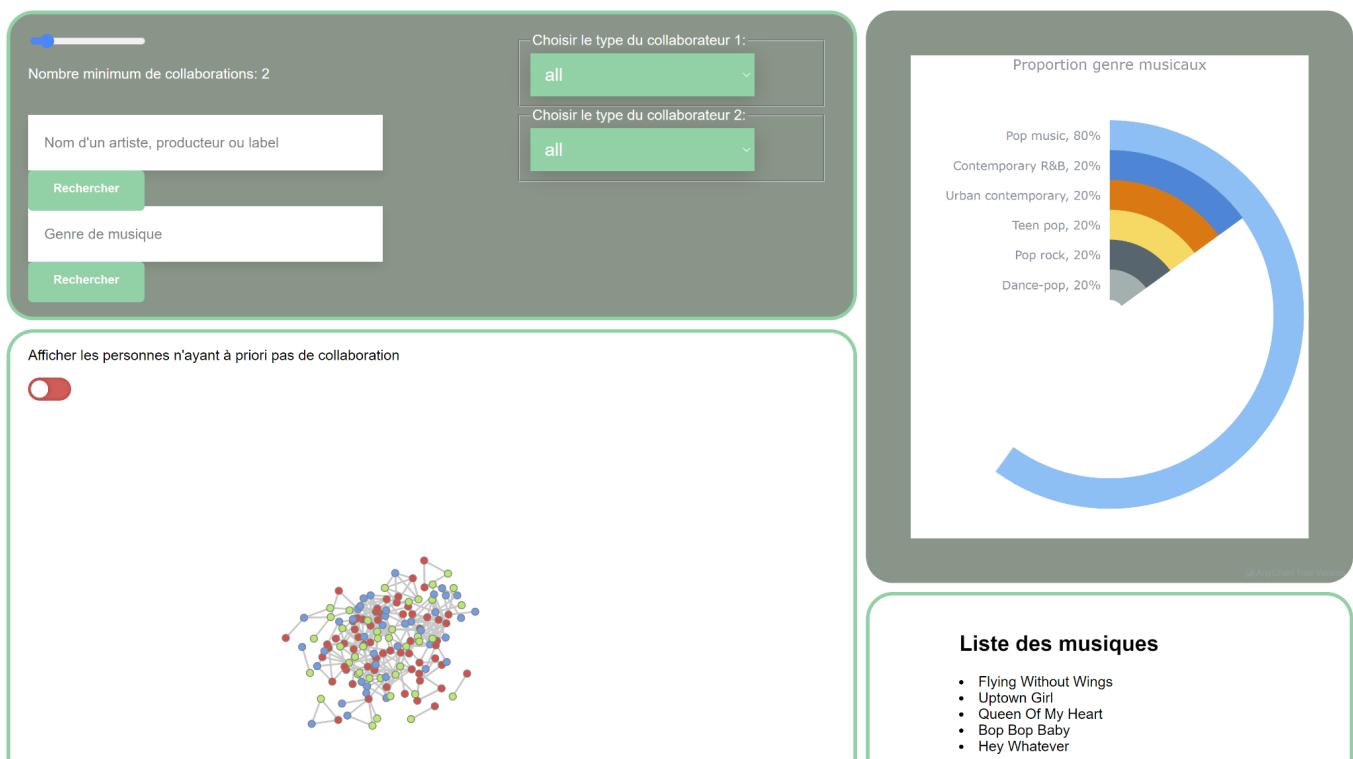


Projet de data visualization - Les collaborations

Par Antoine Huot-Marchand & Loic Madern



Introduction

Le but de ce projet est de mettre en place une chaîne de traitement pour la visualisation de données issues du projet wasabi. Wasabi étant un corpus de chansons enrichi en métadonnées. Il comporte près de deux millions de musiques, accessible depuis une API.

Notre chaîne de visualisation se divise en quatres principales étapes à savoir l'acquisition des données, l'enrichissement des données, l'abstraction des données pour l'affichage enfin la visualisation. Notre objectif est de réaliser une visualisation mettant en scène les collaborations de différents artistes, labels et producteurs.

Utilisateurs ciblés

Les personnes cibles pour notre application sont des fans de musique voulant consulter les collaborations d'un ou des acteur(s) de la musique (artistes, label et producteurs). Notre solution est destinée tout autant à un fan d'un artiste qu'à un musicophile amateur d'un genre de musique spécifique. Nous avons illustré cela grâce à l'élaboration de personas voir ci-contre la prochaine partie du rapport.

Personas

On distingue 2 types de personas un fan d'un artiste en particulier fan de U2 et un autre fan du genre musical electro.



