Parcours Data Scientist : Projet 2

Analysez des données de systèmes éducatifs

Samuel Monet
23 Octobre 2010

Sommaire de la présentation

- Description des données
- Doublons et Données manquantes
- Sélection des informations pertinentes
- Présélection de pays
- Nettoyage des données pour affinage des résultats
- Sélection d’indicateurs
- Analyse de la présélection de pays
- Proposition finale

Description des données

On a 5 tableaux,

```
data=pd.read_csv("EdStatsData.csv")
countries=pd.read_csv("EdStatsCountry.csv")
indicators=pd.read_csv("EdStatsSeries.csv")
infoindicators=pd.read_csv("EdStatsFootNote.csv")
countryseries=pd.read_csv("EdStatsCountry-Series.csv")
```

une fois nettoyés des colonnes vides,

```
data.dropna(axis=1,how='all',inplace=True)
countries.dropna(axis=1,how='all',inplace=True)
indicators.dropna(axis=1,how='all',inplace=True)
infoindicators.dropna(axis=1,how='all',inplace=True)
countryseries.dropna(axis=1,how='all',inplace=True)
```

on obtient:

```
print(data.shape)
print(countries.shape)
print(indicators.shape)
print(infoindicators.shape)
print(countryseries.shape)
```

```
(886930, 69)
(241, 31)
(3665, 15)
(643638, 4)
(613, 3)
```

Doublons et données manquantes

- Le tableau « data »

	Country Name	Country Code	Indicator Name	Indicator Code	1970	1971	1972	1973	1974	1975	...	2055	2060	2065	2070	2075	2080	2085	2090	2095	2100
0	Arab World	ARB	Adjusted net enrolment rate, lower secondary, ...	UIS.NERA.2	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	Arab World	ARB	Adjusted net enrolment rate, lower secondary, ...	UIS.NERA.2.F	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

```
data[data.duplicated(['Country Code','Indicator Code'])]
```

Renvoi un tableau vide

```
a=list(data.columns)[4:]
data.dropna(subset=a,how='all',inplace=True)
data.shape
```

```
(357405, 69)
```

On a donc un tableau sans doublons mais avec plus de 500 000 lignes vides

Doublons et données manquantes

- Le tableau « indicators »
  - On ne trouve toujours pas de donnée dupliquées
  - Pas non plus de valeurs manquantes :

```
indicators.dropna(subset=['Series Code'],how='any',inplace=True)
indicators.shape
```

```
(3665, 15)
```

- Le tableau « countryseries »
  - Pas de valeurs dupliquées
  - Pas de valeurs manquantes :

```
countryseries.isnull().values.any()
```

```
False
```

Doublons et données manquantes

- Infoindicators
  - Pas de doublons
  - Pas de valeurs manquantes
  - On transforme les années en int :

	CountryCode	SeriesCode	Year	DESCRIPTION
0	ABW	SE.PRE.ENRL.FE	YR2001	Country estimation.
1	ABW	SE.TER.TCHR.FE	YR2005	Country estimation.

```
def changedate(date):
    return int(date[2:])

infoindicators['Year']=infoindicators['Year'].apply(changedate)
```

Doublons et données manquantes

- Countries
  - Pas de valeurs manquantes
  - Pas de valeurs dupliquées

```
countries[4:6]
```

	Country Code	Short Name	Table Name	Long Name	2-alpha code	Currency Unit	Special Notes	Region	Income Group	WB-2 code	...	Government Accounting concept	IMF data dissemination standard	Latest population census	Latest household survey	Inc exp
4	AND	Andorra	Andorra	Principality of Andorra	AD	Euro	NaN	Europe & Central Asia	High income: nonOECD	AD	...	NaN	NaN	2011. Population figures compiled from adminis...	NaN	
5	ARB	Arab World	Arab World	Arab World	1A	NaN	Arab World aggregate. Arab World is composed o...	NaN	NaN	1A	...	NaN	NaN	NaN	NaN	

2 rows × 31 columns

```
# On va créer un nouveau tableau avec juste les régions
regions=countries[countries.Region.isnull()]
countries.dropna(subset=['Region'],how='all',inplace=True)
print(regions.shape,countries.shape)
```

```
(27, 13) (214, 13)
```

On voit que les régions n'ont rien en colonne région, on en profite pour séparer en deux tables : « regions » et « countries »

Sélection des informations pertinentes

- Data (Tout semble pertinent)

	Country Name	Country Code	Indicator Name	Indicator Code	1970	1971	1972	1973	1974	1975	...	2055	2060	2065	2070	2075	2080	2085	2090	2095	2100
0	Arab World	ARB	Adjusted net enrolment rate, lower secondary, ...	UIS.NERA.2	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	Arab World	ARB	Adjusted net enrolment rate, lower secondary, ...	UIS.NERA.2.F	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

- Countryseries (A première vue rien de très intéressant)

	CountryCode	SeriesCode	DESCRIPTION
0	ABW	SP.POP.TOTL	Data sources : United Nations World Population...
1	ABW	SP.POP.GROW	Data sources: United Nations World Population ...
2	AFG	SP.POP.GROW	Data sources: United Nations World Population ...

Sélection des informations pertinentes

- Infoindicators (rien de très intéressant à première vue)

  Éventuellement les années mais

  ça doit coller avec le tableau data

	CountryCode	SeriesCode	Year	DESCRIPTION
0	ABW	SE.PRE.ENRL.FE	2001	Country estimation.
1	ABW	SE.TER.TCHR.FE	2005	Country estimation.
2	ABW	SE.PRE.TCHR.FE	2000	Country estimation.

- Countries

  Beaucoup d’infos mais pas

  grand-chose d’intéressant :

  - Les noms
  - La région
  - Éventuellement Income Group

```
countries.columns
```

```
Index(['Country Code', 'Short Name', 'Table Name', 'Long Name', '2-alpha code',
       'Currency Unit', 'Special Notes', 'Region', 'Income Group', 'WB-2 code',
       'National accounts base year', 'National accounts reference year',
       'SNA price valuation', 'Lending category', 'Other groups',
       'System of National Accounts', 'Alternative conversion factor',
       'PPP survey year', 'Balance of Payments Manual in use',
       'External debt Reporting status', 'System of trade',
       'Government Accounting concept', 'IMF data dissemination standard',
       'Latest population census', 'Latest household survey',
       'Source of most recent Income and expenditure data',
       'Vital registration complete', 'Latest agricultural census',
       'Latest industrial data', 'Latest trade data',
       'Latest water withdrawal data'],
      dtype='object')
```

Sélection des informations pertinentes

- Indicators

	Series Code	Topic	Indicator Name	Short definition	Long definition	Periodicity	Base Period	Other notes	Aggregation method	and exceptions	General comments	Source	conc metho
0	BAR.NOED.1519.FE.ZS	Attainment	Barro-Lee: Percentage of female population age...	Percentage of female population age 15-19 with...	Percentage of female population age 15-19 with...	NaN	NaN	NaN	NaN	NaN	NaN	Robert J. Barro and Jong-Wha Lee: http://www.b...	

