown}. La progression dans les exercices est gratifiée de 5% des points sur la note finale. Ce “bonus” incite fortement les étudiants à réaliser tous les exercices chez eux et nous enregistrons un taux de participation de l’ordre de 98%. Bien sûr, cela les incite également à étudier la matière, prérequis obligatoire pour être capable de réaliser ces exercices. Les étudiants arrivent en classe avec leurs questions sur la matière : nous insistons bien sur le fait qu'il est normal qu'ils n'aient pas tout compris et nous travaillons ensemble en présentiel les points à éclaircir.

- **Niveau 3 :** Après la séance de questions-réponses, les étudiants attaquent en présentiel des projets GitHub gérés à l'aide de GitHub Classroom et {ghclass}. Ils doivent analyser des données biologiques et rédiger un rapport en Quarto. À ce niveau, les projets sont individuels et guidés. Cela signifie que les différentes étapes de l'analyse sont suggérées dans le template du rapport Quarto et l'interprétation se fait en sélectionnant les phrases qui conviennent dans une liste à choix multiple. Ces projets sont corrigés de manière semi-automatique avec {learnitgrid}. Ce dernier gère aussi une batterie de tests écrits avec {testthat} que les étudiants peuvent utiliser pour vérifier leurs réponses directement. Un test incorrect est assorti d'un lien cliquable vers des suggestions pour corriger l'erreur. Ces projets comptent pour 10% des points.
- **Niveau 4 :** Des projets GitHub similaires à ceux du niveau 3, mais *non* guidés et réalisés par groupes de deux à quatre étudiants terminent la formation. Les instructions sont ici minimales et aucune batterie de tests ne vient épauler les étudiants. Ils sont donc placés en situation “réelle” à devoir analyser des données biologiques par eux-mêmes. Pour certains projets, ils doivent également choisir des données ouvertes qu'ils souhaitent traiter à partir de sites comme Zenodo (<https://zenodo.org>), Dryad (<https://datadryad.org>)... 20% des points sont attribués à ces projets de groupe.

Les projets GitHub sont corrigés par grille critériée avec des commentaires expliquant ce qui peut être amélioré. Pour chaque module, les étudiants travaillent dans les projets pendant deux séances totalisant 6h en présentiel et les complètent ensuite à domicile. À l'issue de ce travail, une interrogation écrite ou un exercice pratique sous forme de challenge (contre la montre ou compétition pour le meilleur modèle, par exemple) permet une évaluation individuelle de la progression et complète la note attribuée pour le module.

Le travail se fait dans RStudio sur le cloud (<https://saturncloud.io>) de sorte que les étudiants ont tous accès strictement aux mêmes ressources matérielles et à la même configuration logicielle sous Linux, et ce, qu'ils soient en classe ou chez eux. Ils posent leurs questions dans les “issues” des dépôts GitHub de leurs projets et ils ont accès à tout moment à leur progression dans les exercices et les projets (rapport de progression à la volée sous forme d'une application Shiny). L'apprentissage selon cette méthode a fait l'objet d'une publication qui détaille les résultats obtenus sur trois années successives (Engels, Grosjean, and Artus 2023).

Tout le matériel didactique développé dans le cadre de ces cours (y compris des centaines d'exercices et les templates d'une quarantaine de projets) est disponible dans l'organisation GitHub ‘BioDataScience-Course’ (<https://github.com/BioDataScience-Course>). Il est distribué sous license MIT ou Attribution-NonCommercial-ShareAlike 4.0 International selon le type de contenu. Le cours en ligne donne accès également à ce matériel dans son contexte à partir de <https://wp.sciviews.org>. Les packages R cités ici sont accessibles depuis CRAN (<https://cran.r-project.org>) ou depuis les organisations GitHub ‘SciViews’ ou ‘BioDataScience-Course’.

Référence

Engels, Guyliann, Philippe Grosjean, and Frédérique Artus. 2023. “Teaching Data Science to Students in Biology Using r, RStudio and Learnr: Analysis of Three Years Data.” *Foundations of Data Science* 5 (2): 266–85. <https://doi.org/10.3934/fods.2022022>.

Index

- Alglave Baptiste, 58, 59
Andrade Antonio, 74, 75
Aubert Julie, 54
Audigier Vincent, 52, 53
Barbieri Antoine, 11, 12
Barthelemy Tanguy, 33, 34
Bernard Jérémie, 58, 59
Bertrand Julie, 9, 10
Bichat Antoine, 54
Bocher Erwan, 58, 59
Bouhlel Nizar, 36, 37
Bouzillé Guillaume, 64, 65
Carayon David, 30, 31
Chion Marie, 21, 22
Comets Emmanuelle, 7–10
Curis Emmanuel, 87, 88
Dechaux Terence, 60, 61
Dorchies David, 27–29
Drouilhet Remy, 81–83
Dugas Julien, 76, 77
Durrieu Gilles, 50, 51
Engels Guyliann, 43, 44
Fay Colin, 94, 95
Fayette Lucie, 9, 10
Floc'hlay Swann, 19, 20
Friguet Chloé, 48, 49
Gavra Ioana, 46, 47
Girard Antoine, 16–18
Goumri Samra, 35
Gousseff Matthieu, 58, 59
Gramia Ion, 50, 51
Grosjean Philippe, 43, 44, 96, 97
Guhl Mélanie, 9, 10
Guichard-Sustowski Ketsia, 46, 47
Guyader Vincent, 78–80
Husson François, 48, 49
Irz Pascal, 27–29, 46, 47, 74, 75
Jacqmin-Gadda Hélène, 11, 12
Katsahian Sandrine, 41, 42
Lasserre Virginie, 87, 88
Laurencelle Marc, 70–73
Lavalley-Morelle Alexandra, 7, 8
Lavenu Audrey, 41, 42
Le Marrec Loïc, 46, 47
Le Saux - Wiederhold Elisabeth, 58, 59
Leconte François, 58, 59
Legris Maxime, 62, 63
Leroux Riwan, 68, 69
Leroy Arthur, 21, 22
Lohier Théophile, 70–73
Maigné Elise, 30, 31, 57
Mariadassou Mahendra, 66, 67
Mary-Huard Tristan, 5
Matabos Marjolaine, 68, 69
Mentré France, 7, 8
Midoux Cédric, 66, 67
Mondy Cédric, 74, 75
Mullaert Jimmy, 7, 8
Murris Juliette, 41, 42
Ngounou Bakam Yves Ismaël, 55, 56
Nowak Céline, 74, 75
Olivier Fanny, 32
Perrier Victor, 35, 92, 93
Philibert Aurore, 76, 77
Pierre-Jean Morgane, 64, 65
Pommeret Denys, 55, 56
Proust-Lima Cécile, 6
Raillard Nicolas, 45
Rey Jean-François, 30, 31
Richard Benoît, 74, 75
Rodrigues Coelho Bruno André, 2–4, 13–15
Salmon Maëlle, 85, 86
Sanchez Isabelle, 30, 31
Santagostini Pierre, 36, 37
Sarrazin Jozée, 68, 69
Say Yann, 23, 24
Stamm Aymeric, 89–91
Straboni Camille, 38–40
Thelen Véronique, 46, 47

Tran Joseph, 30, 31

Tzourio Christophe, 11, 12

Vaugoyeau Marie, 25, 26

Vroylandt Thomas, 84

Sponsors



