



Le thème de la Data Analyse de porte sur la pandémie du covid-19

Groupe 8: Massi Djellouli; Fatima Brahmi; Mohamed Safi; Nouara Ouidkaci; Nazim GAOUIR

Importation des librairies nécessaire

```
In [44]: import pandas as pd
import matplotlib.pyplot as plt
import plotly.graph_objects as go
import plotly.express as px
import datetime
```

```
In [4]: #source :https://ourworldindata.org/coronavirus-source-data
```

Lecture de la Data Frame

```
In [5]: df = pd.read_csv("https://covid.ourworldindata.org/data/covid-covid-data.csv",index_col='date',low_memory=False)
```

```
In [6]: df
```

	iso_code	continent	location	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed
date									
2020-01-23	AFG	Asia	Afghanistan	NaN	0.0	NaN	NaN	0.0	NaN
2020-01-24	AFG	Asia	Afghanistan	NaN	0.0	NaN	NaN	0.0	NaN
2020-01-25	AFG	Asia	Afghanistan	NaN	0.0	NaN	NaN	0.0	NaN
2020-01-26	AFG	Asia	Afghanistan	NaN	0.0	NaN	NaN	0.0	NaN
2020-01-27	AFG	Asia	Afghanistan	NaN	0.0	NaN	NaN	0.0	NaN
...
2020-12-06	ZWE	Africa	Zimbabwe	10718.0	101.0	128.000	291.0	0.0	2.286
2020-12-07	ZWE	Africa	Zimbabwe	10839.0	121.0	127.000	294.0	3.0	2.571
2020-12-08	ZWE	Africa	Zimbabwe	10912.0	73.0	111.857	303.0	9.0	3.714
2020-12-09	ZWE	Africa	Zimbabwe	11007.0	95.0	125.429	304.0	1.0	3.857
2020-12-10	ZWE	Africa	Zimbabwe	11081.0	74.0	93.857	305.0	1.0	3.571

