R, pour un écosystème du traitement des données ? L'exemple de la linguistique.

Nicolas Ballier

Ce chapitre reprend les grandes lignes de la conférence invitée donnée aux journées « Données, Métadonnées des corpus et catalogage des objets en sciences humaines et sociales » organisées les 6 et 7 juin 2016 à la MSHS de Poitiers. Le chapitre est paru en 2019 aux Presses Universitaires de Rennes dans le volume L'enjeu des métadonnées dans les corpus textuels. Un défi pour les sciences humaines*, sous la direction de Paul Caron, de* Rodolphe Defiolle et de Marie-Hélène Lay (pp. 97-118.)

C'est cette version de référence qui doit servir pour les citations :

Ballier, N. (2019) R, pour un écosystème du traitement des données ? L'exemple de la linguistique, in Paul Caron, Rodolphe Defiolle et Marie-Hélène Lay (eds) *L'enjeu des métadonnées dans les corpus textuels. Un défi pour les sciences humaines*, Presses Universitaires de Rennes, pp. 97-118.

VERSION AUTEUR

Nicolas Ballier (Université de Paris (Diderot), CLILLAC-ARP)

R, pour un écosystème du traitement des données ? L'exemple de la linguistique

Introduction

L'objectif de ce chapitre est en partie de convaincre de l'intérêt de R et de son environnement dans une perspective d'importation, de visualisation, et de traitement (notamment statistique) des données. Je souhaiterais décrire les conséquences pour l'analyse linguistique du recours croissant de certains linguistes à R, à la fois langage de programmation et logiciel. Il s'agit de contribuer partiellement à la micro-histoire de la troisième révolution de la grammatisation (Auroux, 1991), celle qui s'opère autour des corpus, des approches informatiques et du développement d'internet. Je cherche à déployer une épistémologie du logiciel d'analyse linguistique, croisant le statut des catégories linguistiques, leurs ontologies sous-jacentes et celles inhérentes aux pratiques des linguistes (Ballier, 2004). La tâche est ici plus complexe car le logiciel est également un langage de programmation, qui a permis de nombreuses incorporations d'autres logiciels, via une intégration dans le même langage de programmation (R). *R* renvoie ici aussi bien au langage de programmation qu'à l'environnent de travail (le logiciel R, en ligne de commandes, ou son interface graphique la plus répandue, RStudio, qui propose également une coloration syntaxique).

On a aussi en partie la linguistique de sa technologie, de ses conditions de production. Le logiciel R, apparu au tournant des années 2000, à peu près en même temps que Praat, logiciel d'analyse de la parole devenu standard de fait, indépendamment des qualités des concurrents, n'a pas (encore) acquis pour la linguistique dans son ensemble le statut que Praat a acquis au sein de la communauté des phonéticiens. Il me semble, néanmoins, possible de parler de R pour la linguistique, même s'il y a aussi une historicité dans la propagation des usages de R, les phonéticiens et les psycholinguistes sont davantage rompus aux données statistiques (et à R) que les spécialistes de pragmatique ou d'histoire de la linguistique. La pragmatique (même computationnelle) ou la philosophie du langage ne disposent pas encore, à ma

connaissance, de bibliothèques de programmes en R (appelées 'packages') spécifiques, mais toute une série de domaines en ont développé[1]. J'illustrerai par quelques exemples ce que R 'fait' aux données linguistiques (et aux métadonnées), comment, au-delà d'une éventuelle inféodation à la linguistique quantitative, ce langage de programmation et son arsenal de bibliothèques est susceptible de contribuer à changer certaines modalités de l'analyse linguistique, voire d'un certain rapport au savoir. L'objectif est aussi de suggérer, que, à condition de disposer de données et pas d'un seul exemple de linguiste du fauteuil (*armchair linguist*), on est susceptible de repenser à nouveaux frais la linguistique comme une discipline, la bibliothèque des packages offrant une version plausible d'une encyclopédie version 2.0.

Il s'agit donc également de s'efforcer de produire une réflexion sur l'évolution des conditions de production de la connaissance, sur ce qu'il conviendrait d'appeler avec Michel Pêcheux les matérialités discursives 2.0. On voudra bien excuser le terme d'*écosystème,* qui pourrait être perçu comme une concession à la modernité la plus rance. Il ne s'agit pas seulement d'avoir du 'big data' et 'des humanités numériques' plein la bouche, mais s'efforcer de les penser, non pas comme une concession obligatoire à l'ère du temps, mais comme la nécessité de réfléchir à une partie du monde qui arrive (voire, est déjà là, dans d'autres régions du globe), comme un ensemble de pratiques matérielles ayant des incidences sur la conceptualisation des objets de recherche. Ce terme d'*écosystème* est une manière commode de référer à l'interaction entre les données, R et les différents logiciels auxquels il donne accès via des passerelles (*wrappers*) qui permettent de « communiquer » avec des programmes écrits dans d'autres langages. De ce point de vue, *l'écosystème* renvoie en partie à la relation entre données et métadonnées. Je cherche à décrire une partie de l'interaction entre les données linguistiques (en amont de R) et R et de l'interaction entre R et ses sorties (en aval de R), avec, au milieu, R et ses bibliothèques de programmes (packages). Le point de vue est à la fois interne (je suis un linguiste) et en partie externe (je m'interroge sur R, son écosystème et ses incidences sur les pratiques d'une certaine linguistique).

A partir de la description sommaire de quelques chaînes de traitements des données dans l'analyse linguistique, je proposerai, à partir de ces exemples de pratiques d'analyse, une réflexion sur la manière dont changent nos rapports aux données et la manière dont nous abordons nos objets de recherche. Que fait le linguiste quand il utilise R ? Dans un premier temps, je caractériserai quelques propriétés de R pour l'analyse linguistique. Dans un second temps, je montrerai que le traitement des données linguistiques a poussé certains programmeurs à simplifier certains usages de R, en tout cas à accommoder R pour mieux 'gérer' les données linguistiques. Je caractériserai cette évolution à partir de la comparaison de deux manuels d'initiation à la linguistique quantitative (Baayen, 2008 et Levshina, 2015), qui révèlent une périodisation des pratiques, même sur le temps court. Puis, je proposerai une analyse de conséquences probables (et pour certaines, avérées) de l'utilisation de R pour l'analyse de données linguistiques. J'opposerai des conséquences sur l'amont (donc, en partie, sur l'usage des métadonnées) et des conséquences sur l'aval. J'esquisserai quelques conséquences épistémologiques possibles de ce rapport aux données, dans la conjoncture de la troisième révolution de la grammatisation. Enfin, je dessinerai les contours d'une nouvelle praxis des données, conséquences des mutations en cours.

---

[1] Pour un graphe montrant le développement exponentiel des packages de R: http://www.r-bloggers.com/rs-growth-continues-to-accelerate/ (dernière consultation le 23 août 2017).

1. Quelques propriétés de l'écosystème de R pour le traitement des données

Je ne reviendrai pas ici sur les propriétés intrinsèques du langage de programmation, sa genèse et ses idiosyncrasies, ni sur le type d'entités (objets) qu'il permet de manipuler et encore moins sur ce qui fait son succès[2] ou ses limites[3].

1. Une décentralisation réticulaire : les packages

Je propose une rapide caractérisation de la manière dont les linguistes ont pu s'approprier R en me centrant, non sur les packages spécifiquement crées pour les besoins des linguistes, mais en intégrant la démarche des linguistes au fonctionnement plus classique de R : une sorte de logique décentralisée dont l'essentiel consiste en l'adjonction de packages. On peut chercher à décrire l'ensemble de bibliothèques de R comme un graphe où figurent les principaux points nodaux des bibliothèques les plus importantes[4]. C'est ainsi que, pour l'extraction des données du Web, toute une série de bibliothèques a été crée pour récupérer les données sous Twitter, moyennant le recours aux packages {curl} (Ooms, Wickham, 2016), {httr} (Wickham, 2016a) et {twittR} (Gentry, 2016). Les interfaces graphiques pour la représentation de réseaux ont été implémentées ({igraph}, Csardi, Nepusz, 2006), pour la visualisation des réseaux). Les librairies sont souvent accompagnées de sites extrêmement clairs et particulièrement riches et pédagogiques, tels le <http://rmarkdown.rstudio.com/>, qui explicite les procédures d'exportations de documents, constituant ainsi comme un véritable logiciel intégré de fichiers PDF ou de présentations. Le stade ultime est l'incorporation dans un environnement de programmation (type Vim), qui permet de se passer de toute interface graphique.

