

# Rapport Time Series and Machine Learning



RODRIGUES Julien  
CHAUVIN Louis

2023-2024

Dans le cadre de l'unité Time Series et Machine Learning, nous avons à réaliser un projet final mettant en application les notions vues en cours. Afin de mener ce projet nous nous sommes posés la question suivante : **Pouvons-nous prédire la courbe des anomalies de température globale en fonction de différents facteurs?**

L'interrogation sur la possibilité de prédire la courbe des anomalies de température globale en fonction de divers facteurs revêt une importance cruciale à l'heure actuelle. Avec le changement climatique devenant une préoccupation mondiale, comprendre les tendances climatiques et anticiper les variations de température devient impératif. Cette question nous invite donc à explorer les relations complexes entre les émissions de gaz à effet de serre, les phénomènes naturels, et les activités humaines, afin de mieux appréhender l'évolution du climat. Une réponse à cette interrogation pourrait non seulement renforcer nos capacités de prédiction climatique, mais pourrait également apporter des éclaircissements sur les décisions politiques et les actions visant à atténuer les impacts du changement climatique à l'échelle mondiale.

Nous avons dû définir des facteurs possibles pour notre projet. Il existe des facteurs évidents et connus de tous mais nous avons aussi choisi de définir des facteurs qui n'ont potentiellement pas de corrélation observée avec les anomalies de température.

Nous avons donc les facteurs suivants :

- facteur démographique avec la population mondiale, les répartitions rurales et urbaines
- évolution des anomalies de températures : globale, terrestre, maritime
- utilisation des terres et émissions de GHG dans l'agriculture
- évolution d'internet dans le monde
- émissions des GHG par les différents secteurs
- production énergétique par pays
- etc...
- 

Toutes les données sont des données open data directement récupérées sur les sites Food and Agriculture Organization of the United Nations, Kaggle, Organisation Mondiale de la Santé, The World Bank.

Notre but est d'extraire les données des différents facteurs pour créer notre dataframe et faire nos modèles dessus.

Avec ce rapport nous allons vous présenter l'approche scientifique que nous avons eu pour répondre au mieux à problématique. Nous allons détailler les cheminements de pensée ainsi que les résultats et leurs véracités.

# 1. Récupération des datasets

## A. Emissions GHG (CO2,CH4,N2O,Fgases,GHG)

Nous avons récupéré le premier jeu de données, il s'agit des fichiers csv du type d'émission en fonction des secteurs.

Fichiers csv :

- "CO2-emissions-by-sector.csv"
- "CH4-emissions-by-sector.csv"
- "N2O-emissions-by-sector.csv"
- "Fgases-emissions-by-sector.csv"
- "GHG-emissions-by-sector.csv"

Voici une présentation et description des différents secteurs des fichiers csv :

-Agriculture : Les émissions associées aux activités agricoles. Cela peut inclure les émissions provenant de la gestion des sols, de l'utilisation d'engrais, de la fermentation entérique chez les animaux d'élevage, etc.

-Buildings : Les émissions provenant du secteur des bâtiments, généralement liées à la consommation d'énergie pour le chauffage, la climatisation, l'éclairage et d'autres besoins énergétiques dans les bâtiments résidentiels et commerciaux.

-Fuel Exploitation : Les émissions résultant de l'exploitation des combustibles fossiles, y compris l'extraction, la production et le traitement du pétrole, du gaz et du charbon.

-Industrial Combustion : Les émissions résultant de la combustion de combustibles fossiles dans le secteur industriel pour la production de chaleur et d'énergie nécessaire aux processus industriels.

-Power Industry : Les émissions de CO2 provenant du secteur de la production d'électricité, générées par la combustion de combustibles fossiles dans les centrales.

-Processes: Les émissions provenant de divers processus industriels autres que la combustion, tels que la fabrication de ciment, la production chimique, etc. Ce sont des exemples, il en existe plein d'autres.

-Transport: Les émissions générées par le secteur des transports, comprenant les émissions provenant des véhicules routiers, du transport maritime, aérien et ferroviaire, ainsi que d'autres moyens de transport.

-Waste : Les émissions provenant de la gestion des déchets, y compris la décomposition des déchets organiques dans les décharges et les émissions associées à d'autres processus de traitement des déchets. Total /cap : Les émissions totales par habitant, une mesure qui prend en compte l'ensemble des émissions d'un pays ou d'une région divisées par la population.

Nous avons utilisé la fonction `.rename()` de pandas pour renommer la colonne 'Category' en 'Year' de chaque dataframe. Pour merger les dataframe en fonction de la colonne 'Year', nous allons appliquer un suffixe à chaque colonne des dataframe en fonction du type de gaz à effet de serre émis.

#### *Overview de la dataframe pour les émissions de CO2*

| Year | Agriculture-CO2 | Buildings-CO2 | Fuel Exploitation-CO2 | Industrial Combustion-CO2 | Power Industry-CO2 | Processes-CO2 | Transport-CO2 | Waste-CO2  | Total CO2/cap-CO2 |      |
|------|-----------------|---------------|-----------------------|---------------------------|--------------------|---------------|---------------|------------|-------------------|------|
| 0    | 1970            | 49143283      | 2926458212            | 1562255701                | 3744304794         | 3823699383    | 915670265     | 2796286627 | 7605313           | 4.28 |
| 1    | 1971            | 49143283      | 2939568254            | 1574399637                | 3511720997         | 3910981426    | 921879024     | 2876504749 | 7779282           | 4.18 |
| 2    | 1972            | 49143283      | 3056870832            | 1654467714                | 3602041957         | 4189105946    | 990161466     | 3045881595 | 7957965           | 4.31 |
| 3    | 1973            | 49143283      | 3120449101            | 1828873474                | 3788025165         | 4524711259    | 1030394452    | 3221973378 | 8139190           | 4.47 |
| 4    | 1974            | 49900345      | 3040634678            | 1824605510                | 3774962595         | 4603893544    | 1007746632    | 3191503489 | 8320052           | 4.37 |

Nous avons finalement mergé les dataframes en fonction de la colonne 'Year'.

## B. Températures globales, terrestres et maritimes

Nous avons récupéré les valeurs des anomalies de température globales, terrestres et maritimes.

Fichiers csv :

- "data\_land\_temperatures.csv"
- "data\_ocean\_temperatures.csv"
- "global\_land\_ocean\_temperatures.csv"

Les données ont été récupérées sur le site National Centers for Environmental Information :

[https://www.ncei.noaa.gov/access/monitoring/climate-at-a-glance/global/time-series/globe/land\\_ocean/all/8/1850-2023](https://www.ncei.noaa.gov/access/monitoring/climate-at-a-glance/global/time-series/globe/land_ocean/all/8/1850-2023). On remarque que les données de températures globales ne correspondent pas à la moyenne des températures terrestres et maritimes par année. Il semblerait qu'il y ait des facteurs qui entrent en jeu dans le calcul de cette valeur.. Pour éviter toute incohérence nous avons choisi d'utiliser les données récupérées.

### 1. Températures terrestres

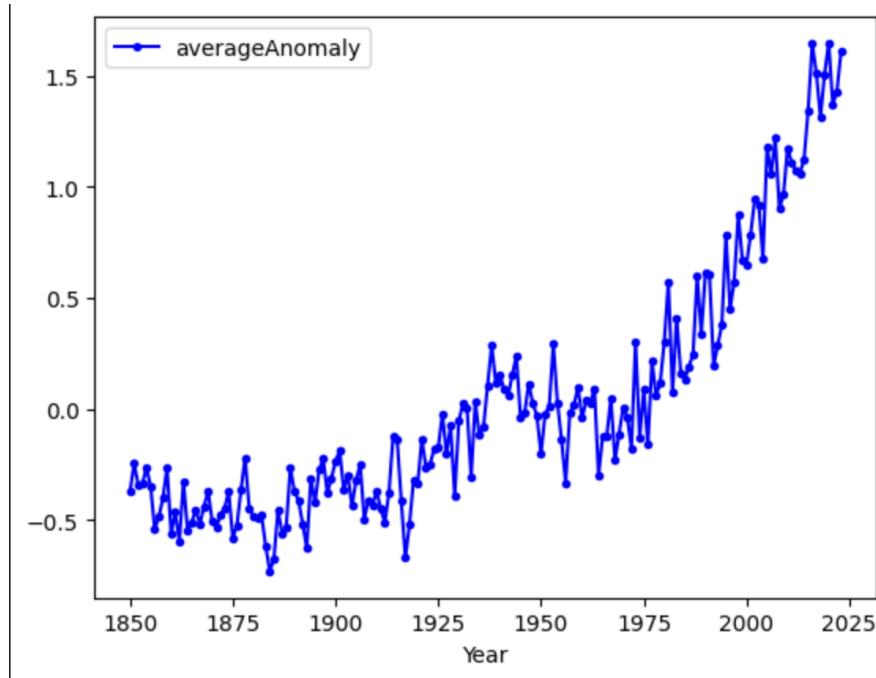
Il s'agit des données des anomalies de températures mesurées au sol dans le monde.

Affichage du dataframe :

|                       | Year   | Anomaly |                      | Year | averageAnomaly |          |
|-----------------------|--------|---------|----------------------|------|----------------|----------|
| 0                     | 185001 | -0.95   |                      | 0    | -0.373333      |          |
| 1                     | 185002 | -0.25   |                      | 1    | -0.246667      |          |
| 2                     | 185003 | -0.47   |                      | 2    | -0.343333      |          |
| 3                     | 185004 | -0.43   |                      | 3    | -0.336667      |          |
| 4                     | 185005 | -0.45   |                      | 4    | -0.265833      |          |
| ...                   | ...    | ...     |                      | ...  | ...            |          |
| 2080                  | 202305 | 1.19    |                      | 169  | 2019           | 1.508333 |
| 2081                  | 202306 | 1.37    |                      | 170  | 2020           | 1.647500 |
| 2082                  | 202307 | 1.41    |                      | 171  | 2021           | 1.371667 |
| 2083                  | 202308 | 1.66    |                      | 172  | 2022           | 1.430000 |
| 2084                  | 202309 | 2.34    |                      | 173  | 2023           | 1.608889 |
| 2085 rows x 2 columns |        |         | 174 rows x 2 columns |      |                |          |

Nous sommes passés du premier datafram au second. Pour ce faire, nous avons dû déconstruire la colonne ‘Year’ en 2 colonnes ‘Year’ et ‘Month’, puis nous avons appliqué un groupby() de la librairie python pour faire la moyenne des valeurs mensuelles pour l’année.