61819 rows × 49 columns

Affichage détaillé de différentes caractéristiques de la Data Frame

```
In [7]: df.shape
```

```
Out [7]: (61819, 49)
```

```
In [35]: df.dtypes
```

```
Out [35]: iso_code      object
continent    object
location     object
total_cases  float64
new_cases    float64
new_cases_smoothed  float64
total_deaths  float64
new_deaths   float64
new_deaths_smoothed  float64
total_cases_per_million  float64
new_cases_per_million   float64
total_deaths_per_million float64
new_deaths_per_million  float64
new_deaths_smoothed_per_million float64
reproduction_rate      float64
icu_patients            float64
icu_patients_per_million float64
hosp_patients           float64
hosp_patients_per_million float64
weekly_icu_admissions   float64
weekly_icu_admissions_per_million float64
weekly_hosp_admissions float64
weekly_hosp_admissions_per_million float64
new_tests               float64
total_tests             float64
total_tests_per_thousand float64
new_tests_per_thousand float64
new_tests_smoothed      float64
new_tests_smoothed_per_thousand float64
positive_rate           float64
tests_per_case          float64
tests_units             object
stringency_index        float64
population              float64
population_density      float64
median_age              float64
aged_65_older          float64
aged_70_older          float64
gdp_per_capita         float64
extreme_poverty         float64
cardiovasc_death_rate  float64
diabetes_prevalence     float64
female_smokers           float64
male_smokers             float64
handwashing_facilities float64
hospital_beds_per_thousand float64
life_expectancy         float64
human_development_index float64
date_column            object
dtype: object
```

```
In [26]: df.columns
```

```
Out [26]: Index(['iso_code', 'continent', 'location', 'total_cases', 'new_cases', 'new_cases_smoothed', 'total_deaths', 'new_deaths', 'new_deaths_smoothed', 'total_cases_per_million', 'new_cases_per_million', 'new_deaths_per_million', 'total_deaths_per_million', 'new_deaths_smoothed_per_million', 'reproduction_rate', 'icu_patients', 'icu_patients_per_million', 'hosp_patients', 'hosp_patients_per_million', 'weekly_icu_admissions', 'weekly_icu_admissions_per_million', 'weekly_hosp_admissions', 'weekly_hosp_admissions_per_million', 'new_tests', 'total_tests', 'total_tests_per_thousand', 'new_tests_per_thousand', 'new_tests_smoothed', 'new_tests_smoothed_per_thousand', 'positive_rate', 'tests_per_case', 'tests_units', 'stringency_index', 'population', 'population_density', 'median_age', 'aged_65_older', 'aged_70_older', 'gdp_per_capita', 'extreme_poverty', 'cardiovasc_death_rate', 'diabetes_prevalence', 'female_smokers', 'male_smokers', 'handwashing_facilities', 'hospital_beds_per_thousand', 'life_expectancy', 'human_development_index', 'date_column'],
dtype='object')
```

```
In [27]: #le pourcentage de cellule vide dans chaque colonne
(df.isna().sum() / 61565) * 100
```

```
Out [27]: iso_code      0.524649
continent    1.050922
location     0.000000
total_cases  14.566718
new_cases    0.203037
new_cases_smoothed  1.755868
total_deaths  27.453191
new_deaths   0.203037
new_deaths_smoothed  1.755868
total_cases_per_million  15.067002
new_cases_per_million   0.272496
new_cases_smoothed_per_million  2.272395
total_deaths_per_million  27.933079
new_deaths_per_million   0.727686
new_deaths_smoothed_per_million  2.272395
reproduction_rate      33.395598
icu_patients           92.583448
icu_patients_per_million  92.583448
hosp_patients          91.025745
hosp_patients_per_million  91.025745
weekly_icu_admissions  99.649151
weekly_icu_admissions_per_million  99.649151
weekly_hosp_admissions  99.163486
weekly_hosp_admissions_per_million  99.163486
new_tests              59.977260
total_tests            60.064972
total_tests_per_thousand  60.064972
new_tests_per_thousand  59.977260
new_tests_smoothed     55.331763
new_tests_smoothed_per_thousand  55.331763
positive_rate          58.504020
tests_per_case         59.491594
tests_units            53.608381
stringency_index       12.705271
population             2.634614
population_density     2.634614
median_age             2.634614
aged_65_older         6.307155
aged_70_older         5.771136
gdp_per_capita         4.733209
extreme_poverty       35.279786
cardiovasc_death_rate  3.787866
diabetes_prevalence    2.634614
female_smokers         25.309835
male_smokers           26.359133
handwashing_facilities 50.502721
hospital_beds_per_thousand 12.718265
life_expectancy        4.104927
human_development_index 4.210184
date_column           0.000000
dtype: float64
```

```
In [34]: # Remplacer les cellules qui sont des NaN par 0
df.fillna(0, inplace=True)
df
```

	iso_code	continent	location	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed
date									
2020-01-23	AFG	Asia	Afghanistan	0.0	0.0	0.000	0.0	0.0	0.000
2020-01-24	AFG	Asia	Afghanistan	0.0	0.0	0.000	0.0	0.0	0.000
2020-01-25	AFG	Asia	Afghanistan	0.0	0.0	0.000	0.0	0.0	0.000
2020-01-26	AFG	Asia	Afghanistan	0.0	0.0	0.000	0.0	0.0	0.000
2020-01-27	AFG	Asia	Afghanistan	0.0	0.0	0.000	0.0	0.0	0.000
...
2020-12-06	ZWE	Africa	Zimbabwe	10718.0	101.0	128.000	291.0	0.0	2.286
2020-12-07	ZWE	Africa	Zimbabwe	10839.0	121.0	127.000	294.0	3.0	2.571
2020-12-08	ZWE	Africa	Zimbabwe	10912.0	73.0	111.857	303.0	9.0	3.714
2020-12-09	ZWE	Africa	Zimbabwe	11007.0	95.0	125.429	304.0	1.0	3.857
2020-12-10	ZWE	Africa	Zimbabwe	11081.0	74.0	93.857	305.0	1.0	3.571

61819 rows × 50 columns

```
In [28]: #Afficher les statistiques rapide ;
df.describe()
```

```
Out [28]:
```

