Traitement de données massives

Projet Partie I

Victor Durand
Tristan Chrétien
4 ICS

Sommaire

Objectif du projet

L'objectif de ce projet est de recommander des images en fonction des préférences utilisateurs basées sur les caractéristiques des images sur un jeu de données défini. Le programme aura la forme d'un Notebook Jupyter documenté.

Sources des données

Le dataset sélectionné provient de la plateforme Kaggle.com, cette dernière propose des ressources et des outils pour Data Scientist. Nous nous sommes tournés vers le dataset "Pokemon Image Dataset" proposé par *VISHAL SUBBIAH* sous licence *Attribution 4.0 International (CC BY 4.0)*. Le téléchargement de ce dataset utilise l'API proposée par Kaggle afin de s'authentifier lors du téléchargement.

Le dataset Pokemon Image Dataset est accessible via le lien suivant :
https://www.kaggle.com/datasets/vishalsubbiah/pokemon-images-and-types

Taille du Dataset

Le dataset "Pokemon Image Dataset" contient à ce jour 809 images dont 721 au format PNG et 88 au format JPG. Toutes les images possèdent une taille de 120x120 px portant le poids du dataset à 2.5Mo.

Métadonnées des images

Les images du dataset ne présentent aucune données EXIF, par conséquent nous n'avons pu extraire que la taille de l'image. Fort heureusement, ce dataset contient un fichier CSV contenant des caractéristiques sur chaque Pokémon représenté par l'image. Les métadonnées extraites de ce fichier sont donc son nom, son type principal et son type secondaire. La taille de l'image, ses deux couleurs principales et son chemin dans l'arborescence de fichier sont quant à elles extraites directement depuis l'image ou par le programme. Toutes ces métadonnées sont ensuite stockées au format JSON dans un unique fichier nommé metadata.json.

Préférences utilisateurs

Les préférences utilisateurs sont générées aléatoirement pour un nombre d'utilisateurs défini en paramètre. Par défaut, 10 utilisateurs sont créés, chacun possède 8 images favorites et 8 détestées.
Etant donné que la génération des images favorites est aléatoire, il faut que les images détestées ne fassent pas partie des images favorites. Les métadonnées de chaque image sélectionnées sont regroupées dans un fichier JSON nommé *user_preferences.json* regroupant ainsi les préférences de tous les utilisateurs (paires de type préférés, id des images favorites et détestés ainsi que les couleurs favorites).

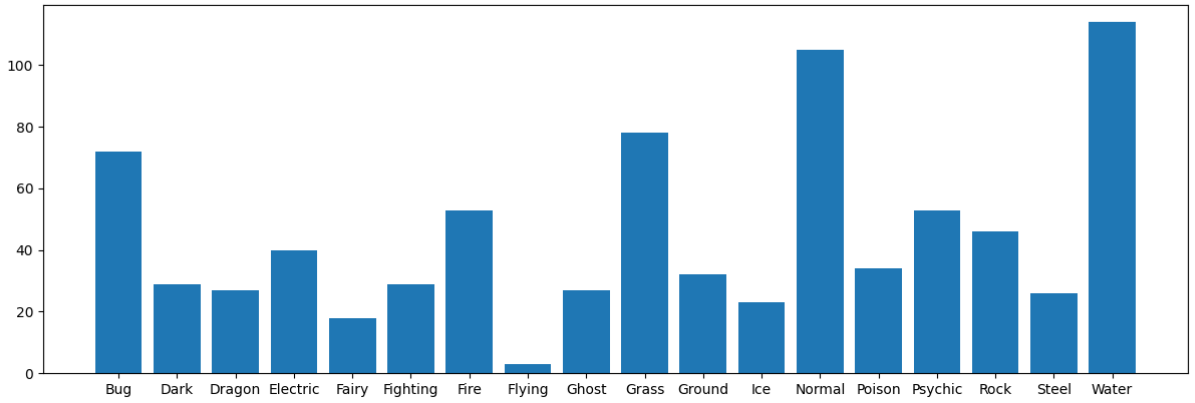
Analyse des images et métriques

L'analyse des données, de ce data set a été essentiellement faite sur les couleurs et les catégories des Pokémon, en effet, lors de l'apprentissage, ça a été les métriques utilisées pour l'apprentissage des données avec les couleurs, les couleurs qui se ressemblent et les catégories du Pokémon, que nous avons transformé avec un LabelEncoder de sklearn pour faciliter l'apprentissage.
Ensuite pour classifier les images, nous avons ensuite utilisé en Decision tree classifier de sklearn qui est une méthode basée sur l'utilisation d'un arbre de décision comme modèle prédictif. Après apprentissage, nous avons trouvé pour chaque utilisateur un certain nombre d'images qui correspondaient aux métriques énoncées précédemment.

