

## Projet Big Data

Loris / Gaetan / Samuel

Lien vers le dépôt github : [https://github.com/Darukity/BigData\\_Efrei](https://github.com/Darukity/BigData_Efrei)

### Prérequis réalisés

Tableau des prérequis réalisables ou non et pourquoi		
Prérequis	Status (réalisable ou non avec databrick community)	Commentaire
1. Apache Iceberg ou Hudi (à la place de Delta)	✗ Non réalisable	<ul style="list-style-type: none"><li>- Databricks Community Edition ne supporte pas nativement Iceberg ou Hudi.</li><li>- Ces formats nécessitent une configuration avancée ou des clusters premium.</li></ul>
2. Mise en place d'Airbyte	✗ Non réalisable	<ul style="list-style-type: none"><li>- Airbyte peut être installé indépendamment de Databricks et utilisé pour gérer l'ingestion.</li><li>- Cependant requiert des tokens générable uniquement par des comptes premium de databricks.</li></ul>
3. Use case Streaming avec Kafka/Redpanda	✗ Non réalisable	<ul style="list-style-type: none"><li>- La version Community Edition ne prend pas en charge les connecteurs de streaming comme Kafka/Redpanda.</li><li>- Requier des configurations avancées et des clusters premium.</li></ul>
4. Layer de compute différent (Druid/Dremio)	✗ Non réalisable	<ul style="list-style-type: none"><li>- La version Community Edition est limitée à Spark comme moteur de calcul.</li><li>- Druid ou Dremio nécessitent des configurations séparées non supportées par Databricks Community</li></ul>

Tableau des prérequis réalisables ou non et pourquoi		
		Edition.
5. Unity Catalog	✗ Non réalisable	<ul style="list-style-type: none"> <li>- Unity Catalog n'est pas disponible dans Databricks Community Edition.</li> <li>- Requiert une version Premium ou Entreprise.</li> </ul>
6. LakeFS ou Nessie pour le versioning	✗ Non réalisable	<ul style="list-style-type: none"> <li>- LakeFS ou Nessie peuvent être configurés indépendamment pour gérer le versioning des données sur un stockage compatible (comme Azure Blob Storage ou AWS S3).</li> <li>- L'intégration avec Databricks Community pourrait être limitée à des étapes manuelles.</li> </ul>

J'ai quand même essayé Airbyte :

1. installation de Airbyte en local avec docker, suivre la doc ou exécuter la commande suivante: `curl -L https://raw.githubusercontent.com/airbytehq/airbyte/master/run-ab-platform.sh -o run-ab-platform.sh && bash run-ab-platform.sh`
2. Uploader ses datasets sur un cloud type google Drive et créer des liens de partage pour tous

Mon Drive > datasets BigData ▾

Type ▾ Contacts ▾ Date de modification ▾ Source ▾

Nom	Propriétaire	Dernière modification	Taille du fich	
combats.csv	moi	14:34 moi	567 Ko	⋮
pokemon_stats.csv	moi	14:37 moi	43 Ko	⋮
pokemon_trading_cards.csv	moi	15:05 moi	1,4 Mo	⋮

3. Aller sur localhost:8000 (adresse d'airbyte) puis créer 3 nouvelles sources pour chaque dataset

Search			
All statuses			
NAME	CONNECTOR	DESTINATION	LAST SYNC
Pokemon_Combat	File (CSV, JSON, Excel, Feather, Parquet)	0	
Pokemon_stats	File (CSV, JSON, Excel, Feather, Parquet)	0	
File (CSV, JSON, Excel, Feather, Parquet)	File (CSV, JSON, Excel, Feather, Parquet)	0	

files

Pokemon\_Combat

File (CSV, JSON, Excel, Feather, Parquet) v0.5.3

Settings

Connections

Source Settings

Source name

Pokemon\_Combat

Dataset Name

combat.csv

File Format

csv

Storage Provider

HTTPS: Public Web

Optional fields

URL

https://drive.usercontent.google.com/download?id=1e-O4EgtADWzXH38zJG-3iHXjk-b4lyw&e