	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed	total_cases_per_million	n
count	5.285100e+04	61694.000000	60738.000000	4.491700e+04	61694.000000	60738.000000	52543.000000	
mean	2.727200e+05	2256.037832	2228.814717	8.347782e+03	51.280157	50.972719	3788.110146	
std	2.195398e+06	21572.786296	21059.985080	6.539131e+04	431.640602	417.099826	7694.437288	
min	1.000000e+00	-10.034.000000	-425.000000	1.000000e+00	-1918.000000	-232.143000	0.001000	
25%	3.790000e+02	0.000000	0.429000	2.200000e+01	0.000000	0.000000	78.655000	
50%	3.850000e+03	14.000000	19.714000	1.240000e+02	0.000000	0.286000	604.255000	
75%	3.793000e+04	263.000000	284.857000	1.063000e+03	4.000000	4.571000	3836.048000	
max	6.959255e+07	697958.000000	624498.571000	1.581856e+06	12848.000000	10897.857000	93056.364000	

8 rows × 45 columns

1- Analyse de la pandémie à l'échelle mondiale

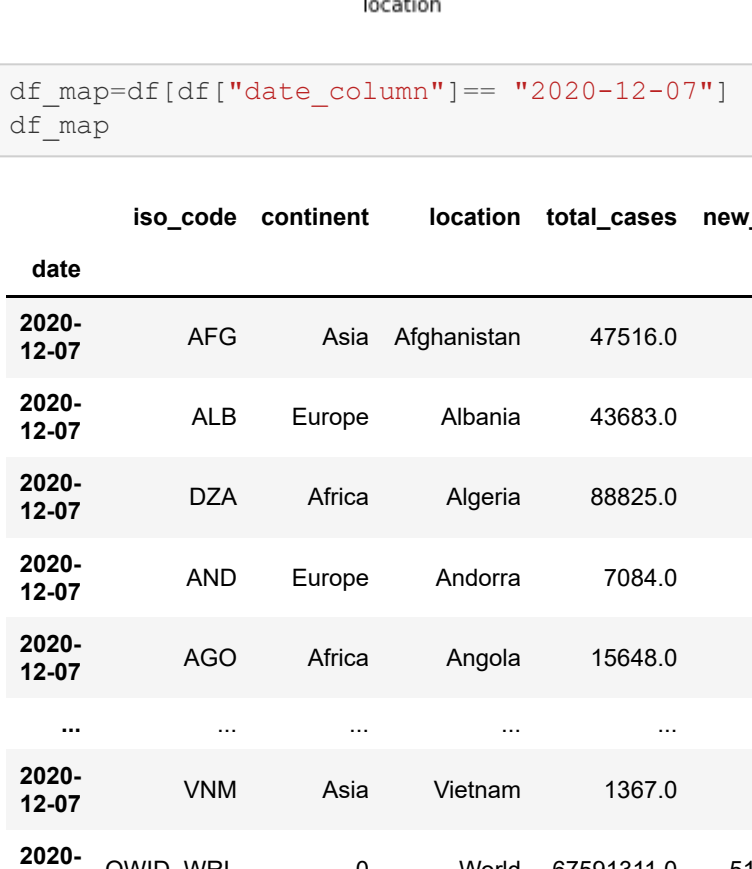
```
In [29]: #Duplication de la colonne date
df['date_column'] = df.index
```

```
In [30]: #Nouvelle Data Frame à l'échelle des continents grace à un groupage
df_cont=df.groupby('continent')['new_cases'].sum()
```

Graphique à bar qui schématise le nombre total de cas en million et par continent

```
In [31]: df_cont.plot.bar(x='continent',y='new_cases')
```