Seules les 3 premières colonnes sont intéressantes et on va commencer par la colonne « Topic » pour essayer de mieux cibler notre recherche

Présélection de pays

```
indicators.Topic.unique()
```

```
array(['Attainment', 'Education Equality',
       'Infrastructure: Communications', 'Learning Outcomes',
       'Economic Policy & Debt: National accounts: US$ at current prices: Aggregate indicators',
       'Economic Policy & Debt: National accounts: US$ at constant 2010 prices: Aggregate indicators',
       'Economic Policy & Debt: Purchasing power parity',
       'Economic Policy & Debt: National accounts: Atlas GNI & GNI per capita',
       'Teachers', 'Education Management Information Systems (SABER)',
       'Early Child Development (SABER)',
       'Engaging the Private Sector (SABER)',
       'School Health and School Feeding (SABER)',
       'School Autonomy and Accountability (SABER)',
       'School Finance (SABER)', 'Student Assessment (SABER)',
       'Teachers (SABER)', 'Tertiary Education (SABER)',
       'Workforce Development (SABER)', 'Literacy', 'Background',
       'Primary', 'Secondary', 'Tertiary', 'Early Childhood Education',
       'Pre-Primary', 'Expenditures', 'Health: Risk factors',
       'Health: Mortality',
       'Social Protection & Labor: Labor force structure', 'Laber',
       'Social Protection & Labor: Unemployment',
       'Health: Population: Structure', 'Population',
       'Health: Population: Dynamics', 'EMIS',
       'Post-Secondary/Non-Tertiary'], dtype=object)
```

On va commencer par « Infrastructure : Communications »

Présélection de pays

```
indicators[indicators.Topic=='Infrastructure: Communications']
```

	Series Code	Topic	Indicator Name	Short definition	Long definition	Periodicity	Base Period	Other notes	Aggregation method	Limitations and exceptions	General comments	Sou
610	IT.CMP.PCMP.P2	Infrastructure: Communications	Personal computers (per 100 people)	NaN	Personal computers are self-contained computer...	Annual	NaN	NaN	Weighted average	NaN	Restricted use: Please cite the International ...	Internati Telecommunica Union, Worlc
611	IT.NET.USER.P2	Infrastructure: Communications	Internet users (per 100 people)	NaN	Internet users are individuals who have used t...	Annual	NaN	NaN	Weighted average	Operators have traditionally been the main sou...	Please cite the International Telecommunicatio...	Internati Telecommunica Union, Worlc

On a donc deux indicateurs fondamentaux pour la société : les pourcentages de personnes ayant accès à internet, les pourcentages de personne ayant un ordinateur.
En mettant cela en parallèle avec les 15-24 on devrait pouvoir commencer à pré-sélectionner un échantillon de pays plus restreint.

Présélection de pays

On cherche donc la population de 15-24. On a vu, en regardant un peu les indicateurs, qu'il y avait des indicateurs justement pour les 15-24

```
a=[]
for i in indicators['Indicator Name']:
    if '15-24' in i.lower():
        a.append(i)
print(len(a),a)
```

```
11 ['Youth literacy rate, population 15-24 years, female (%)', 'Youth literacy rate, population 15-24 years, gende
r parity index (GPI)', 'Youth literacy rate, population 15-24 years, male (%)', 'Youth literacy rate, population 1
5-24 years, both sexes (%)', 'Population, ages 15-24, female', 'Population, ages 15-24, male', 'Population, ages 1
5-24, total', 'Youth illiterate population, 15-24 years, both sexes (number)', 'Youth illiterate population, 15-24
years, female (number)', 'Youth illiterate population, 15-24 years, male (number)', 'Youth illiterate population,
15-24 years, % female']
```

```
indicators[indicators['Indicator Name']=='Population, ages 15-24, total']
```

	Series Code	Topic	Indicator Name	Short definition	Long definition	Periodicity	Base Period	Other notes	Aggregation method	Limitations and exceptions	General comments	Source	Statisti concept a methodolo
2506	SP.POP.1524.TO.UN	Population	Population, ages 15-24, total	Population, ages 15-24, total is the total pop...	Population, ages 15-24, total is the total pop...	NaN	NaN	NaN	NaN	NaN	NaN	UNESCO Institute for Statistics (Derived)	N

Présélection de pays

```
data[data['Indicator Code']=='SP.POP.1524.TO.UN'].shape
```

```
(192, 69)
```