Optional fields

Test the source

Retest saved source


Delete this source

Cancel

Test and save

#### 4. Configurer Databricks comme destination (endroit où ça bloque)

Dans Airbyte pour pouvoir faire ingérer les datasets dans databricks on doit aller dans sources -> marketplace -> databricks lakehouse malheureusement on est bloqué ici car il nous demande un acces token qui ne peut pas être généré en version community voir : <https://community.databricks.com/t5/data-engineering/can-i-use-databricks-cli-with-community-edition/td-p/17394>

 Airbyte

Connections

Sources


Destinations

Builder


Settings


Help

Dark mode


Destination name 

Databricks Lakehouse


☒ Agree to the Databricks JDBC Driver Terms & Conditions 


Server Hostname 

community.cloud.databricks.com.


HTTP Path 


sql/protocolv1/o/3946793225956359/0105-141722-g5ltv23v


Access Token 




Required


Data Source 


[Recommended] Managed tables 

 Optional fields


Port  Optional


443

Databricks catalog  Optional

Default Schema  Optional

default

☐ Support schema evolution for all streams. 

☒ Purge Staging Files and Tables 

Settings

Workspace admin

Identity and access

Security

Compute

Notifications

Advanced

User

Profile

Preferences

Developer

Notifications

Profile

Manage your Databricks profile

Display name

Your display name

Samuel CHARTON (samuel.charton@efrei.net)

Groups

Your group memberships

admins

Password

Password for your Databricks account

Change

## **Le projet :**

Nous devons trouver des données à analyser. Effectuer des transformations/nettoyage et répondre à notre problématique.

## **Les données :**

**Nous avons utilisé 3 différents datasets pour notre projet, un dataset sur les combats, un dataset sur les cartes pokémons et un dataset sur les statistiques des pokémons en général (carte ou jeu)**

### **1. Pokémon Team Combat Dataset**

#### **Contexte**

Ce data set à été inspiré par la série et le jeu Pokémon. En explorant des datasets Pokémon, on a découvert plusieurs façons d'enrichir les données, par exemple en introduisant des combats d'équipes où deux équipes de 6 Pokémon s'affrontent.

#### **Contenu**

Ce dataset se compose de 4 fichiers CSV. Deux d'entre eux (*combat.csv* et *pokemon.csv*) proviennent de ce dataset : [Pokémon Challenge](#). Les deux autres fichiers ont été créés à partir d'un modèle conceptuel imaginé par le créateur du dataset : des combats d'équipes 1 contre 1.

Ce dataset contient des informations relatives aux combats Pokémon, incluant des combats 1 contre 1 et des combats d'équipe (6 contre 6).

#### **Dans notre contexte**

Nous n'avons utilisé que le dataset des combats en 1v1 pour cet exercice.

### **2. Pokémon Trading Cards Dataset**

#### **Contexte**

Le commerce des cartes Pokémon est une activité populaire et bien ancrée, que ce soit parmi les collectionneurs ou les joueurs. Ce dataset permet d'explorer les cartes Pokémon disponibles à la vente, leurs caractéristiques, et leur prix, offrant une perspective unique sur leur valeur économique en fonction de leur rareté, génération, popularité, ou numéro de carte.

#### **Contenu**

Le fichier **pokemon\_cards.csv** contient des informations sur les cartes Pokémon mises en vente sur le site [chaoscards.co.uk](#). Chaque entrée du dataset comprend :

**Pokemon** : Le nom du Pokémon représenté sur la carte.

**Card Type** : Le type de la carte (*Standard*, *Reverse Holo*, etc.).

**Generation** : La génération à laquelle la carte appartient.

**Card Number** : Le numéro de la carte dans la série à laquelle elle appartient.

**Price (£)** : Le prix de la carte en livres sterling.

## **Aperçu des données**

### **Répartition des types de cartes :**

**Standard** : 52 %

**Reverse Holo** : 37 %

**Autres** : 11 %

### **Répartition par génération :**

**Sword and Shield - Fusion Strike** : 2 %

**Sun and Moon - Cosmic Eclipse** : 2 %

**Autres** : 96 %

### **Répartition des prix :**

La majorité des cartes se situent dans une fourchette de prix modérée :

24 816 cartes sont comprises entre **0,09 £ et 45,09 £**.

