SAE 303 : Description et prévision de données temporelles

Groupe : Nayl Saifoudine, Andreja Jovanovic et Hassan Ajdahim

Partie 1 :

Rapport : Analyse des Créations d'Entreprises en France (2000-2024)

Ce rapport détaille l'ensemble des étapes et méthodes utilisées pour analyser la série chronologique des créations d'entreprises en France sur la période 2000-2024. L'objectif principal était de répondre aux questions posées : isoler les composantes de la série (tendance, saisonnalité), les analyser, et établir des prévisions en utilisant des méthodes statistiques appropriées. Ce document vise à fournir une explication claire et complète des choix, étapes et résultats obtenus.

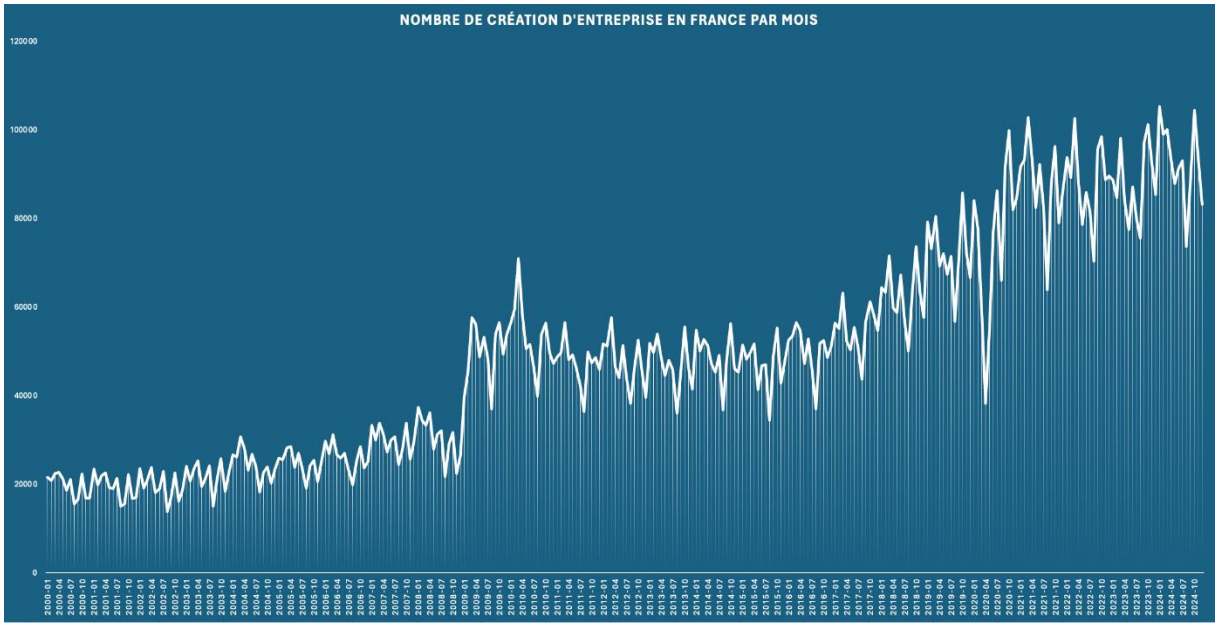
Choix de la série chronologique et justification

Pour cette étude, nous avons sélectionné la série chronologique des **créations mensuelles d'entreprises en France (**https://www.insee.fr/en/statistiques/series/116025894**)**, couvrant 25 années, de janvier 2000 à décembre 2024. Ce choix est motivé par plusieurs aspects :

1. **Pertinence économique** : Les créations d'entreprises sont un indicateur clé des dynamiques économiques et reflètent les cycles administratifs et financiers du pays. Cette série permet d'identifier les fluctuations régulières et les comportements macro-économiques.
2. **Existence d'une saisonnalité marquée** : Une observation initiale des données brutes montre des pics récurrents (en janvier et octobre, par exemple) ainsi que des creux marqués (notamment en août). Ces fluctuations saisonnières sont révélatrices de cycles prévisibles, influencés par des facteurs comme les vacances estivales ou les débuts d'années fiscales.
3. **Longue période d'analyse** : Avec 25 ans de données, la série est suffisamment complète pour permettre une analyse approfondie des tendances et de la saisonnalité, et pour produire des prévisions fiables.

