**Test technique**

**Introduction**

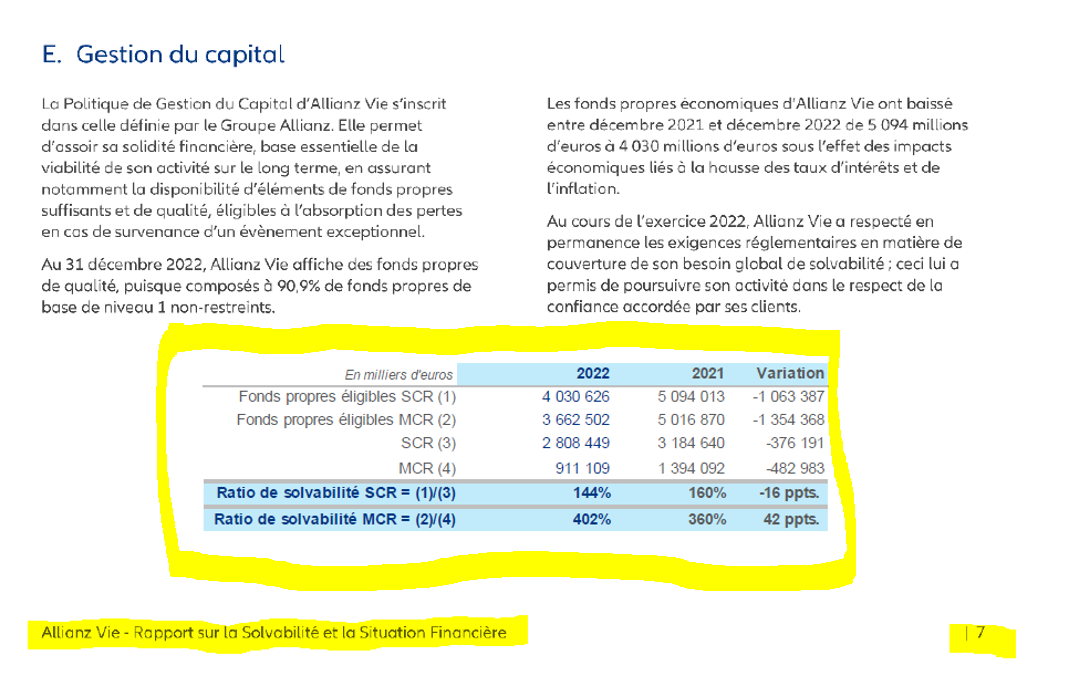
Nous vous remercions de l'intérêt que vous portez à notre offre de stage. Dans le cadre du processus de sélection, nous avons élaboré un test technique visant à évaluer votre aisance en Python ainsi que vos compétences en résolution de problématiques, en mettant particulièrement l'accent sur la présentation des résultats, la pertinence des traitements effectués, et la qualité du code rendu.

**Présentation de la problématique**

Dans les sujets NLP, un Data Scientiste est fréquemment confronté à la lecture de documents dans divers formats, notamment au format PDF. Les défis rencontrés concernent souvent les résultats produits par les OCR [[1]](#footnote-1) utilisés pour lire le contenu des documents, en particulier :

1. La présence d'informations inutiles, telles que **les bas de pages et les hauts de pages,** dans le texte **reconstruit** à partir de l'OCR.

2. La rétention d'informations non structurées provenant de tableaux, nécessitant souvent une séparation du contenu contextuel des paragraphes du contenu des tableaux et des graphes.



**L'objectif de cet exercice est de récupérer uniquement le contenu utile des rapports, excluant les informations indésirables notamment les bas de pages, les hauts de pages, les informations provenant des tableaux et des graphes.**

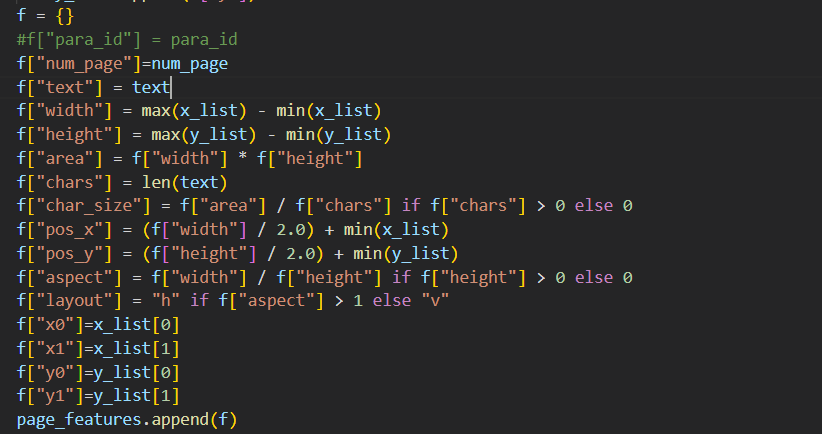
Les rapports que nous traitons pour cet exercice, sont les rapports SFCR des assureurs. Ce sont des rapports publiés périodiquement par les compagnies d’assurance, pour fournir des informations sur la situation financière, la solvabilité et la gestion des risques de l'entreprise.