Quelques cartes rares atteignent des prix bien plus élevés, allant jusqu'à **899,99 £**.

## **3. Pokémon Stats Dataset**

### **Contexte**

Ce dataset contient des informations détaillées sur 721 Pokémon, incluant leurs statistiques de base et leurs types. Il a été largement utilisé pour enseigner les bases des statistiques aux enfants et pour introduire de manière ludique le machine learning. Le dataset se concentre sur les jeux Pokémon traditionnels, excluant les cartes Pokémon ou Pokémon Go.

### **Contenu du Dataset**

Les colonnes du fichier incluent :

# (ID) : Identifiant unique pour chaque Pokémon.

Nom : Le nom du Pokémon.

Type 1 : Le type principal du Pokémon, influençant ses forces et faiblesses face aux attaques.

Type 2 : Un type secondaire pour certains Pokémon ayant une double typologie.

Total : La somme des statistiques de base, une indication générale de la puissance d'un Pokémon.

HP (Points de Vie) : Indique la quantité de dégâts qu'un Pokémon peut encaisser avant de s'évanouir.

Attack : La puissance des attaques physiques, comme *Scratch* ou *Punch*.

Defense : La résistance aux attaques physiques.

SP Atk (Attaque Spéciale) : La puissance des attaques spéciales, comme *Fire Blast* ou *Bubble Beam*.

SP Def (Défense Spéciale) : La résistance aux attaques spéciales.

Speed : Détermine quel Pokémon attaque en premier à chaque tour.

## Sources

Les données proviennent de plusieurs sites fiables :

[pokemon.com](http://pokemon.com)

[pokemondb](http://pokemondb)

bulbapedia

## Utilisation

Ce dataset peut être utilisé pour analyser les statistiques des Pokémon et leurs types.

Une question intéressante à étudier est de savoir si une combinaison de deux variables peut prédire le type d'un Pokémon. Cela pourrait être utilisé pour créer des exemples visuels d'apprentissage automatique dans des contextes éducatifs.

## Aperçu des Données

Répartition des Types :

Eau (Water) : 14 %

Normal : 12 %

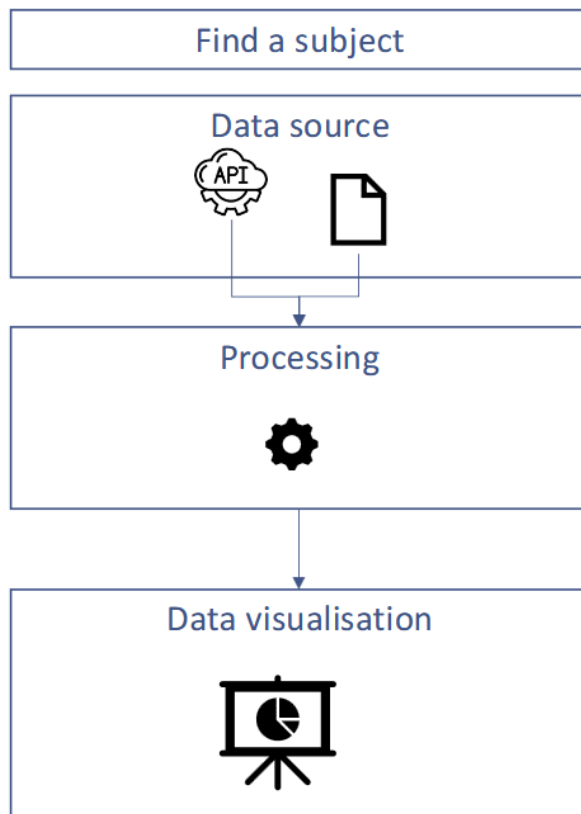
Autres : 74 %

Répartition des Scores Totaux :

Entre 180 et 420 : majorité des Pokémon.