Nom
Antoine le Calloch

Intitulé de poste
Artisan de crêpes bretonne

Âge
22 ans

Moyens d'écouter la musique

- Ordinateur avec l'application deezer
- Enceinte embarquée à forte résonance

Objectifs

Créer une playlist de musique d'ambiance pour son bar à crêpes. Il voudrait des musiques du genre electro avec beaucoup de collaborations pour avoir une playlist riche.

Hobbies

- Aller en festival de musique
- Faire du surf
- Faire du parkour (gymnastique urbaine)

Tâches des personas (visual goals)

- Observer la liste de musique présente dans les collaborations
- Analyser la proportion des genres musicaux présent dans les collaborations
- Explorer les collaborations la multitude de collaborations
- Connaître le nombre de collaborations entre deux collaborateurs
- Filtrer par genre en particulier
- Filtrer par nom
- Filtrer par type (artistes, label et producteurs)
- Filtrer par nombre de collaborations

Traitements des données

Requêtes des données

Tout d'abord, il a été nécessaire de choisir quelles données étaient nécessaires pour accomplir les tâches utilisateurs (définies plus haut) et mettre en place les visualisations. La requête des données (via le fichier `fetch_and_transform_v2`) est réalisée grâce à un appel de la méthode GET sur l'api de wasabi (wasabi.i3s.unice.fr/api/v1/artist_all/). La requête étant coûteuse, et le nombre de données étant très conséquent, nous avons pris le parti de ne récupérer qu'une partie des données totales, ce qui est déjà entièrement suffisant pour appliquer nos différentes visualisations. Le choix du nombre de données à récupérer peut être modifié avec la valeur du offset

(72000 dans notre cas). Plus l'offset est proche de 0, plus le nombre de données récupérées sera grand. La requête porte sur 3 types de données différentes (songs, albums and artists) ce qui nécessite la création de 3 dataframes distincts.

Filtrage des données

Bien entendu, pour chacun des dataframes, beaucoup d'arguments sont inutiles à la réalisation de nos tâches utilisateurs et à la création de nos visualisations. Nous avons alors décidé de filtrer chacun des dataframes pour ne garder que les attributs ayant du sens dans le cadre de notre projet qui concerne les collaborations entre artistes.

Voici les attributs que nous avons décidé de garder à l'origine :

- *SONGS* : id, id_album, title, language_detect, genre, producer, recordLabel, writer, recorded
- *ALBUMS*: _id, name, genre, id_artists, title, country, language
- *ARTISTS* : id, name, locationInfo, genres, type, gender, recordLabel, labels

Les *id* de chacun des dataframes permettent de faire le lien entre chacune des données car une chanson a été écrite par un ou plusieurs artistes et fait partie d'un album en particulier.

Nous avons décidé de choisir ses attributs afin de travailler sur le nom des chansons et des albums sur lesquels les différents artistes ont collaboré (d'où les attributs *name*). Il est intéressant de voir les collaborations avec d'autres types d'entités comme les producteurs et labels (ce qui justifie la présence des attributs *producer*, *recordLabel*, *labels* et *writer*). Enfin, nous avons jugé qu'il serait intéressant de connaître les langues et pays dans lesquels se font les collaborations (*language_detect*, *country*, *locationInfo*) dans quels genre de musique (*genre*) elles s'inscrivent.

Cette étape de filtrage est essentielle car elle permet de réduire la taille des dataframes pour une meilleure optimisation afin de ne pas traiter des données superflues.

Transformation des données

Les données précédemment requêtées et filtrées ne suffisent évidemment pas à construire nos différentes visualisations. Nous avons alors créé une autre table *COLLABS* permettant de recenser des champs propres à une collaboration entre deux entités.

Ce nouveau dataframe a donc comme attribut :

`collaborator1 (source), type1, location_1, collaborator2 (target), type2, location_2, list_of_songs, genres, percent, nb_collabs (value)`

