



SXS

Projet Ingénierie et Entreprise

---

# **Machine Learning pour la valorisation de données vol ADS-B**

---

*Auteurs :*

Alexis VANDEWALLE

Naihao LIU

Ziqing WU

Loïc TIBERGHEN

Baptiste

GAUDRON-DESJARDINS

Zhengyu JIANG

*Encadrants :*

Nicolas CANOUE

Matthieu BESSAC

Version 0.1 du  
9 mars 2021

# Table des matières

<b>Introduction</b>	<b>1</b>
Contexte . . . . .	1
Objectifs . . . . .	1
Parties prenantes et Livrables . . . . .	2
Contraintes associées au projet . . . . .	2
<b>Organisation du projet</b>	<b>4</b>
Work Breakdown Structure et Planning . . . . .	4
Association des tâches aux membres du projet et Organigramme . . . . .	4
Suivi . . . . .	5
Gestion des risques et Opportunités . . . . .	7
<b>Développement du projet</b>	<b>8</b>
Récupération et réorganisation des données . . . . .	8
Calcul des descripteurs . . . . .	9
Développement de l'application Web . . . . .	10
Classification des compagnies aériennes . . . . .	10
<b>Résultats</b>	<b>12</b>
<b>Conclusion et perspectives</b>	<b>13</b>

# Introduction

## Contexte

L'entreprise Liebherr est une entreprise familiale allemande fondée en 1949. En 2016, Liebherr employait près de 41000 personnes et son chiffre d'affaire s'élevait à 9 milliard d'euros. Son domaine d'activité est assez varié puisqu'elle fabrique des engins de construction, des machine-outils, des réfrigérateurs ou encore des équipements aéronautiques. Liebherr commercialise notamment des systèmes d'air conditionné qui équipent la totalité des avions de la gamme A320 et assure également leur maintenance.

Liebherr a constaté que plusieurs de ses clients rencontrent parfois les mêmes problèmes de maintenance sur ces systèmes et suppose que ceux-ci sont dû au mode d'utilisation des avions. Afin d'être plus efficace dans sa gestion de la maintenance, Liebherr a souhaité améliorer le suivi des vols des avions contenant leurs systèmes.

Pour cela, il existe des données open-sources qu'il est possible d'exploiter notamment grâce à l'Automatic Dependent Surveillance-Broadcast (ADS-B). C'est un système de surveillance coopératif pour le contrôle du trafic aérien, qui regroupe la position, vitesse de montée, vitesse de descente et altitude envoyés par les avions à des stations positionnées au sol.

## Objectifs

Le but de notre projet est d'établir une classification des compagnies aériennes par modes d'opérations. Ils ont à priori un impact important sur le design des systèmes d'air commercialisés par la société et peuvent expliquer les problèmes rencontrés en service par les compagnies aériennes.

Afin de réaliser cette classification, nous souhaitons exploiter les données de vol ADS-B. Notre objectif principal sera d'établir une classification des compagnies aériennes par modes d'opérations similaires. Cela implique :

- extraire des données ADS-B les données de chaque avion de type A320
- calculer les phases de vol pour chaque vol

- trouver des descripteurs et les calculer pour chaque phase de vol
- réaliser une application web pour avoir un aperçu global des données calculées
- établir la classification finale à partir des données calculées pour chaque vol

## Parties prenantes et Livrables

Notre client est la société Liebherr représentée par Nicolas Canouet qui est chargé de l'encadrement du projet. D'un point de vue gestion de projet, Matthieu Bessac se charge de nous accompagner.

Les livrables demandés sont présentés ci-dessous :

- un plan de développement cohérent
- un compte-rendu répondant au problème initialement formulé par le client
- une soutenance PIE présentant le projet et son déroulement
- les codes sources commentés et documentés, afin qu'ils soient facilement utilisables par le client

## Contraintes associées au projet

Notre projet se limite à l'étude des données produites par les avions de la gamme A320 puisque notre client ne cherche qu'à établir une classification pour ce type d'avion.

Pour le reste, la forme des rendus par le client est assez libre. Nous prévoyons donc de présenter nos résultats sous la forme d'une application web accompagnée d'un rapport. Cependant, les personnes impliquées dans le projet ne seront plus disponibles une fois le projet terminé. Nous nous devons donc de faire en sorte que nos rendus soient les plus clairs possible afin que le client puisse exploiter au mieux nos résultats.

La figure 1 replace le projet dans son contexte à l'aide d'un diagramme pieuvre.

