

# Projet2 Analysez des données de systèmes éducatifs



Découvrir un jeu de données et se familiariser avec les outils de Data Scientist

**AL SAMMAN Wassim – Data Scientist Apprenti**  
**Mentor : Panayotis PAPOUTSIS**



# Problématique et le but du projet

- La demande du Manager Mark.
- Le but d'expansion à l'international.
- Comprendre les différents fichiers de données.
- Trouver les données les plus pertinentes.
- Concentrer à réaliser le plus fort potentiel du client.
- Trouver les pays les plus importants pour l'expansion.

# Première découverte des données

- Sont des données de la banque mondiale (The World Bank EdStats All Indicator Query).
- Téléchargeables sur le site (Education Statistics).
- Plus que 4000 Indicateurs sur différents thèmes comme : Education, Population, Economie...
- Les Indicateurs sont classés dans des catégories sur le site (Available Indicators).
- Ce site est le clé pour trouver des Indicateurs dans nos fichiers. Soit par nom, soit par code.
- Dans la photo, c'est un exemple du premier fichier (EdStatsCountry) ouvert sur Jupyter. La photo montre qu'une petite partie de ce fichier.

	Country Code	Short Name	Table Name	Long Name	2-alpha code	Currency Unit	Special Notes	Region	Income Group	WB-2 code	...	IMF dissemination stand
0	ABW	Aruba	Aruba	Aruba	AW	Aruban florin	SNA data for 2000-2011 are updated from offici...	Latin America & Caribbean	High income: nonOECD	AW	...	N
1	AFG	Afghanistan	Afghanistan	Islamic State of Afghanistan	AF	Afghan afghani	Fiscal year end: March 20; reporting period fo...	South Asia	Low income	AF	...	General D Dissemina Sys (GD
2	AGO	Angola	Angola	People's Republic of Angola	AO	Angolan kwanza	April 2013 database update: Based on IMF data,...	Sub-Saharan Africa	Upper middle income	AO	...	General D Dissemina Sys (GD
								Europe &	Upper			General D

# Premier pas sur Jupyter

- Importer les librairies nécessaires pour la suite du travail : Pandas, Numpy, matplotlib...
- Lire les fichiers : `pd.read_csv()`.
- Utiliser des fonctions pour visualiser le contenu des fichiers : `.head()`, `.columns`, `.loc[]`, `.describe()`...

```
[4]: df1.head()
```

Out[4]:

	Country Code	Short Name	Table Name	Long Name	2-alpha code	Currency Unit	Special Notes
0	ABW	Aruba	Aruba	Aruba	AW	Aruban florin	Special data 2011 update from official figures
1	AFG	Afghanistan	Afghanistan	Islamic State of Afghanistan	AF	Afghan afghani	Major report

```
In [5]: df1.columns
```

```
Out[5]: Index(['Country Code', 'Short Name', 'Table Name', 'Long Name', '2-alpha code', 'Currency Unit', 'Special Notes', 'National accounts base year', 'SNA price valuation', 'Lending System of National Accounts', 'PPP survey year', 'Balance of Payments', 'External debt Reporting status'])
```

```
In [5]: df1.describe()
```

