UNIVERSITÉ PARIS 1
PANTHÉON SORBONNE

Master 2 Traitement de l'information et data-science en entreprise (TIDE)

Apprentissage Statistique

Projet

GROUPE :

DADDIO Melanie (a travaillé sur tout)

FIX Emma (a travaillé sur tout)

RAKOTOMANANA Zava (a travaillé sur l'ACP)

MARZOUK Moustafa (??)

Année universitaire 2024/2025

Table des matières

Introduction

La pandémie de COVID-19 a eu un impact majeur sur la santé publique dans le monde entier, et en particulier en Asie. L'objectif de cette étude est d'analyser les données COVID-19 pour identifier les facteurs qui ont influencé la propagation du virus et l'efficacité des mesures de santé publique dans différents pays asiatiques. Nous avons donc choisi d'analyser comment les pays asiatiques ont fait face à la pandémie en termes de mortalité, et lesquels semblent avoir nécessiter une aide extérieure.

Pour ce faire, nous commencerons par réaliser une analyse descriptive détaillée des données, afin de mieux comprendre les indicateurs disponibles et détecter d'éventuelles anomalies. Ensuite, nous réaliserons une étude détaillée des taux nous permettant d'avoir une première approche sur l'impact de la pandémie selon les pays et de repérer ceux en difficulté. Enfin, afin d'approfondir cette étude, nous réaliserons également une Analyse en Composantes Principales (ACP) afin d'éventuellement identifier des groupes de pays aux profils similaires.

I. Présentation de la base de l'étude et des variables

Les données utilisées proviennent de Worldometer, un site qui fournit des statistiques en temps réel sur divers indicateurs, y compris la pandémie de COVID-19. Le site compile des données provenant de sources fiables, telles que les gouvernements et les organisations de santé mondiale.

Nous utilisons l'extraction d'un tableau contenant des informations sur les cas de COVID-19, les décès, le nombre de tests réalisés et d'autres indicateurs de santé publique pour les pays asiatiques. Par ailleurs, nous n'avons pas de données concernant la Corée de Nord, cette suppression d'une observation peut être dû au fait que les données collectées sur ce pays étaient particulièrement étranges (ex : nombre de morts totale = 74).

Notre base de données est composée de 11 variables et 49 observations.

Voici les différentes variables :

- ID: Identifiant unique
- Country: Nom du pays
- TotalCases: Nombre total de cas enregistrés
- TotalDeaths: Nombre total de mort enregistrées
- TotalRecovered: Nombre total de gens ayant survécu
- ActiveCases: Nombre de gens qui ont actuellement le virus
- TotalCasesPerMillion: Nombre de cas enregistré par million d'individu
- TotalDeathsPerMillion: Nombre de morts enregistrées par million d'individu
- TotalTests: Nombre total de tests effectués (RTPCR + RAT + autres tests)
- TotalTestsPerMillion: Nombre de tests effectués par million d'individus
- TotalPopulation: Population du pays

II. Analyse détaillée des données

1. Statistiques univariées

skimpy summary

Data Summary

Dataframe	Values
Number of rows	49
Number of columns	11

Data Types

Column Type	Count
int64	6
float64	4
string	1

number

column	NA	NA %	mean	sd	p0	p25	p50	p75	p100	hist
ID	0	0	25	14.29	1	13	25	37	49	
TotalCases	0	0	1575000	4943000	77	115100	412100	905600	33970000	
TotalDeaths	1	2.0408163265306123	23730	68850	3	1360	5106	17800	450800	
TotalRecovered	0	0	1510000	4829000	64	91070	391400	847500	33290000	
ActiveCases	0	0	41390	89900	0	1670	14790	29240	484900	
TotalCasesPerMillion	0	0	43870	46260	67	6484	24310	78710	159700	
TotalDeathsPerMillion	1	2.0408163265306123	455.4	489.4	3	128.5	326.5	626.8	2329	
TotalTests	1	2.0408163265306123	29530000	86520000	4951	2335000	7978000	25100000	583600000	
TotalTestsPerMillion	1	2.0408163265306123	1173000	1864000	5741	141000	424800	1032000	8688000	
TotalPopulation	0	0	94950000	281800000	442800	5252000	17010000	51330000	1439000000	

string

column	NA	NA %	shortest	longest	min	max	chars per row	words per row	total words
Country	0	0	UAE	Saudi Arabia	Afghanistan	Yemen	7.24	1.1	53

End

Toutes les variables sont des variables quantitatives, mise à part la variable *Country* qui est une variables qualitative.

Les variables *TotalDeaths*, *TotalDeathsPerMillion*, *TotalTests* et *TotalTestsPerMillion* contiennent chacune une valeur manquante :

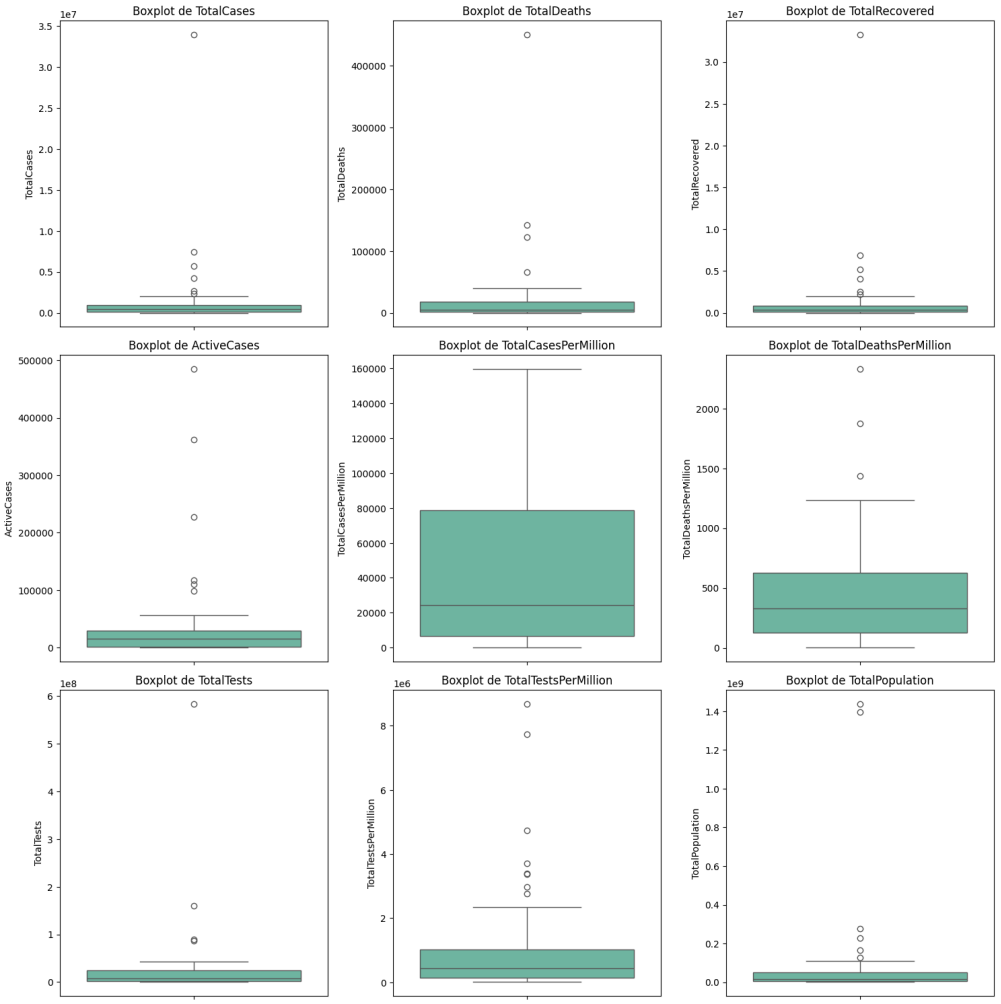
- Pour les variables *TotalDeaths* et *TotalDeathsPerMillion*, il manque des données pour le pays Macao.
- Pour les variables *TotalTests* et *TotalTestsPerMillion*, il manque des données pour le pays du Tadjikistan.

Il est important de souligner que pour ces pays les valeurs des variables renseignées sont inférieures à la moyenne. Il faudra donc prêter une attention particulière à ces deux pays pour la suite de notre étude.

A présent, si on analyse plus en détail le tableau de description de chaque variable, on constate que les écarts types sont souvent supérieurs ou proches de la moyenne, cela indique que les valeurs sont très dispersées. Cette dispersion peut être dû à la différence de taille de la population des pays qui n'ont ainsi pas été touchés de la même manière par la covid.
De plus, on peut supposer une présence de valeurs extrêmes qui augmente la dispersion des données, notamment pour les pays surpeuplés comme l'Inde. Par exemple, *TotalTests* a un maximum de 5,8 milliards alors que la médiane est seulement de 7,9 millions.
La variable *TotalPopulation* montre un écart conséquent entre son minimum à 4,4 millions et un maximum à 1,4 milliard ce qui tend à confirmer que la taille de la population peut jouer dans la variabilité des valeurs des autres variables.
Enfin la médiane est souvent inférieure à la moyenne ce qui indique une distribution asymétrique à droite, ce qui est répandue dans le cas des épidémies.

A) Analyse graphique des variables quantitatives

Afin de faire une analyse plus détaillée nous traçons les histogrammes de chaque variable quantitative ainsi que leur boxplot.