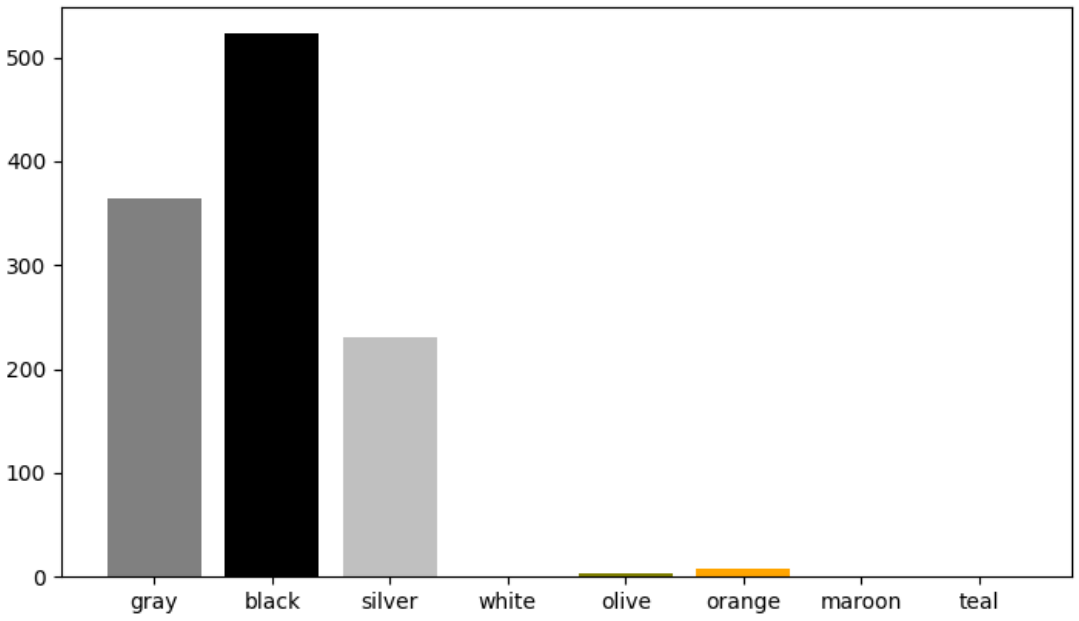
Visualisation des résultats

Afin de permettre une visualisation des données obtenues, nous avons générés 4 graphiques différents :

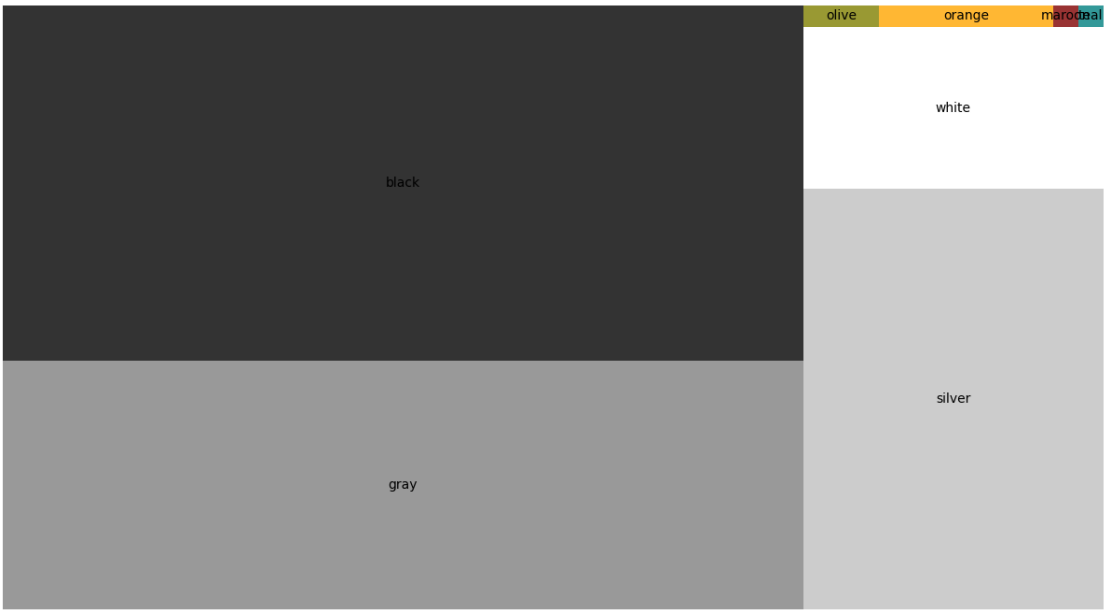
- Un affichage en diagramme bâton du nombre de Pokémon regroupé par type