La documentation des librairies est organisée par ordre alphabétique des fonctions et des jeux de données. Si le nom du package donne parfois droit à une entrée spécifique dans la documentation, la pratique n'est pas toujours très éclairante. A l'inverse des bibliothèques Matlab (fort chères, et d'abonnement annuel), les bibliothèques de R sont gratuites, mais leur documentation est fort inégalement détaillée. L'utilisateur lambda est ponctuellement protégé par la communauté, mais uniquement du point de vue de l'opérationnalité : le CRAN[5] vérifie l'exécution des routines (et met à l'encan des archives ce qui n'est plus compatible), l'absence de messages d'erreurs, mais pas la validité

---

[2] Pour un comparatif des points forts et faibles de R, ainsi que pour un panorama synthétique des différentes méthodes abordées en linguistique quantitative, on se reportera avec profit à Mizumoto, Plonsky (2014). La richesse des visualisations y est démontrée par l'exemple, à partir de l'incrustation d'informations supplémentaires sur les graphiques sous R. La flexibilité y est illustrée par l'interfaçage avec un serveur apache et des bases de données MYSQL comme dans le cas de la plate-forme de tests en ligne CONCERTO, développée à l'Université de Cambridge (http://www.psychometrics.cam.ac.uk/newconcerto), qui permet de créer ses tests psycholinguistiques en ligne (dernière consultation le 23 août 2017).

[3] Dans une conférence disponible sur Youtube, John Cook explique à quel point R peut apparaître insatisfaisant pour un programmeur en tant que langage de programmation, sorte de 'Comment réussir quand on est mal fichu' (« quirky, flawed and an enormous success », Cook, 2013).

[4] http://blog.revolutionanalytics.com/2014/11/a-look-at-the-igraph-package.html (dernière consultation le 23 août 2017)

[5] Le CRAN (Comprehensive R Archive Network), https://www.r-project.org/ (dernière consultation le 23 août 2017).

scientifique des fonctions implémentées. Rien n'interdirait à un plaisantin d'implémenter dans un package hébergé par le CRAN une fonction d'addition qui procèderait dans le codage du package à de la soustraction ou à de la multiplication. Rien n'assure de la justesse et de la pertinence du codage, si ce n'est la pratique de versionnage, les interactions avec les utilisateurs et les forums, forme 2.0 de *la communauté qui vient*.

2. Les chaînes de traitement des données : *Input/process/output* et ses avatars

On peut décrire les trois étapes classiques du rapport entre les données et un logiciel comme trois temps : *input/process/ouput.* Sous R, cette trilogie se décline en importation des données (et nous verrons les enjeux en termes de format de données privilégiés), visualisation et traitement statistique des données. C'est la troisième étape qui a été la plus riche en évolutions ces dernières années : les résultats ne sont plus seulement exportables sous forme de graphiques, de tableaux de données résultantes ou de simples résultats statistiques (p-valeur et autres mesures de significativité). C'est ce qui participe pleinement d'un écosystème : le code de R peut être conservé sous forme de script, et, par cascades de bibliothèques, s'intégrer dans une véritable chaine de production intellectuelle.

En amont, les scripts sont intégrés dans des ouvrages (ou dans des cours), déposés sur des serveur d'échanges de code et de données (Github) et servent de dépôt pour les versions béta de packages. Ils participent d'une économie des échanges et du partage sur le mode collaboratif (on peut littéralement 'repiquer' du code, l'améliorer et le rendre de nouveau accessible à la communauté pour permettre des améliorations successives sur ces plateformes d'échange numérique comme Github[6]). De nombreux blogs postent en ligne des analyses critiques, des études de cas qui comprennent des lignes de code R, voire des liens vers le script entier.

[6] Il y a une historicité à cette pratique: Github est une plateforme connue, mais ce n'est pas la première, d'autres l'ont précédée, voire d'autres la suivront. Cette relative volatilité du code et l'obsolescence apparemment inévitable d'un script R (trop lié au package, aux dépendances, aux versions Java, entre autres, pour être complètement pérenne de manière absolue et autonome) rend nécessaire une réflexion d'ensemble sur l'archivage et la conservation du code, ce qui est l'un des enjeux du projet Software Heritage (https://www.softwareheritage.org). Il s'agit d'organiser la patrimonialisation du code, en archivant les versions successives de scripts et des divers états des langages de programmation. L'interface d'interrogation de tous ces langages de programmation n'est pas encore finalisée, mais elle ne manquera pas d'intérêt pour les linguistes. Cette remarque n'est pas gratuite : l'un des enjeux de l'open data est la possibilité de répliquer à l'identique et forcer de constater que le code contenu dans Baayen (2008) n'est plus aujourd'hui intégralement exécutable en raison de la disparition du package {Design}. D'une certaine manière, un tel degré de dépendance à la technologie (aux mises à jour) est relativement inédit, mais ce n'est pas un cas isolé. Lorsque la société xwaves a fait faillite, une partie des laboratoires de phonétique a perdu son outil de consultation privilégié de certains corpus oraux. Daniel Hirst a ainsi dû faire réaligner l'intégralité de tout le corpus MARSEC en l'associant à deux projets doctoraux pour réaligner et ré-annoter ce qui est devenu le AIX-MARSEC (Auran et al. 2004), mais toute une partie des annotations grammaticales et prosodiques du corpus initial (Knowles et al. 1986) ont été perdues dans l'opération.

En aval, R peut maintenant servir à rédiger sa thèse, son diaporama ou son article en couplant R avec LaTeX. On peut structurer sa présentation de conférence, puis la faire figurer sur la Toile[7], mettre en ligne des représentations 3D et interactives (où l'utilisateur peut lui même choisir son mode de visualisation de données ou le type d'analyse qu'il souhaite privilégier (package {Shiny}, Chang *et al*., 2015). On peut même construire sa page web (par exemple via les espaces mis à disposition par RStudio[8]), voire structurer le site sur lequel figure l'ouvrage décrivant les fonctionnalités de R[9]. C'est quasiment un cas de réflexivité du producteur au consommateur de code, lui même interpellé pour devenir producteur à son tour. Il y a bien tout un écosystème autour du code sous R, environnement de travail véritablement lié au web, avec les inconvénients des nombreuses mises à jour (et des incertitudes sur les bonnes versions des packages qui permettent de faire effectivement tourner le code que l'on vient de trouver en ligne), mais aussi avec de possibilités vertigineuses de gains de productivité. De manière anecdotique, il manque parfois des lettres à l'alphabet pour permettre à Stefan Gries d'indiquer l'ensemble des articles parus ou à paraître en une année. De manière plus sérieuse, tout un écosystème de production et de publication(s) est disponible. C'est le propre de RStudio que de pousser cette logique intégrative : l'interface est plus conviviale, favorise les importations de données (et packages) et l'exportation de documents en ligne via Shiny. Nous reviendrons sur les risques d'une telle logique productiviste, même si elle fait les délices des agences d'évaluation. Cette partie là de la révolution de la grammatisation fait bon ménage avec ce productivisme. La logique de développement *input/process/output* dans une interface unique est redoutablement efficace. Dans sa version positive, elle rend plus facile le contrôle des méthodes et peut éventuellement servir à limiter la fraude scientifique.

3. R et la « linguistique » : où comment R organise de fait une encyclopédie des domaines scientifiques : les CRAN Task views

L'ensemble de bibliothèques disponibles sous R n'est pas entièrement anarchique. Il fait l'objet d'une organisation de la connaissance qui fonctionne en somme comme une encyclopédie des domaines scientifiques : les *CRAN Task views*. « Évidemment », un package leur est consacré, ({ctv}, Zeileis, 2005)): il s'agit d'un regroupement des principales bibliothèques relatives à un savoir. Il n'existe pas (encore?) de CRAN Task view consacré à la linguistique. Il existe un répertoire de tâches à accomplir dans le domaine du TAL (Traitement Automatique du Langage), le NLP (*Natural Language Processing*) *Task view*. L'influence du monde de l'informatique est sensible dans cette orientation de la portion du savoir consacrée à la linguistique. On peut se livrer à une cartographie des familles de packages, qui montrerait comment le savoir est, si ce n'est structuré par le logiciel, du moins pris en charge par une partie de la communauté scientifique. Plus précisément, on pourrait détailler les familles de packages disponibles pour le TAL et, à partir de cette typologie fonctionnelle, interroger les finalités, les champs linguistiques et le type d'entités manipulées. Il y a loin du répertoire de packages disponibles à l'appropriation des usages. L'une des difficultés centrales pour

---

[7] http://mapage.noos.fr/admeli/poznan/#1 (dernière consultation le 23 août 2017)

[8] HUMA-NUM vient de proposer une mise à disposition, en phase de test, de ce service.