source	type1	location_1	target	type2	location_2	list_of_songs	genres
530	Richie McDonald	writer	Frank J. Myers	writer		I'm Already There	Country music
531	Gary Baker (songwriter)	writer	Frank J. Myers	writer		I'm Already There	Country music
532	Peter Cetera	writer	David Foster	writer		Hard To Say I'm Sorry	c("Progressive rock", "Adult contemporary music", [...])
533	David Hodges	writer	Ben Moody	writer		What About Now	c("Pop rock", "Alternative rock")
534	John Shanks	writer	Gary Barlow	writer	Lighthouse		c("Pop music", "Pop rock")
535	Mark London	writer	Don Black (lyricist)	writer		To Sir, With Love	character(0)
536	Andrew Ridgeley	writer	George Michael	writer		c("Club Tropicana", "George Michael:Careless Whisp [...]")	c("Post-disco", "Post-grunge", "Smooth jazz", "Alt [...]")
537	Don Was	writer	David Was	writer		Where Did Your Heart Go?	c("Dance-rock", "Post-disco", "Soft rock")
538	Tommy Cunningham	writer	Marti Pellow	writer		Angel Eyes (Home And Away)	Soft rock
539	Tommy Cunningham	writer	Graeme Clark (musician)	writer		Angel Eyes (Home And Away)	Soft rock
540	Tommy Cunningham	writer	Neil Mitchell (musician)	writer		Angel Eyes (Home And Away)	Soft rock
541	Marti Pellow	writer	Graeme Clark (musician)	writer		Angel Eyes (Home And Away)	Soft rock
542	Marti Pellow	writer	Neil Mitchell (musician)	writer		Angel Eyes (Home And Away)	Soft rock
543	Graeme Clark (musician)	writer	Neil Mitchell (musician)	writer		Angel Eyes (Home And Away)	Soft rock
544	Johnny Franz	producer	Burt Bacharach	producer		c("I Just Don't Know What To Do With Myself", "Wis [...]")	c("Alternative rock", "Blues rock", "Pop music", " [...]")
545	Johnny Franz	producer	Hal David	producer		c("I Just Don't Know What To Do With Myself", "Wis [...]")	c("Alternative rock", "Blues rock", "Pop music", " [...]")
546	Johnny Franz	producer	Jack White	producer		I Just Don't Know What To Do With Myself	c("Alternative rock", "Blues rock", "Pop music", " [...]")
547	Burt Bacharach	producer	Hal David	producer		c("I Just Don't Know What To Do With Myself", "Don [...]")	c("Alternative rock", "Blues rock", "Pop music", " [...]")
548	Burt Bacharach	producer	Jack White	producer		I Just Don't Know What To Do With Myself	c("Alternative rock", "Blues rock", "Pop music", " [...]")
549	Hal David	producer	Jack White	producer		I Just Don't Know What To Do With Myself	c("Alternative rock", "Blues rock", "Pop music", " [...]")
550	Tom Dowd	producer	The Rascals	producer		Good Lovin'	c("Blue-eyed soul", "Rhythm and blues")

En effet une collaboration se définit à l'aide de deux entités qui collaborent (**collaborator1**, **collaborator2**) possédant chacun un type ou statut (**type1**, **type2**) se trouvant parmi les différentes catégories (writer, label, producer). Chaque collaborateur a une localisation qui lui est propre (**location_1**, **location_2**). Enfin, leur collaboration concerne une liste de chansons (**list_of_songs**) pouvant appartenir à plusieurs genres de musiques (**genres**) avec une proportion différente (**percent**). De plus, il est important de rappeler le nombre de collaborations effectuées (**nb_collabs**) qui correspond finalement aux nombre de chansons indiquées.

percent	value
100	1
100	1
c("100", "100", "100", "100")	1
c("100", "100")	1
c("100", "100")	1
0	1
c("50", "50", "50", "50", "50", "50")	2
c("100", "100", "100")	1
100	1
100	1
100	1
100	1
100	1
100	1
c("50", "50", "100", "50")	2
c("50", "50", "100", "50")	2
c("100", "100", "100", "100")	1
c("20", "20", "100", "20", "20", "60", "20")	5
c("100", "100", "100", "100")	1
c("100", "100", "100", "100")	1
c("100", "100")	1

Voici donc le procédé utilisé pour construire ce dataframe :

Nous avons d'abord séparé le remplissage du dataframe selon les 6 types de collaborations existants :

- writer - writer
- writer - producer
- writer - label
- producer - label
- producer - producer
- label - label

Cela est utile puisque certains attributs dépendent du type des collaborateurs. Plutôt que d'ajouter des conditions à un code déjà complexe, cette séparation permet de RUN indépendamment chaque partie de la table *COLLABS* et peut parfois faire gagner du temps si l'on ne s'intéresse qu'à un ou plusieurs types de collaborations.

Pour chacune de ses parties indépendantes, le procédé reste néanmoins le même à quelques détails près.

