

Outils de recherche en sciences sociales numériques

Table of contents

Avant-propos

Avant propos de Yannick Dufresne.

Données massives, causalité et sciences sociales : Changements et réflexions sur l'avenir

L'apparition des données massives (*big data*) dans le paysage technologique représente un cas de phénomène hautement technique dont les effets politiques et sociaux sont remarquables. Depuis quelques années, la discussion publique s'est en effet rapidement emparée du sujet, au point de transformer un développement technologique en phénomène social. Les données massives se trouvent ainsi régulièrement présentées dans l'espace public à la fois comme un moyen puissant de développement et d'innovation technoscientifique, de même que comme une menace à la stabilité de certaines normes sociales telles que la confidentialité des informations privées. Il n'est d'ailleurs pas rare que le discours public s'inquiète du danger que poseraient les données massives à la séparation des sphères publique et privée, pourtant centrale à la conception libérale du rôle de la politique qui structure la majorité des débats sociaux, en amalgamant parfois de manière trop rapide l'objet et l'utilisation qui en est faite. Toutefois, ce même discours public s'emporte aussi rapidement à propos des gains technologiques monumentaux réalisés par l'utilisation des données massives.


Dans le domaine des sciences sociales, les avancées dues à l'utilisation des données massives se font de plus en plus fréquentes et l'impact des données massives dans le domaine de la recherche sociale est en ce sens indéniable. Toutefois, d'un point de vue épistémologique, l'utilisation des données massives en recherche en sciences sociales dans les dernières années laisse plusieurs questions ouvertes dans son sillage.

Comment l'utilisation des données massives change-t-elle la pratique des sciences sociales? Les données massives causeront-elles un changement de paradigme scientifique?

Ce chapitre ne prétend pas offrir de réponses définitives à ces questions, mais plutôt des pistes de réflexion par le biais d'une introduction critique de certains points relatifs aux impacts des données massives sur la recherche en sciences sociales. Premièrement, nous présentons une conceptualisation des données massives. Deuxièmement, nous nous penchons sur les impacts des données massives en sciences sociales et soulignons tout particulièrement comment elles affectent les enjeux de la *validité* interne et externe dans le domaine des sciences sociales. Cela nous offre aussi l'opportunité d'aborder le sujet important de la différence entre les données expérimentales et observationnelles. Finalement, nous proposons quelques pistes de réflexion sur l'avenir des données massives en sciences sociales en identifiant certains changements *épistémologiques* que ces données pourraient potentiellement entraîner.

Définition des données massives

Il existe au moins trois approches conceptuelles permettant de définir les « données massives » (voir Figure 1.1.).



1. Définition de base	Quantité importante de données dont la nature, le type, la source, etc. varient
2. Définition technique/technologique	Ensemble de <i>pratiques</i> de collecte, de traitement et d'analyse de ces données
3. Définition sociologique	Innovation technique et technologique, de même que les effets sociaux qui l'accompagne

1. Premièrement, les données massives représentent une *quantité importante de points d'information* qui varient selon la nature, le type, la source, etc. Ici, la distinction entre données massives et données plus traditionnelles (ou « non-massives ») est simplement quantitative.
2. Deuxièmement, les données massives constituent un *ensemble de pratiques* de collecte, de traitement et d'analyse de ces points d'information. Les données massives représentent une technique, c'est-à-dire une manière ou une méthode nouvelle de faire de la recherche.
3. Finalement, d'une perspective sociologique, les données massives représentent les impacts sociaux de ces importants développements technologiques. Cette perspective souligne le caractère essentiellement social des données massives, en portant notamment attention aux risques liés à la confidentialité des données, aux enjeux relatifs au consentement et à l'autorisation de collecte des informations, aux innovations en intelligence artificielle, etc.

Dans les domaines scientifiques et technologiques, la définition courante attribuée aux données massives intègre des éléments de ces trois niveaux d'analyse en se référant à la composition et à la fonction des données. Premièrement, la *composition* des données massives est généralement conceptualisée comme comprenant « 4V » : le volume, la variété, la vélocité et la véracité. Cette conceptualisation jouit d'un large consensus scientifique (Chen, Mao et Liu, 2014; Gandomi et Haider, 2015; Kitchen et McArdle 2016). Par ailleurs, plusieurs chercheurs ont élargi cette définition de la composition des données massives en y incluant, par exemple, la variabilité et la valeur des points de données (Kitchen et McArdle 2016). Deuxièmement, la *fonction* des données massives comprend les innovations relatives à l'optimisation, à la prise de décision et à

l’approfondissement des connaissances qui résultent de leur utilisation. Ces fonctions touchent des domaines sociaux disparates, incluant le souci d’efficacité et de rendement des secteurs privé et public ainsi que la recherche scientifique pure (Gartner 2012).

Les données massives et les sciences sociales

Dans le domaine des sciences sociales, les changements causés par l’utilisation des données massives en recherche sont significatifs. Plusieurs n’hésitent d’ailleurs pas à les qualifier de changements de paradigme dans l’étude des phénomènes sociaux (Anderson 2008; Chandler 2015; Grimmer 2015; Kitchin 2014; Monroe et al. 2015). Dans le cas qui nous intéresse, deux dimensions majeures méritent d’être abordées : (1) une première relative à la validité (interne et externe) des données massives et (2) une seconde relative à la différence entre les données expérimentales et les données observationnelles. Ces deux dimensions sont présentées de manière simultanées dans les prochaines sections.

La validité de la mesure en sciences sociales

La validité de la mesure constitue une exigence méthodologique centrale à la recherche en sciences sociales. Les scientifiques cherchent effectivement à s’assurer que ce qui est mesuré — par un sondage, une entrevue, un thermostat ou tout autre outil de mesure — constitue bel et bien ce qui est censé être mesuré. Adcock et Collier définissent plus spécifiquement l’application de la validité de la mesure en sciences sociales en affirmant que des scores (y compris les résultats de classification qualitative) doivent capturer de manière significative les idées contenues dans le concept correspondant (2001: 530).

Toutefois, les problèmes liés à la validité de la mesure sont nombreux et ont une importance considérable. Dans l’étude des phénomènes sociaux et humains, la validité de la mesure prend d’ailleurs une complexité supplémentaire du fait que les données collectées par le biais d’une mesure constituent le *produit de l’observation* d’un phénomène, mais non pas le phénomène en soi. Ainsi, lorsque, dans le contexte d’une recherche, on propose de mesurer l’humeur de l’opinion publique (le phénomène en soi) sur un enjeu politique, on utilise généralement un sondage qui a pour fonction de mesurer le pouls d’un échantillon de la population d’intérêt (ce qui est réellement observé). Cependant, ce que ce sondage mesure ne constitue pas tout à fait l’opinion publique elle-même, mais plutôt un segment populationnel qui se veut le plus souvent représentatif de l’humeur de l’opinion publique. Ceci est tout aussi vrai pour les sondages à petits échantillons que pour ceux utilisant des données massives. Autrement dit, la mesure et les données collectées ne représentent pas le phénomène — l’opinion publique — en soi.

On a déjà mentionné que la validité de la mesure a de l’importance puisqu’elle garantit que ce qui est mesuré représente réellement ce qu’on croit mesurer. Toutefois, pour être plus spécifique, dans une approche positiviste, la validité de la mesure se traduit généralement par une logique de classification des valeurs attribuées aux différentes manifestations distinctes

d'un même phénomène. Par exemple, une mesure de la démocratie comme celle proposée par *Freedom House*, fréquemment utilisée en science politique, classe les libertés civiles et les droits politiques des États du monde par degré afin de construire un index, ou une échelle, allant d'un autoritarisme complet à une démocratie parfaite. Les scores représentent, dans ce contexte, une mesure artificielle, mais ordonnée et logique, des idées contenues dans le concept de démocratie telles que libertés civiles et droits politiques. On peut ainsi dire que la question de la validité de la mesure est un élément central de ce qui unit (1) le phénomène social étudié (la démocratie), (2) son opérationnalisation (via les libertés civiles et droits politiques) et (3) la méthode de mesure utilisée pour observer et classer d'une certaine façon le phénomène et les données qui en découlent (dans le cas de *Freedom House*, des codeurs travaillant de manière indépendante les uns des autres).

La validité des données massives

En ce qui a trait aux données massives, la question de la validité de la mesure constitue un défi nouveau. Les données massives ont en effet comme avantage d'offrir aux chercheurs soit de nouveaux phénomènes à étudier, soit de nouvelles manifestations et nouvelles formes à des phénomènes déjà étudiés. Les données massives permettent donc d'agrandir la connaissance scientifique.

L'étude de King et al. (2013) représente un cas éclairant de phénomène social que l'utilisation des données massives permet désormais d'étudier. En se basant sur la collecte de plus de 11 millions de publications en ligne, King et ses collègues ont pu mesurer la censure exercée par le gouvernement chinois sur ces réseaux sociaux. En utilisant des données massives nouvelles, les auteurs ont donc pu observer une manifestation inédite de censure massive qui, sans de telles données, serait probablement demeurée mal comprise d'une perspective scientifique. Le nombre de recherches basées sur l'utilisation des données massives similairement innovantes en sciences sociales est par ailleurs en croissance constante (Beauchamp 2017; Bond et al. 2012; Poirier et al. 2020; Bibeau et al. 2021).

Cependant, il faut aussi souligner que les données massives, en raison de leur complexité, peuvent avoir pour désavantage d'embrouiller l'étude des phénomènes sociaux. Les opportunités scientifiques liées aux données massives s'accompagnent en effet de certaines difficultés méthodologiques. Parmi ces difficultés, trois enjeux sont particulièrement cruciaux : (1) la validité interne, (2) la validité externe et (3) la question d'un changement de posture ou d'orientation épistémologique en sciences sociales causé par les données massives.

Validité interne des données massives

Premièrement, les données massives peuvent représenter un défi à la validité interne des études en sciences sociales en rendant pragmatiquement difficile l'établissement de *mécanismes causaux clairs*. Ce défi est notamment une conséquence du fait que la plupart des données

sont présentement issues d'un processus de génération (*data-generating process*) qui est hors du contrôle des chercheur.e.s. Les données massives proviennent en effet habituellement de sources diverses qui sont externes aux projets de recherche qui les utilisent. Elles ne sont pas donc générées de manière aléatoire sous le contrôle des chercheur.e.s.

Un des problèmes liés à cette situation est qu'il est difficile de garantir une source *exogène* de variation par laquelle les chercheur.e.s éliminent l'effet potentiel des facteurs confondants (*confounders*). Règle générale, la distribution aléatoire d'un traitement et d'un contrôle dans une expérience en laboratoire ou sur le terrain représente le standard le plus élevé permettant de fournir cette source exogène de variation, notamment parce qu'elle l'attribution aléatoire du traitement ou du contrôle est entièrement sous le contrôle du chercheur.e.s menant l'expérience. Cependant, en ce qui à trait à la plupart des données massives, elles sont générées de manière indépendante du contrôle du chercheur.e.s, et sont donc soumises aux mêmes enjeux et problèmes (biais) que les données observationnelles traditionnelles.

Pour le dire autrement, le défi de validité interne avec les données massives constitue un enjeu relatif à la qualité des données. Ce n'est évidemment pas un défi propre ou unique aux données massives. Ce défi s'applique également aux autres types de données. Cependant, dans l'état actuel des choses, le volume et la variété — deux des 4V — des données massives — textuelles, numériques, vidéos, etc. — peuvent miner la qualité de l'inférence causale entre une cause et une conséquence que permet habituellement un processus contrôlé de génération des données. En somme, la validité interne des données massives est une fonction de la qualité de ces mêmes données.

Validité externe des données massives

Deuxièmement, les données massives représentent aussi un défi important pour la validité externe des recherches en sciences sociales (Tufekci 2014; Lazer et Radford 2017; Nagler et Tucker 2015). Un des problèmes les plus évidents concerne la *représentativité* des données massives collectées.

Comme le soulignent Lazer et Radford (2017), la quantité de données, en soi, ne permet pas de corriger pour la non-représentativité des données. Les données massives sont ainsi soumises au même problème de biais de sélection que les autres types de données observationnelles, tels un sondage ou une série d'entrevues, traditionnellement utilisés en sciences sociales.

Le cas célèbre de l'erreur de prédiction du *Literary Digest* lors de la campagne présidentielle américaine de 1936 illustre bien ce problème. Lors de cette campagne, le *Literary Digest* a prédit à tort la victoire du candidat républicain Alf Landon sur le président démocrate sortant Franklin D. Roosevelt, puisque son échantillon de répondants surreprésentait les électeurs plus aisés, traditionnellement plus républicains, au détriment des électeurs moins aisés, plus généralement proches du Parti démocrate. Cette erreur de surreprésentation dans l'échantillon est due au fait que le *Literary Digest* a effectué un échantillonnage basé sur les listes téléphoniques et le registre des propriétaires de voitures, biaisant par le fait même l'échantillon au détriment

des électeurs plus pauvres ne possédant pas de téléphone ou d'automobile, mais qui constituaient un électorat favorable à Roosevelt (Squire 1981). Le biais de sélection du sondage a ainsi sous-estimé le soutien populaire de Roosevelt de plus de 20 points de pourcentage.

Aujourd'hui, l'utilisation des données massives est soumise aux mêmes enjeux méthodologiques. L'accumulation massive de données ne permet pas de compenser pour la qualité des données. Les données massives, comme les données plus traditionnelles, sont soumises aux conséquences induites par le processus de génération des données (*data generating process*) comme un échantillonnage.

Toutefois, depuis quelques années, le développement de nouvelles méthodes de pondération des données offre des pistes de solutions. La grande quantité de données massives permet notamment d'appliquer des méthodes de pondération bien plus efficaces pour corriger les échantillons non-représentatifs (Wang et al. 2015).

Données expérimentales

La question du processus de génération des données devient plus claire quand on considère comment les *données observationnelles* et les *données expérimentales* permettent d'effectuer des inférences de manière distincte (voir Figure 2). Toutefois, pour bien comprendre ce point, il faut comprendre les notions de données expérimentales et d'inférence causale, qui sont centrales au domaine de la causalité en recherche.

En quelques mots, l'essence de la démarche causale se résume comme suit : le processus de génération de données expérimentales a pour objectif d'assurer la validité d'une inférence causale estimée sur un échantillon sur l'ensemble de la population visée.

Plus spécifiquement, le processus de génération des données permet aux chercheur.e.s de s'assurer que la distribution du traitement entre les deux groupes, traitement et contrôle, est entièrement aléatoire. De manière technique, cette distribution aléatoire du traitement entre les deux groupes permet de garantir une source exogène (à l'opposé de endogène) de variation sur la variable indépendante (x). *Cette source exogène de variation permet, quant à elle, d'éliminer l'endogénéité entre la variable indépendante (x) et le résidu (e^*).*

Autrement dit, le fait de distribuer au hasard le traitement entre les membres du groupe traitement et ceux du groupe contrôle assure que la variation dans les résultats ne vient pas d'autres facteurs non-contrôlés (le résidu, e), mais plutôt du traitement lui-même (la variable indépendante, x). En distribuant le traitement de manière aléatoire, on s'assure que les différences dans les résultats sont vraiment dues au traitement et non à d'autres facteurs.

Il s'agit là d'assurer le respect de la condition d'indépendance, essentielle à la validité de l'identification de l'effet causal étudié. Autrement dit, en éliminant l'endogénéité entre la x et e , on s'assure que l'effet observé n'est pas dû à une variable confondante.

Pour revenir aux données massives, celles-ci ne peuvent pas résoudre les enjeux liés aux inférences causales ou explicatives (Grimmer, 2015). Elles sont en effet également soumises aux mêmes impératifs issus du processus de génération des données.

Données observationnelles

En ce qui a trait aux données observationnelles, il y a deux points importants. Premièrement, des méthodes d'inférence basées sur des approches par design (*design-based methods*) comme une méthode de régression sur discontinuité ou de variable instrumentale peuvent également garantir des inférences explicatives et causales valides. Elles nécessitent toutefois plusieurs postulats plus restrictifs dont l'objectif est d'imiter ou de recréer, de la manière la plus fidèle possible, une distribution aléatoire du traitement – ce que la littérature appelle un *as-if random assignment* (comme si l'attribution était aléatoire) (Dunning, 2008).

Dans un contexte observationnel, les données massives peuvent donc permettre d'augmenter la précision des estimations causales. Effectivement, comme dans un modèle de régression linéaire, plus l'échantillon est grand, plus l'estimation du coefficient causal ou probabiliste est précise. Par exemple, un échantillon large dans un modèle de régression sur discontinuité permet de restreindre la largeur de bande autour du seuil, garantissant ainsi une distribution presque parfaitement aléatoire des données et une validité plus élevée à l'estimation de l'effet causal.

Un autre exemple pourrait être l'utilisation du « matching », souvent utilisé dans les études économétriques. Supposons que vous souhaitez estimer l'effet d'un programme éducatif sur les résultats scolaires d'étudiant.e.s. Le devis de recherche idéal serait d'assigner aléatoirement les étudiant.e.s au programme (le groupe traitement) ou non (le groupe contrôle). Toutefois, puisque ce devis idéal peut être difficilement réalisable, un grand nombre de données pourrait permettre de trouver pour chaque étudiant.e dans le groupe traitement un étudiant.e « jumeau » dans le groupe contrôle. Ce « jumeau » serait similaire en âge, sexe, antécédents socio-économiques, etc. Il serait ensuite possible de comparer les résultats scolaires de ces jumeaux pour estimer l'effet du programme. Plus l'échantillon est grand, plus l'estimation sera précise et fiable, parce qu'il y aura plus de jumeaux possibles à appairer, réduisant ainsi le biais dû aux variables non observées.

Il s'agit d'un exemple où les données massives augmentent la validité interne de l'étude, même si les données sont de nature observationnelle et non expérimentale.

Deuxièmement, un échantillon de données massives observationnelles issues d'une plateforme comme X — anciennement Twitter — ou Facebook peut fournir une *description* plus fine de certaines dynamiques sociales observées sur les réseaux sociaux. Cependant, c'est la manière dont sont collectées les données de cet échantillon de données massives qui garantit la représentativité de l'échantillon — avec pour objectif l'absence d'un biais de sélection — et non pas la quantité de données. Généralement, le biais d'un échantillon est une conséquence de la non-représentativité des répondants; dans notre exemple, les utilisateurs des médias sociaux ne sont généralement pas représentatifs de la population entière.

Dans un tel cas, des méthodes de pondération sur des données observationnelles peuvent compenser pour la sur- ou la sous-représentativité de sous-groupes dans un échantillon afin d’assurer la validité de l’inférence entre échantillon et population. Les données massives ont ici une importance puisqu’une pondération fiable nécessite une quantité substantielle d’observations. Une pondération *a posteriori* sera donc plus fiable plus l’échantillon est grand. Les données massives ont ainsi une valeur ajoutée afin d’établir des inférences descriptives plus précises et sophistiquées.

Validité écologique et observation par sous-groupes

Les données massives peuvent aussi jouer d’autres rôles importants relatifs à la validité externe. Premièrement, les données massives facilitent effectivement la validité externe de certaines études en accroissant la validité écologique (*ecological validity*) des tests expérimentaux, c’est-à-dire le réalisme de la situation expérimentale (Grimmer, 2015: 81). En effet, la variété des sources et des formats de données permet aux chercheurs d’imiter plus fidèlement la réalité sur le terrain vécue par les participants aux études.

Deuxièmement, la quantité importante de données rend possible l’observation d’effets précis, spécifiques et inédits par sous-groupes (Grimmer 2015: 81). Alors qu’auparavant, la taille réduite des échantillons ne permettait pas d’effectuer des inférences valides pour des sous-groupes de la population — les écarts-types par sous-groupes étaient trop grands, rendant difficile l’estimation précise d’un paramètre comme la moyenne et impossible celle d’un coefficient —, la taille énorme des échantillons de données massives permet aux chercheurs d’estimer des paramètres qui étaient demeurés extrêmement imprécis jusqu’à aujourd’hui. Notre compréhension des phénomènes sociaux s’en trouve par le fait même approfondie de façon considérable.

	Données observationnelles	Données expérimentales
Processus de génération des données	Non contrôlé par le chercheur	Contrôlé par le chercheur
Type d’inférence causale	Locale (LATE) ou populationnelle (ATE)	Populationnelle (ATE)
Méthodes	Approches par design	Distribution aléatoire du traitement
Exemples	Régression sur discontinuité, variable instrumentale	Expérience de terrain, laboratoire

Figure 1: image2_2

Conclusion : trois questions ouvertes pour le futur

Comme nous venons de le voir, la quantité et la variété nouvelle des données massives permettent à la fois un approfondissement de l'analyse de certains phénomènes et l'ouverture de nouvelles avenues de recherche. L'analyse des données massives peut permettre de mettre en lumière des tendances subtiles échappant aux ensembles d'informations plus restreints.

Il faut toutefois souligner que les données massives représentent une complexification de l'analyse des phénomènes en sciences sociales d'une perspective non pas seulement méthodologique/technique mais également épistémologique.

Cela soulève au moins trois questions d'importance, dont les réponses ne nous sont pas encore accessibles, pour l'avenir de la recherche en sciences sociales : (1) les données massives entrent-elles (partiellement du moins) en conflit avec l'impératif de parcimonie qui caractérise la science moderne?; (2) ces données sont-elles dans la continuité ou représentent-elles une coupure dans la tradition béhavioraliste en sciences sociales (et en science politique tout particulièrement)?; (3) et finalement, de manière reliée, les données massives proposent-elles ou non une manière de dépasser l'individualisme méthodologique qui caractérise les sciences sociales contemporaines?

1 Le logiciel libre et le code source ouvert

Ce chapitre ne présentera pas un outil en soi. Il vise à initier les lecteurs et les lectrices à la philosophie du logiciel libre et surtout de situer ces réflexions dans le contexte de la recherche en sciences sociales. En fait, l'outil ici est plutôt de l'ordre réflexif que concret. À la fin de ce chapitre, les lecteurs et lectrices seront en meilleures positions afin de situer les outils qui seront présentés dans le grand univers des logiciels libres et payants. Ils et elles pourront comprendre les motivations derrière le développement et l'utilisation de tels logiciels. Sans vouloir divulguer quoi que ce soit, la création et l'utilisation de logiciels libres dépassent le simple calcul coûts et bénéfices utilitaires, gratuit contre payant. Tout cela s'inscrit dans des réflexions philosophiques, éthiques et épistémologiques plus grandes qui continuent, à ce jour, d'influencer les développeurs et les utilisateurs. Nous verrons tout au long de ce chapitre que certaines de ces motivations rejoignent la méthode scientifique, qui est au cœur de notre quête de savoir et de compréhension du monde sociale (King et al., 2021). Ainsi, ce chapitre jette la base réflexive qui est derrière les choix des outils numériques présentés dans ce livre, où nous avons tenté de joindre les avantages des logiciels libres avec certains qui sont payants dans le but de créer un environnement de travail à la fois individuel et collaboratif. Avant d'aller plus en profondeur, une certaine distinction mérite d'être faite, qui permettra d'éviter certaines confusions quant au but de ce livre et à la terminologie utilisée.

Il est donc important de faire une première distinction entre une méthode, dite méthodologique, et un outil numérique. La méthodologie est le champ de la philosophie des sciences qui s'intéresse à l'étude des méthodes scientifiques ou techniques. Celles-ci visent à collecter et à analyser des données, suivant les impératifs scientifiques, et qui ont pour but de contribuer au savoir et à la connaissance. Il faut faire attention puisque parfois, dans certains livres ou dans certains articles, il est possible que les auteurs ou les autrices parlent des méthodes qu'ils et elles ont utilisées comme étant des *outils*. D'un point de vue méthodologique, lié à la philosophie des sciences, il est approprié d'utiliser ce genre de vocabulaire. Ainsi, la régression linéaire, le *clustering*, les entretiens semi-dirigés et l'analyse de contenu sont des méthodes. En revanche, les outils dits numériques qui sont présentés dans ce livre ne sont pas des méthodes scientifiques. Des outils numériques comme R, Dropbox ou GitHub ne sont pas des méthodes. Ils sont des outils qui permettent de structurer sa pensée, d'organiser son espace et son environnement de travail, et d'implémenter son protocole de recherche afin de collecter et d'analyser des données et d'en dériver des conclusions. Il n'est donc pas approprié de considérer qu'un outil numérique est synonyme de méthode.

Ce livre, et par extension ce chapitre, ne vise pas à présenter des *outils méthodologiques* - compris ici comme étant des outils qui permettent de *désigner*, d'exécuter et d'évaluer une re-

cherche (Brady & Collier, 2010). Ils visent plutôt à présenter des *outils numériques* qui, comme mentionnés dans le paragraphe précédent, permettent de *structurer sa pensée, d'organiser son espace de travail et d'implémenter certaines méthodes*. Cette distinction est importante pour le reste de ce livre, et surtout pour la compréhension de son contenu. Les lecteurs et les lectrices, au fil des pages, acquerront des compétences et du savoir à propos des outils numériques. Celles-ci leur permettront de développer un nouveau langage à partir duquel ils et elles pourront réfléchir et penser leur recherche, et surtout interagir avec les autres personnes dans leurs champs; avec qui ils et elles pourront plus aisément collaborer en organisant leur environnement de travail; avec qui ils et elles pourront partager des documents et leurs résultats.

Pour atteindre ces objectifs, il est important de commencer pas la base - comprendre d'où vient le logiciel libre et qu'est-ce que c'est. Le logiciel libre a une place importante dans les outils numériques, sans parler de l'influence qu'il a eue et qu'il continue d'avoir aujourd'hui. Afin de bien comprendre ce dont il est question, nous présenterons, dans un premier temps, l'historique de cette philosophie et de ce mouvement afin de le situer temporellement. De cette façon, nous pourrons mieux comprendre ses motivations et ses revendications, mais aussi ses influences actuelles. Ensuite, nous distinguerons le logiciel payant du logiciel libre, pour ensuite aborder la différence entre le logiciel libre et le code ouvert. Après coup, nous aborderons en quoi ces réflexions sont intéressantes et importantes pour les sciences sociales à l'ère du numérique, ainsi que les avantages et les inconvénients qui y sont liés. À partir de cette section, nous pourrons montrer comment les outils numériques s'inscrivent dans chacune des grandes étapes de la recherche - avant, pendant et après. Finalement, avec ces quelques notions en poche, nous présenterons les différents critères sur lesquels nous nous sommes appuyés pour sélectionner les différents outils qui sont présentés dans ce livre. Ces critères sont nés de la jonction entre la philosophie du logiciel libre et l'expérience de recherche en tant que chaire, dont avec la participation de collaborateurs externes.

1.1 Logiciels Libres

1.1.1 Le monde du libre

« Vous n'avez pas à suivre une recette avec précision. Vous pouvez laisser de côté certains ingrédients. Ajouter quelques champignons parce que vous en raffolez. Mettre moins de sel, car votre médecin vous le conseille — peu importe. De surcroît, logiciels et recettes sont faciles à partager. En donnant une recette à un invité, un cuisinier n'y perd que du temps et le coût du papier sur lequel il l'inscrit. Partager un logiciel nécessite encore moins, habituellement quelques clics de souris et un minimum d'électricité. Dans tous les cas, la personne qui donne l'information y gagne deux choses : davantage d'amitié et la possibilité de récupérer en retour d'autres recettes intéressantes. » - Richard Stallman (Williams et al., 2010)

Cette analogie illustre bien trois concepts au cœur de la philosophie de Richard Stallman, souvent considéré comme le père fondateur du logiciel libre : liberté, égalité, fraternité. Les

utilisateurs de ces logiciels sont libres, égaux, et doivent s'encourager mutuellement à contribuer à la communauté. Ainsi, un logiciel libre est généralement le fruit d'une collaboration entre développeurs qui peuvent provenir des quatre coins du globe. Au centre de ce mouvement se trouve une réflexion éthique, dont les militants font compagne depuis le début des années 1980, à propos de la liberté des utilisateurs. La Free Software Foundation (FSF), fondée par Richard Stallman en 1985, définit rapidement le logiciel «libre» [free] comme étant garant de quatre libertés fondamentales de l'utilisateur: la liberté d'utiliser le logiciel sans restrictions, la liberté de le copier, la liberté de l'étudier, puis la liberté de le modifier pour l'adapter à ses besoins et le redistribuer¹. Il s'agit ainsi d'un logiciel dont le code source² est disponible, afin de permettre aux internautes de l'utiliser tel quel ou de le modifier à leur guise. L'accès au code source devient essentiel afin de permettre à l'utilisateur de savoir ce que le programme fait réellement. Seulement de cette façon, l'utilisateur peut *contrôler* le logiciel, plutôt que de se faire contrôler par ce dernier (Stallman, 1986).

1.1.2 Émergence et sémantique du *libre*

Plusieurs situent les débuts du mouvement du logiciel libre avec la création de la licence publique générale GNU, en 1983, à partir de laquelle va se développer une multitude de programmes libres. Parmi les plus populaires, on retrouve notamment le navigateur Firefox, la suite bureautique OpenOffice et l'emblématique système d'exploitation Linux, qui se développe d'ailleurs à partir de la licence GNU³. Aujourd'hui, il s'agit d'un véritable phénomène sociétal: des milliers d'entreprises, d'organisations à but non lucratif, d'institutions ou encore de particuliers adoptent ces logiciels, dont la culture globale et les valeurs (entraide, collaboration, partage) s'arriment avec le virage technologique de plusieurs entreprises. Les logiciels libres ont différents usages, en passant par la conception Web, la gestion de contenu, les systèmes d'exploitation, la bureautique, entre autres. Ils permettent donc de répondre à plusieurs types de besoins numériques et informatiques.

Attention, le logiciel libre est avant tout une philosophie, voire un mouvement de société. C'est une façon de concevoir la communauté du logiciel, où le respect de la liberté de l'utilisateur est un impératif éthique (Williams et al., 2010). Par conséquent, le terme libre, *free* en anglais, porte à confusion. Celui-ci ne signifie pas qu'un logiciel libre est nécessairement gratuit. Certes, plusieurs sont effectivement téléchargeables gratuitement. Toutefois, il est aussi possible de (re)distribuer des logiciels libres payants. Par ailleurs, aucun logiciel libre n'est réellement « gratuit » dans la mesure où son déploiement et son utilisation nécessitent généralement différents coûts, dont les degrés sont variables en fonction des compétences et de l'infrastructure

¹La redistribution doit évidemment respecter certaines conditions précises, dont l'enfreint peut mener à des condamnations [<http://www.softwarefreedom.org/resources/2008/shareware.html>]

²Pour rester dans les analogies culinaires, le code source est au logiciel ce que la recette est à un plat: elle indique les actions à effectuer, une par une, pour arriver à un résultat précis. Encore une fois, ce dernier peut-être adapté, modifié, bonifié.