[9] Par exemple le site http://r4ds.had.co.nz, *Data for science*, d'Hadley Wickham (dernière consultation le 23 août 2017).

les usagers de R est qu'un même algorithme (ou une même fonction) est susceptible de recevoir plusieurs implémentations dans les packages de R différents sans que le profane sache *a priori* lequel utiliser et pourquoi (et surtout sans savoir que les options par défaut ne sont pas nécessairement les mêmes pour une fonction d'un package à un autre). Reste que cette bibliothèque de bibliothèques[10] gérée par le CRAN autorise une relative transférabilité des savoirs, une certaine transversalité des méthodes et, à tout le moins, une homogénéisation des pratiques.

Je décris dans la section suivante ce que le traitement de données langagières a notamment eu comme conséquence sur R, en externe et en interne du point de vue de la linguistique.

2 R et les données : comment les données linguistiques ont changé certaines approches de R ?

Dans sa logique de développement, un certain nombre de packages se sont chargés d'importer des ressources d'analyse venues du monde de la linguistique, puis des packages ont été développés pour mieux traiter les chaînes de caractères et les textes. Une réflexion, ou au moins une pratique est en cours, prenant les textes en objet dans une approche un peu mixte de fouilles de textes (*text mining*).

1. L'intégration / l'incorporation de certaines opérations d'annotation

L'utilisation de certaines bibliothèques de programmes à substrat linguistique se fait par le biais de *wrappers* (des intégrateurs, des passerelles), qui permettent d'avoir accès à des annotations de bas niveau (Treetagger) et même de haut niveau, comme le package {coreNLP} (Arnold, Tilton, 2016) pour l'annotation des structures syntaxiques en relation de dépendance ou pour la reconnaissance d'entités nommées. On peut également citer, parmi les ressources linguistiques de l'anglais, la bibliothèque {wordnet} (Feinerer et al., 2016), qui permet d'interroger les synonymes ou de rechercher les co-hyponymes. Une partie du travail de la communauté scientifique autour de R a donc consisté à porter sous R des outils d'annotation existant par ailleurs (Arnold, Tilton, 2015). Des packages plus spécifiques ont été crées pour la gestion de données textuelles.

2. simplification en vue des traitements des chaînes de caractères : la suite {tidyverse}

Conséquence importante de l'absence relative de convivialité de R pour le traitement des chaînes de caractères, toute une suite de bibliothèques ont été créées sous l'égide de l'un des développeurs de RStudio les plus influents, Hadley Wickham. La convivialité pour l'utilisateur est particulièrement soignée : toute une série de familles de fonctions est réutilisée d'un package à un autre de la suite {tidyverse} (<http://tidyverse.org/>, (Wickham, 2017). Comme son nom l'indique, il s'agit de faciliter l'accès aux données, dans la phase la plus ingrate : la préparation et le nettoyage des données. De fait, dans l'utilisation quotidienne de R, 80% du temps passe à la mise en forme des données préalablement à leur traitement, surtout si on travaille avec des données initiales relativement sales (extractions du web). Toute cette série de packages

[10] J'ai décrit naguère (Ballier, 2010) la triangulation des domaines dans lesquels je me représente l'analyse linguistique. Je crains que ce « trièdre des savoirs » soit à reconsidérer si R entre dans la danse.

vise à fluidifier le traitement de données dans une perspective intégrationniste (Grolemund, 2014). Entre autres, elle facilite le traitement des chaînes de caractères (notamment à partir de {stringR}, Wickham, 2015), et réutilise des noms de fonctions très similaires d'un package à un autre. On voit ici très concrètement la prise en compte volontaire de problèmes spécifiques des données langagières. Les évolutions de R sont, par contrecoup, sensibles dans les manuels de référence des linguistes quantitativistes.

3. esquisse de périodisation du rapport aux données langagières

En guise de récapitulatif de plusieurs années de pratique irrégulière, d'abord rétive et souvent ambiguë, de mon utilisation de R, je propose une périodisation sommaire de ce langage de programmation devenu environnement de travail. Je l'explicite à partir de la comparaison de deux manuels d'introduction à l'analyse linguistique à partir de R, à mes yeux également remarquables bien que différents dans leur approches. Une comparaison de Baayen (2008) et de Levshina (2015) permet de mesurer le chemin parcouru, y compris en termes de conceptualisation et de pratiques[11].

On peut caractériser l'évolution comme une manière à la fois de limiter le formalisme mathématique et de le traduire en termes accessibles aux linguistes, en précisant les conditions de validité de modèles statistiques. Levshina (2015) rappelle ainsi à quelles conditions on peut utiliser l'ANOVA ou les régressions, entre autres. Baayen (2008) donne, quant à lui, un deuxième chapitre potentiellement rebutant pour le linguiste un peu profane en mathématiques, mais fondamental pour la compréhension de la loi de Poisson et autres lois de distributions. La deuxième différence notable est l'adoption d'un environnement à interface graphique (RStudio) dans Levshina (2015), quand bien même Baayen (2008) procédait déjà par une logique de scripts. La troisième différence notable joue au niveau de la visualisation. Levshina (2015) entérine le triomphe de {ggplots}, package programmé par Hadley Wickham (Wickham, 2016b) et tête de pont de la suite de packages {tidyverse} pour la visualisation. Baayen (2008) privilégie les illustrations à partir de {lattice} (Sarkar, 2008), qui offre de graphiques relativement riches pour l'époque, moins paramétrables que {ggplot}, mais particulièrement idoine pour les représentations des évolutions en parallèle des variables sur un même graphique. Baayen 2008 utilise des packages de scientifiques, tels le célèbre {e1071} (Meyer *et al*., 2014). Il est en contact étroit avec Douglas Bates, grand spécialiste de la régression et auteur lui même du package {caret} (Kuhn, 2008). Les deux ouvrages sont accompagnés d'un package qui contient des jeux de données exclusivement linguistiques, qui sont souvent des extractions (jeux de données partiels) de données linguistiques ayant été utilisées dans des publications particulièrement innovantes. Baayen est auteur ou co-auteur de publications servant à la trame de son livre. Tous ses chapitres reposent sur des analyses et des jeux de données commentés dans des articles publiés. Il n'est pas rare que Baayen soit le responsable de la partie statistique de l'analyse, comme dans le cas de l'étude de Joan Bresnan (Bresnan *et al.*, 2007) sur l'application du modèle de régression logistique à l'analyse de la syntaxe. Se fait jour une culture du jeu de données strictement linguistique. Avec une extraction légèrement différente du même jeu de données initial, le manuel de Keith Johnson (Johnson, 2011) explique également la régression à partir de l'analyse de (Bresnan *et al*. 2007). Enfin, Levshina (2015) travaille avec de packages eux-mêmes programmés par

[11] Je fais aussi allusion aux nombreux *bootcamps* ou sessions intensives de formation à R dont Stefan Gries s'est fait le champion et dont les deux éditions de son *Statistics for Linguists* (Gries, 2013) sont également la trace.

des linguistes, tels {polytomous}, rédigé par le finlandais Antti Arppe (2013) et véritable cas d'école. Son analyse de la sémantique de verbes d'opinion, objet de sa thèse pour l'analyse du finlandais (Arppe, 2008) a pu être appliquée par d'autres linguistes à l'analyse d'autres langues, tel l'anglais, dans les travaux de Caroline Krawczak (Krawczak, 2014). Bien sûr, cette périodisation est incomplète puisque l'histoire est loin d'être terminée. À peine son livre sous presse, Natalia Levshina donnait une communication sur les approches bayésiennes et regrettait publiquement de ne pas avoir consacré de chapitres à ces modèles dans son ouvrage. On voit que la pratique de R s'accompagne d'une véritable pratique bibliographique des packages; Levshina (2015) a ainsi recours au package FactomineR porté par une équipe de statisticiens de Rennes (Husson et al., 2016). Il convient ensuite d'examiner plus précisément ce que R fait aux données linguistiques.

3. Quelques conséquences sur l'analyse linguistique des données

Le recours à R permet d'envisager quelques aspects de la troisième révolution de la grammatisation. Il ne s'agit plus, contrairement à l'entreprise de la deuxième révolution de la grammatisation, de proposer une description de pied en cap de la langue (sa consignation par l'écriture, sa description par les dictionnaires et les grammaires), mais d'indiquer comment interagissent différents outils dans les processus d'analyse de la langue, à partir des versions numériques des textes. Je souhaiterais suggérer qu'il ne s'agit plus d'une analyse « simple », mais d'une véritable chaîne de traitement de données, qui se caractérise par des conséquences en amont sur la structuration des données privilégiée et, en aval, par plusieurs mutations.