Il manque quand même beaucoup de données, on va chercher les population totales pour extrapoler.

```
indicpop=[]
for a in indicators['Series Code']:
    if 'SP.POP' in a and 'FE' not in a and 'MA' not in a:
        indicpop.append(a)
print(len(indicpop),indicpop)
```

```
66 ['SP.POP.0014.TO', 'SP.POP.0014.TO.ZS', 'SP.POP.0305.TO.UN', 'SP.POP.0406.TO.UN', 'SP.POP.0509.TO.UN', 'SP.POP.
0510.TO.UN', 'SP.POP.0511.TO.UN', 'SP.POP.0609.TO.UN', 'SP.POP.0610.TO.UN', 'SP.POP.0611.TO.UN', 'SP.POP.0612.TO.U
N', 'SP.POP.0709.TO.UN', 'SP.POP.0710.TO.UN', 'SP.POP.0711.TO.UN', 'SP.POP.0712.TO.UN', 'SP.POP.0713.TO.UN', 'SP.P
OP.1014.TO.UN', 'SP.POP.1015.TO.UN', 'SP.POP.1016.TO.UN', 'SP.POP.1017.TO.UN', 'SP.POP.1018.TO.UN', 'SP.POP.1115.T
O.UN', 'SP.POP.1116.TO.UN', 'SP.POP.1117.TO.UN', 'SP.POP.1118.TO.UN', 'SP.POP.1215.TO.UN', 'SP.POP.1216.TO.UN', 'S
P.POP.1217.TO.UN', 'SP.POP.1218.TO.UN', 'SP.POP.1316.TO.UN', 'SP.POP.1317.TO.UN', 'SP.POP.1318.TO.UN', 'SP.POP.131
9.TO.UN', 'SP.POP.1418.TO.UN', 'SP.POP.1419.TO.UN', 'SP.POP.1524.TO.UN', 'SP.POP.1564.TO', 'SP.POP.1564.TO.ZS', 'S
P.POP.AG00.TO.UN', 'SP.POP.AG01.TO.UN', 'SP.POP.AG02.TO.UN', 'SP.POP.AG03.TO.UN', 'SP.POP.AG04.TO.UN', 'SP.POP.AG0
5.TO.UN', 'SP.POP.AG06.TO.UN', 'SP.POP.AG07.TO.UN', 'SP.POP.AG08.TO.UN', 'SP.POP.AG09.TO.UN', 'SP.POP.AG10.TO.UN',
'SP.POP.AG11.TO.UN', 'SP.POP.AG12.TO.UN', 'SP.POP.AG13.TO.UN', 'SP.POP.AG14.TO.UN', 'SP.POP.AG15.TO.UN', 'SP.POP.A
G16.TO.UN', 'SP.POP.AG17.TO.UN', 'SP.POP.AG18.TO.UN', 'SP.POP.AG19.TO.UN', 'SP.POP.AG20.TO.UN', 'SP.POP.AG21.TO.UN
', 'SP.POP.AG22.TO.UN', 'SP.POP.AG23.TO.UN', 'SP.POP.AG24.TO.UN', 'SP.POP.AG25.TO.UN', 'SP.POP.GROW', 'SP.POP.TOTL
']
```

Le dernier : SP.POP.TOTL représente la population totale des pays

Présélection de pays

- J’ai créé 2 tables une pour internet et une pour computer (En fait une aurait suffit)

```
internet=['SP.POP.1524.TO.UN','SP.POP.TOTL','IT.NET.USER.P2']
population_internet=pd.DataFrame()

for i in internet:
    population_internet=pd.concat([population_internet,data[data['Indicator Code']==i]],ignore_index=True)
```

- Je les ai un peu simplifié/nettoyé :

```
population_internet['Max']=population_internet.max(axis=1,numeric_only=True)
echantillon_internet=population_internet[['Country Name','Country Code','Indicator Name','Max']].copy()
echantillon_internet.columns=['Country','Code','Indicator','Value']
def modifinternet(x):
    if '15-24' in x:
        return '15-24'
    elif 'Population, total' in x:
        return 'total'
    else:
        return 'internet'
echantillon_internet.Indicator=echantillon_internet.Indicator.apply(modifinternet)
echantillon_internet.sort_values(by=['Code','Indicator'],inplace=True)
```

Présélection de pays

Je transforme ce tableau en un tableau plus exploitable :

```
echantillon_internet.head()
```

	Country	Code	Indicator	Value
8	Aruba	ABW	15-24	1.445500e+04
466	Aruba	ABW	internet	9.354245e+01
226	Aruba	ABW	total	1.048220e+05
0	Afghanistan	AFG	15-24	7.252785e+06
457	Afghanistan	AFG	internet	1.059573e+01

```
internet=pd.DataFrame(columns=['Country','Internet','Population','15-24'])

for i in range(countries.shape[0]):
    code=countries.iloc[i]['Country Code']
    df=echantillon_internet[echantillon_internet.Code==code]
    if df[df.Indicator=='internet'].empty or df[df.Indicator=='15-24'].empty and df[df.Indicator=='total'].empty:
        continue
    else:
        D={'Country':df[df.Indicator=='internet'].iloc[0]['Country'],\
           'Internet':df[df.Indicator=='internet'].iloc[0]['Value']}
        if not df[df.Indicator=='total'].empty:
            D['Population']=df[df.Indicator=='total'].iloc[0]['Value']
        if not df[df.Indicator=='15-24'].empty:
            D['15-24']=df[df.Indicator=='15-24'].iloc[0]['Value']
    for cle,valeur in D.items():
        internet.loc[code,cle]=valeur
```

Présélection de pays

- On obtient finalement :

	Country	Internet	Population	15-24
ABW	Aruba	93.5425	104822	14455
AFG	Afghanistan	10.5957	3.4656e+07	7.25278e+06
AGO	Angola	13	2.88135e+07	4.25935e+06
ALB	Albania	66.3634	3.28654e+06	641446
AND	Andorra	97.9306	84462	8715

	Country	Computer	Population	15-24
ABW	Aruba	9.91768	104822	14455
AFG	Afghanistan	0.390148	3.4656e+07	7.25278e+06
AGO	Angola	0.646019	2.88135e+07	4.25935e+06
ALB	Albania	4.59354	3.28654e+06	641446
ARB	Arab World	6.67668	3.69762e+08	NaN

- Chacun avec quelques valeurs manquantes en 15-24 :

```
internet.isnull().sum()
```

```
Country        0
Internet       0
Population     0
15-24         15
dtype: int64
```

```
computer.isnull().sum()
```

```
Country       0
Computer      0
Population    0
15-24         9
dtype: int64
```

Présélection de pays

- On impute les valeurs manquantes en extrapolant à partir de la moyenne des 15-24 dans les pays dont on a les valeurs (on aurait pu être plus précis en raisonnant par zone géographique mais l'idée est surtout d'avoir un ordre de grandeur)

```
df=internet.dropna(how='all')
a=df['15-24']/df['Population']
moyenneint=a.mean()

for (i,country) in internet.iterrows():
    if pd.isnull(country['15-24']):
        internet.loc[i,'15-24']=internet.loc[i,'Population']*moyenneint
```