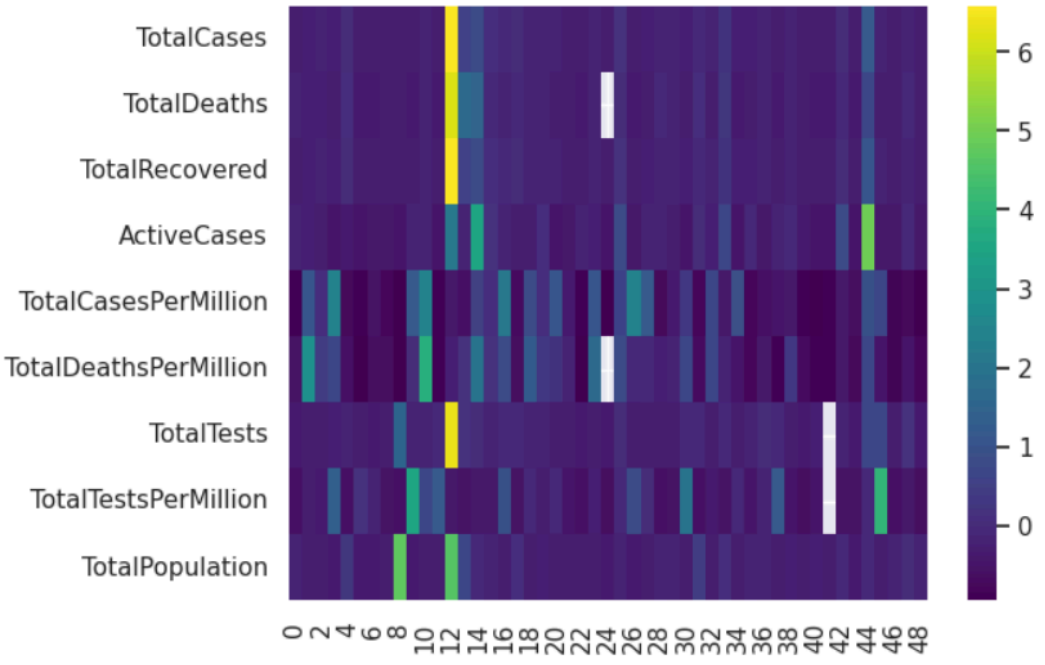
- *TotalCases*, *TotalDeaths*, *TotalRecovered*, *ActiveCases*, *TotalTests* et *TotalPopulation*

Tout d'abord, on constate que les boxplot sont très tassés ce qui peut être dû à la présence de valeurs extrêmes très éloigné de la valeur du dernier quartile.
Chaque boxplot de ces variables a une allure plutôt similaire ce qui est logique puisqu'elles semblent liées entre elles.

- *TotalCasesPerMillion*, *TotalDeathsPerMillion* et *TotalTestPerMillion*

Pour les variables "par million", la répartition est plus étalée, ce qui est sans doute dû au fait que ces variables sont rapportées à la taille de la population, les valeurs extrêmes sont donc bien moins écartées du boxplot que pour les variables précédentes. Ainsi les variables "par million " semblent un peu plus pertinentes pour notre étude.

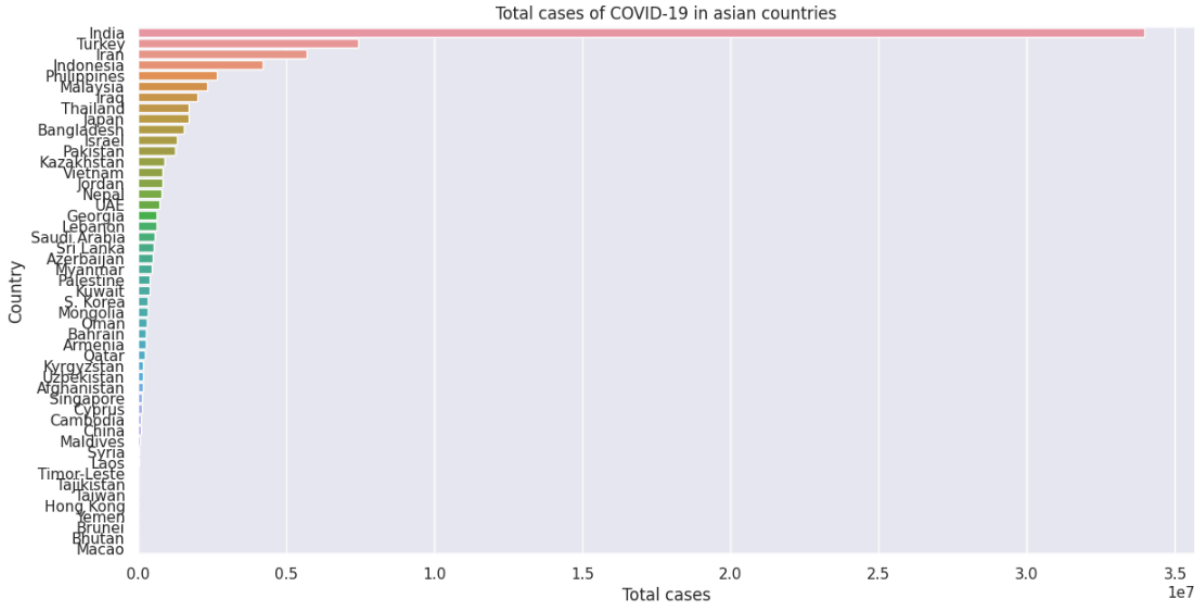
Nous avons repéré grâce aux boxplot ci-dessus la présence de valeurs extrêmes dans les données uniquement présentes dans les variables non pondérées par la population, ce qui montre là encore l'intérêt de prendre en compte la taille de la population.
Aussi, afin de savoir si elles sont réparties sur les observations où si elles se concentrent sur certaines, nous traçons une heatmap sur les variables quantitatives.

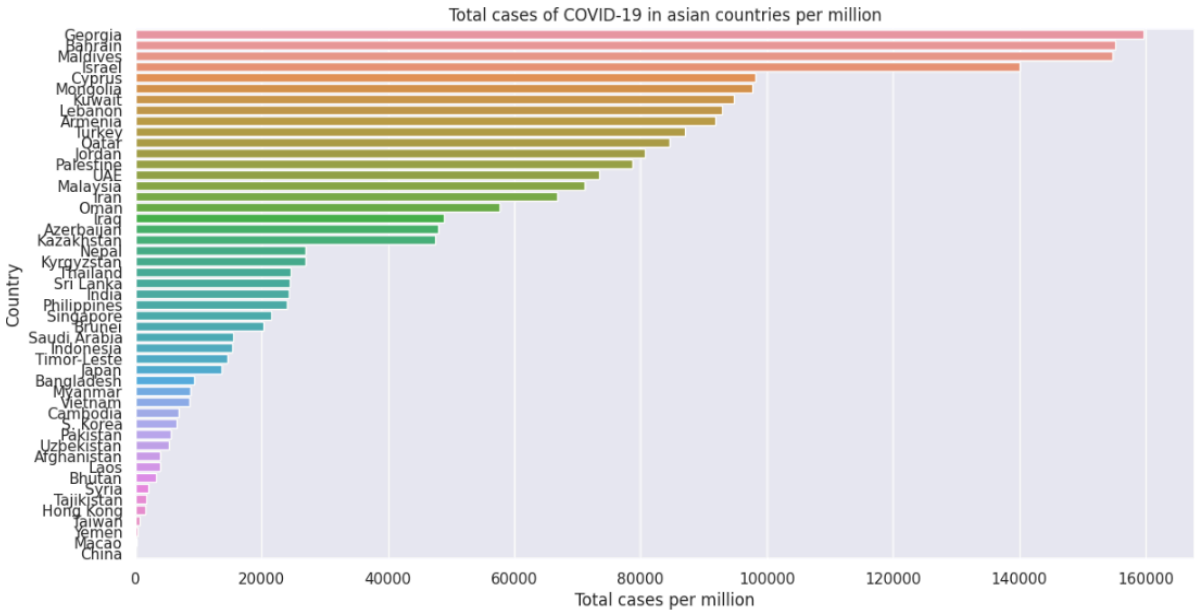


On peut ainsi voir que les valeurs extrêmes ne semblent pas être réparties sur les observations mais plutôt se concentrer sur l'observation 12, qui correspond à l'Inde dans notre base, ce qui semble plutôt logique.

B) Comparaison des variables pondérées et non pondérées en fonction des pays

En traçant des diagrammes en barre des différentes variables quantitatives non pondérées par la population et des variables pondérées, on peut remarquer que nous n'obtenons pas le même classement des pays.
Par exemple, en comparant le nombre total de cas par pays et le nombre de cas par million d'individus par pays, on peut remarquer que l'Inde qui était à la première place est reléguée à la 25e place dans le second diagramme.
Aussi, en comparant le nombre total de mort par pays et le nombre de mort par million d'individus par pays, la Chine qui est le pays le plus peuplé au monde passe de 23e à avant dernière. Il est tout de même important de souligner que le fait que la chine, pays le plus peuplé au monde, soit 23eme du classement en nombre de morts est tout de même surprenant. On pourrait donc se questionner sur la véracité des données.





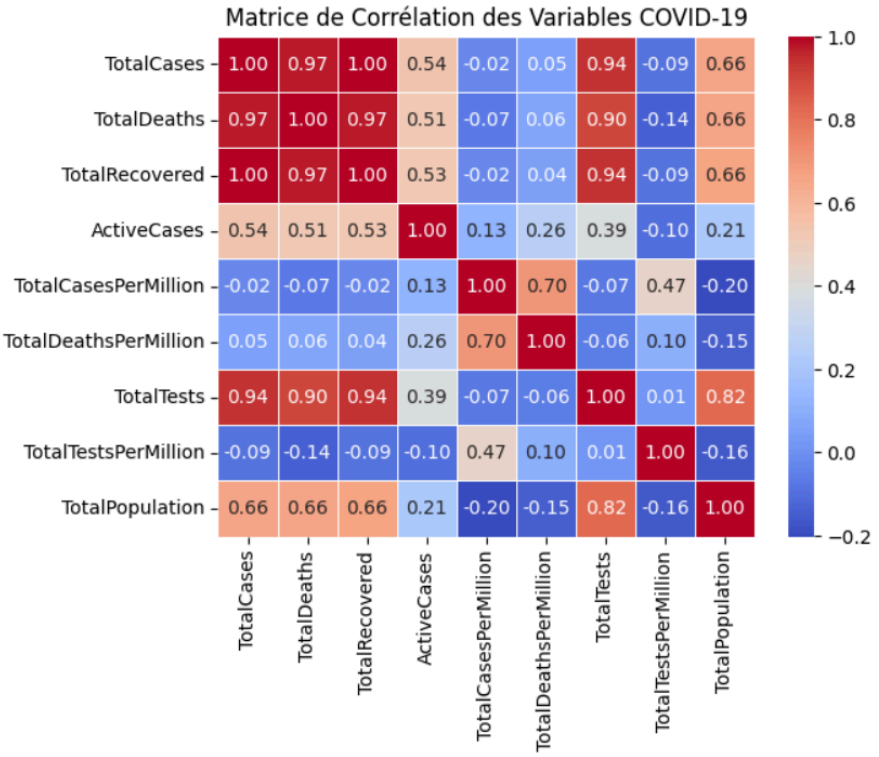
Ces visualisations soulignent à nouveau qu'il sera important dans la suite de l'étude de prendre en compte la taille de la population du pays en fonction des analyses que nous réalisons et de distinguer les variables *PerMillion* des variables "classiques".