1. La tabulation des données ou le triomphe des formats d'échanges et d'exportation (.csv)

La forme privilégiée dans R pour la présentation des données est le *dataframe*, c'est-à-dire un tableau susceptible d'accueillir des données quantitatives (comme l'un des autres types d'objet manipulable sous R, la matrice) mais également des données qualitatives (telles que des catégories morphosyntaxiques). Typiquement, le *dataframe* peut être chargé à partir d'un fichier au format .csv (*comma separated values*), et même d'un fichier .txt où les données auraient été tabulées. Il est assez facile de rédiger ou d'adapter des scripts sous Praat (logiciel d'analyse acoustique) respectant ce type de format pour extraire automatiquement des données. Ce format de données constitue le mode de présentation par excellence de données linguistiques pour la linguistique quantitative. Dans cette structuration des données, chaque observation est une ligne et chaque colonne une variable de l'analyse.

En d'autres termes, c'est la fin des effectifs cumulés dans la présentation des données linguistiques et, surtout, la structuration de leurs résultats. Seuls de rares modèles procèdent par cumul de fréquences, c'est par exemple le cas des modèles ndl (*naive discriminant learning*, Baayen, 2011). Il y a donc bien un impact de l'outil utilisé sur le mode de présentation de données. Sous R, selon que l'on souhaite procéder à des régressions logistiques ou à d'autres types d'analyses, on peut ainsi disposer de jeux de données longs (où chaque observation correspond à une ligne) ou de jeux de données courts (à fréquence cumulée). « Naturellement », des packages ont été créés pour

simplifier ce type de manipulations. Enfin, il est possible de transformer des données quantitatives en données qualitatives[12] (*dummy coding*).

Cette analyse de données tabulées participe d'une ré-horizontalisation des données linguistiques, alors que le concordancier avait amorcé une culture du vertical, quasi-paradigmatique (autour du pivot de la concordance se lisent le contexte droit et le contexte gauche). Le jeu de données tabulé peut également posséder une variable catégorielle (le mot), mais la compréhension de cette 'observation' se fait dans la consultation horizontale des autres colonnes. On pourrait simplifier en disant que le script le fait automatiquement. Pour prendre l'exemple de la sociolinguistique, les modèles mixtes seraient une forme de couronnement de l'analyse labovienne, le traitement sous R proposant l'analyse de l'interaction entre les colonnes relatives aux méta-données et aux données linguistiques)[13]. L'émergence du format CoNLL (Buchholz, Marsi, 2006) participe également de ce type de représentation horizontale, cette fois-ci comme sténographie d'une structure syntaxique. Le recours croissant à des classifieurs type TiMBL (Daelemans *et al*., 2004) relève de la même logique horizontale : tous les traits (*features*) associés à une variable catégorielle sont susceptibles de faire l'objet d'une sélection automatique du trait le plus pertinent. Un logiciel d'analyse syntaxique comme CESAX (Komen, 2011) propose ainsi une sortie de données de type TiMBL. Dans mon jeu de données linguistique contemporain sous R, voisinent des colonnes relatives aux marqueurs étudiés, des colonnes relatives à des propriétés quantitatives, des colonnes récapitulant les métadonnées du corpus, et bientôt des colonnes résultantes de traitement sous R.

2. L'émergence d'une documentalité hybride

Le *dataframe* est donc une structure de codage de données, qui permet de surcroît des enrichissements successifs. Rien n'est plus facile sous R que de générer une colonne opérant sur une colonne donnée. L'utilisation de packages permet de mettre en œuvre des procédures d'annotations automatiques qui génèrent des propriétés inhérentes aux données (telles que les métriques de complexité, de lisibilité avec le package {koRpus (Michalke, 2016), le nombre de syllabes par mot avec le package {qdap} (Rinker, 2013), l'annotation du haut niveau et de bas niveau avec le packages {koRpus} et {tm} (Feinerer, Hornik, 2015), ainsi qu'avec {coreNLP} (Arnold, Tilton, 2016). Avec des scripts, on peut donc à loisir enrichir tout texte de ses « paradonnées », qui sont donc ses propriétés inhérentes (nombre de mots, de phrases, de syllabes et métriques diverses caractérisant les propriétés textuelles d'un corpus donné). L'industrialisation des scripts revient à pouvoir disposer un répertoire d'annotations supplémentaires disponibles par script interposé. Disposer d'annotations automatiques comme dans un prolongement du texte activable sur demande, c'est aussi contribuer à déplacer les frontières entre transcription et annotations (voir dans ce volume la contribution de Gabriel Bergounioux). Le moment décisif de l'analyse devient celui de la transcription orthographique du texte, le reste en « découle », avec les erreurs inhérentes à

[12] C'est l'une des grandes forces de ce langage de programmation orienté objet que de pouvoir réaffecter une variable quantitative en une variable qualitative, mais c'est aussi un piège redoutable. C'est le phénomène de coercition de données. Ce processus peut se produire sans que l'utilisateur en soit averti au préalable. Par quoi on voit qu'il est facile de se tromper sous R.

[13] L'histoire ne s'arrête pas là, voir par exemples les forêts aléatoires utilisées dans Tagliamonte, Baayen, 2012.

l'automatisation (mais en partie prises en compte dans les fourchettes de valeurs de précisions décernées dans les campagnes d'évaluation). R permet donc de manipuler des jeux de données, structures tabulaires hybrides comportant potentiellement des méta-données, des extractions de données annotées, et des ré-annotations générées automatiquement.

De ce point de vue, R permet l'analyse de jeux de données hybrides, représentant à la fois des données qualitatives et quantitatives, des séquences textuelles, des scores ou des mots. Il participe d'une « documentalité hybride ». Je reprends ici le concept du philosophe italien Maurizio Ferraris, qui cherche dans ses travaux à décrire une "théorie de la documentalité" (Ferraris, 2014) et à penser l'impact du numérique sur l'architecture et les pratiques langagières. Il est possible d'entrevoir déjà ce que le traitement des corpus 'fait' au texte unique et linéaire dans son parcours lorsque tous les textes sont consultés en même temps dans un concordancier : l'analyse de documents potentiellement très nombreux se substitue à l'examen de séquences linéaires textuelles (la textualité). Le concordancier autorise la transversalité de l'ensemble des documents en parcourant le pivot de la concordance: on a accès à l'ensemble des textes comportant une occurrence du mot considéré et repéré dans le pivot de la concordance (la documentalité). Avec la « documentalité hybride », les données importées sous R (typiquement, les corpus), peuvent faire l'objet des calculs immédiats qui fonctionnent comme autant de caractérisations possibles de ses propriétés. Potentiellement, il y a quasiment une récursivité des opérations d'ajout informationnel : toute une gamme de propriétés quantitatives (seuils de significativité, information mutuelle, fréquence dans les corpus de référence, etc.) peut à son tour enrichir les annotations textuelles. Le caractère hybride de la documentalité tient entre autres à cette possibilité d'incorporer du quantitatif et du qualitatif dans une structure accueillant de la donnée. C'est un peu plus qu'un tableur, car le statut des données contenues dans chaque colonne est codé dans la structure du *data frame* ; que le nom de la variable contenue dans la colonne figure ou pas dans le jeu de données. C'est une des subtilités qui rebute l'utilisateur novice de R : on distingue pour des jeux de données sous forme de « tableaux » potentiellement deux structures de contenus : deux types d'objets distingués dans le langage de programmation, les matrices et les *data frame*. On peut importer des données codées dans un tableur, mais on assigne en quelque sorte aux colonnes du jeu de données des métadonnées : les types de variables contenues dans chaque colonne pourront être spécifiées. L'une des commandes les plus utilisées sous R est celle qui affiche les propriétés[14] de chaque colonne (`summary()`) de l'objet considéré (le jeu de données). La pluralité de type de données analysables dans ces jeux de données permet de nouvelles explorations.