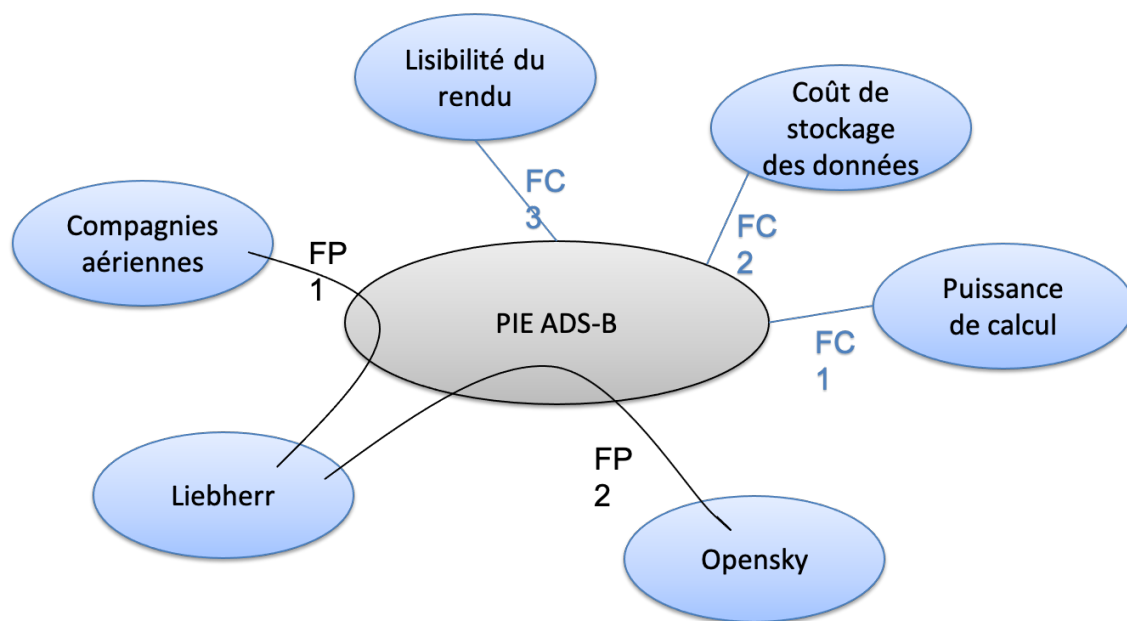


FIGURE 1 – Diagramme pieuvre du projet. FP1 : classer les compagnies en fonction de leurs modes d’opération, FP2 : exploiter les données ADS-B, FC1 : ne pas dépasser les ressources informatiques disponibles, FC2 : respecter les coûts associés au stockage des données, FC3 : rendre un projet exploitable par le client.

# Organisation du projet

## Work Breakdown Structure et Planning

Afin que chacun puisse s'investir dans le projet, il était nécessaire de le diviser en tâches simples et de bien cerner l'ensemble des choses à faire. Le WBS en figure 2 présente ces principales tâches, qui seront réparties à chacun des membres du projet dans la section suivante.

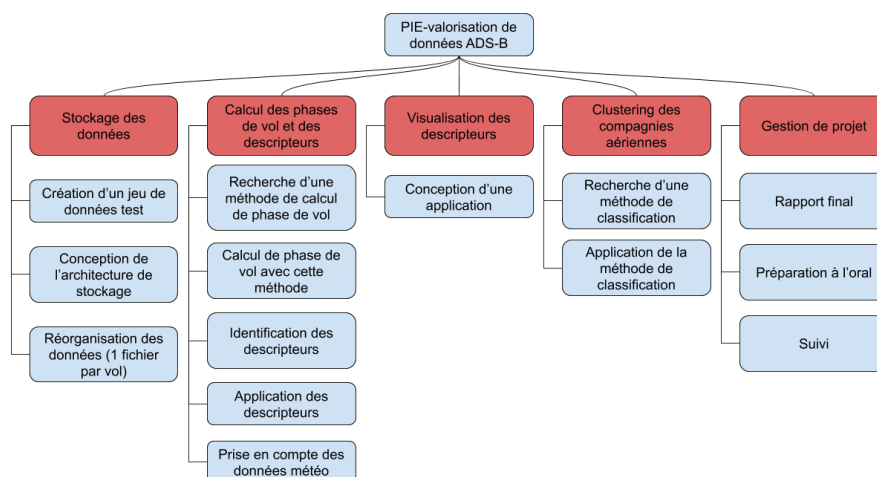


FIGURE 2 – WBS

Par ailleurs, il était important d'agencer ces travaux dans le temps afin de bien respecter les délais, ainsi que de voir ce qui pouvait être fait en parallèle. Les figures 3, 5 et 4 montrent le planning que nous avons construit et suivi au long de l'année.

