Université de Reims Champagne-Ardenne
Faculté des sciences économiques et sociales
Faculté des sciences exactes et naturelles
Master 2 Analyse et politique économique
Master 2 Mathématiques et applications
Parcours : Statistique pour l'évaluation et la prévision

Tout ce que vous avez toujours voulu savoir sur RBloggers

par

Ouael ETTOUILEB, Adrien SAGRAFENA,
Oumaima FARISS, Ayoub BRIDAOUI et Salomé THIRIOT

Apprentissage Automatique

Encadrant : Frédéric Blanchard

Table des matières

Liste des tableaux

Table des figures

1 Introduction

Tout d'abord, il semble important de rappeler ce qu'est “R”. Le langage “R”, ainsi que les logiciels open source liés à ce dernier, sont une référence dans le monde des statisticiens. En effet, il est largement utilisé dans le développement de logiciels statistiques, l'analyse de données et la prévision et estimation.

R-bloggers est un agrégateur de plusieurs blogs au sein desquels des participants échangent. Ainsi, une communauté se crée autour d'une "R-blogosphère".
Son fonctionnement est simple, la page principale est composée de chaque début d'article, à l'intérieur de chaque article se trouve un lien qui nous amène au blog original, ainsi que des liens vers d'autres articles connexes.

Le but de R-Bloggers est simple : il repose sur un échange de connaissances spécifiquement liées à R. En effet, chacun peut s'y rendre afin de publier ou de répondre à un article ou bien apprendre de nouvelles notions à ce sujet. Ainsi, nous pouvons distinguer deux catégories de participants : les blogueurs et les utilisateurs.

Concernant les blogueurs, nous allons nous intéresser aux plus connus, souvent créateurs de packages. Tout d'abord, François Husson est à l'origine de certains packages, notamment FactoMineR et Factoshiny. Il publie principalement sur les thèmes de l'analyse de données et du clustering. Nous pouvons aussi évoquer Arthur Charpentier qui rédige des articles sur la statistique, la finance ainsi que sur l'économétrie. Ensuite, Hadley Wiskham est l'auteur de package Tidyverse. Il a ainsi permis de faire progresser la datavisualisation.

Ainsi, nous pouvons nous demander quels sont les centres d'intérêt de la communauté des bloggeurs du site R-Bloggers. Pour répondre à cette interrogation, nous allons devoir explorer et analyser les publications du site, plus particulièrement le contenu textuel des articles.
Notre base de données s'appuie sur une récolte d'articles depuis 2008 jusqu'à fin 2021, soit une base composée de 34611 articles, un article correspondant à une ligne de la base. Celle-ci est composée du titre de l'article, du lien renvoyant au blog d'origine, de la date de publication, de son auteur, ainsi que de l'activité de la publication (nombre de likes et de commentaires).

Finalement, l'activité de Rbloggers peut se résumer par quelques chiffres clés : d'une part, de trois à cinq articles publiés quotidiennement alimentent ce site, d'autre part 28 544 blogueurs à travers le monde entier alimentent le contenu du site.

Pour conclure, une véritable communauté s'est créée au sein de ce blog autour du langage R. Cela a permis de construire une relation d'entraide entre les utilisateurs et les blogueurs, mais également de développer davantage les fonctionnalités de ce langage, qui, nous le rappelons, est gratuit.

2 Problématique

Le nombre d'articles publiés sur R-bloggers dépasse la capacité humaine d'analyse et d'interprétation. Par conséquent, établir un résumé de l'activité des blogueurs ou des thématiques abordées est impossible à notre échelle. Les techniques récentes de machine learning et de NLP ont démontré leur grande efficacité dans ce domaine.

Nous allons les mettre en oeuvre afin de répondre à la problématique suivante :
Quels sont les sujets d'intérêt de la communauté des blogueurs du site R-Bloggers ?

3 Méthodologie

L'identification des sujets d'intérêts sur R-Bloggers.com peut être résolue à l'aide de techniques d'apprentissage non supervisé de type clustering. Étant donné que nous sommes à la recherche des thématiques (topics) présents sur R-Bloggers, les approches qu'on utilisera principalement seront les approches dites de Topic Modeling.

Dans cette section, nous allons présenter la méthodologie utilisée pour répondre à notre problématique. Nous détaillerons, la méthode de récupération et de pré-traitement de données ainsi que les algorithmes utilisés et l'interprétation des clusters obtenus.

3.1 Topic Modeling

En préambule, nous allons définir le Topic Modeling et son utilité dans le cadre de notre problématique.