Plus de 600 Pokémon très puissants (par exemple, légendaires).

### **Les traitements effectués :**



Nous avons effectué le projet de façon classique, en stockant tout d'abord les fichiers csv dans un conteneur ds-bronze sur azure et créer les conteneurs ds-silver et ds-gold.

Par la suite nous avons importé les données dans databricks c'est ce que fait le notebook 01 - Init Pokemon Ingest.

Ensuite nous avons nettoyé les données en enlevant les null, les pokémons avec des noms contenant Mega ou autre prefixe/suffixe pour "normaliser" les noms des pokémons à travers tous les datasets. C'est le rôle de la partie 02 - Pokemon Ingest.

Pour finir on créer les différentes tables qui vont nous permettre d'analyser différents points de nos données avec 03 - Intervention Process et on les visualise avec des cas concrets dans 04 - Exploitation.



## Points d'analyse

Les points d'analyses principaux ont été :

1. Voir quelle carte à les meilleures statistiques
2. Si ses stats influent sur son taux de victoire en 1 v 1 dans le jeu vidéo et à vérifier
3. Si les cartes les plus chères, sont forcément les plus fortes.

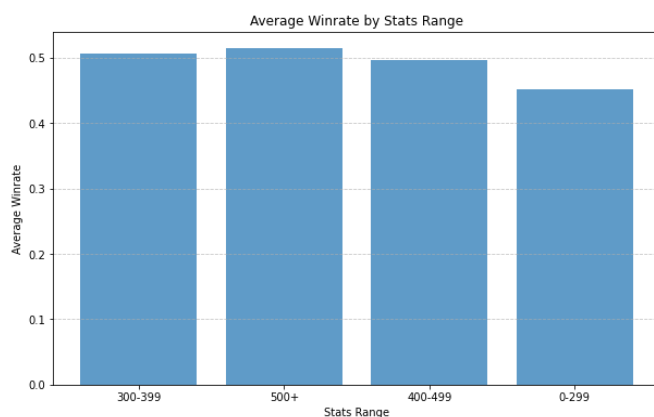
## Résultats des analyses

Grâce au traitement des données on peut arriver aux conclusions suivantes :

Top 10 des pokémons avec les meilleurs stats dans nos datasets avec leurs win rate :