## Association des tâches aux membres du projet et Organigramme

Pour la répartition des travaux, un organigramme ainsi qu'une matrice d'attribution des tâches à chaque membre du projet ont été effectués et présentés en figure 6 et 7.

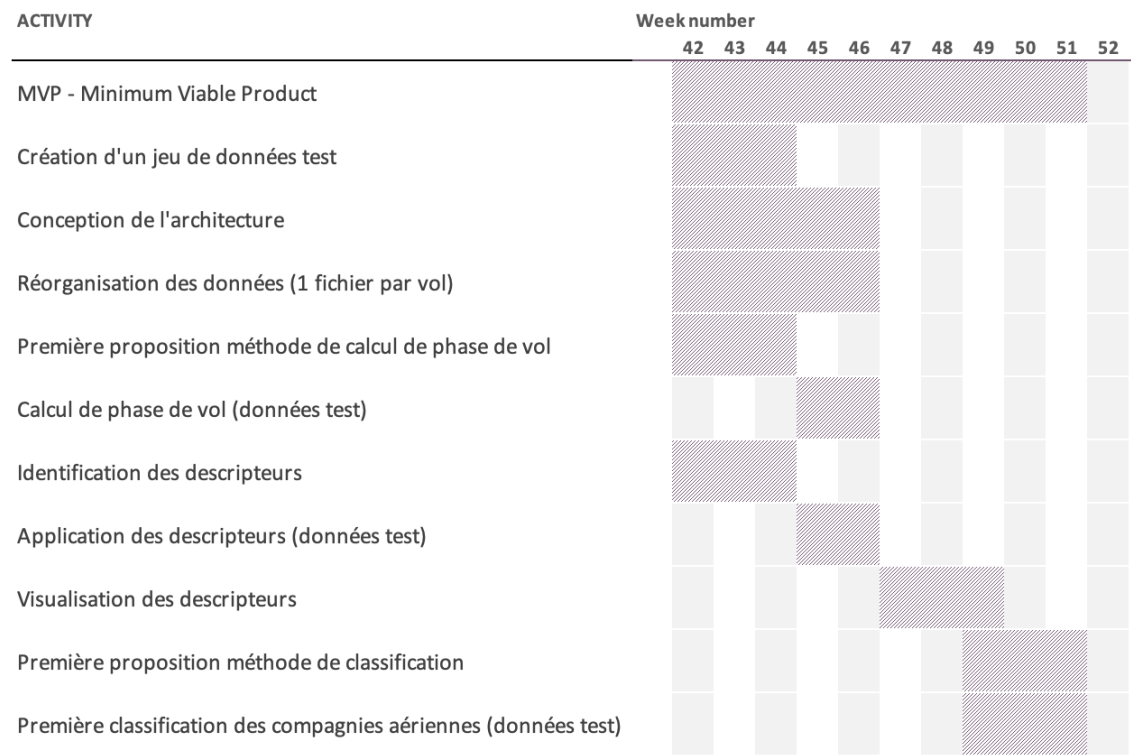


FIGURE 3 – planning associé au produit minimal(MVP)

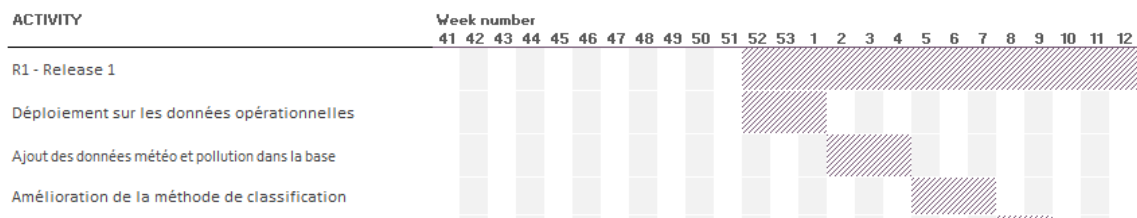


FIGURE 4 – planning associé au produit final

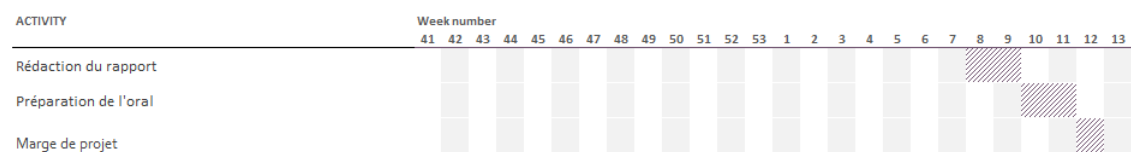


FIGURE 5 – planning associé à la gestion de projet

## Suivi

Afin que la communication au sein du projet soit de bonne qualité, nous avons mis en place un drive google où sont stockés les documents qui concernent la gestion du projet (cahier charge,