Le Topic Modeling peut être défini comme l'ensemble de techniques statistiques permettant de découvrir des thématiques qui peuvent exister dans une collection de documents. Ces Topics retrouvés représentent des clusters de mots similaires [5].

Dans notre projet, nous allons appliquer l'ensemble de ces techniques afin de repérer les différentes thématiques abordées sur R-Bloggers.com, ainsi que pour identifier les thématiques abordées pour chaque auteur. Finalement, nous proposerons un outil sous format Dashboard permettant de résumer l'ensemble des données obtenues et l'activité des membres sur le site.

3.2 Collection de données

Pour commencer, nous avons collecté les données [1] de R-bloggers.com à l'aide d'un script [2] Python en utilisant le package BeautifulSoup. Nous avons récupéré les articles publiés de janvier 2008 à décembre 2021. Il s'agit de plus de 28544 articles annotés par le titre correspondant, la date de parution, le nom de l'auteur et nombre de commentaires.

3.3 Prolongement des mots (Word Embeddings)

Une fois les données collectées, il faut convertir les documents en valeurs numériques afin qu'ils soient exploitables par les algorithmes. Pour ce faire, nous avons utilisé le prolongement des mots.

1. Voir données sur notre répertoire github `https://github.com/ettouilebouael/Rbloggers_dashboard`
2. Voir code joint web_scaper.ipynb

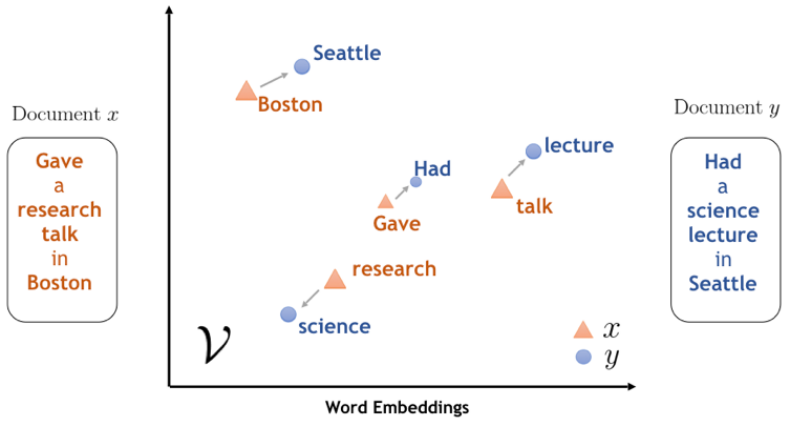

FIGURE 1 – Illustration du Word Embedings.
Lecture : Des mots similaires ont des vecteurs proches (p.ex. Research et Science)

En effet, le **Word Embeddings** transforme chaque mot du corpus en un vecteur de nombres réels et fait en sorte que les mots apparaissant dans des contextes similaires aient des vecteurs similaires [6]. Cela est obtenu en utilisant des réseaux de neurones artificiels entraînés sur des corpus très volumineux (p.ex BERT, FastAI, Word2vec, etc) pour prédire un mot à partir de son contexte et vice-versa. Ces modèles ont démontré des résultats impressionnants ces dernières années en NLP.

Afin extraire les différents prolongements de mots qui serviront de données d'entraînement pour nos algorithmes de clustering, nous avons utilisé BERT (Bidirectional Encoder Representations from Transformers). Ce dernier est un modèle de deep learning basé sur une architecture Transformer pré-entraînée sur des corpus volumineux par Google.

3.4 Réduction de dimension avec UMAP

Le vecteur de prolongement de mots est de dimension $(28544, 384)$, par conséquent, la réduction de la dimension des données est une étape critique avant le clustering. Il y a deux raisons à cela. Tout d'abord, cette réduction permet de visualiser les données afin de repérer visuellement les clusters. Ensuite, certains algorithmes comme DBSCAN ne sont pas extensibles (scalables) en grande dimension.

Étant donné que le clustering obtenu par l'utilisation conjointe de l'ACP et de l'algorithme des k-means, ou même par les k-means seulement [3], n'était pas satisfaisant, nous avons utilisé UMAP (Uniform Manifold Approximation and Projection for Dimension Reduction). C'est une technique de réduction de dimension non linéaire qui a pour objectif de maintenir les structures de données en grande dimension dans une dimension faible en préservant simultanément les structures locales et globales des données. UMAP suppose l'existence d'une variété dans les données dans l'espace de grande dimension, qui peut être construite à l'aide d'un graphe de k proches voisins. L'algorithme construit donc une représentation graphique des données dans l'espace de grande dimension, puis cherche la projection la plus similaire à cette représentation dans l'espace de faible dimension en minimisant l'entropie croisée entre les deux représentations graphiques par descente de gradient [3].