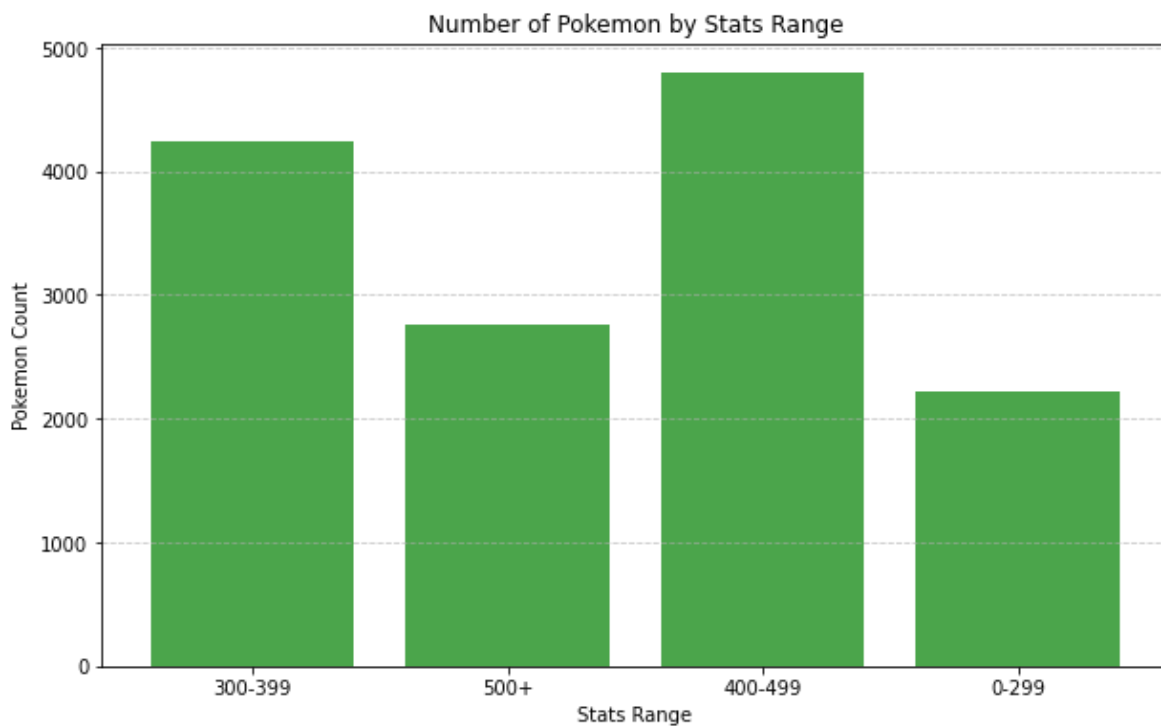
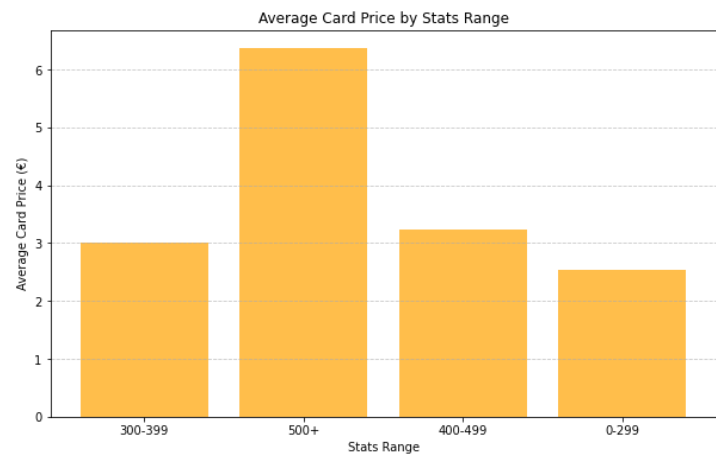
Pokemon	Total_Stats	Total_Combat	Win_Rate
Arceus	720	61	0.5901639344262295
KyuremWhite Kyurem	700	67	0.6716417910447762
Palkia	680	57	0.7368421052631579
Reshiram	680	58	0.8448275862068966
HoopaHoop Unbound	680	50	0.5
Ho-oh	680	62	0.6290322580645161
Yveltal	680	59	0.9322033898305084
Dialga	680	45	0.5555555555555556
Zekrom	680	51	0.19607843137254902
Xerneas	680	64	0.578125

On peut voir que le pokémon avec le plus gros total de stat (hp + def + att + att\_spé, etc...) n'est pas celui qui gagne le plus.



Ce graphique nous apprend en revanche que on a plus de chance de gagner si votre pokémon à 500 ou plus de total de stats.

Celui-ci nous apprend que plus la carte a de bonnes statistiques plus elle coûte cher ce qui est logique. Nous pouvons observer un gap très conséquent entre les 400-499 et 500+



Ce graphique ajoute une donnée en plus (Pokemon\_count) qui nous permet de vérifier que le gap entre top 1 et 2 du graphique précédent n'est pas biaisé (si il y a majoritairement des pokémons avec 500+ de stats par exemple)

## Conclusion

On peut conclure que ce n'est pas parce que la carte est chère que notre winrate sera excellent mais une carte qui vaut cher à plus de chance de gagner.