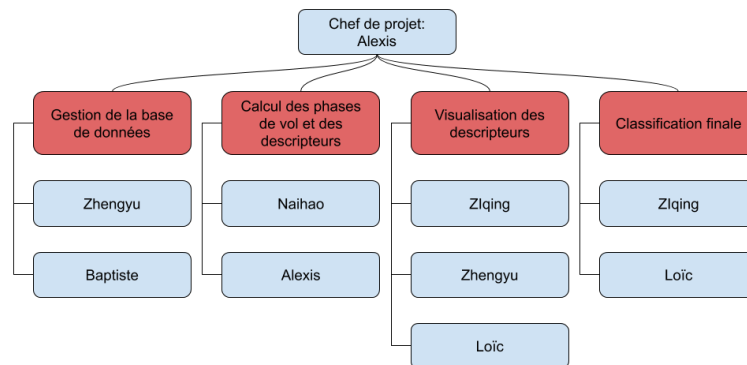


FIGURE 6 – Organigramme

		Alexis	Baptiste	Naihao	Loïc	Zhengyu	Ziqing
MVP	Création d'un jeu de données test	I	R	I	I	A	I
	Conception de l'architecture	C	R	C	C	A	C
	Réorganisation des données (1 fichier par vol)	C	R	C	C	A	C
	Première proposition méthode de calcul de phase de	R	C	A	C	C	C
	Calcul de phase de vol (données test)	R	I	A	I	I	I
	Identification des descripteurs	R	C	A	C	C	C
	Application des descripteurs (données test)	R	I	A	I	I	I
	Visualisation des descripteurs	C	C	C	R	R	A
	Première proposition méthode de classification	C	C	C	A	C	R
R1	Première classification des compagnies aériennes	I	I	I	A	I	R
	Déploiement sur les données opérationnelles	A	R	R	R	R	R
	Ajout des données météo et pollution dans la base	R	R	A	R	R	R
	Amélioration de la méthode de classification	R	R	R	A	R	R
Gestion de projet	Rédaction du rapport	A	R	R	R	R	R
	Préparation de l'oral	R	R	R	A	R	R
	Suivi du projet	R	R	R	R	R	A

FIGURE 7 – matrice RACI, R :responsable, A :accountable, C :consulted, I :informed

compte rendu de réunion,...). Par ailleurs, nous avons mis en place un répertoire github pour assurer les travaux de développement et de rédaction des documents qui sont fait en équipe. Ces deux répertoires sont accessibles et modifiables depuis n'importe quel poste et par n'importe quel membre du projet.

Chaque semaine, une réunion a été organisé ce qui a permis de faire un point régulier de l'avancement du projet ainsi que sur la répartition des tâches restantes au sein de l'équipe.

D'autre part, nous avons utilisé la librairie pandas sous python pour assurer le traitement des données, ainsi que django pour réaliser une applications web.



## Gestion des risques et Opportunités

La gestion des risques est un point clé dans le succès du projet car elle permet d'anticiper des actions correctrices de manière à ce que les risques identifiés arrivent avec la probabilité la plus faible possible. Nous avons identifié les risques suivant :

- Perte des données. Une panne matériel peut survenir à tout moment et endommager les données. Ce risque semblait initialement assez faible car nous pensions pouvoir toujours récupérer les données en les téléchargeant une nouvelle fois. Or, avec la nouvelle méthode de récupération de données (en continu), cela n'était plus possible. Ainsi, il a été choisi de les stocker sur le serveur de l'école pour plus de sûreté.
- Difficulté d'accès aux données. Si les données sont stockées sur un ordinateur personnel, il y a un risque que cela ralentisse le développement des codes de calculs. C'est pourquoi nous avons envisagé de stocker les données sur un disque accessible depuis n'importe quel ordinateur de l'école.
- Logiciel inconnu par certain membre du groupe. Nous avons du prendre en compte le fait que tous les membres ne connaissent pas les outils et librairies utilisées (Usage de git, flask, panda...) et donc évaluer le temps de travail en fonction de cela.
- Crise du coronavirus et mauvaise communication interne : Dans le contexte actuel, il est très probable qu'un des membres tombe malade. Nous avons donc fait en sorte que la majorité des tâches puissent se faire depuis un ordinateur personnel (usage de git). De plus, la communication entre les membres est plus difficile. Pour communiquer régulièrement entre nous sur l'avancement du projet, nous avons mis en place un groupe de conversation messenger ainsi qu'un drive Google où sont déposés les documents relatifs à la gestion du projet.