Analyse du modèle additif

En analysant les données brutes (graphique ci-dessous), nous avons conclu que la série suit un **modèle additif**, où les composantes (tendance, saisonnalité et bruit) s'additionnent pour produire les valeurs brutes.



Pourquoi un modèle additif ?

Nous avons opté pour un modèle additif en raison des caractéristiques suivantes :

1. **Saisonnalité constante** : Les fluctuations saisonnières (pics et creux) restent **stables en amplitude** au fil du temps, indépendamment de la tendance générale. Par exemple, les creux d'août restent similaires, même lorsque le niveau global des créations d'entreprises augmente.
2. **Tendance linéaire globale** : La tendance générale montre une augmentation régulière des créations d'entreprises, compatible avec un modèle additif. Dans un modèle multiplicatif, la saisonnalité serait proportionnelle à la tendance, ce qui n'est pas le cas ici.
3. **Stabilité des variations saisonnières** : Les variations mensuelles, comme les pics en janvier ou octobre, ne deviennent pas plus importantes à mesure que le nombre total de créations d'entreprises augmente. Cette stabilité suggère que les composantes saisonnières agissent de manière additive.

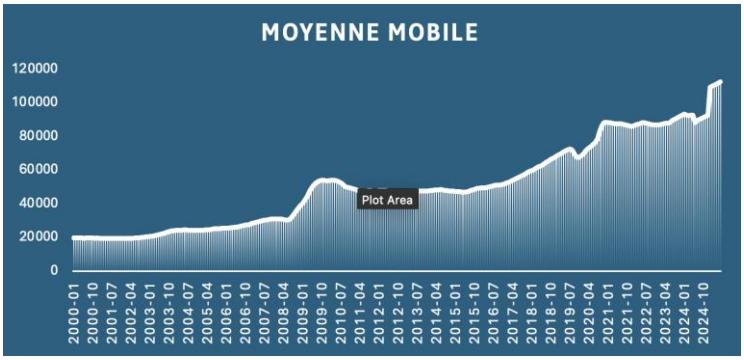
Isolation et Analyse de la Saisonnalité (Question 3)

La composante saisonnière a été isolée en suivant un processus en plusieurs étapes :

1. Calcul de la moyenne mobile

Nous avons calculé une **moyenne mobile sur 12 mois** (Comme 12 est pair, on utilise les 11 mois autour de la valeur initiale et ½ de la valeur de chaque extrémité des données) pour chaque point de la série. Cette moyenne mobile a permis de lisser les données brutes et

d’extraire la tendance tout en éliminant les fluctuations à court terme. Le résultat représente la tendance générale des créations d’entreprises.



2. Détermination de la composante saisonnière

Pour chaque point de données, nous avons soustrait la moyenne mobile des données brutes afin d’obtenir la composante saisonnière. Cette composante représente les fluctuations spécifiques à chaque mois, indépendamment de la tendance.

3. Moyenne des composantes saisonnières par mois

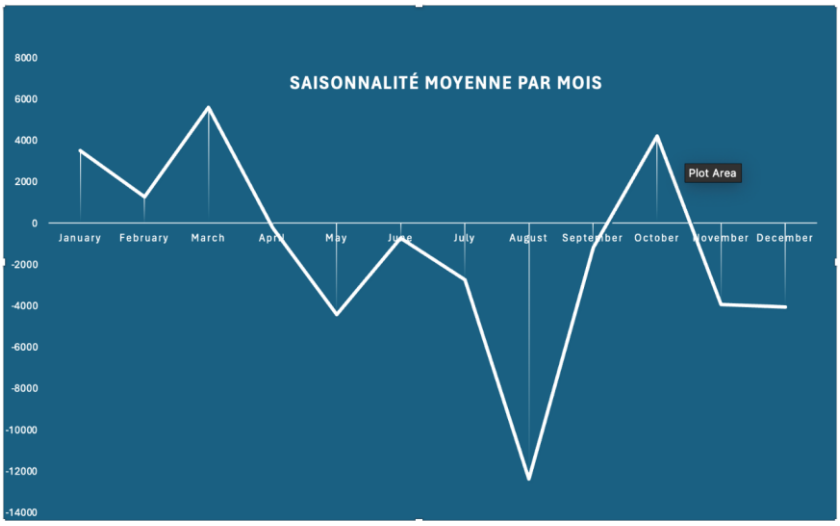
Pour chaque mois (janvier, février, etc.), nous avons calculé la moyenne des composantes saisonnières sur l’ensemble des années observées. Par exemple, nous avons pris toutes les valeurs de la composante saisonnière pour les mois de janvier et avons calculé leur moyenne. Cela a permis d’obtenir un tableau de la saisonnalité moyenne par mois, illustré dans le graphique ci-dessous. Ce graphique montre clairement les fluctuations récurrentes :