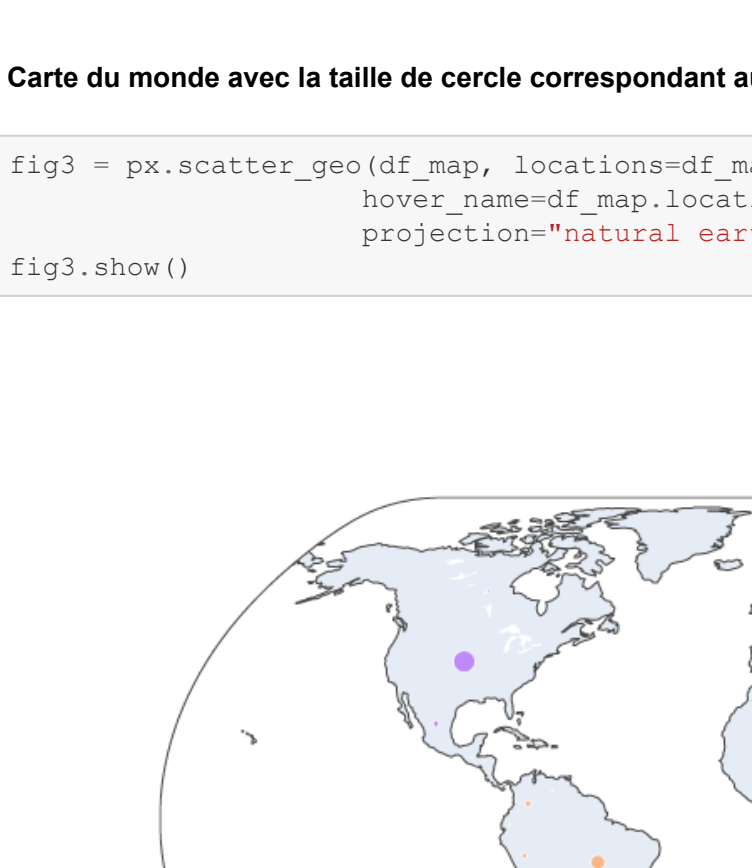
```
Out [31]: <matplotlib.axes._subplots.AxesSubplot at 0x1b1b49c2940>
```



Graphique à bar qui schématise le nombre d'hospitalisation par centaine depuis le début pandémie

```
In [32]: df['hospital_beds_per_thousand'].value_counts().plot.bar()
```

```
Out [32]: <matplotlib.axes._subplots.AxesSubplot at 0x1b1b40f24c0>
```



```
In [33]: df['hospital_beds_per_thousand'].hist()
```

```
Out [33]: <matplotlib.axes._subplots.AxesSubplot at 0x1b1b4898e20>
```



Graphique qui montre le nombre total de pertes humaine et par pays

```
In [23]: df_pays = df.groupby('location')['new_deaths'].sum()
```

```
In [24]: df_pays.plot.area(x='location',y='new_deaths')
```

```
Out [24]: <matplotlib.axes._subplots.AxesSubplot at 0x1b1b40f24c0>
```