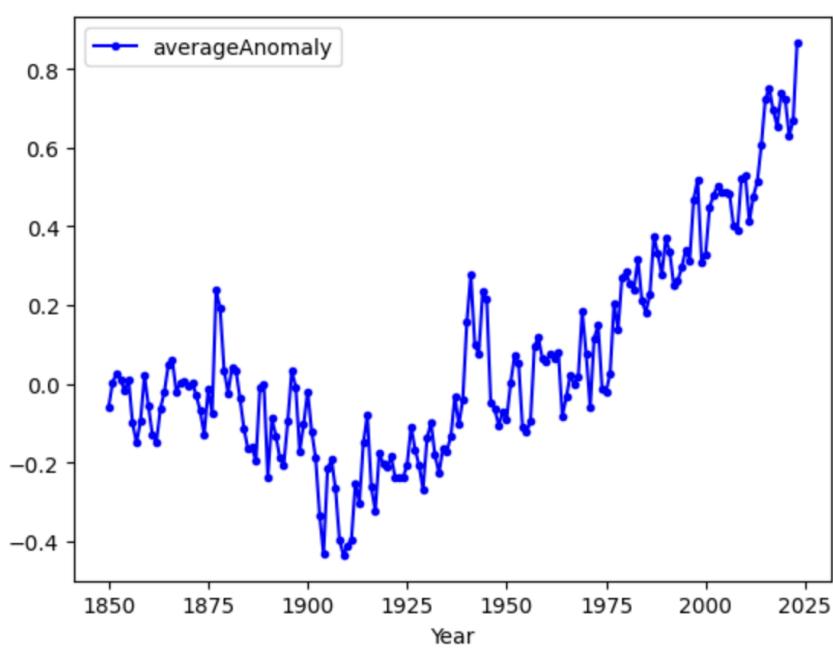
*Plot des anomalies de températures terrestres :*



## 2. Températures maritimes.

Ce sont les températures mesurées au niveau des océans. Nous avons appliqué la même méthode que pour les températures terrestres. Nous obtenons le graphe suivant.

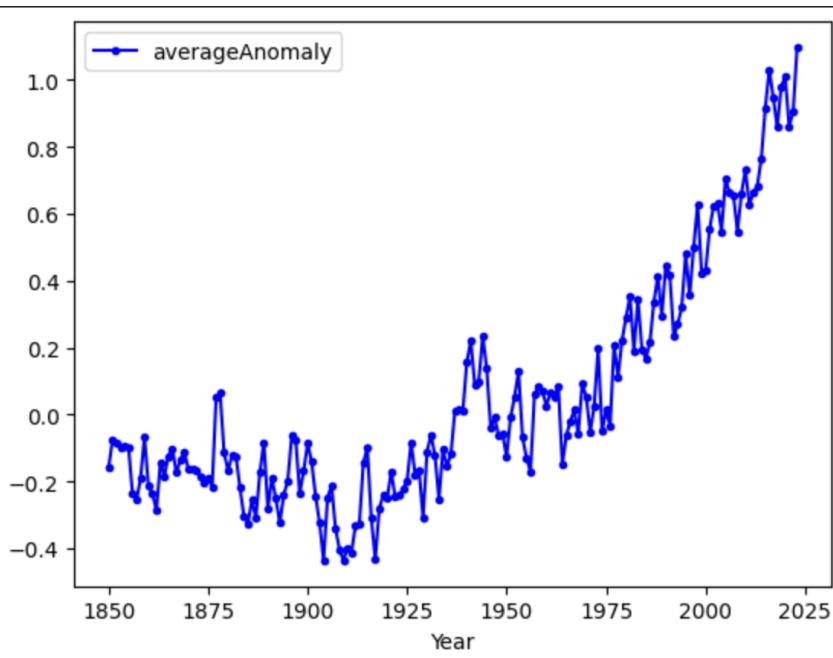
*Plot des anomalies de températures maritimes :*



### 3. Températures globales

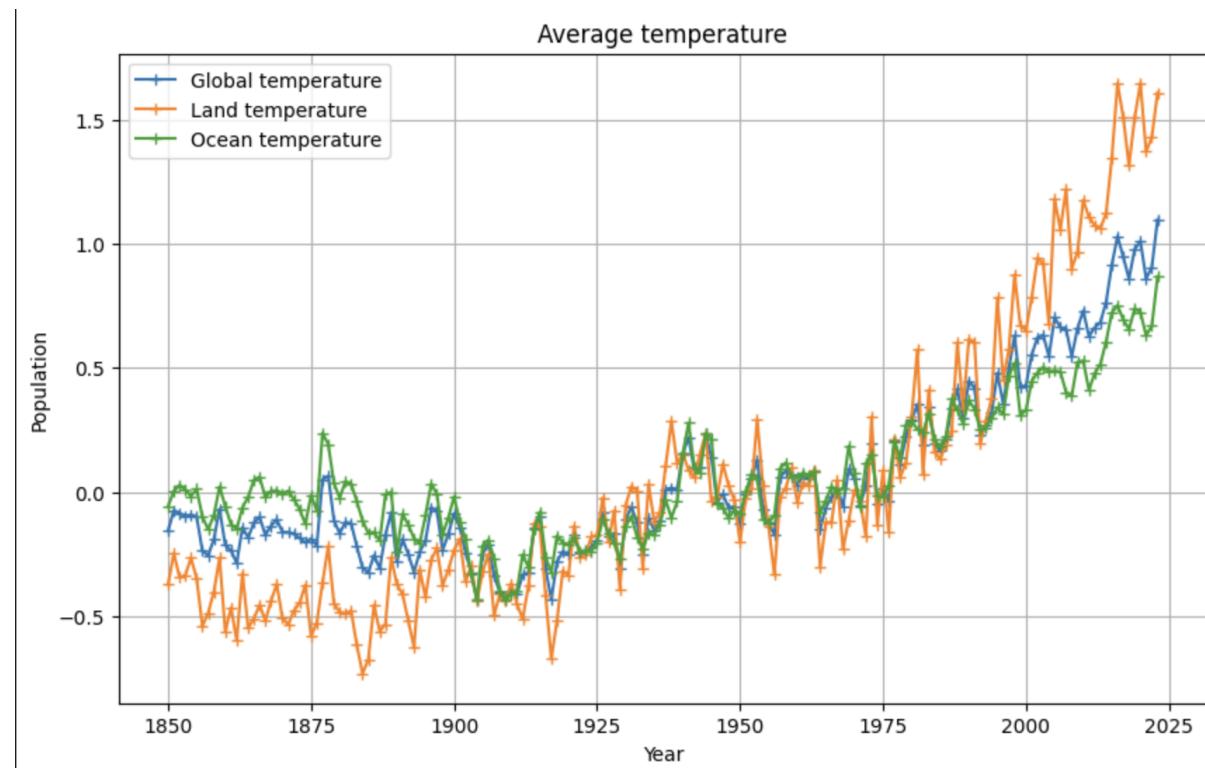
Nous avons de nouveau appliqué la même méthode.

*Plot des anomalies de températures globales :*



**Nous avons regroupé toutes les dataframes en une seule en les mergeant suivant la colonne des années**

*Plot des anomalies de températures globales, terrestres et maritimes :*



### C. Populations mondiales, répartitions rurales/urbaines mondiales

Il s'agit du facteur démographique de notre projet. Nous pouvons récupérer les répartitions rurales et urbaines pour chaque pays entre 1950 et 2021. Nous avons récupéré les données sur le site FAO(Food and Agriculture Organization) des Nations Unies : <https://www.fao.org/faostat/en/#data/OA>.

A partir des données pour chaque pays, nous avons appliqué des méthodes pour créer des valeurs mondiales et pouvoir mieux les utiliser dans nos futurs modèles.

Fichier csv :

- "FAOSTAT\_data\_en\_10-21-2023.csv"

**Si on observe les données en détail surtout pour la plage d'années 1950-1960 et 2010-2021, on peut remarquer des anomalies. Les données sont beaucoup plus grandes que la réalité. Par exemple, pour l'année 2021 il y a +9 milliards d'habitants. Avec des recherches nous avons trouvé pourquoi. Le site disposait des données de 1960 à 2010. Ils ont fait un modèle de prédiction et ont prédit les données pour la plage d'années 1950-1960 et 2010-2021. Lors de la conception de nos modèles, il faudra prendre en compte ces incertitudes dues au modèle utilisé par le site.**

## Dataframe des populations mondiales, répartitions rurales/urbaines mondiales

|   | Domain Code | Domain            | Area Code (M49) | Area        | Element Code | Element                       | Item Code | Item                      | Year Code | Year | Unit    | Value    | Flag | Flag Description                        | Note |
|---|-------------|-------------------|-----------------|-------------|--------------|-------------------------------|-----------|---------------------------|-----------|------|---------|----------|------|---|------|
| 0 | OA          | Annual population | 4               | Afghanistan | 511          | Total Population - Both sexes | 3010      | Population - Est. & Proj. | 1950      | 1950 | 1000 No | 7480.461 | X    | Figure from international organizations | NaN  |
| 1 | OA          | Annual population | 4               | Afghanistan | 551          | Rural population              | 3010      | Population - Est. & Proj. | 1950      | 1950 | 1000 No | 7286.991 | X    | Figure from international organizations | NaN  |
| 2 | OA          | Annual population | 4               | Afghanistan | 561          | Urban population              | 3010      | Population - Est. & Proj. | 1950      | 1950 | 1000 No | 465.127  | X    | Figure from international organizations | NaN  |
| 3 | OA          | Annual population | 4               | Afghanistan | 511          | Total Population - Both sexes | 3010      | Population - Est. & Proj. | 1951      | 1951 | 1000 No | 7571.537 | X    | Figure from international organizations | NaN  |
| 4 | OA          | Annual population | 4               | Afghanistan | 551          | Rural population              | 3010      | Population - Est. & Proj. | 1951      | 1951 | 1000 No | 7352.856 | X    | Figure from international organizations | NaN  |

Nous avons dropé les colonnes qui étaient inutiles.

|   | Area        | Element                       | Year | Value    |
|---|-------------|-------------------------------|------|----------|
| 0 | Afghanistan | Total Population - Both sexes | 1950 | 7480.461 |
| 1 | Afghanistan | Rural population              | 1950 | 7286.991 |
| 2 | Afghanistan | Urban population              | 1950 | 465.127  |
| 3 | Afghanistan | Total Population - Both sexes | 1951 | 7571.537 |
| 4 | Afghanistan | Rural population              | 1951 | 7352.856 |

Le format tel quel de notre dataframe n'est pas adapté pour l'utilisation dans un modèle.