³Pour une liste plus exhaustive, les lecteurs et lectrices peuvent aller consulter le répertoire du gouvernement du Canada: <https://code.open.canada.ca/fr/logiciels-libres.html#>

dont disposent les utilisateurs (coût d'apprentissage, coûts d'entretien, etc.). Enfin, il est important de garder en tête que les logiciels libres possèdent eux aussi une licence - cette dernière est d'ailleurs garante des libertés que confèrent les logiciels libres aux utilisateurs.

La grande liberté que ce type de logiciel offre favorise notamment la collaboration entre les utilisateurs, et ce, à une échelle pouvant être internationale. Les interactions entre les chercheurs créent une dynamique d'« innovation ascendante » et d'entraide (Couture, 2014). En d'autres termes, l'accessibilité et la collaboration favorisent le développement et l'amélioration de ces logiciels. Selon certains, et comparativement aux logiciels privés, les logiciels libres ont un niveau plus élevé d'innovation (Smith, 2002). Contrairement aux logiciels propriétaires, ceux qui se développent de manière privée et fermée, les logiciels libres permettent à tous les utilisateurs de participer au développement. Ceux-ci partagent ensuite leurs améliorations, ce qui stimule à son tour de nouvelles initiatives. De plus, il est raisonnable de penser que l'utilité des améliorations, ainsi que l'utilisation qui en est faite par les utilisateurs, permet de générer un savoir collaboratif (**couture20a?**).

Il y a aussi certains avantages économiques, dont un faible coût d'acquisition et de renouvellement pour les particuliers. Cet avantage individuel génère plusieurs externalités positives. Tout d'abord, certains logiciels statistiques ainsi que certains programmes informatiques coûtent plusieurs centaines, voire des milliers de dollars, et dans certains cas doivent être renouvelés annuellement. Cela augmente les coûts associés à l'utilisation du logiciel et par conséquent limite son accessibilité. Comparativement, pour les logiciels libres, la licence d'acquisition coûte bien souvent moins cher, et aucun renouvellement de licence n'est demandé dans la plupart des cas. L'argent sauvé des licences peut alors être investi dans le développement du logiciel libre (Béraud, 2007). De plus, étant donné que les chercheurs doivent souvent faire face à des contraintes budgétaires, les logiciels libres deviennent des outils intéressants afin de minimiser les coûts de la recherche (**yu_munoz-justicia22a?**). Il s'agit d'un avantage encore plus important et intéressant pour les chercheurs dans les pays du Sud global (Santillán-Anguiano & González-Machado, 2023). L'accessibilité de ces ressources permet donc de réduire l'écart dans la production scientifique entre les pays du Sud et ceux du Nord. De plus, elle permet à tous de bénéficier d'outils pédagogiques accessibles, ce qui favorise l'acquisition ainsi que le développement de compétences méthodologiques.

Dans le cadre d'une formation universitaire, il peut être pertinent d'enseigner aux étudiants à se servir de logiciel statistique ou d'analyse de texte. L'acquisition de ces compétences peut être précieuse tant pour ceux et celles qui souhaitent se diriger vers le milieu académique, que pour ceux et celles qui visent le marché professionnel. D'ailleurs sur le site web de la banque d'emplois du gouvernement du Canada, les conditions d'emplois sont en ce moment⁴ très bonnes, et une pénurie de main-d'œuvre est anticipée, entre 2022-2031, dans les emplois en analyse de données. Ces compétences sont d'autant plus précieuses aujourd'hui, dans le monde de données dans lequel nous vivons.

⁴En date d'écrire ces lignes, avril 2024.

Il est important de souligner que la transition vers les logiciels libres ne doit pas se faire seulement sur des bases économiques, mais dans une perspective globale de changement de culture. Changer pour des raisons purement économiques viendrait à violer l'essence même de la philosophie du logiciel libre, qui se veut surtout être un esprit de collaboration et de transparence. Par conséquent, il est important d'incorporer aussi les valeurs et la philosophie dans notre utilisation. Amélioration constante, entraide, savoir partagé et plusieurs milliers de contributeurs (Couture, 2014), ces éléments résument très bien la philosophie du logiciel libre.

1.1.3 Illustrations

Considérons quelques exemples de logiciels libres et de logiciels payants afin de mettre en relief les trois éléments présentés ci-haut: 1) la liberté de l'utilisation et de la contribution, 2) les coûts d'acquisition et 3) les compétences acquises et développées. Dans les chapitres suivants, des outils numériques tels que R, SPSS, STATA, qui permettent de mener des analyses statistiques, ou encore la suite Office, Quarto, LaTeX, qui permettent de formater un document écrit, seront présentés. Cette section ne remplace en aucun cas une lecture approfondie et détaillée des chapitres suivants. Au besoin, nous recommandons fortement aux lecteurs et aux lectrices de se référer au reste du livre. En ce qui concerne le premier trio, R est le seul logiciel libre du groupe. Il est accessible gratuitement à partir du site web de CRAN, et une grande communauté d'utilisateurs contribue activement à son développement. Par exemple, la compagnie Posit est derrière le développement de **RStudio**, Quarto et Positron⁵ qui sont toutes des extensions à code ouvert de R. Ensuite, plusieurs utilisateurs ont développé des bibliothèques avec des commandes et des fonctions supplémentaires, qui sont gratuites et dont le code est disponible sur des plateformes comme GitHub.

Contrairement à R, des logiciels comme SPSS et STATA, qui sont des logiciels propriétaires, nécessitent une licence privée afin de pouvoir les utiliser. L'achat d'une licence doit aussi être situé avec ses propres besoins puisque, dans le cas de SPSS, les licences ne donnent pas toutes accès aux mêmes fonctionnalités. Ainsi, la licence de base ne donne pas accès à l'utilisation de la régression, alors que la licence Premium le permet. De plus, la licence doit être renouvelée tous les ans, et les prix varient entre 1 700\$ et 5 194\$ pour la licence web à un seul utilisateur. En ce qui concerne STATA, la logique est similaire à celle de SPSS. Différents types de licence sont offerts avec des fonctionnalités supplémentaires, notamment en termes de rapidité de l'exécution des fonctions et de la capacité à traiter une large quantité d'observations. Les licences annuelles éducationnelles, donc pour les étudiants, se situent entre 126\$ et 506\$ par année. Autrement, le prix varie en 1 248\$ et 1 950\$ par année. De plus, SPSS et STATA sont développés uniquement par les compagnies IBM et par StataCorp respectivement. Bien qu'ils n'offrent pas la même flexibilité que R et que les utilisateurs ne peuvent pas contribuer au développement au même titre que R, ils offrent d'importantes ressources pour les utilisateurs,

⁵Un nouvel IDE, qui est toujours en cours de développement, qui souhaite offrir une interface optimisée pour R et Python construit sur VS Code

notamment par l’entremise d’un service à la clientèle, de documentations et de formation web, par exemple. Ainsi, les utilisateurs sont “pris en charge” directement par la compagnie afin de leur fournir de l’aide et du support.

En ce qui concerne les langages de balisage, LaTeX ou Quarto peuvent être utilisés gratuitement sur des interfaces comme RStudio ou VS Code. Ils offrent donc une grande flexibilité étant compatible avec plusieurs interfaces. Ainsi, une fois que les compétences avec le langage ont été développées, il est facile de les transposer d’une interface à une autre. À l’inverse, la suite Office offre ses propres interfaces qui sont plutôt intuitives, mais qui contraignent l’utilisateur à des fonctions prédéfinies. De plus, le coût d’acquisition d’une licence de la suite Office varie entre 79\$ et 109\$ par année. Similairement à STATA et SPSS, Microsoft offre plusieurs ressources directement sur leur site web pour les utilisateurs. Ainsi, un certain “encadrement” est offert par la compagnie.

Deux derniers éléments sont importants d’être soulevés. Il ne faut pas penser que les logiciels libres sont dénués de support et de documentations. La plupart des logiciels libres, et des extensions comme les librairies sur R, offrent beaucoup de documentations, et plusieurs forums d’utilisateurs partagent leurs problèmes et leurs solutions. De plus, certains tutoriels sont disponibles sur YouTube ou sur des plateformes comme Datacamp et CodeAcademy. Ainsi, les utilisateurs ne sont pas totalement laissés à eux-mêmes. De plus, plusieurs universités offrent des licences pour des logiciels, comme Office ou SPSS, aux étudiants et aux étudiantes pour éviter que ceux-ci aient à déboursier d’importantes sommes supplémentaires afin d’acquérir ces logiciels.

1.1.4 Logiciel libre et code ouvert

Parallèlement au logiciel libre, il y a aussi le code ouvert, ou *open source*. A priori, la dénomination du logiciel libre et celle du *code ouvert* semblent suggérer qu’il s’agit de synonymes. Dans les deux cas, le lecteur pourrait croire que l’on fait référence à des logiciels, par exemple, qui sont exempts de restrictions d’utilisations et auxquelles les utilisateurs peuvent participer au développement. Cependant, il y a une distinction importante entre les deux.

Bien que les deux renvoient sensiblement aux mêmes types de logiciels, les tenants de ces approches ne partagent pas la même perspective. Comme Stallman (2022) l’explique, le logiciel libre est d’abord et avant tout un mouvement qui fait « campagne pour la liberté des utilisateurs de l’informatique ». Le code ouvert, quant à lui, met l’accent sur les avantages pratiques, plutôt que de militer pour des principes.

Le terme *code ouvert* sera introduit seulement en 1998 afin de clarifier l’ambiguïté dans la dénomination « logiciel libre » ⁶, *free software* en anglais, afin de spécifier que le code source était accessible, et non pas que le logiciel était « gratuit » (ballhausen19a?). De plus, les

⁶Soit ceux qui ont été conçus suivant les principes philosophiques et « moraux » qui sous-tendent ce mouvement.

logiciels à code ouvert doivent respecter un certain nombre de critères quant à la distribution de leurs logiciels (Open Source Initiative, 2006).

Rappelons tout d’abord que le logiciel libre se définit sur la base de quatre libertés: 1) liberté d’utiliser le programme comme désiré; 2) liberté d’étudier le fonctionnement du programme et de le modifier pour ses propres besoins; 3) liberté de redistribuer des copies; 4) liberté de distribuer des copies de la version « améliorer » du programme pour ses pairs (**ballhausen19a?**). Concernant le *code ouvert*, tout logiciel qui souhaite être inclus sous cette appellation doit respecter dix critères: 1) Redistribution gratuite; 2) doit inclure le code source; 3) doit permettre les modifications et les travaux dérivés; 4) l’intégrité du code source; 5) ne doit pas discriminer des personnes et/ou groupes; 6) ne doit pas restreindre personne dans l’utilisation du logiciel pour un domaine d’activité; 7) distribution d’une licence pour l’utilisation; 8) la licence ne doit pas être spécifique pour un produit; 9) la licence ne doit pas placer de restriction sur d’autres programmes; 10) la licence doit être technologiquement neutre⁷ (Open Source Initiative, 2006).

Il est aussi utile de les distinguer des logiciels « non libres », soit les logiciels propriétaires: « Son utilisation, sa redistribution ou sa modification sont interdites, ou exigent une autorisation spécifique, ou sont tellement restreintes qu’en pratique vous ne pouvez pas le faire librement » (Système d’exploitation GNU, 2023). Par contraste, la licence libre confère des droits de propriétaire. L’utilisateur a le droit d’installer le logiciel sur autant d’ordinateurs que désiré, le modifier selon ses besoins et le distribuer avec ou sans ses modifications. Il peut même demander d’être payé pour distribuer des copies, avec ou sans ses modifications.

Le logiciel libre et le *code ouvert* ont certaines similitudes puisqu’ils adhèrent tous les deux à la même vision du logiciel, ainsi que de son accessibilité. Toutefois, il est important tout de même de les distinguer puisqu’ils ont des origines différentes, et qu’ils mènent à certaines pratiques qui sont différentes. La prochaine section utilise un cas concret afin d’expliquer l’effet du libre, et l’utilité que cela peut avoir.

1.2 Les sciences sociales à l’ère du numérique: les enseignements de la philosophie du logiciel libre

En quoi est-ce que ces deux concepts, issus du monde de l’informatique, sont-ils intéressants et/ou important pour les sciences sociales? Pour répondre à cette question, il est important de retourner à la base, soit de se questionner sur ce que constitue la recherche scientifique dans les sciences sociales.

⁷Pour plus d’informations sur ces caractéristiques, nous encourageons les lecteurs à se référer au lien web de Open Source Initiative (2006). Ils y trouveront un contenu détaillé pour chacune des caractéristiques susmentionnées.

Dans leur célèbre ouvrage *Designing Social Inquiry*, King et al. (2021)⁸ propose quatre critères qui définissent la recherche dite scientifique: 1) le but est l'inférence; 2) les procédures sont publiques; 3) les conclusions sont incertaines; 4) le contenu est la méthode. La philosophie du code ouvert et les avantages pratiques du logiciel libre s'arriment parfaitement avec plusieurs de ces critères.

Penchons-nous sur le critère de la transparence des procédures et celui de la méthode comme étant le contenu. Plusieurs outils numériques rendent possibles le partage et l'accès public des données et de la méthode utilisée. Certains de ces outils seront d'ailleurs abordés dans les chapitres suivants. L'accès aux données et aux procédures d'analyse est un impératif scientifique. Comme nous l'aborderons un peu plus loin, il y a toujours des concessions à faire lors des investigations scientifiques. Par conséquent, un meilleur accès aux procédures et aux données utilisées permet de cibler plus facilement les limites de certaines recherches et de les combler lors de recherche ultérieure. De plus, l'arrivée des données massives ouvre de nouvelles portes, mais surtout de nouveaux défis relatifs à la validité interne et externe ainsi qu'au type de données récoltées et à la validité écologique. Le livre de Marres (2017) est très intéressant à ce sujet. Face au constat que la vie sociale se trouve affecter par les changements numériques, il nous faut en tant que chercheur du monde social réfléchir à notre façon de comprendre les changements qui s'opèrent. Ainsi, face à ces défis, une des solutions se trouve notamment dans un meilleur partage et dans une meilleure accessibilité aux données et aux procédures.

Il est possible de faire certains liens avec les deux autres critères de King et al. (2021). Premièrement, comme le but de la science est l'inférence, soit tenter d'expliquer des phénomènes sociaux qui s'inscrivent dans des catégories plus larges que nos observations directes⁹, il est important que nos conclusions soient soumises au plus grand nombre possible et pas uniquement au comité éditorial d'une revue scientifique. D'une part, la validité de nos résultats a un potentiel politique important. Plusieurs décisions peuvent être prises sur la base des connaissances et de la compréhension des dynamiques sociales. Il est donc important que les résultats de recherche qui informent ces décisions soient le plus rigoureux possible, et qu'ils aient été soumis à l'examen critique par le plus grand nombre d'individus. D'autre part, et comme King et al. (2021) le font remarquer, les conclusions sont toujours incertaines. L'examen critique et la reproduction des protocoles sont donc une nécessité dans ce contexte d'incertitude constant. La science ne cherche pas à être dogmatique. Au contraire, l'incertitude caractérise bien la science. Les résultats sont toujours incertains, et ce que la recherche vise à faire c'est de renforcer notre niveau de confiance envers certaines explications tout en écartant les explications alternatives. Comme plusieurs de ces phénomènes ne sont pas homogènes et qu'ils ne sont pas immuables, nous devons constamment revoir les explications et notre compréhension de ces dynamiques.

Dans la même lancée, l'ouvrage *Rethinking Social Inquiry* (Brady & Collier, 2010), une réponse à King et al. (2021), partage, en partie, cette définition de la recherche scientifique. Pour les

⁸Ce livre est aussi connu sous l'acronyme *KKV*, en référence à la première lettre du nom de famille de chacun des auteurs.

⁹Par exemple, les mouvements sociaux, le comportement électoral, les guerres et les révolutions, et bien plus.

auteurs, les scientifiques du monde social possèdent plusieurs outils qui leur permettent de *designer, d'exécuter et d'évaluer* une recherche. Ces outils sont des procédures et des pratiques employées par les chercheurs des traditions quantitatives et qualitatives. Ils ont aussi des techniques analytiques qui leur permettent de *développer des preuves qui sont convaincantes*¹⁰.

Chacun de ces outils a toutefois ses forces et ses faiblesses. Inspiré de Przeworski & Teune (1970), Brady & Collier (2010) parlent ainsi de *compromis*. Selon eux, la pertinence d'un outil méthodologique dépend de la question de recherche, du but de la recherche et de son contexte. Le choix d'un outil, sur cette base, entraîne des compromis qui empêchent d'atteindre tous les buts analytiques simultanément. Même lorsqu'il y a adéquation entre la question et la méthode, plusieurs autres embûches peuvent entraîner ces compromis, comme la disponibilité des données par exemple, qui limitent par la suite la qualité et la précision des résultats, ce qui explique, en partie, l'incertitude envers les conclusions.

Il est intéressant de concevoir la réalité sociale comme étant un prisme ayant un nombre infini de face. Chacune de ces faces correspond à une compréhension partielle de la réalité. Pour y avoir accès, nous devons utiliser une méthode de collecte et d'analyse de données, informées par une question de recherche et/ou par la théorie¹¹. Étant donné que chaque outil méthodologique a ses limites et que nous devons constamment faire des compromis, notre compréhension de la réalité n'est que partielle une fois la recherche complétée. De plus, il ne faut pas oublier que la compréhension que nous avons de cette face reste incertaine.

En sommes, les enseignements de la philosophie du code ouvert, les avantages pratiques du logiciel libre et les outils qui en découlent ont le potentiel de permettre non seulement une plus grande transparence des protocoles scientifiques, mais aussi un plus grand partage du savoir, et ce, du début de la recherche jusqu'à la publication des résultats.

1.3 Inconvénients et défis

Jusqu'à présent, nous avons surtout présenté des avantages liés aux logiciels libres. Toutefois, il n'y a pas que des points positifs, et ne pas aborder certaines limites serait malhonnête.

¹⁰L'objectif des auteurs est de permettre le dialogue et la réconciliation entre les tenants des deux principales traditions méthodologiques: les quantitativistes et les qualitativistes. D'où leur vision de la science comme se devant de développer des preuves qui seront considérées comme convaincantes par les chercheurs de ces deux traditions.

¹¹Nous préférons formuler la phrase ici en stipulant que le choix d'une méthode de recherche peut être informé par la théorie et notre question de recherche simultanément, mais peut aussi être fait seulement à partir d'une question de recherche. Cela fait référence à deux processus de recherche différents soit la méthode hypothético-inductive, et celle déductive.

1.3.1 La courbe d'apprentissage

Dans leur texte, Paura & Arhipova (2012) soulèvent une critique faite envers certains logiciels libres, notamment envers R. Le problème principal d'enseigner les statistiques avec des logiciels libres est qu'ils peuvent être compliqués à apprendre ainsi qu'à utiliser; par conséquent, les étudiants passeraient plus de temps à tenter de résoudre les erreurs de programmation plutôt que d'apprendre les statistiques¹². Il est vrai que ces logiciels demandent un investissement en temps, afin d'être en mesure de mener ses propres analyses statistiques. Par exemple, R demande l'apprentissage d'un langage de programmation afin de pouvoir utiliser le logiciel à son plein potentiel. De plus, la syntaxe de certaines librairies demande aussi un certain temps d'adaptation. À titre de comparaison, le logiciel SPSS offre une interface beaucoup plus intuitive que R, dans lequel l'utilisateur peut simplement cliquer sur les différents menus dans la barre d'outils afin de sélectionner les analyses qu'il ou elle souhaite faire. SPSS présente ses résultats dans des tableaux qui sont clairs et lisibles, contrairement à R où plusieurs fonctions présentent les résultats directement dans la console, sous un format moins "esthétique". De plus, la plupart des logiciels payants viennent avec un certain service à la clientèle. En d'autres termes, lorsque les utilisateurs rencontrent des problèmes techniques, ils peuvent se référer au manuel d'utilisateur ou bien par l'entremise de l'assistance technique qui est offerte par la compagnie. À l'opposé, la plupart des logiciels libres ne sont pas accompagnés d'un service à la clientèle. Les utilisateurs doivent donc se "débrouiller" par eux-mêmes lorsqu'ils rencontrent des difficultés et des problèmes.

Toutefois, lorsque l'on compare le coût d'apprentissage avec les bénéfices tirés, il est plus difficile de soutenir qu'il s'agit uniquement d'un désavantage. Dans un premier temps, la syntaxe de programmation de R n'est pas parmi les plus complexes à apprendre, et elle s'intègre très bien avec certaines extensions, dont Quarto ou RMarkdown, qui offrent de multiples possibilités pour transmettre le résultat de ses recherches, que ce soit par l'entremise d'un rapport, d'un article scientifique, un site web ou même un blogue. Surtout, la logique derrière la syntaxe de base de R et celle d'une nouvelle librairie reste sensiblement inchangée. Par conséquent, lorsque nous avons une bonne compréhension du fonctionnement de base de R, l'apprentissage d'une nouvelle librairie se fait relativement rapidement. Certaine, comme `dplyr` du `tidyverse` facilite grandement la manipulation des données comparativement aux commandes de base. Dans un deuxième temps, en comparant le coût, soit d'apprendre le langage de R, avec les bénéfices, de mener ses propres analyses de données et de formater les résultats pour les présenter, il est assez clair que tous ceux et celles qui souhaitent, de près ou de loin, travailler avec des données quantitatives, les bénéfices dépassent largement le coût. D'autant plus que ces compétences s'inscrivent dans la longue durée, alors que l'apprentissage est plutôt de courte à moyenne durée. Dans un troisième temps, bien que certains logiciels offrent des alternatives plus intuitives, elles n'offrent pas la même flexibilité que la plupart des logiciels libres offrent. Pour résumer, bien que l'apprentissage d'un langage de programmation demande un investissement en temps, les

¹²Sur cet enjeu, nous conseillons aux lecteurs et lectrices de lire le chapitre 2 sur les langages de programmation ainsi que le chapitre 8 sur l'intelligence artificielle. Plusieurs trucs et astuces seront présentés dans ces chapitres.

bénéfices générés par ces nouvelles compétences dépassent le coût initial. Finalement, bien que les logiciels libres n’offrent pas de service à la clientèle, il existe bien souvent des forums d’utilisateurs sur le web où il est possible de trouver des réponses à ses questions et/ou de poser ses questions. Par conséquent, bien qu’il ne s’agit pas d’un service directement offert par l’outil numérique en question, il est toujours possible de trouver du support et de l’aide par l’entremise de la communauté d’utilisateur.

1.3.2 Problème de transparence

L’arrivée des sciences informatiques a fait émerger des problèmes de reproductibilité des protocoles scientifiques (Janssen, 2017). Le problème principal est relatif à l’accès au code utilisé par les chercheurs. Par exemple, il est possible de réaliser des analyses statistiques avec R sans partager le code utilisé, ce qui limite la transparence du processus scientifique. Dans cette situation, il est difficile de savoir si des erreurs de codage ont été commises, volontairement ou involontairement, affectant ainsi les résultats partagés.

Afin de remédier à ce problème, certains outils tels que GitHub¹³ participent à la transparence des résultats scientifiques (**fortunato_galassi21a?**). Ce logiciel permet aux chercheurs de partager leur code afin qu’il puisse être accessible pour tous. Il est important de mentionner ici que l’installation et la configuration de GitHub peuvent s’avérer difficiles pour ceux et celles qui ne sont pas initiés à l’informatique. Cela constitue une certaine barrière dans son utilisation. Toutefois, nous souhaitons tout de même présenter l’utilité de ce logiciel puisqu’il permet de rendre les processus ainsi que les résultats de recherche plus transparents. Par exemple, si l’on réalise une analyse statistique de la relation entre l’économie et le vote, nous pourrions partager l’ensemble du code que nous avons utilisé sur GitHub. D’une part cela permettrait aux utilisateurs de vérifier si les résultats sont honnêtes, et d’autre part de réutiliser le code pour mener leurs propres analyses.

Cependant, le partage du code reste encore majoritairement volontaire. Janssen et al. (2020) soutiennent que plus d’effort et d’actions concertés doivent être mis en place afin d’améliorer l’accessibilité aux codes. Toujours selon ces auteurs, les journaux scientifiques pourraient exiger que les auteurs rendent leur code public lors du processus de publication. D’ailleurs, les résultats d’une expérience sur les facteurs qui influencent les chercheurs à partager leur code démontrent que les initiatives individuelles ne seront pas suffisantes pour une augmentation du partage du code (**krahmer_etal23a?**). Par conséquent, rendre le code accessible devrait devenir un standard institutionnalisé.

1.3.3 Appropriation capitaliste

Dans ce cas-ci, il s’agit plutôt d’un défi auquel le logiciel libre est confronté plutôt qu’une critique quant aux limites de son utilisation. En fait, l’accès au code source ainsi que la liberté

¹³une plateforme publique *code ouvert* sur laquelle nous pouvons héberger et partager notre code.

et la possibilité de contribuer au développement du logiciel constitue un avantage intéressant pour les compagnies privées. Par conséquent, nous avons assisté à une intégration partielle du logiciel libre dans la logique capitaliste (Bessen, 2002; Broca, 2013). Certaines compagnies profiteraient des utilisateurs comme une main-d'œuvre gratuite afin de bonifier leur logiciel, ce qui permet, dans certains cas, de générer des revenus commerciaux dont l'entreprise est la seule bénéficiaire (**couture20a?**). Attention, il ne faut pas penser que toutes les compagnies agissent de manière prédatrice. Le but ici est de souligner que certaines pratiques commerciales trouble l'essence du mouvement du logiciel libre, qui se veut davantage être un outil de collaboration accessible, plutôt qu'un moyen pour générer des profits. Il est important de garder en tête les valeurs et la philosophie qui a donné lieu à ce mouvement.

1.4 La chronologie de la recherche

Malgré ces limites et ces défis, nous pensons que les différents outils numériques ont leur place en sciences sociales, et qu'ils s'inscrivent parfaitement à chaque étape de la recherche. Bien que le partage soit encore majoritairement sur une base volontaire, adopter cette pratique dès maintenant est important pour s'engager vers une science plus ouverte et transparente. Quant à cette dernière caractéristique, il ne faut pas croire que le partage des résultats se limite à une conférence ou à une publication scientifique. Bien au contraire, tout chercheur qui souhaite être le plus transparent doit s'engager dans ce processus dès la conception de sa recherche.

Avant de présenter différents outils qui existent pour cela, il faut préciser en quoi la transparence et l'accessibilité sont bénéfiques pour tous. D'une part, et comme nous l'avons présentée dans la section précédente, la transparence est une caractéristique fondamentale de toute recherche qui se veut scientifique. Elle est aussi liée à l'intégrité du chercheur. Il est impératif de faire état du processus qui mène à notre conclusion, de la sélection de nos données jusqu'à l'analyse. C'est de cette façon que nous pouvons juger de la validité et de la fiabilité des inférences, et surtout de ses limites. D'autre part, la transparence favorise aussi l'apprentissage (King et al., 2021). De rendre publique et accessible, à l'aide des outils numériques, les données, le code et la ou les méthodes utilisées permet de contribuer non seulement aux débats méthodologiques, mais aussi permet à d'autres chercheurs d'apprendre sur l'utilité et l'utilisation de ces méthodes.

1.4.1 Avant la recherche

Au fil des prochains chapitres, les lecteurs et lectrices apprendront une nouvelle langue. Au même titre qu'une langue comme le français, l'anglais ou le japonais, ce langage permet de communiquer, et surtout de réfléchir. De nouveaux concepts, une façon de penser et de réfléchir à leur recherche, et surtout à son organisation. C'est au travers de la langue que nous pouvons réfléchir. Ainsi, à l'aide de ces termes, les lecteurs et les lectrices seront outillés pour concevoir leur recherche et leur organisation avant même de l'entamer. De la visualiser dans toute son

ampleur, de cibler leurs besoins en fonction de leur but et de leurs intérêts et ainsi organiser leur environnement de travail en conséquence.

De plus, motivée par cet objectif inhérent à la science, la transparence exige du chercheur un engagement dès les premières étapes de sa recherche. Tout d’abord, il ou elle peut déjà établir son *workflow* en créant son dépôt GitHub dans lequel il rendra accessibles son code et ses données. Plus d’informations seront présentées dans le chapitre 3. Ensuite, une fois le *design* de recherche¹⁴ fait, le ou la chercheur peut le préenregistrer sur le site web de *Open Science Framework*¹⁵. L’objectif de ce site web est que les chercheurs puissent faire état de leurs hypothèses avant la collecte et l’analyse des données afin d’éviter toute manipulation malhonnête *post hoc*. Il s’agit d’un mécanisme qui se veut contraignant afin que les chercheurs s’en tiennent à ce qu’ils ou elles avaient prévu de mener comme recherche.

1.4.2 Pendant la recherche

Une fois la recherche débutée, plusieurs outils s’offrent pour une gestion efficace du flux de travail. Plusieurs d’entre eux seront présentés dans les chapitres suivants. Ces outils permettent notamment de sauvegarder les données et l’avancement du projet sur des serveurs externes (le *cloud*), comme Dropbox, afin d’éviter de tout perdre en cas de problème. D’autres permettent de produire le matériel nécessaire pour l’analyse de nos données, tel que R. Certains permet aussi de partager avec des collaborateurs l’avancement de notre projet et les scripts de nos analyses, comme Git et GitHub. D’autres permettent de structurer notre texte pour la production d’un article scientifique ou d’un chapitre de livre, comme Overleaf et Quarto. Ils permettent notamment d’ajouter des tableaux et des graphiques tirés directement des analyses, le tout dans un format respectant les exigences matérielles pour la publication.

Nous ne voulons pas sous-entendre qu’il n’y a pas de limites à tous ces outils. Chaque chapitre de ce livre présentera les points positifs et négatifs des outils qui y seront abordés.