3. Plus de détails sur l'ensemble des expérimentations sur le répertoire github du projet : `https://github.com/ettouilebouael/Rbloggers_dashboard`

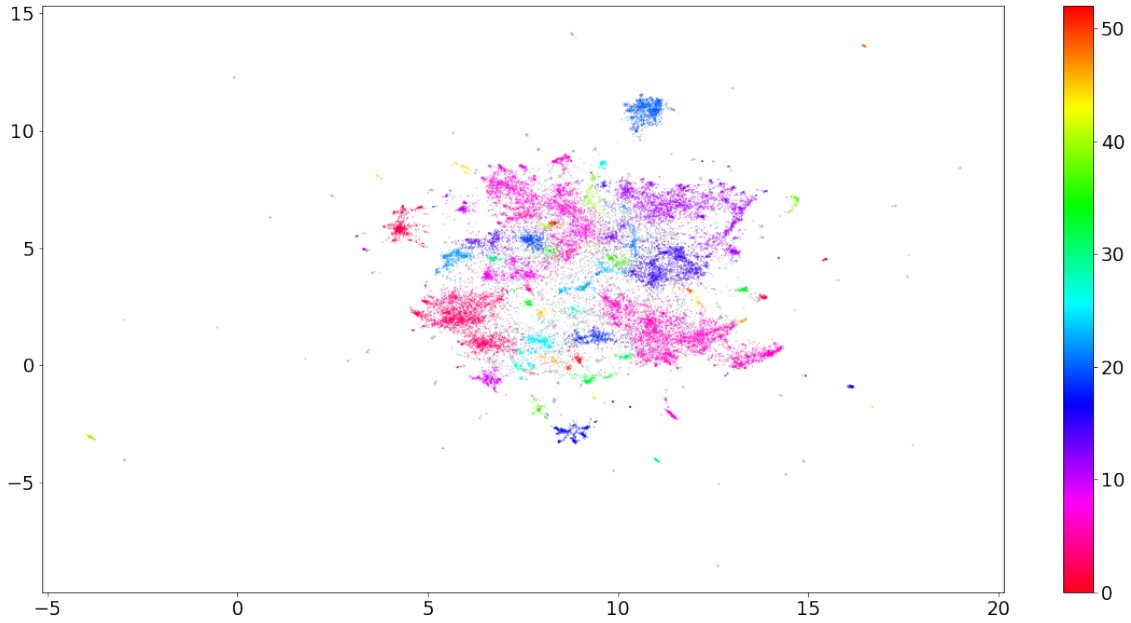

FIGURE 2 – Représentation des données sur les deux axes de UMAP

En pratique la pertinence des résultats dépend de deux hyperparamètres : le nombre de k voisins $n\_neighbors$ et la distance minimal entre les points dans l'espace de moindre dimension min_dist. Des petites valeurs pour le $n\_neighbors$ permettent de se concentrer sur les structures locales et des grandes valeurs permettent de se concentrer sur les structures globales. En ce qui concerne le min_dist, il contrôle la façon dont UMAP est autorisé à regrouper les points. Des petites valeurs amènent à des groupements denses, ce qui est très utile pour le clustering, tandis que des grandes valeurs empêchent l'algorithme d'empaqueter les points ensembles [2].
En effet, le choix des hyperparamètres de UMAP dépend des données et du cas d'usage. Nous avons donc essayé plusieurs combinaisons. Finalement, en visualisant les axes UMAP, nous avons retenu la combinaison la plus satisfaisante pour faire un clustering avec $n\_neighbor = 5$, $min\_dist = 10$ et avec un nombre de composantes égal à 5.

3.5 Clustering avec HDBSCAN

Dans la figure 2 de l'étape précédente, nous remarquons que les données sont réparties sur les deux axes de UMAP sous forme de clusters denses et séparés par de zones de faible densité. Dans ce contexte, l'utilisation d'un algorithme de clustering basé sur la densité (p.ex. DBSCAN ou HDBSCAN) semble la plus adaptée. Comme premier choix, nous pouvons implémenter DBSCAN. Ce dernier définit les clusters comme des régions continues de haute densité. En effet, pour chaque observation, DBSCAN compte combien d'observations sont situées à moins de ϵ distance. Si une observation dispose d'au moins $min\_samples$ observations dans son voisinage de distance ϵ, elle se considère comme observation cœur et son voisinage appartient au même cluster. Les observations éloignées des observations cœurs seront considérées comme des valeurs aberrantes (outliers) et ne seront pas incluses dans un cluster. Il faut noter que le nombre de classes est déterminé par le modèle durant l'entraînement [7].