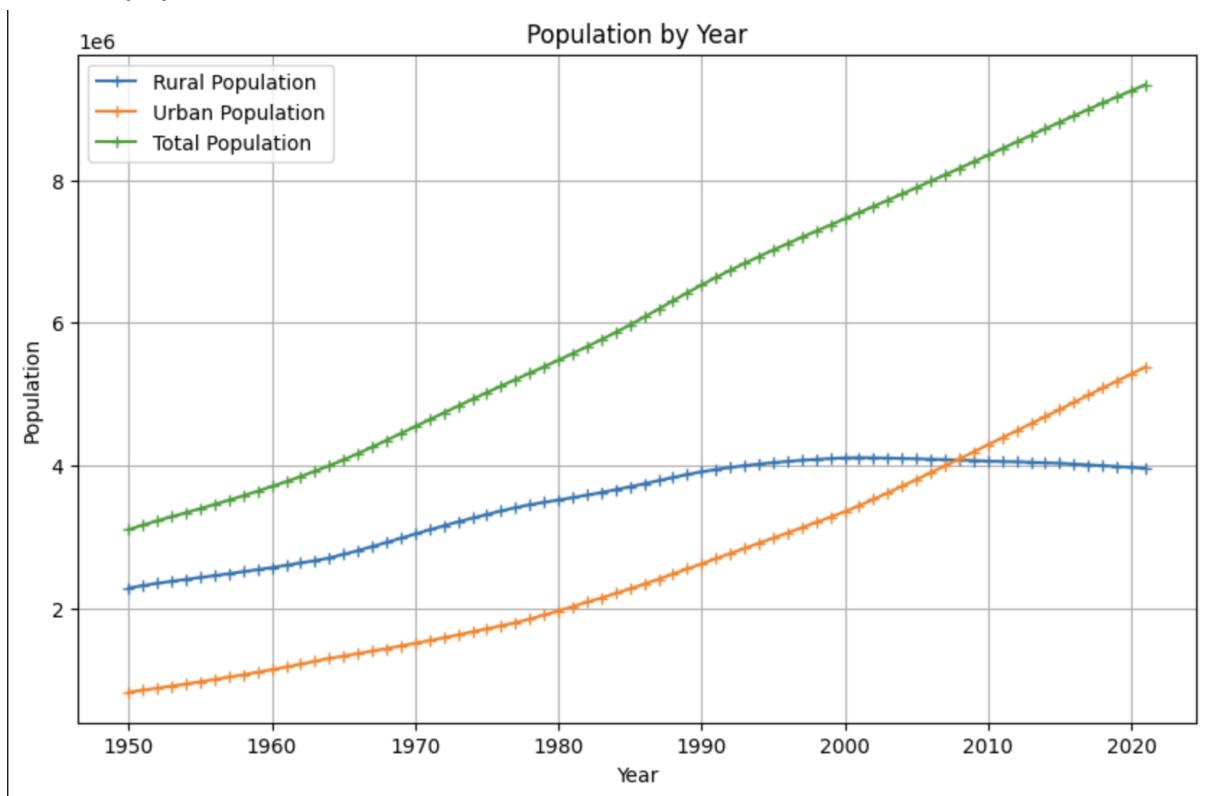
Nous avons envisagé plusieurs solutions. Nous avons choisi d'utiliser la fonction groupby() de pandas avec la fonction sum() pour regrouper toutes les données par pays et construire des données mondiales. En utilisant cette méthode nous nous retrouvons avec la colonne 'Year' qui se répète 3 fois. Une fois pour chaque valeur de Element. Ce n'est pas encore suffisant. Il faudrait transformer les valeurs de la colonne 'Element' en colonnes et les valeurs dans 'Value' deviendront les valeurs associées.

Nous avons utilisé la fonction pivot, pour transformer notre dataframe en pivot table. La nouvelle colonne 'Total Population - Both sexes' ne correspond pas à la somme des colonnes 'Urban population' et 'Rural population'. Nous avons drop la colonne et avons créé la nouvelle colonne 'Total population' comme la somme des deux.

## Graphe de l'évolution des populations rurales et urbaines dans le monde

Avec la nouvelle dataframe qui est maintenant utilisable dans des modèles de prédiction, nous pouvons visualiser la tendance d'évolution des courbes au cours des années.

*Plot des populations totales, rurales et urbaines mondiales :*



## 2. Modèles Temporels et Réseau de Neurones

### A. Modèles préliminaires : vers un premier modèle sur les données de températures globales, terrestres et maritimes

Pour une étude préliminaire, nous avons choisi d'implémenter un modèle statistique "simple" comme un modèle ARIMA, ARMA ou SARIMA.

Nous avons utilisé un modèle ARIMA. Nous allons détailler les raisons de ce choix et discuter des résultats que nous pourrons obtenir. Il ne s'agira que d'un modèle d'exploration. La raison pour laquelle il ne s'agit que d'un modèle d'exploration est que la variable endogène des anomalies de température ne dépend pas d'autres facteurs exogènes éventuels.

Premièrement pour pouvoir faire le modèle de prédiction, nous avons transformé la colonne 'Year' en Index et en objet DateTime.

#### 1. Modèle pour les températures globales

Nous récupérons les données des températures globales. Nous utilisons la méthode de Dickey-Fuller pour voir si la série est stationnaire. Nous pouvons déjà constater qu'elle n'est pas stationnaire puisque les variances et moyennes ne semblent pas constantes. Mais cette méthode de visualisation est subjective, nous avons fait un test de Dickey-Fuller pour s'en assurer.

La p-value est de: 0.998

Le test de Dickey-Fuller renvoie une p-valeur de 0.998 supérieure à 0.05. Nous pouvons rejeter l'hypothèse nulle et affirmer que la série temporelle des températures globales n'est pas stationnaire.

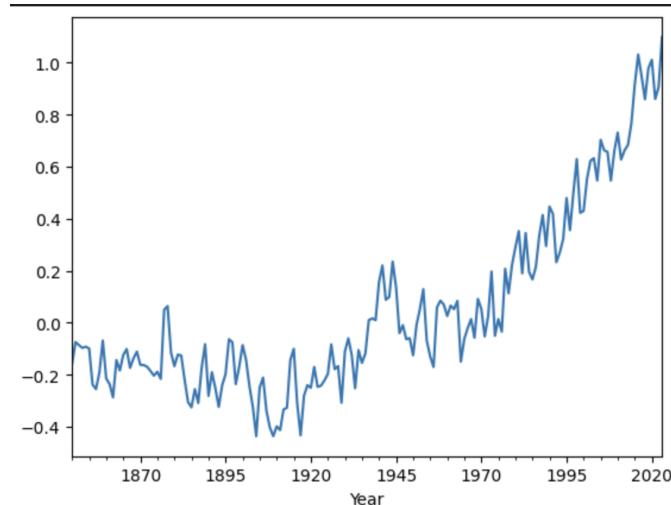
Nous pouvons alors utiliser 2 modèles possibles :

- Un modèle ARMA mais en utilisant la méthode de différenciation pour stationnariser la série temporelle.
- Un modèle ARIMA sur la série temporelle directement.

Nous avons implémenté ces 2 méthodes.

**Modèle ARMA avec une méthode de différenciation pour stationner la série temporelle :**

La série temporelle n'indique pas la présence de saisonnalité. Mais pour s'en assurer, nous avons retirer la saisonnalité avec la fonction `seasonal_decompose()`

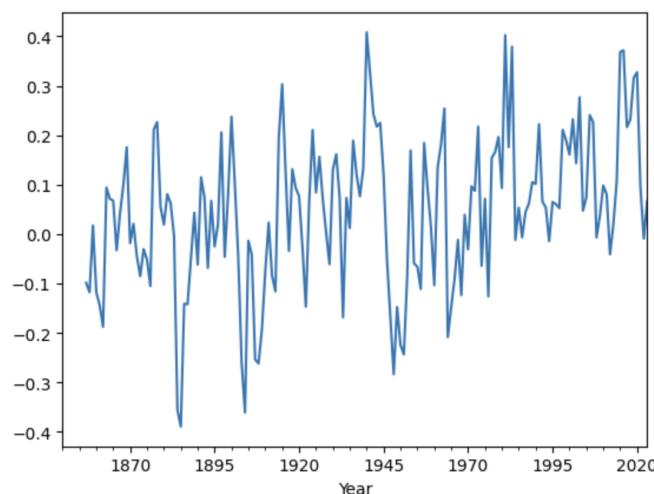


p-value adf : 0.997651554053703

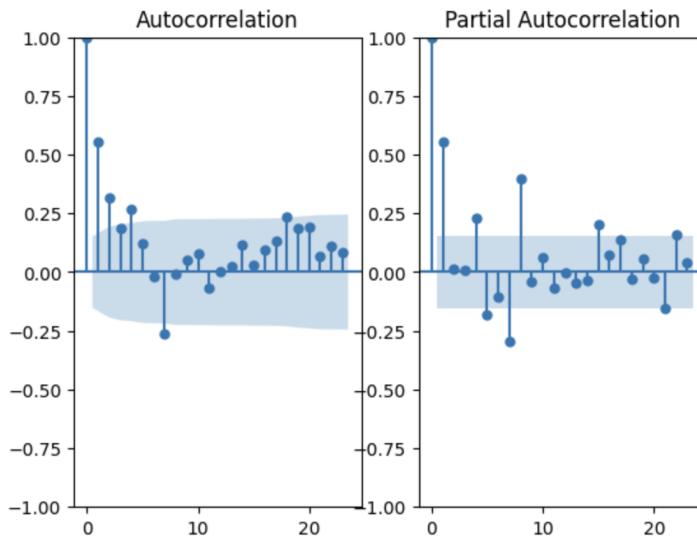
On remarque une infime différence entre les deux p-values calculées par le test de Dickey-Fuller.