Nous avons préalablement utilisé pour vous, l'OCR API Vision de Google Cloud Platform (GCP) pour extraire le contenu brut des rapports SFCR au format JSON. Vous trouverez les rapports au format PDF dans le sous dossier **data/pdfs/**. Vous trouverez également les fichiers JSON produits par l’OCR dans le sous dossier **data/ocr/**

L'avantage de l’utilisation de l’API Vision de GCP, réside dans la possibilité d'obtenir des informations supplémentaires telles que la taille des caractères et la position des caractères.

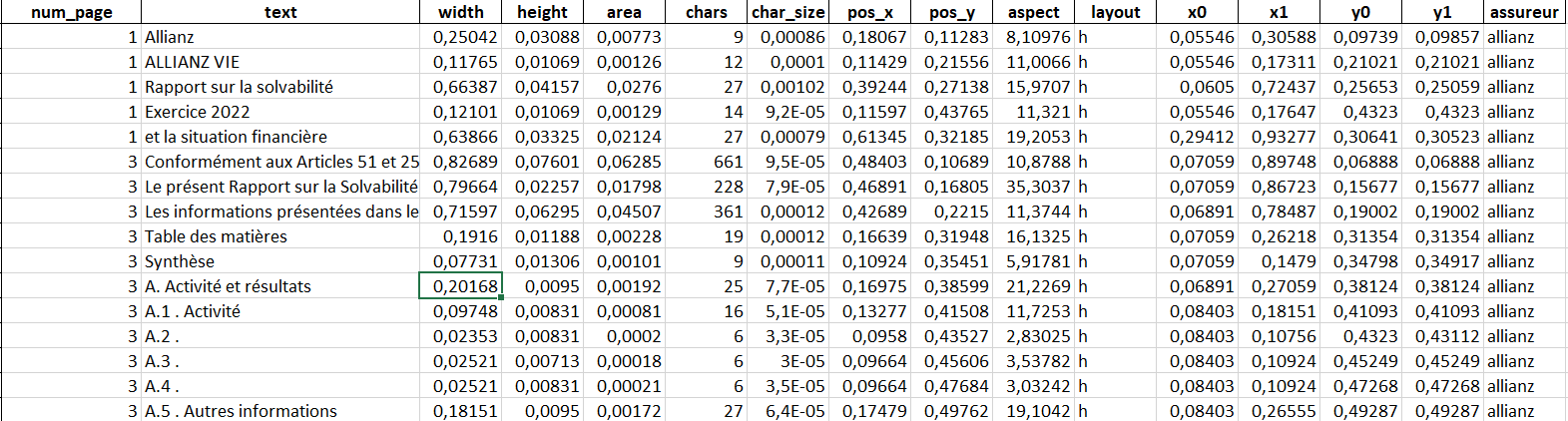
**Déroulement de l'exercice :**

Nous vous fournissons le code **« helper.py** », qui contient la fonction « ***produce\_brut()*** » pour exploiter les fichiers JSON générés par l'OCR. Cette fonction produit un tableau Excel des blocs textuels inclut dans le rapport, avec un maximum d'informations sur le rapport. Les informations inclues dans le rapport sont définies dans le bloc suivant de la fonction :



Vous avez la liberté de modifier le bloc (dans l’image ci-dessus), si nécessaire, afin de peaufiner les statistiques existantes ou d'incorporer d'autres données pertinentes à vos yeux pour la suite de l'exercice.

Après application de la fonction ***produce\_brut()*** à un JSON produit par l’OCR, vous aurez un exemple de tableau suivant :



Votre tâche consiste à mettre en place une fonction capable de :

* Détecter le contenu inutile des rapports, notamment les bas de pages, les hauts de pages et le contenu des tableaux.
* Identifier les paragraphes.
* Repérer les grands titres des différentes parties du rapport.

La fonction que vous développerez prendra en entrée l'output de `produce\_brut()` et ajoutera une nouvelle colonne, "Label", au tableau résultant. Les valeurs de cette colonne seront soit "Inutile", "Paragraphe", ou "Titre".

Il est impératif que cette fonction puisse être généralisée pour une labélisation automatique applicable à d'autres rapports SCFR. Nous mettons à votre disposition plusieurs exemples de SFCR ainsi que leurs rapports correspondants pour faciliter vos tests et permettre une solution robuste et généralisable.

Il est recommandé d’accompagner votre travail par une présentation PPT, pour illustrer les résultats de votre travail.

Enfin, il est attendu de créer un repo sur votre Github perso et de mettre l’ensemble de vos travaux sur ce repo, de nous envoyer le lien vers votre repo par mail afin que l’on puisse évaluer vos travaux.

**Indications :**

Vous pouvez explorer des stratégies basées sur la taille des caractères et la longueur des blocs pour proposer une première solution de détection des titres, des paragraphes et des contenus inutiles.

Toute initiative ou proposition pertinente sera grandement appréciée et pourra faire la différence dans l'évaluation de votre test.

La propreté du code ainsi que sa documentation seront également des critères évalués.

Nous vous souhaitons bon courage et restons à votre disposition pour toute clarification nécessaire.

1. OCR : logiciels de Reconnaissance Optique de Caractères [↑](#footnote-ref-1)