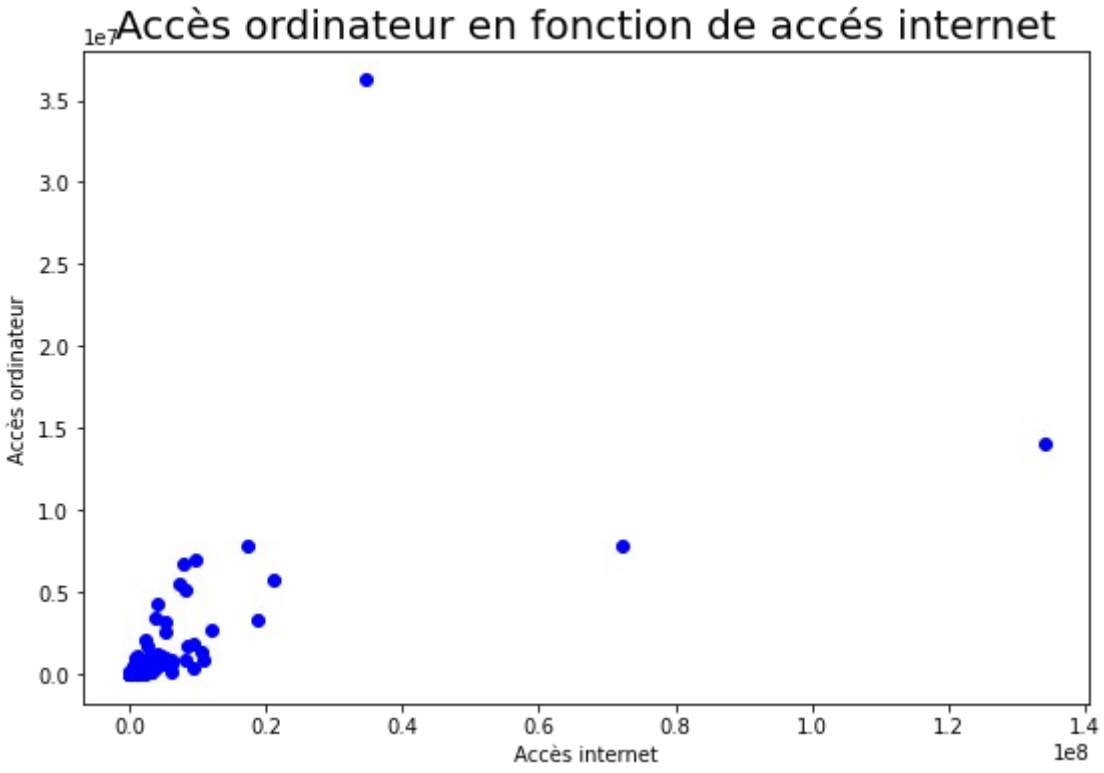
- On regarde les indicateurs :

```
internet['TargetPop']=internet['Internet']*internet['15-24']/100
print('min:',internet['TargetPop'].min(),'max:',internet['TargetPop'].max(),\
      'moyenne:',internet['TargetPop'].mean(),'sigma:',internet['TargetPop'].std(),'mediane:',internet['TargetPop']
```

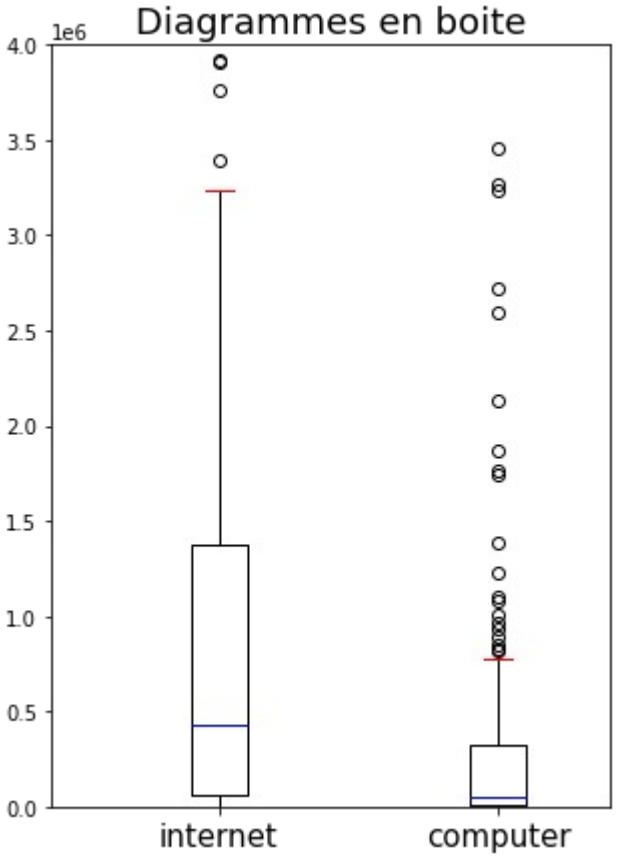
```
min: 0.0 max: 134019380.628 moyenne: 2991236.3303325246 sigma: 11498435.450027643 mediane: 531447.974743
```