La p-value est toujours trop grande, la série temporelle n'est toujours pas stationnaire. Il faut encore modifier la série. Nous avons utilisé la différenciation.

Nous avons choisi le meilleur paramètre possible pour faire la différenciation. La valeur  $d = 7$  pour la différenciation est la limite de stationnarité. On va choisir cette valeur même si on perd quand même un peu l'allure de base.



Nous avons affiché les PACF (Partial Autocorrelation) et ACF (Autocorrelation)



On voit 5 pics significatifs sur le PACF, on peut intuiter un AR(5). On voit aussi 5 pics significatifs sur l'ACF, on peut intuiter un MA(5).

On débutera par un ARMA(5,5).

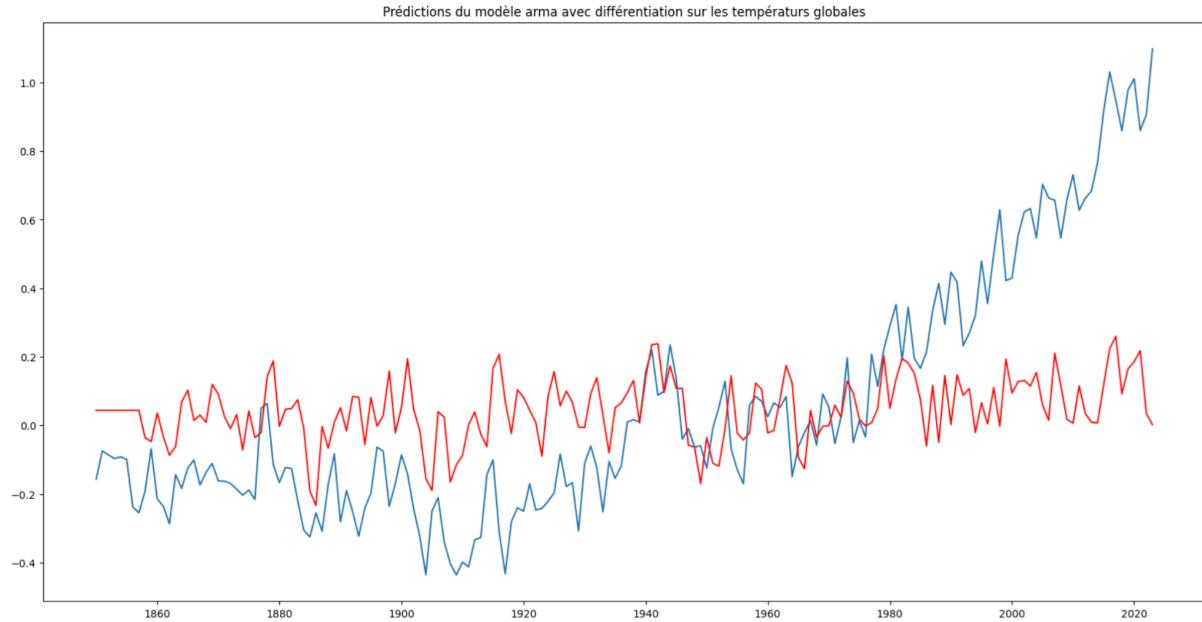
Après plusieurs essais, nous avons défini la meilleure configuration pour le modèle ARMA.

ARMA : (3,0,1)

| SARIMAX Results         |                            |                   |          |       |        |        |
|-------------------------|----------------------------|-------------------|----------|-------|--------|--------|
| Dep. Variable:          | y                          | No. Observations: | 174      |       |        |        |
| Model:                  | ARIMA(3, 0, 1)             | Log Likelihood    | 115.919  |       |        |        |
| Date:                   | Sun, 12 Nov 2023           | AIC               | -219.838 |       |        |        |
| Time:                   | 23:44:43                   | BIC               | -200.884 |       |        |        |
| Sample:                 | 01-01-1850<br>- 01-01-2023 | HQIC              | -212.149 |       |        |        |
| Covariance Type:        | opg                        |                   |          |       |        |        |
|                         | coef                       | std err           | z        | P> z  | [0.025 | 0.975] |
| const                   | 0.0441                     | 0.020             | 2.150    | 0.032 | 0.004  | 0.084  |
| ar.L1                   | -0.2469                    | 0.102             | -2.427   | 0.015 | -0.446 | -0.047 |
| ar.L2                   | 0.5444                     | 0.070             | 7.779    | 0.000 | 0.407  | 0.682  |
| ar.L3                   | -0.1538                    | 0.087             | -1.772   | 0.076 | -0.324 | 0.016  |
| ma.L1                   | 0.8819                     | 0.073             | 12.081   | 0.000 | 0.739  | 1.025  |
| sigma2                  | 0.0145                     | 0.002             | 8.322    | 0.000 | 0.011  | 0.018  |
| Ljung-Box (L1) (Q):     | 0.20                       | Jarque-Bera (JB): | 0.22     |       |        |        |
| Prob(Q):                | 0.66                       | Prob(JB):         | 0.90     |       |        |        |
| Heteroskedasticity (H): | 0.77                       | Skew:             | -0.08    |       |        |        |
| Prob(H) (two-sided):    | 0.32                       | Kurtosis:         | 2.94     |       |        |        |

Les p-values obtenues pour les composants du modèle sont correctes, hormis la 3ème composante de l'AR.

Nous pouvons faire un predict sur les données que nous avons et observer la véracité de nos résultats. On peut voir que la prédition avec le modèle ARMA n'est pas adaptée à notre cas. On pourrait l'expliquer la différenciation effectuée pour stationnariser la série. On peut voir que la série stationnaire ne ressemble plus à la série d'origine.

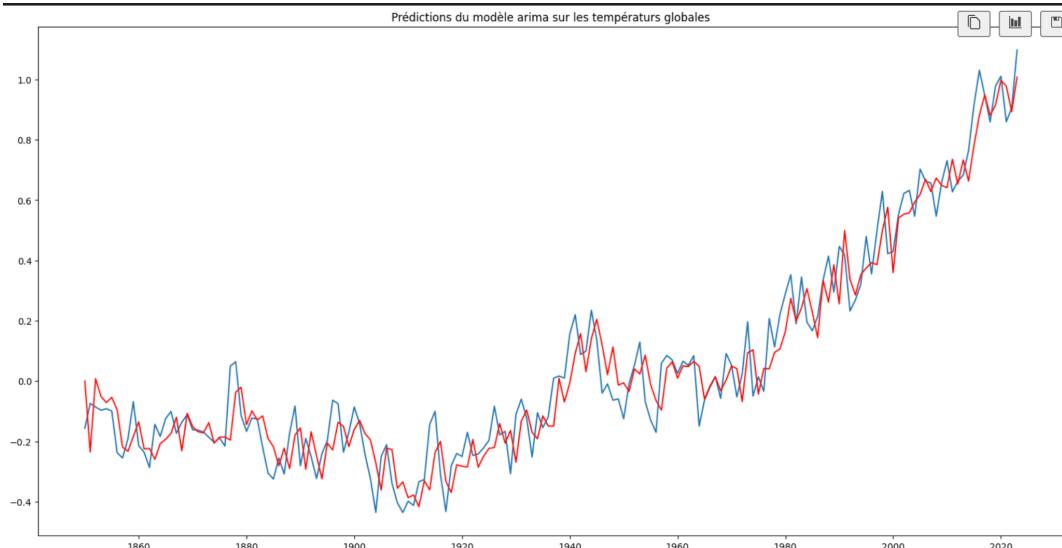


### Modèle ARIMA sur la série temporelle directement :

A contrario du modèle ARMA, nous n'avons pas besoin de faire de différenciation. Nous pouvons appliquer directement le modèle.

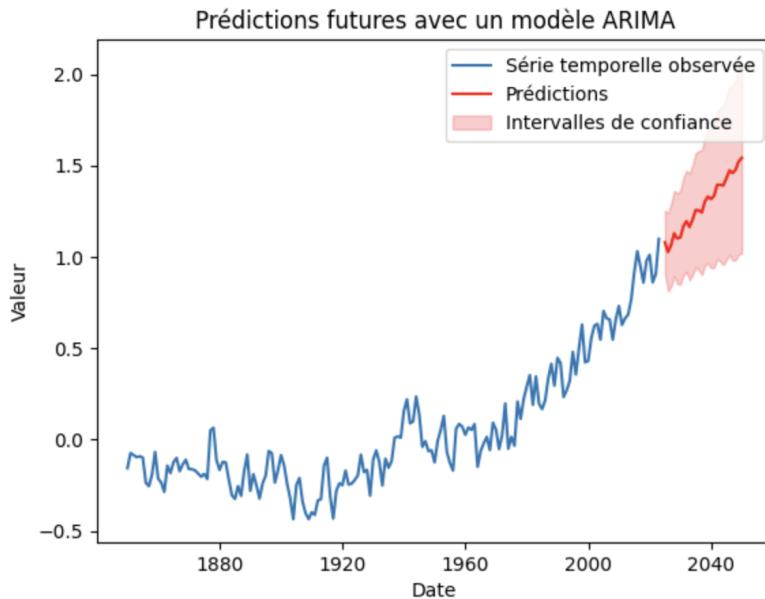
Après plusieurs essais, nous avons défini la meilleure configuration pour le modèle ARIMA.