En pratique, la pertinence des résultats dépend de deux hyperparamètes ϵ et $min\_samples$. Cependant, fixer la valeur ϵ pourrait être problématique car bien souvent, les clusters n'ont pas la même densité. Une alternative à DBSCAN est HDBSCAN. Il est basé sur DBSCAN, mais, l'hyperpramètre ϵ est remplacé par $min\_cluster\_size$ [4]. Ce dernier détermine la taille des clusters finaux. Ceci est plus simple à fixer dans notre cas car nous savons que nous cherchons des clusters généraux et des clusters spécifiques également. Par conséquent, nous devons fixer des petites valeurs pour $min\_samples$ et des grandes valeurs pour $min\_cluster\_size$.

L'évaluation du résultat de clustering a été réalisée en trois étapes :

1. Taux des outliers : même si la qualité des clusters est convaincante, un taux élevé de valeurs aberrantes n'est pas intéressant pour notre problématique car plusieurs thématiques intéressantes pourraient être ignorées ;
2. Comparaison d'indice silhouette : Il s'agit de mesurer l'exactitude de l'affectation d'une observation à un cluster. Une valeur proche de +1 signifie que l'observation a été affectée de son propre cluster. Une valeur proche de -1 signifie que cette observation a été affectée au mauvais cluster [7] ;
3. Cohérence des clusters identifiés : il s'agit d'interpréter les clusters manuellement et juger leur pertinence.

Notre objectif est de trouver des clusters cohérents, avec une bonne mesure de qualité (indice silhouette) et en minimisant le taux des outliers.

Pour cela, une recherche par quadrillage d'hyperparamètres a été réalisée sur les valeurs de $min\_cluster\_size = [5, 10, 20, 30, 40, 50, 70]$ et $min\_samples = [10, 20, 30, 40, 50, 60, 70]$. La meilleure clustering est obtenue avec $min\_cluster\_size = 20$ et $min\_sample = 40$. Le score silhouette correspond est de 0.5 et avec un taux de valeurs aberrantes égal 0.39.

3.6 Création de Topics avec TF-IDF

Dans cette étape, nous allons interpréter les clusters obtenus dans l'étape précédente. Pour ce faire, nous avons utilisé TF-IDF (Term Frequency(TF) — Inverse Dense Frequency(IDF)) afin d'identifier les mots les plus importants dans chaque cluster. TF-IDF est une valeur qui mesure le poids d'un mot dans un document. En effet, les mots peu fréquente ou uniques portent plus d'information et de valeur que les mots fréquents à travers le corpus. Par conséquent, ils auront une valeur TF-IDF plus élevée car ils sont plus discriminants. TF-IDF peut être calculé comme suit : $w_{i,j} = tf_{i,j} \times log\frac{N}{df_i}$ où i représente le mot, j le document et N le nombre total des documents. TF-IDF calcul donc $tf_{i,j}$ la fréquence d'un mot i dans le document j multiplié par $log\frac{N}{df_i}$ le logarithme de l'inverse de la proportion de documents j du corpus qui contiennent le terme i [1].

Pour ce faire, dans un premier temps, nous avons supprimé les "stops words" du corpus. Ensuite, nous avons créé une représentation vectorielle des articles en comptant la fréquence de chaque mot dans chaque article. Finalement, nous avons calculé les TF-IDF pour chaque mot dans chaque article. Pour l'interprétation, nous avons retenu les mots les plus importants dans chaque article (Tableau 1).

4 Thématiques abordées sur R-Bloggers

top words n1	top words n2	top words n3	top words n4	top words n5	size	Cluster Name
portfolio	returns	risk	volatility	return	910	Finance and Risk Management
bayesian	posterior	distribution	prior	stan	733	Statistics
conference	talks	talk	user	community	661	Conferences
game	probability	numbers	puzzle	player	599	Probabilities problems
map	maps	spatial	raster	leaflet	560	Geography and Spatial Data
regression	linear	mathbf	boldsymbol	models	538	Linear Models
ggplot2	ggplot	plot	plots	axis	497	ggplot2
cran	packages	install	version	dependencies	454	Intall, load and update packages
revolution	analytics	enterprise	sas	webinar	444	Business newsletters and webinars
temperature	stations	climate	weather	station	420	Ecology
vector	loop	functions	vectors	list	402	R Tips and Tutorials
shiny	app	ui	server	apps	397	Shiny apps
players	teams	team	season	league	385	Sports
election	vote	party	voting	census	383	Elections
rcpp	release	dirk	changes	page	377	Packages Realeses
series	forecast	forecasting	arima	seasonal	348	Time series
markdown	knitr	document	latex	rmd	324	Rmarkdown
science	business	scientist	scientists	learning	305	Data Science and Business
hugo	think	course	yeah	science	287	Online Courses
book	books	chapter	chapters	edition	228	Books