Nous avons listé ci-dessous des opportunités qui peuvent nous permettre d'améliorer le développement du projet.

- Matériel informatique de l'école. L'école dispose de ressources informatiques importantes qui peuvent nous permettre d'effectuer des calculs rapidement et de stocker une quantité importante de données. Afin de profiter de cela, nous avons fait une demande de matériel à l'école.
- La librairie Openap permet de calculer les différentes phases d'un vol. En parvenant à l'utiliser, on peut gagner beaucoup sur cette tâche du projet. Elle est de plus très bien documentée, son utilisation sera donc probablement assez simple.

# Développement du projet

Ce chapitre présente les différentes étapes techniques qui ont été réalisées pour mener à bien le projet.

## Récupération et réorganisation des données

La figure 8 résume ce qui a été fait au niveau du traitement des données :

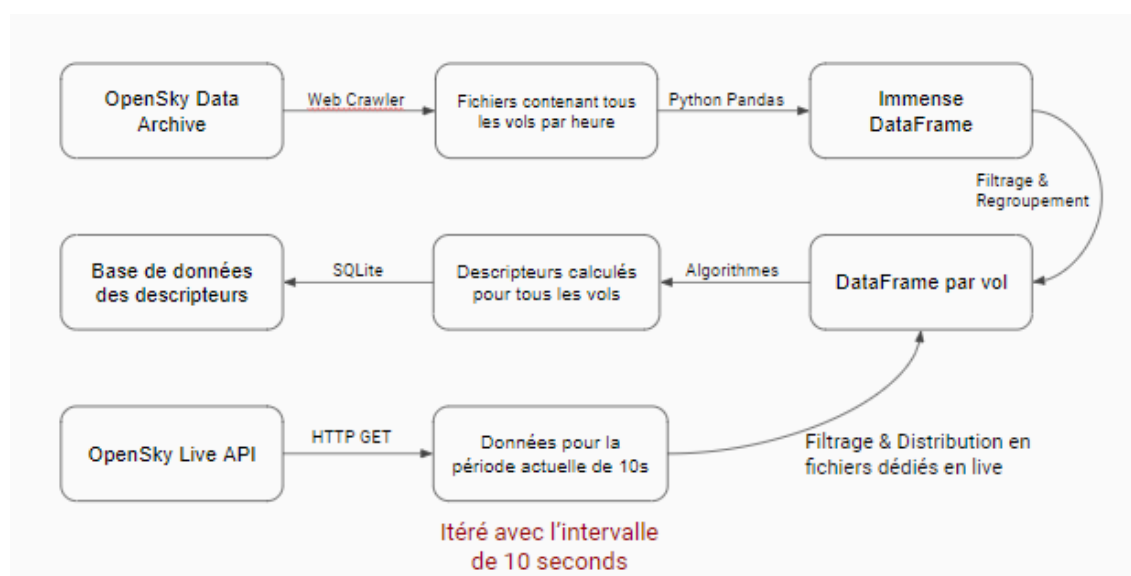


FIGURE 8 – diagramme organisation des données

Initialement, les données de vols étaient stockées sur le site d'OpenSky à l'adresse "<https://opensky-network.org/>". C'est un site qui stocke les fichiers dans lesquels sont enregistrés à un instant fixé une photo de l'état du ciel, c'est à dire toutes les informations des avions qui volent (identité, position, vitesse, direction...).

Ce mode de stockage était très difficilement exploitable pour 2 raisons :

- L'accès aux données se faisait sur Internet
- Les informations étaient classées temporellement, alors qu'il est plus simple de faire des analyses lorsque les fichiers portent sur l'intégralité d'un seul vol

Nous avons donc utilisé tout d'abord un web crawler pour regrouper les données dans des fichiers contenant les informations sur les vols par heure. Puis à l'aide de la librairie Pandas de python, nous avons pu construire une grande base de données, pour ensuite la réorganiser afin d'obtenir un fichier par vol.