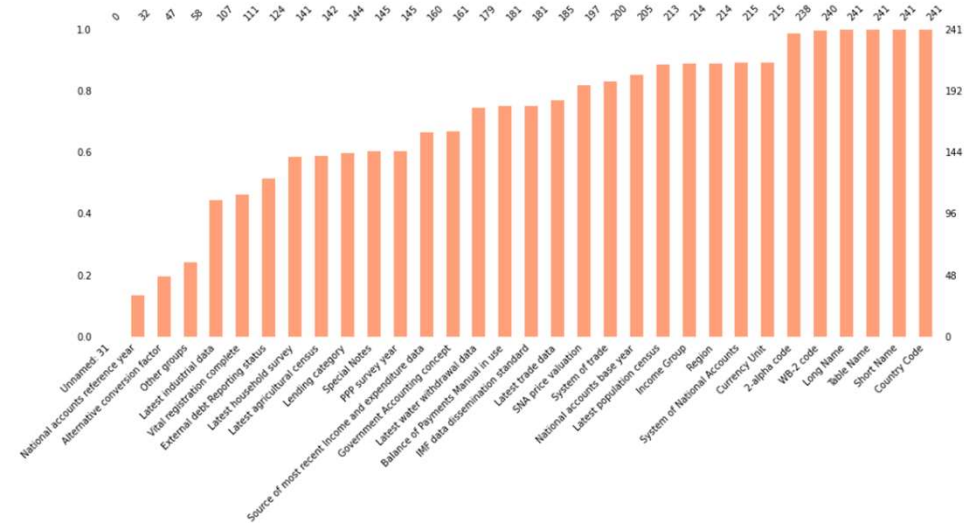
```
Out[5]:
```

	National accounts reference year	Latest industrial data	Latest trade data	Unnamed: 31
count	32.00000	107.000000	185.000000	0.0
mean	2001.53125	2008.102804	2010.994595	NaN
std	5.24856	2.616834	2.569675	NaN
min	1987.00000	2000.000000	1995.000000	NaN
25%	1996.75000	2007.500000	2011.000000	NaN
50%	2002.00000	2009.000000	2012.000000	NaN
75%	2005.00000	2010.000000	2012.000000	NaN
max	2012.00000	2010.000000	2012.000000	NaN

# Meilleure visualisation avec la librairie Missingno

- D'abord, Il faut installer Missingno:
  - Depuis Jupyter: !pip install missingno.
  - Sur Anaconda Prompt: pip install missingno.
- Visualise avec un graphique le taux de remplissage et la corrélation entre les données.
- Dispose des fonctions comme : .bar, .matrix, .heatmap et .dendrogram.
- Chacun de fonctions donne un graphique différent.
- Choix de fonction selon notre but.
- La fonction choisie est : missingno.bar.
- Un exemple de missingno.bar.

```
In [10]: missingno.bar(df1, color="lightsalmon", sort="ascending", figsize=(16,6), fontsize=10);
```



# Choisir les données les plus pertinentes à la demande du Manager

- Le point clé du choix est basé sur les Indicateurs.
- Sur le site (Available Indicators), on trouve les noms et les codes des Indicateurs.
- Le choix a été basé sur 4 thèmes : Population, Education, Déploiement internet et matériels informatiques, Economique.
- Vérifier notre choix avec les fichiers du projet.
- Au-dessous la ligne de code qui permet de vérifier l'Indicateur choisi.

```
In [26]: df3.loc[df3['Indicator Code'] == 'IT.NET.USER.P2' ]
```

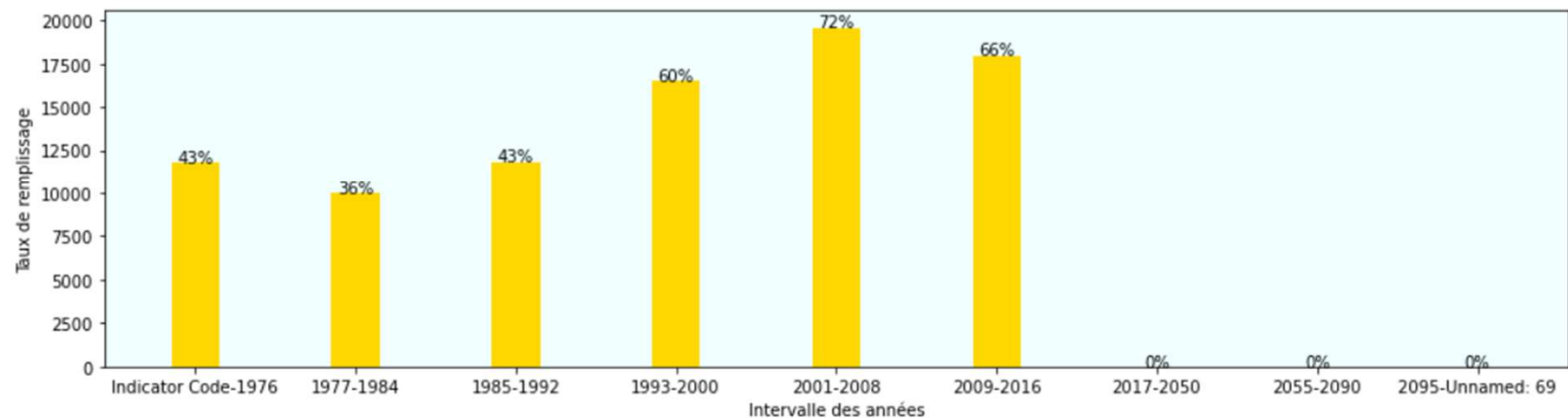
```
Out[26]:
```

	Country Name	Country Code	Indicator Name	Indicator Code	1970	1971	1972	1973	1974	1975	...
1375	Arab World	ARB	Internet users (per 100 people)	IT.NET.USER.P2	NaN	NaN	NaN	NaN	NaN	NaN	...
			Internet								