3. L'exploration de nouvelles données

L'émergence des données hybrides n'est qu'une nuance de la palette des nouveaux types de données que R permet de traiter. Toute une série de signaux sont susceptibles d'être ainsi analysés, par exemple, issu de plates–formes physiologiques ou autres (EEG, *eye-tracking,* images des ultrasons, etc.). On peut également envisager des données que le concordancier ne peut pas vraiment gérer, telles que les traces numériques. À l'inverse, la structure du data frame sous R permet le rétablissement de mots visés par le scripteur dans la suite des saisies clavier, malgré ses retours en arrière, puis des textes

[14] Je ne détaille pas ici les structures de contenant, facteur, vecteur, liste, array, hash, et leurs conséquences pour l'analyse et la programmation.

correspondants (Ballier *et al.*, 2019). Enfin, les relatives facilités d'extraction des données du web par le couplage à des packages de collectes comme {curl} (Ooms, Wickham, 2016), {httpr} (Wickham, 2016a) ou {scraper} (Acton, 2010) ont permis l'émergence de données nouvelles telles que les *emojis* (cf. Hough *et al.*, 2016). R permet un accès assez large à tout un spectre de données linguistiques. Sans vouloir nécessairement parler d'un saut de paradigme, il convient de décrire certaines conséquences épistémologiques du recours à R.

4. L'écosystème de R et quelques conséquences épistémologiques

Je détaille un certain nombre de conséquences prévisibles de l'analyse linguistique conduite avec R, pas nécessairement une nouvelle *épistémé*, mais des tendances lourdes.

1. Statut des métadonnées (et de leur environnement de traitement)

Une partie du cahier des charges de ce chapitre consiste à se positionner sur le statut des 'entreprises de métadonnées' à travers l'analyse du domaine et de ses outils. Pour reprendre la problématique liminaire du volume, je serai tenté de dire que le recours à R est compatible avec des conceptions «top down». Il est essentiellement 'top down' dans sa nouvelle version RStudio, qui permet spécifiquement de caractériser le type de variable, les propriétés des données consignées dans chaque colonne lorsque l'on importe un jeu de données. Pour autant, les efforts de convivialité des interfaces et des traitements n'empêchent pas le constat d'une tension entre deux tendances structurantes contradictoires: l'entropie des packages, qui sont autant de tendances centrifuges (car il est difficile de maintenir ainsi une culture réticulaire et décentralisée en autant de communautés que d'auteurs de packages) et les tentations centralisatrices récurrentes des intégrateurs.

Malgré les évolutions de R, et la montée en puissance de l'interface graphique RStudio, il serait simpliste de croire que le traitement des métadonnées induit une solution unique, qui serait cette chaîne de traitement que nous avons décrite. R est traversé par deux mouvements contradictoires. La multiplicité de communautés scientifiques qui produisent leur propres packages ainsi que l'extension à d'autres logiciels via des passerelles participe d'une relative tendance à l'entropie. Parallèlement, régulièrement, des tentations centripètes se font jour, on cherche à doter R d'une interface centralisante, de préférence graphique, qui permette de jouer le rôle d'intégrateur. Ce fut le cas de l'interface Rcmdr (Fox *et al.*, 2010) et de ses packages supplémentaires. Le package {rattle} (Williams, 2009) rassemblait les principales techniques du *data mining*, le package {factomineR} (Husson et al., 2016) celles de l'analyse factorielle, de l'analyse des correspondances et de l'analyse en composantes principales. De nos jours, c'est RStudio qui joue ce rôle d'intégrateur de technologies en ayant simplifié l'interface de gestion de packages.

Il ne semble pas y avoir de relation nécessaire entre la richesse structurale des packages de R (caricaturalement, tout logiciel peut se voir ajouter une passerelle vers R, tout modèle statistique ou tout algorithme peut se porter sous R) et l'usage de métadonnées. En revanche, on peut noter l'émergence des données « embarquées » dans les packages. On a vu que l'utilisation de R pouvait conduire à un relatif changement du statut du jeu de données (le jeu de données iris sert à comprendre des algorithmes de classification), le développement de packages de linguistique permet également la diffusion de jeux de données et de méthodes privilégiées, si ce n'est afférentes (les

verbes de pensée, d'opinion et le package {polytomous}, Arppe, 2013). On peut même envisager le développement de packages pour valoriser des travaux doctoraux, où figureraient les principaux scripts ou fonctions ainsi que les jeux de données collectés dans la thèse. Ceci participe de l'économie de la connaissance et du "paradigme" de l'*open data* (données ouvertes).

2. Le paradigme des données ouvertes (open data)

Disposer d'un logiciel libre, et d'un système ouvert, voire très ouvert, avec ses multiples passerelles et packages, permet de s'approprier les démarches scientifiques. De par la cumulativité des scripts, on a accès à la mise à l'épreuve des différentes méthodes, nous offrant ainsi une véritable falsifiabilité des théories, ou en tout cas des analyses statistiques. L'écosystème de R permet la réplicabilité des expériences, la falsifiabilité du propos, d'autant que l'interface RStudio permet de sauver des projets entiers (englobant dans un même répertoire les données, les objets sauvegardés dans l'environnement, et le fichier de script, dans lequel on peut créer des fonctions spécifiques pour le traitement de ses données). Toute une partie de l'économie et des activités qui se construisent autour de R participe d'une philosophie générale qui prône le partage des données (ce qui signifie des gains en temps de traitement énormes pour les corpus) et favorise l'interopérabilité et la récupérabilité des données. On est à l'opposé d'une privatisation des données, d'une confiscation de résultats, ou d'une dépendance à une interface propriétaire, voire payante[15]. Dans sa version optimiste, les politiques de données ouvertes réunissent les conditions de possibilité d'une méta-analyse. Il devient beaucoup plus aisé de proposer une analyse critique des modèles statistiques utilisés si on dispose à la fois des données et des scripts correspondants aux fonctions que l'on a utilisé pour modéliser les données. À l'instar de ce qui a eu lieu en psychologie, on peut se livrer à une critique des méthodes statistiques utilisées (cf. Wagenmakers *et al.*, 2012 ou, pour l'analyse rétrospective des articles sur le domaine des corpus d'apprenants, Paquot, Plonsky, 2015). On pourrait ainsi voir poindre le cycle vertueux d'une méta-analyse : les analyses métacritiques des modèles statistiques permettent, en retour, une reconceptualisation que permettent les données sur les modèles. On peut souhaiter voir progresser la politique qui est celle de la revue en ligne *PLOS One*[16], qui incite très fortement les auteurs à déposer données et scripts en même temps que leurs articles. Cette logique de mise à disposition des données s'illustre également dans les données embarquées dans les packages, qui produisent des effets sur le rapport à la connaissance.

3. La culture du jeu de données, ou quand l'objet d'étude se fait accès à la méthode

Au prix d'un investissement initial qui peut être considérable, l'écosystème R permet de se donner accès à des domaines différents du savoir, en se raccrochant dans un premier temps aux branches que constituent les lignes de code et à la conceptualisation des jeux de données. Ils sont par exemple détaillés avec force précision en annexe du livre *Programmation et analyse statistique avec R* (Paroissin, 2015). Le jeu de données (typiquement, le jeu de données *iris*, exemple classique pour les problématiques de classification) participe d'un ensemble de pratiques qui

[15] Ne rêvons pas, il existe toute une galaxie d'entreprises qui ne demandent pas mieux que d'orienter des activités lucratives autour de R, au grand dam de l'un des pères fondateurs de R.

[16] http://journals.plos.org/plosone/ (dernière consultation le 23 août 2017)

renouvellent notre rapport aux données. C'est une sorte de praxis des contenus qui se met en place dans cette culture partagée des *datasets*. Le paradoxe est que la maîtrise du jeu de données, la connaissance du ressort des paramètres dans l'analyse qu'on peut en faire (la longueur des pétales et des sépales permet de prédire les trois classes d'iris), la connaissance des zones problématiques, ce savoir lié à la donnée, permet de mieux comprendre toute méthode d'analyse qui est proposée pour en rendre compte. C'est une sorte de métalinguistique à rebours, où les données permettent de comprendre la méthode qui les traite, l'objet traité donne un accès indirect à la méthode. Est-on pour autant condamnés à la donnée, comme un horizon indépassable de l'analyse ? En d'autres termes, R contraint-il à l'analyse quantitative ?