2. Statistiques bivariées

A présent, il serait intéressant d'analyser les relations entre les variables.
Pour commencer, nous allons regarder si certaines variables sont corrélées. Par exemple, pour savoir si le nombre de décès est lié à la taille de la population du pays, le nombre de cas touchés par la covid et le nombre de tests effectués.

A) Corrélation

Nous traçons la matrice de corrélations des variables numériques afin d'avoir une première idée du lien entre les variables.



Coefficient de corrélation très élevé (> 0.8)
On constate directement que le nombre de personnes mortes ou ayant survécu, le nombre de cas et le nombre des tests sont très corrélées positivement les unes avec les autres.
Ceci est parfaitement logique puisqu'un pays ayant eu beaucoup de cas aura forcément un nombre plus élevé de tests effectués qu'un pays avec peu de cas (a supposé que le pays encourage les tests pour faire face à la maladie), ainsi qu'un nombre de morts ou survivants élevé.

Le seul autre coefficient de corrélation très élevé est celui entre *TotalTests* et *TotalPopulation* ce qui semble plutôt logique, plus la population est grande plus le nombre de tests effectué devrait être grand (avec la même supposition que précédemment).

Coefficient de corrélation relativement élevée (> 0.4)
La variable *ActiveCases* et *TotalPopulation* sont moyennement corrélée positivement avec les variables *TotalCases*, *TotalDeaths* et *TotalRecovered*. Nous pouvons interpréter cela comme le fait que le nombre total de cas, morts et survivants évolue parallèlement au nombre de cas actifs mais pas au même rythme ce qui est plutôt intuitif. Aussi le niveau de population d'un pays joue sur le nombre de cas et par conséquent de décès et de survivant, ce qui est encore là, logique.
Il est important de noter que les variables *TotalCasesPerMillion* et *TotalDeathsPerMillion* ont un coefficient de corrélation élevé, mais tout de même moins que *TotalCases* et *TotalDeaths*.

Coefficient de corrélation proche de 0
En revanche, il est important de noter que la corrélation des variables *TotalCasesPerMillion, TotalTestsPerMillion* et *TotalDeathsPerMillion* avec *TotalCases, TotalDeaths, TotalTests* et *TotalRecovered* est très proche de 0, ce qui peut sembler contre intuitif au premier abord.
Nous allons donc étudier cela plus en détail dans la suite en effectuant des tests.

Préalablement au test, afin de pouvoir les réaliser, nous devons imputer des valeurs aux données manquantes. Nous le faisons par méthode du plus proche voisin car nous avons des données qui varient beaucoup d'un pays à l'autre, donc nous utiliserons seulement les 3 pays les plus proches (considérant les variables quantitatives) des deux pays avec des données manquantes pour imputer des valeurs.

B) Test d'indépendance : test de Pearson

Les tests d'indépendances permettent de définir s'il existe un lien entre deux variables, ainsi que d'exclure des variables explicatives potentiellement non porteuses d'information.
Les tests statistiques sont très sensibles à la taille de l'échantillon. Un même coefficient de corrélation n'aura pas la même significativité sur un petit échantillon (ici le cas) que sur un grand échantillon.

Test d'indépendance : test de Pearson
Nous avons décidé d'utiliser ce test car il est adapté aux variables quantitatives.
Son intérêt est d'apporter plus de pertinence et fiabilité aux coefficients de corrélation.
Pour choisir les variables dont nous allons tester la corrélation, nous utilisons la matrice de corrélation que nous avons réalisée précédemment.
On pose les hypothèses de départ :

- H0 : Variables indépendantes si p-value > 0.05
- H1 : Variables non indépendantes si p-value < 0.05

Voici quatre tests à titre d'exemple :

TotalCases vs TotalDeaths

```
PearsonRResult(statistic=0.9745004403630758, pvalue=3.7247188614058195e-32)
```

La première sortie correspond au coefficient de corrélation, il correspond bien à la valeur du coefficient de la matrice de corrélation. La seconde à la p-value qui est ici inférieur à 0.05, ainsi, les deux variables ne sont pas indépendantes comme nous l'avions déduit avec la matrice de corrélation.

TotalCasesPerMillion vs TotalDeathsPerMillion

```
PearsonRResult(statistic=0.7012758113319598, pvalue=1.999930048420256e-08)
```

On peut faire les mêmes déductions que ci-dessus pour ces deux variables.

Voici deux des tests et résultats les plus pertinents :

TotalCases vs TotalDeathsPerMillion

```
PearsonRResult(statistic=0.04862154045405934, pvalue=0.7400698527596838)
```

On peut voir que comme sur la matrice de corrélation, le coefficient de corrélation entre les deux variables est proche de 0 et la p-value est supérieur à 0.05. On conclut donc que les deux variables, *TotalCases* et *TotalDeathsPerMillion* sont indépendantes.
Or cela ne paraît pas logique par rapport aux deux résultats que nous avons eu ci-dessus. Pourtant il l'est bien, et cela s'explique par une différence d'échelle entre les deux variables. En effet, le test de Pearson mesure la relation linéaire standardisée entre deux variables ce qui n'est pas le cas ici.

TotalCases vs TotalCasesPerMillion

```
PearsonRResult(statistic=-0.019727020222367612, pvalue=0.892977683067629)
```

Ici aussi on pourrait s'attendre à un résultat différent, mais en effet, nous avons :
TotalCases = Population × TotalCasesPerMillion / 1_000_000
TotalCasesPerMillion ne suit pas une relation linéaire simple avec *TotalCases,* la population agit ici comme un facteur caché.

On utilise la corrélation partielle pour voir si, une fois l'effet de la population retiré, la corrélation devient forte et on obtient :

	n	r	CI95%	p-val
pearson	49	0.155497	[-0.13, 0.42]	0.291264

Nous restons donc sur la même conclusion que précédemment : les variables *TotalCases* et *TotalCasesPerMillion* ne sont pas corrélées. Nous supposons que ce résultat étonnant est dû à la population variant fortement d'un pays à l'autre,ainsi nous n'avons pas une relation linéaire entre *TotalCases et TotalCasesPerMillion*. Malgré cela, on peut voir que la corrélation entre les deux variables passe de proche de 0 et 0.15 une fois l'effet de la population retiré, ce qui prouve que la taille de la population joue en effet sur la corrélation entre les deux variables.

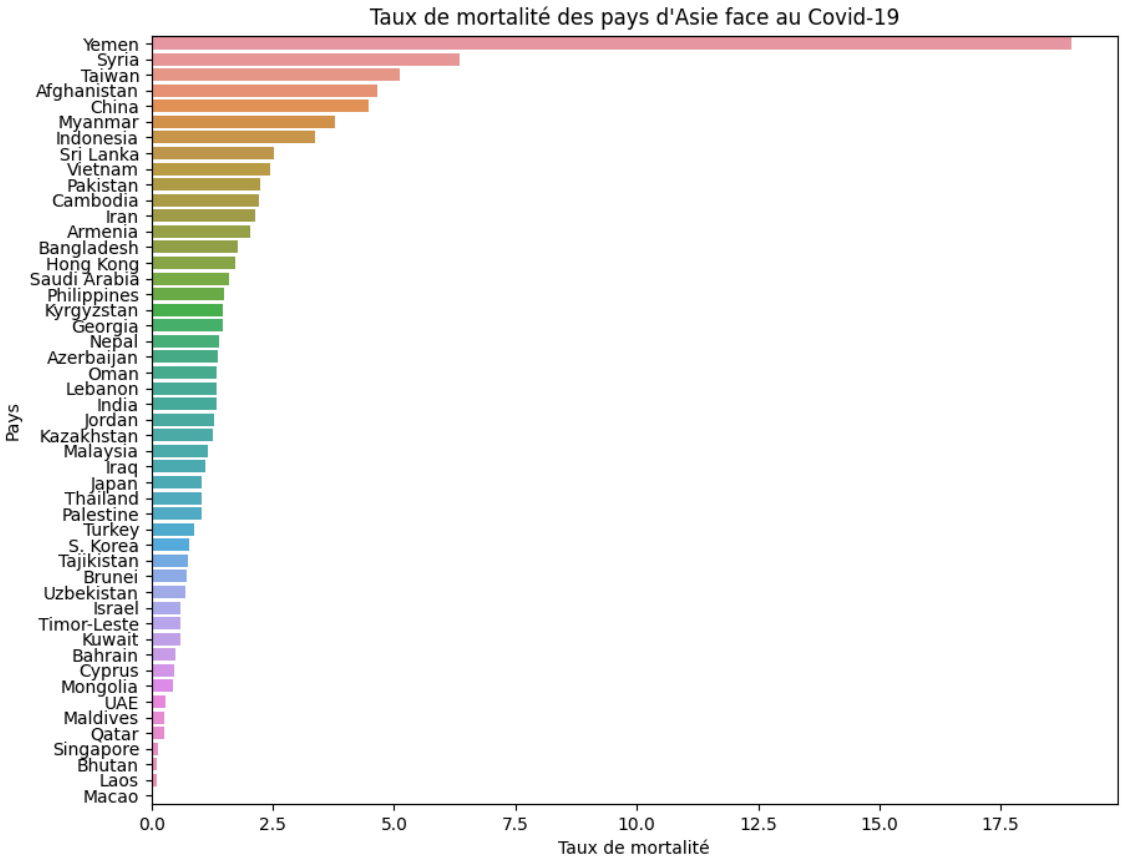
III. Étude des taux

Nous avons déterminé précédemment que les variables “par millions” étaient plus pertinentes. Nous avons donc décidé de les utiliser pour répondre à notre problématique, à savoir si certains pays s’en sortent mieux que d’autres ou si certains nécessitent une aide particulière.

Pour cela, nous avons commencé par calculer les taux, nous avons représenté graphiquement les résultats (classement des pays), puis nous avons instauré des seuils afin d’isoler les pays les plus en difficulté.

Taux de mortalité : *TotalDeathsPerMillion / TotalCasesPerMillion*

Nous avons choisi de calculer le taux de mortalité des différents pays face à la pandémie du Covid-19 car c’est un indicateur clé. Il peut nous permettre de détecter un pays dans une situation critique, qui n’a pas un système de santé efficace : manque d’accès aux soins, manque de mesures prises, pandémie mal contrôlée… et qui pourrait ainsi nécessiter de l’aide.
Il nous permet aussi de comparer les pays entre eux étant donné que l’on prend pour les calculer les variables qui prend en compte la taille de la population.