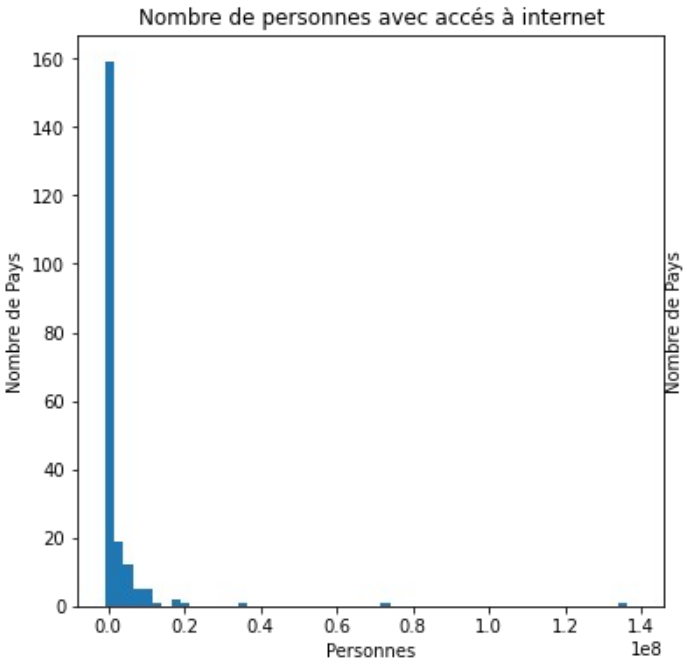
Présélection de pays

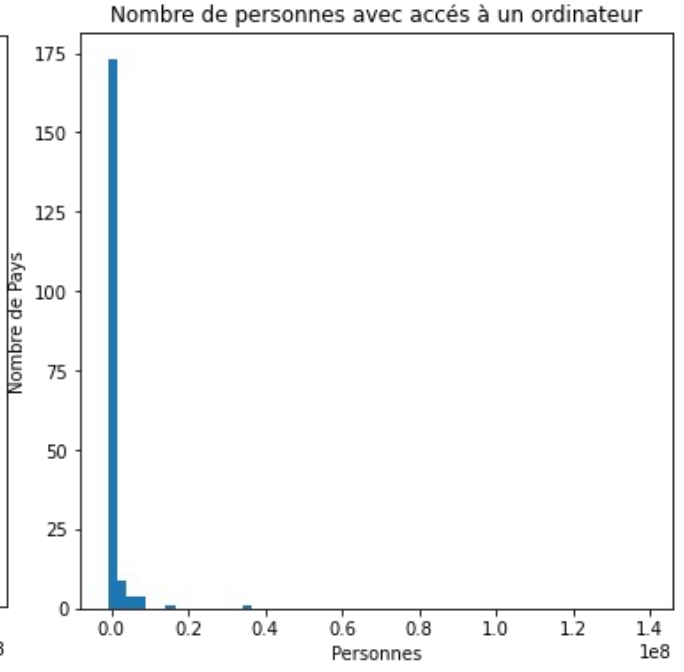
Pas de corrélation entre les deux donc on va déterminer un seuil de sélection des pays pour le nouvel échantillon



Présélection de pays







Présélection de pays

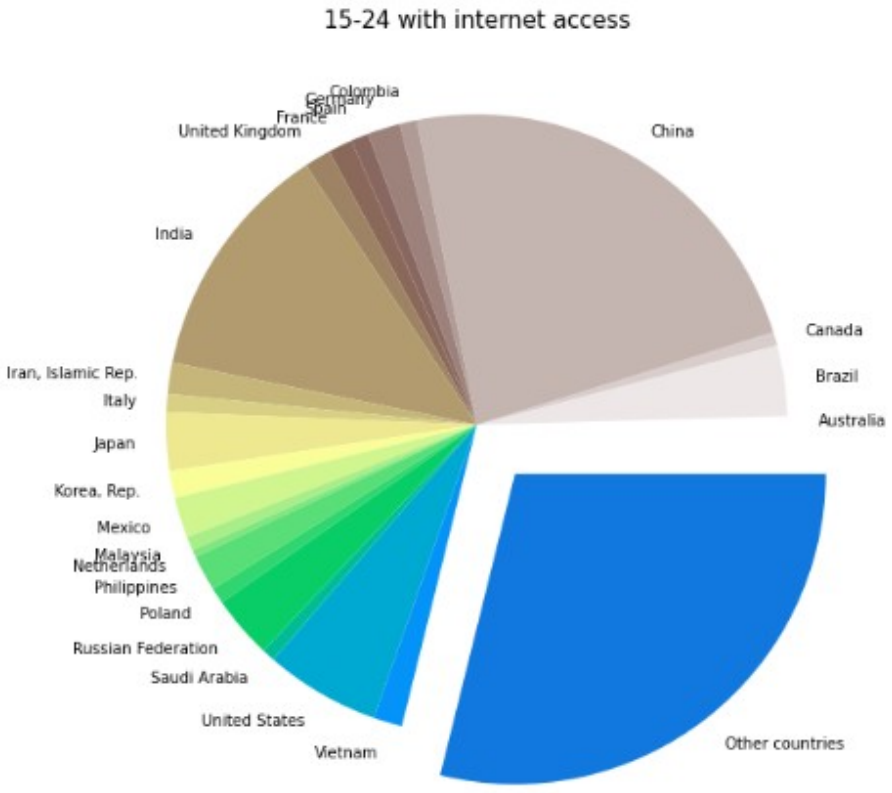
- On va sélectionner les pays :
  - avec plus de 1.000.000 de « 15-24 » avec un accès à un ordinateur
  - ET 1500000 avec un accès internet :

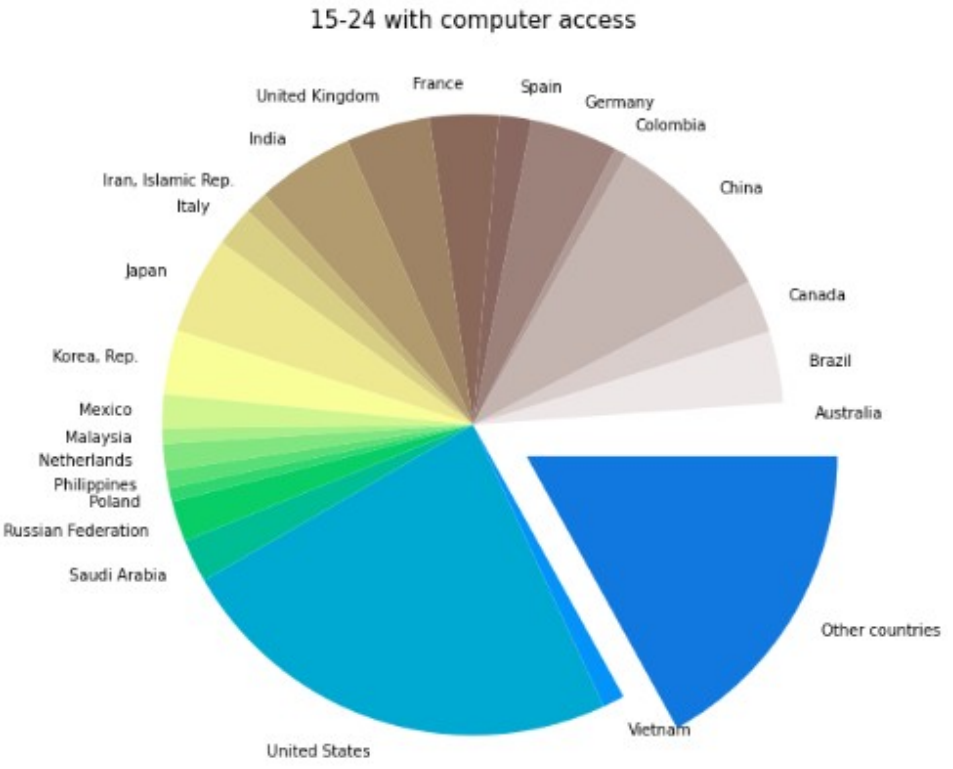
```
payscibles=pd.merge(internet[internet.TargetPop>1500000],computer[computer.TargetPop > 1000000],on=['Country'],how=
payscibles.dropna(how='any',inplace=True)
payscibles=pd.merge(payscibles,countries[['Country Code','Table Name']],left_on=['Country'],right_on=['Table Name']
payscibles=payscibles[['Country Code','Country','TargetPop_internet','TargetPop_computer']]
payscibles.head()
```

