Projet 1 Academy

Nacim BOUGHANMI

Je suis Data Scientist dans **une start-up de la EdTech**, nommée **Academy**, qui propose des contenus de formation en ligne pour un public de niveau lycée et université.
Mark, mon manager, m'a convié à une réunion pour vous présenter le projet d'expansion à l'international de l'entreprise.

Voici les différentes questions que Mark aimerait explorer :

- Quels sont les pays avec un fort potentiel de clients pour nos services ?

- Pour chacun de ces pays, quelle sera l'évolution de ce potentiel de clients ?

- Dans quels pays l'entreprise doit-elle opérer en priorité ?

Quels sont les pays avec un fort potentiel de clients pour nos services ?

I. Analyse pré-exploratoire de ce jeu de données

Entrée [2]:

```
#Import des fichiers CSV du projet

df_statsCountry = pandas.read_csv("EdStatsCountry.csv")
df_statsCountrySeries = pandas.read_csv("EdStatsCountry-Series.csv")
df_statsData = pandas.read_csv("EdStatsData.csv")
df_statsFootNote = pandas.read_csv("EdStatsFootNote.csv")
df_statsSeries = pandas.read_csv("EdStatsSeries.csv")
```

J'importe l'intégralité de mon dataset afin de réaliser un pré-analyse pour sélectionner la Data Frame la plus pertinente.

II. Sélection du Data Frame df_StatsData

```
Entrée [7]: #Affichage de tous les Indicteur Name de la df_statsData

            df_statsData['Indicator Name']
```

```
Out[7]: 0         Adjusted net enrolment rate, lower secondary, ...
        1         Adjusted net enrolment rate, lower secondary, ...
        2         Adjusted net enrolment rate, lower secondary, ...
        3         Adjusted net enrolment rate, lower secondary, ...
        4         Adjusted net enrolment rate, primary, both sex...
                                      ...
        886925    Youth illiterate population, 15-24 years, male...
        886926    Youth literacy rate, population 15-24 years, b...
        886927    Youth literacy rate, population 15-24 years, f...
        886928    Youth literacy rate, population 15-24 years, g...
        886929    Youth literacy rate, population 15-24 years, m...
        Name: Indicator Name, Length: 886930, dtype: object
```

Après une prés-analyse j'ai choisi la df_StatsData, car elle contient des indicateurs pertinent pour la suite de l'analyse.

II. Sélection du Data Frame df_StatsData

Entrée [9]:
```
#Selection indicateurs pertinent.
#La df_schooling correspond aux inscriptions dans l'enseignement supérieur, tous programmes confondus et tous sexes confondus par

df_schooling= df_statsData.loc[df_statsData['Indicator Name']=="Gross enrolment ratio, tertiary, both sexes (%)"]
df_schooling
```

Out[9]:

	Country Name	Country Code	Indicator Name	Indicator Code	1970	1971	1972	1973	1974	1975	...	2060	2065	2070	2075	2080	2085
1339	Arab World	ARB	Gross enrolment ratio, tertiary, both sexes (%)	SE.TER.ENRR	5.89729	5.890620	6.169500	6.531240	6.933240	7.47784	...	NaN	NaN	NaN	NaN	NaN	NaN
5004	East Asia & Pacific	EAS	Gross enrolment ratio, tertiary, both sexes (%)	SE.TER.ENRR	3.15454	3.047160	3.155800	3.243450	3.513950	3.81330	...	NaN	NaN	NaN	NaN	NaN	NaN

Entrée [10]:
```
df_schooling['Country Name'].value_counts()
```

Out[10]:
```
Qatar                          1
Mexico                         1
Comoros                        1
Congo, Rep.                    1
Bermuda                        1
                              ..
Gabon                          1
Hong Kong SAR, China           1
Estonia                        1
St. Martin (French part)       1
Malaysia                       1
Name: Country Name, Length: 242, dtype: int64
```

Entrée [11]:
```
#Calcul du nombre de Na pour l'année 2014

df_schooling['2014'].isnull().value_counts()
```

Out[11]:
```
False    144
True      98
Name: 2014, dtype: int64
```

Entrée [13]:
```
#Calcul du nombre de Na pour l'année 2013

df_schooling['2013'].isnull().value_counts()
```

Out[13]:
```
False    150
True      92
Name: 2013, dtype: int64
```

Je vérifie que la quantité de données pour mon indicateur soit suffisante pour que mon analyse soit bien représentative de la réalité.

III. Finalisation de la df_schooling

```
Entrée [13]: #Création d'une liste.

colonnes = ['Country Name','2014']
df_schooling[colonnes]
```

J'ai créé une liste qui permet d'afficher **seulement les indicateurs pertinents** et supprimer tous les pays sans données.