4. Quantitativisme obligatoire ?

Les techniques de fouilles de données (*data mining)* se prêtent particulièrement bien à l'analyse de la documentalité hybride. Mais cela ne constitue pas nécessairement l'unique horizon des données linguistiques. A titre personnel, je serais un jour tenté, comme par provocation, de proposer des relectures de concepts issus de la linguistique énonciative avec R pour suggérer qu'il n'en est rien. Se produit sans doute en ce moment une inflexion, pas nécessairement exclusivement quantitative, mais durable, vers une partie de l'appareil critique statistique : les techniques de visualisation des données, les tests de significativité, les intervalles de confiance relèvent d'une forme de pratique qu'on peut juger appréciable et qui sont sans doute susceptibles de demeurer. On voit bien sûr poindre la lame de fond du quantitatif, qu'on l'appelle *big data* ou *data science*, mais on peut aussi prévoir le nécessaire retour du balancier que l'on peut appeler de nos vœux pour le futur. Passée la phase d'expansion de statistiques, puis sa période critique, reviendra sans doute une analyse davantage qualitative, mais qui conservera ce que j'ai envie de considérer comme les acquis de la rigueur de l'appareil statistique, tels que les tests de significativité. De manière plus globale, on peut espérer une dimension méta-critique, une possibilité de retour aux données par des techniques d'analyse des *outliers*, et, plus généralement, une meilleure pensée du rapport de l'échantillon à la population, la prise en compte de ce que nos données ne sont jamais qu'un type de prélèvement possible (parmi d'autres) sur la langue. En termes statistiques, il s'agit de repenser plus finement le rapport entre l'échantillon et la population : le corpus n'est pas la langue, mais il est sans doute plus représentatif, ou du moins davantage généralisable, parce que relevant de procédures répétables, que des séries d'exemples collectées au hasard des lectures.

Le recours à R ne tranche pas non plus en faveur d 'une école plus que d'une autre en raison de la multiplicité des méthodes d'analyses qui ont été portées sous R. Je soutiendrais volontiers que, dans le cadre d'une utilisation qui se limiterait à la visualisation de données, il peut se borner à une interface d'approche qualitative des données, et n'a pas forcément vocation à un traitement quantitatif. A l'inverse, je serais prêt à soutenir que c'est l'exigence de comparabilité des ressources construites (les corpus), sans parler de l'émergence de plates-formes qui produisent de la donnée, qui a conduit à une explosion du quantitatif en linguistique[17].

[17] Ou sans doute plus exactement à un *revival*, une redécouverte après coup. De la collection linguistique de Larousse des années soixante-dix, le volume sur la linguistique et la statistique n'a pas eu l'écho mérité, pas plus que les travaux de Pierre Guiraud (Bergounioux, 2016).

Celle-ci ne saurait avoir le dernier mot, ne serait-ce qu'en raison de la possibilité de critiquer le modèle statistique utilisé. On parle parfois de métacritique de modèle. Dans l'absolu, on pourrait aussi s'offrir une analyse de type métalinguistique du code de R, montrer sa relative impopularité chez les programmeurs de profession et s'interroger sur la manière dont ce langage de programmation s'interface avec les données langagières. Dans le détail du code, on dispose de procédés métalinguistiques qui permettent d'inhiber l'interprétation de certains caractères (l'antislash avant le caractère réservé permet de le considérer comme une mention et non comme un usage, il y a donc du procédé autonymique dans le langage de programmation R). Il y a encore toute une réflexion à conduire, en linguiste, sur les propriétés « symboliques » de ce langage de programmation. Entre autres difficultés d'ordre conceptuel, se pose la question des modalités d'appropriation de ce langage de programmation.

5. Révolution dans les pratiques ?

L'apprentissage de R favorise ce que d'aucuns appelleraient une perspective « actionnelle » (*learning by doing*), on apprend mieux des méthodes statistiques (et l'exécution des scripts correspondants) en s'appropriant la méthode nécessaire afin de traiter ses propres données. Le renouvellement (et l'accélération de leur développement) des chaînes de production à base de R suppose un temps d'apprentissage considérable. Il ne s'agit pas seulement de codage, mais de compréhension de l'ensemble du traitement des données, *a fortiori* si on dépasse la simple visualisation des données et que l'on s'oriente vers la modélisation.

1. Vers une reconfiguration des prérequis ?

Révolution dans les pratiques et notamment dans la nécessité de la transparence de la démarche, le recours aux données et aux scripts rendus publics induit une nécessaire inflexion dans les pratiques des linguistes. D'autres dangers menacent le linguiste (comme les erreurs imputables à la programmation ou à la compréhension des enjeux des modèles statistiques utilisés) ; ou plutôt, le linguiste doit s'entourer de précautions supplémentaires, ce qui le transforme chaque année davantage en utilisateur de plus en plus averti de l'informatique, et exige un minimum de compréhension de la statistique. La réplicabilité des démarches reposant sur ce type d'outils a donc un coût. Pour filer la métaphore d'Henri Adamczewski, au départ employée pour renvoyer à la nécessaire ascèse de la terminologie linguistique, il y a un « ticket d'entrée » pour la linguistique, et R vient de faire flamber les prix. On peut certes compter sur une diversification et une multiplication des sources d'accès à la connaissance. Le savoir change son format de diffusion et ses supports : la conférence enregistrée, le diaporama associé et le script sont une des modalités, le manuel et son site afférent en sont une autre. Les billets publiés sur les forums et les blogs sont également autant de traces d'appropriation des bibliothèques, des fonctions ou des problématiques. A de très rares exceptions, la question que se pose l'utilisateur a déjà sa réponse en ligne. C'est la progressivité et la relinéarisation de tout ce savoir réticulaire qui fait difficulté, l'appropriation dans un texte ou un script cohérent de ces sources éparpillées.

Je ne détaillerai pas ici l'épineuse question de l'appropriation des modélisations statistiques, je voudrais juste signaler l'importance accrue de la psychologie et de la psycholinguistique comme discipline potentiellement plus avancées dans le recours aux statistiques et à leur analyse critique (méta-statistique, cf. Wagenmakers *et al.*, 2012).

Les modélisations les plus susceptibles d'être importées par la linguistique pourraient bien être les modèles de la biologie, qui peuvent avoir des problématiques similaires (survie des espèces sur un territoire donné…). Outre les manuels d'initiation à la linguistique quantitative mentionnés supra, il existe toute une série d'articles très explicites sur les modélisations utilisées. Ainsi, par exemple, le linguiste peut s'initier au modèle des forêts aléatoires (Tagliamonte, Baayen, 2012) ou du *naive discriminant learning* (Baayen *et al.*, 2011) à partir de l'analyse linguistique contenue dans un article qui donne en annexe un script (plus ou moins commenté) correspondant à l'application du modèle statistique en question. La diversité des modèles statistiques disponibles rajoute à la complexité de l'appropriation de R. Sa polyvalence en fait pourtant un atout.

2. Diversité des usages de ce langage de programmation

R est une architecture ouverte qui ne présuppose pas la clôture de ses usages au nom de sa finalité, là où un concordancier ou un visualisateur de la parole type Praat sont téléonomiquement davantage fermés (indépendamment de leur éventuelle adjonction de plugins). En quoi cet outil (pas un méta-outil, mais un outil-pivot) se laisse-t-il appréhender de manière différente ? Il faudrait sans doute plus travailler en transversalité entre les différentes disciplines pour lesquelles R est susceptible d'être utilisé, (du moins dans son enseignement : R pour les biologistes contient des packages de génétique susceptible d'intéresser les linguistes qui ont également des problématiques de séquences à décoder et à ordonnancer). Dans les différents domaines où on peut l'utiliser, R est potentiellement un intégrateur. Dans certaines des pratiques radicales, ce langage de programmation orienté objet est mis à contribution pour bien des fonctions, y compris celles autrefois assurées par d'autres logiciels. Ainsi Stefan Gries se sert-il de R pour établir des concordances et des sorties tabulées de ses données qui transforment R en concordancier. Dans les cas (avérés) de fétichisme extrême, R peut aussi servir à envoyer son courrier (ou à faire envoyer par des comptes twitter des messages de relance à ses doctorants[18]), il n'y a guère que le café qui ne se fasse pas sous R. Bref, avec R, on peut tout faire (sauf le café).

3 Déplacement de la fracture numérique et culture numérique

Reste qu'avant de jubiler avec R, il y a du travail à fournir. On peut estimer que l'utilisation de R, même médiée par une logique de scripts, va conduire à un déplacement de la fracture numérique. Pour l'humaniste contemporain, la question ne sera pas d'utiliser ou non des ordinateurs, mais plutôt d'accepter d'employer des lignes de commande, et, dans une approche plus conviviale centrée sur une interface graphique (RStudio), d'utiliser et de réadapter des scripts. La fracture numérique va donc se déplacer entre ceux qui franchiront le Rubicon de la ligne de code (et de l'absence d'interface graphique), dans le cadre d'une pédagogie du package, et plus généralement, du script et de l'algorithme. Il y aurait à redire sur un certain machisme inhérent à une certaine culture informatique, et à son dédain affiché (*real men don't click*) pour les interfaces graphiques, affublées du terme de « clique-bouton », reste que la question est celle de la ligne de commande, véritable fracture numérique qui oppose tout une partie des "littéraires" aux autres chercheurs. Accepter l'écosystème de R suppose toute une pédagogie du script, et plus encore, des modèles mathématiques sous-jacents. De ce point de vue, au-delà des exhortations au gigantisme et de

[18] http://www.r-datacollection.com/blog/Programming-a-Twitter-bot/ (dernière consultation le 23 août 2017)

l'improvisation permanente, il y a encore beaucoup à faire en termes d'aménagement des cursus universitaires pour les humanités numériques.