1.4.3 Après la recherche

Une fois l’article écrit et publié, les auteurs peuvent décider de rendre accessible le matériel produit et utilisé en version gratuite sur le web. Il s’agit du *open science*, ou de la *science ouverte* (Chakravorty et al., 2022).

Cela permet de rendre non seulement les publications accessibles à tous sur internet gratuitement, mais aussi les données utilisées, par exemple, pour la réalisation de l’étude. Ces données peuvent ensuite être réutilisées par d’autres chercheurs. Il s’agit d’avantages très important, et surtout très prometteur. D’une part, un plus grand accès au savoir scientifique aux décideurs

¹⁴Généralement, un design de recherche comprend les éléments suivants: Introduction, revue des écrits, problématiques, une question de recherche, un cadre théorique, des hypothèses ainsi qu’une section sur la méthode et les données utilisées.

¹⁵Lien vers le site web: <https://osf.io>

publics permettra de prendre des décisions qui s'appuient sur la science, et idéalement, sur une pluralité de sources scientifiques afin de pouvoir évaluer adéquatement les effets positifs et négatifs. Finalement, la réutilisation des données peut réduire considérablement les coûts de la recherche. Bien souvent, la production des données peut engendrer des coûts importants. Par exemple, la production d'un sondage peut coûter plusieurs milliers de dollars, et les données sont bien souvent utilisées qu'une seule fois. Le partage des données de manière gratuite permet à des chercheurs qui n'ont pas toujours les moyens de financer un sondage d'avoir accès à des données. En somme, il s'agit de décloisonner le savoir des milieux académiques. Ce qui vaut non seulement pour les publications, mais aussi pour les données et les protocoles de recherche.

Toutefois, le libre accès reste confronté à certains défis. D'une part, il y a un transfert de la charge financière qui se fait parfois vers les chercheurs et les universités. Afin que les publications soient accessibles en libre accès, les chercheurs et les universités doivent souvent payer des sommes importantes. Cela soulève plusieurs questions quant aux capacités des universités du Sud global et pour les chercheurs hors universités, par exemple, de pouvoir assumer la charge financière qui est liée à la publication en accès libre (Greussing et al., 2020; Powell et al., s. d.). D'autre part, bien que plusieurs pensent que les données ouvertes et la science ouverte puissent contribuer à réduire les inégalités dans la production du savoir entre le Nord et le Sud, certains restent sceptiques quant à ce scénario. En fait, les données ouvertes permettent de réduire les coûts d'accès aux données, mais ne réduisent pas les coûts de production pour autant. Par conséquent, un scénario évoqué par Serwadda et al. (2018) est un retour à la recherche parachute¹⁶. En d'autres termes, les chercheurs du Sud resteraient des acteurs de second plan, et qui pour étudier leur propre société, devraient utiliser des données de chercheurs du Nord qui auraient été produits sans la participation des chercheurs locaux. Finalement, il y a aussi des enjeux quant à la confidentialité des répondants et des utilisateurs Chiware & Skelly (2023). Plusieurs individus, comme dans le cas d'entrevues ou d'un sondage, vont accepter de participer à l'enquête parce que leurs réponses seront anonymisées et qu'elles ne seront pas partagées publiquement. Tous ces éléments soulèvent d'importantes réflexions éthiques quant aux bonnes pratiques à développer dans le cadre de la science ouverte. La science ouverte à un avenir très prometteur et peut générer des retombées positives, à conditions quelle s'intéresse aux différents défis auxquels elle est confrontée.

¹⁶La recherche parachute est une « pratique extractive par laquelle des chercheurs - généralement issus de pays dotés de ressources élevées - effectuent des recherches et extraient des données et des échantillons de régions ou de populations non autochtones, généralement des contextes ou des pays à faibles ressources, sans reconnaître de manière appropriée l'importance de l'infrastructure et de l'expertise locales. Ce faisant, les chercheurs étrangers ne parviennent pas à établir des collaborations équitables et à long terme avec des partenaires locaux. » (odenyTimeEndParachute2022?)

1.5 Critères de sélection

Au cours des prochains chapitres, plusieurs outils seront présentés. Étant donné que le but de l'ouvrage n'est pas uniquement d'offrir une perspective sur le monde du numérique, mais surtout d'offrir des conseils pratiques aux lectrices et lecteurs, certains choix ont dû être faits dans la sélection des outils. Bien que ces choix soient arbitraires, ils sont tout de même informés par certains critères. Au moment d'écrire ces lignes, peu de littérature existe sur le sujet. Par conséquent, l'élaboration de ces critères s'est faite de manière inductive, soit à partir de l'expérience des auteurs. Ils sont aussi informés par des considérations pratiques, tels que la popularité de l'utilisation par une communauté et par un champ d'études. Pour être pleinement transparent, la grande majorité des autrices et auteurs de ce livre sont issues de la science politique.

Malgré cela, nous pensons tout de même que ces critères sont pertinents et informatifs pour les autres disciplines des sciences sociales. Par le fait même, nous souhaitons introduire le débat avec les autres champs. Nous sommes convaincus que ce dialogue sera riche et fructueux. Pour l'instant, tenons-nous-en aux six critères ci-dessous. Tous les outils présentés dans ce livre ne respectent pas nécessairement parfaitement ces critères. Certaines considérations pratiques limitent le plein respect de ces critères. Surtout, bien que le logiciel libre ait été mis de l'avant tout au long de ce chapitre, ce ne sont pas tous les outils numériques présentés dans ce livre qui sont des logiciels libres. Certes, la philosophie du logiciel libre a influencé la sélection et l'utilisation de ces outils, avant même la rédaction de ce livre. Cependant, face à des contraintes pratiques, certains logiciels payants sont utilisés et seront présentés. Bien évidemment, les logiciels libres ne sont pas absolus. Ils peuvent cloisonner les chercheurs entre les utilisateurs et les non-utilisateurs de ces logiciels. Un tel scénario est loin d'être souhaitable. En aucun cas ils ne devraient limiter la collaboration scientifique.

Les lectrices et lecteurs sont encouragés à réfléchir à propos de leur propre besoin afin de déterminer quels outils elles et ils devraient utiliser. Par exemple, si dans leur communauté, les gens utilisent Python plutôt que R, alors nous recommandons d'utiliser Python. En d'autres termes, les critères et les outils de ce livre ne visent pas un dogmatisme, et un rejet total de tous les autres outils qui ne sont pas couverts dans ce livre. Il est important d'évaluer ses besoins afin de choisir quels outils devraient être utilisés.

1.5.1 Accessibilité (Gratuit ou peu dispendieux)

L'accessibilité au plus grand nombre de ces outils est importante pour l'atteinte d'une science plus inclusive, et qui respecte les moyens de chacun. L'accès à des outils de qualité ne devrait pas être caché des verrous d'accès payants.

Cette accessibilité est aussi importante pour deux autres raisons. La première, comme le livre vise à donner des connaissances pratiques dans l'utilisation de ces outils numériques, il est important que les lecteurs et lectrices aient facilement accès à ce qui sera présenté. Cela

permettra de reproduire au fur et à mesure de la lecture les différentes étapes et pratiques qui sont présentées. La deuxième, est un corolaire du premier, une fois ces compétences acquises, les lecteurs et lectrices pourront facilement les réutiliser pour réaliser les diverses tâches qu'ils et elles ont besoin d'accomplir.

1.5.2 Existence d'une communauté d'utilisateurs

Ensuite, nous avons considéré l'existence d'une communauté d'utilisateur pour les différents outils qui sont présentés dans ce livre. Ces communautés permettent une grande collaboration entre les différents utilisateurs, et servent souvent de forum d'aide lorsque des problèmes surviennent. Par conséquent, les lecteurs et lectrices auront accès à plusieurs forums d'aide sur internet, au besoin, pour la plupart des outils qui sont présentés dans ce livre. Ainsi, au besoin, ils et elles auront accès à des ressources supplémentaires en ligne lorsqu'une question ou un problème surviendra.

Il se peut que certains logiciels libres n'aient pas un guide d'utilisateur qui soit fourni avec le téléchargement du logiciel. Parfois, cela peut être difficile pour ceux et celles qui souhaitent apprendre à utiliser certains de ces outils de se retrouver et de répondre aux questions qui surviennent. C'est pour ces raisons que nous avons considéré l'existence d'une communauté d'utilisateurs, et l'existence de forum d'aide sur le web. Par exemple, pour toutes les questions liées au code, nous pouvons aller sur le site web de *Stack Overflow*, qui est un important forum d'échange, et sur lequel nous pourrions trouver plusieurs informations et réponses à nos questions.

1.5.3 Popularité dans le champ

Nous avons choisi certains outils sur d'autres notamment à cause de leur popularité dans le champ et en sciences sociales. Par exemple, en sciences sociales, pour les analyses quantitatives, c'est le logiciel R qui est prédominant aujourd'hui. Non seulement dans son utilisation, mais aussi dans les formations offertes, comme dans le cadre de l'école d'été de la *Inter-university Consortium for Political and Social Research (ICPSR)*. Par conséquent, il s'agit d'une considération pratique. Dans le monde du numérique et de la programmation, certains de ces outils deviennent une forme de langage. Ainsi, la maîtrise de ce langage permet de dialoguer avec les autres chercheurs de notre champ, ce qui favorise les débats et la recherche collaborative, par exemple.

1.5.4 Compatibilité avec d'autres outils

Plusieurs des outils que nous présentons sont issus de notre expérience de recherche. Nous utilisons plusieurs d'entre eux notamment parce qu'ils permettent une certaine synergie dans le processus de la recherche. Ainsi, ils s'intègrent bien les uns avec les autres et permettent

une connectivité. Par exemple, Zotero pour la gestion de sa bibliographie et Quarto pour l'écriture de sa recherche se combinent et permettent ainsi de sauver beaucoup de temps dans l'écriture et la gestion des références. Cette intégration des différents outils est importante puisqu'elle favorise non seulement la productivité et une optimisation individuelle, mais aussi collaborative.

1.5.5 Transparence et répliquabilité

D'autres outils qui sont présentés visent à rendre accessibles les résultats de recherche au plus grand nombre. C'est notamment le cas de GitHub, qui sera présenté dans ce livre. Ces plateformes d'entreposage permettent d'héberger des données et des bribes de code, ce qui favorise la reproduction et la transparence des recherches. Ces éléments sont importants pour la science en général, tout en s'inscrivant dans cet objectif de science ouverte.

Cela favorise aussi l'apprentissage des gens qui souhaitent apprendre à réaliser certaines analyses. Il est possible de trouver beaucoup d'extrait de code sur ces plateformes, que nous pouvons tout simplement copier-coller dans nos propres analyses. Ce faisant, nous pouvons apprendre comment réaliser certaines tâches et performer certaines analyses grâce à ces bribes. Il y a donc d'importantes externalités positives qui sont produites en rendant accessible une partie de nos efforts de recherche.

1.5.6 Adaptabilité et flexibilité

Nous avons voulu présenter des outils qui ont la plus grande flexibilité et adaptabilité possible. Nous souhaitons que tous et toutes puissent trouver des outils qui puissent être adaptés à leurs besoins, immédiats comme futurs. Cette adaptabilité est importante surtout lorsqu'on considère l'investissement en temps qui est nécessaire pour la maîtrise de ces outils. Il est donc important que cet apprentissage ne soit pas à recommencer au complet chaque fois que nos besoins changent. Certes, il se peut que certains approfondissements soient à faire dans le temps. Cependant, ceux-ci devraient être minimes lorsque nous avons de bonnes bases. La logique de plusieurs de ces outils reste inchangée, il s'agit bien souvent d'apprendre une nouvelle commande, ce qui n'est pas très coûteux en temps lorsqu'on connaît le fonctionnement de l'outil. De plus, plusieurs de ces outils permettent d'avoir un plus grand contrôle sur la production et le formatage du texte et de graphiques, par exemple. Cette flexibilité est un grand avantage lors de la rédaction et de l'analyse des données.

1.6 Conclusion

En guise de conclusion nous souhaitons mettre l'accent sur un apprentissage important qui se fait de manière implicite dans l'acquisition de ces compétences pratiques: réfléchir de manière scientifique. À la lecture de ce livre, les lecteurs et lectrices auront fait des acquis très

importants, et seront mieux outillés pour réfléchir, organiser et produire leurs recherches, peu importe qu'elle soit orientée vers le milieu académique ou professionnel. Nous pouvons décliner le tout en trois principaux arguments.

Le premier concerne les gains à long terme. Nous avons déjà exposé, en partie, cet argument dans la section des critères. Les outils que nous présentons dans ce livre sont, pour la plupart, très versatiles. Nous pouvons réaliser beaucoup de tâches avec ceux-ci. Bien qu'ils soient, pour certains, coûteux en temps dans leur apprentissage, leurs bénéfices dépassent largement leur coût. Le chapitre 7, à propos des langages de balisage, approfondira ce point davantage, et expliquera pourquoi l'apprentissage d'un langage comme L^AT_EX à ses bénéfices, en comparaison avec Microsoft Word.

Le deuxième concerne la synergie entre tous ces outils. Comme nous l'avons brièvement expliqué dans les critères de sélection, l'apprentissage de ces différents outils favorise le développement d'une synergie entre les différentes étapes et besoin d'une recherche; notamment dans le développement de sa capacité à récolter, à entreposer, à partager, à analyser et à publier ses données ainsi que ses résultats de recherche.

Le troisième concerne le développement de sa « pensée de chercheur ». Comprendre et maîtriser certains de ces outils permet de se doter de capacités réflexives qui pourront être mobilisées dès les premiers moments de la recherche, soit ceux de la conceptualisation. Dès qu'un intérêt de recherche se développe nous pourrions déjà réfléchir à propos de sa faisabilité, et de quelle manière pourrais-je récolter et analyser des données qui me permettront de répondre à ma question. Cela permet aussi de découvrir de nouvelles méthodes d'analyses de données. Simplement par ces étapes préliminaires, nous gagnons beaucoup de temps et d'itérations simplement par la capacité que nous avons à pouvoir penser une recherche qui pourra être réalisée dans la mesure du possible. De plus, ces acquis contribuent au développement de son jugement critique et d'analyse, fort utile lorsqu'on lit des articles et des ouvrages scientifiques. De cette façon, nous sommes en meilleure position afin de mieux comprendre, d'analyser et de critiquer ce que les autres chercheurs ont fait dans le cadre de leurs recherches.

2 Langages de programmation

2.1 R ou ne pas R?

Plusieurs notions liées à l'ère numérique, notamment à ce qui a trait aux opportunités et difficultés que cette dernière peut amener, ont été présentées par l'entremise du chapitre précédent. C'est un monde de possibilité qui s'offre à ceux qui maîtrisent les nouveaux outils des temps modernes. Mais comment en arriver là ? Le présent chapitre a pour but de présenter certains outils flexibles et péreins permettant la réalisation de nombreuses tâches. Une des premières étapes permettant de notamment réaliser la collecte, l'analyse et la visualisation graphique de données ainsi que la rédaction de documents est l'apprentissage d'un langage de programmation. Bien que plusieurs langages de programmation existent, le présent ouvrage priorise le langage **R**. Les sections suivantes présentent ce langage de programmation, ces forces et ces faiblesses ainsi que les raisons de son utilisation. Enfin, la dernière section présente un environnement de programmation qui se prête bien à son utilisation.

2.2 Pourquoi R?

Comme mentionné précédemment, il existe plusieurs langages de programmation. **R** a deux types de compétiteurs : les logiciels à licences comme SAS, STATA et SPSS, et les langages *OpenSource* tels que Python et Julia. **R** est un langage de programmation *OpenSource* développé par des statisticiens, pour des statisticiens, dans les années 1990 (Tippmann, 2015). **R** prend ses racines dans le langage de programmation S, créé notamment par Ross Ihaka et Robert Gentleman. Ces derniers ont fait des choix non orthodoxes lors de l'élaboration du langage, qui font aujourd'hui la popularité de ce logiciel auprès d'un large pan de la communauté académique. En effet, Morandat et al. (2012) rapporte que le langage a été élaboré afin qu'il soit intuitif et qu'il permette aux nouveaux utilisateurs de rapidement réaliser des analyses.

Le langage de programmation **R** a plusieurs avantages qui font de lui un outil puissant et utile pour tout chercheur. L'un de ses grands avantages est qu'il est *OpenSource*. Ayant déjà abordé le sujet dans le chapitre précédent, il sera question ici de simplement rappeler les grandes lignes de l'argument, à savoir que : 1) l'*OpenSource* est gratuit d'utilisation; 2) l'*OpenSource* est développé de façon bottom-up, ce qui lui procure une grande flexibilité; et 3) il permet aux utilisateurs de créer leurs propres fonctions. À l'inverse, les logiciels à licences sont coûteux, rigides et l'ajout de fonctionnalités se fait par les développeurs internes à la compagnie. Ces

formalités rendent le processus plus lent et réduisent l'éventail des possibilités pour la personne chercheuse. Ceci étant dit, certains avanceront que c'est justement ce processus interne lent qui assure la validité et la fiabilité des analyses effectuées par SAS, STATA ou SPSS. Or, dans son livre dédié aux utilisateurs de SPSS et de SAS, Muenchen (2011) soulève le point que bien souvent, ce sont des individus atomisés qui développent les nouvelles fonctionnalités de ces langages et que le processus de révisions se fait ensuite par des comités internes de testeurs. Il en va de même pour le développement des *packages* R dans la mesure où ce dernier se voit testé et amendé par plusieurs programmeurs indépendants dans un processus itératif des plateformes telles que GitHub. De plus, bien des nouvelles techniques statistiques sont développées pour R par des chercheurs qui publient leur travail dans des journaux académiques revus par des pairs, assurant la qualité du procédé. Le fait que SAS et SPSS permettent à leur utilisateur d'intégrer des routines R à leur programme est un indicateur fort ne serait-ce que de l'utilité de R (Muenchen, 2011). Le langage de programmation **R** permet également de réaliser une grande quantité de tâches de recherche. En effet, les personnes programmant en **R** peuvent notamment manipuler et visualiser des données, faire différents types d'analyses, créer des fonctions et faire des boucles en plus de pouvoir combiner **R** avec certains langages de balisages.

D'un autre côté, l'utilisation du langage de programmation **R** peut être perçue comme ayant certains inconvénients. Plusieurs disent que la courbe d'apprentissage peut être plus grande que celle de programmes à licences. La véracité de cet argument est discutable. Les programmes demandant des licences ont également un coût d'entrée. De plus, les nouvelles itérations de ces logiciels amènent des changements demandant une période d'adaptation pour la personne chercheuse. D'autres disent que le développement *OpenSource*, spécifiquement celui du langage de programmation **R**, se fait de façon anarchique. Cela est davantage une question d'opinion et de conception du monde qu'une vérité. Le développement de *package* se fait effectivement de manière décentralisée et toute personne sachant programmer en **R** peut collaborer à cette communauté. Bien qu'il n'y ait pas d'autorité centrale, les *packages* sont regroupés sur le *Comprehensive R Archive Network* (CRAN) (voir le <https://cran.r-project.org/> pour plus d'information). Le site a une politique de dépôt stricte, ainsi les *packages* doivent être suffisamment documentés. Il est également possible d'y télécharger le langage de programmation **R**. Ce langage, ainsi que ces différents *packages*, sont disponible sur Windows, macOS et Linux.

2.3 Où coder en R ?

Un environnement de développement intégré (IDE) permet aux programmeurs de consolider les différents aspects de l'écriture d'un programme informatique. Ils permettent de réaliser toutes les activités courantes d'un programmeur – l'édition du code, la construction des exécutables et le débogage – au même endroit. Les environnements de développement intégrés sont conçus pour maximiser la productivité du programmeur. Ils fournissent de nombreuses fonctionnalités – notamment la coloration syntaxique ainsi que le contrôle de version – pour créer, modifier et compiler du code. Certains environnements de développement intégré sont dédiés à un langage de programmation spécifique. Par conséquent, ils contiennent des fonctionnalités qui sont plus

compatibles avec les paradigmes de programmation du langage auquel ils sont associés. Enfin, il existe de nombreux environnements de développement intégré multilingues.

Comme mentionné précédemment, R est un des langages de statistiques et d'exploration de données les plus populaires en sciences sociales. R est pris en charge par de nombreux environnements de programmation. Plusieurs ont été spécialement conçus pour la programmation en R – le plus notable étant RStudio – tandis que d'autres sont des environnements de programmation universels – tels que Visual Studio Code – et prennent en charge R via des plugins. Il est également possible de coder en R à partir d'une interface en ligne de commande. Une telle méthode permet la communication entre l'utilisateur et son ordinateur. Cette communication s'effectue en mode texte : l'utilisateur tape une « ligne de commande » – c'est-à-dire du texte dans le *terminal* – pour demander à son ordinateur d'effectuer une opération précise, telle que rouler un fichier de code R.

La suite du chapitre présente RStudio, notamment à travers ses avantages et inconvénients, mais également des exemples de ses fonctionnalités.

2.4 Qu'est-ce que RStudio ?

RStudio est un projet open source destiné à combiner les différentes composantes du langage de programmation R en un seul outil (Allaire, 2011). RStudio fonctionne sur tous les systèmes d'exploitation, y compris Windows, Mac OS et Linux. En plus de l'application de bureau, RStudio peut être déployé en tant que serveur pour permettre l'accès Web aux sessions R s'exécutant sur des systèmes distants (Allaire, 2011). RStudio facilite l'utilisation du langage de programmation R en offrant de nombreux outils permettant à son utilisateur d'aisément réaliser ses tâches. Parmi les plus utiles, on retrouve notamment une fenêtre d'aide, de la documentation sur les différents packages R, un navigateur d'espace de travail, une visionneuse de données et une prise en charge de la coloration syntaxique (Horton, Kleinman, 2015). De plus, RStudio permet de coder dans plusieurs langages et de supporter une grande quantité de formats. Il fournit également un support pour plusieurs projets ainsi qu'une interface pour utiliser des systèmes de contrôle, tels que GitHub (Horton, Kleinman, 2015).

RStudio a plusieurs avantages. Son utilisation est facile à apprendre pour les débutants. Les principaux éléments d'un IDE sont intégrés dans une disposition à quatre volets (Verzani, 2011). Cette disposition comprend une console, un éditeur de code source à onglets pour organiser les fichiers d'un projet, un espace pour l'environnement de travail et un quatrième volet où il est notamment possible d'afficher des graphiques ou de la documentation sur différents packages. Ce volet permet d'ailleurs d'accéder au répertoire des *packages* disponibles pour R en plus de permettre à l'utilisateur de consulter l'arborescence de ses fichiers. De plus, on y retrouve la possibilité de créer plusieurs espaces de travail – appelés projets – qui facilitent l'organisation de différents *workflows*.

Il y a plusieurs autres aspects de RStudio que les programmeurs apprécient. Parmi ceux-ci se trouve le fait qu'il peut être utilisé via un navigateur Web pour un accès à distance (Verzani, 2011). De plus, RStudio supporte plusieurs langages de programmation ainsi que différents langages de balisage. Qui plus est, de nouvelles fonctionnalités sont régulièrement ajoutées pour satisfaire les besoins de la communauté scientifique. Enfin, R logiciel est également souvent mis à jour.

Parmi ce que certains considèrent comme étant les points faibles de RStudio, on retrouve des éléments liés à la configuration. Certains utilisateurs trouvent que le nombre de raccourcis est limité. D'autres trouvent que le *set up* des différents panneaux n'est pas ergonomique, ou même qu'il n'est pas possible de pouvoir suffisamment personnaliser l'environnement de programmation. De plus, certains utilisateurs ont rapporté que RStudio était plus lent que d'autres alternatives pour quelques opérations, surtout celles comprenant de longs codes.

2.5 Comment utiliser RStudio ?

Bien que de nombreux éléments puissent être personnalisés, la disposition par défaut de RStudio est composée de quatre volets principaux (Verzani, 2011). Dans le coin supérieur gauche se trouve le cadran principal. C'est dans celui-ci que l'utilisateur passera la plus grande partie de son temps. On y modifie des fichiers de différents formats et il est possible d'y afficher des bases de données. Dans le coin inférieur gauche se trouve la console ainsi que le terminal. Dans cette première, on peut interagir avec R de la même manière que dans le cadran principal, mais le code ne sera pas enregistré. Le terminal, pour sa part, est le point d'accès de communication entre un usager et son ordinateur. Bien que les différents systèmes d'exploitation viennent avec un terminal déjà intégré, il est aussi possible d'y accéder à partir de RStudio.

On retrouve, dans le coin supérieur droit, l'espace de travail. Ce cadran contient trois éléments : *l'environnement global*, *l'historique* et *les connections*. *L'environnement global* est l'endroit où l'utilisateur peut voir les bases de données, les fonctions et les différents autres objets R qui sont actifs. Il peut cliquer sur les divers éléments actifs pour les consulter. L'onglet *historique* permet à l'utilisateur de consulter les derniers morceaux de code R qu'il a roulé ainsi que les dernières commandes écrites dans la console. L'onglet *connections*, pour sa part, permet de connecter son IDE à une variété de sources de données et d'explorer les objets et les données qui la composent. Il est conçu pour fonctionner avec une variété d'autres outils pour travailler avec des bases de données en R dans RStudio.

Le cadran dans le coin inférieur droit, pour sa part, contient plusieurs outils très utiles pour les usagers de RStudio. L'onglet *Files* permet à l'utilisateur de naviguer dans les fichiers que contient son ordinateur sans avoir à sortir de RStudio. L'onglet *Plots* permet de visualiser les graphiques générés à partir de R, que ce soit en utilisant *ggplot2*, *lattice* ou *base R*. L'onglet *Packages* permet de consulter les packages installés précédemment par l'utilisateur en plus de pouvoir en consulter la documentation. C'est aussi un des différents endroits à partir d'où il est possible d'installer des packages avec RStudio. L'onglet *Help* permet à l'utilisateur de chercher

et de consulter de la documentation sur de nombreux sujets, notamment sur les différentes fonctions en R ainsi que sur les packages. Pour sa part, l'onglet *Viewer* permet la visualisation de contenu web local.

Enfin, l'utilisateur peut modifier les dimensions par défaut pour chacun des quatre cadrans principaux. En cliquant sur la division des sections, il est possible d'ajuster l'allocation horizontale de l'espace. De plus, chaque côté dispose d'un autre séparateur pour ajuster l'espace vertical. Qui plus est, la barre de titre de chaque cadran comporte des icônes pour ombrer un composant, maximiser un cadran verticalement ou modifier la taille de l'espace de travail (Verzani, 2011; Nierhoff et Hillebrand, 2015).

2.6 Conclusion

Le langage de programmation R est un outil très utile pour toutes sortes de tâches notamment reliées aux statistiques et à la visualisation graphiques. Sa maîtrise est requise pour accéder à plusieurs emplois, autant dans le monde académique que dans les secteurs publics et privés. Avec un peu de chance, le présent chapitre vous a éclairé sur son utilité et sa pertinence dans le monde du travail contemporain. Bien que le langage de programmation R ne doivent pas obligatoirement être utilisé avec RStudio, nous pensons que pour la plupart des usagers, leur utilisation conjointe est bénéfique et souhaitée. RStudio permet également d'utiliser différents langages de balisage compatibles avec R, facilitant l'utilisation de plusieurs outils complémentaires. L'apprentissage du langage de programmation R apparaît également être une valeur sûre. Sa longévité dans plusieurs sphères ainsi que la forte croissance de sa base d'utilisateurs laisse présager que d'en connaître au moins les bases est un énorme avantage pour tout le monde. Pour ceux qui sont particulièrement intéressés par le langage de programmation R et qui désirent s'impliquer dans sa communauté, il existe plusieurs conférences internationales et nationales sur R – notamment *RConference* and *useR!* – et un journal académique, *The R Journal*. On retrouve également différentes communautés telle que *R-Ladies* qui met de l'avant la diversité des genres dans la communauté du langage de programmation R. Le langage de programmation R est plus qu'un simple outil statistique, il est le centre d'une grande communauté de gens qui ont à coeur des principes liés à l'inclusion et à l'avancement humain.

3 Outils de gestion de projet

3.1 À la quête de l'optimisation

Le monde de la recherche en sciences sociales numériques est en constante évolution, offrant de nouvelles opportunités mais aussi des défis uniques. Dans cette quête incessante pour optimiser notre efficacité et notre collaboration, l'utilisation des bons outils devient la clé de la réussite. Que vous soyez un chercheur en herbe ou un professionnel chevronné, la manière dont vous organisez vos méthodes de travail et gérez vos ressources peut déterminer la qualité et l'impact de vos résultats.

3.2 L'importance d'une méthode de travail efficace

Avant même de plonger dans les détails des méthodes de recherche et des analyses, il est crucial de poser les bases d'une méthode de travail efficace. Qu'il s'agisse de travailler en solitaire ou en équipe, l'ordre et la structure sont des éléments essentiels. Des dossiers bien organisés, une arborescence claire et un entreposage sécurisé deviennent les piliers sur lesquels repose votre productivité. Après tout, un environnement de travail organisé engendre des résultats ordonnés.

Ce chapitre vous emmènera à découvrir une gamme d'outils conçus pour répondre aux besoins spécifiques des chercheurs en sciences sociales numériques. Dans une quête pour maximiser votre temps, améliorer vos flux de travail et renforcer vos collaborations, nous explorerons trois types d'outils qui vous guideront dans cette quête d'optimisation :

1. **Logiciels de communication** : La communication transparente est le cœur d'une collaboration réussie. Nous explorerons des outils tels que Slack qui facilitent les échanges en temps réel, connectant les chercheurs, même à distance, pour un partage rapide d'idées et d'informations.
2. **Logiciels de gestion de versions décentralisé** : Nous plongerons dans le monde de Git et GitHub, des outils indispensables pour le suivi des versions et la collaboration efficace sur le code source.

3. **Outils d'entreposage de données** : Que vous traitiez des données sensibles ou non, la conservation sécurisée de vos informations est primordiale. Des plateformes telles que Dropbox et Amazon Web Services (AWS) offrent des espaces sécurisés pour entreposer et partager vos données avec votre équipe.

Chacun de ces outils est une pièce du puzzle, conçue pour vous aider à gagner du temps, à collaborer de manière plus fluide et à renforcer la qualité de votre recherche en sciences sociales numériques. Plongeons dans ces outils avec un désir commun d'optimisation et d'excellence dans notre travail.