	Country Code	Country	TargetPop_internet	TargetPop_computer
0	AUS	Australia	2.58221e+06	1.75996e+06
1	BRA	Brazil	2.11566e+07	5.71313e+06
2	CAN	Canada	4.04028e+06	4.24746e+06
3	CHN	China	1.34019e+08	1.4078e+07
4	COL	Colombia	5.2319e+06	1.00907e+06

On vérifie qu'on a pas éliminer une trop grosse partie de la population mondiale cible:

Présélection de pays





Nettoyage des données pour affinage des résultats

- On a maintenant 23 pays
- On sélectionne dans le data set seulement ces pays

```
allcountries=data['Country Code'].unique()
dataech=data.copy()
for i in allcountries:
    if i not in codepays:
        dataech=dataech[dataech['Country Code']!=i]
```

- On ne sélectionne que les indicateurs disponibles pour plus de 80 % de ces pays

```
somme=dataech.groupby('Indicator Code').count()
somme.head()
presel=somme[somme['Country Name']>=19]
L=presel.index.values.tolist()
indicatorsech=indicators[indicators['Series Code'].isin(L)]
```

Il nous reste finalement 1539 indicateurs soit moins de 50 % de l'échantillon initial

Nettoyage des données pour affinage des résultats

- On ne sélectionne que les « Topics » susceptibles de nous intéresser :

  'Attainment','Learning Outcomes','Teachers','Background','Secondary', 'Tertiary', 'Expenditures'

- On nettoie un peu plus en sélectionnant les indicateurs par mots clés, on supprime tous les primary, out of school, les indicateurs genrés…

```
for i in indicators['Indicator Name'].unique().tolist():
    if 'male and female' in i.lower() or ('female' not in i.lower() and 'male' not in i.lower()
        L.append(i)
echantillon=echantillon[echantillon['Indicator Name'].isin(L)]
echantillon['Indicator Name'].unique()
```

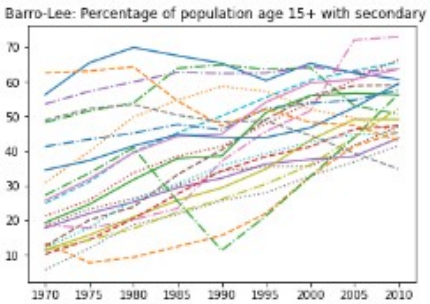
Nettoyage des données pour affinage des résultats

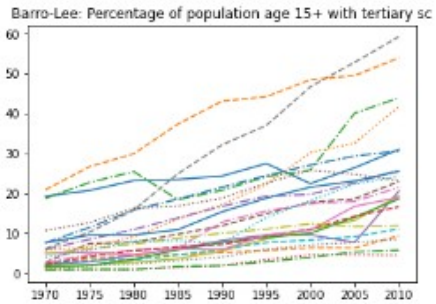
- On garde finalement les indicateurs suivants :
  - Expenditure on secondary as % of total government expenditure (%)
  - Expenditure on tertiary as % of total government expenditure (%)
  - Total inbound internationally mobile students, both sexes (number)
  - Enrolment in tertiary education per 100,000 inhabitants, both sexes
  - Pupil-teacher ratio in tertiary education (headcount basis)
  - Pupil-teacher ratio in secondary education (headcount basis)
  - Barro-Lee: Percentage of population age 15+ with secondary schooling. Total (Incomplete and Completed Secondary),
  - Barro-Lee: Percentage of population age 15+ with tertiary schooling. Total (Incomplete and Completed Tertiary)
  - Enrolment in secondary education, private institutions, both sexes (number),
  - Projection: Mean years of schooling. Age 20-39. Total,
  - Projection: Population in thousands by highest level of educational attainment. Post Secondary. Total

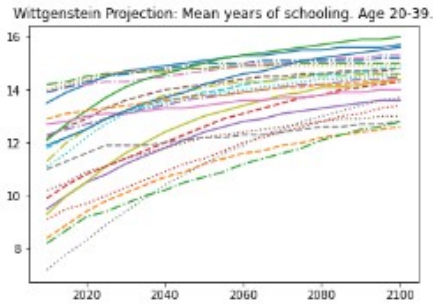
Nettoyage des données pour affinage des résultats

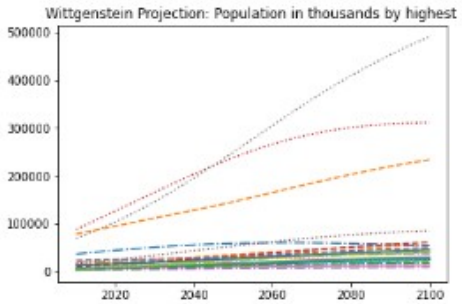
```
newdata=payscibles.copy()
fig = plt.figure(figsize=(28,14))
i=0
for code in specific_codes:
    #je fais une copie de la sélection de mon indicateur
    ech=echantillon[echantillon['Indicator Code']==code].copy()
    #je nettoie cette copie en supprimant les colonne dont je n'ai pas besoin:
    indicateur=ech.iloc[0]['Indicator Name']
    if len(indicateur)>60:
        indicateur=indicateur[:60]
    ech.dropna(axis=1,how='all',inplace=True)
    del ech['Country Code']
    del ech['Indicator Name']
    del ech['Indicator Code']
    #Je déplace la colonne country en fin de tableau pour faire un ffill
    ech['Country']=ech['Country Name']
    del ech['Country Name']
    ech=ech.ffill(axis=1)
    ech2=ech[[ech.columns.tolist()[-1],ech.columns.tolist()[-2]]]
    ech2.columns=['Country',indicateur]
    newdata=pd.merge(newdata,ech2,on='Country',how='outer')
    #je trace l'indicateur
    ech.index=ech['Country']
    del ech['Country']
    x=np.array(ech.columns,dtype=int)
    fig.add_subplot(3,4,i+1)
    style=['-','--','-.',':']
    j=0
    for (country,y) in ech.sort_index().iterrows():
        plt.plot(x, np.array(y), label=country,linestyle=style[j%4])
        j+=1
    plt.title(indicateur)
    i+=1
```