Cette manière de fonctionner nous a été utile pour la première partie de notre projet : cela nous a permis d'avancer dans le prototypage des solutions suivantes tels que le calcul des phase de vol et des descripteurs. Cependant, nous ne nous sommes pas arrêtés à cette manière de fonctionner afin de pouvoir récupérer plus de vols. En effet, le site d'OpenSky ne fournit que les données de vols des lundis des 6 derniers mois, ce qui est limitant quand à la taille de la base de données. De plus, nous avons voulu dans un second temps intégrer les données météo dans notre analyse.

Nous avons donc développé 2 scripts pour récupérer ces données de manière continue ce qui est avéré être efficace (données ADS récupérées toutes les 10 secondes, données météo récupérées toutes les heures). Finalement, nous avons obtenu autour de 100000 vols pour la base de données, ce qui est une base de données convenable pour faire de la classification.

## Calcul des descripteurs

Une fois la réorganisation des données effectuée, nous avons essayé d'identifier les descripteurs qui pourraient être pertinents pour classer les avions. Entre autres, nous avons repérés :

- La durée du vol
- La vitesse moyenne et sa variance, son minimum et son maximum
- La vitesse de montée et sa variance, son minimum et son maximum
- La variation de hauteur de l'avion durant la phase

Avant de faire l'analyse sur les descripteurs, il nous semblait pertinent de découper ces vols en plusieurs phase (décollage, montée, croisière, descente, atterrissage) afin d'en sortir des descripteurs pertinent et cohérent suivant le mode de vol. Pour cela, le fichier "flight-phase.py" parcourt les différents vols et utilise la fonction FlightPhase de la librairie openap de python pour déterminer les différentes phases pour chaque vols. Un travail de tri a aussi été effectué pour éviter de garder des données de mauvaise qualité (absence de valeurs et valeurs incohérentes). Etant donné que le fichier peut demander de longs calculs, celui-ci s'exécute sous la norme MPI pour paralléliser les calculs.

Avec les fichiers contenant les labels de phase de vols, il devient assez simple de calculer les descripteurs en utilisant les fonctions mean, std, min et max de python. Le fichier flight-descriptors.py les calcule pour chaque vol et pour chaque phase (avec certains descripteurs qui sont spécifiques à chaque phase). Ce fichier renvoie 4 tables SQL : une pour la phase de montée de tous les vols, une pour la phase de croisière, une pour la phase de descente, et une table donnant des informations générales relatives à chaque vol.

Une fois tous les descripteurs calculés, il reste à choisir une manière de les traiter ainsi qu'une visualisation pertinente afin de classifier les compagnies aériennes. Pour la forme des résultats, nous avons choisi de développer une application Web qui offrira beaucoup de possibilités de visualisation.

## Développement de l'application Web

Pour développer l'application Web, nous avons utilisé "Flask", un micro framework open-source de développement web en Python.

## Classification des compagnies aériennes

Cette section présente les choix effectués pour la visualisation des descripteurs. La difficulté de cette partie résidait dans le traitement des descripteurs qui étaient au nombre de 33 et dans le choix de la visualisation.

Plusieurs analyses ont donc été effectuées. La première méthode que nous avons essayé est l'AFD (Analyse Factorielle Discriminante).

Nous avons aussi effectué une Classification Ascendante Hierarchique (CAH). Cette méthode prend en entrée le dataframe contenant tous les descripteurs et les normalise pour avoir des objets comparables. La CAH regroupe d'abord les compagnies aériennes ayant les caractéristiques les plus proches pour ensuite regrouper celles les plus éloignées. Cela conduit à l'obtention d'un dendrogramme comme présenté figure ... Une grande longueur de trait correspond à un grand nombre de caractéristiques différentes entre les groupes. Afin d'effectuer des clusters de compagnies aériennes pertinents, il reste à effectuer des coupes à ces endroits là, car c'est là où il y a beaucoup de caractéristiques différentes entre les groupes. Cette méthode est pratique et assez visuelle pour la classification.

Enfin, une dernière méthode utilisant la divergence de Jensen-Shannon a été utilisée. C'est une méthode qui mesure les similarités entre plusieurs densité de probabilités. Pour un descripteur fixé, nous pouvons créer pour chaque compagnie aérienne la distribution de ce descripteur autour de sa médiane. En sélectionnant les compagnies aérienne souhaitées, nous obtenons la corrélation entre ces compagnies pour le descripteur étudié.

# Résultats

## **Conclusion et perspectives**