On ne prend pas en compte le résultat pour Macao dans notre analyse car la valeur de *TotalDeathsPerMillion* est manquante.

Taux de récupération : *TotalRecoveredPerMillion / TotalCasesPerMillion*

Nous avons aussi calculé le taux de récupération des différents pays face à la pandémie du Covid-19. Cet indicateur est complémentaire au premier, car on peut considérer que lorsqu'une personne a le Covid-19, soit elle se rétablit, soit elle décède, mis à part les cas encore actifs où nous n'avons pas encore la données concernant un rétablissement ou un décès.

Nous considérerons que le taux de mortalité est plus fiable que le taux de récupération car un décès sera quasi automatiquement déclaré (certificat de décès).

Instauration de seuils

Afin de faire ressortir de la liste des pays d'Asie les pays en grande difficulté, nous avons instauré des seuils. Les seuils de cas et de mortalité ont été mis à 0.9, ainsi nous faisons ressortir les 5 pays qui ont le plus de cas par millions d'habitants et les 5 pays qui accusent le plus de mortalité face à la pandémie. Nous instaurons aussi un seuil à 0.1 afin de faire ressortir les 5 pays ayant réalisé le plus de tests par million d'habitants.

```
Liste des pays au dessus du seuil de cas :

      Country  TotalCasesPerMillion  TotalTestsPerMillion  TauxMortalite
3    Bahrain                155162             3713470.0       0.503989
9     Cyprus                 98184             7731251.0       0.467490
10   Georgia                159718             2336217.0       1.458195
16    Israel                140074             2973538.0       0.606108
26  Maldives                154797             2765865.0       0.273261

Liste des pays au dessus du seuil de mortalité :

         Country  TotalCasesPerMillion  TotalTestsPerMillion  TauxMortalite
0    Afghanistan                  3884               19103.0       4.660144
8          China                    67              111163.0       4.477612
39         Syria                  2048                5741.0       6.347656
40        Taiwan                   683              282957.0       5.124451
48         Yemen                   306                8651.0      18.954248

Liste des pays en dessous du seuil de tests :

         Country  TotalCasesPerMillion  TotalTestsPerMillion  TauxMortalite
0    Afghanistan                  3884               19103.0       4.660144
24         Macao                   117                7495.0            NaN
39         Syria                  2048                5741.0       6.347656
46    Uzbekistan                  5241               40425.0       0.705972
48         Yemen                   306                8651.0      18.954248

Liste des pays qui dépassent plusieurs seuils :

         Country  TotalCasesPerMillion  TotalTestsPerMillion  TauxMortalite
0    Afghanistan                  3884               19103.0       4.660144
39         Syria                  2048                5741.0       6.347656
48         Yemen                   306                8651.0      18.954248
```

Analyse

L’analyse des données présentées montre de grandes différences entre les pays dans leur gestion de la pandémie de COVID-19. Ces différences apparaissent notamment dans le nombre de cas par million d’habitants, le taux de mortalité et le nombre de tests réalisés.

Certains pays, comme Bahreïn, Chypre, la Géorgie, Israël et les Maldives, enregistrent un nombre très élevé de cas par million d’habitants. Cela peut s'expliquer par une forte propagation du virus, mais aussi par une capacité à détecter les cas grâce à un dépistage massif. Par exemple, Chypre et Israël effectuent un grand nombre de tests, ce qui peut expliquer pourquoi ils détectent autant de cas. Pourtant, aucun de ces pays ne figure parmi ceux ayant un taux de mortalité très élevé, ce qui laisse penser que leur stratégie de dépistage a pu jouer un rôle dans la limitation du nombre de décès.

D'autres pays, comme l’Afghanistan, la Chine, la Syrie, Taïwan et le Yémen, présentent un taux de mortalité plus élevé. Le Yémen est celui qui affiche le taux de mortalité le plus important, avec 18,95%, soit près de trois fois plus que la Syrie, qui arrive en deuxième position. La situation en Syrie est particulièrement préoccupante, notamment à cause de la guerre qui a gravement affaibli ses infrastructures de santé. Beaucoup de malades du COVID-19 n'ont probablement pas eu accès à des soins adaptés, ce qui complique l'analyse de son taux de mortalité.

À l’opposé, certains pays ont réussi à limiter la mortalité. Singapour, avec 27 décès par million d'habitants, et le Bhoutan, avec seulement 4 décès par million, font partie des pays les moins touchés en termes de mortalité. Cela pourrait s'expliquer par des politiques de dépistage et de gestion sanitaire plus efficaces.

Certains pays, comme l’Afghanistan, Macao, la Syrie, l’Ouzbékistan et le Yémen, ont effectué un nombre très faible de tests par million d’habitants. Cela peut indiquer un manque de moyens ou une absence de stratégie de dépistage efficace. De plus, un faible nombre de tests signifie souvent que de nombreux cas ne sont pas détectés, ce qui rend le nombre réel de contaminations difficile à estimer. Ainsi, dans ces pays, un taux de mortalité élevé peut être en partie lié à l’absence de détection précoce et à un accès limité aux soins.

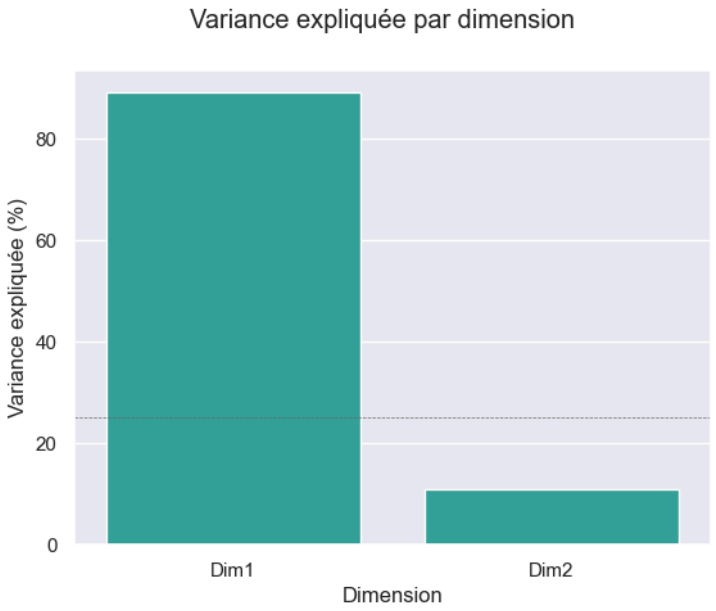
On remarque aussi que l’Afghanistan, la Syrie et le Yémen figurent à la fois parmi les pays ayant un faible dépistage et un taux de mortalité élevé. Cela met en lumière les difficultés qu’ils rencontrent, notamment en matière d’accès aux soins et de gestion de la pandémie. Le manque de tests empêche d’avoir une vision claire de la situation sanitaire et complique la mise en place de mesures efficaces. C'est peut-être aussi la raison pour laquelle ces pays n'apparaissent pas parmi ceux ayant officiellement déclaré un grand nombre de cas.

IV. Analyse en composantes principales

Afin de pousser notre analyse plus loin et de ne pas seulement baser nos résultats sur l'étude des taux, nous avions jugé pertinent de réaliser une Analyse en Composantes Principales (ACP). En effet, l'ACP est généralement utilisée dans deux situations : pour réduire la dimensionnalité des données ou pour la représentation des données. Étant donné que l'objectif de notre étude est d'identifier les pays les plus touchés de ceux ayant mieux géré la crise, nous utiliserons l'ACP dans le but d'observer des similitudes entre les pays.

Pour ce faire, nous avons décidé de nous concentrer sur les variables « par million » comme au préalable dans notre analyse. Nous avons testé différentes configurations de ces variables et avons constaté que l'exclusion de la variable *TotalTestsPerMillion* améliore significativement l'interprétation des résultats. Le nombre de tests réalisés ne reflète pas nécessairement la gestion de la pandémie comme vu précédemment. En effet, un pays développé peut avoir effectué un grand nombre de tests par précaution, mais de même un pays submergé par la pandémie peut également avoir dû réaliser un grand nombre de tests en urgence. Ainsi, prendre en compte le nombre de tests ne permet pas de comparer efficacement les pays, car cette variable peut être influencée par des facteurs indépendants de la gravité de l'épidémie.

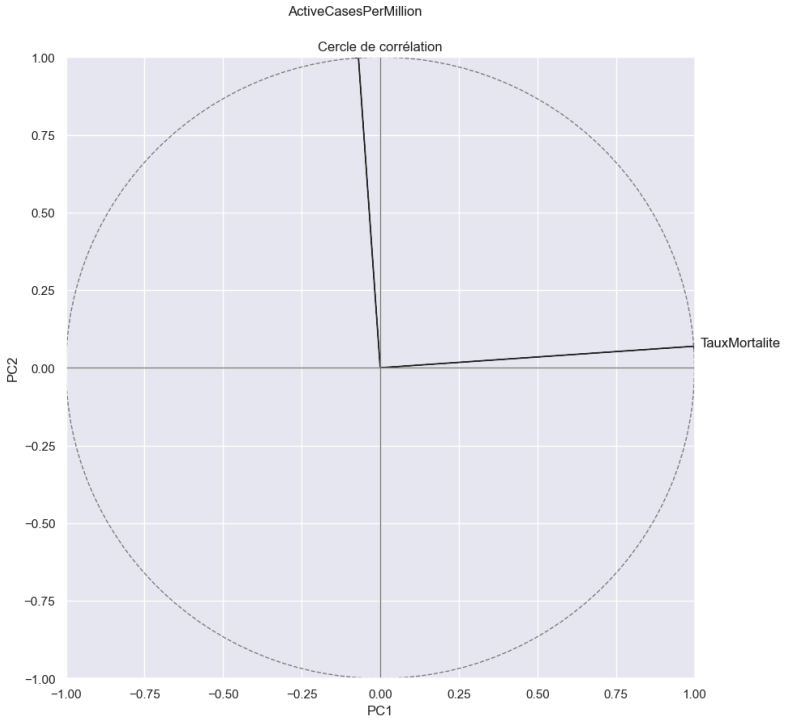
Dans un premier temps, il nous a paru judicieux de nous concentrer sur le nombre de cas actifs par million (que nous avons recalculé à partir de la variable *ActiveCases*) et le taux de mortalité. En effet, ces deux variables à elles seules remplissent selon nous tous les critères pour indiquer comment un pays a fait face à la pandémie.
Il est important de souligner que nous aurions pu utiliser la variable sur taux de récupération, mais comme émis précédemment ses informations sont complémentaires à celles du taux de mortalité.