TABLE 1 – Les mots les plus importants dans les 20 clusters les plus grands parmi 87 clusters, annotés par nos interprétations, voir annexe A pour le tableau complet

Le tableau ci-dessus liste les cinq termes les plus populaires pour l'ensemble des articles de chaque cluster obtenu. Prenons un exemple concret : portfolio, returns, risk, volatility et return sont les 5 mots les plus fréquemment trouvés dans le cluster "Financial and Risk Management". Cela semble totalement en phase avec le thème : les mots comme volatility, portfolio et même risk dans le contexte sont presque exclusivement du champ lexical de la finance et de la gestion de risque. Les résultats présentés ici et la cohérence des mots listés pour chaque cluster renforcent notre conviction sur la pertinence de notre démarche et de la méthode associée.

5 Thématiques abordées par les auteurs connus

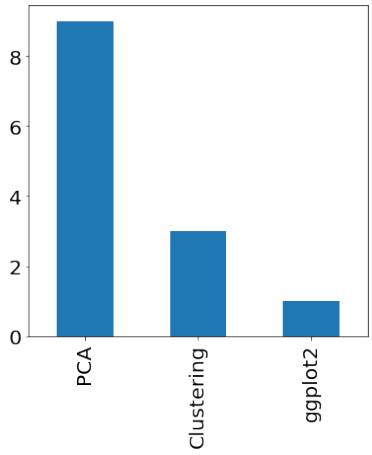

FIGURE 3 – Nombre d'articles par thématiques pour Francois Husson

Francois Husson : chercheur au CNRS (IRMAR - UMR 6625), professeur de statistique à l'Agrocampus Ouest de Rennes. Le créateur de FactoMineR et Factoshiny ne traite que trois sujets qui sont : l'analyse des composantes principales qui permet l'analyse de données, le package ggplot 2 qui est une librairie disponible dans R développée pour la visualisation de données et le clustering qui a pour but de partitionner les données afin de les analyser.

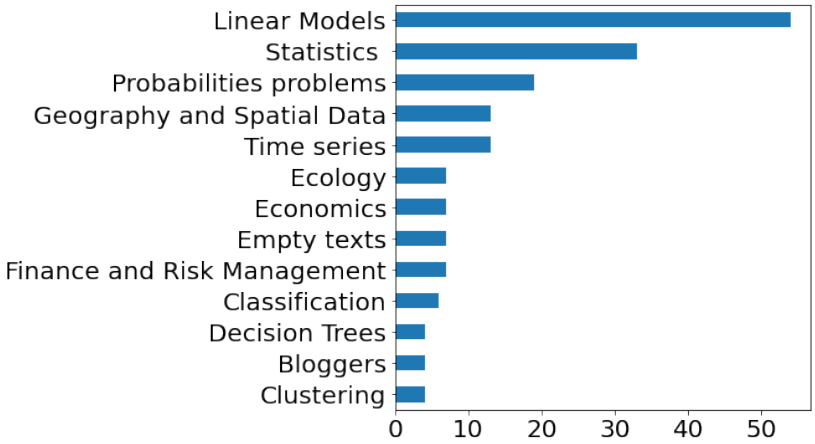

FIGURE 4 – Nombre d'articles par thématiques pour Arthur Charpentier

Arthur Charpentier : chercheur en actuariat, finance et économétrie au CNRS (CREM - UMR 6211), professeur à l'université du Québec à Montréal (UQAM). Les trois principales notions abordées par Arthur Charpentier sont les modèles linéaires qui permettent d'expliquer une variable choisie, les statistiques et le calcul des probabilités.