En effet, pour chacune des chansons présentes dans le dataframe *SONGS*, nous allons vérifier si la chanson a nécessité une ou plusieurs collaborations. Ainsi, pour chacune de ces collaborations, nous allons insérer une ligne au dataframe *COLLABS* avec les informations de la collaboration

(si celle-ci n'est pas déjà présente dans *COLLABS*). Dans le cas où la collaboration a déjà été identifiée dans la table *COLLABS*, la ligne sera modifiée avec l'ajout des informations propres à la chanson comme l'ajout du titre dans *list_of_songs*, du ou des genres dans *genres* s'ils ne sont pas déjà présents, la modification de la proportion des genres musicaux de la collaboration puis le nombre de collaborations.

Enfin, nous avons aussi décidé de créer un second dataframe *UNKNOWN_COLLAB_ARTISTS* contenant les informations sur les potentiels collaborateurs dont on ne connaît pas la liste des collaborations. Pour ce faire, il a fallu récupérer le nom des collaborateurs non présents dans *COLLABS* depuis la table *ARTISTS*.

	name	statut	location
1	Tommy Dorsey	writer	United States, Pennsylvania, Shenandoah
2	Tricky	writer	England, Bristol, Knowle West
3	Trick Pony	writer	United States, Tennessee, Nashville
4	Trijntje Oosterhuis	writer	The Netherlands
5	Trik Turner	writer	United States, Arizona, Phoenix
6	Tommy Castro	writer	United States, California, San Jose
7	Telarc International Corporation	label	
8	Tommy Edwards	writer	NA
9	Trillville	writer	United States, Georgia, Atlanta
10	Disturbing tha Peace	label	
11	Trina Broussard	writer	NA
12	Trina	writer	United States, Florida, Miami
13	Coke Boys Records	label	
14	Tommy Keene	writer	United States, California, West Hollywood
15	Not Lame Recordings	label	
16	Tommy Johnson	writer	United States, Mississippi, Terry
17	Paramount Records	label	
18	Tommy Lee	writer	United States, California, West Covina
19	Leathür Records	label	

Ainsi, ce traitement des données a nécessité plusieurs étapes dont les requêtes, le parsing des données, la transformation. Certaines difficultés sur la construction du dataframe et la transformation des données notamment sur les attributs de type liste ont pu ralentir notre avancée.

Certaines méthodes retenues nous ont permis de contourner ces difficultés :

- str_split, str_sub et gsub nous ont permis de créer des attributs de type liste à partir d'une chaîne de caractères complexe.
- grep nous a permis de vérifier si un élément était déjà présent dans une liste de caractères (pour ne pas se retrouver avec une liste contenant des doublons par la suite).
- mutate nous a enfin permis de modifier notre dataframe au fur et à mesure.

Néanmoins, il serait envisageable d'enrichir nos dataframes dans le futur avec la prise en compte des pays (envisagée à l'origine) et une nouvelle visualisation sur les albums présents dans chaque collaboration avec des informations comme la longueur des chansons etc...

Data visualisation

Visualisation network

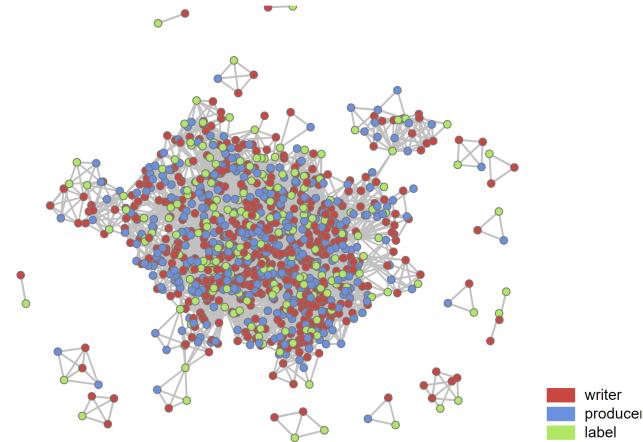
Justification du choix du type de visualisation

Concernant le choix du type de données que nous avions, nous avons considéré que le type de visualisation adapté serait le network. En effet, il est possible de représenter une collaboration avec un lien entre deux collaborateurs qui seraient eux représentés par des nœuds.