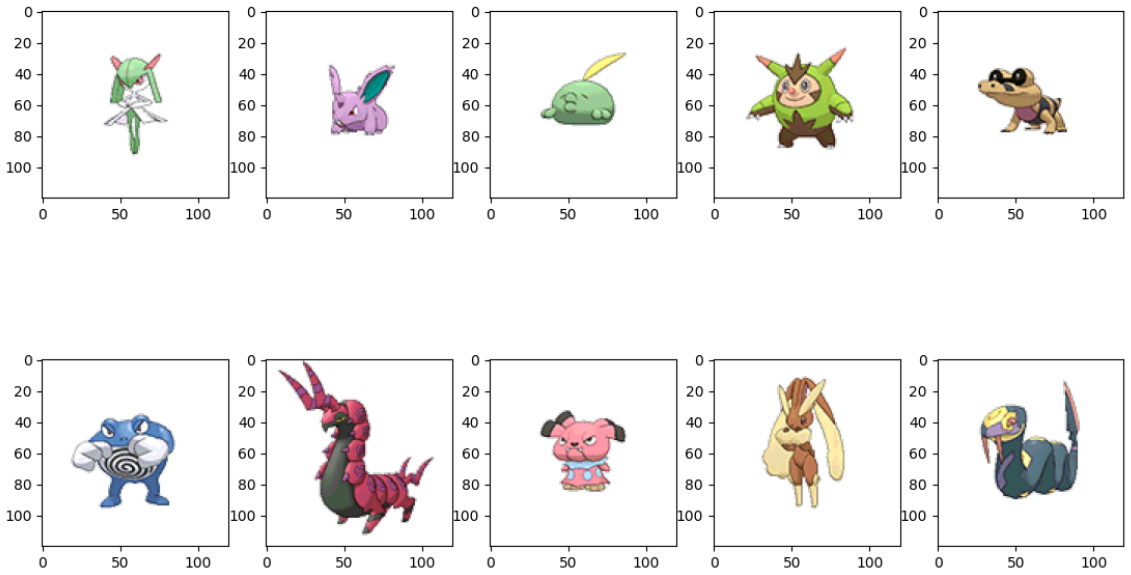
- Un affichage en diagramme du nombre d'image regroupé par couleur principale



- Le nombre d'image regroupé par couleur principale sous la forme de surface de couleur



- Une visualisation d’images du dataset sous un même graphe



Auto évaluation du travail

Nous avons fait une exploitation du mieux que nous pouvons sur nos mais notre dataset ne contenant pas que de métadonnées, ce qui nous a vite limité sur la recommandation. De plus, les images que nous avons utilisées ont un problème sur la couleur où l'on voit très bien que les principales sont noires, blanc, gris. Cela est peut-être dû à une nuance pas assez prononcé des couleurs ou encore le fond de, c'est dernier. Et une colonne tags que nous n'avons pas pu remplir dû au manque de données.

Notre algorithme pour le calcul des métadonnées sur les images, met un peu de temps à s'exécuter.
Nous avons utilisé plusieurs graphiques pour pouvoir avoir un maximum d'informations visuelles.

Nous avons créé des faux utilisateurs avec une multitude de données comme des images favorites, dislikes…
Mais ces derniers ne sont pas assez complets, ce qui peut entraîner des défaillances dans l'apprentissage.

Remarques

exercices et les possibilités d'amélioration.

Nous avons trouvé que les séances pratiques (TP) étaient peut-être un peu trop guidées ce qui complique l'analyse des éléments importants.

Conclusion

Pour conclure, ce projet nous a permis de voir la plupart des approches et méthode sur l'analyse des données avec ses différentes grandes étapes :

- Acquisition de données : récupérer le dataset sur kaggle, le tout automatiquement
- Extraction de données : extraire un fichier zip pour le transformer en en vrai dataset avec une multitude d'images.
- Nettoyage de données : Enlever les données vides ou inule
- Transformation de données : Acquisition des métadonnées sur les images, comme les couleurs principales
- Stockage de données : Stocker nos données des images dans un fichier Json
- Data analysis modeling : Système de recommandation
- Analyses de données : Traitement des données pour comprendre au maximum comment elle fonctionne et leurs significations
- Visualisation de données : avec les différents graphes obtenus, sur les études des couleurs notamment

Nous avons rencontré quelques problème au début pour la compréhension des dimensions sur les dataframes.