3.3 Gestion Individuelle

3.3.1 Gestion de tâches

Structurer ses tâches est un processus fondamental pour mener un projet à terme. Particulièrement dans le monde académique, où les travaux s'échelonnent souvent sur plusieurs années, il est facile de perdre de vue ses objectifs ou de prendre des détours coûteux en temps si le chemin vers le produit final est mal défini. Gérer et structurer ses tâches de manière efficace facilite la mesure des progrès et permet de constamment vérifier si ceux-ci sont encore alignés avec les objectifs finaux.

3.3.1.1 Comment

Gérer ses tâches de façon efficace passe par une structuration claire des objectifs du projet. Il est important de connaître la destination finale afin de choisir la meilleure direction pour y parvenir. Pour ce faire, il est utile de schématiser ou de lister la conception de la version finale du projet. Dans l'idéal, à quoi ressemble-t-il dans sa forme aboutie? Une fois cette vision clairement définie, il est possible de désagréger le projet en grandes étapes. Que faut-il accomplir, à l'échelle macro, pour atteindre les objectifs fixés?

À cette étape, il est crucial de prendre en compte les ressources financières, temporelles et humaines disponibles. Cela permet de déterminer de manière réaliste ce qui est possible. Identifier ces grandes étapes contribue à la création d'un plan de projet structuré où chaque phase est clairement définie. Cela aide à anticiper les besoins en ressources et à ajuster les échéances en conséquence.

La révision continue est également un élément clé du processus de gestion des tâches. En réévaluant régulièrement l'état d'avancement du projet par rapport au plan initial, il est possible d'apporter des ajustements nécessaires pour rester sur la bonne voie. Cet astuce permet de répondre aux changements inévitables qui surviennent au cours de la recherche, qu'ils soient dus à des découvertes inattendues, des changements dans les directives institutionnelles ou des feedbacks des pairs.

3.3.1.2 Quand

Avec des objectifs bien définis et des étapes claires pour y parvenir, la structure du projet est complète. Il est donc temps de se lancer dans la gestion des tâches. En fonction des objectifs établis, certaines tâches sont plus importantes que d'autres. En effet, un projet est vraisemblablement composé de tâches qui doivent être réalisées avant que d'autres soient amorcées. Le défi est de déterminer efficacement ce qui doit être priorisé. L'agilité est un processus de travail qui facilite cette priorisation. En agilité, des objectifs sont fixés dans le temps, et sont évalués de manière constante. Les tâches sont déterminées en fonction de l'avancement et des blocants des objectifs.

Avec des objectifs bien définis et des étapes claires pour y parvenir, la structure du projet est complète. Il est donc temps de se lancer dans la gestion des tâches. En fonction des objectifs établis, certaines tâches sont plus importantes que d'autres. Un projet est généralement composé de tâches qui doivent être réalisées dans un ordre spécifique, où certaines doivent impérativement précéder d'autres. Le défi est de déterminer efficacement ce qui doit être priorisé pour maintenir une progression fluide et efficace.

L'agilité est un processus de travail qui facilite cette priorisation. En adoptant une approche agile, les objectifs sont fixés dans le temps et sont constamment évalués. Cela permet une adaptation rapide et une réponse aux changements sans compromettre les résultats finaux. De cette façon, les tâches sont déterminées et ajustées en fonction de l'avancement du projet et des éventuels obstacles rencontrés. Le projet avance de façon incrémentale.

Pour une mise en œuvre efficace de l'agilité, il est utile de planifier ses objectifs sur une période de quelques semaines, connues sous le nom de sprints en méthode Scrum, où on évalue le travail accompli et on redéfinit les priorités pour la prochaine période. Ces sprints permettent de s'assurer de rester concentré sur les tâches qui apportent le plus de valeur au projet et d'ajuster les plans en temps réel en fonction des résultats obtenus.

3.3.1.3 Où

Toutes ces pratiques deviennent rapidement complexes si elles ne sont pas encadrées dans un environnement qui permet d'en faire le suivi. Il peut être judicieux de faire appel à des outils de gestion de projet qui supportent l'agilité, tels que Notion ou Mondays. Ces outils permettent de visualiser les tâches à faire sous forme de tableaux de bords interactifs, dans lesquels il est possible de les déplacer en fonction de leur statut d'avancement. Ces outils permettent de structurer les tâches d'un projet et d'en faire le suivi facilement du début à la fin.

Il est également judicieux de faire appel à des outils de gestion de projet qui supportent l'agilité, tels q. Ces outils permettent de visualiser les tâches sous forme de tableaux de bord interactifs où les tâches peuvent être déplacées, modifiées ou mises à jour en temps réel. Ils favorisent la transparence et la communication entre les membres de l'équipe, essentielles pour une gestion agile des tâches.

Enfin, il est crucial d'intégrer des pratiques de réflexion et d'amélioration continue. Après chaque sprint, l'équipe devrait se réunir pour une rétrospective afin de discuter de ce qui a bien fonctionné et de ce qui pourrait être amélioré. Cette culture de l'amélioration continue est au cœur de l'agilité et contribue à l'efficacité et à la réussite du projet à long terme.

Pour déterminer quelles tâches accomplir et dans quel ordre, voici un court processus par étapes :

1. Élaborer les tâches en fonction des objectifs de sprint.
2. Déterminer la linéarité des tâches, c'est-à-dire, quelle tâche doit être accomplie afin d'en débuter une autre.
3. Quantifier le poids de chaque tâche. Certaines tâches sont plus longues que d'autres. Adopter un système qui vous permet d'identifier quelles tâches prendront quelques minutes seulement (comme l'envoi du courriel), et quelles tâches prennent plusieurs jours. Si une tâche est trop longue, c'est un signe qu'elle pourrait être désagrégée en plusieurs tâches plus petites. Cela facilite également le suivi.
4. Donner une échéance réaliste à chaque tâche, en fonction des étapes précédentes. Idéalement, toutes les tâches ne sont pas dues pour la même date, pour éviter un goulot d'étranglement. Les échéances aident à prioriser les tâches.
5. Prioriser les tâches qui ont l'échéance la plus serrée. Si certaines tâches accumulent un retard, c'est peut-être parce que vous devez réévaluer les échéances, les objectifs, ou encore parce qu'il y a des blocants dans vos méthodes de travail. Faire un tel suivi permet d'évaluer sa propre efficacité dans ses méthodes de travail.

L'utilisation d'outils numériques pour la gestion des tâches ne signifie pas qu'il faut abandonner l'agenda papier ou le cahier de notes. Plusieurs trouvent essentiels de prendre des notes et de se faire des listes de tâches à la main. Il est tout à fait possible de combiner les méthodes. À chaque début de semaine, mettez à jour votre gestionnaire de tâches, puis faites votre liste de tâches à la main en conséquence, et planifiez votre semaine. De cette façon, vous savez chaque jour le travail à prioriser.

3.3.2 Enregistrement de protocole

Après avoir établi l'importance de la gestion des tâches et comment une approche agile peut optimiser ce processus, il est complémentaire de d'aborder l'enregistrement méthodique de ces tâches et des étapes du projet. Cette documentation assure la transparence, la répliquabilité et la rigueur scientifique de la recherche. L'enregistrement du protocole de recherche sert plusieurs objectifs clés qui se connectent directement à la gestion agile des tâches. Il agit comme une archive vivante des décisions prises, des méthodes utilisées et des modifications apportées tout au long du projet. Cela permet la vérification et la validation des résultats, et de maintenir une vision claire de l'évolution du projet.

L'enregistrement de protocole en science est une pratique facultative, mais de plus en plus populaire, qui consiste à documenter et à déposer de manière détaillée le plan de recherche d'une

étude avant que celle-ci ne soit menée. Cette démarche s'inscrit dans le cadre des pratiques de recherche ouverte et transparente. Elle a plusieurs avantages : D'abord, rendre les protocoles de recherches publics démontre un engagement envers des méthodes rigoureuses. Cela augmente la confiance envers les résultats obtenus et les méthodes employées. Une démarche détaillée permet aussi la répliquabilité du projet, en offrant aux autres chercheurs du domaine les étapes détaillées employées pour se rendre aux résultats.

Un autre avantage est d'éviter qu'une étude soit réalisée par deux chercheurs au même moment. En enregistrant sa recherche, tous peuvent consulter les recherches en cours, et ainsi s'assurer que leurs projets sont uniques. De cette manière, les ressources académiques sont maximisées. L'enregistrement du protocole permet aussi d'évaluer la recherche par les pairs avant d'amorcer sa réalisation, ce qui peut augmenter la crédibilité de l'étude, faciliter la publication dans une revue scientifique, et gagner du temps dans la réalisation du projet.

Enfin, au coeur du concept de l'enregistrement de protocole se trouve l'idée de l'intégrité de la recherche. La science, par définition, est transparente dans ses démarches. Rendre public ses intentions et devoir justifier chaque modification s'inscrit dans cette optique d'intégrité. Un tel processus rend difficile la chasse au résultats, un fléau en science où les chercheurs privilégient l'atteinte de résultats avant la démarche. Le système très compétitif et chronophage de la publication scientifique encourage ce genre de pratique. L'enregistrement de protocole tente d'encourager des pratiques transparentes, qui sont bien accueillies par les revues scientifiques.

Pour enregistrer votre protocole de recherche, il faut suivre ces quatre étapes :

1. Préparer le protocole. Un document détaille les hypothèses, les méthodes et les analyses prévues. l'objectif est de rédiger un document suffisamment détaillé pour permettre à d'autres chercheurs de reproduire l'étude. La rédaction de ce document n'est pas une perte de temps, car une majorité devrait pouvoir être réutilisée dans l'étude finale.
2. Enregistrer le protocole. L'enregistrement se fait dans un registre public. Il existe différents registres, ouverts à tous les domaines (Open Science Framework, Research Registry) ou plus spécifiques aux sciences sociales (EGAP, AEA RCT).
3. Valider le protocole. Le protocole est évalué par les pairs, pour assurer sa complétude, puis il devient public. Ainsi, vous obtenez des commentaires avant même de soumettre à une revue, ce qui peut vous faire sauver du temps en apportant des modifications avant la réalisation, plutôt qu'après.
4. Suivi du protocole. À chaque étape de la recherche, les chercheurs confirment qu'ils suivent le processus annoncé, ou justifient les changements apportés, ce qui assure la transparence dans leurs démarches scientifiques.

3.3.3 Code en open access (Repository github)

L'enregistrement de protocole est seulement un volet de la transparence en science. De même que pour les protocoles, la mise à disposition du code source employé pour les analyses permet non seulement de vérifier les résultats publiés mais aussi de renforcer la confiance dans les

conclusions de la recherche. Le partage du code, structuré et commenté, via une plateforme comme GitHub, s'inscrit dans cette démarche de recherche ouverte.

Les langages de programmation sont très fluides et décentralisés, et les codes employés pour arriver à un même résultat peuvent varier d'une personne à l'autre. Rendre son code public permet la reproductibilité et la transparence dans la manipulation et l'analyse de ses données. Plusieurs revues exigent de rendre le code en libre accès lors d'une soumission.

1. Préparation du code
 - a. ReadMe
 - b. Structuration du code + commentaires
2. Dépôt du code sur github
 - a. Création du repo
 - b. Gestion des version (git)

3.3.4 Stockage des données (Dropbox, autres clouds)

L'entreposage des données occupe une place cruciale dans la recherche en sciences sociales numériques. La manière dont vous entreposez et gérez vos données peut avoir un impact significatif sur la sécurité, la confidentialité et la reproductibilité de votre travail. Dans cette section, nous allons aborder différents aspects de l'entreposage de données, des outils disponibles et de l'importance d'une gestion efficace de vos fichiers.

3.3.4.1 Entreposage de données non sensibles

Au fil du temps, de nombreux outils d'entreposage ont émergé pour répondre aux besoins variés des chercheurs en sciences sociales. Des solutions populaires incluent Dropbox, Google Drive, OneDrive et Amazon S3 d'AWS. L'histoire de ces outils témoigne de l'évolution des besoins d'entreposage et de collaboration.

Lorsqu'il s'agit d'entreposer vos données de recherche, la règle d'or est de ne jamais perdre d'informations précieuses. Cette préoccupation prend toute son importance lorsqu'un chercheur en sciences sociales, seul ou en équipe restreinte, se lance dans un projet. Pour répondre à ce besoin, les services d'entreposage cloud tels que Dropbox, Google Drive et OneDrive se révèlent indispensables. Voici quelques avantages d'un entreposage sur le cloud pour la recherche :

1. *Sauvegarde automatique* : Les solutions cloud sauvegardent automatiquement vos fichiers, garantissant que vous ne perdrez jamais vos données en cas de panne d'ordinateur ou d'accident.

2. *Accessibilité universelle* : Vous pouvez accéder à vos fichiers à partir de n'importe quel appareil avec une connexion Internet, ce qui favorise la flexibilité dans la gestion de vos projets.
3. *Partage facilité* : Les services cloud permettent de partager facilement des fichiers et des dossiers avec des collègues, même en dehors de votre équipe de recherche. Cela favorise la collaboration et la communication.

Il est important de noter que le choix d'un service cloud dépend de vos besoins et de vos préférences. Considérez des facteurs tels que la capacité d'entreposage, les fonctionnalités de partage, la convivialité et la compatibilité avec vos outils de recherche existants.

Dropbox est connu pour sa simplicité d'utilisation et sa convivialité. Il peut être un choix approprié pour entreposer des fichiers non sensibles, partager des documents avec des collègues et faciliter la collaboration.

Pour utiliser Dropbox efficacement, organisez vos fichiers en arborescence logique. Créez des dossiers spécifiques pour chaque projet et partagez-les avec les membres de votre équipe. Pour éviter de pousser des fichiers sensibles sur GitHub, ajoutez le nom de dossier à exclure dans un fichier *.gitignore*.

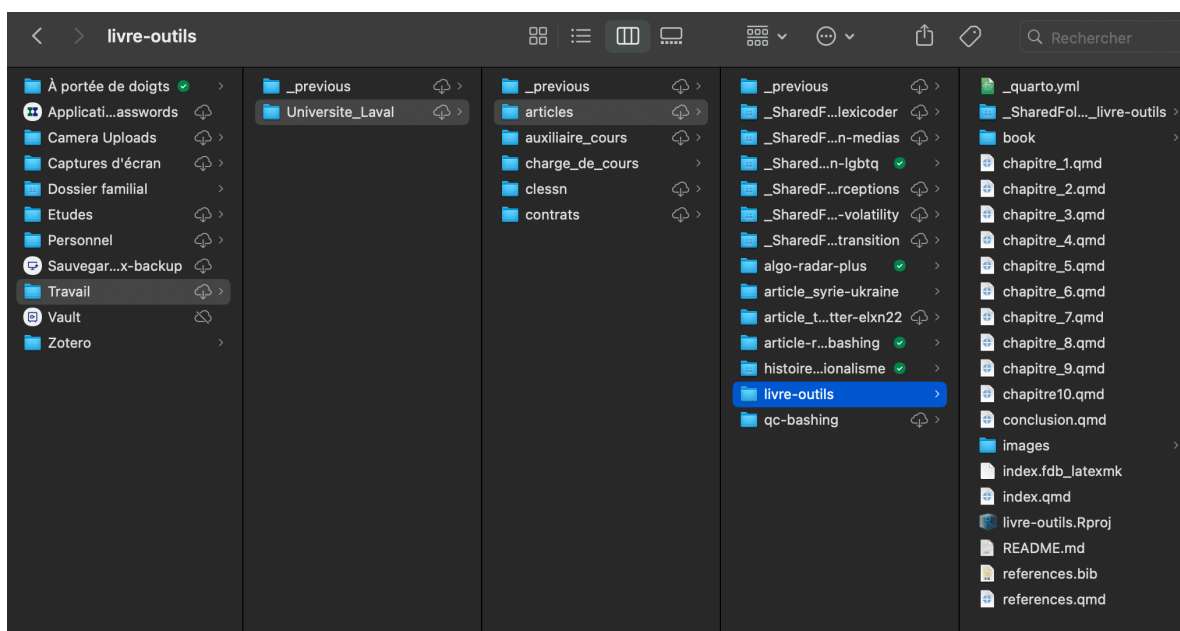


Figure 3.1: image1

Dropbox offre un suivi automatique des modifications, ce qui vous permet de remonter dans le temps pour restaurer des versions antérieures de vos fichiers. Cela garantit l'intégrité de vos données et vous permet de revenir à des versions précédentes si nécessaire. De plus, l'archivage

de dossiers et de projets complets peut aider à conserver une vue chronologique de votre travail au fil du temps.

Il est également crucial de considérer la taille de vos données. Si vous traitez des fichiers volumineux tels que des images, des vidéos ou des ensembles de données massifs, il peut être judicieux d'utiliser un service cloud pour entreposer ces fichiers et les partager avec vos collaborateurs, plutôt que de les pousser sur des plateformes de gestion de versions comme GitHub.

Pour les données sensibles, les services cloud tels que Dropbox et Google Drive peuvent ne pas être suffisamment sécurisés. C'est là que des solutions comme AWS entrent en jeu. Cependant, il est important de noter que l'utilisation d'AWS peut s'avérer complexe, en particulier pour un jeune chercheur travaillant en solo ou en petite équipe.

3.4 Gestion en équipe

3.4.1 Coordination d'équipe (Slack Mattermost)

Logiciel de gestion de communication (Slack)

Dans tout bon projet de recherche, la communication est primordiale. Que ce soit pour décrire les avancements, discuter des étapes à venir, entretenir un partenariat avec des partenaires ou simplement structurer ses pensées, la plateforme par laquelle vous communiquez vous accompagne à chacune des étapes du travail. Il est donc important de choisir un outil qui convient bien à vos projets et de prendre le temps de l'apprivoiser et d'optimiser son utilisation.

Il y a tellement de plateformes différentes pour communiquer qu'il faut être prudent par rapport au nombre utilisé. Si vous ne faites pas un choix, vous pouvez, sans vous en rendre compte, mêler Teams, courriels, Zoom et autres. Rapidement, vous perdez le contrôle de ce qui est dit. Nous vous proposons d'opter pour un logiciel de gestion de communication. Il existe plusieurs logiciels du genre, tels que Microsoft Teams, Slack, Google Workspace et Workplace. Toutes ces options peuvent vous permettre de collaborer efficacement en équipe. Dans le cadre de nos travaux, nous utilisons Slack. C'est donc principalement de cet outil que nous parlerons dans cette section, mais n'hésitez pas à vérifier quelle plateforme correspond le mieux à vos besoins.

Peu importe votre niveau d'implication, la collaboration et la communication sont inévitables en recherche. La science n'est pas une discipline qui se développe en solitaire, elle nécessite des échanges et des débats. Les équipes de recherche sont souvent dispersées géographiquement. Même si vous travaillez actuellement seulement avec votre directeur, il est certain que plusieurs équipes de recherche dans votre département utilisent un tel logiciel. Un courriel peut faire l'affaire pour une discussion ponctuelle qui se règle rapidement. Cependant, dans une équipe de travail dynamique, où plusieurs membres participent à divers projets, les courriels deviennent

rapidement chaotiques, il est difficile de retracer ce qui a été dit et de conserver les pièces jointes. Les discussions deviennent rapidement trop complexes pour le médium utilisé.

Les logiciels de gestion de communication ont été conçus spécifiquement pour répondre aux besoins des équipes collaboratives. Vous y trouverez leur facette la plus attrayante : une structure simple et adaptée. Les chaînes et fils de discussions permettent de garder des traces et de se retrouver facilement dans ce qui a été dit. Une autre force de ces logiciels est la centralisation des outils de travail. Sur Slack, comme sur Teams, vous pouvez faire des appels en visioconférence à l'endroit où vos conversations écrites se trouvent. Il est aussi possible d'y télécharger l'application mobile, ce qui facilite l'accessibilité et la connexion des membres de l'équipe. Tout avoir structuré à son goût au même endroit et à portée de main, cela permet de structurer sa pensée plus efficacement, d'éviter les oublis et de réduire le stress.

3.4.1.1 Comment utiliser votre logiciel efficacement

Une fois que vous êtes convaincu d'aller de l'avant avec un de ces outils, vous devrez apprendre à bien vous en servir. Voici quelques trucs qui pourront vous aider à optimiser son utilisation. Les points ci-dessous font référence à Slack, mais peuvent très bien être adaptés à d'autres plateformes.

3.4.1.2 Structuration

Il est important de bien réfléchir à la structuration de vos chaînes. Si vous ne faites pas ce travail, les chaînes peuvent se multiplier rapidement et les conversations se mettent alors à s'entrecroiser, vous faisant ainsi perdre le fil. L'objectif de ces outils étant d'éviter ces problèmes, vous ne voulez pas perdre l'avantage comparatif que vous venez tout juste de gagner face aux courriels ! La structuration des chaînes devrait être similaire à celle de votre équipe de recherche. Si vous utilisez Notion ou un autre logiciel du genre, la structure des deux outils devrait être la même. Nous vous proposons d'avoir une chaîne pour chacun des projets. Si le projet est trop gros et que la conversation devient chaotique, pensez à créer une sous-chaîne (un sous-projet) qui vous permettra d'aborder un sujet précis, sans mêler les discussions. Pour faciliter la structuration des chaînes, vous pouvez utiliser des préfixes, pour classer les chaînes par thème, ou autre typologie qui vous convient. Également, utilisez les Espaces d'équipe. Chaque équipe devrait avoir son propre espace, avec ses propres chaînes. Vous pouvez faire partie de plusieurs équipes et naviguer à travers les espaces. Si plusieurs équipes partagent un même espace de travail, vous pourriez perdre le contrôle de sa structure.

3.4.1.3 Maintenance

Slack est un espace dynamique, tout comme votre équipe ! La structure que vous avez choisie n'est pas permanente. Vous devriez rapidement vous questionner à savoir si elle convient

toujours à vos activités. Votre espace d'équipe est comme votre réel lieu de travail, faites-y régulièrement le ménage pour vous assurer que tout est propre et en ordre. Archivez les chaînes qui ne sont plus pertinentes ou actives, puisque vous pourrez toujours les désarchiver quand cela sera nécessaire. Épinglez des messages importants et des documents utiles aux projets dans les chaînes appropriées. Faites le tour de ce qui est épinglé à l'occasion pour vérifier si c'est encore pertinent. Cela peut paraître énergivore, mais l'efficacité de votre travail d'équipe va en bénéficier. Également, rappelez aux membres de votre équipe d'utiliser les bonnes chaînes pour chacune des discussions. Il ne faut pas que les conversations se croisent à travers les chaînes. Chaque chaîne a son utilité et doit être utilisée en conséquence. Les appels d'équipe doivent aussi se faire dans les bonnes chaînes. Quand vous êtes en appel, utilisez le fil de discussion pour conserver des traces écrites des points abordés dans la réunion. Les fils de discussions sont en général un bon outil pour ne pas se perdre dans une discussion. Si l'usage des mauvaises chaînes est un problème récurrent, il est possible que la structure que vous employez est mal adaptée à vos travaux. Vous pouvez alors retourner à la planche à dessin. Assurez-vous que toute l'équipe comprenne bien comment utiliser Slack. Si ce n'est pas le cas, formez-les. Une structure adaptée et une équipe bien formée peuvent faire des miracles.

3.4.1.4 Collaboration

La grande majorité des conversations devraient se faire dans les chaînes. Les conversations privées ont leur utilité, vous vous en servirez. Il est parfois nécessaire d'avoir des discussions plus confidentielles et de parler rapidement à quelqu'un sur un sujet éphémère. Toutefois, par soucis de transparence et d'inclusion, toute discussion à propos d'un projet devrait se faire dans sa chaîne. Si vous jugez qu'un membre d'une chaîne ne devrait pas lire ce que vous avez à dire sur le projet, c'est qu'il ne devrait pas faire partie de la chaîne. Par rapport aux membres, trouvez le bon équilibre par rapport à qui devrait être dans quelle chaîne. L'objectif n'est pas d'exclure et de cacher du contenu, vous voulez une équipe transparente. Vous voulez que vos membres restent bien informés de l'avancement des projets sans les submerger d'information qui ne leur est pas utile. C'est à vous de trouver la formule gagnante. Invitez vos partenaires externes dans votre espace d'équipe. Créez des chaînes spécifiques aux partenaires pour que les conversations externes soient tout aussi organisées. N'invitez pas vos partenaires dans vos chaînes privées, question de confidentialité. Si un partenaire n'a pas l'habitude d'utiliser Slack ou l'outil que vous utilisez, proposez-lui de vous y joindre quand même. Moins vous utilisez les outils des autres, plus vous gardez centralisées vos communications et évitez de jongler avec plusieurs plateformes.

3.4.1.5 Optimisation personnelle

Une fois que la structure d'équipe est définie et que vos membres et vos partenaires sont à l'aise avec l'utilisation de la plateforme, il est temps d'organiser la structure de votre Slack personnel. Créez des sections pour trier les chaînes. La structure d'équipe est essentielle, mais

une fois qu'elle est déterminée, chaque membre n'utilise pas forcément les chaînes de la même façon. Vous pouvez vous créer une section de favoris, ou encore différentes sections par rapport aux différents thèmes pour y faciliter la navigation. Également, ajustez vos paramètres de notifications. C'est à vous de déterminer quelle chaînes méritent de produire des alertes, et à quels moments vous souhaitez les recevoir. Slack a plusieurs applications intégrées qui facilitent la compatibilité avec vos autres outils. Vous pouvez connecter votre calendrier, votre Notion et votre GitHub pour recevoir des alertes pertinentes. Allez explorer ces applications pour déterminer lesquelles vous conviennent.

Tel que mentionné précédemment, plusieurs logiciels peuvent convenir à vos besoins. Puisque nous utilisons Slack, voici quelques raisons qui pourraient vous convaincre d'opter pour cette option ou de vous en éloigner. Sachez que cette liste n'est pas du tout exhaustive, mais reflète simplement quelques-unes de nos observations par rapport à notre outil de travail.

- **Avantages**

L'utilisation de Slack est très intuitive. Nous l'utilisons régulièrement dans des cours, et les étudiants apprennent rapidement à l'utiliser. La distinction entre les chaînes publiques accessibles à tous les membres d'un espace d'équipe et les chaînes privées est claire et simple d'utilisation. Slack offre aussi une fonction de recherche, qui vous permet de retrouver des messages à travers les chaînes. L'intégration de applications qui font le pont avec d'autres outils est fort appréciée. Enfin, Slack est utilisé partout dans le monde par des équipes de toutes les tailles et dans tous les domaines. C'est un outil très présent en recherche académique qui facilite la collaboration et la multidisciplinarité. Les chances sont élevées que vos partenaires utilisent déjà l'outil, ou au minimum en aient déjà entendu parlé.

- **Inconvénients**

Si vous avez l'habitude d'utiliser les outils d'une suite, comme celles de Microsoft ou de Google, il est possible que vous trouviez l'intégration de ces outils à Slack moins pratique que si vous utilisiez les plateformes proposées par ces compagnies. Également, gardez en tête que la version gratuite de Slack a plusieurs limitations. Elle implique notamment un limite de temps par rapport à l'archivage des messages, que vous ne pourrez pas retracer après 90 jours. Les coûts pour utiliser Slack à son plein potentiel peuvent être élevés, mais puisque ce genre d'outils est de plus en plus répandu, il est fort possible que son utilisation soit financée par votre département.

3.4.2 Gestion de tâches en équipe (Notion, Monday)

3.4.3 Gestion de versions en équipe (Pull-push, pull-requests)

Lorsque l'on aborde le domaine de la recherche scientifique en sciences sociales numériques, la collaboration et la gestion efficace du code deviennent des éléments cruciaux pour progresser

dans ses projets. Dans cette optique, les outils de gestion de versions décentralisés ont pris une place prépondérante. Parmi eux, Git et GitHub se démarquent tant par leur popularité que par leur efficacité.

3.4.3.1 Avantages

Git, développé par Linus Torvalds en 2005, s'est imposé comme le système de gestion de versions décentralisé de référence. Sa principale force réside dans sa capacité à suivre l'évolution d'un projet en enregistrant les modifications apportées au code source. Chaque modification est enregistrée sous forme de dépôts (*commits*), avec un message explicatif, permettant aux collaborateurs de comprendre facilement les évolutions du projet.

GitHub, lancé en 2008, est une plateforme qui utilise Git comme base pour l'entreposage et la gestion de projets. C'est une vitrine virtuelle où les développeurs peuvent héberger leurs dépôts Git et collaborer de manière transparente. L'aspect social de GitHub, avec ses fonctionnalités de suivi des projets, de gestion des problèmes et de demandes de fusion, en fait un lieu de choix pour les projets en code source ouvert et collaboratifs.

En sciences sociales numériques, où le partage et la collaboration sont essentiels, Git et GitHub offrent plusieurs avantages majeurs. Tout d'abord, ils permettent de suivre les modifications apportées au code, ce qui facilite la reproductibilité des résultats. Les chercheurs peuvent revenir à n'importe quelle version précédente du code, ce qui est particulièrement utile pour corriger des erreurs ou analyser l'impact de différentes approches.

De plus, Git et GitHub favorisent le travail collaboratif. Plusieurs chercheurs peuvent travailler sur le même projet simultanément, chacun dans sa branche de développement. Une fois les modifications effectuées, il est possible de fusionner les branches pour intégrer les changements. Cette approche évite les conflits majeurs et facilite la répartition des tâches au sein de l'équipe.

Enfin, l'aspect de code source ouvert de GitHub permet aux chercheurs en sciences sociales numériques de partager leurs codes avec la communauté académique et de bénéficier des contributions d'autres chercheurs. Cela favorise un environnement de partage des connaissances et de collaboration fructueuse.

3.4.3.2 Inconvénients

Cependant, Git et GitHub ne sont pas sans leurs défis. La courbe d'apprentissage peut être raide pour les débutants, car ces outils impliquent des concepts spécifiques tels que les branches, les conflits de fusion et les requêtes de tirage. De plus, bien que GitHub offre un niveau de gratuité pour les projets en code source ouvert, des frais peuvent être appliqués pour des fonctionnalités avancées ou pour des projets privés.

3.4.3.3 Comment les utiliser efficacement (en parallèle à Dropbox, etc.)

Pour utiliser Git et GitHub efficacement dans un contexte de recherche en sciences sociales numériques, il est recommandé de suivre quelques bonnes pratiques. Tout d’abord, il est important de structurer son dépôt Git de manière logique, en organisant les fichiers et les dossiers de manière cohérente. Les messages de commit doivent être descriptifs et clairs, pour permettre à tous les collaborateurs de comprendre les changements effectués.