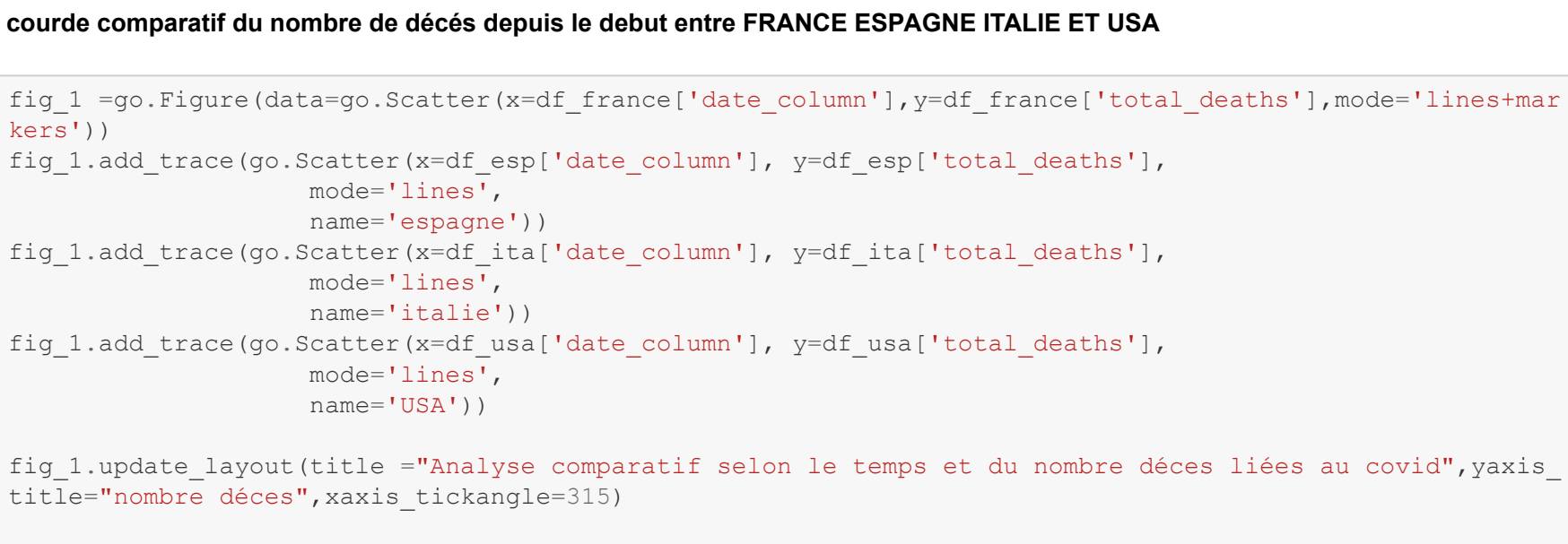
```
In [87]: df_map=df[df['date_column']=='2020-12-07']
df_map
```

	iso_code	continent	location	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed
date									
2020-12-07	AFG	Asia	Afghanistan	47516.0	210.0	177.429	1900.0	26.0	15.000
2020-12-07	ALB	Europe	Albania	43683.0	695.0	785.857	922.0	17.0	16.000
2020-12-07	DZA	Africa	Algeria	88825.0	573.0	803.714	2527.0	11.0	13.714
2020-12-07	AND	Europe	Andorra	7084.0	34.0	48.429	78.0	0.0	0.286
2020-12-07	AGO	Africa	Angola	15648.0	57.0	72.714	354.0	0.0	0.857
...
2020-12-07	VNM	Asia	Vietnam	1367.0	1.0	2.857	35.0	0.0	0.000
2020-12-07	OWID_WRL	0	World	67591311.0	517809.0	618047.000	1544532.0	8477.0	10867.857
2020-12-07	YEM	Asia	Yemen	2385.0	46.0	27.429	649.0	10.0	4.286
2020-12-07	ZMB	Africa	Zambia	17931.0	15.0	40.571	364.0	0.0	1.000
2020-12-07	ZWE	Africa	Zimbabwe	10839.0	121.0	127.000	294.0	3.0	2.571

191 rows × 50 columns

Carte du monde avec la taille de cercle correspondant au nombre total de cas recensés

```
In [93]: fig3 = px.scatter_geo(df_map, locations=df_map.iso_code, color=df_map.continent,
                           hover_name=df_map.location, size=df_map.total_cases,
                           projection='natural earth')
fig3.show()
```



```
In [16]: dfz = px.data.gapminder().query("year==2007")
dfz.dtypes
```

```
Out [16]: country      object
continent    object
year         int64
lifeExp     float64
pop         int64
gapPerCap   float64
iso_alpha   object
iso_num     int64
dtype: object
```

Création de nouveau Data Frame à partir de l'original en utilisant des masques

```
In [81]: df_france = df[df['location']=='France']
df_esp = df[df['iso_code']=='ESP']
df_ita=df[df['iso_code']=='ITA']
df_usa=df(df['iso_code']=='USA')
```

```
Out [81]:
```