- **Pics saisonniers** : En janvier, mars et octobre.

En mars, le pic pourrait s’expliquer par la fin du premier trimestre fiscal. Les entrepreneurs profitent souvent de cette période pour finaliser leurs projets de création d’entreprises après avoir analysé les résultats de l'année précédente ou pour bénéficier de dispositifs fiscaux avantageux liés au début d'année. En octobre, un autre pic est visible, probablement lié à une préparation stratégique avant la fin de l'année. Les entrepreneurs anticipent souvent des formalités administratives ou des opportunités économiques liées aux fêtes de fin d'année.

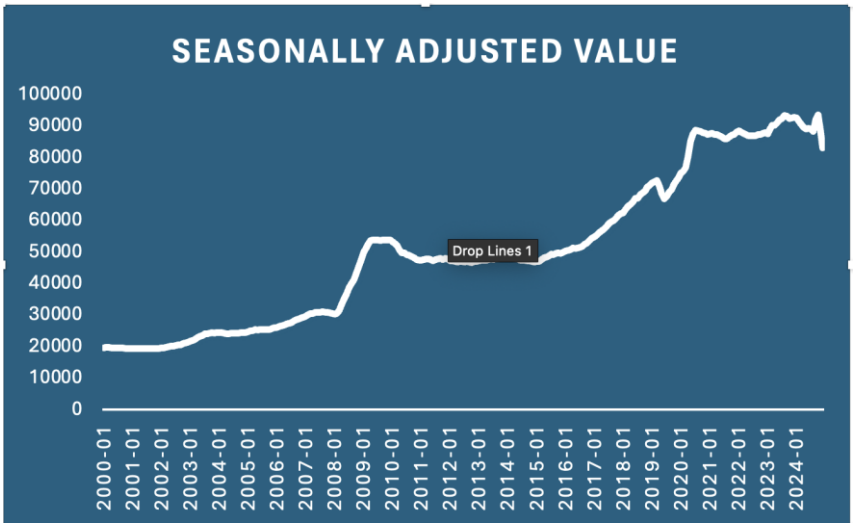
- **Creux saisonniers** : En août, avec une baisse importante.

En revanche, le creux d'août s'explique logiquement par les vacances estivales. Durant cette période, les activités administratives ralentissent, et de nombreux entrepreneurs préfèrent reporter leurs démarches à la rentrée, en septembre. Cela reflète le cycle économique global en France, où août est traditionnellement marqué par une diminution de l'activité dans de nombreux secteurs.



5. Valeurs ajustées saisonnièrement

En soustrayant la composante saisonnière des données brutes, nous avons obtenu les **valeurs ajustées saisonnièrement**. Ces valeurs permettent d'éliminer l'effet saisonnier, rendant la tendance et les variations aléatoires plus claires.