Il est également conseillé de travailler sur des branches distinctes pour chaque fonctionnalité ou modification majeure. Cela facilite la gestion des changements et minimise les conflits lors de la fusion. Les chercheurs devraient également consulter régulièrement les projets et les problèmes sur GitHub pour encourager une communication ouverte et résoudre rapidement les problèmes.

L’utilisation de Git et de GitHub peut être complémentaire à d’autres outils d’entreposage, tels que Dropbox ou Google Drive. Ces derniers peuvent être utilisés pour entreposer des fichiers non liés au code, tels que des données brutes non sensibles ou des documents de recherche, tandis que Git et GitHub gèrent le code source et ses évolutions.

Bien qu’il existe plusieurs alternatives à l’utilisation combinée de Git et de GitHub sur le marché, ces deux plateformes liées continuent de dominer le domaine de la gestion de versions décentralisée. Parmi les alternatives notables, on peut citer Mercurial, Bitbucket, GitLab et SourceForge. Chacun de ces outils offre des fonctionnalités similaires à celles de Git et GitHub, mais il est important de comprendre pourquoi Git et GitHub restent les choix privilégiés pour les chercheurs en sciences sociales numériques.

3.4.3.4 Pourquoi prioriser Git et GitHub pour les chercheurs en sciences sociales

1. *Intégration et adoption répandue* : Git est devenu un standard de facto dans l’industrie du développement logiciel. Sa popularité et son adoption répandue signifient que de nombreuses ressources d’apprentissage, des tutoriels et des forums de support sont disponibles en ligne, ce qui facilite l’utilisation de cet outil pour les chercheurs en sciences sociales débutants. GitHub, en tant que plateforme principale de gestion des versions, bénéficie également d’une grande base d’utilisateurs et d’une communauté active, ce qui encourage la collaboration et le partage des connaissances.
2. *Facilité de collaboration* : Git et GitHub sont conçus pour faciliter la collaboration entre les individus et les équipes. Les chercheurs en sciences sociales travaillent souvent ensemble sur des projets de recherche, et la capacité de suivre les modifications, de gérer les conflits et de fusionner les contributions devient essentielle. L’interface conviviale de GitHub, avec des fonctionnalités telles que les demandes de fusion et les commentaires en ligne, simplifie grandement la collaboration.

3. *Visibilité et partage* : GitHub brille par sa fonctionnalité de projet open source, qui permet aux chercheurs en sciences sociales de partager leurs travaux avec la communauté mondiale. Les projets en code source ouvert sont visibles et accessibles à tous, favorisant ainsi la collaboration et l'examen par les pairs. Cela peut être particulièrement bénéfique pour les chercheurs souhaitant contribuer à des initiatives académiques et collaborer à des projets interdisciplinaires.
4. *Suivi des versions et recherche reproductible* : Les chercheurs en sciences sociales doivent s'assurer que leurs travaux sont reproductibles et vérifiables. Git permet de suivre les versions du code, ce qui signifie que les chercheurs peuvent retrouver facilement des versions antérieures pour reproduire des analyses spécifiques ou corriger des erreurs. Cette fonctionnalité est cruciale pour maintenir l'intégrité des résultats de recherche.
5. *Infrastructure et sécurité* : GitHub offre une infrastructure robuste pour l'entreposage sécurisé des dépôts Git. Les chercheurs peuvent être assurés que leurs travaux sont sauvegardés et protégés contre les pertes de données accidentelles. De plus, les contrôles d'accès et les autorisations granulaires de GitHub permettent aux chercheurs de contrôler qui peut accéder et contribuer à leurs projets.

En somme, Git et GitHub offrent aux chercheurs en sciences sociales numériques un moyen puissant de gérer leur code, de collaborer efficacement et de contribuer à la communauté académique grâce à l'open source. Bien que leur apprentissage puisse représenter un défi initial, les avantages qu'ils apportent en termes de suivi des versions, de collaboration et de partage des connaissances en font des outils essentiels dans l'arsenal de tout chercheur moderne.

3.4.3.5 Pratiques à éviter sur GitHub pour les chercheurs en sciences sociales

Lorsque les chercheurs en sciences sociales utilisent GitHub pour partager leur code, collaborer sur des projets et contribuer à la communauté académique, il est essentiel de connaître les pratiques à éviter. En effet, certaines erreurs peuvent compromettre la sécurité, la confidentialité et l'efficacité de la recherche. Voici quelques éléments à éviter :

1. *Entreposer des informations sensibles* : Évitez d'entreposer des données sensibles ou confidentielles sur GitHub. Cela inclut les données de sondages, les informations personnelles identifiables et tout autre contenu pouvant porter atteinte à la vie privée des individus. Assurez-vous de supprimer ou de masquer soigneusement ces informations avant de les télécharger sur la plateforme.
2. *Inclure des mots de passe et clés d'accès* : Ne jamais inclure de mots de passe, de clés d'accès ou d'informations d'identification dans votre code source. Cela peut compromettre la sécurité de vos systèmes et de vos données. Utilisez plutôt des méthodes sécurisées pour gérer ces informations, telles que les variables d'environnement ou les fichiers de configuration externes.

3. *Entreposer des fichiers lourds* : Évitez d’entreposer des fichiers volumineux sur GitHub, notamment des fichiers binaires, des données brutes massives ou des ensembles de données volumineux. Ces fichiers peuvent ralentir les opérations de clonage et de fusion, ce qui affecte la performance globale du dépôt. Utilisez plutôt des services d’entreposage dédiés pour ces fichiers et fournissez des liens vers ces ressources dans votre dépôt.
4. *Inclure des identifiants personnels* : Évitez de publier vos propres identifiants personnels, tels que des numéros de sécurité sociale, des numéros de carte de crédit ou d’autres informations confidentielles. Ces informations pourraient être exploitées à des fins malveillantes si elles tombent entre de mauvaises mains.
5. *Ignorer les pratiques de branches et de fusion* : Évitez de fusionner directement du code dans la branche principale (habituellement appelée *main* ou *master*). Utilisez plutôt des branches distinctes pour les fonctionnalités et les corrections, et suivez les pratiques de fusion pour intégrer proprement les changements. Ignorer ces pratiques peut entraîner des conflits et une perte de trace des modifications.
6. *Ignorer les commentaires des collaborateurs* : Lorsque vous travaillez avec d’autres chercheurs, ne négligez pas les commentaires et les suggestions qu’ils fournissent. Les retours d’expérience et les idées des autres peuvent contribuer à améliorer la qualité de votre code et de vos analyses.
7. *Ne pas documenter* : Évitez de ne pas documenter votre code. Une documentation claire et détaillée est essentielle pour permettre à d’autres chercheurs de comprendre vos méthodes et vos résultats. Utilisez des commentaires explicatifs et fournissez des explications sur la manière d’exécuter votre code.

En suivant ces conseils et en évitant ces erreurs courantes, les chercheurs en sciences sociales peuvent garantir la sécurité, la qualité et l’efficacité de leurs projets sur GitHub. La responsabilité de préserver la confidentialité des données et de créer un environnement de travail collaboratif et respectueux repose sur les épaules de chaque contributeur.

3.4.3.6 Exemple d’utilisation de Git et de GitHub pour un chercheur en sciences sociales

Dans le contexte de la recherche en sciences sociales numériques, la gestion efficace du code, la collaboration transparente et la préservation des données sensibles sont des impératifs. Imaginons que vous êtes un jeune chercheur en sciences sociales qui étudie l’impact des médias sur l’opinion publique. Vous utilisez le langage de programmation R pour analyser des données de médias et des données de sondage. Bien que vous travailliez seul, vous souhaitez rendre votre travail accessible à votre équipe pour validation et permettre à vos collègues de contribuer aux améliorations. Voici comment vous pouvez utiliser Git et GitHub pour gérer votre projet de manière structurée et collaborative.

3.4.3.6.1 Étape 1 : Création d'un répertoire local et initialisation de Git

Ouvrez votre terminal et naviguez vers le dossier où vous souhaitez enregistrer votre projet.

```
cd chemin/vers/votre/dossier
```

Créez un nouveau répertoire pour votre projet et accédez-y.

```
mkdir mon_projet
```

```
cd mon_projet
```

Initialisez Git dans ce répertoire.

```
git init
```

3.4.3.6.2 Étape 2 : Ajout de votre code et de vos fichiers

Ajoutez vos fichiers R contenant le code pour l'analyse des médias et des sondages dans le répertoire. Par exemple, vous pouvez avoir des fichiers *analyse_medias.R* et *analyse_sondages.R*.

Utilisez la commande `git status` pour vérifier l'état de vos fichiers.

```
git status
```

3.4.3.6.3 Étape 3 : Ajout, validation et commit de vos modifications

Ajoutez vos fichiers pour qu'ils soient prêts à être validés.

```
git add -A
```

Validez vos modifications avec un message descriptif.

```
git commit -m "Ajout du code d'analyse des médias et des sondages"
```

3.4.3.6.4 Étape 4 : Création du répertoire sur GitHub et du lien avec votre répertoire local

Allez sur GitHub et connectez-vous à votre compte. Créez un nouveau répertoire vide avec le nom *mon_projet*.

De retour dans votre terminal, ajoutez le lien GitHub à votre répertoire local.

```
git remote add origin https://github.com/votre-utilisateur/mon_projet.git
```

3.4.3.6.5 Étape 5 : Push de votre travail sur GitHub

Envoyez vos commits locaux vers GitHub.

```
git push -u origin master
```

3.4.3.6.6 Étape 6 : Collaboration avec vos collègues

Si vos collègues souhaitent contribuer à votre projet, ils peuvent *forker* votre répertoire sur GitHub, ce qui créera une copie dans leur propre compte.

Lorsqu'ils ont fait des modifications dans leur copie, ils peuvent soumettre une *pull request* pour vous demander de fusionner leurs modifications dans votre répertoire principal.

3.4.3.6.7 Étape 7 : Pull des modifications de vos collègues

Lorsque vos collègues ont soumis des modifications et vous ont demandé de les fusionner, vous pouvez mettre à jour votre répertoire local avec leurs changements.

```
git pull origin master
```

3.4.3.6.8 Étape 8 : Répéter le processus

Répétez les étapes 2 à 7 au fur et à mesure que vous développez votre projet, ajoutez du code, effectuez des analyses et collaborez avec vos collègues. Assurez-vous de valider et de pousser régulièrement vos modifications pour maintenir le dépôt à jour.

3.4.3.7 GitHub Desktop

Alors que le terminal reste une approche fondamentale pour maîtriser Git et GitHub, il existe des outils conviviaux tels que GitHub Desktop qui offrent une alternative intuitive. Cet outil simplifie le processus de gestion de versions décentralisée, en particulier pour ceux qui souhaitent commencer par une approche visuelle. Cependant, comprendre son fonctionnement et équilibrer les avantages et les inconvénients est essentiel.

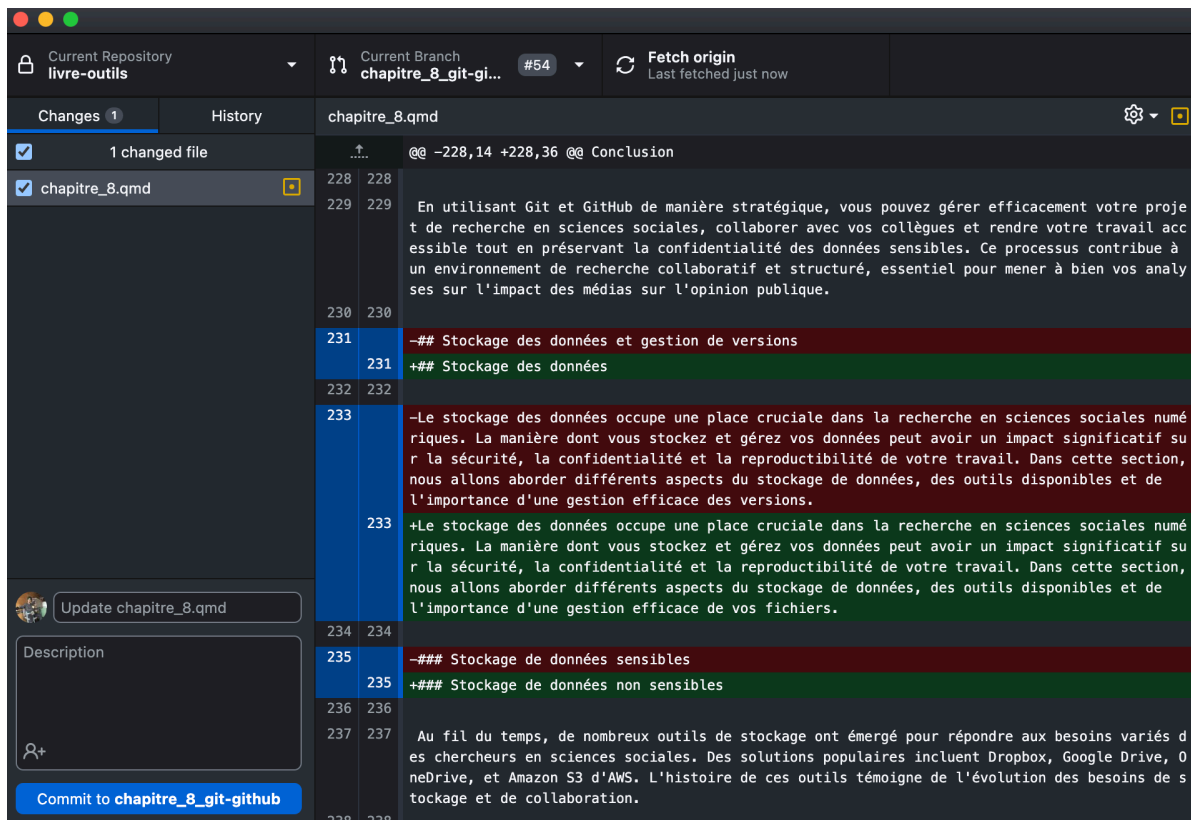


Figure 3.2: image

GitHub Desktop fournit une vue claire de vos dépôts, de vos modifications, de vos branches et de vos demandes de fusion. Il élimine la nécessité de mémoriser les commandes en ligne de terminal, ce qui peut être un défi pour certains chercheurs. L'application simplifie également la résolution des conflits lors de la fusion des branches.

Toutefois, en utilisant GitHub Desktop, il est possible de perdre la compréhension des commandes Git en ligne de commande, ce qui pourrait devenir un inconvénient si vous devez travailler dans un environnement sans interface visuelle. De plus, GitHub Desktop est spécifiquement conçu pour interagir avec GitHub. Si vous devez travailler avec d'autres plateformes de gestion de versions, cela pourrait poser des problèmes.

La décision entre l'utilisation du terminal et de GitHub Desktop dépend de vos préférences et de vos besoins. Pour les chercheurs qui débutent, GitHub Desktop offre une transition en douceur vers les concepts de gestion de versions. Cependant, il est important de ne pas se limiter à une interface visuelle. Comprendre les commandes Git en ligne de commande reste essentiel pour résoudre des problèmes complexes, gérer des projets avancés et collaborer avec d'autres chercheurs qui utilisent des approches basées sur le terminal.

3.4.3.8 Conclusion

En utilisant Git et GitHub de manière stratégique, vous pouvez gérer efficacement votre projet de recherche en sciences sociales, collaborer avec vos collègues et rendre votre travail accessible tout en préservant la confidentialité des données sensibles. Ce processus contribue à un environnement de recherche collaboratif et structuré, essentiel pour mener à bien vos analyses sur l'impact des médias sur l'opinion publique.

##Stockage de données (AWS, Valeria)

3.4.4 Entreposage de données sensibles

Lorsqu'il s'agit d'entreposer des données sensibles, tels que des données de sondage comportant des informations personnelles identifiables, la sécurité et la confidentialité sont essentielles. Comme abordé précédemment, GitHub n'est pas adapté à l'entreposage de telles données en raison de ses caractéristiques publiques et de son orientation vers le code source ouvert. Une solution courante est d'utiliser des services de cloud sécurisés, tels qu'AWS, qui offrent des mesures de sécurité robustes pour protéger vos données sensibles.

AWS regroupe un ensemble de services *cloud* proposés par Amazon. Il offre une vaste gamme de services, allant de l'entreposage et de la gestion des données à la computation et à l'analyse avancée. AWS est conçu pour offrir une infrastructure hautement évolutive et sécurisée, ce qui en fait un choix attrayant pour les chercheurs qui gèrent des données sensibles. L'outil présente de multiples avantages:

1. *Sécurité robuste* : AWS met l'accent sur la sécurité, avec des fonctionnalités telles que le chiffrement des données en transit et au repos, la gestion des accès basée sur les rôles et la conformité à des normes de sécurité strictes.
2. *Scalabilité* : AWS permet de faire évoluer vos ressources en fonction des besoins, garantissant des performances optimales même lorsque vos projets de recherche croissent en taille et en complexité.
3. *Flexibilité* : AWS propose une variété de services adaptés à différentes utilisations, allant de l'entreposage de données au calcul intensif pour l'analyse avancée.
4. *Collaboration simplifiée* : Bien que le coût d'entrée soit généralement bas, la possibilité de partager des ressources avec des collègues et de travailler en équipe rend AWS adapté à la collaboration.

AWS n'est pas le seul service cloud disponible. Microsoft Azure et Google Cloud Platform (GCP) sont des concurrents majeurs offrant des fonctionnalités similaires. Lorsque vous choisissez un fournisseur, prenez en compte les coûts, la convivialité et les fonctionnalités offertes. Le coût d'utilisation d'AWS peut varier en fonction des services utilisés, de la quantité de données entreposées et de la capacité de calcul requise. Lorsque vous travaillez seul, le coût peut

sembler élevé par rapport à l'utilisation de solutions gratuites telles que Dropbox. Cependant, en équipe, la répartition des coûts peut rendre AWS plus abordable.

3.4.4.1 Exemple d'utilisation d'AWS pour entreposer et accéder à des données de sondages dans RStudio

Imaginez un jeune chercheur en sciences sociales qui travaille sur une analyse comparative de données de sondages recueillies sur plusieurs décennies. Pour maintenir la sécurité des données sensibles et faciliter l'accès pour les analyses dans RStudio, il décide d'utiliser AWS pour l'entreposage et la gestion de ses données.

3.4.4.1.1 Étape 1 : Création d'un compte AWS et configuration

Le chercheur crée un compte AWS et configure ses paramètres de sécurité, y compris la configuration de l'authentification à deux facteurs pour renforcer la sécurité de son compte.

3.4.4.1.2 Étape 2 : Création d'un espace d'entreposage S3

Le chercheur crée un compartiment Amazon S3 (Simple Storage Service) pour entreposer ses données de sondage. Il choisit une région AWS et définit les paramètres de sécurité appropriés, tels que le chiffrement des données.

3.4.4.1.3 Étape 3 : Transfert des données vers Amazon S3

Le chercheur transfère les données de sondage dans son compartiment Amazon S3 à l'aide de l'interface en ligne AWS ou d'outils d'importation.

3.4.4.1.4 Étape 4 : Configuration des autorisations

Pour sécuriser davantage les données, le chercheur configure les autorisations d'accès aux données dans Amazon S3. Il attribue des rôles et des politiques d'accès spécifiques aux utilisateurs, garantissant que seules les personnes autorisées peuvent accéder aux données.

3.4.4.1.5 Étape 5 : Configuration d'accès dans RStudio

Le chercheur installe le package *aws.s3* dans RStudio pour accéder à ses données entreposées dans Amazon S3. Il configure également les informations d'identification AWS dans son environnement RStudio.

3.4.4.1.6 Étape 6 : Accès et analyse des données dans RStudio

À l'aide du package *aws.s3*, le chercheur peut maintenant accéder à ses données directement dans RStudio par quelques lignes de code. Il peut charger les données dans des structures de données R et effectuer des analyses statistiques, des visualisations et des croisements.

3.4.4.1.7 Étape 7 : Sécurité et conservation des données

Après avoir effectué ses analyses, le chercheur peut choisir de conserver les données de sondage dans Amazon S3 en utilisant les politiques de conservation appropriées. Il peut également archiver des copies de sauvegarde pour garantir l'intégrité des données à long terme.

Dropbox se concentre principalement sur l'entreposage et la collaboration de fichiers, alors que AWS offre une gamme de services *cloud*, y compris l'entreposage sécurisé de données sensibles et la mise en place d'infrastructures évolutives. GitHub, d'autre part, se concentre sur la gestion de versions et la collaboration de code source. Chaque outil a son propre domaine d'expertise et peut être utilisé de manière complémentaire pour différents aspects de la recherche.

3.4.5 Conclusion

L'entreposage des données est une étape cruciale dans la recherche en sciences sociales numériques. Choisissez des outils adaptés à la sensibilité des données, privilégiez des services sécurisés comme AWS pour les données sensibles, et utilisez Dropbox pour la collaboration et l'entreposage de fichiers non sensibles. Une gestion efficace des versions, de la structure des dossiers et de la sécurité garantira l'intégrité de vos données et facilitera la collaboration tout au long de vos projets de recherche.

4 Outils de gestion de la littérature

4.1 Introduction

La réalisation d'une revue de littérature représente la première marche à franchir avant d'initier tout projet de recherche. C'est en s'immergeant dans le dialogue établi entre chercheurs et chercheuses qui nous ont précédés que nous pouvons véritablement saisir l'évolution des idées et les contours actuels des débats sur un sujet de recherche. Cette compréhension permet non seulement de cerner les zones de connaissances encore inexplorées, mais aussi de situer comment inscrire notre propre recherche dans un domaine circonscrit. Loin d'être une simple collection de sources choisies aléatoirement, elle doit suivre une démarche méthodique et réfléchie, intégrant des critères tels que la pertinence, la crédibilité, la nouveauté sur le sujet, la diversité des perspectives, la qualité méthodologique, l'impact et la citation des travaux et leur objectivité. En choisissant des sources qui répondent à ces exigences, on s'assure que la revue de littérature offre une perspective complète et actuelle sur le sujet. La gestion rigoureuse de la littérature n'est pas seulement un gage de rigueur académique; elle fournit aussi des directions claires pour la poursuite des travaux dans le domaine exploré. En dressant un portrait structuré et critique de la littérature existante, le chercheur ou la chercheuse établit les fondements solides de sa démarche scientifique. Cette étape préliminaire éclaire la formulation de la question de recherche, en mettant en lumière les interstices où de nouvelles contributions sont possibles et souhaitables. Ainsi, une revue de littérature systématique n'est pas simplement un recensement de connaissances ; c'est une étape cruciale qui façonne le cadre de la recherche future, garantissant que chaque nouvelle étude contribue de manière significative au corps grandissant de connaissances dans un domaine.

4.2 Snowballing

4.2.1 Introduction à la méthode

Dans le cadre de l'élaboration d'une recherche scientifique, la phase de revue de la littérature constitue une étape fondamentale permettant de circonscrire le champ d'étude, d'identifier les travaux antérieurs pertinents et de déceler les lacunes dans les connaissances existantes. Parmi les différentes approches méthodologiques adoptées pour mener à bien cette revue, la méthode du snowballing se distingue par son adaptabilité et sa capacité à englober un large spectre de

travaux pertinents, en particulier dans les phases préliminaires de la recherche où la question d'étude peut ne pas être entièrement définie.

Le snowballing est une technique de recherche bibliographique qui se fonde sur la sélection initiale de quelques publications clés dans le champ d'intérêt. Par la suite, à partir de ces références clés, le chercheur identifie de manière itérative d'autres travaux pertinents en examinant les références citées (backward snowballing) ou les articles citant ces travaux (forward snowballing) (Nightingale, 2009). La méthode du forward snowballing est particulièrement utile lorsque les chercheurs cherchent à comprendre l'impact de certains articles fondamentaux et la manière dont ils ont influencé les recherches ultérieures. Il permet aux chercheurs de suivre l'évolution des idées et des méthodologies au fil du temps, en voyant comment les concepts initiaux ont été élargis, remis en question ou appliqués dans différents contextes. D'un autre côté, la méthode de backward snowballing est pertinente lorsque l'objectif est d'identifier la littérature fondamentale sur un sujet. En fouillant dans les références des articles clés, les chercheurs peuvent découvrir les travaux fondateurs qui ont jeté les bases des tendances actuelles de la recherche. Bref, de manière générale, le snowballing est une méthode particulièrement pertinente lorsque la question de recherche est encore à un stade général ou ouvert, permettant ainsi une exploration exhaustive du domaine sans la contrainte d'une hypothèse de recherche spécifique préalablement établie.

4.2.2 Mise en œuvre pratique

La mise en œuvre du snowballing débute par l'identification d'un petit ensemble d'articles récents et pertinents sur le sujet d'étude. Cette sélection initiale doit refléter la diversité du domaine en termes d'éditeurs, d'années de publication et d'auteurs. La revue systématique se poursuit ensuite par un examen itératif des références incluses dans ces articles, ainsi que des publications qui les citent, jusqu'à ce que plus aucun nouvel article pertinent ne soit identifié. Il est recommandé de compiler les informations essentielles de chaque publication dans une grille d'analyse, facilitant ainsi l'évaluation de leur pertinence et l'organisation des lectures ultérieures.

L'adoption d'outils de programmation, tels que R, dans la mise en œuvre pratique du snowballing, peut considérablement améliorer l'analyse des données bibliographiques. R, un langage de programmation et un environnement logiciel dédiés aux statistiques et à la visualisation graphique, fournit une gamme étendue de packages permettant l'analyse et la représentation graphique des informations issues de la recherche bibliographique. Ainsi, la capacité de R à générer des visualisations graphiques est particulièrement pertinente afin d'interpréter les données issues de snowballing. Le package ggplot2 permet de créer des graphiques détaillés illustrant, par exemple, la distribution des publications les plus pertinentes dans un champ en fonction de certains critères, tels que le réseaux de co-citations et l'année de publication, ce qui facilite ainsi la compréhension des dynamiques au sein du domaine étudié.

4.2.3 Avantages

L'un des principaux avantages du snowballing réside dans sa simplicité et sa rapidité d'exécution, offrant la possibilité de couvrir efficacement un vaste domaine de recherche avec une charge de travail relativement modérée. Contrairement aux revues systématiques classiques qui exigent l'accès à plusieurs bases de données et un protocole de recherche rigoureux, le snowballing permet d'obtenir des résultats en se basant sur un nombre restreint de sources initiales. De plus, cette méthode présente un bon ratio entre les publications sélectionnées pertinentes et celles jugées hors sujet, facilitant ainsi l'identification des travaux véritablement significatifs pour la recherche en cours.

4.2.4 Défis

Malgré ses avantages, le snowballing n'est pas exempt de défis. La méthode peut conduire à une surreprésentation de certaines perspectives, auteurs ou écoles de pensée, introduisant un biais dans la revue de littérature. De plus, la sélection des articles clés et la détermination de leur pertinence initial reposent fortement sur le jugement du chercheur, ce qui peut occasionner des erreurs de sélection. Enfin, en raison de son caractère moins formalisé par rapport à d'autres méthodes de revue systématique, le snowballing peut être perçu comme moins rigoureux dans certains milieux académiques, ce qui peut impacter la reconnaissance des travaux de recherche qui l'utilisent. En revanche, le snowballing offre une méthode flexible et efficace pour la revue de la littérature, particulièrement adaptée aux phases exploratoires d'une recherche. Malgré certains défis inhérents à son utilisation, cette approche permet d'embrasser une grande diversité de travaux et de perspectives, contribuant ainsi à une compréhension approfondie et nuancée du sujet à l'étude.

4.3 Scoping review

4.3.1 Introduction à la méthode

La revue de portée, ou scoping review, représente une approche systématique et structurée de revue de la littérature, principalement utilisée afin d'explorer des questions de recherche étendues ou complexes. Son objectif fondamental est de cartographier l'état actuel des connaissances sur un sujet donné, en identifiant les concepts clés, les principaux domaines de recherche ainsi que les lacunes existantes dans un domaine d'étude spécifique (Munn et al., 2018). Cette méthode se distingue par son champ d'application large et son objectif exploratoire, permettant ainsi une compréhension holistique des thématiques étudiées.

Contrairement au snowballing, qui démarre avec un ensemble restreint de documents initiaux et les élargit de manière itérative en ajoutant de nouvelles sources identifiées à travers les

références des travaux précédents, la revue de portée adopte une stratégie de recherche systématique plus large. Elle vise à couvrir une variété étendue de sources d'information afin de fournir une vue d'ensemble complète et nuancée de la question de recherche. Ainsi, tandis que le snowballing peut être idéal pour approfondir un sujet à partir d'un point de départ connu, la revue de portée offre une image panoramique et détaillée, idéale pour les étapes initiales de la recherche lorsque l'exploration et la clarification du champ sont nécessaires.

4.3.2 Mise en oeuvre pratique

La mise en oeuvre d'un scoping review partage de nombreux points communs avec une revue systématique, mais se distingue par ses objectifs exploratoires et sa portée étendue. Selon Levac et al. (2010), cette démarche devrait être réalisée par une équipe multidisciplinaire, tout en restant transparente et répliquable par une autre équipe. Arksey and O'Malley ont défini un cadre méthodologique en cinq étapes afin d'orienter efficacement la réalisation d'une revue de portée :

- Identification de la question de recherche : Cela implique de définir clairement et assez précisément le champ d'étude afin de guider la revue.
- Identification des études pertinentes : Cela nécessite le développement d'une stratégie de recherche exhaustive pour rassembler une bassin de documents potentiels.
- Sélection des études : À cette étape, les études sont évaluées et sélectionnées en fonction de critères préalablement définis afin de s'assurer qu'elles répondent aux objectifs de la revue.
- Extraction et organisation des données : Les informations clés de chaque étude sélectionnée sont extraites et systématiquement organisées pour faciliter l'analyse et la synthèse.
- Synthèse, résumé et communication des résultats : Les données sont ensuite regroupées et analysées pour identifier les tendances, les thèmes, les lacunes dans la recherche existante et les domaines nécessitant des recherches futures, avec un rapport final qui résume et présente les résultats de manière cohérente.

En suivant ces étapes, les chercheurs et chercheuses peuvent mener une revue de portée de manière systématique et rigoureuse, permettant une exploration complète du sujet tout en fournissant des orientations précieuses pour les recherches futures.