	iso_code	continent	location	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed	tot
date										
2020-01-23	USA	North America	United States	1.0	0.0	0.000	0.0	0.0	0.000	
2020-01-24	USA	North America	United States	1.0	0.0	0.000	0.0	0.0	0.000	
2020-01-25	USA	North America	United States	2.0	1.0	0.000	0.0	0.0	0.000	
2020-01-26	USA	North America	United States	2.0	0.0	0.000	0.0	0.0	0.000	
2020-01-27	USA	North America	United States	5.0	3.0	0.000	0.0	0.0	0.000	
...	
2020-12-06	USA	North America	United States	14756882.0	175664.0	196235.429	282299.0	1113.0	2203.714	
2020-12-07	USA	North America	United States	14949417.0	192435.0	201176.000	283703.0	1404.0	2236.857	
2020-12-08	USA	North America	United States	15165295.0	219879.0	209210.429	286249.0	2546.0	2229.571	
2020-12-09	USA	North America	United States	15386562.0	221267.0	209233.286	289373.0	3124.0	2275.286	
2020-12-10	USA	North America	United States	15611014.0	224452.0	210200.714	292141.0	2768.0	2259.429	

324 rows × 50 columns

```
In [18]: #Conversion de type object à numerie
df_france['total_deaths']=pd.to_numeric(df_france['total_deaths'])
```

```
<ipython-input-18-bbdc2acd64f>:1: SettingWithCopyWarning:
A value is being set on a copy of a slice from a DataFrame.
Try using loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