# Choisir la période d'analyse

- Deux intérêt pour la période : Une période récente, une période où le pourcentage de données est bon.
- Décider à l'aide du taux du remplissage selon les années.
- La nécessité de créer une fonction pour ne pas répéter à chaque fois le code.
- La fonction **def analyse\_periode(df,rangee,first)**.
- Appliquer la fonction sur toutes les données **dfALLInd**.

```
In [22]: analyse_periode(dfALLInd,8,3);
```



# Eliminer les pays qui sont vides

- Le but d'analyser les pays.
- Taux de remplissage n'est pas par colonne.
- La fonction **def analyse\_pays(df)**.
- Appliquer la fonction sur le DataFrame de la population dfpop.

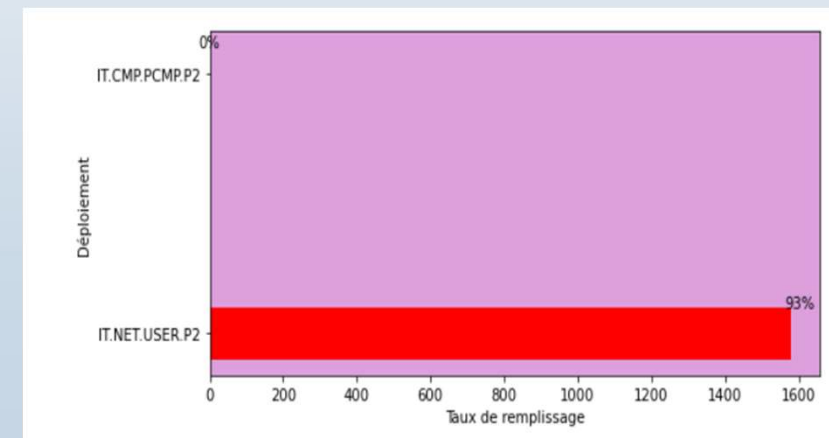
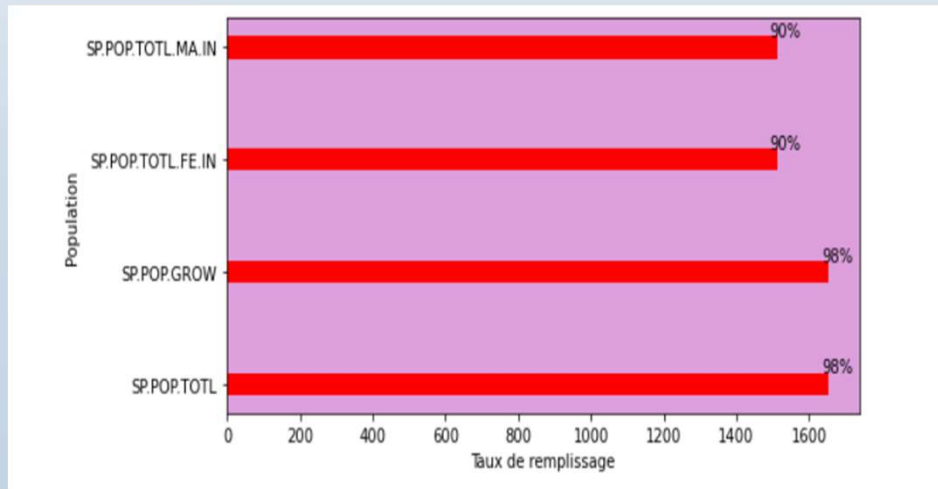
```
In [25]: analyse_pays(dfpop);
```


Country Code	Value (approximate)
NRU	0.1
VGB	0.1
MAF	0.2
ERI	0.2
SXM	0.2
MNP	1.0
AND	1.0
FRO	1.0
KNA	1.0
GIB	1.0
XXX	1.0
TCA	1.0
TUV	1.0
MHL	1.0
ASM	1.0
DMA	1.0
MCO	1.0
BMU	1.0
SMR	1.0
IMN	1.0
PLW	1.0
CYM	1.0



# Créer un DataFrame satisfait la période et les pays et choisir le meilleur indicateur

- Avoir le DataFrame dans la période d'étude et en éliminant les pays vides avec la fonction **def final(df)**.
- Choisir un seul indicateur.
- Réaliser le choix avec la fonction **indper(df, IndGr)**.
- Créer le DataFrame final **dfIndB** qui contient les 4 meilleurs indicateurs choisis.