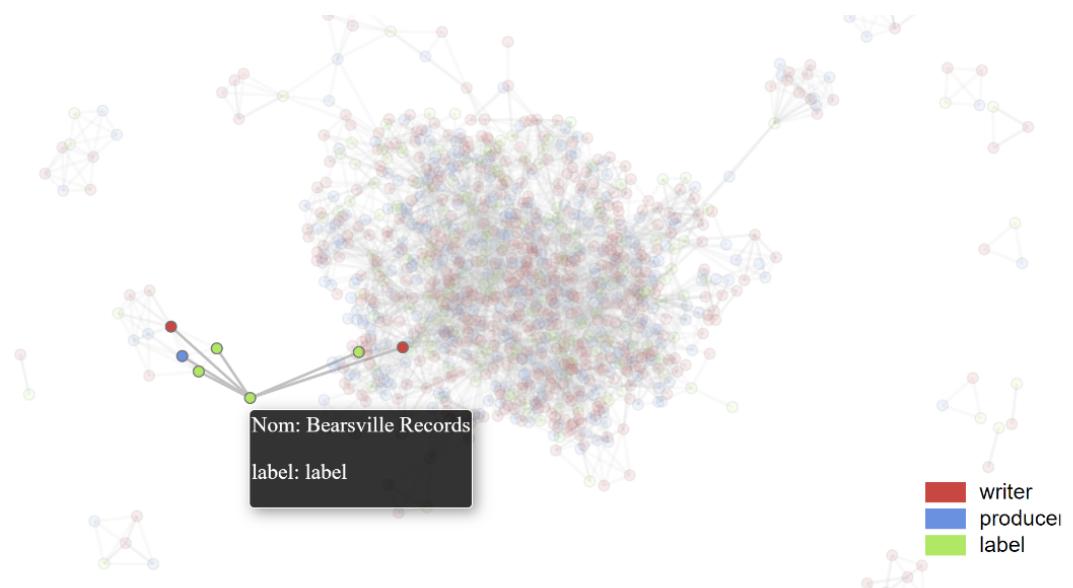
Overview (par Antoine & Loic)

La visualisation se constitue ainsi : un network où les nœuds représentent les collaborateurs et les liens représentent les collaborations. Les nœuds ont une couleur parmi le rouge, le bleu ou le vert pour distinguer les types de collaborateurs (artiste, label, producteur). Ces couleurs ont été choisies car elles sont facilement différenciables les unes des autres. Leur contraste étant assez prononcé pour pouvoir les différencier.

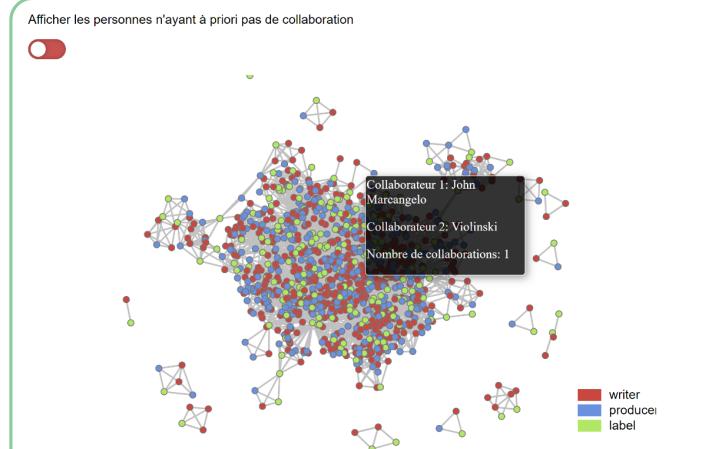
Il est à noter qu'un lien donné représente l'ensemble des collaborations, réalisé entre les deux collaborateurs. Également un nœud peut avoir n liens avec n nœuds différents. Cela signifie que le nœud qui représente l'artiste ou label ou le producteur a fait plus n collaborations, avec n collaborateurs différents.



Il est également possible d'interagir avec le network, de sélectionner un nœud et de les déplacer, isoler.