df_france['total_deaths']=pd.to_numeric(df_france['total_deaths'])
```

2- Analyse a l'échelle de France

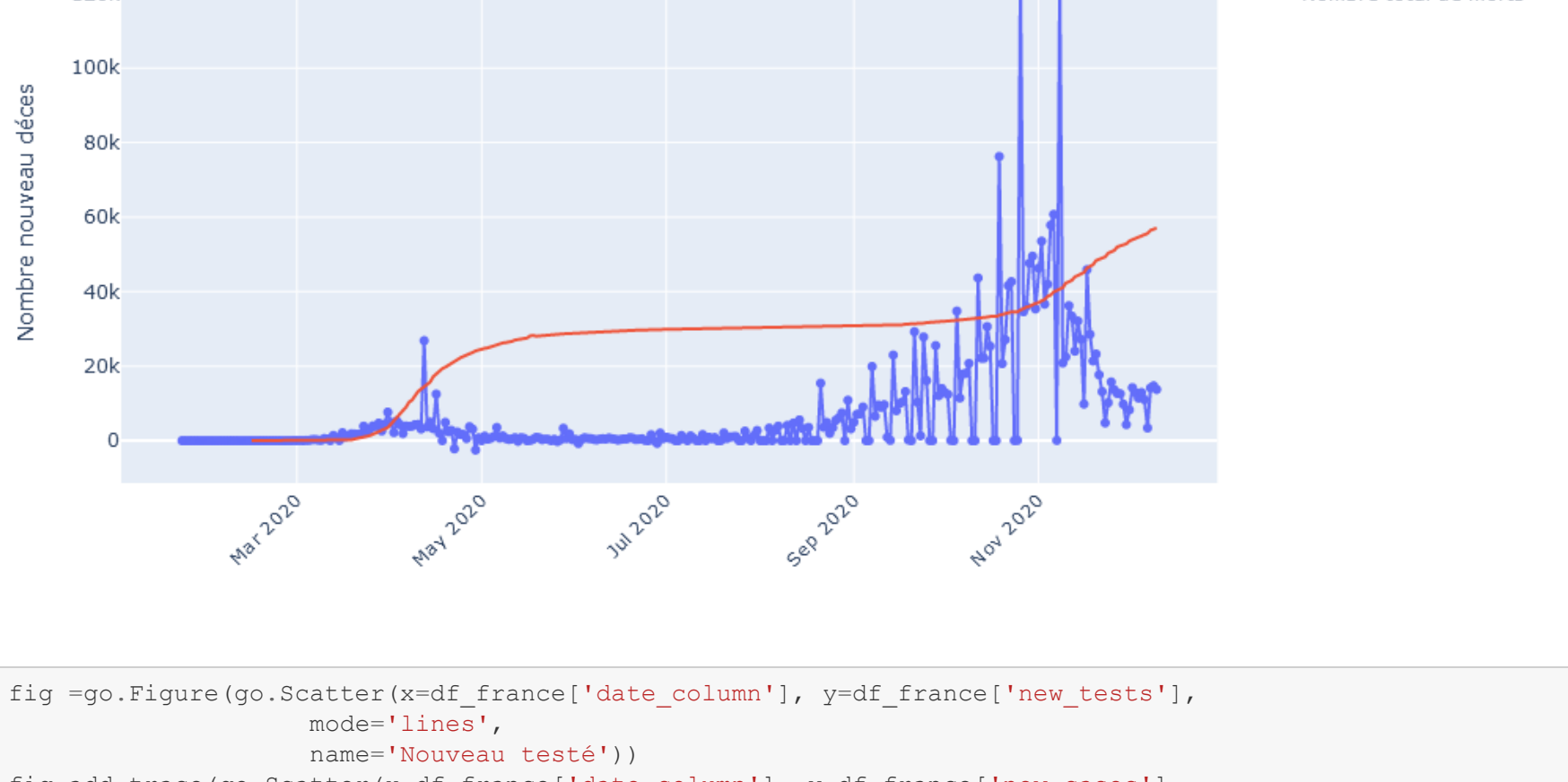
```
In [19]: df_france.columns
```

```
Out [19]: Index(['iso_code', 'continent', 'location', 'total_cases', 'new_cases', 'new_cases_smoothed', 'total_deaths', 'new_deaths', 'new_deaths_smoothed', 'total_cases_per_million', 'new_cases_per_million', 'new_deaths_per_million', 'total_deaths_per_million', 'new_deaths_smoothed_per_million', 'icu_patients', 'icu_patients_per_million', 'hosp_patients', 'hosp_patients_per_million', 'weekly_icu_admissions', 'weekly_icu_admissions_per_million', 'weekly_hosp_admissions', 'weekly_hosp_admissions_per_million', 'new_tests', 'total_tests', 'total_tests_per_thousand', 'new_tests_per_thousand', 'new_tests_smoothed', 'new_tests_smoothed_per_thousand', 'positive_rate', 'tests_per_case', 'tests_units', 'stringency_index', 'population', 'population_density', 'median_age', 'aged_65_older', 'aged_70_older', 'gdp_per_capita', 'extreme_poverty', 'cardiovasc_death_rate', 'diabetes_prevalence', 'female_smokers', 'male_smokers', 'handwashing_facilities', 'hospital_beds_per_thousand', 'life_expectancy', 'human_development_index', 'date_column'],
dtype='object')
```