ARIMA : (4,2,7).



Le modèle ARIMA est visiblement le meilleur modèle. En faisant la différentiation sur la série temporelle nous avons perdu l'allure de la série.

Maintenant que nous avons un modèle de prédition nous pouvons faire des prédictions pour les années futures, c'èd de 2024 à 2050 avec des **Intervalles de Confiance**.

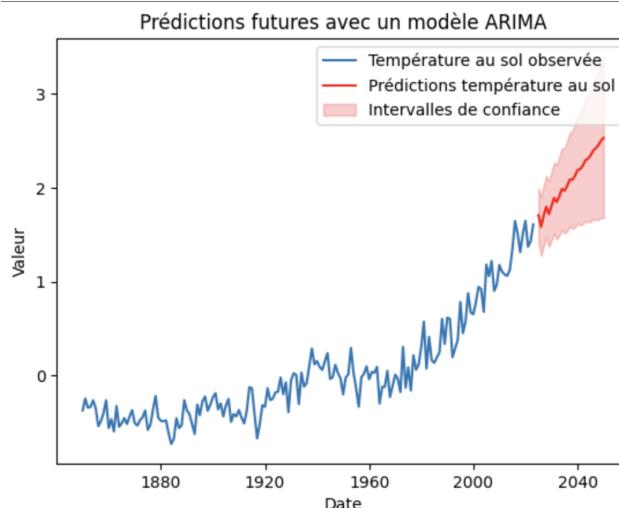


On peut voir que les anomalies de température globales tendent à augmenter. L'intervalle de confiance me semble, toutefois, correct. L'intervalle est assez étroit et suit les oscillations.

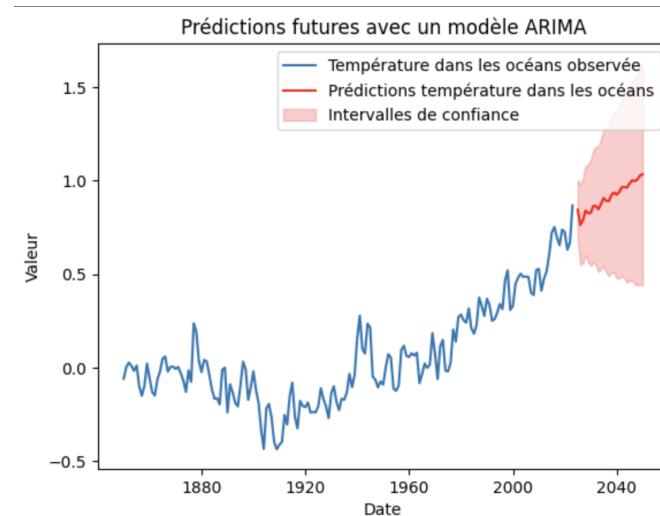
## 2. Modèles pour les températures terrestres et maritimes

Nous avons refait la même méthode d'implémentation pour les températures terrestres et maritimes.

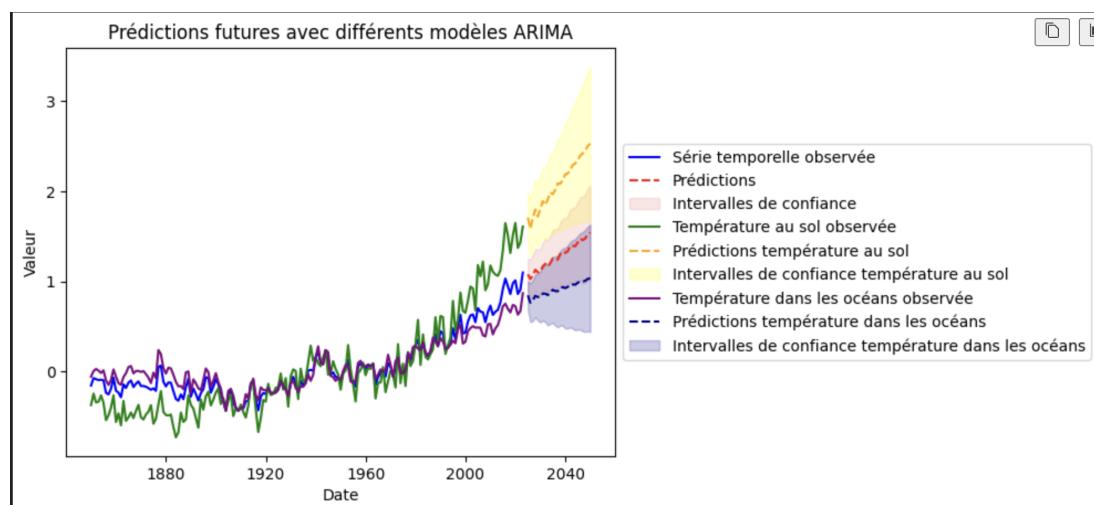
*Prédictions futures sur les températures au sol du modèle ARIMA :*



## Prédictions futures sur les températures au niveau des océans du modèle ARIMA :



## Regroupement de toutes les prédictions futures des anomalies globales, terrestres et maritimes



On peut voir que toutes les anomalies de température tendent à augmenter sans s'améliorer. Le modèle ne prend pas en compte d'autres facteurs exogènes. Mais si on ne prend que ces prédictions, on peut voir que le scénario catastrophique annoncé par le GIEC d'une augmentation des températures terrestres de +2°C pour 2050 risque d'être dépassé.

## B. Modèle ARIMAX : nous souhaitons prédire les anomalies globales en fonction de facteurs exogènes

Nous avons construit la datafarme suivante :

| Year       | averageAnomaly-ocean | averageAnomaly-land | averageAnomaly-global | Rural population | Urban population | Total population | log-rural-population | log-urban-population | log-total-population |
|------------|----------------------|---------------------|-----------------------|------------------|------------------|------------------|----------------------|----------------------|----------------------|
| 1950-01-01 | -0.090000            | -0.202500           | -0.125000             | 2280229.823      | 820258.014       | 3100487.837      | 14.639787            | 13.617374            | 14.947070            |
| 1951-01-01 | 0.000833             | -0.023333           | -0.008333             | 2314969.612      | 848615.115       | 3163584.727      | 14.654907            | 13.651361            | 14.967216            |
| 1952-01-01 | 0.072500             | 0.011667            | 0.053333              | 2346274.223      | 876864.771       | 3223138.994      | 14.668339            | 13.684108            | 14.985866            |
| 1953-01-01 | 0.054167             | 0.292500            | 0.129167              | 2374872.565      | 905829.315       | 3280701.880      | 14.680454            | 13.716606            | 15.003568            |
| 1954-01-01 | -0.111667            | 0.027500            | -0.068333             | 2401878.756      | 935690.068       | 3337568.824      | 14.691762            | 13.749040            | 15.020753            |
| 1955-01-01 | -0.123333            | -0.140000           | -0.129167             | 2428203.147      | 966675.531       | 3394778.678      | 14.702662            | 13.781515            | 15.037749            |
| 1956-01-01 | -0.095000            | -0.333333           | -0.170000             | 2454649.464      | 998455.413       | 3453104.877      | 14.713495            | 13.813965            | 15.054784            |
| 1957-01-01 | 0.095833             | -0.020000           | 0.059167              | 2481796.367      | 1031267.971      | 3513064.338      | 14.724493            | 13.846300            | 15.071999            |
| 1958-01-01 | 0.117500             | 0.016667            | 0.085000              | 2509646.174      | 1065298.708      | 3574944.882      | 14.735652            | 13.878766            | 15.089460            |
| 1959-01-01 | 0.062500             | 0.096667            | 0.070000              | 2538426.784      | 1100422.902      | 3638849.686      | 14.747055            | 13.911205            | 15.107178            |

Nous avons décidé de modéliser et de prédire les anomalies de température globale en utilisant un modèle ARIMAX (AutoRegressive Integrated Moving Average with eXogenous factors). Ce choix de modèle s'explique par le fait que la série temporelle ne présente pas de saisonnalité et n'est pas stationnaire.

Nous avons commencé par diviser nos données en deux ensembles distincts, soit un ensemble d'entraînement et un ensemble de test. Cette division, effectuée avec une répartition de 80%, vise à utiliser une partie des données pour former notre modèle et une autre partie pour évaluer ses performances.

Les variables endogènes ont été définies pour l'ensemble d'entraînement. Dans notre cas, la variable endogène est l'« averageAnomaly-global », représentant les anomalies de température globale. Quant aux variables exogènes, nous avons sélectionné celles qui semblent avoir une corrélation avec notre variable cible, telles que les anomalies océaniques, terrestres, et les populations rurales, urbaines, et totales, toutes transformées de manière appropriée en logarithme pour les adapter à l'échelle des valeurs.

En utilisant ces variables, nous avons construit un modèle ARIMAX avec une ordonnance (1, 1, 1)

Nous avons ajusté le modèle aux données d'entraînement, optimisant ainsi les paramètres pour minimiser les erreurs de prédiction. Le modèle a appris à reconnaître les schémas temporels et les relations avec les variables exogènes.

Une fois le modèle entraîné, nous l'avons utilisé pour faire des prédictions sur l'ensemble de test. Ces prédictions nous permettent d'évaluer la capacité du modèle à généraliser ses connaissances à de nouvelles données.