Nous commençons par regarder quel plan factoriel expliquent au mieux la variance des données. On constate que les deux premiers axes expliquent presque toute la variance des données, avec 89 % pour le premier plan factoriel et 11 % pour le second plan factoriel . Ainsi, presque toute la partie des variations entre les pays peut être résumée par le premier plan factoriel.Cela indique que la majeure partie des variations entre les pays peut être résumée en un seul axe principal, tandis que la seconde composante apporte une information complémentaire mais assez faible.

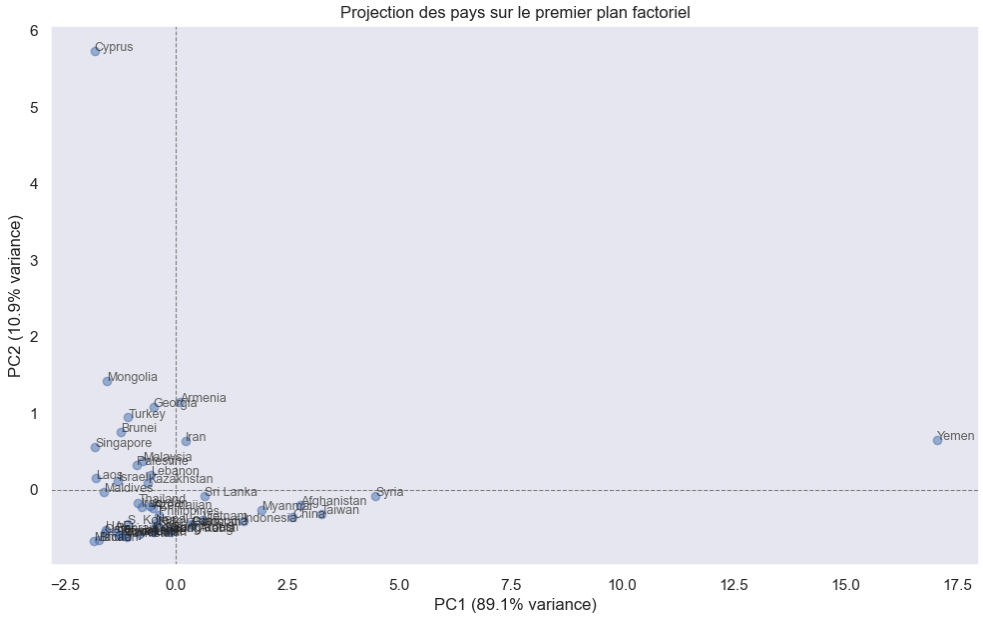
Par la suite, nous avons réalisé une analyse des valeurs propres. Cette dernière confirme que le premier plan factoriel possède une valeur propre bien supérieure (8.098534) à celle du second plan (0.986797), ce qui nous confirme son importance dans la structuration des données.

Puis, quand on se penche sur les contributions des variables à la formation des axes, on constate que le taux de mortalité est fortement corrélé avec le premier plan factoriel avec une contribution de 8.059775, ce qui indique que cette variable joue un rôle majeur dans la différenciation des pays sur cet axe. À l'inverse, le nombre de cas actifs par million d'habitants est principalement représenté sur le second plan factoriel avec une contribution équivalente de 0.982074.



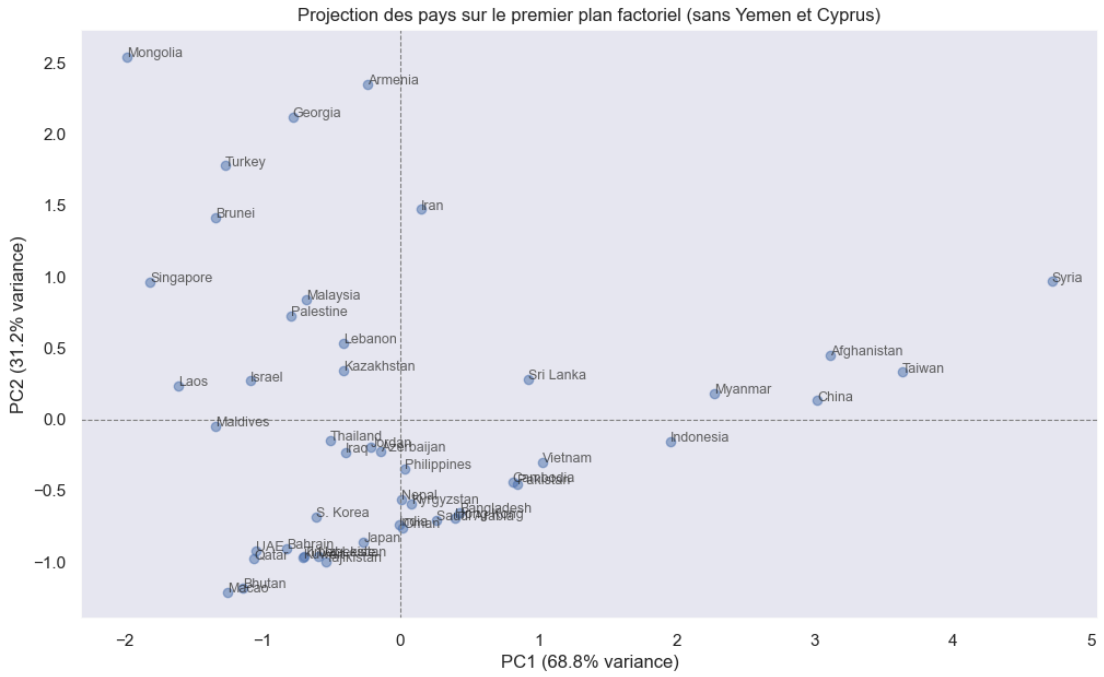
Pour mieux visualiser nos interprétations nous avons réalisé le cercle des corrélations.
Tout d'abord, le premier plan est fortement associé à la variable *TauxMortalite* qui est presque complètement alignée avec cet axe, ce qui signifie que le premier plan capte la majeure partie de la variabilité liée au taux de mortalité, c'est-à-dire la gravité de la crise sanitaire à travers la mortalité. De plus, le second plan factoriel, quant à lui, est presque exclusivement représenté par la variable *ActiveCasesPerMillion* qui explique la variabilité du

nombre de cas actifs par million d'habitants soit dynamique de l'épidémie en termes de propagation du virus.
Enfin, si l'on regarde le cercle dans son intégralité, on constate que les deux vecteurs forment un angle droit. Cela indique une indépendance statistique entre le nombre de cas actifs par million et le taux de mortalité. Ainsi, un pays ayant un grand nombre de cas en circulation ne présente pas obligatoirement une mortalité élevée et vice-versa.



Enfin, nous réalisons un graphique de projection des pays sur les deux premiers plans factoriels. Ce graphique nous permettra de rendre beaucoup plus visuel nos résultats.

Pour l'interprétation des résultats nous nous appuierons sur le cercle de corrélation précédant et principalement les sens des flèches des variables.
Au premier coup d'œil on observe directement deux pays : le Yémen et Chypre. Comme lors de notre analyse des taux, le Yémen se démarque par un nombre de décès très important par rapport au nombre de cas confirmés. Tant dis que Chypre se distingue par un grand nombre de cas actifs par million pour un taux de mortalité relativement bas. Du fait que ces deux pays écrasent la représentation des autres, nous avons décidé de relancer notre code en les retirant de notre base afin de rendre notre projection plus lisible.



Tout d'abord, on observe que les pays situés à droite, comme la Syrie, l'Afghanistan et Taiwan, ont un nombre de décès très important. Tant dis que les pays situés à gauche, comme Singapour, le Laos et les Maldives, ont un taux de mortalité faible.
Ensuite, bien que moins influent, le second plan factoriel met en évidence une variabilité liée au nombre de cas actifs par million d'habitants. Ainsi, les pays placés en haut du graphique, comme la Mongolie et l'Arménie, présentent un grand nombre de cas actifs par million. Cela peut indiquer une circulation élevée du virus. À l'inverse, les pays en bas du graphique, comme le Macao et le Qatar, ont peu de cas actifs.
Quant aux pays regroupés autour de l'origine, ils ont des situations relativement similaires en termes de mortalité et de circulation du virus, comme le Kazakhstan et le Liban.

Ainsi, on peut distinguer à droite les pays qui ont une mortalité plutôt élevée, indépendamment du nombre de cas actifs. En haut, plutôt ceux qui ont une épidémie encore très active, mais avec un impact sanitaire modéré en termes de mortalité. Enfin ceux situés près de l'origine, qui suggèrent une homogénéité dans la gestion de la crise.

Etant donné que la variable *TauxMortalite* a été créée à partir des variables *TotalCasesPerMillion* et *TotalDeathsPerMillion*, nous nous sommes demandés si prendre ces variables à la place du taux de mortalité ne serait pas plus judicieux. Nous avons donc effectué la même démarche qu'auparavant mais cette fois-ci avec les variables *TotalCasesPerMillion*, *TotalDeathsPerMillion* et *ActiveCasesPerMillion*.

La démarche est la même que la précédente, nous regarderons l'explicativité de chaque plan factoriel, la contribution de nos variables aux divers axes, le cercle de corrélation ainsi que la projection de chaque pays sur les plans factoriel.

Cependant malgré le lien entre les variables citées ci-dessus, nous n'obtenons pas les mêmes conclusions. De plus, les résultats obtenus à partir de cette seconde ACP ne coïncident pas avec ceux obtenus par analyse des taux.

Ainsi, l'ACP réalisé à partir du nombre de cas actifs par million et du taux de mortalité reste la plus pertinente.

Cependant, ces résultats sont à prendre avec précaution au vu de la qualité de certaines données de pays comme le Macao par exemple. De plus, ils ne nous permettent pas de réellement distinguer de "classes". Ces derniers permettent, tout comme l'analyse des taux, simplement de distinguer des pays ayant subi beaucoup de décès par rapport à leur nombre de cas actifs (Yémen, Syrie et Afghanistan), et d'autres ayant un grand nombre de cas actifs par million pour un taux de mortalité relativement bas (Mongolie).