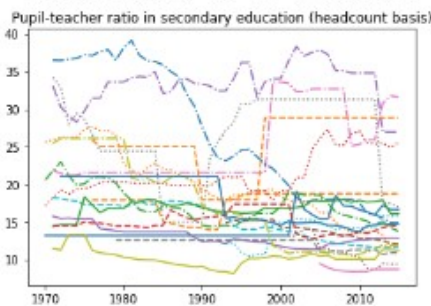
Nettoyage des données pour affinage des résultats

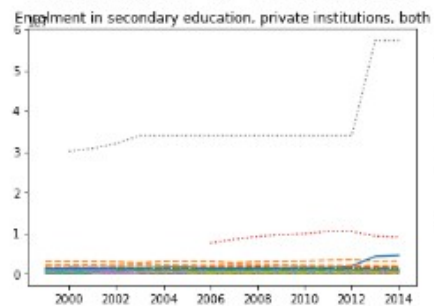


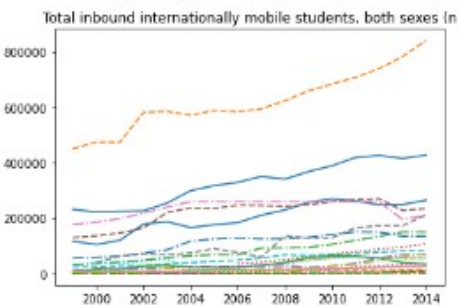


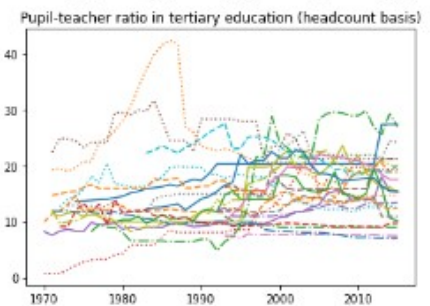


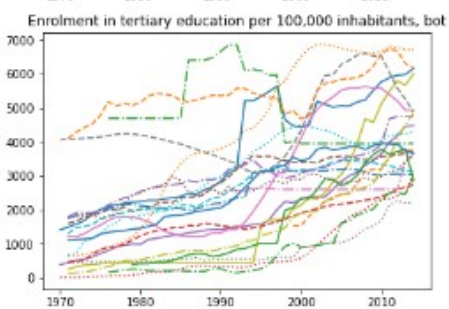


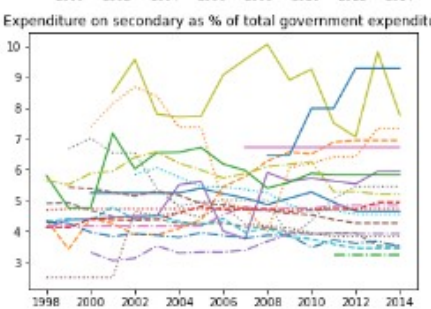


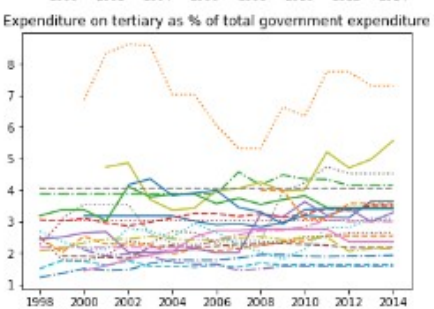












Nettoyage des données pour affinage des résultats

Je décide finalement d'éliminer les indicateurs suivants qui ne me semblent finalement pas tellement pertinents :

- Projection: Mean years of schooling. Age 20-39. Total
- Expenditure on secondary as % of total government expenditure (%)
- Expenditure on tertiary as % of total government expenditure (%)

Analyse de la présélection de pays

- On convertit les pourcentages, les pour 1000 et les pour 100 000 en nombres.
- On ajoute les étudiants avec les étudiants d'origine étrangères, on supprime les colonnes qui ne nous servent plus et on obtient le tableau ci-dessous :

```
newdata['secondarylevelpop']=newdata['Pop']*newdata['%pop_sec']/100
newdata['superieurlevelpop']=newdata['Pop']*newdata['%pop_ter']/100
newdata['secondarylevelproj']=newdata['Pop']*newdata['proj_postsec_1000']/1000
newdata['students']=newdata['Pop']*newdata['ter_enrol_100000']/100000+newdata['inbound']
newdata=newdata[['Country','Int','Comp','ratio_sec','ratio_ter','private_sec','secondarylevelpop',\
                 'superieurlevelpop','secondarylevelproj','students']]
```

	Country	Int	Comp	ratio_sec	ratio_ter	private_sec	secondarylevelpop	superieurlevelpop	secondarylevelproj	students
0	Australia	2.582206e+06	1.759958e+06	16.000000	27.472490	937368.0	1.464760e+07	7.474594e+06	5.781519e+08	1.750630e+06
1	Brazil	2.115664e+07	5.713126e+06	16.691740	18.955730	3204247.0	9.441976e+07	1.941554e+07	1.154825e+10	7.787873e+06
2	Canada	4.040284e+06	4.247457e+06	18.798639	9.081420	193637.0	1.837182e+07	1.593337e+07	1.341450e+09	1.583891e+06
3	China	1.340194e+08	1.407801e+07	13.815440	19.487909	9019335.0	9.163986e+08	6.176419e+07	4.289955e+11	4.231496e+07
4	Colombia	5.231900e+06	1.009071e+06	25.577351	15.364080	979097.0	2.126154e+07	9.988547e+06	1.451864e+09	2.177225e+06

Analyse de la présélection de pays

- On fait une fonction score pour classer les pays
  - On va attribuer à chaque pays un score pour :
    - La technologie (Internet+computer)
    - Le potentiel actuel (Population totale avec niveau secondaire et supérieur + population secondaire privée et supérieure pondérées par les nombres d'élèves par enseignant)
    - Le potentiel futur (Les projections secondaires pondérées des nombres d'élèves par enseignants)

- Afin que la démographie ne pèse pas trop on a établi des scores de 0 à 3 par catégorie en fonction du quartile du pays.