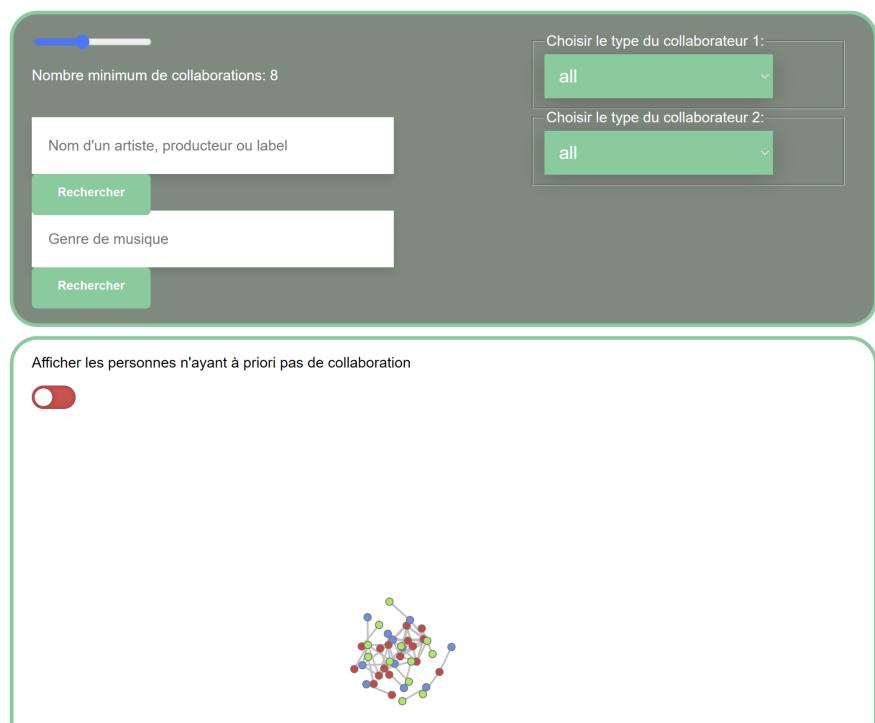


Il est également possible d'avoir plus d'information sur une collaboration en passant la souris par dessus un lien (hover).



Zoom & Filter (par Loic)

Il existe plus de fonctionnalités notamment des filtres pour afficher un nombre minimum de collaborations via un curseur. Dans l'exemple ci-dessous, le nombre minimum de collaborations est de 8.



Il est possible également de chercher les collaborations d'un artiste, producteur ou label par leur nom. Dans l'exemple ci-dessous on cherche U2 via la barre de recherche.

The screenshot shows a search interface with the following elements:

- A slider for "Nombre minimum de collaborations: 0".
- A search input field containing "U2".
- A "Rechercher" button.
- A second search input field containing "Genre de musique".
- A "Rechercher" button.
- Two dropdown menus labeled "Choisir le type du collaborateur 1:" and "Choisir le type du collaborateur 2:", both set to "all".
- A toggle switch labeled "Afficher les personnes n'ayant à priori pas de collaboration" (Display people who have no prior collaboration) which is turned off.
- A network graph node for "U2" with the label "Nom: U2" and "label: writer".

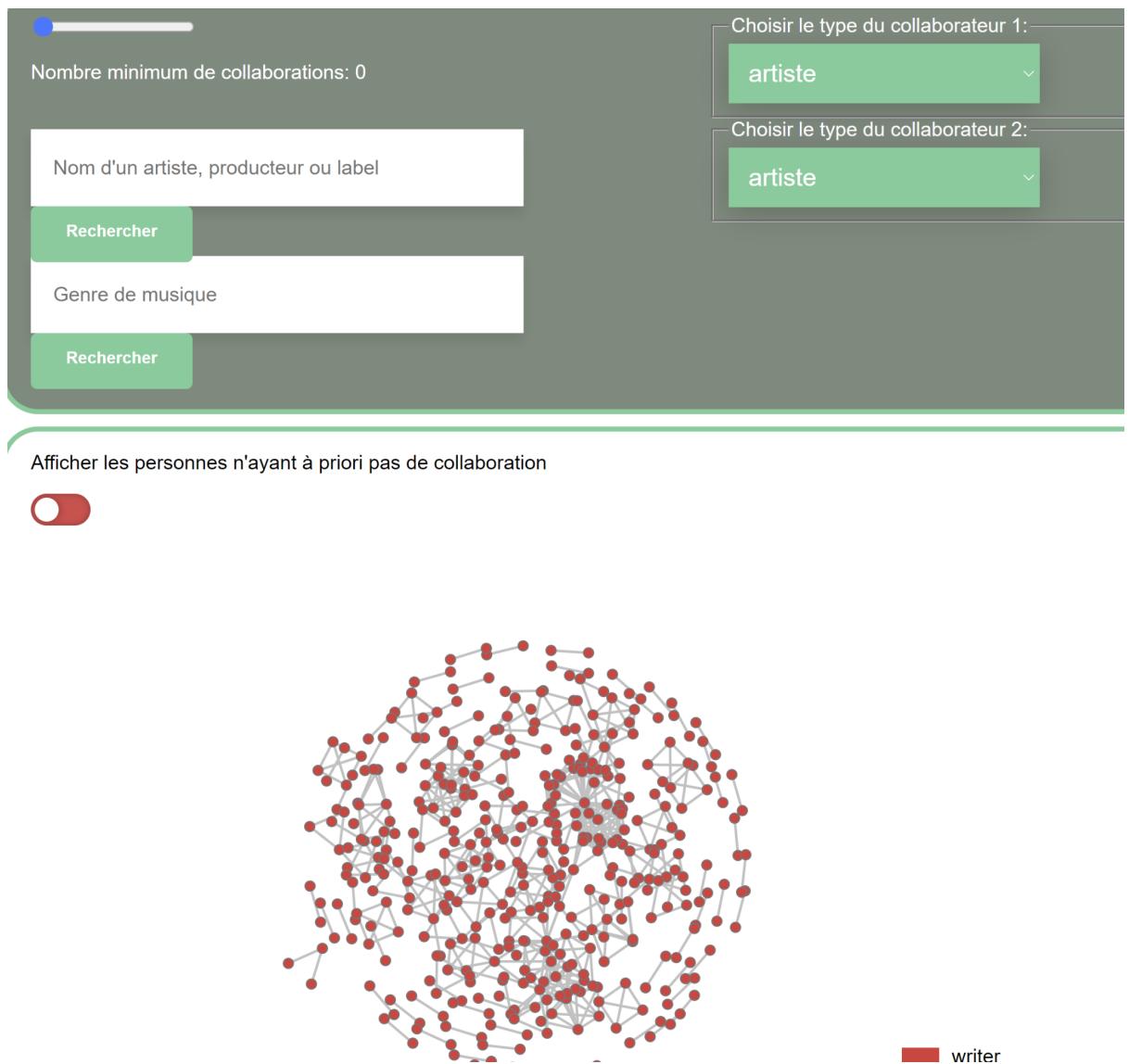
Les filtres peuvent être combinés. Notamment filtrer un genre musical et un nom. Dans l'exemple ci-dessous on cherche les collaborations de U2 dans le genre pop. C'est pour cela que le nœud représente U2 possède moins de liens.