La logique du script s'apparente à l'apprentissage des langues étrangères[19] (c'est un langage de programmation), mais elle suppose aussi une explicitation des concepts informatiques et statistiques sous-jacents à certaines fonctions. La culture numérique passe aussi par une connaissance des algorithmes et de ce qu'ils font (par exemple, les algorithmes « gloutons » vs. les algorithmes «paresseux»). Cela ne va pas de soi de faire du code et l'algorithme mérite aussi qu'on le pense, Barbara Cassin a ainsi montré la nécessité de penser en philosophe des algorithmes, tels que le trop célèbre PageRank de Google (Cassin, 2013).

La pédagogie du package est à la fois l'approfondissement du domaine scientifique considéré et des fonctions et jeux de données qui le composent. La documentation est organisée par ordre alphabétique et parfois complétée par des sites dédiés. Parmi les stratégies envisageables et envisagées pour s'initier aux packages et aux lignes de codes sous R, le portage d'une partie des packages dans une interface en ligne a été tenté plusieurs fois avec un certain succès. *A minima*, l'utilisateur a recours aux potentialités de l'écosystème : l'interface permet de saisir ses données ou de les copier et les résultats s'affichent. C'est par exemple le cas pour la normalisation des données phonétiques et le site NORM (Thomas & Kendall 2017). Dans un second temps, l'utilisateur se rendra compte des limites que l'interface lui impose et lui apparaîtra alors l'intérêt de la programmation pour obtenir un résultat spécifique. Les serveurs shiny permettent un système mixte qui permet d'afficher le code correspondant à tout changement dynamique en ligne (moyennant un abonnement pour le diffuseur). Le site http://langtest.jp/ permet à l'utilisateur de voir les schémas de visualisation, quelques statistiques élémentaires et le code correspondant (Mizumoto & Plonsky, 2014).

Je terminerai en décrivant ce qui est sans doute mon jeu de langage (Ballier, 2010) du moment: la pédagogie du package. S'immerger dans la logique du package permet de s'initier à la logique d'un domaine de recherche ou d'un type de modélisation. On retrouve ici l'épineuse question du caractère essentiel de la documentation du package (et de certaines de ses limites pour certaines des réalisations sous R). Interface possible tant avec les modélisations statistiques qu'avec des traitements externalisées de données, R peut donner forme (et accès aux contenus) à l'humanisme de la Renaissance des trop rebattues humanités numériques. Ce n'est pas un logiciel, c'est un univers.

Conclusion : des mots fétiches aux actes ?

Je voudrais en guise de conclusion revisiter quelques injonctions contemporaines qui font florès dans le paysage des appels d'offre et suggérer que, pour une fois, R est susceptible de leur (re)donner un contenu. Il ne s'agit pas de suggérer que R soit la panacée ou l'horizon indépassable de la linguistique, mais que c'est une voie qui gagne à être réfléchie et expérimentée.

Big data

[19] Brown (1991) parle des statistiques comme d'un langue étrangère (statistique langue étrangère, sur le modèle de français langue étrangère) ; Mizumoto, Plosnky 2014 voient dans R une possible *lingua franca*, un langage vernaculaire *a minima*. Il n'en demeure pas moins qu'il faut apprendre cette *lingua franca*.

R a une limite importante : il stocke l'intégralité des données en mémoire vive, ce qui limite le volume des données qu'on est susceptible de lui faire traiter. « Naturellement », on a inventé des packages pour détourner, puis contourner cette difficulté. Outre les commandes classiques pour mesurer le temps d'exécution d'une fonction, on a inventé des packages pour comparer les performances. On a ensuite cherché à compiler en C toute une partie du traitement le plus lourd. Est par la suite venue une architecture distribuée et parallèle. L'étape actuelle autorise des connexions à des clusters (serveurs en ligne) via une interface avec Spark (package {sparkr}, puis {Sparkly} (Luraschi, 2017)) et par des serveurs (RStudio server pro). En clair, même si les données supérieures à 16 gigaoctets sont difficilement traitables en local, R n'est pas condamné à l'entrée de gamme du Big Data.

Travail collaboratif

Accepter le temps d'apprivoiser ce langage de programmation permet de s'insérer dans une communauté d'autres utilisateurs et d'autres pratiques. Il peut paraître douteux de vouloir faire de R le levier d'un contre-modèle d'une politique de la recherche au moment où une partie de l'analyse financière cherche à s'accaparer l'outil R (*Revolution analytics* a été racheté par Microsoft) et à monétiser la science des données au service des prédictions boursières ou autres (voir les récriminations de l'un des pères du langage sur son blog), reste que cette communauté d'utilisateurs de R déploie une activité scientifique qui donne des raisons d'espérer. Plus que jamais, 90% de ce qu'il y a à savoir est déjà en ligne : en cas de difficulté dans la programmation, il y a des fortes chances que votre question ait déjà été posée, et résolue fort rapidement et la plupart du temps fort civilement, dans des forums collaboratifs, mais la plupart du temps en anglais.

Interdisciplinarité et intradisciplinarité

Chacun pourra grimacer à sa guise des errements de l'autre hors de son champ, mais les littéraires doivent bien apprendre du code, essayer de comprendre un minimum ce que font les tests et les modèles statistiques et les autres scientifiques peuvent travailler sur des données textuelles. Chacun pourra se sentir trahi de voir ce que font des étrangers au domaine avec leur objet d'étude, mais on pourra aussi se réjouir de ces possibilités de dialogue. Il ne s'agit pas de sombrer dans l'angélisme, mais de constater qu'un linguiste comme Stefan Gries a certes une approche résolument quantitative, inspirée des modélisations statistiques, mais qui trouve à s'exercer sur une série d'objets linguistiques assez variée, et devenue finalement rare dans la linguistique actuelle. Toute une partie de son activité de chercheur consiste à formuler en termes compréhensibles par la communauté de ses collègues statisticiens les problématiques des étudiants qu'il dirige, et, réciproquement, à traduire les réponses mathématiques en enjeux linguistiques. Le résultat est que ses travaux déploient une intradisciplinarité au sein de la linguistique (problématiques de grapho-phonématique, de morphologie, de sémantique et de syntaxe) qui force l'admiration. C'est aussi une sorte de figure contemporaine qui constitue une forme de pratique d'analyse linguistique qui n'est pas sans évoquer l'esprit des Encyclopédistes, à une période où la linguistique est devenue trop complexe et trop technique, en un mot, trop spécialisée, pour, en certaines occasions, être autre chose qu'une balkanisation de communautés hyper-spécialisées entre linguistique de corpus de l'écrit et linguistique de corpus de l'oral.

Ce chapitre doit beaucoup au parcours d'Harald ('R') Baayen, comme une recherche constante d'une applicabilité des modélisations statistiques aux données linguistiques, et ce, depuis ses tout premiers travaux de psycholinguiste sur la productivité du lexique, aux travaux fondateurs de Keith Johnson, qui a su admirablement croiser dans son manuel (Johnson, 2008) domaines linguistiques et modèles statistiques, aux articles et manuels de Stefan Gries et à nos discussions lors de son séjour comme professeur invité de Paris Diderot en 2014, à la richesse des ressources en ligne sur R, notamment développées par Stefan Evert et Marco Baroni, aux discussions avec mes collègues Timothée Giraud et Claude Legras (RIATE), Jean-Baptiste Yunès (LIAFA), Aurélie Fischer et Clément Levrard (LPMA), avec mes collègues linguistes Pascal Amsili, Marie Candito, Benoit Crabbé, Laura Goudet et Vincent Renner et avec mes étudiants, et, je l'espère, futurs collègues, Maelle Amand, Thomas Gaillat, Paula Lissón, Adrien Méli, Erin Pacquetet et Véronique Pouillon.

BIBLIOGRAPHIE:

ACTON R.M. (éd), 2010, *ScrapeR: Tools for Scraping Data from HTML and XML Documents*. < https://cran.r-project.org/web/packages/scrapeR/scrapeR.pdf>. Consulté le 30/08/2017.