# Avoir le DataFrame sans des données manquantes

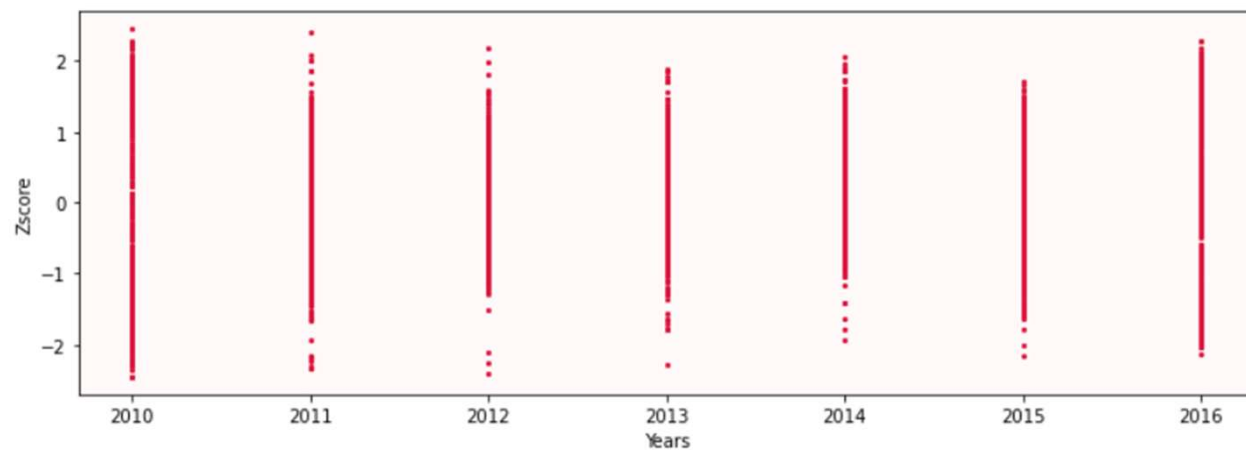
- L'intérêt de remplir les données manquantes.
- Les méthodes pour remplir:
  - Remplir par moyenne.
  - Supprimer des lignes qui ont des valeurs manquantes.
  - Utiliser des algorithmes.
- La méthode que j'ai choisi.
- La fonction **fillbymean(df,NumberNan)**.

# Utiliser Z-Score pour vérifier la qualité du remplissage

- Pourquoi cette méthode ?
- La fonction **checkbyzscore(df)**.

```
In [38]: dfIndBF=fillbyme(dfIndB,4)  
         checkbyzscore(dfIndBF)
```

```
Out[38]: (Text(0, 0.5, 'Zscore'), 6174)
```



# Les pays les plus potentiels

- Filtrage sur les noms des pays. Eliminer les pays qui ont des noms : High income, Low & middle income...
- Uniformiser les indicateurs avec la fonction **normalize(df,decimal)**.
- L'idée de Scoring.
- Appliquer la fonction **def scoring\_pays(df)** sur le DataFrame final des meilleurs indicateurs, rempli, uniformisé, et sans les noms des pays éliminés **dfIndBFRN**.
- Prendre en compte la langue du pays.

```
In [46]: scoring_pays(dfIndBFRN)
```

Out[46]:

	Country Code	Country Name	Score
0	PRY	Paraguay	0.829
1	ALB	Albania	0.813
2	ARM	Armenia	0.810
3	BGR	Bulgaria	0.806
4	ECU	Ecuador	0.806
...	...	...	...
227	TCA	Turks and Caicos Islands	0.295
228	ERI	Eritrea	0.292
229	MAF	St. Martin (French part)	0.251
230	CUW	Curacao	0.236
231	CHI	Channel Islands	0.174

232 rows × 3 columns



# Conclusion

- Ce que j'ai appris dans ce projet.
- Trouver une approche à suivre.
- S'habituer à une nouvelle méthode du travail.
- La nécessité de développer ses outils informatiques.
- Optimiste pour la suite.
- Le but de respecter les délais.