Pour mesurer l'efficacité de nos prédictions, nous avons choisi de calculer la Mean Squared Error (MSE) sur les données de test.

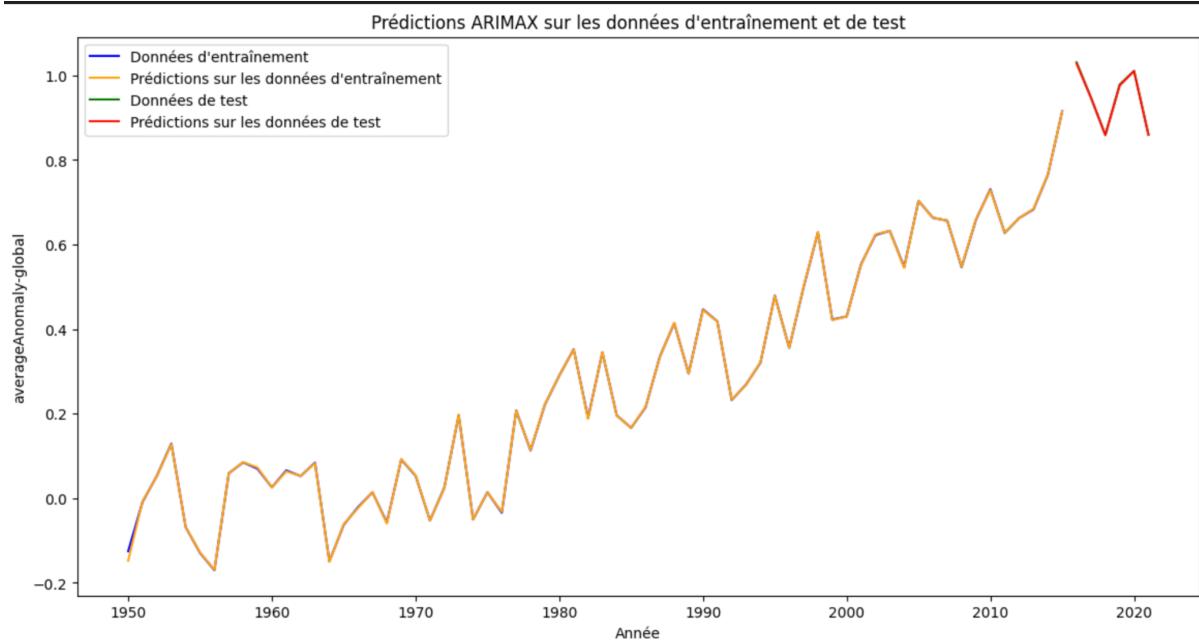
MSE sur les données de test : 1.2514308657183772e-06

Nous avons fait une méthode de Validation Croisée avec une métrique MSE pour vérifier la pertinence des résultats.

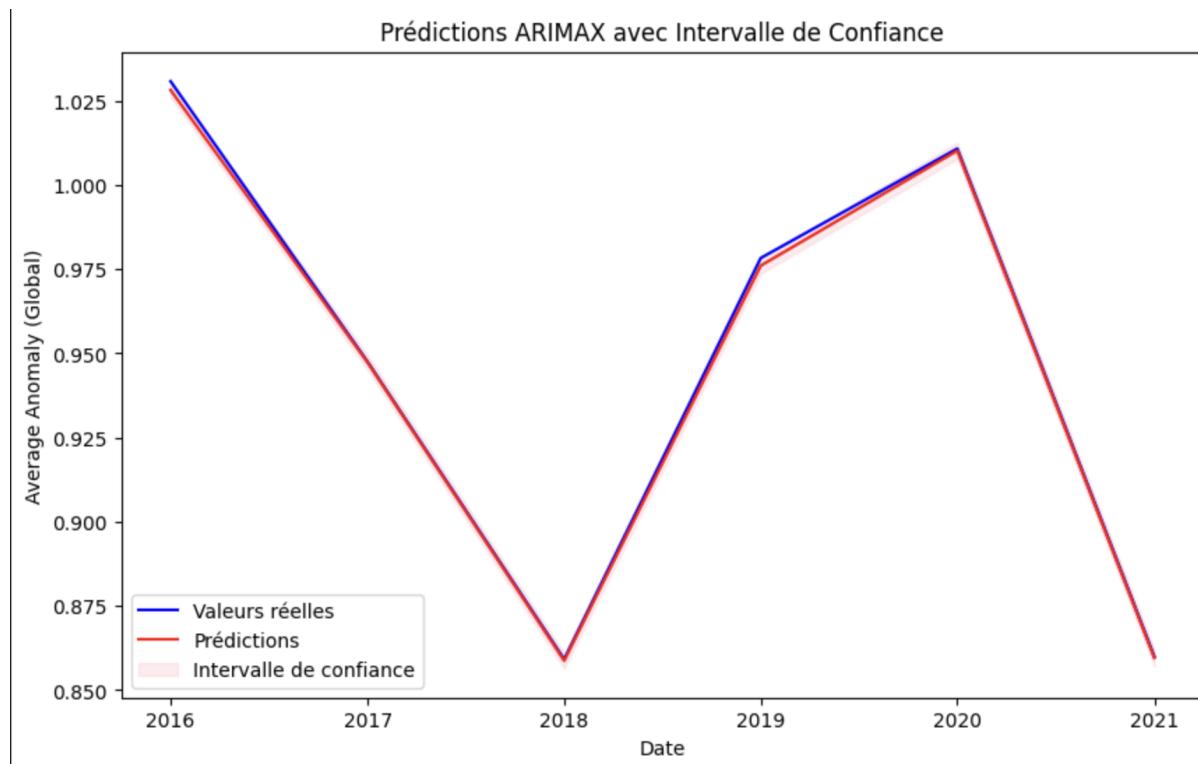
MSE pour la validation croisée 1: 7.047532941260657e-05  
MSE pour la validation croisée 2: 1.449050099566505e-06  
MSE pour la validation croisée 3: 3.956949538100025e-06  
MSE pour la validation croisée 4: 1.7829377890263276e-06  
MSE pour la validation croisée 5: 9.483565333689199e-07  
MSE pour la validation croisée 6: 1.7329525767956342e-06  
MSE pour la validation croisée 7: 1.0696984740278656e-06  
MSE pour la validation croisée 8: 1.1970070537486592e-06  
MSE pour la validation croisée 9: 7.124322318485583e-07  
MSE pour la validation croisée 10: 2.1083444460968564e-06  
Moyenne des MSE : 8.543305815518592e-06

Nous avons affiché les prédictions sur les données d'apprentissage et sur les données de test.

### Prédictions sur train et test



## Prédictions sur test



En première conclusion, nous pouvons voir que le modèle ARIMAX a été particulièrement performant puisqu'il réussit à prédire aussi bien la tendance de la courbe que les variations.

Nous sommes quand même surpris d'un tel résultat, ça ne devrait pas être possible à part s'il s'avère que nous sommes des génies, mais je ne pense pas que ça soit le cas.

Il y a un risque d'overfitting...

Nous aurions aimé faire un forecast de 2022 à 2050 mais la modèle AIMAX étant dépendant des facteurs exogènes il aurait fallu construire des valeurs pour chaque facteur. Nous pouvons réutiliser les modèles ARIMA faits précédemment. Mais pour les données de population, il faudrait faire une tendance d'évolution jusqu'à une valeur de seuil, par exemple 9,7 milliards pour la population mondiale.

## C. Modèle RNN

**Notebook :** Modèle RNN Data mean temperatures.ipynb

**Modèle :** RNN\_Data\_mean\_temperatures\_final /  
RNN\_Data\_mean\_temperatures\_final.h5

Pour entraîner le modèle, nous nous sommes basés sur le dataset utilisé précédemment contenant les données d'anomalies de température terrestre, océaniques et globales depuis 1850 (moyenne par an).

Notre étape de prétraitement impliquait la normalisation des données d'anomalie de température en utilisant le MinMaxScaler, qui recalcule les données dans une plage [0, 1], facilitant ainsi la convergence de notre modèle RNN. Nous avons ensuite converti les données normalisées en un DataFrame Pandas et créé des séquences avec une taille de fenêtre spécifiée. Ces séquences servent de caractéristiques d'entrée à notre modèle, lui permettant d'apprendre des dépendances temporelles inhérentes aux données de séries temporelles.

Pour la validation du modèle, nous avons isolé un sous-ensemble de données jusqu'à l'année 2000, assurant la robustesse de notre modèle en testant ses capacités prédictives sur des données non vues. Ce sous-ensemble a subi le même processus de normalisation, et nous avons préparé les séquences de validation de la même manière que nos données d'entraînement, mettant ainsi en place les conditions pour une évaluation des performances du modèle.

Lors de la phase de développement du modèle, nous avons conçu une architecture de réseau de neurones récurrents en utilisant keras. Notre modèle comprend une couche GRU avec une dimension latente de 5, qui capture les dépendances temporelles, suivie d'une couche dense qui produit l'anomalie de température prédite. Le modèle a été compilé avec l'optimiseur RMSprop et la fonction de perte d'erreur quadratique moyenne, prêt pour l'entraînement. Nous obtenons un modèle totalisant 156 paramètres entraînables.