Conclusion

Trois points importants ressortent de cette analyse.

D'abord, on observe un lien probable entre le nombre de tests et la gestion de la crise : les pays qui ont effectué davantage de tests ont souvent un taux de mortalité plus faible, ce qui suggère que détecter rapidement les cas permet de mieux contrôler l'épidémie.

Ensuite, les contextes économiques et politiques jouent un rôle majeur. Les pays en guerre ou en difficulté économique, comme le Yémen et l'Afghanistan, ont montré des indicateurs plus alarmants, ce qui souligne l'importance des ressources et des infrastructures dans la gestion d'une crise sanitaire et le besoin éventuel d'une aide extérieure pour ces deux pays.

Il faut malgré tout interpréter ces chiffres avec prudence. Un pays qui effectue beaucoup de tests peut le faire soit dans un but préventif, soit en raison d'une propagation déjà hors de contrôle.

Cette analyse a permis de mettre en avant l'importance d'une approche globale qui combine prévention, accès aux soins et mise en place de mesures adaptées pour faire face aux pandémies futures.

ANNEXE

DADDIO Melanie - FIX Emma - MARZOUK Moustafa - RAKOTOMANANA Zava

Identifier quels pays s'en sortent relativement bien et lesquels nécessitent une attention immédiate ?

1 Installation des packages nécessaires à l'étude

[ ]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

from scipy.stats import pearsonr

from sklearn.decomposition import PCA

from sklearn.preprocessing import StandardScaler
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

from skimpy import skim

# from pingouin import partial_corr
```

2 Import des données

[ ]:
```
#import
CovidCases = pd.read_csv("CovidCases.csv", sep=",")

CovidCases
```

3 Compréhension des données

[ ]:
```
skim(CovidCases)
```

[ ]:
```
# vérifier qu'il n'y ai pas de valeurs "bizarre"
CovidCases[CovidCases['TotalCases']<CovidCases["TotalDeaths"]]
```

[ ]:
```
CovidCases[CovidCases['TotalCasesPerMillion']<CovidCases["TotalDeathsPerMillion"]]
```

[ ]:
```
CovidCases[CovidCases['TotalPopulation']<CovidCases["TotalCases"]]
```

[ ]:
```
CovidCases[CovidCases['TotalPopulation']<CovidCases["TotalRecovered"]]
```

Il ne semble pas avoir des valeurs “bizarre”.

3.1 Visualisation des données :

[ ]:
```
continue_var = CovidCases.select_dtypes(include=['int64', 'float64']).columns.
 ↪tolist()
categorical_var = CovidCases.select_dtypes(include=['object']).columns.tolist()
```

3.1.1 Visualisation des variables continues :

[ ]:
```
# On affiche les histogrammes des variables continues

n_cols = 3  # Nombre de colonnes par ligne
n_rows = (len(continue_var) + n_cols - 1) // n_cols  # Calculer le nombre de␣
 ↪lignes nécessaires

plt.figure(figsize=(n_cols * 5, n_rows * 5))

for i, col in enumerate(continue_var[1:]):
    plt.subplot(n_rows, n_cols, i + 1)
    sns.histplot(CovidCases[col], color='skyblue', bins=20)
    plt.title(f'Histogramme de {col}')
    plt.xlabel(col)
    plt.ylabel('Fréquence')

plt.tight_layout()
plt.show()
```

[ ]:
```
# BOXPLOT
n_cols = 3
n_rows = (len(continue_var) + n_cols - 1) // n_cols

plt.figure(figsize=(n_cols * 5, n_rows * 5))
```

```
for i, col in enumerate(continue_var[1:]):
    plt.subplot(n_rows, n_cols, i + 1)
    sns.boxplot(y=CovidCases[col], palette='Set2')
    plt.title(f'Boxplot de {col}')
    plt.ylabel(col)

plt.tight_layout()

plt.show()
```

[ ]:
```
# Visualisationdes valaurs extremes heatmap
CC = CovidCases.iloc[:,2:]
CC_c = CC.sub(CC.mean())
CC_cr = CC_c.div(CC_c.std())
CC_cr = CC_cr.T
sns.set()
sns.heatmap(CC_cr,cmap="viridis")
plt.show()
```

3.1.2 Visualistion des variables continues en fonction des pays :

[ ]:
```
#barplot
#nombre total de cas en fonction du pays
plt.figure(figsize=(14, 7))
sns.barplot(x='TotalDeaths', y='Country', data=CovidCases.
 ↪sort_values('TotalDeaths', ascending=False))
plt.title('Total Death of COVID-19 in Asian Countries')
plt.xlabel('Total Death')
plt.ylabel('Country')
plt.show()

#nombre de cas par million d'individus en fonction du pays
plt.figure(figsize=(14, 7))
sns.barplot(x='TotalDeathsPerMillion', y='Country', data=CovidCases.
 ↪sort_values('TotalDeathsPerMillion', ascending=False))
plt.title('Total Death of COVID-19 in Asian Countries per million')
plt.xlabel('Total Death per million')
plt.ylabel('Country')
plt.show()
```

[ ]:
```
#nombre total de mort en fonction du pays
plt.figure(figsize=(14, 7))
sns.barplot(x='TotalDeaths', y='Country', data=CovidCases.
 ↪sort_values('TotalDeaths', ascending=False))
plt.title('Total deaths of COVID-19 in asian countries')
plt.xlabel('Total deaths')
```

```
plt.ylabel('Country')
plt.show()

#nombre de mort par million d'individus en fonction du pays
plt.figure(figsize=(14, 7))
sns.barplot(x='TotalDeathsPerMillion', y='Country', data=CovidCases.
 ↪sort_values('TotalDeathsPerMillion', ascending=False))
plt.title('Total deaths of COVID-19 in asian countries per million')
plt.xlabel('Total deaths per million')
plt.ylabel('Country')
plt.show()
```

[ ]:
```
#nombre total de test en fonction du pays
plt.figure(figsize=(14, 7))
sns.barplot(x='TotalTests', y='Country', data=CovidCases.
 ↪sort_values('TotalTests', ascending=False))
plt.title('Total tests of COVID-19 in asian countries')
plt.xlabel('Total tests')
plt.ylabel('Country')
plt.show()

#nombre de test par million d'individus en fonction du pays
plt.figure(figsize=(14, 7))
sns.barplot(x='TotalTestsPerMillion', y='Country', data=CovidCases.
 ↪sort_values('TotalTestsPerMillion', ascending=False))
plt.title('Total tests of COVID-19 in asian countries per million')
plt.xlabel('Total tests per million')
plt.ylabel('Country')
plt.show()
```

3.2 Analyse de relation entre variables :

[ ]:
```
#plot de x en fonction de y (pour préparer une éventuelle prédiction)
sns.pairplot(CovidCases.iloc[:,2:])
```

On a decidé de ne pas preter plus d'attention aux pairplot car nous ne remarquons pas un tendance particulière.

[ ]:
```
# matrice correlation
corr_matrix = CovidCases.iloc[:,2:].corr()
sns.heatmap(corr_matrix, annot=True, cmap="coolwarm", fmt=".2f", linewidths=0.5)
plt.title("Matrice de Corrélation des Variables COVID-19")
plt.show()
```

4 Données manquantes

En regardant sur le site on remarque que jusqu'à mi 2022 le Macao avait un total de mort déclaré à 0. Pour se pays nous imputerons donc les valeurs manquantes "totaldeath" et "totaldeathpermillion" à 0.

Pour le Tajikistan nous n'avons pas trouvé le nombre de tests dans le site nous allons donc emputer par la mediane. De plus la variable test ne nous a pas parue tres pertinente pour la suite de notre analyse et nous allons surment pas l'utiliser.

[ ]:
```
CovidCases.loc[CovidCases["Country"] == "Macao", :] = CovidCases.
 ↪loc[CovidCases["Country"] == "Macao", :].fillna(0)
```

[ ]:
```
print(CovidCases.isnull().sum())

#imputation par médiane
CovidCases.fillna(CovidCases.median(numeric_only=True), inplace=True)

print(CovidCases.isnull().sum())
```

5 Tests d'independance : test de corrélation de pearson

[ ]:
```
#TotalCases vs TotalDeaths
print(pearsonr(CovidCases.TotalCases, CovidCases.TotalDeaths))

#TotalCasesPerMillion vs TotalDeathsPerMillion
print(pearsonr(CovidCases.TotalCasesPerMillion, CovidCases.
 ↪TotalDeathsPerMillion))

#TotalCases vs TotalDeathsPerMillion
print(pearsonr(CovidCases.TotalCases, CovidCases.TotalDeathsPerMillion))

#TotalCases vs TotalCasesPerMillion
print(pearsonr(CovidCases.TotalCases, CovidCases.TotalCasesPerMillion))
```

[ ]:
```
# Corrélation partielle en contrôlant la population
# corr_p = partial_corr(data=CovidCases, x='TotalCases',␣
 ↪y='TotalCasesPerMillion', covar='TotalPopulation', method='pearson')
# print(corr_p)
```

6 Tests de redondance

[ ]:
```
# tester si une variables est une combinaison lineaire c'autres variables
q = CovidCases.select_dtypes(include=[np.number])
q = q.iloc[:,1:]
for col in q.columns:
```

```
    X = q.drop(columns=[col])
    y = q[col]
    model = LinearRegression()
    model.fit(X, y)
    r2 = model.score(X, y)

    if r2 > 0.95:
        print(f" {col} est très bien prédite par les autres variables (R² = {r2:
↪.3f}), elle est redondante")
```

On voit bien que les variables qui existent avec “PerMillion” et celles qui existent sans qui sont redondantes, ce qui est totalemnt logique.

7 Étude de taux

7.1 Calculs des taux et visualisation du classement des pays

[ ]:
```
# taux de mortalité
CovidCases['TauxMortalite'] = CovidCases['TotalDeathsPerMillion'] /␣
↪CovidCases['TotalCasesPerMillion'] * 100
plt.figure(figsize=(10,8))
sns.barplot(x='TauxMortalite', y='Country', data=CovidCases.
↪sort_values('TauxMortalite', ascending=False))
plt.ylabel('Pays')
plt.xlabel('Taux de mortalité')
plt.title("Taux de mortalité des pays d'Asie face au Covid-19")
plt.show()
```

[ ]:
```
# taux de récupération (on calcul le par million nous même)
CovidCases["TotalRecoveredPerMillion"] = (CovidCases["TotalRecovered"]␣
↪*1000000)/CovidCases["TotalPopulation"]
CovidCases["TauxRecuperation"] = CovidCases['TotalRecoveredPerMillion'] /␣
↪CovidCases['TotalCasesPerMillion'] * 100
plt.figure(figsize=(10,8))
sns.barplot(x='TauxRecuperation', y='Country', data=CovidCases.
↪sort_values('TauxRecuperation', ascending=False))
plt.ylabel('Pays')
plt.xlabel('Taux de récupération')
plt.title("Taux de récupération des pays d'Asie face au Covid-19")
plt.show()
```

[ ]:
```
# taux de cas actif
CovidCases["TauxCasActifs"] = CovidCases['ActiveCases'] /␣
↪CovidCases['TotalCases'] * 100
plt.figure(figsize=(10,8))
```