Analyse de la présélection de pays

```
score=newdata.copy()
score['techno']=score['Int']+score['Comp']
score['now']=score['secondarylevelpop']+score['superieurlevelpop']+score['students']*score['ratio_ter']+score['priv
score['future']=score['secondarylevelproj']*(score['ratio_ter']+score['ratio_sec'])
scorefinal=score[['Country','techno','now','future']].copy()

q_tech=[scorefinal.techno.quantile(0.25*i) for i in range(1,4)]
q_act=[scorefinal.now.quantile(0.25*i) for i in range(1,4)]
q_futur=[scorefinal.future.quantile(0.25*i) for i in range(1,4)]

def quartile(x,L):
    if x<L[0]:
        return 0
    elif x<L[1]:
        return 1
    elif x<L[2]:
        return 2
    else:
        return 3

scorefinal.techno=[quartile(t,q_tech) for t in scorefinal.techno]
scorefinal.now=[quartile(t,q_act) for t in scorefinal.now]
scorefinal.future=[quartile(t,q_futur) for t in scorefinal.future]

scorefinal['Total']=scorefinal.techno+scorefinal.now+scorefinal.future
scorefinal.sort_values(by='Total',ascending=False,inplace=True)
scorefinal.head(8)
```

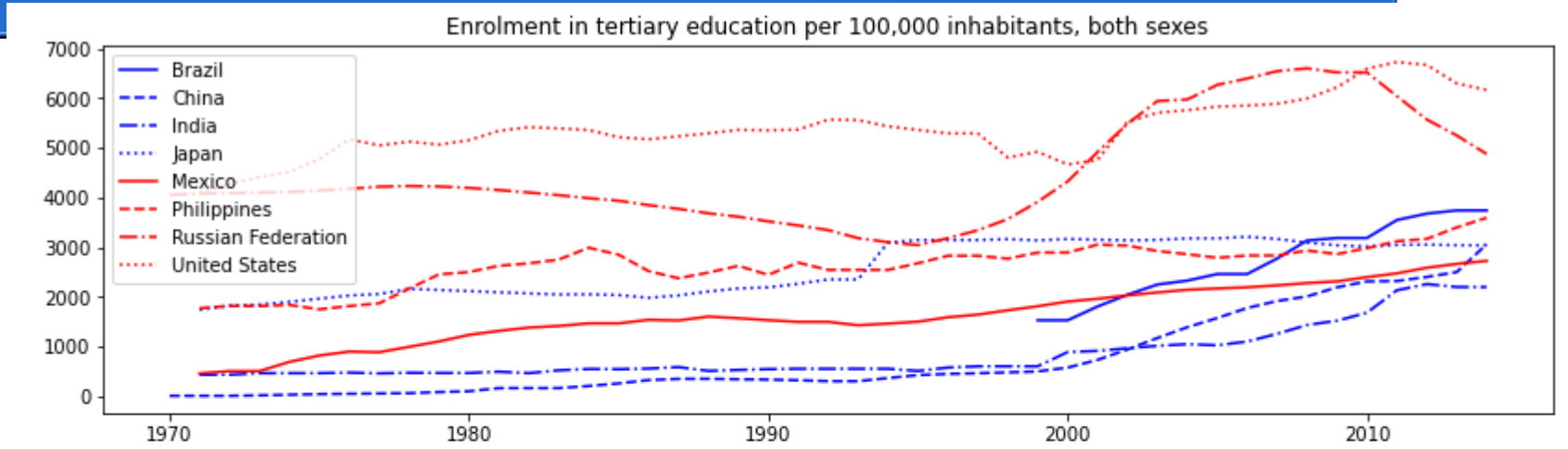
	Country	techno	now	future	Total
21	United States	3	3	3	9
3	China	3	3	3	9
9	India	3	3	3	9
1	Brazil	3	3	3	9
19	Russian Federation	3	3	2	8
17	Philippines	2	3	3	8
14	Mexico	2	2	3	7
12	Japan	3	2	2	7

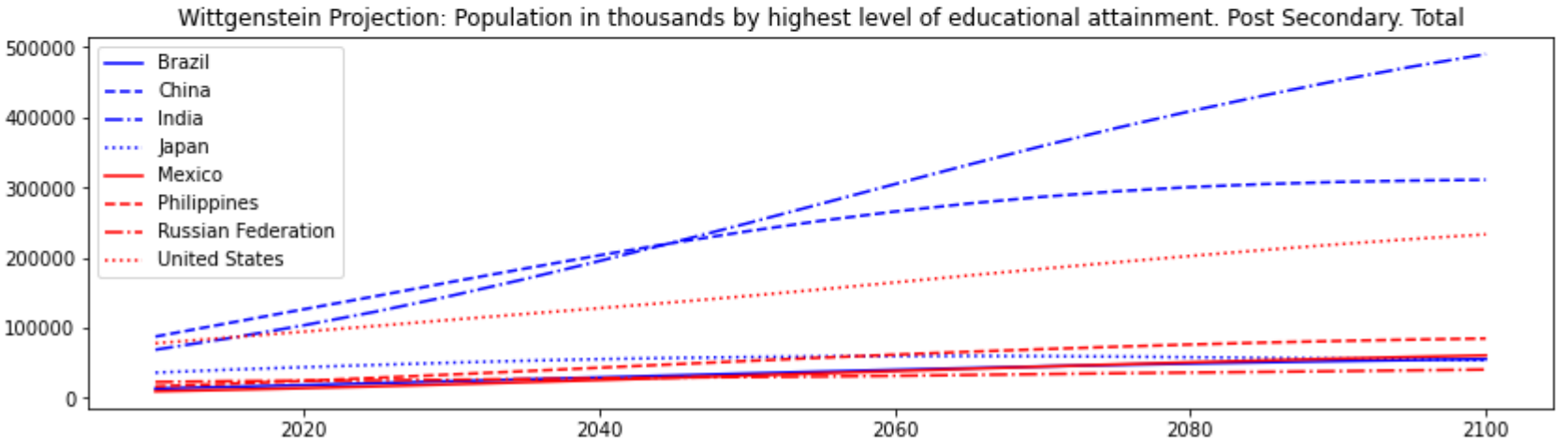
Proposition de choix de pays pour opérer en priorité

Pour finaliser regardons un peu plus en détail les courbes de certains indicateurs afin de voir un peu mieux les tendances:

- Les projections de fin du secondaires
- Les taux d’accès au supérieur

Proposition de choix de pays





Proposition finale

- Le top 3 est clairement :
  - Inde, Chine et USA

- Pour le top 5 j’ajouterai :
  - Philippines
  - Brésil ou Mexique

  Bonnes croissance d’accès au supérieur ces dernières années

  Bonnes perspectives d’accès à la fin du secondaire dans les prochaines décennies