courde comparatif du nombre de décès depuis le début entre FRANCE ESPAGNE ITALIE ET USA

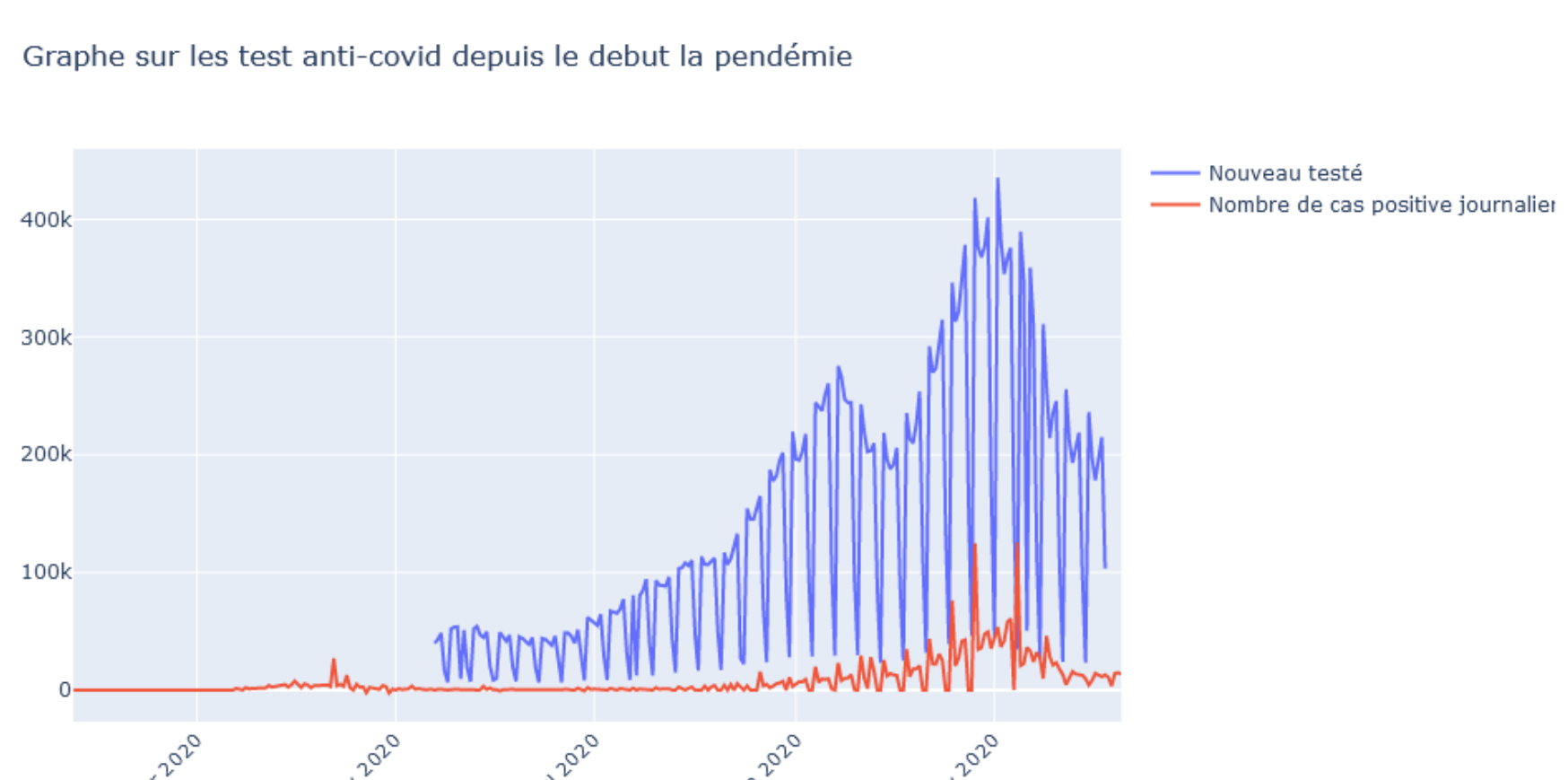
```
In [84]: fig1 = go.Figure(data=go.Scatter(x=df_france['date_column'],y=df_france['total_deaths'],mode='lines+markers'))
fig1.add_trace(go.Scatter(x=df_esp['date_column'], y=df_esp['total_deaths'],
                           mode='lines',
                           name='Espagne'))
fig1.add_trace(go.Scatter(x=df_ita['date_column'], y=df_ita['total_deaths'],
                           mode='lines',
                           name='Italie'))
fig1.add_trace(go.Scatter(x=df_usa['date_column'], y=df_usa['total_deaths'],
                           mode='lines',
                           name='USA'))
fig1.update_layout(title="Analyse comparatif selon le temps et du nombre décès liés au covid",yaxis_title="Nombre nouveau décès",xaxis_tickangle=315)
fig1.show()
```

Analyse comparatif selon le temps et du nombre décès liés au covid



```
In [21]: fig1 = go.Figure(data=go.Scatter(x=df_france['date_column'],y=df_france['new_deaths'],mode='lines+markers'))
fig1.update_layout(title="Analyse selon le temps du nombre nouveau décès",yaxis_title="Nombre nouveau décès",xaxis_tickangle=315)
fig1.show()
```

Analyse selon le temps du nombre nouveau décès



```
In [53]: fig = go.Figure(data=go.Scatter(x=df_france['date_column'],y=df_france['new_cases'],mode='lines+markers'))
fig.add_trace(go.Scatter(x=df_france['date_column'], y=df_france['total_deaths'],
                           mode='lines',
                           name='Nombre total de morts'))
fig.update_layout(title="Analyse selon le temps du nombre nouveau décès",yaxis_title="Nombre nouveau décès",xaxis_tickangle=315)
fig.show()
```

Analyse selon le temps du nombre nouveau décès