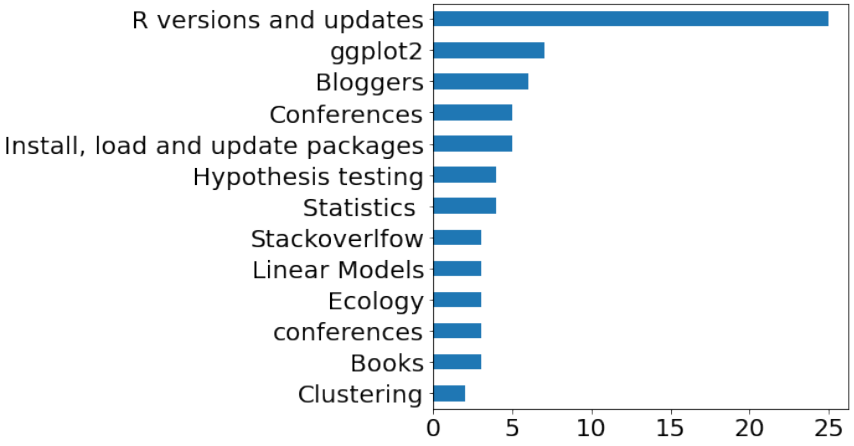

FIGURE 5 – Nombre d'articles par thématiques pour Tal Galili

Tal Galili : le fondateur de Rbloggers rédige principalement à propos des différentes versions existantes de R, et de celles du package ggplot2 susnommé.

Rviews : une communauté dans la communauté RBloggers. Rviews met en place des meetings et des conférences afin de produire de nouveaux packages R. Ainsi, les trois thématiques les plus fréquemment abordées sont Package Picks (présentation des packages récemment développés et potentiellement prometteurs) qui a pour but d'aider à la décision d'un choix de package plutôt qu'un autre, Finance and Risk management qui, comme son nom l'indique, traite de la gestion des risques financiers, et pour finir Conférences qui est justement la préoccupation première de Rviews.

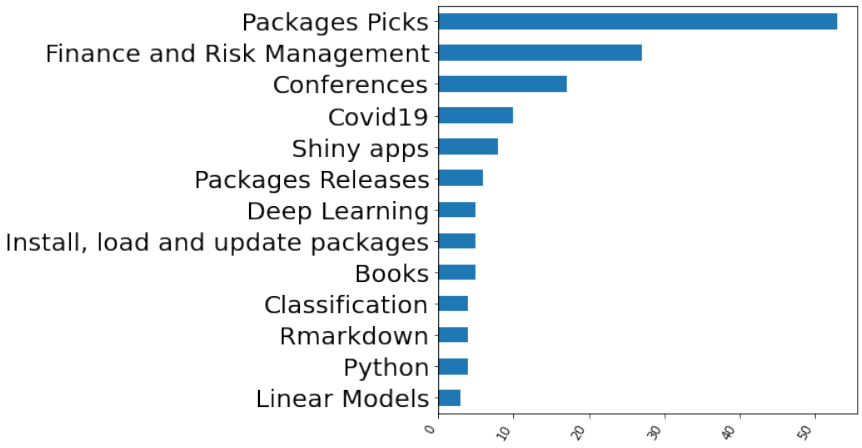

FIGURE 6 – Nombre d'articles par thématiques pour Rviews

6 Discussion

À l'aide du Topic modeling, nous avons pu découvrir les thématiques d'intérêt sur Rbloggers. Les résultats obtenus semblent satisfaisants.

Cependant, lors de son exécution, HDBSCAN n'a considéré que 60% des articles, étant donné que le reste a été considéré comme des outliers. Ce fut le prix à payer pour avoir des résultats cohérents et publiables dans ce rapport.

Il faut noter également qu'avec HDBSCAN, nous n'arriverons pas à faire des prédictions directement pour des nouveaux articles, parce qu'il n'existe pas une méthode *predict* sur la version Scikit-Learn de HDBSCAN.

Plusieurs articles ont été mal ou partiellement collectés par notre script Python, ce qui a pu introduire du biais dans nos analyses.

7 Conclusion

Ce projet fut pour nous l'occasion d'aborder une problématique très ambitieuse aux contenus très denses.

Nous avons dû en autodidactes nous confronter à un très riche panel sur le plan technique : la collecte de données sur le web et les réseaux sociaux, le NLP et ses techniques, l'algorithme HDBSCAN, la datavisualisation sous forme de dashboard interactif, le déploiement de l'application Shiny

obtenue sous forme d'une page HTML. Nous avons implémenté cette solution par étapes successives, répartissant au mieux les tâches et utilisant le versionnage du projet par GIT pour optimiser notre travail, ce qui nous a permis de gérer la colossale charge de travail et d'aboutir.

À ce jour, les pistes d'amélioration restent cependant nombreuses et très ouvertes : améliorer le pipeline de collecte de données pour éviter les quelques articles mal récupérés, permettre une intégration à intervalles réguliers automatisée des nouvelles publications, prédire la classe de nouveaux articles (avec un K-NN par exemple) et confronter le résultat obtenu à la réalité et aux thématiques que nous avons dégagées en amont.