4.3.3 Avantages

Opter pour une revue de portée plutôt qu'une revue de littérature, un snowballing ou même une revue systématique présente plusieurs avantages distinctifs. Premièrement, contrairement à une revue de littérature classique qui peut manquer de structure et être subjective, la revue de portée offre une approche systématique et transparente, permettant de cartographier

exhaustivement un champ d'étude. Cela la rend particulièrement utile pour explorer des domaines de recherche étendus ou émergents, où la compréhension des concepts, des tendances et des lacunes est nécessaire. Contrairement au snowballing, qui part d'un nombre restreint de sources et s'étend de manière itérative, la revue de portée utilise une méthode de recherche plus vaste et systématique, fournissant ainsi une vue d'ensemble plus complète et nuancée du sujet. En comparaison avec les revues systématiques, qui se concentrent sur des questions de recherche spécifiques et nécessitent des critères d'inclusion et d'évaluation de la qualité stricts, les revues de portée embrassent une variété plus large de matériaux et de types d'études, ce qui est idéal pour les phases exploratoires de la recherche.

4.3.4 Défis

La réalisation d'un scoping review présente également divers défis méthodologiques et pratiques significatifs. Un des obstacles majeurs réside dans la formulation d'une question de recherche précise et bien définie. Une question trop vague ou trop large peut entraîner des difficultés lors de la sélection des sources, créant une incertitude quant aux études à inclure ou à exclure. De plus, la nature expansive d'une revue de portée, couvrant un large éventail d'informations, exige d'importantes ressources humaines, temporelles et financières, ce qui peut limiter sa faisabilité pour certaines équipes de recherche. Par ailleurs, bien que la méthodologie d'un scoping review soit rigoureuse, elle souffre d'un manque de standardisation universelle dans le domaine académique, ce qui peut conduire à des variations dans la qualité et l'approche des revues produites. Cette absence de consensus méthodologique ajoute une couche de complexité, rendant parfois le processus de revue plus exigeant et sujet à interprétation.

4.3.5 Revue Systématique

Dans le domaine des sciences sociales, les revues systématiques et les méta-analyses offrent des outils précieux pour synthétiser et évaluer de manière rigoureuse les preuves issues de multiples études. Ces approches permettent de dégager des tendances, d'identifier des consensus ou des divergences dans la littérature et d'éclairer les débats politiques et sociaux grâce à une base de données empirique solide.

4.3.5.1 Importance des Revues Systématiques en sciences sociales

Les revues systématiques en sciences sociales permettent d'aborder des questions complexes et multidimensionnelles telles que les facteurs influençant la réussite des démocraties, l'efficacité des politiques publiques, ou encore l'impact des mouvements sociaux. En recensant de manière exhaustive et méthodique la littérature existante sur un sujet donné, les chercheurs peuvent offrir une synthèse objective qui reflète l'état actuel des connaissances. Cela est particulièrement

utile dans un domaine où les études peuvent être influencées par des biais idéologiques, permettant ainsi de distinguer les preuves empiriques solides des opinions ou des théories moins étayées.

4.3.5.2 Méta-Analyse

La méta-analyse, souvent intégrée dans le cadre d'une revue systématique, va plus loin en combinant quantitativement les résultats de différentes études pour produire une estimation globale de l'effet étudié. Dans le contexte des sciences sociales, cela peut signifier, par exemple, quantifier l'effet des campagnes d'information sur la participation électorale ou mesurer l'impact des politiques économiques sur la réduction de la pauvreté. La méta-analyse offre ainsi une puissance statistique accrue et une meilleure généralisabilité des conclusions, transcendant les limites des études individuelles.

Bien que complémentaires, les revues systématiques et les méta-analyses diffèrent dans leur approche et leur finalité :

- **Revue Systématique** : Elle vise à rassembler de manière exhaustive toutes les études pertinentes sur une question de recherche, en évaluant leur qualité et en synthétisant leurs conclusions de manière descriptive. Elle est particulièrement utile pour cartographier le paysage de la recherche et comprendre la diversité des approches et des résultats.
- **Méta-Analyse** : Elle se concentre sur la synthèse quantitative des données issues de multiples études. En appliquant des méthodes statistiques pour intégrer les résultats, elle cherche à estimer l'effet global d'un phénomène, offrant une réponse numérique précise à une question de recherche.

4.3.5.3 Valeur Ajoutée dans les sciences sociales

L'adoption de ces méthodologies en sciences sociales enrichit le débat académique et informe la pratique politique en fournissant des preuves empiriques consolidées. Elles aident à surmonter le problème de la surabondance d'informations et de la variabilité des résultats d'études individuelles, offrant ainsi une base plus solide pour la prise de décision politique et l'élaboration de théories. De plus, en identifiant les lacunes dans la recherche existante, elles orientent les futures enquêtes vers les zones moins explorées ou controversées, contribuant à l'avancement de la discipline.

La revue systématique peut être menée en suivant les 24 étapes décrites par (Citer muka_etal20).

1. Définir la question de recherche.
2. Établir l'équipe.
3. Définir la stratégie pour l'obtention des références.

4. Définir les critères d'inclusion et d'exclusion.
5. Définir le formulaire d'extraction des données.
6. Écrire le protocole de recherche servant à guider le processus.
7. Appliquer la stratégie de recherche à diverses bases de données.
8. Rassembler les références et les résumés.
9. Éliminer les doublons.
10. Filtrer les résultats à partir des titres et résumés.
11. Comparer les résultats des codeurs.
12. Télécharger les textes entiers et appliquer les critères d'inclusion et d'exclusion
13. Contacter les experts du champs et s'informer à propos des éléments manquants ou non-publiés
14. Rechercher manuellement des références additionnelles
15. Sélectionner les références à inclure et dessiner le diagramme de flux
16. Extraire les données des articles
17. Évaluer la qualité des articles et les risques de biais
18. Préparer la base de donnée pour l'analyse.
19. Construire une synthèse descriptive des données.
20. Décider si une méta-analyse est appropriée.
21. Explorer l'hétérogénéité.
22. Évaluer les biais.
23. Évaluer la qualité des preuves et la confiance dans les résultats
24. Rédiger le rapport de la revue systématique

4.4 Outil de gestion de la littérature

4.4.1 Covidence

Les outils numériques de données massives, comme Covidence, jouent un rôle crucial en facilitant le travail des chercheurs lors de la récolte de données pour des analyses empiriques, et en offrant des ressources essentielles durant d'autres étapes du cycle de recherche. Covidence, géré par une compagnie sans but lucratif, est spécifiquement conçu pour aider dans la réalisation de revues systématiques de littérature. Cette plateforme en ligne simplifie l'évaluation d'une quantité importante d'études scientifiques, en réduisant le temps nécessaire et en rendant le processus plus simple et intuitif. Reconnu pour ses trois phases méthodiques — « Title and abstract screening », « Full text review » et « Extraction » — Covidence facilite l'importation de données volumineuses depuis des bases de données bibliographiques et interroge plusieurs bibliothèques. Cela offre un accès à des milliers d'études pertinentes qui aident les chercheurs à élaborer un cadre théorique exhaustif. La revue de littérature sur Covidence implique un double codage, ce qui signifie que l'évaluation des études est effectuée manuellement par deux codeurs, permettant ainsi une analyse rigoureuse et détaillée des informations recueillies.

La première phase, le « Title and abstract screening », consiste à examiner les titres et résumés des articles récupérés. Pour optimiser cette tâche, il est crucial de définir des critères précis pour évaluer la pertinence des articles par rapport au sujet étudié. Durant cette étape, souvent prolongée en raison du volume conséquent de la littérature, les chercheurs doivent régulièrement se consulter pour résoudre les divergences d'opinions et parvenir à un consensus.

Après la révision des titres et résumés, vient le « Full text review », qui implique l'examen complet des textes pré-sélectionnés. Les chercheurs doivent alors voter « oui », « non », ou « peut-être » pour décider de la conservation des textes, ce qui peut inclure ou exclure un article ou le faire progresser vers l'étape suivante. Cette phase, bien qu'elle concerne moins de documents, reste exigeante et chronophage à cause des discussions nécessaires pour résoudre les désaccords.

La dernière étape, celle de l'extraction, consiste à collecter les données pertinentes des études retenues en se basant sur une grille de codification prédéfinie, visant à obtenir un consensus entre les codeurs. L'extraction révèle les théories, les méthodologies et les conclusions des études sélectionnées.

Une fois les étapes de la revue systématique complétées, Covidence simplifie l'exportation des données extraites sous formes de tableaux, graphiques et rapports pour la méta-analyse ou la rédaction d'articles scientifiques. Bien que Covidence soit largement utilisé et supporté par de nombreuses universités via des licences, d'autres plateformes comme DistillerSR, Archie, et Rayyan sont également populaires parmi les chercheurs, chacune répondant à des besoins et des budgets variés.

4.4.2 Avantages de Covidence

L'outil Covidence représente une avancée significative dans la pratique des revues systématiques, offrant une série d'avantages qui facilitent le processus de recherche. Sa capacité à importer et gérer efficacement les références de multiples bases de données bibliographiques permet une organisation optimale des articles potentiels à inclure dans une revue. De plus, Covidence soutient la collaboration entre chercheurs, autorisant plusieurs utilisateurs à travailler simultanément sur le même projet, ce qui favorise une approche plus inclusive et diversifiée de l'analyse. La plateforme automatise également les premières étapes de sélection des articles, réduisant ainsi la charge de travail manuel et le potentiel de biais de sélection. Les fonctionnalités pour l'extraction de données et l'évaluation de la qualité sont également des points forts, garantissant une analyse systématique et cohérente des informations recueillies. En outre, l'intégration de Covidence avec d'autres outils de recherche élargit sa compatibilité et son utilité dans le flux de travail académique.

4.4.3 Critères de sélection de l'outil Covidence

Bien que Covidence soit un outil développé par une organisation à but non lucratif, il ne répond pas entièrement aux critères du logiciel libre car son code source n'est pas ouvert pour modification et redistribution libre par la communauté. Cela pourrait limiter les utilisateurs qui souhaitent personnaliser l'outil pour des besoins spécifiques. Toutefois, son modèle de licence permet une large utilisation académique et non commerciale, ce qui le rend accessible à de nombreux chercheurs. Covidence se distingue par sa capacité à intégrer des données provenant de diverses bases de données bibliographiques, ce qui simplifie considérablement le processus de revue systématique de la littérature. L'outil est également compatible avec des référentiels de données comme EndNote et Zotero, facilitant ainsi l'importation et la gestion des références. Covidence jouit d'une large adoption dans la communauté des chercheurs en sciences de la santé, ce qui témoigne de sa fiabilité et de son efficacité. La plateforme bénéficie également d'un solide support des universités et d'institutions de recherche, qui offrent souvent des licences institutionnelles, rendant l'outil encore plus accessible. L'interface de Covidence est conçue pour être intuitive, ce qui réduit la courbe d'apprentissage pour les nouveaux utilisateurs. Elle guide les chercheurs à travers les différentes phases de la revue systématique de manière structurée, rendant le processus moins ardu comparé aux méthodes traditionnelles. Les fonctionnalités de double codage, de filtrage et d'extraction de données offrent une méthodologie rigoureuse et standardisée, essentielle pour maintenir la qualité des revues systématiques. Covidence excelle dans la facilitation de la collaboration entre chercheurs. La plateforme permet à plusieurs utilisateurs de travailler simultanément sur le même projet, de discuter des inclusions ou exclusions d'études, et de résoudre les désaccords efficacement. Cet aspect est particulièrement précieux dans des projets de grande envergure impliquant des équipes dispersées géographiquement. Coût et accessibilité

4.4.4 Inconvénients de Covidence

Cependant, l'utilisation de Covidence n'est pas exempte de contraintes. Le coût de l'abonnement peut constituer un obstacle pour certains chercheurs, limitant l'accès à un outil par ailleurs utile. La nécessité d'une familiarisation avec la plateforme introduit une courbe d'apprentissage qui peut retarder son adoption efficace, surtout pour ceux qui n'ont pas d'expérience préalable avec des outils similaires. Bien que Covidence propose des modèles pour l'extraction de données et l'évaluation de la qualité, ceux-ci peuvent ne pas convenir à tous les types d'études, en particulier celles qui nécessitent une approche plus personnalisée. Malgré ces défis, Covidence reste un outil précieux pour la conduite de revues systématiques, nécessitant une évaluation attentive de ses avantages et inconvénients dans le contexte spécifique de chaque projet de recherche.

4.5 Conclusion

Comme nous l'avons souligné précédemment, la revue de littérature constitue la première étape d'un processus de recherche. Il s'agit d'une tâche qui peut s'avérer fastidieuse, mais l'adoption des méthodologies et des outils décrits dans la section précédente la rend nettement plus facile et structurée. Ayant désormais en main la littérature nécessaire pour faire avancer sa recherche, il est primordial de disposer d'une méthode de gestion des références rigoureuse. Dans la section suivante, nous dévoilerons les fondements de la gestion des références et introduirons un outil utile qui facilite ainsi la structuration et l'accès à des sources d'information.

4.6 La gestion des références

4.6.1 Pourquoi citer ?

La citation des sources est une pratique incontournable dans le monde académique, essentielle à la préservation de la crédibilité académique et au maintien des normes éthiques. Elle sert de fondement à la contextualisation de nos recherches, nous permettant de situer nos travaux au sein d'un cadre scientifique établi et reconnu. Ce processus de contextualisation facilite non seulement la compréhension de l'évolution des connaissances dans un domaine donné, mais contribue également à la création d'une base de connaissances solide et dynamique, sur laquelle d'autres travaux peuvent être bâtis (Zaid et al., 2017). La référencement rigoureuse des travaux antérieurs garantit la reproductibilité des expériences et des analyses, un pilier central de la méthodologie scientifique. En fournissant des détails précis sur les méthodes et résultats, nous ouvrons la voie à la validation et à l'éventuelle réfutation de nos travaux, renforçant ainsi l'intégrité de la recherche (Hughes, 2013). De plus, la citation adéquate des sources est une marque de respect envers les contributions des autres chercheurs, assurant une juste attribution du mérite. Cela reconnaît l'importance de chaque découverte et idée dans l'avancement de la science, tout en prévenant le plagiat, une faute grave dans la recherche académique (Racz & Marković, 2018). Enfin, une référencement minutieuse aide à éviter les biais, en exposant clairement les fondements sur lesquels se base notre recherche. Cela permet une évaluation critique des sources et des perspectives, encourageant une approche plus équilibrée et nuancée dans l'analyse scientifique (Kostoff & Cummings, 2013). En résumé, la citation des sources est un acte fondamental qui englobe et adresse de multiples aspects cruciaux de la recherche académique : de la crédibilité et la contextualisation à la reproductibilité, de la création d'une base de connaissances solide à l'attribution correcte du mérite, tout en combattant le plagiat et en minimisant les biais. C'est dans ce contexte que des outils tels que Zotero prennent toute leur importance, en facilitant la gestion rigoureuse des références et en soutenant les chercheurs dans leur quête de rigueur et d'excellence académiques.

4.6.2 À quoi sert un logiciel de gestion bibliographique ?

Un outil de référence bibliographique est un logiciel conçu pour aider les scientifiques à gérer et à organiser leurs références bibliographiques de manière efficace. Ces outils s'avèrent particulièrement utiles lors de la rédaction d'articles de recherche, de thèses, de mémoires ou d'autres documents académiques. Voici quelques-unes des fonctions principales d'un tel outil :

1. Collecte de références : Les outils de référence bibliographique permettent aux personnes utilisatrices de collecter et d'importer des références bibliographiques à partir de bases de données, de catalogues de bibliothèques, de sites Web ou d'autres sources. Certains outils offrent même la possibilité d'extraire automatiquement les métadonnées à partir de documents PDF.
2. Organisation et classement : Les références collectées peuvent être organisées en différentes catégories et dossiers. Cela facilite la recherche ultérieure et permet de garder une vue d'ensemble claire de la bibliographie.
3. Citation et génération de bibliographies : L'un des avantages majeurs des outils de référence est leur capacité à générer automatiquement des citations et des bibliographies conformes à différents styles de citation (APA, MLA, Chicago, etc.). Ce processus permet de gagner énormément de temps en formatage. Les personnes utilisatrices peuvent insérer des références directement dans leurs documents sans avoir à se soucier des détails de formatage.
4. Collaboration : Certains outils offrent la possibilité de collaborer en ligne, ce qui donne l'occasion à plusieurs personnes de travailler sur une bibliographie commune. Cela peut être utile pour les projets de groupe ou de recherche partagée comme c'est le cas dans une chaire de recherche. En plus d'utiliser un même logiciel, l'utilisation d'un outil de référencement contribue à économiser du temps par la centralisation des données sur un même interface.
5. Recherche et exploration : De nombreux outils de référence bibliographique offrent des fonctionnalités de recherche avancée qui facilitent la découverte de nouvelles références liées à un sujet spécifique.
6. Synchronisation et sauvegarde : Les références et les bibliographies peuvent être synchronisées sur plusieurs appareils, ce offre la possibilité aux personnes utilisatrices d'accéder à leurs références où qu'elles soient. Les sauvegardes régulières assurent que les données ne soient pas perdues en cas de problème technique.
7. Suivi de lecture : Certains outils permettent aux personnes utilisatrices de suivre les articles et les documents qu'elles ont lus, ce qui est particulièrement utile pour garder une trace de la littérature pertinente.

8. Importation et exportation : Les outils de référence bibliographique autorisent généralement l'importation et l'exportation des références dans différents formats, ce qui facilite le transfert de données.

En résumé, un outil de référence bibliographique simplifie grandement le processus de gestion des références bibliographiques, de formatage ainsi que de création de bibliographies. De plus, ces outils offrent la flexibilité de changer de style de citation instantanément, facilitant l'adaptation aux exigences variées des revues scientifiques et permettant aux chercheurs de se consacrer à l'essence de leurs recherches sans se préoccuper des contraintes formelles et des détails de formatage. D'ailleurs, il existe divers outils de référence bibliographique, tels que : Zotero, EndNote et Mendeley.

Chaque logiciel offre des caractéristiques uniques tout en partageant des objectifs communs fondamentaux. Ils visent principalement à optimiser l'efficacité et la collaboration en centralisant les références, une commodité indéniable pour les chercheurs et les équipes académiques. Le choix d'un logiciel adapté aux besoins spécifiques des personnes l'utilisant dépend de plusieurs facteurs, notamment de la nécessité de partager les résultats de recherche et de collaborer sur des projets communs. Lorsque la collaboration est au cœur d'un projet, il est judicieux que tous les membres de l'équipe adoptent le même outil pour faciliter l'échange d'informations et la cohésion du groupe.

Zotero, EndNote et Mendeley, bien qu'ils partagent des principes de base similaires, se distinguent par des fonctionnalités spécifiques qui peuvent mieux s'aligner sur les préférences et exigences individuelles. La sélection d'un logiciel doit donc être guidée par une évaluation attentive de ses capacités à répondre aux besoins de l'utilisateur, tout en considérant des aspects cruciaux, tels que : le partage, la collaboration et la facilité d'utilisation. Il est essentiel de souligner l'importance de la préférence personnelle dans ce choix. L'interface utilisateur, la facilité d'intégration dans les flux de travail existants et la compatibilité avec d'autres outils numériques sont des critères qui influent grandement sur l'expérience utilisateur et, par conséquent, sur la productivité. En fin de compte, l'outil idéal est celui qui non seulement facilite la gestion des références mais s'intègre de manière efficace dans le quotidien académique de l'utilisateur, lui permettant ainsi de se concentrer pleinement sur la substance de ses recherches.

4.6.3 Pourquoi Zotero?

Zotero se distingue par sa gratuité et son accessibilité en tant que logiciel libre, avec un code ouvert largement soutenu par une communauté active sur GitHub, qui compte plus de 13 000 contributions. Cette plateforme propose une vaste gamme de fonctionnalités et permet l'ajout d'extensions pour enrichir son utilisation, ce qui le rend particulièrement puissant tout en restant facile à utiliser. Zotero est compatible avec diverses plateformes, notamment Windows, Mac, Linux, iOS et Android, facilitant ainsi la collaboration entre les membres d'une équipe de recherche qui utilisent différents systèmes. La bibliothèque Zotero peut être synchronisée sur plusieurs appareils via le service cloud payant de Zotero ou en configurant un espace de stockage

cloud personnel. Ce logiciel s'intègre parfaitement dans les projets de recherche utilisant LaTeX ou Quarto, permettant de générer et de maintenir à jour automatiquement des fichiers .bib. De plus, Zotero fonctionne avec des logiciels de traitement de texte tels que LibreOffice et Microsoft Office, et offre la possibilité de créer des bibliographies et des citations dans plus de 9 000 styles de citation différents, répondant ainsi aux divers besoins des chercheurs.

Zotero offre l'avantage significatif de centraliser les sources bibliographiques et les fichiers associés, simplifiant grandement le partage de documents au sein des équipes de recherche. Avec Zotero, il est possible d'ajouter des PDF et de les synchroniser dans des groupes de travail, ce qui élimine le besoin de recourir à des dossiers partagés ou d'envoyer des documents par courriel ou via des plateformes de partage de fichiers. Cette centralisation permet également de réaliser des recherches par mot-clé à travers l'ensemble des sources d'une bibliothèque, facilitant la récupération rapide de sources spécifiques.

Cependant, Zotero présente quelques inconvénients, notamment sa gestion parfois difficile de très grandes bibliothèques contenant des milliers de fichiers, ce qui peut nécessiter l'achat d'espace de stockage supplémentaire. Bien que performant, le logiciel requiert parfois la saisie manuelle d'informations que le connecteur intégré ne détecte pas automatiquement, représentant un potentiel défi pour les utilisateurs. Ces quelques défis soulignent l'importance d'évaluer les besoins spécifiques en matière de gestion bibliographique avant de choisir Zotero comme solution.

Zotero est souvent utilisé en combinaison avec BibLaTeX via l'extension Better BibTeX pour exporter et actualiser automatiquement des bibliographies au format .bib. BibLaTeX, une extension moderne pour gérer les bibliographies dans LaTeX et Quarto, s'utilise couramment avec Biber, un outil de traitement bibliographique avancé compatible avec BibLaTeX. Biber propose des fonctionnalités telles que le tri poussé, la gestion de multiples bibliographies et le traitement de divers formats de données bibliographiques. BibLaTeX, prenant en charge de nombreuses langues, est idéal pour la rédaction de documents destinés à un public international. L'exportation de bibliothèques Zotero sous forme de fichiers .bib pour leur utilisation avec BibLaTeX est simplifiée grâce à Better BibTeX, qui assure la mise à jour automatique de ces fichiers. Il est recommandé de maintenir dans votre fichier .bib uniquement les références utilisées, organisées par ordre alphabétique, afin de faciliter la collaboration et le partage des ressources.

4.6.4 Conclusion

Pour conclure, l'adoption de Zotero comme outil de gestion bibliographique se révèle être un choix judicieux pour tout chercheur soucieux de l'efficacité et de la rigueur dans le processus de documentation scientifique. Au-delà de la simple facilitation du travail de recherche en équipe, Zotero se distingue par sa capacité à optimiser la gestion des citations et des bibliographies, permettant ainsi une économie de temps considérable et une réduction des risques d'erreurs. Sa fonctionnalité de centralisation des sources et de leurs fichiers associés offre un avantage

notable en termes d'organisation et d'accès rapide à l'information, cruciale dans le cadre de recherches approfondies ou pluridisciplinaires. L'intégration de Zotero dans les environnements académiques, même en dehors des contextes de recherche, comme l'enregistrement des lectures pour des cours ou des séminaires, prépare efficacement les utilisateurs à des pratiques de recherche plus poussées et renforce la culture de la gestion rigoureuse des références. Cette initiation précoce est d'autant plus pertinente que Zotero se prête à une variété de styles de citation, répondant ainsi aux exigences diverses des publications académiques. Il est important de souligner que la maîtrise de Zotero, bien que facilitée par de nombreux tutoriels et ressources en ligne, représente un investissement en temps qui se trouve largement compensé par les bénéfices en termes d'efficacité et de qualité du travail de recherche. En outre, l'accès gratuit et le caractère open-source de Zotero témoignent de son engagement en faveur d'une diffusion élargie du savoir et d'une collaboration scientifique ouverte.

5 Outils de collecte de données

La révolution numérique engendrée par l'émergence du Big Data représente un important défi pour le monde des sciences sociales (Manovich, 2011; Burrows et Savage, 2014). Elle constitue également une opportunité de recherche enrichissante et innovante permettant une compréhension plus accrue des phénomènes sociaux étudiés par la communauté scientifique (Connelly et al., 2016). Cette meilleure compréhension est permise, entre autres, par l'accès à des données massives concernant les trois acteurs clés de la société démocratique: les citoyens, les médias et les décideurs (Schroeder, 2014; Kramer, 2014). Si l'accès à ces données représente un défi éthique et théorique, tel qu'explicité lors des chapitres précédents, elle représente également un défi technique pour les chercheurs.euses voulant exploiter le potentiel et les opportunités offertes par les données massives (Burrows et Savage, 2014).

Le chapitre qui suit vise à offrir un portrait des outils de collectes de données pouvant être exploitées par les scientifiques désirant entreprendre des recherches en sciences sociales numériques. Toutefois, la portée de ce chapitre s'étend plus loin que les outils traditionnels de collecte de données en abordant le potentiel émanant de la programmation en matière de collecte de données brutes pouvant être analysées par les chercheurs.ses. Les lignes suivantes mettront un accent particulier sur le web scraping et les opportunités de recherche qui en découlent. Ce chapitre offre donc un tour d'horizon des outils disponibles à la communauté scientifique tout en présentant les manières d'exploiter le web scraping et en offrant des exemples concrets de son utilisation.

5.1 Les outils traditionnels de collecte de données numériques en sciences sociales

Le champ d'étude de la science politique repose sur l'étude de trois types d'acteurs distincts ayant un impact sur la condition socio-économique et politique d'une société : les décideurs, les médias et les citoyens. La recherche sur les décideurs comprend entre autres l'analyse des politiques publiques, des partis politiques, de stratégies électorales ou encore l'analyse de discours de politiciens ou d'organisations. L'étude des médias repose largement sur le rôle des médias dans la formation des priorités et des jugements des citoyens quant aux enjeux politiques, de même que sur leur capacité d'influencer l'agenda des politiciens. En ce qui concerne les citoyens, le champ d'étude de l'opinion publique se consacre à l'analyse des comportements et des attitudes politiques des individus. De plus, de nombreuses recherches visant à comprendre

le rôle des citoyens dans une société démocratique portent sur l'influence de la société civile de même que sur l'effet des mouvements sociaux.

L'opinion publique est traditionnellement étudiée par le biais de données de sondages. L'émergence des technologies numériques a grandement transformé la collecte de données de sondages, qui sont désormais conceptualisés et administrés de manière beaucoup plus efficace. En effet, le numérique permet donc de créer un questionnaire, de cibler une population et de la contacter, d'entreposer les données des répondants pour ainsi les visualiser, le tout à un coût réduit et plus rapidement que s'il avait été conduit manuellement (Nayak & K. A., 2019). Ainsi, les sondages en ligne ont une portée internationale, permettent le suivi de la ligne du temps, offrent des options qui contraignent le répondant à répondre à certaines questions et permettent d'utiliser des arbres de logique avancés que les sondages manuels ne permettent pas. Parmi les plateformes web les plus reconnues de construction et d'Administration de sondage, Qualtrics figure en tête de liste. Cette plateforme est une des plus reconnues et utilisée à l'international, tant dans le milieu académique que dans le secteur privé. En plus d'offrir des outils de collecte de données et de sondages, Qualtrics est utilisé dans le marketing et dans la gestion de l'expérience client. Il est donc pertinent de se familiariser avec cet outil, car il offre des compétences pratiques pour la recherche, mais également pour obtenir des opportunités de carrière. Qualtrics offre plusieurs services pratiques pour la collecte de données, avec des options flexibles pour la programmation et l'administration des sondages. Par exemple, Qualtrics s'adapte à différents formats en fonction de l'appareil du répondant (Evans & Mathur, 2018). Son principal désavantage provient de son coût d'acquisition, qui est relativement dispendieux.

Au niveau des médias, l'arrivée de données massives permet de nouvelles avenues de recherche pour les chercheurs.euses en sciences sociales en raison de l'importante quantité de données accessibles aux chercheurs.euses, ce qui permet une compréhension accrue des réalités médiatiques modernes, marquées par la fragmentation. L'outil Factiva offre un accès à l'ensemble des articles d'une panoplie de médias provenant d'une vaste sélection de pays. Le moteur de recherche est opéré par Dow Jones et offre également l'accès à des documents d'entreprises. En revanche, l'accès qu'il offre aux contenus médiatiques est particulièrement pertinent pour la communauté scientifique en communication et en sciences sociales. Il offre l'accès à plus de 15 000 sources médiatiques provenant de 120 pays. Il permet de télécharger une quantité illimitée de documents RTF, un format de fichier de texte, pouvant contenir jusqu'à 100 articles chacun. En outre, ils peuvent être sélectionnés automatiquement en cochant le bouton proposant de sélectionner les 100 articles de la page de résultat. Chaque page de résultat contient 100 articles à la fois. Enfin, Factiva permet également de filtrer les doublons. Ainsi, Factiva permet d'avoir accès facilement à des données utiles pour l'analyse textuelle d'articles médiatiques. Comme les textes deviennent accessibles rapidement et simplement aux chercheurs.euses, cet outil optimise considérablement l'analyse de contenu par thèmes ou par ton. Cependant, ce ne sont pas tous les médias qui sont accessibles sur Factiva. Dans l'optique où un média recherché n'est pas trouvable sur Factiva, le logiciel Eureka représente une bonne alternative. Eureka se concentre principalement sur les médias francophones (autant au Québec qu'en Europe). La structure d'Eureka est similaire à celle de Factiva.