The screenshot shows a search interface with the following elements:

- A slider for "Nombre minimum de collaborations: 0".
- A search input field containing "U2".
- A "Rechercher" button.
- A second search input field containing "Pop".
- A "Rechercher" button.
- Two dropdown menus labeled "Choisir le type du collaborateur 1:" and "Choisir le type du collaborateur 2:", both set to "all".
- A toggle switch labeled "Afficher les personnes n'ayant à priori pas de collaboration" (Display people who have no prior collaboration) which is turned off.
- A network graph node for "U2" with the label "Nom: U2" and "label: writer".

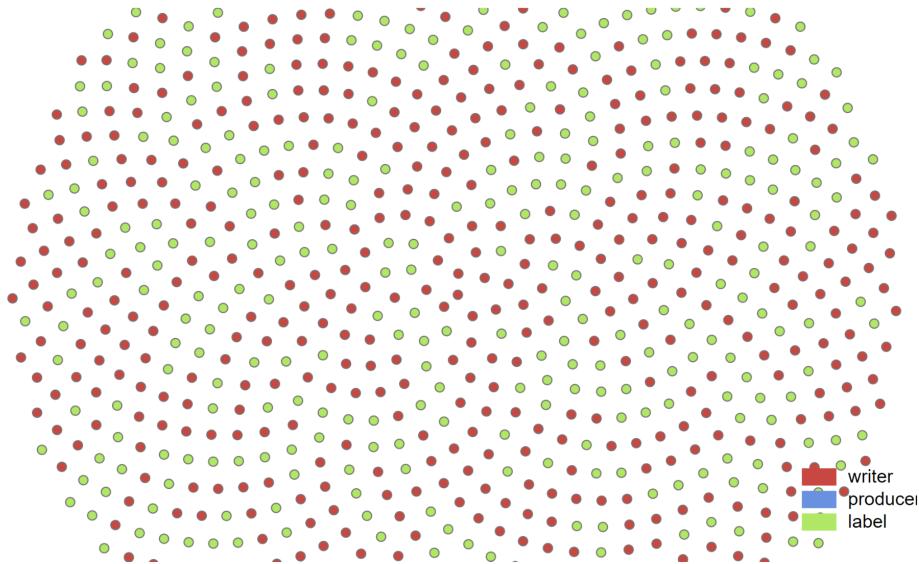
Enfin il est possible de filtrer par genre avec les 2 collaborateurs (artiste, producteur, label).

Dans l'exemple ci-dessous nous avons décidé d'afficher uniquement les collaborations entre artistes.



Les artistes n'ayant pas à priori de collaboration sont affichables également à l'aide d'un bouton activable ou désactivable.