Références

[1] "6.2. Feature extraction." [Online]. Available : https://scikit-learn/stable/modules/feature_extraction.html

[2] "Basic UMAP Parameters — umap 0.5 documentation." [Online]. Available : https://umap-learn.readthedocs.io/en/latest/parameters.html

[3] "How UMAP Works — umap 0.5 documentation." [Online]. Available : https://umap-learn.readthedocs.io/en/latest/how_umap_works.html

[4] "Parameter Selection for HDBSCAN* — hdbscan 0.8.1 documentation." [Online]. Available : https://hdbscan.readthedocs.io/en/latest/parameter_selection.html

[5] "Topic model," Nov. 2021, page Version ID : 1055578481. [Online]. Available : https://en.wikipedia.org/w/index.php?title=Topic_model&oldid=1055578481

[6] "*Word embedding*," Jan. 2022, page Version ID : 189581800. [Online]. Available : https://fr.wikipedia.org/w/index.php?title=Word_embedding&oldid=189581800

[7] A. Geron, "Hands-on machine learning with scikit-learn and tensorflow : concepts, tools, and techniques to build intelligent systems," 2017.

A Tableau des clusters complet

top words n1	top words n2	top words n3	top words n4	top words n5	top words n6	top words n7	top words n8	size	cluster_2
portfolio	returns	risk	volatility	return	strategy	portfolios	stock	910	Finance and Risk Management
bayesian	posterior	distribution	prior	stan	probability	normal	mcmc	733	Statistics
conference	talks	talk	user	community	consortium	group	meetup	661	Conferences
game	probability	numbers	puzzle	player	solution	number	sum	599	Probabilities problems
map	maps	spatial	raster	leaflet	sf	coordinates	polygons	560	Geography and Spatial Data
regression	linear	mathbf	boldsymbol	models	logistic	variable	fit	538	Linear Models
ggplot2	ggplot	plot	plots	axis	bar	color	labels	497	ggplot2
cran	packages	install	version	dependencies	downloads	installed	devel	454	Intall, load and update packages
revolution	analytics	enterprise	sas	webinar	hadoop	big	microsoft	444	Business newsletters and webinars
temperature	stations	climate	weather	station	year	co2	water	420	Ecology
vector	loop	functions	vectors	list	operator	object	apply	402	R Tips and Tutorials
shiny	app	ui	server	apps	application	input	reactive	397	Shiny apps
players	teams	team	season	league	player	game	football	385	Sports
election	vote	party	voting	census	votes	voters	state	383	Elections
rcpp	release	dirk	changes	page	eddelbuettel	cran	devel	377	Packages Realeses
series	forecast	forecasting	arima	seasonal	forecasts	seasonality	models	348	Time series
markdown	knitr	document	latex	rmd	rmarkdown	html	file	324	Rmarkdown
science	business	scientist	scientists	learning	big	analytics	machine	305	Data Science and Business
hugo	think	course	yeah	science	datacamp	courses	people	287	Online Courses
book	books	chapter	chapters	edition	programming	practical	science	228	Books
words	word	text	sentiment	corpus	idf	topic	documents	226	Sentiment Analysis
frame	columns	column	frames	rows	row	names	variables	215	Dataframes
covid	19	cases	deaths	vaccine	pandemic	countries	virus	211	Covid19
ephemera	lorem	ipsum	aki	edits	read	ps	town	202	Empty texts
parallel	cores	foreach	gpu	memory	cluster	cpu	parallelization	186	Parallezation
rstudio	addin	addins	ide	install	shortcuts	pane	ctrl	184	Rstudio
tweets	twitter	tweet	followers	najib	razak	words	rt	176	Twitter
neural	deep	network	layer	keras	learning	networks	layers	175	Deep Learning
bike	crime	taxi	citi	bikes	trips	city	trip	163	Transports
ropensci	review	software	community	peer	reviewers	submitted	packages	159	rOpenSci
test	hypothesis	anova	null	tests	power	significance	sample	158	Hypothesis testing
language	programming	learn	learning	statistical	software	packages	functions	152	Piping
earl	conference	mango	london	speakers	boston	abstracts	solutions	149	conferences
python	pandas	reticulate	languages	language	science	programming	dplyr	148	Python
database	sql	query	mysql	odbc	connection	server	table	144	SQL and Databases
race	runners	lap	driver	marathon	f1	runner	medal	140	Marathons and Races