ARNOLD T. et TILTON L., 2015, *Humanities Data in R*, Heidelberg, Springer.

ARNOLD T. et TILTON L. (éd), 2016, *Package 'coreNLP'*. < https://cran.r-project.org/web/packages/coreNLP/coreNLP.pdf> Consulté le 30/08/2017.

ARPPE A., 2008, *Univariate, bivariate, and multivariate methods in corpus-based lexicography: a study of synonymy*, Publications of the Department of General Linguistics, University of Helsinki.

ARPPE A., 2013, *Package 'polytomous': Polytomous Logistic Regression for Fixed and Mixed Effects (R package version 0.1. 4)* < http://cran.r-project.org/web/packages/polytomous/polytomous.pdf> Consulté le 30/08/2017.

AUROUX S., 1991, *La révolution technologique de la grammatisation. Introduction à l'histoire des sciences du langage*, Liège, Mardaga.

BAAYEN R.H., 2008, *Analyzing linguistic data: A practical introduction to statistics using R*, Cambridge University Press.

BAAYEN R.H., 2011, « Corpus linguistics and naive discriminative learning », *Revista Brasileira de Linguística Aplicada*, 11, 2, p. 295‑328.

BAAYEN R.H. *et al.*, 2011, « An amorphous model for morphological processing in visual comprehension based on naive discriminative learning. », *Psychological review*, 118, 3, p. 438.

BALLIER, N. PACQUETET, E. & ARNOLD, T. ,2019, Investigating Keylogs as Time-Stamped Graphemics, in *Proceedings of Graphemics in the 21st Century*, Brest 2018 (Yannis Haralambous, Ed.), Brest: Fluxus Editions, 353–365; DOI: https://doi.org/10.36824/2018-graf-ball

BALLIER N., 2004, *Praxis métalinguistiques et ontologie des catégories*, Document de synthèse d'HDR, Université Paris X-Nanterre. <https://hal-univ-diderot.archives-ouvertes.fr/tel-01277523> Consulté le 30/08/2017.

BALLIER N., 2010, « "Je sais bien mais quand même" ou quand les linguistes se défient du langage. », dans CHIANTARETTO J.-F. *et al.* (éd.), *Langage et confiance*, Paris, In Press, p. 29‑44.

BRESNAN J. *et al.*, 2007, « Predicting the dative alternation », *Cognitive foundations of interpretation*, ed. by BOUME G., KRAEMER I., ZWARTS J. Amsterdam: Royal Netherlands Academy of Science, p . 69‑94.

BUCHHOLZ S. et MARSI E., 2006, « CoNLL-X shared task on multilingual dependency parsing », *Proceedings of the Tenth Conference on Computational Natural Language Learning,* Association for Computational Linguistics, p. 149‑164.

CASSIN B., 2013, *Google-moi*, Paris, Albin Michel.

CHANG W. *et al.*, 2015, « Shiny: web application framework for R », *R package version* 0.11,1. < https://cran.r-project.org/web/packages/shiny/index.html> Consulté le 30/08/2017.

COOK J., 2013, *The R Language: The Good The Bad & The Ugly*, <https://www.youtube.com/watch?v=6S9r_YbqHy8&t=1006s> Consulté le 30/08/2017.

CSARDI G. et NEPUSZ T., 2006, « The igraph software package for complex network research », *InterJournal, Complex Systems*, 1695, 5, p. 1‑9.

DAELEMANS W. *et al.*, 2004, « Timbl: Tilburg memory-based learner », <https://ilk.uvt.nl/downloads/pub/papers/Timbl_6.3_Manual.pdf> Consulté le 30/08/2017.

FEINERER I. *et al.*, 2016, « Package 'wordnet' ». <https://cran.r-project.org/web/packages/wordnet/index.html> Consulté le 30/08/2017.

FEINERER I. et HORNIK K. (éd), 2015, *Tm: text mining package*. < URL http://CRAN. R-project. org/package= tm> Consulté le 30/08/2017.

FERRARIS M., 2014, *Documentalità: perché è necessario lasciar tracce*, Bari ,Gius. Laterza & Figli Spa.

FOX J. *et al.*, 2010, « Rcmdr: R commander, R package version 1.6-3 », *CRAN.Rproject. org/package= Rcmdr.(Mars 2012)*.

GENTRY J. (éd), 2016, *Package 'twitteR'*. <https://cran.rproject.org/web/packages/twitteR/twitteR.pdf> Consulté le 30/08/2017.

GRIES S.T., 2013, *Statistics for linguistics with R: A practical introduction*, Berlin, Walter de Gruyter.

GROLEMUND G., 2014, *Hands-On Programming with R: Write Your Own Functions and Simulations*, Sebastopol (CA) , O'Reilly Media, Inc.

HOUGH J. *et al.*, 2016, « Duel: A multi-lingual multimodal dialogue corpus for disfluency, exclamations and laughter », disponible sur HAL [https://hal.archives-ouvertes.fr/hal-01371394].

HUSSON F. *et al.*, 2016, « Package 'FactoMineR' ». < https://cran.r-project.org/web/packages/FactoMineR/index.html> Consulté le 30/08/2017.

JOHNSON K., 2011, *Quantitative methods in linguistics*, John Wiley & Sons.

KOMEN E.R., 2011, « Cesax: coreference editor for syntactically annotated XML corpora », *Reference manual*. Nijmegen, Netherlands: Radboud University Nijmegen.

KRAWCZAK K., 2014, « Shame and its near-synonyms in English: A multivariate corpus-driven approach to social emotions », dans *Emotions in Discourse*, I. Novakova, P. Blumenthal & D. Siepmann (Eds.)., Frankfurt, Peter Lang, p. 84–94.

KUHN M., 2008, « Caret package », *Journal of Statistical Software*, 28, 5, p. 1‑26.

LEVSHINA N., 2015, *How to do Linguistics with R: Data exploration and statistical analysis*, Amsterdam, John Benjamins Publishing Company.

LURASCHI J. (éd), 2017, *Sparklyr: R Interface to Apache Spark*.

<https://cran.rstudio.com/web/packages/sparklyr/index.html> Consulté le 30/08/2017.

MEYER D. *et al.*, 2014, « E1071: Misc Functions of the Department of Statistics (e1071), TU Wien. R package version 1.6-3 », <*https://cran.r-project.org/web/packages/e1071/index.html*> Consulté le 30/08/2017.

MICHALKE M. (éd), 2016, *KoRpus: An R Package for Text Analysis*.

< http://reaktanz.de/?c=hacking&s=koRpus> Consulté le 30/08/2017.

OOMS J. et WICKHAM H. (éd), 2016, *Package « curl »*. <https://cran.r-project.org/web/packages/curl/index.html> Consulté le 30/08/2017.

PAQUOT M. et PLONSKY L., 2015, « Quantitative research methods and study quality in learner corpus research », dans *Presentation at the Third Learner Corpus Research Conference* (Nijmegen, Netherlands). http://hdl. handle. net/2078.1/165211 Consulté le 30/08/2017.

PAROISSIN C., 2015, *Programmation et analyse statistique avec R*, Paris, Ellipses Marketing.

RINKER T. (éd), 2013, *Qdap: Quantitative Discourse Analysis Package*, University at Buffalo/SUNY, Buffalo, New York.

SARKAR D., 2008, *Lattice: multivariate data visualization with R*, Heidelberg, Springer Science & Business Media.

TAGLIAMONTE S.A. et BAAYEN R.H., 2012, « Models, forests, and trees of York English: Was/were variation as a case study for statistical practice », *Language variation and change*, 24, 2, p. 135‑178.

WAGENMAKERS E.-J. *et al.*, 2012, « An agenda for purely confirmatory research », *Perspectives on Psychological Science*, 7, 6, p. 632‑638.

WICKHAM H., 2015, « Stringr: Simple, consistent wrappers for common string operations », *R package version*, 1, 0, .

WICKHAM H. (éd), 2016a, *Package « httr »*. < https://cran.r-project.org/web/packages/httr/index.html> Consulté le 30/08/2017.

WICKHAM H. (éd), 2016b, *Ggplot: An implementation of the Grammar of Graphics in R*.

WICKHAM H. (éd), 2017, *Tidyverse: Easily Install and Load « Tidyverse » Packages*. <https://cran.r-project.org/web/packages/tidyverse/index.html> Consulté le 30/08/2017.

WILLIAMS G.J., 2009, « Rattle: a data mining GUI for R », *The R Journal*, 1, 2, p. 45‑55.

ZEILEIS A., 2005, « CRAN task views », *R News*, 5, 1, p. 39‑40.