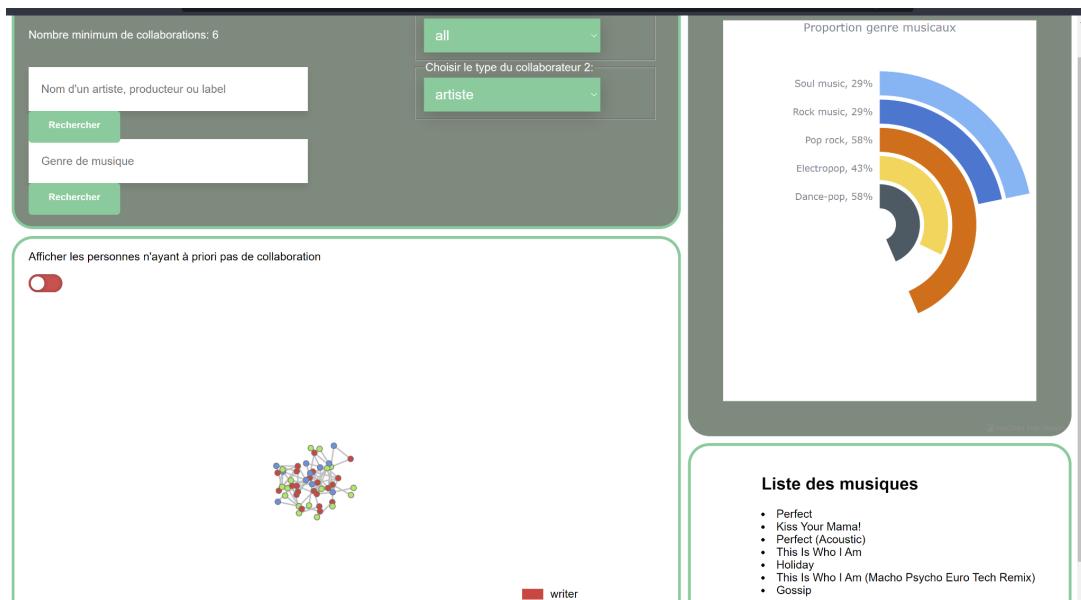
Afficher les personnes n'ayant à priori pas de collaboration



Visualisation Radial Bar chart

Details on demand (par Antoine)

Nous allons voir un nouveau type de visualisation, une visualisation chart complète celle du network. Elle permet de visualiser la liste des musiques réalisées lors des collaborations entre les artistes. Elle permet également d'afficher l'ensemble des genres et leur proportion. Pour cette deuxième visualisation nous avions choisi le radial bar chart. Ce chart permet de mettre en avant toutes les proportions de genre musicaux recensées dans une collab donnée. Il convient parfaitement à notre type de données puisqu'on possède une liste de genres et une liste des pourcentages correspondant à la proportion de chacun de ces genres. Or un graphique du type pie chart n'aurait pas pu convenir car la somme des proportions de tous les genres n'est pas égale à 100%. Cela s'explique par le fait qu'une chanson peut s'inscrire dans plusieurs genres à la fois.



Enfin il y a bien évidemment autant de titres musicaux que nombre de collaborations. Car le nombre de collaborations est compté à partir du nombre de titres musicaux. Ces titres sont affichés à l'aide d'une liste à puces en dessous du graphique. Les données du radial bar chart et de la liste à puce se mettent donc à jour dynamiquement à chaque fois que l'utilisateur passe sa souris sur une nouvelle collaboration depuis le network graph (overview).

Implémentation

La visualisation network a été réalisée en utilisant le langage JS avec la librairie 3djs notamment à l'aide de la librairie force. Le principe a été de charger les données des nœuds des collaborateurs, des liens des collaborations et des nœuds des personnes n'ayant à priori pas collaboré. Tout ceci était soumis à des conditions qui respectaient les filtres. Ces données sont chargées grâce à ce genre de JSON (voir ci-dessous).



```
data.json
{
  "nodes": [...],
  "links": [...],
  "unknown_collabs": [...]
}
```

Pour le chart, son implémentation a pu se faire grâce à la librairie javascript anychart et aux données lues dans links notamment les champs genres et percent (voir ci-dessous).

```
"genres": ["Rock and roll", "Gospel music", "Rhythm and blues"],
"percent": ["100", "100", "100"],
```

Dans l'exemple ci-dessus, chaque genre à son pourcentage associé. Il est possible d'avoir plusieurs mêmes pourcentages pour plusieurs genres car une collaboration peut représenter plusieurs genres musicaux.

Conclusion

Ce projet nous a permis de comprendre et mettre en application les principes fondamentaux d'une visualisation. Notamment d'utiliser le mantra de Schneiderman's Mantra “overview, filter and zoom and details on demand”.

Nous avons également pu mettre en place un chaîne de visualisation l'acquisition des données, l'enrichissement des données, l'abstraction des données pour l'affichage enfin la visualisation.

Enfin nous avons gagné en compétence notamment en termes de librairie de visualisation JS et de manipulation de données en R.