5.2

Recherche de texte libre

Recherche Guidée

Exemples

Date: Au cours des 3 derniers mois Doublons: Désactivé

Rechercher

Source: Toutes les sources

Auteur: Tous les auteurs

Société: Toutes les sociétés

Recherche des Experts de Factiva

Sujet: Tous les sujets

Secteur économique: Tous les secteurs économiques

Région: Toutes les régions

Chercher:

Langue: Toutes les langues

Plus d'options:

Effacer la recherche

Rechercher

Figure 5.1: image3_1

EUREKA

RECHERCHER DOSSIERS PUBLICATIONS PDF

Recherche avancée

Mots clés dans tout le texte

ET OU SANS

dans le titre

dans l'introduction

dans le nom de l'auteur

Ajouter une zone de mots clés

Sources

Sélectionnez vos sources par : ☒ groupe de sources ☐ critères de sources ☐ nom de source

Domaine de recherche

Tout le contenu

Période

Depuis 30 jours

Astuces de recherche

"pomme verte"

contient la phrase exacte « pomme verte »

blanc & noir

contient à la fois « blanc » et « noir »

rouge | vert

contient « rouge » ou « vert » ou les deux

pomme & (verte | rouge)

contient « pomme » ainsi que « verte » ou « rouge » ou les deux

bières ! *bières blondes*

contient « bières », mais pas « bières blondes »

voiture \$2 sport

contient « voiture » suivi de « sport » avec un maximum de deux mots d'écart

automobile %2 salon

contient « automobile » et « salon » (peu importe l'ordre) avec un maximum de deux mots d'écart

Recommencer

Recherche

Figure 5.2: image3_2

PIÈGE: NE PAS SE RESTREINDRE AUX OUTILS TRADITIONNELS DE RECHERCHE. Ces outils sont très utiles et relativement faciles à utiliser. Il ne faut toutefois pas tomber dans

le piège de se limiter aux outils traditionnels de recherche. En effet, les récentes transformations technologiques élargissent considérablement le champ de possibilités offertes à la communauté scientifique, notamment en raison de la nature massive des données qui lui est accessible. Non seulement ces données sont nombreuses, mais elles sont accessibles par le biais de connaissances de base en programmation. La section suivante aborde un outil fondamental de la collecte de données en sciences sociales numériques, les extracteurs webs.

5.3 Le web scraping - collecter automatiquement des données provenant de sites web

Chacun des acteurs démocratiques énumérés précédemment peut également être étudié par le biais d'extracteurs qui offrent un accès à des données numériques massives. Les extracteurs de données numériques sont des infrastructures de code permettant d'extraire des données brutes d'une source numérique définie. La section suivante explique comment ces extracteurs peuvent être utiles dans un contexte de recherche en sciences sociales numériques.

L'émergence du numérique représente une opportunité hors pair d'accès à un volume important de données, qui permettent ainsi une analyse approfondie des phénomènes politiques contemporains. Toutefois, l'accès à de telles données peut s'avérer complexe, non-fiable ou encore coûteux. Par exemple, des données parlementaires peuvent être accessibles sur les sites internet des parlements et institutions en questions. L'accès à ces données se voit toutefois complexifié par la nécessité d'avoir des identifiants ou encore de payer pour les dites données. De plus, la qualité de ces données n'est pas assurée, en plus du fait qu'elles peuvent être mal-structurées. Ainsi, l'accès à des données massives représente un défi considérable pour la communauté scientifique tentant d'entreprendre des recherches utilisant un volume important de données.

C'est dans cette optique que les extracteurs de données numériques peuvent être utiles. Plus précisément, le web scraping permet l'extraction de données provenant de sites webs qui seront ensuite converties dans un format utile aux scientifiques de données. Dans un contexte de recherche en science politique, le développement de scrapers permet de récolter automatiquement les données de sites internet pertinents qui pourront ensuite être utilisées afin de mener à terme un projet de recherche. Les sites desquels les données seront extraites dépendent du sujet de recherche d'intérêt de la personne entreprenant la recherche. Par exemple, un code peut extraire de manière automatisée les débats des parlements, les communiqués de presse des gouvernants, les plateformes électorales des partis politiques, ce qui offre un accès inégalé aux chercheurs.euses aux données de décideurs. De telles données pourraient mener à des analyses poussées sur le contenu et le ton des débats parlementaires. Dans une autre optique, des extracteurs peuvent également offrir l'accès aux données provenant de médias socionumériques comme Twitter (maintenant X) ou Facebook . Un extracteur peut, par exemple, être en mesure de répertorier l'ensemble des Tweets de journalistes, de politiciens ou encore de citoyens

de manière automatisée, offrant un accès inégalé aux chercheurs.euses à des données massives exclusives.

PIÈGE: LA LÉGALITÉ DES EXTRACTEURS DE DONNÉES

Il faut toutefois être vigilant quant à la nature des données extraites. Avant d'extraire quelque information, il est absolument primordial de s'assurer que les données soient publiques, faute de quoi l'extraction serait illégale. Il est donc recommandé de prendre connaissance des termes et conditions des sites webs étudiés afin de s'assurer de la légalité de l'extraction de données. Il est également important de respecter toute norme de droit d'auteurs et de propriété intellectuelles ou physique des données. De plus, ce n'est pas parce que des données sont publiques qu'il est nécessairement légal de les extraire et de les utiliser en recherche. En effet, il ne faut pas faire ressortir dans les données extraites quelque information privée qui pourrait permettre d'identifier des individus (comme des numéros de téléphone, des adresses courriels, des codes postaux, etc.)

RACCOURCI: LES API.

L'élaboration d'extracteurs est toutefois facilitée par l'existence d'API (Application programming interface) sur les plateformes exploitées. L'API d'un site ou d'une application, souvent fournie par le site, permet à un tiers d'avoir accès à du code expliquant le fonctionnement de la plateforme étudiée, ce qui en facilite l'extraction de données. Par exemple, Twitter possédait, avant les changements de directions récents, un API qui facilitait l'élaboration d'un extracteur. En contrepartie, Facebook ne possède pas d'API, ce qui rend l'accès à ses données beaucoup plus complexe. Un API fournit des données structurées dans un format lisible tel que JSON (JavaScript Object Notation). En raison de l'automatisation, les API réduisent les chances d'erreurs dans le processus de scraping, ils ont tendance à maintenir une interface plus stable et conviviale. C'est un grand changement comparé aux fichiers HTML, où ces derniers ont des mises en page qui changent fréquemment de structure.

Un extracteur peut également offrir l'accès à des données médiatiques, en codant un accès à des fils RSS ou encore aux HTML des médias extraits. Les fils RSS sont des formats de données qu'il est possible de recevoir automatiquement lors de mise à jour sur un site particulier. Par exemple, La Presse change ses Unes plusieurs fois dans une même journée. En extrayant l'accès aux fils RSS, il est possible de recevoir lesdites mises à jour automatiquement. Ce processus accélère grandement la collecte de données. Bien que les API aient souvent une limite de taux, restreignant le nombre de requêtes possible, cela aide à éviter la surcharge sur les serveurs et garantit en même temps une utilisation équitable des ressources.

Finalement, les API simplifient le travail de chacun et chacune par les mises à jour et maintenances. Pour être plus précis, les API, étant maintenus par les fournisseurs de services, sont modifiés au fur et à mesure que le site évolue. Par exemple, si la structure de X est modifiée, son API sera également modifiée. Cela évite à ceux demandant l'accès d'ajuster leurs scripts de scraping pour prendre en compte les changements sur le site.

En somme, l'utilisation d'extracteurs webs facilite grandement l'acquisition de données massives. Plutôt que d'avoir à payer pour des données dont la qualité n'est pas assurée, l'utilisation d'extracteurs permet un accès plus facile et précis à des données provenant de sites web. L'élaboration d'un extracteur est toutefois une tâche complexe qui requiert un certain nombre de connaissances en lien avec les langages de programmation. Le chapitre 2 du présent ouvrage offre un survol du langage fonctionnel R, qui est utilisé par de nombreux développeurs lors de l'écriture d'extracteurs. R est également reconnu pour ces fonctionnalités statistiques qui sont, elles aussi, abordées ultérieurement dans ce livre. L'utilisation d'extracteurs webs permettent aux chercheurs.euses de faire plusieurs coups d'une seule pierre. Non seulement le développement d'extracteurs permet de se familiariser avec le langage R, qui est un atout essentiel dans la recherche en sciences sociales numériques, mais ce même développement permet un accès inégalé à des données massives qui pourront ensuite être analysées. Le développement d'extracteurs est donc relativement simple dépendamment du site que l'on vise à extraire. Ainsi, des connaissances en R sont essentielles au développement d'extracteurs, et la complexité du code évoluera dépendamment du site qui sera extrait. La section suivante présente les outils nécessaires à l'entreprise d'extraction de données web sur R.

5.4 Extraire des données avec R

5.4.1 L'importance de comprendre la structure du code HTML

Afin d'être à l'aise avec les extracteurs web, il peut être un atout de se familiariser avec la structure de base du langage HTML, dont l'acronyme signifie "Hypertext Markup Language". Il s'agit d'un langage de code qui permet la description du contenu de la page web. La structure du code HTML est hiérarchique, ce qui signifie que le code est divisé en différentes sections qui occupent différents rôles. Ces sections sont délimitées par des "tags", qui définissent le début ou la fin d'une section du site. Ce sont ces différentes sections qui seront accessibles pour l'extraction, et le code HTML permet de comprendre ce qui se situe dans ces sections. La Figure [] représente un exemple de base de code HTML.

```
<!DOCTYPE html>
<html>
<body>

<h1>My First Heading</h1>

<p>My first paragraph.</p>

<h2>My Second Heading</h2>

<p>My second paragraph.</p>

<h2>My Third Heading</h2>

</body>
</html>
```

Figure 5.3: image3_3

Dans l'exemple ci-haut, le tag `<h1` représente le titre du HTML. Le signe `<p` permet de débiter le paragraphe de texte suivant le titre, et cette section devra être terminée par le sigle `p>`. Les sections `<p>` délimitent les paragraphes écrits dans chacune des sections. Les sigles `<h2>` produisent une sous-section et un sous titre, qui pourra être complété d'un paragraphe écrit. La Figure 2 démontre le texte produit par le code HTML.

My First Heading

My first paragraph.

My Second Heading

My second paragraph.

My Third Heading

Figure 5.4: image3_4

La structure de base du langage HTML est somme toute simple et intuitive et tous les sites web sur internet sont fondés sur du code HTML. N'importe qui étant intéressé à extraire des données de sites webs et tirer profit de la simplicité et l'accessibilité des données qui peuvent en émerger devraient donc se familiariser avec ce langage. De nombreuses sources en ligne sont disponibles afin d'apprendre sur le fonctionnement du code HTML. Nous vous encourageons donc fortement à explorer plus en profondeur les structures du code HTML afin d'obtenir une compréhension accrue du fonctionnement des sites webs que vous allez extraire. Comme le code HTML de chaque site est accessible grâce à l'URL, les données présentes sur des sites webs sont plus que jamais accessible à la communauté scientifique.

5.4.2 Le package rvest: son fonctionnement et ses possibilités

Cet ouvrage recommande l'utilisation du paquetage "Rvest" afin de récolter des données sur des pages web, qu'elles aient ou non un API. Rvest est construit autour des paquetages "xml2" et "httr" afin de faciliter la manipulation du HTML et XML. Rvest est principalement conçu pour scraper une seule page web alors que pour scraper de multiples pages, d'autres paquetages

sont recommandés, notamment "polite". Cet ouvrage ne rentre pas dans les détails et ne se concentrera que sur Rvest.

5.4.2.1 Fonctions de base du paquetage rvest utilisant pour exemple le site de LEGISinfo

La première étape est l'installation et le chargement des paquets "tidyverse" et "rvest" sur sa console Rstudio. Il est important de les charger séparément car RVEST ne fait pas partie des paquets de base du TIDYVERSE. Nous installerons ce dernier car il amène des fonctions pratiques au scraping

```
install.packages("rvest")

install.packages("tidyverse")

library(rvest)

library(tidyverse)
```

Pour débiter l'extraction des données, il suffit de copier l'URL de la page web à scraper et la coller dans l'appel de la fonction read_html(). Il est important de stocker l'URL dans l'objet "html_LEGISinfo.

```
html_LEGISinfo <- read_html("https://www.parl.ca/legisinfo/en/bills")

html_LEGISinfo
```

Lors de l'exécution des lignes de code ci-haut, la console retournera les éléments suivants:

```
{html_document}
<html lang="en" xml:lang="en" xmlns="http://www.w3.org/1999/xhtml">
[1] <head>\n<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">\n<me
[2] <body class="body-wrapper ce-parl vh-100">\r\n <header class="d-print-none"><!-- ...
```

Une fois que les éléments que l'on souhaite extraire sont déterminés, il faut les trouver dans le document HTML. Pour ce faire, il faut se référer au style CSS (cascading style sheets), langage définissant la forme visuelle d'un document HTML. Les éléments HTML sont identifiés avec des "sélecteurs CSS", ayant pour but de les regrouper pour faciliter leur extraction. Pour les bases du scraping, il n'est pas primordial de comprendre les détails des sélecteurs CSS. Seule la compréhension de la structure d'un document HTML est nécessaire afin d'en faire l'extraction d'éléments. L'important est d'être en mesure d'identifier les sélecteurs CSS liés aux éléments souhaités, sans avoir à comprendre le sélecteur en question.

D'abord, `html_elements` doit être utilisé en premier pour trouver toutes les observations souhaitées, car cette fonction retourne une liste de tous les noeuds qui matchent avec l'appel de fonction. Le nombre d'observations est indiqué par `xml_nodeset()`. Comme `html_element` retourne seulement le premier élément qui match, il faut l'utiliser en deuxième, après `html_elements`. Cette seconde fonction a pour but de trouver les éléments qui deviendront les variables à extraire. Pour l'exemple de LEGISinfo, nous commencerons par extraire tous les éléments `<a>`. Comme `html_elements` retourne une liste, nous voulons commencer avec cette fonction.

```
html_LEGISinfo |> html_elements("a")
```

Qui retourne les éléments suivants:

```
{xml_nodeset (180)}
[1] <a href="#StartOfContent" class="ce-parl-skipnav sr-only sr-only-focusable">Skip to m .
[2] <a href="//www.parl.ca" class="ce-parl-btn float-left text-nowrap">Parliament of Cana .
[3] <a href="https://visit.parl.ca/index-e.html" rel="external">\n\t\t
```

```
html_LEGISinfo |> html_elements("a")
```

Qui retourne les éléments suivants:

```
{xml_nodeset (180)}
[1] <a href="#StartOfContent" class="ce-parl-skipnav sr-only sr-only-focusable">Skip to m .
[2] <a href="//www.parl.ca" class="ce-parl-btn float-left text-nowrap">Parliament of Cana .
[3] <a href="https://visit.parl.ca/index-e.html" rel="external">\n\t\t
```

```
html_LEGISinfo |> html_elements("a")
```

```
{html_node}
<a href="#StartOfContent" class="ce-parl-skipnav sr-only sr-only-focusable">
```

Suite à l'inspection des éléments, ceux qui nous intéressent sont ceux de classe "bill-tile-container". Il suffit d'ajouter un point "." avant la classe souhaitée lors de l'appel de la fonction afin de rechercher les éléments en fonction de leur classe. Les classes HTML servent à catégoriser les éléments HTML selon un style prédéterminé. Pour l'exemple de LEGISinfo, nous obtenons une liste d'éléments de classe `bill-tile-container` que nous allons stocker dans l'objet `BillTile_LEGISinfo`. Tous les éléments de cette classe auront donc tous une structure ou des comportements similaires entre eux.


```
BillTile_LEGISinfo <- html_LEGISinfo |> html_elements(".bill-tile-container")
```

L'exécution du bloc de code ci-haut produit le résultat suivant dans la console

```
{xml_nodeset (60)}
[1] <a class="bill-tile-container senate" href="/legisinfo/en/bill/44-1/s-1">\r\n\r\n
[2] <a class="bill-tile-container senate" href="/legisinfo/en/bill/44-1/s-2">\r\n\r\n
[3] <a class="bill-tile-container senate" href="/legisinfo/en/bill/44-1/s-3">\r\n\r\n
```

À partir de la liste d'éléments de classe bill-tile-container, nous appelons la fonction `html_element`, qui lorsque appliqué à une liste permet d'extraire la première correspondance de tous les éléments de cette liste au lieu de seulement retourner le premier nœud correspondant du document HTML. Nous cherchons à extraire ici les éléments de classe `parliament-session` par le biais de la ligne de code suivante:

```
BillTile_LEGISinfo |> html_element(".parliament-session")
```

Ce qui produit le résultat suivant dans la console:

```
{xml_nodeset (60)}
[1] <div class="parliament-session">\n<span class="parl-session-number">44th</span> Parlia
[2] <div class="parliament-session">\n<span class="parl-session-number">44th</span> Parlia
[3] <div class="parliament-session">\n<span class="parl-session-number">44th</span> Parlia
```

Bien que moins utile pour la mise en situation actuelle, il est également possible d'extraire les éléments en fonction de leur "id attribute". Pour ce faire, il faut mettre un hashtag (#) avant l'élément à extraire lors de l'appel de la fonction. Le id attribute retourne toujours un seul élément car ils sont uniques à chaque document HTML. Voici la ligne de code et le résultat produit par une telle opération

```
html_LEGISinfo |> html_elements("#StartOfContent")
```

```
{xml_nodeset (1)}
[1] <a id="StartOfContent" tabindex="-1"></a>
```

Nous avons créé précédemment l'objet `BillTile_LEGISinfo` pour ensuite y extraire les éléments de classe `parliament-session`. Nous appelons cette étape l'imbrication des sélections. Lorsque la fonction `html_element` est appliquée à un vecteur de liste `html_elements`, la console retourne le premier nœud correspondant de chaque élément de la liste. Il est important d'utiliser

html_element à cette étape car il retourne un NA même lorsqu'il n'y a pas d'éléments correspondants, alors que html_elements ne retournera pas la valeur manquante. Dans l'exemple de LEGISinfo, c'est exactement ce que l'on a voulu faire pour obtenir les éléments de classe parliament-session. Nous sommes maintenant arrivés à l'étape d'extraire les données souhaitées. C'est assez simple, il ne suffit que d'appliquer la fonction html_text2 sur l'appel de html_element sur l'objet à moissonner, dans ce cas ci BillTile_LEGISinfo. Il est important de prendre en compte que nous connaissons ici les éléments à extraire, car le script du document a été scruté préalablement grâce à la fonction "inspect" de Google Chrome ainsi que les diverses fonctions du package rvest. Afin d'extraire les autres informations souhaitées, nous allons également créer deux autres objets qui seront à leur tour moissonnés. Voici les différentes opérations et leurs résultats dans la console:

Code

```
BillTile_LEGISinfo |> html_element("h4") |> html_text2()
```

Console

```
[1] "S-1" "S-2" "S-3" "S-4" "S-5"...
```

Code

```
BillTile_LEGISinfo |> html_element(".parliament-session") |> html_text2().
```

Console

```
[1] "44th Parliament, 1st session" "44th Parliament, 1st session"...
```

Code

```
BillTile_LEGISinfo |> html_element("h5") |> html_text2()
```

Console

```
[1] "An Act relating to railways"
[2] "An Act to amend the Parliament of Canada Act and to make consequential and related amendments"
[3] "An Act to amend the Judges Act"
...
```

Code

```
BillBS_LEGISinfo <- html_LEGISinfo |> html_elements(".bottom-section")
BillBS_LEGISinfo |> html_element("dd") |> html_text2()
```

Console

```
[1] "Introduced as pro forma bill"
[3] "Bill not proceeded with"
[5] "Royal assent received"
...
```

```
"Senate bill
"Royal assen
"At second r
```

Code

```
Bill_stage <- html_LEGISinfo |> html_elements(".progress-bar-description")
Bill_stage |> html_element("dd") |> html_text2()
```

Console

```
[1] "First reading in the Senate"      "Third reading in the Senate"      "First
[6] "First reading in the House of Commons" "First reading in the House of Commons" "Roya
...
```

Maintenant que nous avons tous les éléments souhaités, il ne reste plus qu'à utiliser la fonction `tibble` du `tidyverse`. Ce package permet de facilement créer des dataframes sur R. Voici le script à produire dans l'exemple de `LEGISinfo`, ainsi que son résultat, un dataframe contenant le numéro de projet de loi, sa session parlementaire, son nom, son statut et son dernier stage de réalisation :

```
Table_LEGISinfo <- tibble(
  Bill = Bill_LEGISinfo |> html_element("h4") |> html_text2(),
  Session = Bill_LEGISinfo |> html_element(".parliament-session") |> html_text2(),
  Name = Bill_LEGISinfo |> html_element("h5") |> html_text2(),
  Status = BillBS_LEGISinfo |> html_element("dd") |> html_text2(),
  Stage = Bill_stage |> html_element("dd") |> html_text2()
)
```

Il est important de noter que ce chapitre ne permet que de scraper des documents html uniques. Afin de scraper plusieurs page web simultanément, il faudra utiliser d'autres paquetages ainsi que des boucles, ce qui est trop complexe pour cet ouvrage d'introduction. Maintenant que vous savez extraire des informations d'un document html pour le mettre dans une base de données, voici d'autres applications pratiques de `rvest` à cet effet.

Il est possible d'extraire les éléments en fonction de leur attribut grâce à `html_attr()`. Un attribut est une information supplémentaire associée à une balise html. Voici l'attribut `href` qui permet d'extraire l'URL du projet de loi en question. De cette façon, il est possible de boucler sur les `href` afin de moissonner divers niveaux d'une page HTML. Lorsque l'extraction se fait sur plusieurs niveaux, la pratique passe du moissonnage pour devenir de l'indexation. Cette pratique, bien que fondamentale, ne sera pas abordée en raison de sa complexité avancée. Tel que mentionné plus haut, cet ouvrage ne se concentre que sur le moissonnage.

```
BillTile_LEGISinfo |> html_attr("href")
```

La ligne de code ci-haut produit le résultat suivant dans la console

```
1] "/legisinfo/en/bill/44-1/s-1"    "/legisinfo/en/bill/44-1/s-2"    "/legisinfo/en/bill/44-1/s-3"
[7] "/legisinfo/en/bill/44-1/s-7"    "/legisinfo/en/bill/44-1/s-8"    "/legisinfo/en/bill/44-1/s-9"
[13] "/legisinfo/en/bill/44-1/s-13"   "/legisinfo/en/bill/44-1/s-14"   "/legisinfo/en/bill/44-1/s-15"
```

Il est également possible d'extraire des tables. Pour cet exemple, le site de LEGISinfo ne comporte malheureusement pas de tables, le script de <https://r4ds.hadley.nz/web scraping> sera donc utilisé. Celui-ci utilise la fonction `minimal_html` pour créer un script html, qui n'est pas nécessaire au moissonnage, mais toutefois utilisé pour cet exemple.

```
htmltest <- minimal_html("
  <table class='mytable'>
    <tr><th>x</th>  <th>y</th></tr>
    <tr><td>1.5</td> <td>2.7</td></tr>
    <tr><td>4.9</td> <td>1.3</td></tr>
    <tr><td>7.2</td> <td>8.1</td></tr>
  </table>
")
htmltest |>
html_element(".mytable") |> html_table()
```

L'opération précédente produit le résultat suivant dans la console

```
# A tibble: 3 × 2
      x     y
  <dbl> <dbl>
1  1.5   2.7
2  4.9   1.3
3  7.2   8.1
```

En conclusion de cette section, lorsque l'on moissonne un document HTML, il est important de ne pas se laisser intimider par la structure du document. Il ne faut pas perdre patience afin de trouver les bons sélecteurs. Ce sont des structures peu familières au début, mais on s'y habitue rapidement. Ensuite, nous recommandons d'utiliser l'outil de développeur de votre navigateur web afin de pouvoir trouver les sélecteurs souhaités. L'interface de Chrome est particulièrement conviviale, et est recommandée. Il suffit de cliquer sur "inspect" suite à un clic droit, et il est possible de chercher les éléments souhaités dans le script. Finalement, avant de scraper le contenu d'un site web, il est important de vérifier s'il n'offre pas déjà une option pour télécharger les données ! C'est le cas de l'exemple utilisé ici (LEGISinfo), il est possible dans certains cas de télécharger les données directement sur le site web, ce qui rend parfois le besoin de moissonner désuet.

5.5 Conclusion et discussion:

Ce chapitre a comme objectif de dresser un portrait des différents outils de collecte de données mis à la disposition des scientifiques s'intéressant aux sciences sociales tout en voulant exploiter le potentiel de la révolution numérique. Bien que non exhaustif, ce chapitre fait un survol d'outils traditionnels de récolte de données numériques en sciences sociales. Par exemple, les outils de sondages ou de récolte de données médiatiques sont présentés dans les paragraphes ci-haut. En revanche, ce chapitre s'ancre autour du postulat qu'il ne faut pas se limiter aux outils de récolte de données traditionnels, et que la révolution numérique engendre d'importantes opportunités d'acquisition de données massives et exclusives par le biais des extracteurs de données. Ces extracteurs ont pour but de moissonner les données présentes sur un site web afin de les rendre disponibles pour l'analyse scientifique. Une section complète de ce chapitre vise à vulgariser le processus d'extraction de données provenant de site web en utilisant l'exemple du site LegisInfo, ce qui permet aux lecteurs.ices de se familiariser avec le processus de moissonnage de données.

Toutefois, un seul chapitre ne permet pas de relever l'ensemble des outils de collecte de données disponibles pour la communauté scientifique. Néanmoins, les outils présentés permettent un aperçu à la fois d'outils plus conventionnels et répandus de collecte de données numériques, mais également de dresser un portrait du potentiel d'extraction permis par la maîtrise de R.

Bibliographie:

Schroeder, R. (2014). Big data and the brave new world of social media research. *Big Data & Society*, 1(2), 2053951714563194.

Chadwick, A. (2017). The hybrid media system: Politics and power. Oxford University Press.

- Connelly, R., Playford, C. J., Gayle, V., & Dibben, C. (2016). The role of administrative data in the big data revolution in social science research. *Social science research*, 59, 1-12.
- Manovich, L. (2011). Trending: The promises and the challenges of big social data. *Debates in the digital humanities*, 2(1), 460-475.
- Burrows, R., & Savage, M. (2014). After the crisis? Big Data and the methodological challenges of empirical sociology. *Big data & society*, 1(1), 2053951714540280.
- Kramer, A. D., Guillory, J. E., & Hancock, J. T. (2014). Experimental evidence of massive-scale emotional contagion through social networks. *Proceedings of the National academy of Sciences of the United States of America*, 111(24), 8788.
- Andrade, C. (2020). The Limitations of Online Surveys. *Indian Journal of Psychological Medicine*, 42(6), 575-576. <https://doi.org/10.1177/0253717620957496>
- Evans, J. R., & Mathur, A. (2018). The value of online surveys: A look back and a look ahead. *Internet Research*, 28(4), 854-887. <https://doi.org/10.1108/IntR-03-2018-0089>
- Nayak, M., & K A, N. (2019). Strengths and Weakness of Online Surveys. 24, 31-38. <https://doi.org/10.9790/0837-2405053138>

6 Les outils de visualisation de données

7 Outils de visualisation graphique

7.1 Une image vaut mille mots

Camille Tremblay-Antoine¹ Nadjim Fréchet²

7.2 Introduction

Une fois les données collectées, nettoyées, traitées et analysées, une partie centrale du travail d'un scientifique de données est de faire parler les résultats de ses tests empiriques. Il s'agit alors de trouver la meilleure manière de rendre l'information digeste pour les experts et initiés de votre discipline académique ou pour le grand public. La visualisation graphique des données est donc centrale afin de vulgariser les résultats d'une recherche empirique.

Mais qu'est-ce qu'une bonne visualisation de données? Quel type de graphique choisir? Quelles couleurs utiliser? Quelles informations mettre en évidence? Ce chapitre ne répond pas à ces questions, une myriade d'ouvrages les ont déjà traitées. Du classique *The Visual Display of Quantitative Information* de Edward Tufte (1983) jusqu'aux plus récents ouvrages tels que *Data Visualization: A Practical Introduction* de Kieran Healy (2018) ou *Fundamentals of Data Visualization* de Claus O. Wilke (2019), les ressources sont nombreuses pour vous aider à améliorer vos compétences en visualisation de données. Il en ressort souvent un adage qui revient sous différentes versions: excellence et intégrité (Tufte, 1983); “Be Brief, Clear, Picturesque, and Accurate” pour Bessler (2023); “accuracy, utility, and efficiency” pour (zhuMeasuringEffectiveData2007?) , “Intégrité, Simplicité, Contexte, Esthétique” pour Arel-Bundock (2021). En somme, bien qu'il n'existe pas de solution toute faite, il est largement reconnu que l'adaptation de la visualisation selon l'objectif et les données à communiquer est cruciale. Il faut équilibrer soigneusement ces éléments.

Ce chapitre se concentre plutôt à faire une sommaire recension des outils de visualisation nécessaires aux personnes s'intéressant à la recherche en sciences sociales. Une première section discute de la sélection des outils. Ensuite, ceux-ci sont présentés selon trois catégories: les outils pour les diagrammes, les outils pour les analyses descriptives et les outils pour visualiser les régressions. Une dernière section ouvre une réflexion sur les visualisations réactives.

¹Université Laval

²Université de Montréal

7.3 Sélection des outils: débat R et Python

Il existe plusieurs outils de visualisation qui répondent à des besoins différents. Nous nous concentrerons sur les outils respectant au mieux les critères de sélections établis au chapitre 1.

Bien entendu, les logiciels tels que *Tableau*, *Stata*, *SPSS*, *SAS* ou encore *Excel* peuvent s'avérer très pertinents selon vos exigences spécifiques. Ils sont souvent dotés d'une interface utilisateur intuitive, facilitant ainsi leur utilisation pour une variété de tâches. Toutefois, ils pourraient présenter certaines limites en matière de personnalisation des analyses et des visualisations. Par ailleurs, bien que certains de ces outils offrent une grande flexibilité, leur coût peut être considérable. Si votre institution possède une licence pour ces logiciels, il demeure judicieux de les utiliser.

Il existe des outils gratuits et offrant un plus grand contrôle et offre plus de flexibilité que les logiciels de visualisation de données. Les logiciels. Programmation possible de personnaliser les graphiques à l'infini.

Bien que ce livre prend position en faveur de *R* comme présenté au chapitre 2, il est important de reconnaître les capacités de *Python* dans le domaine de la visualisation graphique. *Python* est un langage de programmation généraliste et est répandu dans la majorité des universités et sur le marché du travail (**ozgurMatLabVsPython2017?**). Matplotlib, Seaborn et Plotly sont des *packages* de

R est spécialisé en statistiques et scientific research and academia, analytical power of R is virtually unmatched.(**ozgurMatLabVsPython2017?**).