Ainsi nous pouvons voir la **df_schooling** contenant la totalité des valeurs pour **chaque pays pour l'année 2014.**

```
Entrée [14]: #Supression de tout les pays sans données.

df_schooling=df_schooling[colonnes].dropna()
df_schooling
```

Out[14]:

	Country Name	2014
1339	Arab World	28.174959
5004	East Asia & Pacific	39.147720
8669	East Asia & Pacific (excluding high income)	36.470322
12334	Euro area	71.001488
15999	Europe & Central Asia	65.080727
...	...	...
847954	United States	86.663963
851619	Uruguay	56.344040
855284	Uzbekistan	8.574050
866279	Vietnam	30.477739
873609	West Bank and Gaza	44.006870

II. Sélection du Data Frame df_StatsData

Entrée [16]:
```
#df_schooling correspond mainteneant au pays présélectionner en fonction de leur taux d'inscription dans l'enseignement supérieu

df_schooling = df_statsData.loc[df_statsData['Country Name'].isin(['Greece','Korea, Rep.','Australia','Spain','Belarus']),:]
df_schooling[colonnes]
```

Out[16]:

	Country Name	2014
128275	Australia	NaN
128276	Australia	NaN
128277	Australia	NaN
128278	Australia	NaN
128279	Australia	97.138184
...	...	...
747655	Spain	5390.000000
747656	Spain	99.722760
747657	Spain	99.686810
747658	Spain	0.999290
747659	Spain	99.757310

Cette fonction m'a permis de faire apparaitre dans la df_schooling, voici les **5 pays avec le plus grand taux d'inscription dans l'enseignement supérieur** :

1.Australie
2. Espagne
3. Corée du sud
4. Grèce
5. Biélorussie

Pour chacun de ces pays, quelle sera l’évolution de ce potentiel de clients ?

I. Création de la Data Frame PIB

Entrée [17]:
```
#df_PIB correspond aux PIB par habitant en $ US pour chaque pays.

df_PIB=df_statsData.loc[df_statsData['Indicator Name']=="GDP per capita (current US$)"]
```

Entrée [18]:
```
#Application de la liste sur la df_PIB
#Classement dans l'ordre décroissant.

df_PIB=df_PIB[colonnes].dropna()
df_PIB.sort_values(['2014'], ascending=False)
```

Out[18]:

	Country Name	2014
507015	Liechtenstein	179308.075616
514345	Luxembourg	119225.380023
627960	Norway	97199.919096
518010	Macao SAR, China	94004.389829
433715	Isle of Man	89941.644285

Cette Data Frame rassemble l'ensemble des pays avec
comme seul indicateur leur **PIB en $**

Pour faciliter la visualisation des valeur et me faciliter par la suite les calcules entre les différents indicateurs, je transforme l'indicateur **GPD(PIB) en pourcentage.**

Entrée [19]:
```
#df_MaxPIB correspond au PIB maximum de l'année 2014, cette DataFrame me sera utilise par la suite pour passer mon indicteur GPD

df_MaxPIB=df_PIB['2014'].max()
df_MaxPIB
```

Out[19]: 179308.075615568

Entrée [20]:
```
#Passage de l'indicateur du GDP en pourcentage

df_PIB=df_PIB.assign(Percentage=lambda x :(x['2014']/df_MaxPIB*100))
df_PIB.sort_values(['2014'], ascending=False)
```

Out[20]:

	Country Name	2014	Percentage
507015	Liechtenstein	179308.075616	100.000000
514345	Luxembourg	119225.380023	66.491919
627960	Norway	97199.919096	54.208333
518010	Macao SAR, China	94004.389829	52.426189
433715	Isle of Man	89941.644285	50.160398

I. Création de la Data Frame SPIB

Entrée [21]:
```
#df_SPIB correspond au classement des pays sélectionné en fonction de leur PIB.

df_SPIB=df_PIB.loc[df_PIB['Country Name'].isin(['Greece','Korea, Rep.','Australia','Spain','Belarus']) & (df_statsData['Indicator
df_SPIB[['Country Name','2014','Percentage']].sort_values(['Percentage'], ascending=False)
```

Out[21]:

	Country Name	2014	Percentage
129520	Australia	62214.609121	34.697048
745240	Spain	29623.164445	16.520820
470365	Korea, Rep.	27811.366384	15.510381
367745	Greece	21760.979799	12.136085
155175	Belarus	8318.512690	4.639229