```
sns.barplot(x='TauxCasActifs', y='Country', data=CovidCases.
 ↪sort_values('TauxCasActifs', ascending=False))
plt.ylabel('Pays')
plt.xlabel('Taux de cas actifs')
plt.title("Taux de cas actifs dans les pays d'Asie (dernière données rentrées)")
plt.show()
```

7.2 Calculs des seuils et recupération des pays en difficulté

[ ]:
```
# Pays qui sont en dessous/au dessus d'un seuil
pays_cas = CovidCases[(CovidCases['TotalCasesPerMillion'] >␣
 ↪CovidCases['TotalCasesPerMillion'].quantile(0.9))]
pays_mortalite = CovidCases[(CovidCases['TauxMortalite'] >␣
 ↪CovidCases['TauxMortalite'].quantile(0.9))]
pays_test = CovidCases[(CovidCases['TotalTestsPerMillion'] <␣
 ↪CovidCases['TotalTestsPerMillion'].quantile(0.1) )]

print("Liste des pays au dessus du seuil de cas : \n\n", pays_cas[['Country',␣
 ↪'TotalCasesPerMillion', 'TotalTestsPerMillion', 'TauxMortalite']])
print("\nListe des pays au dessus du seuil de mortalité : \n\n",␣
 ↪pays_mortalite[['Country', 'TotalCasesPerMillion', 'TotalTestsPerMillion',␣
 ↪'TauxMortalite']])
print("\nListe des pays en dessous du seuil de tests : \n\n",␣
 ↪pays_test[['Country', 'TotalCasesPerMillion', 'TotalTestsPerMillion',␣
 ↪'TauxMortalite']])
```

[ ]:
```
# Pays au dessus/en dessous de plusieurs seuils
pays_concat = pd.concat([pays_cas['Country'], pays_mortalite['Country'],␣
 ↪pays_test['Country']])
pays_counts = pays_concat.value_counts()
pays_critique = pays_counts[pays_counts >= 2].index.tolist()

print("\nListe des pays qui dépassent plusieurs seuils : \n\n", CovidCases.
 ↪loc[CovidCases['Country'].isin(pays_critique), ['Country',␣
 ↪'TotalCasesPerMillion', 'TotalTestsPerMillion', 'TauxMortalite']])
```

8 PCA

8.1 TEST 1 PCA avec les variables : “ActiveCasesPerMillion”, “TauxMortalite”

[ ]:
```
# Préparation nous le PCA
CovidCases = pd.read_csv("CovidCases.csv", sep=",")
CovidCases.loc[CovidCases["Country"] == "Macao", :] = CovidCases.
 ↪loc[CovidCases["Country"] == "Macao", :].fillna(0)
CovidCases = CovidCases.drop(columns=["ID"])
```

```
CovidCases['ActiveCasesPerMillion'] = CovidCases['ActiveCases'] * 1000000 /␣
 ↪CovidCases['TotalPopulation']
colums_stand = CovidCases.columns[1:]
print(colums_stand)

# Calcul taux mortalité
CovidCases['TauxMortalite'] = CovidCases['TotalDeathsPerMillion'] /␣
 ↪CovidCases['TotalCasesPerMillion'] * 100
```

- Standardiser les données

[ ]:
```
scaler = StandardScaler()
CovidCases[colums_stand] = scaler.fit_transform(CovidCases[colums_stand])
CovidCases.head()
```

- Selection des variables

[ ]:
```
CovidCases_without_country = CovidCases[["ActiveCasesPerMillion",␣
 ↪"TauxMortalite"]]
CovidCases_without_country.head()
```

[ ]:
```
X = CovidCases_without_country.iloc[:, :].values
```

[ ]:
```
X
```

- PCA :

[ ]:
```
pca = PCA()
X_pca=pca.fit_transform(X)
```

[ ]:
```
# Inertie expliquée
explained_variance = pca.explained_variance_ratio_ * 100
explained_variance
```

Choix du nombre d'axes :

[ ]:
```
# Analyse des valeurs propres
n_components = len(pca.explained_variance_)
comp = pd.DataFrame(
    {
        "Dimension" : ["Dim" + str(x + 1) for x in range(n_components)],
        "Valeur propre" : pca.explained_variance_,
        "% variance expliquée" : np.round(pca.explained_variance_ratio_ * 100),
        "% cum. var. expliquée" : np.round(np.cumsum(pca.
 ↪explained_variance_ratio_) * 100)
    },
    columns = ["Dimension", "Valeur propre", "% variance expliquée", "% cum.␣
 ↪var. expliquée"]
```

```
)
comp
```

[ ]:
```
# Scree plot pour choisir le nombre de composantes principales
g_comp = sns.barplot(x = "Dimension",
                     y = "% variance expliquée",
                     palette = ["lightseagreen"],
                     data = comp)
g_comp.set(ylabel = "Variance expliquée (%)")
g_comp.figure.suptitle("Variance expliquée par dimension")
plt.axhline(y = 25, linewidth = .5, color = "dimgray", linestyle = "--") # 25 =␣
 ↪100 / 4 (nb dimensions)
plt.text(3.25, 26, "25%")
```

[ ]:
```
# Calcul du cosinus carré des variables
cos_squared = np.square(pca.components_)
df_cos_squared = pd.DataFrame(cos_squared, columns=['PC{}'.format(i+1) for i in␣
 ↪range(n_components)])
df_cos_squared.index = CovidCases_without_country.columns

print(df_cos_squared)
```

[ ]:
```
# Contribution à la formation de l'axe
components = pca.components_

n_components = X.shape[1]
feature_names=CovidCases_without_country.columns

loadings = pca.components_.T
eigenvalues = pca.explained_variance_
variable_contributions = (loadings**2) * eigenvalues

column_names = [f'PC{i+1}_contrib' for i in range(n_components)]
variable_contrib_df = pd.DataFrame(variable_contributions,␣
 ↪columns=column_names, index=feature_names)

variable_contrib_df
```

[ ]:
```
# Calculer la contribution des individus à la formation des axes
eigenvalues = pca.explained_variance_
contributions = (X_pca**2) / (X_pca.shape[0] * eigenvalues)
contrib_percent = contributions * 100

column_names = [f'PC{i+1}_contrib' for i in range(n_components)]
contrib_df = pd.DataFrame(contrib_percent, columns=column_names)

print(contrib_df)
```

[ ]:
```
# Créer le cercle de corrélation
coeff = np.transpose(pca.components_[0:2, :])
n = coeff.shape[0]
xs = np.array([1, 0])
ys = np.array([0, 1])

plt.figure(figsize=(10, 10))
for i in range(n):
    plt.arrow(0, 0, coeff[i, 0], coeff[i, 1], color='k', alpha=0.9,␣
 ↪head_width=0.02)
    plt.text(coeff[i, 0] * 1.15, coeff[i, 1] * 1.15, feature_names[i],␣
 ↪color='k', ha='center', va='center')

circle = plt.Circle((0, 0), 1, color='gray', fill=False, linestyle='--')
plt.gca().add_artist(circle)
plt.xlim(-1, 1)
plt.ylim(-1, 1)
plt.axhline(0, color='gray', linewidth=1)
plt.axvline(0, color='gray', linewidth=1)
plt.xlabel('PC1')
plt.ylabel('PC2')
plt.title('Cercle de corrélation')
plt.show()
```

[ ]:
```
# Projection des pays
plt.figure(figsize=(12, 7))
plt.scatter(X_pca[:, 0], X_pca[:, 1], alpha=0.5)

for i, country in enumerate(CovidCases.iloc[:, 0]):
    plt.text(X_pca[i, 0], X_pca[i, 1], country, fontsize=9, alpha=0.7)

plt.xlabel("PC1 ({}% variance)".format(round(explained_variance[0], 1)))
plt.ylabel("PC2 ({}% variance)".format(round(explained_variance[1], 1)))
plt.title("Projection des pays sur le premier plan factoriel")
plt.axhline(0, color="grey", linestyle="--", linewidth=0.8)
plt.axvline(0, color="grey", linestyle="--", linewidth=0.8)
plt.grid()
plt.show()
```

[ ]:
```

```

8.2 TEST 1 bis PCA avec les variables : "ActiveCasesPerMillion", "TauxMortalite" (sans le Yemen et Cyprus)

[ ]:
```
# Préparation nous le PCA
CovidCases = pd.read_csv("CovidCases.csv", sep=",")
CovidCases.loc[CovidCases["Country"] == "Macao", :] = CovidCases.
 ↪loc[CovidCases["Country"] == "Macao", :].fillna(0)
CovidCases = CovidCases.drop(columns=["ID"])
CovidCases['ActiveCasesPerMillion'] = CovidCases['ActiveCases'] * 1000000 /␣
 ↪CovidCases['TotalPopulation']
colums_stand = CovidCases.columns[1:]
print(colums_stand)

# Calcul taux mortalité
CovidCases['TauxMortalite'] = CovidCases['TotalDeathsPerMillion'] /␣
 ↪CovidCases['TotalCasesPerMillion'] * 100