<https://www.r-project.org/about.html>

7.4 Outils pour les schémas

Il peut être nécessaire au cours d'un processus scientifique, une présentation, autre de faire des schémas.

Toujours pertinent de faire un croquis à la main, mais lorsque vient le temps de le rendre propre, présentable quels outils s'offrent à nous?

Moody, D. (2007). What Makes a Good Diagram? Improving the Cognitive Effectiveness of Diagrams in IS Development. In W. Wojtkowski, W. G. Wojtkowski, J. Zupancic, G. Magyar, & G. Knapp (Eds.), *Advances in Information Systems Development* (pp. 481–492). Springer US.

Larkin, J. H., & Simon, H. A. (1987). Why a Diagram is (Sometimes) Worth Ten Thousand Words. *Cognitive Science*, 11(1), 65–100.

Suttorp, M. M., Siegerink, B., Jager, K. J., Zoccali, C., & Dekker, F. W. (2015). Graphical presentation of confounding in directed acyclic graphs. *Nephrology Dialysis Transplantation*, 30(9), 1418–1423.

7.4.1 Diagrams.net (anciennement Draw.io)

the best free diagram and flowchart app ### Lucidchart

7.4.2 Miro

7.5 Outils pour les analyses descriptives

7.5.1 R

Lorsque vous souhaitez créer des graphiques en R, les options abondent. De multiples *packages* ont été développés dans le but de visualiser des données. Heureusement, les choix diminuent lorsque l'on regarde ce qui est le plus utilisé dans la communauté. L'objectif n'est pas simplement de présenter les *packages* les plus courants parce qu'ils sont les plus communs. Les *packages* les plus utilisés représentent des outils qui ont été grandement vérifiés et améliorés par la communauté en ligne, dont la documentation est abondante et pour lesquels les ressources d'aide en ligne sont innombrables.

Trois options vous sont présentées: Base R, Lattice et ggplot2. Les avantages et inconvénients respectifs de ces trois approches pour la création de graphiques sont explicités dans les sections suivantes.

7.5.1.1 Base R

Le *Base R* est le langage de base de R et il permet de faire de nombreuses manipulations statistiques sans avoir à installer de *packages* au préalable. Le *Base R* permet notamment de produire des graphiques rapidement. Cela peut être utile pour visualiser la distribution d'une variable ou pour regarder la relation entre deux d'entre elles, par exemple. Pour produire un graphique avec le langage de base R, il suffit de faire appel à la fonction *plot()*. Avec la fonction *plot()*, le codeur peut visualiser la distribution d'une variable seule en spécifiant l'axe des *x* dans cette dernière. Le codeur peut également visualiser la relation entre deux variables en spécifiant à l'intérieur de la fonction celles qui composeront les axes des *x* et des *y* du graphique. Les fonctions *barplot()*, *hist()* ou *boxplot()* disponibles dans le *Base R* permettent de spécifier le style de graphique souhaité, qu'on veuille représenter nos données sous forme de diagramme à barre, d'histogramme ou de diagramme en boîtes (Kabacoff, 2022, p. 119-132).

Alors qu'un peu tout peut être fait avec le *Base R*, ce langage demeure élémentaire; il est difficile d'innover dans la visualisation ou même de produire des graphiques plus sophistiqués. Le *Base R* peut sembler plus simple pour l'exploration de données ou pour produire des graphiques de base rapidement, mais ce langage devient rapidement complexe lorsqu'on cherche à améliorer l'esthétique de son graphique ou à visualiser des relations entre plusieurs variables, ce que *lattice* et *ggplot2* permettent plus facilement (Wickham, 2009, p. 3-4).

7.5.1.2 Lattice

Développé par Deepayan Sarkar, *lattice* cherche à faciliter la visualisation de graphique en facettes. Plus précisément, ce package vise à améliorer les graphiques du Base R en fournissant de meilleures options de graphisme par défaut pour visualiser des relations multivariées. Ce package est donc intéressant pour les chercheurs et les codeurs voulant présenter graphiquement la relation entre plus de deux variables (Kabacoff, 2022, p. 373-377; Sarkar, 2008, 2023). Pour produire un graphique de base avec *Lattice*, le package *lattice* doit préalablement être installé dans la bibliothèque de packages du codeur et chargé dans sa session au début de son code (voir annexe). Par la suite, le codeur doit spécifier le type de graphique souhaité avec la fonction appropriée³. Une fois la fonction choisie, il doit spécifier par une formule les variables *x* et *y* ainsi que la troisième variable à contrôler et à visualiser en facettes (`graph_type(formula | variable en facettes, data=)`).

Cependant, le package *lattice* a pour désavantage d'avoir un modèle formel (une grammaire de graphique) moins compréhensible et intuitif que celui de *ggplot2* lorsque vient le temps d'améliorer l'esthétisme des graphiques. De plus, sa plus faible popularité fait en sorte que ce package demeure moins développé par la communauté de codeurs de R que ne l'est *ggplot2*. Nous examinons plus en détail la grammaire de graphique de ce dernier package ainsi que ses avantages et inconvénients dans la prochaine section (Kabacoff, 2022, p. 373-377 et 390; Wickham, 2009, p. 6).

7.5.1.3 Ggplot 2

Développé principalement par Hadley Wickham, *ggplot2* est un *package R* faisant partie de la collection de *packages* de *tidyverse*. Ainsi, *Ggplot2* peut être utilisé avec les autres *packages* centraux de *tidyverse* ce qui limite de potentiels conflits entre les fonctions de *packages* qui puissent être incompatibles avec *ggplot2*. Par exemple, le *package dplyr* de *tidyverse* est très utile pour analyser, organiser et préparer vos données à visualiser avec *ggplot2* (Wickham et al., 2019; Wickham et al., 2023, p. 30).

Le principal avantage de *ggplot2* reste sa grammaire qui permet à l'utilisateur de rendre ses graphiques beaucoup plus visuellement attrayants en facilitant la personnalisation esthétique. Ceci permet de pousser l'esthétisme de vos graphiques à un très haut niveau par rapport aux autres *packages* de visualisation graphique disponibles en R. Les graphiques *ggplot2* se

construisent couche par couche, soit par l'ajout des différents éléments du graphique au fur et à mesure dans le code du graphique à construire.

7.6 Outils pour visualiser les régressions

7.6.1 modelsummary

(Arel-Bundock, 2022)

7.6.2 Stargazer

7.6.3 Ggplot2 et marginal effect

7.6.4 Aller plus loin: La visualisation interactive des données

Si jusqu'à présent la visualisation des données a été présentée comme une étape permettant de présenter les résultats de recherches, il est également possible de considérer la visualisation comme une utile au processus d'exploration des données comportants de nombreuses dimensions (autres façons de le dire peut-être?). En effet, les formes de visualisations dites interactives permettant d'explorer et même d'analyser les données à même notre graphique ou notre tableau. Cela contribue à mieux comprendre la structures des données, à inspecter plus rapidement ces dernières et même susciter des questions de recherches peut-être omises autrement (citer Sievert, 2020).

- ggplotly et plotly
- Tableaux interactifs? fonctions kable() et kableExtra du package knitr
- Shiny Apps

8 Langages de balisage

8.1 Baliser les sciences sociales : langages et pratiques

Lorsque vous lisez un article scientifique, une page Web ou un curriculum vitæ professionnel, vous vous doutez peut-être que le texte n'est pas toujours produit à l'aide d'un logiciel de traitement de texte comme Microsoft Word, Apple Pages ou LibreOffice Writer. La mise en page complexe réglée au millimètre près, la qualité des figures et des tableaux, l'utilisation de gabarits professionnels, le style des références ou encore la présence d'éléments interactifs sont difficiles et parfois impossibles à reproduire à l'aide d'un logiciel de traitement de texte régulier. L'ajout d'extraits de code, de tableaux de régression ou encore de figures de haute qualité graphique, ainsi que leur personnalisation, nécessitent une interface particulière.

Pour ces raisons et plusieurs autres, les chercheurs en sciences sociales font souvent appel aux langages de balisage, ou *markup languages*. Ceux-ci permettent de produire des documents et pages Web sans les limitations des logiciels de traitement de texte. Le présent livre, par exemple, est écrit à l'aide du langage de balisage Markdown avec l'aide du système de publication Quarto. Les logiciels de traitement de texte et les langages de balisage font tous partie de la catégorie des outils de rédaction. D'entrée de jeu, vous vous demandez peut-être quelle est l'utilité d'apprendre des langages de balisage alors que les logiciels de traitement de texte sont nombreux, simples d'approche et en amélioration constante. Ce chapitre n'a pas pour objectif de décourager l'utilisation de ces logiciels, qui sont utiles et même souvent essentiels pour la production rapide de documents ainsi que pour des tâches de suivi des modifications et de travail avec des équipes multidisciplinaires. Le chapitre tentera plutôt de démontrer que la maîtrise des langages de balisage constitue un avantage pour ceux qui souhaitent s'initier au monde de la recherche académique, même si quelques difficultés initiales d'apprentissage peuvent se présenter. Il s'agira de répondre, tour à tour, aux trois grandes questions suivantes : *Qu'est-ce qu'un langage de balisage? Quand et pourquoi utiliser un langage de balisage? Comment utiliser un langage de balisage?* L'accent sera mis sur Quarto ainsi que sur les langages Markdown et LaTeX, bien que d'autres langages soient aussi abordés.

8.2 Qu'est-ce qu'un langage de balisage?

Un langage de balisage constitue un ensemble de commandes qui peuvent être entremêlées à du texte afin de produire une action informatique. Chaque langage contient son propre ensemble

de commandes cohérentes et complémentaires. De manière plus formelle, ces commandes sont nommées *balises* (*tags* en anglais) et inscrites par le chercheur ou la chercheuse au travers du texte. Les balises constituent une manière de communiquer avec le logiciel utilisé dans un langage qu'il peut comprendre. Par exemple, une balise permet d'indiquer au logiciel que vous désirez qu'une section du texte soit écrite en caractères gras, en italique, à double interligne ou encore que vous souhaitez positionner une image d'une certaine manière au travers du texte. Cette interaction est rendue possible par la standardisation des langages de balisage : chaque balise correspond à une action précise, peu importe le logiciel utilisé, la langue dans laquelle le texte est rédigé, le type d'ordinateur utilisé, etc. Dans votre document source, les balises sont entremêlées au contenu de votre document. Au moment de compiler ce dernier, les balises produisent les actions informatisées qu'elles commandent et laissent comme document final le contenu mis en page tel que vous l'avez défini via les balises utilisées. La compilation est le processus par lequel un document écrit en langage de balisage est transformé en fichier textuel, en format PDF dans le cas de LaTeX par exemple. La Figure ?? montre un exemple d'utilisation du langage de balisage Markdown dans un fichier Quarto sur la plateforme Visual Studio Code. L'écran à droite de l'image montre le fichier PDF résultant du formatage réalisé dans la partie centrale de l'écran. Les balises utilisées sont décrites plus tard dans ce chapitre.

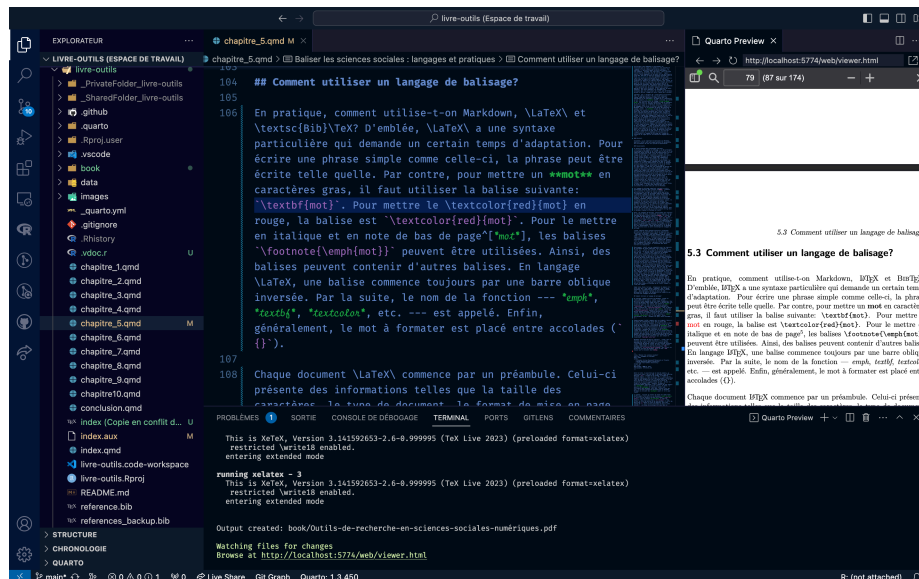


Figure 8.1: Exemple d'utilisation du langage de balisage Markdown dans un fichier Quarto sur la plateforme VS Code

Source : Auteurs du présent chapitre.

Le premier langage de balisage, le Generalized Markup Language (GML), a été inventé en 1969 par les chercheurs Charles F. Goldfarb, Ed Mosher et Ray Lorie pour la compagnie IBM. Goldfarb et ses collègues devaient intégrer trois applications créées avec des langages différents et avec une logique différente pour les besoins d'un bureau de droit. Même après

avoir créé un programme qui permettait aux trois applications d’interagir, ces langages demeuraient différents et avaient chacun leur propre fonctionnement. Le développement de GML a permis de résoudre ce problème en standardisant et en structurant le langage : les mêmes commandes étaient utilisées pour accomplir les mêmes tâches dans chaque programme (**goldfarbRootsSGMLPersonal1996?**). GML a été amélioré durant les décennies suivantes et a été suivi par d’autres langages de balisage, dont LaTeX (1985), BibTeX (1988), HTML (1993), XML (1998), Markdown (2004) et R Markdown (2012) (Extensible Markup Language (XML) 1.0, 1998, Getting Started, 2023, HTML History | Explained, 2023, LaTeX, 2023, Markdown, 2023).

Les langages de balisage permettent d’effectuer différentes tâches. HTML, qui est sans doute le plus connu des langages de balisage, permet de formater des sites Web. XML, quant à lui, permet de structurer de larges volumes de données. LaTeX permet pour sa part de formater du texte et de créer des documents en format PDF. Markdown permet également de créer des documents en format PDF, mais aussi en format HTML ou DOCX — format utilisé pour les documents Word —, contrairement à LaTeX. R Markdown permet d’ajouter des extraits de code R à un fichier en langage Markdown. Enfin, depuis 2022, le système de publication scientifique et technique multilingue Quarto permet de créer des documents qui intègrent des extraits de code R, LaTeX, Python, Julia ou JavaScript, créés dans différents types d’environnements, à un fichier en langage Markdown (**allaire22?**). LaTeX, Markdown, R Markdown et Quarto permettent aussi d’intégrer les références bibliographiques du système de traitement de références BibTeX. Les langages de balisage communiquent ainsi souvent les uns avec les autres au sein d’un même fichier. Le chapitre 6 explique la manière de citer les références en langage BibTeX par le biais de Zotero et de Better BibTeX.

Les balises constituent une manière de donner manuellement des commandes au logiciel que vous utilisez. Si vous utilisez Microsoft Word, vous avez accès à une panoplie de boutons qui vous permettent de formater votre texte. Les balises exercent les mêmes fonctions de formatage pour les fichiers produits en LaTeX ou en Markdown, mais doivent être ajoutées à l’écrit par l’utilisateur. Lorsque vous appuyez sur un bouton ou utilisez une commande comme **Ctrl-G** ou **Cmd-I** dans Word, en réalité, cette commande ajoute des balises au travers de votre texte, mais rend celles-ci invisibles dans l’interface que vous utilisez. Cela permet d’avoir un texte élégant et facile à lire, mais comporte aussi plusieurs inconvénients. Le principal inconvénient est de limiter le pouvoir que vous avez sur le formatage de votre texte. En effet, si les boutons à votre disposition ne vous permettent pas de réaliser une opération, celle-ci sera éternellement impossible à réaliser pour vous. A contrario, les langages de balisage permettent un contrôle presque infini sur les opérations que vous souhaitez réaliser. Incidemment, dans la mesure où vous utilisez le langage approprié pour la tâche que vous souhaitez accomplir, vous devriez être capable de donner exactement la commande nécessaire à votre logiciel. Les langages de balisage, bien qu’ils aient un coût d’apprentissage qui peut s’avérer important et que l’interface de travail soit moins intuitive qu’un document Word, vous offrent une plus grande flexibilité.

Afin d’utiliser un langage de balisage, il est impératif que le logiciel que vous utilisez puisse prendre en compte ce langage. Un logiciel permet rarement d’utiliser n’importe quel langage.

Par exemple, le logiciel `TeXShop` permet seulement d'utiliser le langage `LaTeX`. Il est aussi impératif de bien utiliser le langage de balisage. En effet, comme pour les langages de programmation, les langages de balisage ne peuvent pas déduire ce que vous souhaitez leur faire comprendre. Si vous souhaitez mettre du texte en gras, vous devez utiliser les bonnes balises. La moindre erreur peut être coûteuse, puisqu'une erreur dans la balise que vous utilisez risque de produire une commande incompréhensible et un message d'erreur, le logiciel ne réussissant pas à associer votre balise mal inscrite à une action informatisée. Conséquemment, il est impératif de bien vérifier les balises utilisées afin d'éviter toute erreur qui empêcherait votre document d'être compilé, c'est-à-dire d'être traduit dans son format final¹. Chaque caractère dans une balise est important et il y a rarement plus d'une seule manière de commander une action. Par exemple, en `LaTeX`, il n'y a qu'une seule manière de mettre du texte en gras. Il faut précisément utiliser cette commande: `\textbf{}`. Le positionnement des balises est lui aussi critique : il délimite la portion de texte à laquelle doit être appliquée l'action commandée par la balise.

Il est important de distinguer les langages de balisage des langages de programmation, qui sont abordés plus en détail dans le chapitre 4. En effet, ceux-ci sont similaires à certains égards, mais ont des vocations différentes. Les deux s'appuient sur un langage informatisé, mais les langages et leurs objectifs diffèrent. Un langage de programmation définit des processus informatisés alors qu'un langage de balisage permet d'encoder du contenu de manière à ce que celui-ci soit lisible tant pour l'humain que pour son ordinateur.

Dans le contexte de la recherche en sciences sociales, la programmation est généralement utilisée afin de récolter, d'analyser et de présenter visuellement des données. Une fois cartes, tableaux et graphiques produits, ceux-ci peuvent être enregistrés — par exemple en format PDF ou PNG — et inclus au sein d'un document qui sera formaté en utilisant un langage de balisage. En `R Markdown` et en `Quarto`, des extraits de langage de programmation peuvent être inclus dans des sections bien délimitées de documents écrits en langage de balisage. Plus généralement, le langage de programmation contribue à l'analyse alors que le langage de balisage est essentiellement utile afin de présenter les travaux de recherche, que ce soit dans un document écrit ou sur un site Web. C'est principalement de cette manière que sont utilisés les langages de programmation et de balisage dans le cadre de la recherche en sciences sociales.

8.3 Comment utiliser un langage de balisage?

En pratique, comment utilise-t-on `Markdown`, `LaTeX` et `BIBTeX`? D'emblée, `LaTeX` a une syntaxe particulière qui demande un certain temps d'adaptation. Pour écrire une phrase simple comme celle-ci, la phrase peut être écrite telle quelle. Par contre, pour mettre un **mot** en caractères gras, il faut utiliser la balise suivante: `\textbf{mot}`. Pour mettre le **mot** en rouge,

¹Les logiciels permettent plus ou moins efficacement d'identifier les balises problématiques. Certains ne produisent qu'un message d'erreur sans donner d'indication sur la source du problème, alors que d'autres ciblent très spécifiquement la ligne de syntaxe où se situe la balise problématique.

la balise est `\textcolor{red}{mot}`. Pour le mettre en italique et en note de bas de page², les balises `\footnote{\emph{mot}}` peuvent être utilisées. Ainsi, des balises peuvent contenir d'autres balises. En langage LaTeX, une balise commence toujours par une barre oblique inversée. Par la suite, le nom de la fonction — *emph*, *textbf*, *textcolor*, etc. — est appelé. Enfin, généralement, le mot à formater est placé entre accolades (`{}`).

Chaque document LaTeX commence par un préambule. Celui-ci présente des informations telles que la taille des caractères, le type de document, le format de mise en page, la police de caractères, l'utilisation d'en-têtes et de pieds de page, ainsi que l'utilisation de *packages* LaTeX permettant différentes fonctionnalités de personnalisation du document. Il n'est pas nécessaire ni souhaitable d'apprendre l'ensemble des fonctions et des *packages* LaTeX qui existent. Au contraire, il est souvent mieux de commencer par un gabarit de document qui convient au type de document que vous voulez créer et ensuite de rechercher en anglais sur Stack Overflow la manière d'ajouter des éléments de formatage que vous ne connaissez pas, par exemple en recherchant `highlight latex text`.

Markdown fonctionne de manière similaire à LaTeX, mais se démarque par sa plus grande flexibilité et sa syntaxe beaucoup plus légère. Par contre, il nécessite parfois l'utilisation de balises LaTeX afin de réaliser certaines tâches, comme changer la couleur du texte. Tout document Markdown débute avec un court bloc de syntaxe YAML (acronyme de **Yet Another Markup Language**) qui définit les paramètres généraux du document. Voici un bloc YAML typique pour un document Quarto :

```
---
title: "Baliser les sciences sociales"
subtitle: "Langages et pratiques"
date: today
author:
  - Alexandre Fortier-Chouinard^[University of Toronto]
  - Étienne Proulx^[McGill University]
  - Maxime Blanchard^[McGill University]
format: pdf
toc: true
date-format: "MMMM D, YYYY"
bibliography: livre-outils.bib
---
```

Outre le titre, le sous-titre et le nom des auteurs, on trouve aussi dans l'en-tête YAML la présence d'une table des matières (`toc`), la date et son format, le format du document compilé — dans ce cas-ci, PDF — ainsi que le chemin d'arborescence afin d'accéder au document BibTeX où sont enregistrées les références utilisées. Il est aussi possible d'y définir la taille de

²*mot*

la police de caractères ou encore le gabarit Word servant à définir le format d'un document DOCX à produire. De manière particulièrement importante, c'est l'endroit où sont chargés les *packages* LaTeX qui seront utilisés. En effet, la majorité des *packages* et fonctions LaTeX sont utilisables dans Markdown, alors que l'inverse n'est pas vrai. Il est donc possible de personnaliser un document Markdown en utilisant des *packages* ayant été créés pour LaTeX.

La syntaxe à utiliser au travers du texte est somme toute plutôt simple. Pour mettre un ou plusieurs **mots en gras**, il suffit de les entourer de deux astérisques (****mots en gras****); pour les mettre *en italique*, il faut les encadrer d'une seule astérisque (**en italique**). Pour définir un titre de section ou de sous-section, il suffit de mettre des # devant le titre en question. Plus vous ajoutez de #, plus le titre sera petit et plus il sera considéré à un niveau hiérarchique inférieur dans la structure du texte. La syntaxe Markdown est donc plus légère que celle de LaTeX, dans le but d'en rendre la lecture plus simple pour les utilisateurs et utilisatrices.

Bien que des gabarits Markdown soient disponibles, ceux-ci sont plus rares. Ils se trouvent pour la plupart sur GitHub et sont rendus disponibles par leur créateur. Cela étant dit, leur personnalisation peut s'avérer plutôt complexe. En somme, Markdown est particulièrement pratique pour les documents ne nécessitant pas de respecter un gabarit précis et requérant simplement un document d'allure simple et professionnelle.

Pour sa part, BibTeX a une syntaxe relativement simple. D'emblée, les références BibTeX pour des articles et ouvrages scientifiques sont disponibles sur Google Scholar. Toutefois, pour citer des sites Web ou des articles de médias, la référence doit être écrite à la main selon un format précis. Une bibliographie sur BibTeX peut ressembler à ceci :

```
@book{darwin03,
  address = {London},
  author = {Darwin, Charles},
  publisher = {John Murray},
  title = {{On the Origin of Species by Means of Natural Selection
or the Preservation of Favoured Races in the Struggle for Life}},
  year = {1859}
}
```

```
@article{goldfarb96,
  title={The Roots of SGML: A Personal Recollection},
  author={Goldfarb, Charles F},
  journal={Technical communication},
  volume={46},
  number={1},
  pages={75},
  year={1999},
  publisher={Society for Technical Communication}
}
```

Un fichier BibTeX ne contient rien de plus qu’une série de publications commençant chacune par la balise @ suivie du type d’article — *article*, *book* pour un livre, *incollection* pour un chapitre de livre, *inproceedings* pour une présentation dans une conférence, *unpublished* pour un article non publié et *online* pour un site Web sont parmi les plus connus — et des informations sur la publication mises entre accolades. La première information entre accolades est le code de la référence, par exemple `goldfarb96`. Dans le fichier LaTeX, l’auteur doit écrire `\cite{goldfarb96}` pour voir dans le document PDF compilé (**goldfarbRootsSGMLPersonal1996?**); le lien est automatiquement cliquable et renvoie à la notice bibliographique correspondante. L’ordre des publications dans le document BibTeX a peu d’importance, puisque LaTeX réordonne par défaut la bibliographie en ordre alphabétique.

8.3.1 Environnements d’édition et de compilation

Contrairement à Microsoft Word et Apple Pages, il existe plusieurs options d’environnements d’édition et de compilation spécifiques à chaque langage. Ces environnements sont des plateformes et des logiciels conçus pour faciliter l’édition, la mise en forme et la compilation de documents dans des langages de balisage tels que LaTeX et Markdown. Ils permettent également de rendre plus efficace et conviviale la production de documents tout en fournissant des fonctionnalités spécifiques aux besoins de chaque langage. Il existe une grande diversité d’environnements d’édition et de compilation, et le choix est libre pour la chercheuse ou le chercheur de trouver celui qui convient le mieux à ses besoins ou aux besoins de son groupe de recherche. Les trois options discutées ici sont parmi les plus utilisées par les chercheurs en sciences sociales et peuvent être regroupées en deux catégories : les logiciels de bureau et les éditeurs en ligne.

D’abord, il existe plusieurs logiciels de bureau qui offrent un environnement d’édition et/ou de compilation pour les langages de balisage. Ces logiciels fournissent les programmes principaux, les extensions essentielles et des outils complémentaires de compilation et de visualisation afin de permettre la production de documents écrits en langages de balisage. Le logiciel RStudio, également abordé dans le chapitre 4, permet de produire des documents avec différents langages de balisage et programmation, ainsi que de naviguer entre eux, à partir d’une même fenêtre. Il suffit d’installer certains *packages* contenant les fichiers nécessaires à l’utilisation des langages de balisage. Par exemple, il est possible de produire des documents en LaTeX en utilisant le code suivant dans la console pour installer le *package* nécessaire à l’utilisation de la distribution LaTeX TinyTeX : `install.packages("tinytex")`. Suivant le même principe, il est possible de produire des documents en R Markdown sur RStudio en installant le *package* suivant : `install.packages("rmarkdown")`. Pour Quarto, le téléchargement se fait en ligne, directement à partir du site Web de (Get Started, 2023).

Pour l’écriture en LaTeX, il est également nécessaire d’installer l’une des nombreuses distributions en ligne afin de pouvoir compiler ces documents dans un environnement local. Il existe des distributions telles que MacTeX pour Mac, MikTeX pour Windows et plusieurs autres

(Just, 2013). Ces distributions se distinguent par les différents *packages* avec lesquelles elles sont compatibles.

Un autre environnement régulièrement utilisé pour travailler en langage de balisage est le logiciel de bureau VS Code. VS Code prend en compte un plus grand nombre de langages de programmation et est utilisé par les programmeurs de tous domaines, tandis qu’RStudio est surtout utile pour les chercheurs en sciences sociales qui travaillent surtout en R.

Lorsque vient le temps de collaborer à plusieurs sur un document écrit en Markdown ou en LaTeX, les logiciels de bureau évoqués précédemment nécessitent l’utilisation de GitHub et de Git. L’utilisation de ces éditeurs peut présenter un défi supplémentaire pour les équipes de recherche non initiées. Il existe ainsi des éditeurs en ligne qui permettent de collaborer en temps réel sans passer par Git et GitHub, de manière similaire à Google Docs³. Le plus connu de ces logiciels est Overleaf, qui permet de produire des documents en langage LaTeX. Puisqu’Overleaf permet d’avoir accès à ses documents LaTeX à partir de n’importe quel navigateur, il n’y a pas de dépendance à un logiciel local sur un ordinateur, ce qui constitue un avantage important. La contrepartie de cet avantage est qu’en utilisant Overleaf, l’équipe de recherche est dépendante d’une connexion à Internet. En utilisant le package LaTeX `rmarkdown`, Overleaf peut également inclure du code Markdown. Cependant, Overleaf ne permet pas de créer des documents en format DOCX ou HTML, ce qui constitue une limite de l’application. Overleaf comporte un compteur de mots intégré, ce qui n’est pas le cas des autres logiciels et environnements présentés plus haut.

8.4 Quand et pourquoi utiliser un langage de balisage?

La plupart des langages de balisage permettent de remplir l’une des deux fonctions suivantes, qui sont particulièrement importantes dans le contexte de la recherche en sciences sociales : produire des documents écrits et formater des pages Web. Dans les deux cas, ces actions peuvent être réalisées à partir de logiciels simples, mais ces logiciels ont des limites importantes auxquelles les langages de balisage apportent des solutions⁴.

Pour l’écriture de documents très simples comme une liste d’épicerie ou des notes rapides pendant une conférence, les logiciels de traitement de texte sont tout à fait convenables : ils sont simples et rapides à utiliser, un formatage professionnel du document n’est pas de mise. Utiliser un langage de balisage pour des tâches de base peut en effet rendre la tâche inutilement longue et complexe. Toutefois, plus la complexité d’un document augmente, plus il devient difficile d’obtenir un résultat satisfaisant en utilisant un logiciel de traitement de

³VS Code possède également une extension, Live Share, qui permet de travailler en temps réel sur un même document.

⁴Les langages de balisage permettent également de créer des pages Web. Bien que les pages Web puissent être créées à partir de sites Web comme WordPress, le langage HTML permet de produire des résultats plus *personnalisables*, plus *automatisables* et avec une plus *grande qualité graphique*. Cette question n’est pas abordée en détail dans ce chapitre.