J'ai créé une nouvelle Data Frame **SPIB** qui regroupe seulement les données pour **les 5 pays à fort potentiel.**

II. Création de la Data Frame Utilisateur d'internet

Entrée [22]:
```
#df_Net correspond au pourcentage d'utilisateur d'internet pour 100 habitants pour chaque pays.

df_Net=df_statsData.loc[df_statsData['Indicator Name']=="Internet users (per 100 people)"]
```

Entrée [23]:
```
#Application de la liste sur la df_Net
#Classement dans l'ordre décroissant.

df_Net=df_Net[colonnes].dropna()
df_Net.sort_values(by=['2014'], ascending=False)
```

Out[23]:

	Country Name	2014
411855	Iceland	98.16
169965	Bermuda	96.80
628090	Norway	96.30
287245	Denmark	95.99
107660	Andorra	95.90

Cette Data Frame rassemble l'ensemble des pays avec comme seul indicateur leur **taux d'utilisateur d'internet**

II. Création de la Data Frame SUtilisateur d’internet

Entrée [24]:
```
#df_SNet correspond au classement des pays sélectionné en fonction du pourcentage d'utilisateur.

df_SNet=df_statsData.loc[df_statsData['Country Name'].isin(['Greece','Korea, Rep.','Australia','Spain','Belarus']) & (df_statsDat
df_SNet[colonnes].sort_values(['2014'], ascending=False)
```

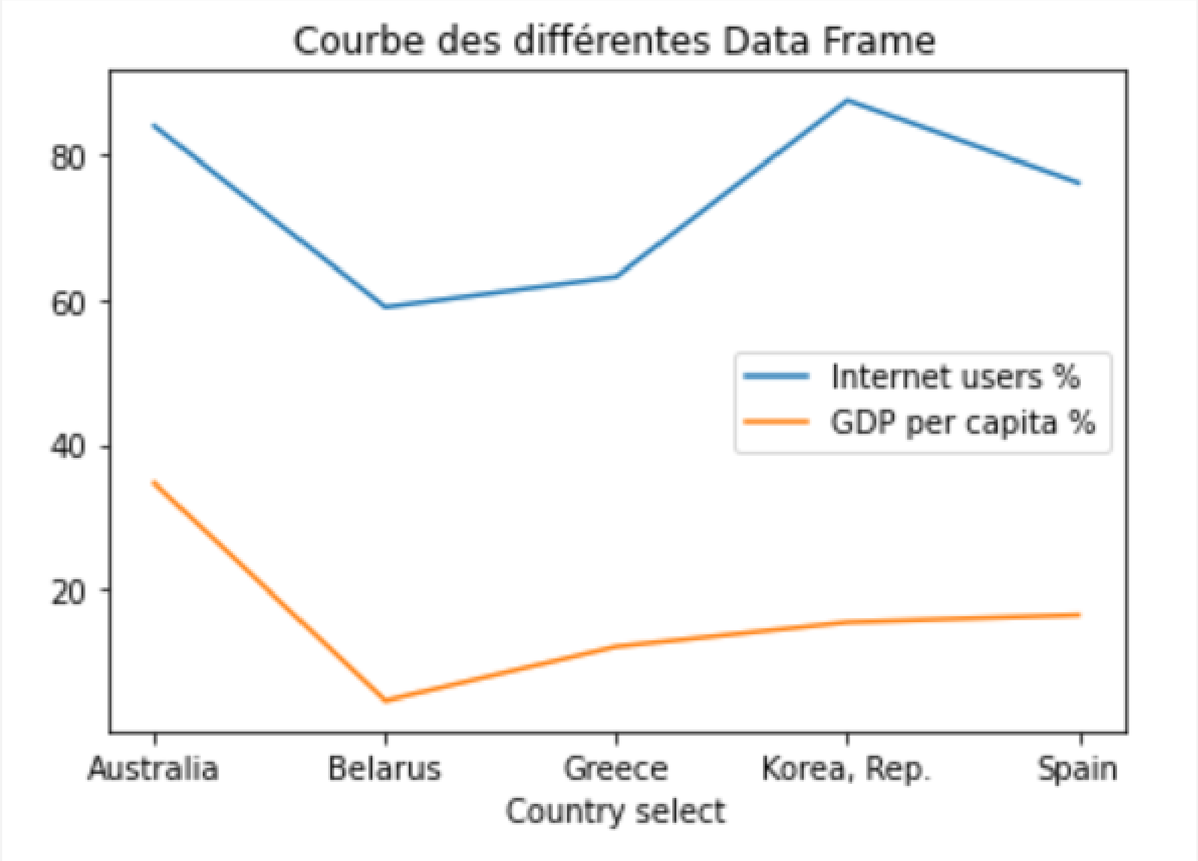
Out[24]:

	Country Name	2014
470495	Korea, Rep.	87.556826
129650	Australia	84.000000
745370	Spain	76.190000
367875	Greece	63.210000
155305	Belarus	59.020000

J'ai créé une nouvelle Data Frame Utilisateur d’internet qui regroupe seulement les données pour **les 5 pays à fort potentiel.**

III. Analyse et évolution potentielle

Voici deux courbes représentant **la Data Frame Utilisateur d'internet en bleu** et **la Data Frame PIB en orange.**



Dans quels pays l'entreprise doit-elle opérer en priorité ?