# Supprimer Yemen et cyprus
CovidCases = CovidCases[CovidCases['Country'] != 'Yemen']
CovidCases = CovidCases[CovidCases['Country'] != 'Cyprus']
print(CovidCases.shape)
```

- Standardiser les données

[ ]:
```
scaler = StandardScaler()
CovidCases[colums_stand] = scaler.fit_transform(CovidCases[colums_stand])
CovidCases.head()
```

- Selection des variables

[ ]:
```
CovidCases_without_country = CovidCases[["ActiveCasesPerMillion",␣
 ↪"TauxMortalite"]]
CovidCases_without_country.head()
```

[ ]:
```
X = CovidCases_without_country.iloc[:, :].values
```

[ ]:
```
X
```

- PCA :

[ ]:
```
pca = PCA()
X_pca=pca.fit_transform(X)
```

[ ]:
```
# Inertie expliquée
explained_variance = pca.explained_variance_ratio_ * 100
explained_variance
```

Choix du nombre d'axes :

[ ]:
```
# Analyse des valeurs propres
n_components = len(pca.explained_variance_)
comp = pd.DataFrame(
    {
        "Dimension" : ["Dim" + str(x + 1) for x in range(n_components)],
        "Valeur propre" : pca.explained_variance_,
        "% variance expliquée" : np.round(pca.explained_variance_ratio_ * 100),
        "% cum. var. expliquée" : np.round(np.cumsum(pca.
 ↪explained_variance_ratio_) * 100)
    },
    columns = ["Dimension", "Valeur propre", "% variance expliquée", "% cum.␣
 ↪var. expliquée"]
)
comp
```

[ ]:
```
# Scree plot pour choisir le nombre de composantes principales
g_comp = sns.barplot(x = "Dimension",
                y = "% variance expliquée",
                palette = ["lightseagreen"],
                data = comp)
g_comp.set(ylabel = "Variance expliquée (%)")
g_comp.figure.suptitle("Variance expliquée par dimension")
plt.axhline(y = 25, linewidth = .5, color = "dimgray", linestyle = "--") # 25 =␣
 ↪100 / 4 (nb dimensions)
plt.text(3.25, 26, "25%")
```

[ ]:
```
# Calcul du cosinus carré des variables
cos_squared = np.square(pca.components_)
df_cos_squared = pd.DataFrame(cos_squared, columns=['PC{}'.format(i+1) for i in␣
 ↪range(n_components)])
df_cos_squared.index = CovidCases_without_country.columns

print(df_cos_squared)
```

[ ]:
```
# Contribution à la formation de l'axe
components = pca.components_

n_components = X.shape[1]
feature_names=CovidCases_without_country.columns

loadings = pca.components_.T
eigenvalues = pca.explained_variance_
variable_contributions = (loadings**2) * eigenvalues

column_names = [f'PC{i+1}_contrib' for i in range(n_components)]
variable_contrib_df = pd.DataFrame(variable_contributions,␣
 ↪columns=column_names, index=feature_names)
```

```
variable_contrib_df
```

[ ]:
```
# Calculer la contribution des individus à la formation des axes
eigenvalues = pca.explained_variance_
contributions = (X_pca**2) / (X_pca.shape[0] * eigenvalues)
contrib_percent = contributions * 100

column_names = [f'PC{i+1}_contrib' for i in range(n_components)]
contrib_df = pd.DataFrame(contrib_percent, columns=column_names)

print(contrib_df)
```

[ ]:
```
# Créer le cercle de corrélation
coeff = np.transpose(pca.components_[0:2, :])
n = coeff.shape[0]
xs = np.array([1, 0])
ys = np.array([0, 1])

plt.figure(figsize=(10, 10))
for i in range(n):
    plt.arrow(0, 0, coeff[i, 0], coeff[i, 1], color='k', alpha=0.9,␣
 ↪head_width=0.02)
    plt.text(coeff[i, 0] * 1.15, coeff[i, 1] * 1.15, feature_names[i],␣
 ↪color='k', ha='center', va='center')

circle = plt.Circle((0, 0), 1, color='gray', fill=False, linestyle='--')
plt.gca().add_artist(circle)
plt.xlim(-1, 1)
plt.ylim(-1, 1)
plt.axhline(0, color='gray', linewidth=1)
plt.axvline(0, color='gray', linewidth=1)
plt.xlabel('PC1')
plt.ylabel('PC2')
plt.title('Cercle de corrélation')
plt.show()
```

[ ]:
```
# Projection des pays
plt.figure(figsize=(12, 7))
plt.scatter(X_pca[:, 0], X_pca[:, 1], alpha=0.5)

for i, country in enumerate(CovidCases.iloc[:, 0]):
    plt.text(X_pca[i, 0], X_pca[i, 1], country, fontsize=9, alpha=0.7)

plt.xlabel("PC1 ({}% variance)".format(round(explained_variance[0], 1)))
plt.ylabel("PC2 ({}% variance)".format(round(explained_variance[1], 1)))
```

```
plt.title("Projection des pays sur le premier plan factoriel (sans Yemen et␣
 ↪Cyprus)")
plt.axhline(0, color="grey", linestyle="--", linewidth=0.8)
plt.axvline(0, color="grey", linestyle="--", linewidth=0.8)
plt.grid()
plt.show()
```

8.3 TEST 2 PCA avec les variables : "TotalCasesPerMillion", "TotalDeathsPerMillion", "ActiveCasesPerMillion"

[ ]:
```
# Préparation nous le PCA
CovidCases = pd.read_csv("CovidCases.csv", sep=",")
CovidCases.loc[CovidCases["Country"] == "Macao", :] = CovidCases.
 ↪loc[CovidCases["Country"] == "Macao", :].fillna(0)
CovidCases = CovidCases.drop(columns=["ID"])
CovidCases['ActiveCasesPerMillion'] = CovidCases['ActiveCases'] * 1000000 /␣
 ↪CovidCases['TotalPopulation']
colums_stand = CovidCases.columns[1:]
print(colums_stand)
```

- Standardiser les données

[ ]:
```
scaler = StandardScaler()
CovidCases[colums_stand] = scaler.fit_transform(CovidCases[colums_stand])
CovidCases.head()
```

- Selection des variables

[ ]:
```
CovidCases_without_country = CovidCases[["TotalCasesPerMillion",␣
 ↪"TotalDeathsPerMillion", "ActiveCasesPerMillion"]]
CovidCases_without_country.head()
```

[ ]:
```
X = CovidCases_without_country.iloc[:, :].values
```

[ ]:
```
X
```

- PCA :

[ ]:
```
pca = PCA()
X_pca=pca.fit_transform(X)
```

[ ]:
```
# Inertie expliquée
explained_variance = pca.explained_variance_ratio_ * 100
explained_variance
```

Choix du nombre d'axes :

[ ]:
```
# Analyse des valeurs propres
n_components = len(pca.explained_variance_)
comp = pd.DataFrame(
    {
        "Dimension" : ["Dim" + str(x + 1) for x in range(n_components)],
        "Valeur propre" : pca.explained_variance_,
        "% variance expliquée" : np.round(pca.explained_variance_ratio_ * 100),
        "% cum. var. expliquée" : np.round(np.cumsum(pca.
 ↪explained_variance_ratio_) * 100)
    },
    columns = ["Dimension", "Valeur propre", "% variance expliquée", "% cum.␣
 ↪var. expliquée"]
)
comp
```

[ ]:
```
# Scree plot pour choisir le nombre de composantes principales
g_comp = sns.barplot(x = "Dimension",
                y = "% variance expliquée",
                palette = ["lightseagreen"],
                data = comp)
g_comp.set(ylabel = "Variance expliquée (%)")
g_comp.figure.suptitle("Variance expliquée par dimension")
plt.axhline(y = 25, linewidth = .5, color = "dimgray", linestyle = "--") # 25 =␣
 ↪100 / 4 (nb dimensions)
plt.text(3.25, 26, "25%")
```

[ ]:
```
# Calcul du cosinus carré des variables
cos_squared = np.square(pca.components_)
df_cos_squared = pd.DataFrame(cos_squared, columns=['PC{}'.format(i+1) for i in␣
 ↪range(n_components)])
df_cos_squared.index = CovidCases_without_country.columns

print(df_cos_squared)
```

[ ]:
```
# Contribution à la formation de l'axe
components = pca.components_

n_components = X.shape[1]
feature_names=CovidCases_without_country.columns

loadings = pca.components_.T
eigenvalues = pca.explained_variance_
variable_contributions = (loadings**2) * eigenvalues

column_names = [f'PC{i+1}_contrib' for i in range(n_components)]
variable_contrib_df = pd.DataFrame(variable_contributions,␣
 ↪columns=column_names, index=feature_names)
```

```
variable_contrib_df
```

[ ]:
```
# Calculer la contribution des individus à la formation des axes
eigenvalues = pca.explained_variance_
contributions = (X_pca**2) / (X_pca.shape[0] * eigenvalues)
contrib_percent = contributions * 100

column_names = [f'PC{i+1}_contrib' for i in range(n_components)]
contrib_df = pd.DataFrame(contrib_percent, columns=column_names)

print(contrib_df)
```

[ ]:
```
# Créer le cercle de corrélation
coeff = np.transpose(pca.components_[0:2, :])
n = coeff.shape[0]
xs = np.array([1, 0])
ys = np.array([0, 1])

plt.figure(figsize=(10, 10))
for i in range(n):
    plt.arrow(0, 0, coeff[i, 0], coeff[i, 1], color='k', alpha=0.9,␣
 ↪head_width=0.02)
    plt.text(coeff[i, 0] * 1.15, coeff[i, 1] * 1.15, feature_names[i],␣
 ↪color='k', ha='center', va='center')

circle = plt.Circle((0, 0), 1, color='gray', fill=False, linestyle='--')
plt.gca().add_artist(circle)
plt.xlim(-1, 1)
plt.ylim(-1, 1)
plt.axhline(0, color='gray', linewidth=1)
plt.axvline(0, color='gray', linewidth=1)
plt.xlabel('PC1')
plt.ylabel('PC2')
plt.title('Cercle de corrélation')
plt.show()
```

[ ]:
```
# Projection des pays
plt.figure(figsize=(12, 7))
plt.scatter(X_pca[:, 0], X_pca[:, 1], alpha=0.5)

for i, country in enumerate(CovidCases.iloc[:, 0]):
    plt.text(X_pca[i, 0], X_pca[i, 1], country, fontsize=9, alpha=0.7)

plt.xlabel("PC1 ({}% variance)".format(round(explained_variance[0], 1)))
plt.ylabel("PC2 ({}% variance)".format(round(explained_variance[1], 1)))
plt.title("Projection des pays sur le premier plan factoriel")
```

```
plt.axhline(0, color="grey", linestyle="--", linewidth=0.8)
plt.axvline(0, color="grey", linestyle="--", linewidth=0.8)
plt.grid()
plt.show()
```