version	upgrade	windows	release	ubuntu	installr	released	latest	134	R versions and updates
color	palette	colors	colour	palettes	colours	hcl	rgb	131	Data Visualisation
clusters	clustering	cluster	algorithm	distance	means	hierarchical	pokemon	127	Clustering
csv	excel	file	files	read	import	xlsx	importing	122	Data Importing
scraping	web	scrape	rvest	html	page	url	site	121	Web Scraping
dplyr	columns	mutate	column	verbs	frame	dplyrxdf	replyr	121	Dplyr
gene	genes	clusterprofiler	species	annotation	ensembl	enrichment	protein	117	Biostatistics
roc	auc	class	validation	curve	cross	threshold	training	117	Classification
armadillo	rcpparmadillo	sparse	release	matrices	upgraded	added	expanded	113	RcppArmadillo
bloggers	blog	posts	blogs	year	site	tal	résumé	113	Bloggers
correlation	correlations	coefficient	variables	spearman	matrix	cor	pearson	112	Correlation
treatment	causal	effect	outcome	intervention	effects	estimate	outcomes	107	Econometrics
tree	trees	xgboost	boosting	forest	random	training	decision	106	Decision Trees
spark	sparklyr	sparkr	drill	apache	scala	sql	cluster	105	Data Engineering
animation	gganimate	animated	animations	gif	lego	art	image	93	Animations
blogdown	hugo	theme	blog	jekyll	website	site	netlify	87	Blogging
date	dates	lubridate	posixct	datetime	timezone	format	interval	85	Date Format
tidyverse	tidy	tidymodels	packages	dplyr	fastverse	tidyr	saudi	84	Tidyverse
matches	runs	ipl	batsman	wickets	bowlers	batsmen	t20	80	Cricket
pca	principal	components	component	variables	pcs	variance	matrix	78	PCA
v0	vignette	provides	v1	implements	al	et	vignettes	73	Packages Picks
shinyproxy	server	docker	shiny	apps	nginx	heroku	app	72	Rshiny Deployement
git	commit	github	repository	commits	control	gitlab	repo	68	Git
plotly	dash	ly	interactive	plot	chart	python	plots	67	Plotly
finance	conference	chicago	presentations	presenters	financial	2012	talks	65	Finance conferences
kaggle	competition	training	challenge	features	dataset	apartment	learning	64	Kaggle
aws	ec2	instance	amazon	server	cloud	ami	rstudio	63	AWS
songs	music	album	song	lyrics	albums	genres	words	62	Music
pipe	magrittr	operator	wrapr	piping	pipeline	pipes	dot	61	R learning
google	api	nldi	query	apis	spreadsheet	request	analytics	59	Google
rankings	poll	languages	survey	popularity	kdnuggets	tiobe	software	59	Programming languages survey
littler	digest	hash	sha	release	debian	repo	cran	59	Debian
dalex	models	explanations	ml	xai	iml	overs	explainers	58	Explainability
movies	movie	episode	imdb	ratings	rating	episodes	films	57	Movies
seed	random	rng	361322	permutation	game	numbers	sample	56	Randomnes
node	nodes	edges	network	gd	igraph	edge	networks	55	Graphs
questions	stackoverflow	answers	stack	question	overflow	help	tags	54	Stackoverlfow
visualization	visual	scatter	chart	visualizations	graphs	charts	visuals	53	Data Visualisation
bio7	rserve	imagej	java	macosx	installation	install	linux	50	Bio7
milanor	meeting	milano	quantide	stefano	oscuri	fiori	andrea	50	Milano meeting
inequality	gini	expectancy	life	income	age	countries	mortality	49	Economics
azure	vm	storage	orch	databricks	oracle	virtual	hdfs	48	Azure and Big Data
species	gbif	ebird	biodiversity	birds	occurrence	bird	records	47	Biodiversity
gold	fantasy	football	mining	appeared	week	analytics	cbs	45	Gold mining week
mcmski	isba	chamonix	iv	conference	bayescomp	sessions	registration	44	MCMSki
table	dt	frame	columns	tree	rows	faster	column	44	dt table
tidymodels	tidytuesday	modeling	video	screencasts	resamples	recipe	screencast	43	Tidymodels
anytime	cctz	rcppcctz	release	civil	dirk	originated	excessive	43	Date Format
graphics	plot	plots	axis	wv	graphs	charts	chart	43	Data Visualisation
statistics	statistical	students	statisticians	teaching	books	maths	sas	42	R Academics
nimble	mcmc	youngstats	hierarchical	mixture	models	bugs	nonparametric	41	NIMBLE

TABLE 2 – Tableau des clusters complet avec les mots les importants annotés par nos interprétations