I. Création d'un Score

Entrée [26]:
```
#Jointure entre la df_paysPIB et la df_paysNet.

df_final = df_SPIB.merge(df_SNet, left_on='Country Name', right_on='Country Name')

#Création de la df_final avec le score pour chaque pays.

df_final['Score'] = (df_final['2014_y'] + df_final['Percentage'])/2
df_final[['Country Name','2014_y','Percentage','Score']].sort_values(['Score'], ascending=False)
```

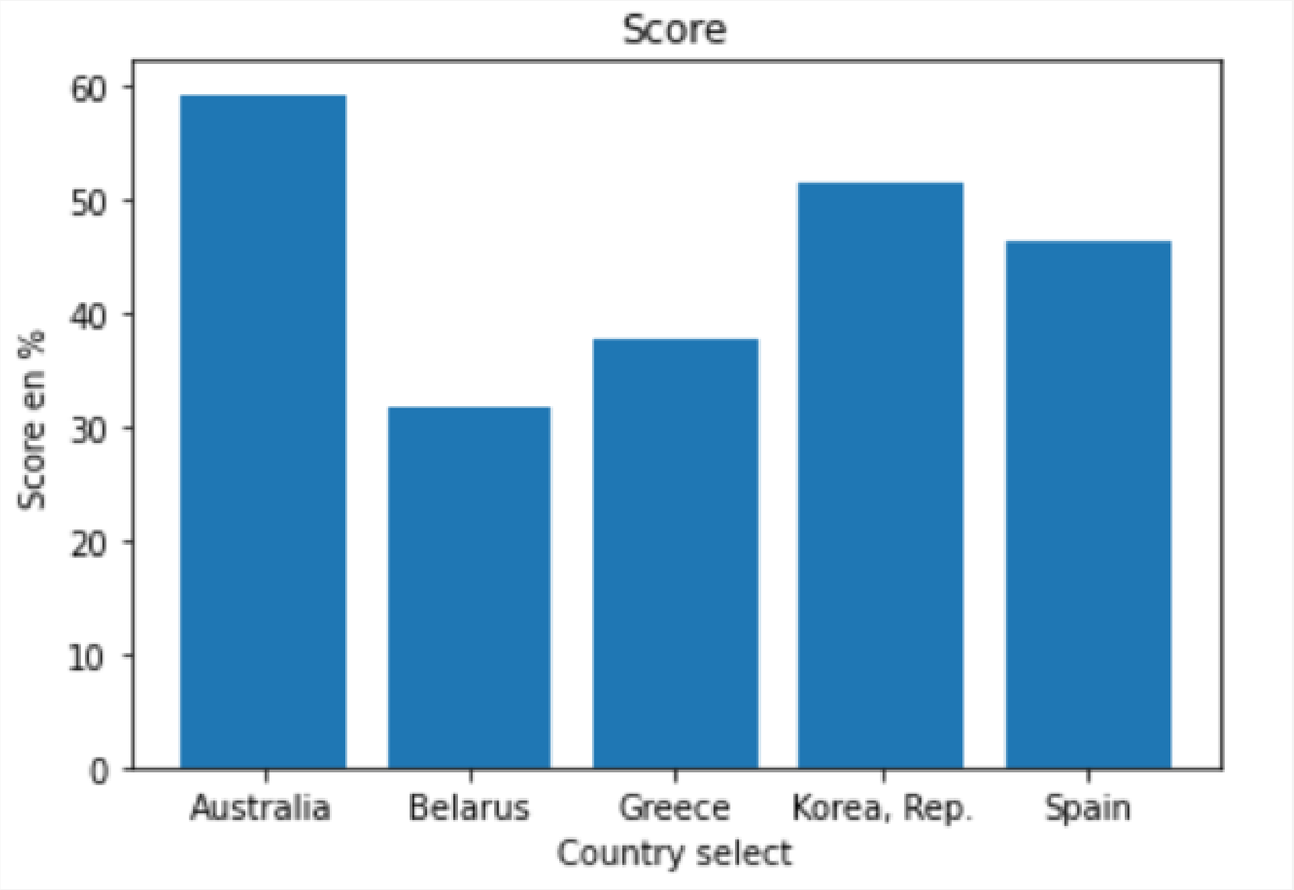
Out[26]:

	Country Name	2014_y	Percentage	Score
0	Australia	84.000000	34.697048	59.348524
3	Korea, Rep.	87.556826	15.510381	51.533604
4	Spain	76.190000	16.520820	46.355410
2	Greece	63.210000	12.136085	37.673042
1	Belarus	59.020000	4.639229	31.829615

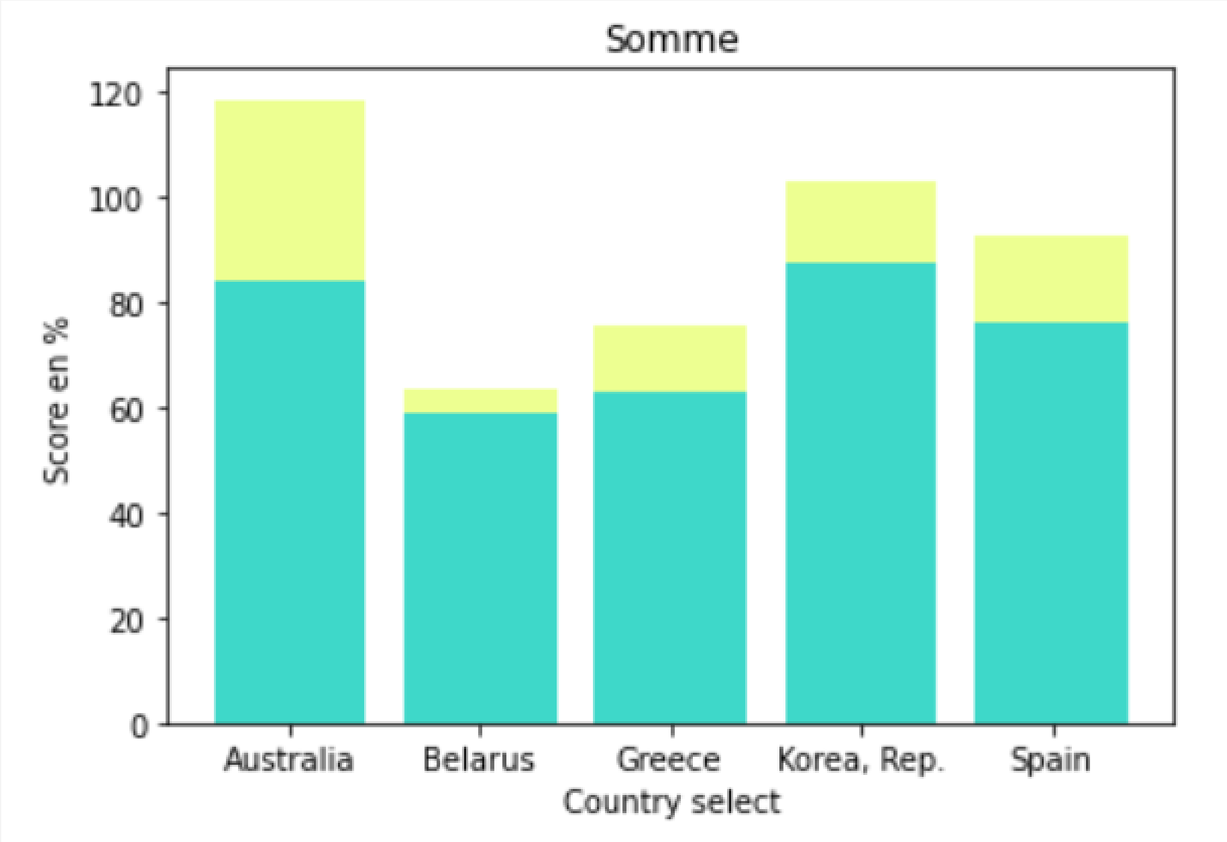
Je vais maintenant créer **un score** qui va nous permettre de voir plus clairement le pays avec **le maximum de potentiel.**

II. Histogramme Score

Cet histogramme qui s'appuie sur le score de chaque pays, il nous permet de voir très clairement qu'un pays se démarque, **l'Australie** avec un score de **59%** suivi de **la Corée du sud 51%** et **l'Espagne 46%.**



III. Histogramme Somme



Cet histogramme quant à lui s'appuie sur la somme des valeurs de **la Data Frame final.**