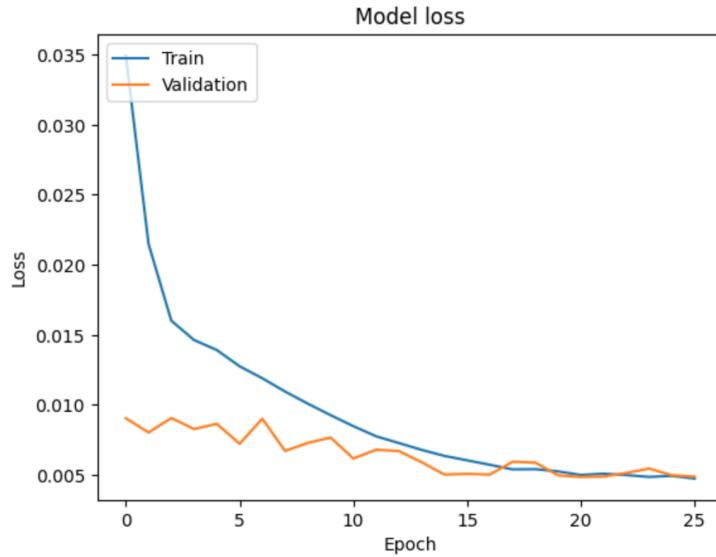
```

Model: "sequential_10"
=====
Layer (type)          Output Shape       Param #
=====
gru_10 (GRU)          (None, 5)           150
dense_10 (Dense)      (None, 1)            6
=====
Total params: 156 (624.00 Byte)
Trainable params: 156 (624.00 Byte)
Non-trainable params: 0 (0.00 Byte)
=====
```

Une fois les séquences de données préparées, nous avons procédé à leur conversion en tableaux Numpy, ce qui est une étape nécessaire pour l'entraînement de notre modèle. Cette conversion a été réalisée pour les ensembles d'entraînement **X\_train** et **y\_train**, les préparant ainsi pour l'alimentation dans l'architecture RNN.

L'entraînement du modèle a été réalisé avec les données préalablement préparées et normalisées. Nous avons utilisé l'EarlyStopping pour surveiller la perte de validation, ce qui permet d'arrêter l'entraînement lorsque les améliorations deviennent négligeables, empêchant ainsi le surapprentissage. Le modèle a été entraîné avec une taille de lot définie par **BATCH\_SIZE** et un nombre d'époques défini par **EPOCHS**, avec les données de validation pour suivre la performance du modèle.

Pour évaluer la performance du modèle durant l'entraînement, nous avons visualisé la courbe de perte pour les ensembles d'entraînement et de validation. Cette visualisation nous a permis de confirmer que le modèle apprend correctement et que la perte diminue de manière stable, ce qui indique que le modèle se généralise bien aux données non vues.



**Diminution rapide de la perte** : La courbe de perte d'entraînement (Train) chute rapidement, ce qui suggère que le modèle apprend efficacement à partir des données d'entraînement.

**Convergence des courbes** : Les courbes de perte d'entraînement et de validation (Validation) semblent converger, ce qui est un bon signe indiquant que le modèle ne surapprend pas trop (overfitting) sur les données d'entraînement.

**Stabilisation de la perte de validation** : La perte de validation semble se stabiliser après quelques epochs, ce qui peut indiquer que des epochs supplémentaires pourraient ne pas apporter d'amélioration significative.

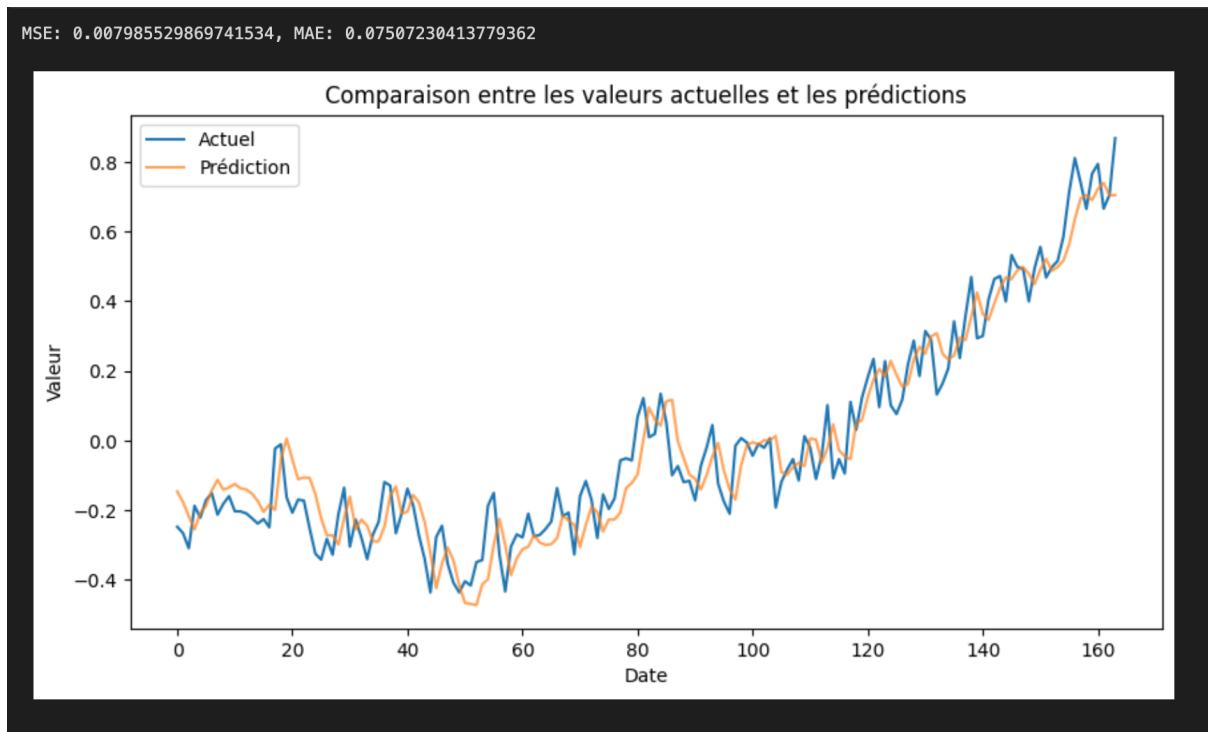
Après avoir normalisé l'ensemble de test en utilisant le même scaler que pour les données d'entraînement, nous avons préparé les séquences de test et les avons converties en tableaux Numpy. Ensuite, nous avons effectué des prédictions sur cet ensemble de test à l'aide de notre modèle. La forme des prédictions, correspondant à la forme des données de test, indique une concordance adéquate pour l'évaluation des performances.

Une fois les prédictions réalisées par le modèle, nous avons inversé la mise à l'échelle des données pour interpréter les résultats dans leur échelle originale. Un tableau de zéros a été créé pour adapter la forme requise par le scaler, et les prédictions ainsi que les données de test y ont été insérées pour l'inversion. Cela a permis de comparer les valeurs prédites avec les valeurs réelles dans leur format d'origine.

Pour évaluer la précision de notre modèle, nous avons calculé l'erreur quadratique moyenne (MSE) et l'erreur absolue moyenne (MAE) entre les prédictions et les

valeurs réelles. Ces métriques sont essentielles pour quantifier la performance du modèle et identifier les marges d'amélioration.

La visualisation suivante illustre la comparaison entre les valeurs réelles et celles prédites par le modèle.



Le modèle semble capturer la tendance ascendante des données, ce qui est bon signe pour la prédition de séries temporelles, surtout pour des données qui pourraient avoir une tendance linéaire ou polynomiale.

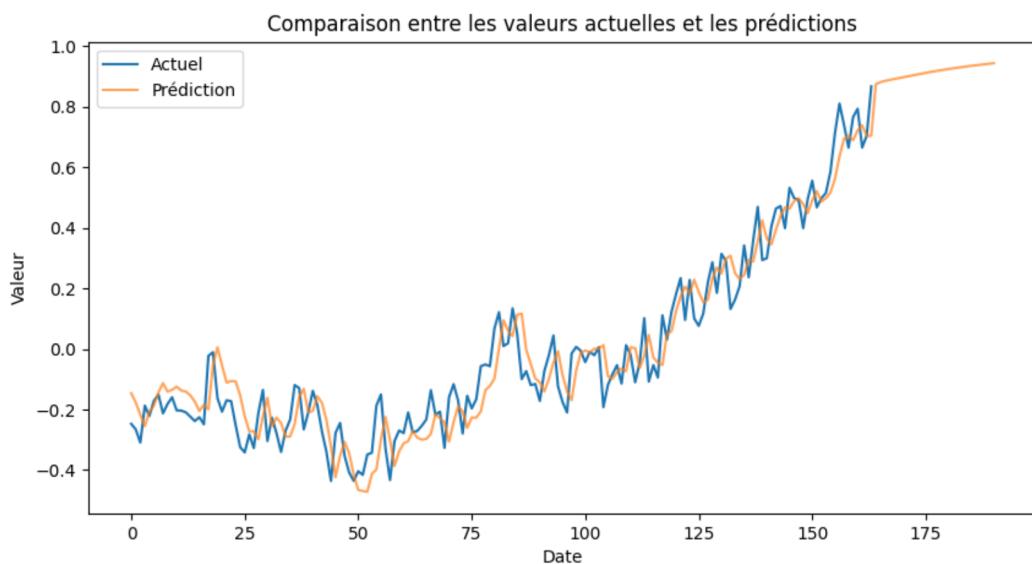
Il semble y avoir quelques écarts entre les valeurs réelles et prédites, notamment là où les données réelles présentent des fluctuations. Cela pourrait indiquer que le modèle pourrait être amélioré pour capturer de manière plus précise les fluctuations à court terme.

Les prédictions semblent être légèrement décalées par rapport aux valeurs réelles. Cela pourrait indiquer un biais dans le modèle ou une nécessité d'ajuster davantage le processus de normalisation et de dénormalisation des données.

Pour étendre notre analyse, nous avons implémenté une méthode de prédition multi-étapes, permettant de générer des prédictions sur un horizon temporel plus long (10 ans). Le modèle a été utilisé itérativement pour prédire la prochaine étape, en utilisant à chaque fois la dernière prédition comme partie de l'entrée pour la prédition suivante. Cette approche est souvent utilisée pour simuler des prédictions dans le futur.

```
# Prolongement sur 10 ans  
model.add(Dense(10))  
✓ 0.0s
```

Enfin, pour préparer les données pour une visualisation étendue des prédictions, nous avons converti les valeurs prédites en un tableau bidimensionnel, correspondant à la structure attendue pour une concaténation ultérieure. Les prédictions initiales et étendues ont été combinées pour former une série temporelle complète, représentant la vision du modèle sur la période historique et prolongée.



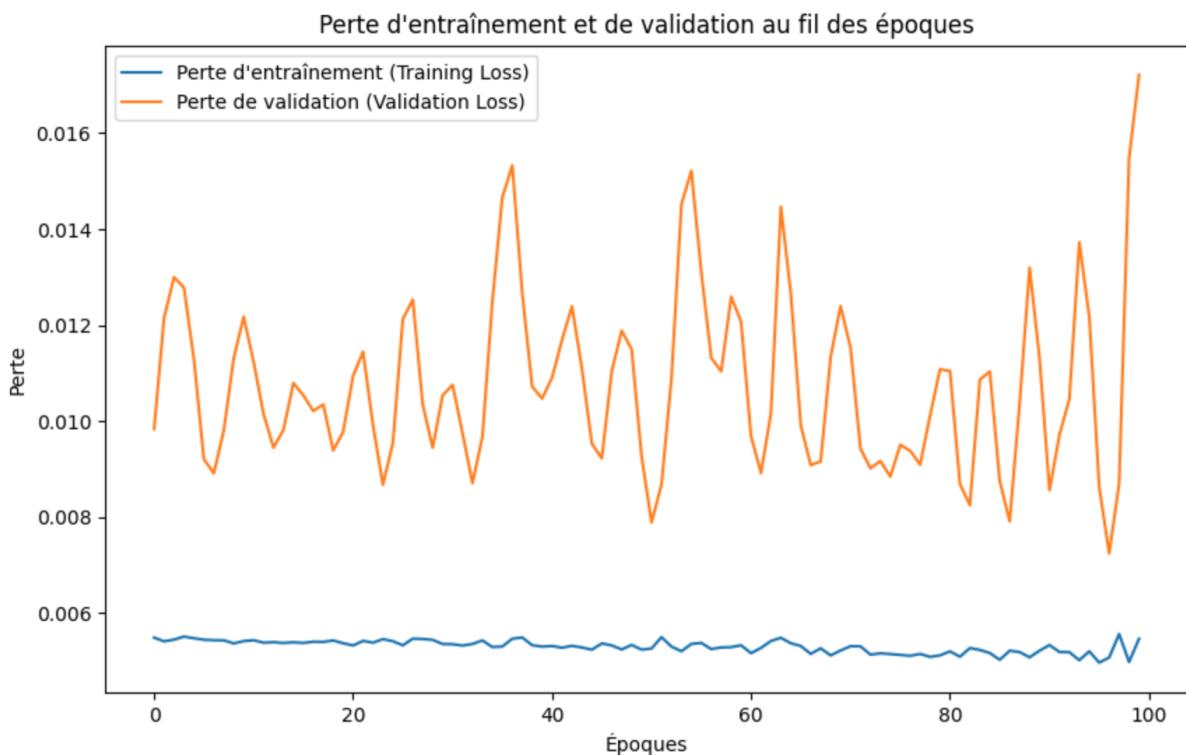
La figure finale illustre la comparaison entre les valeurs réelles et les prédictions prolongées générées par notre modèle. Les prédictions étendues sur l'horizon temporel montrent la capacité du modèle à extrapoler les tendances futures des anomalies de température. On peut observer une tendance légèrement croissante. Cependant, un modèle similaire entraîné sur un dataset plus complet, notamment avec des données d'émission de gaz à effet de serre et démographique pourrait donner une prédition plus proche de la tendance, qui serait logiquement très ascendante.

## D. Modèle RNN sur un modèle plus riche

**Notebook :** Modèle RNN basé un dataset large.ipynb

Nous avons effectué une création de modèle RNN similaire mais sur un dataset plus riche.

**Premiers résultats :**



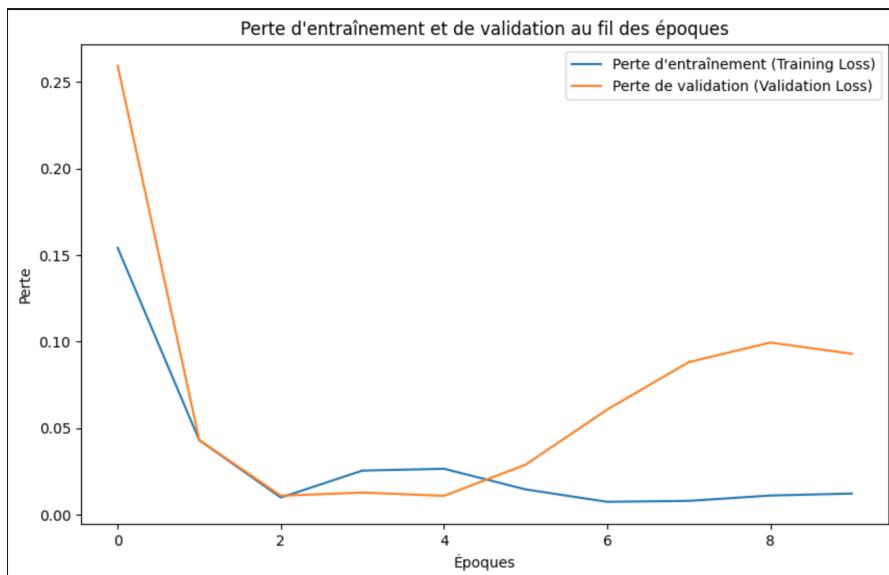
**Perte d'entraînement** : La perte d'entraînement diminue progressivement, ce qui indique que le modèle apprend et s'améliore au fur et à mesure de l'entraînement. La courbe bleue semble se stabiliser vers la fin, ce qui suggère que le modèle commence à converger et que des itérations supplémentaires pourraient ne pas apporter d'amélioration significative.

**Perte de validation** : La perte de validation, en revanche, montre une grande variabilité et augmente considérablement vers les dernières époques. Cette volatilité et l'augmentation de la perte de validation peuvent indiquer un problème de surapprentissage, où le modèle se performe bien sur les données d'entraînement mais a du mal à généraliser sur les données qu'il n'a jamais vues. Cela peut aussi être le signe d'une taille de lot trop petite, d'un taux d'apprentissage inapproprié ou d'autres problèmes liés à la convergence du modèle.

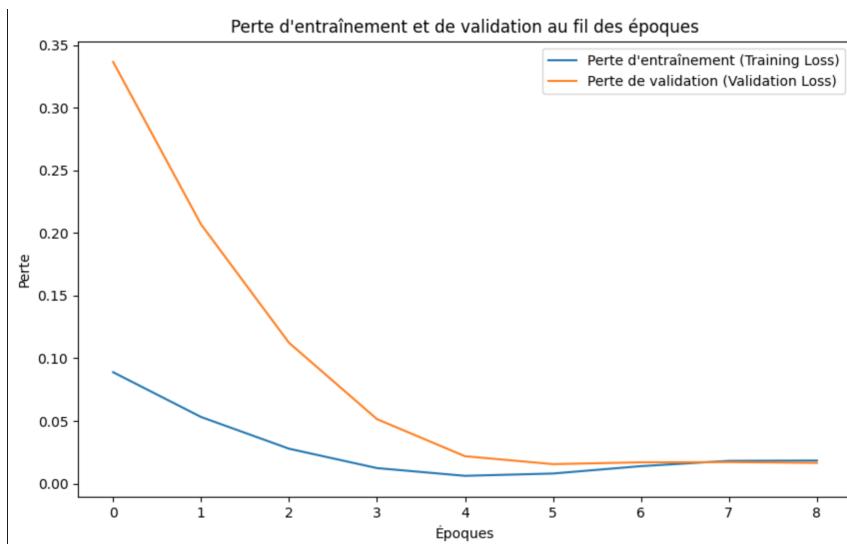
**Surapprentissage** : Le fait que la perte de validation soit sensiblement plus élevée que la perte d'entraînement, et surtout qu'elle augmente vers la fin, suggère que le modèle surajuste les données d'entraînement. Cela signifie que le modèle apprend des détails et du bruit spécifiques à l'ensemble d'entraînement qui ne s'appliquent pas à l'ensemble de validation.

Pour remédier à cela, nous avons mis en place de l'**Early Stopping**.

**Résultat pour 100 epochs et une patience à 9 pour l'Early Stopping**



### Résultat pour 10 epochs et patience à 3 pour l'Early Stopping



Les résultats obtenus sont meilleurs, nous obtenons la convergence des courbes.

## Conclusion

Notre investigation a révélé que le modèle ARIMAX surpassé les autres approches en termes de précision prédictive pour notre ensemble de données. Néanmoins, il convient de noter que nos résultats semblent moins alarmants que le consensus général au sein de la communauté scientifique concernant les projections climatiques futures. Cet écart suggère que notre modèle pourrait bénéficier de l'intégration de variables supplémentaires qui sont des déterminants connus du changement climatique.

Les modèles climatiques sont intrinsèquement complexes et multidimensionnels, et les prévisions précises dépendent non seulement des données historiques, mais également de l'inclusion de facteurs exogènes influençant le climat. En intégrant des données sur les émissions de gaz à effet de serre, qui sont un moteur clé du réchauffement planétaire, ainsi que des variables telles que l'évolution démographique, les changements dans les moyens de transport, et les tendances de l'industrialisation, nous pourrions affiner nos prévisions.

De plus, une méthode de travail plus incrémentale en augmentant progressivement la taille du jeu de données aurait pu conduire à des résultats plus précis.

En conclusion, bien que le modèle ARIMAX présente les meilleures performances, il est clair que l'adoption d'une approche plus holistique, incorporant un spectre plus large de variables environnementales et socio-économiques, est nécessaire pour capturer l'étendue des influences sur les anomalies de température globale.