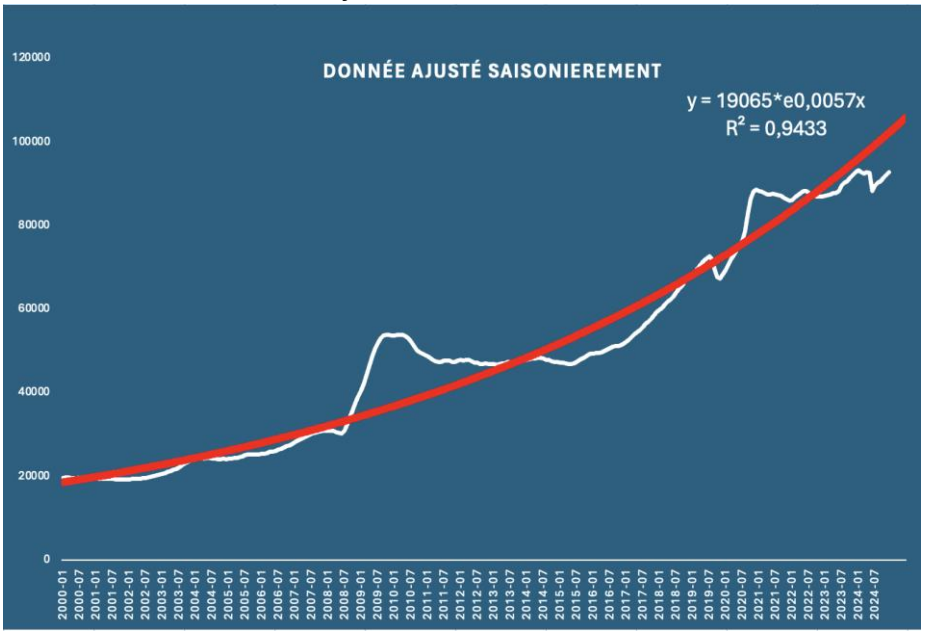
Modélisation et Prévisions

1. Courbe de Tendance Exponentielle

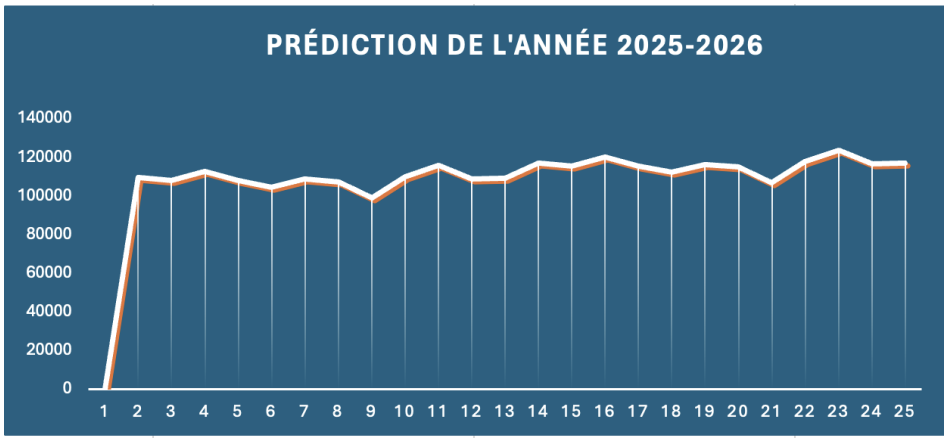
Une courbe de tendance exponentielle a été ajustée sur les valeurs ajustées saisonnièrement pour modéliser la tendance globale de la série. Voici les étapes suivies :

1. **Ajustement de la courbe** : Nous avons utilisé Excel tout d'abord dans le but d'afficher différentes courbes de tendances sur nos données afin d'identifier la courbe de tendance qui y correspond le mieux, ainsi nous avons découvert que la courbe exponentielle ci-dessous correspondait bien à l'allure de nos données d'où l'équation que l'on as sélectionner est :

$$y = 19065 \cdot e^{0,0057x}$$



1. **Validation de la courbe** : Le coefficient de détermination $R^2 = 0{,}9481$ indique une excellente correspondance entre la courbe ajustée et les données observées d'où notre utilisation de cette équation.
2. **Génération de prévisions** : En utilisant l'équation, nous avons inséré des valeurs futures pour x (par exemple, 301, 302 pour les mois de 2025,puisque nous n'avons que 300 lignes dans notre base de donnée) et avons calculé les prévisions pour ces périodes grâce à l'équation fournit par les outils d'Excel comme vu dans le graphique précédent.



Ce graphique représente les prédictions des créations d'entreprises pour les années 2025 et 2026, obtenues à l'aide de la courbe exponentielle avec intégration de la saisonnalité. On observe une évolution relativement régulière, avec des fluctuations mensuelles reflétant les cycles saisonniers identifiés dans les données historiques. Les pics et creux correspondent aux variations saisonnières typiques, comme un pic en janvier et un creux en août, bien que la tendance globale continue de croître.

2. Lissage Exponentiel Double (LED)

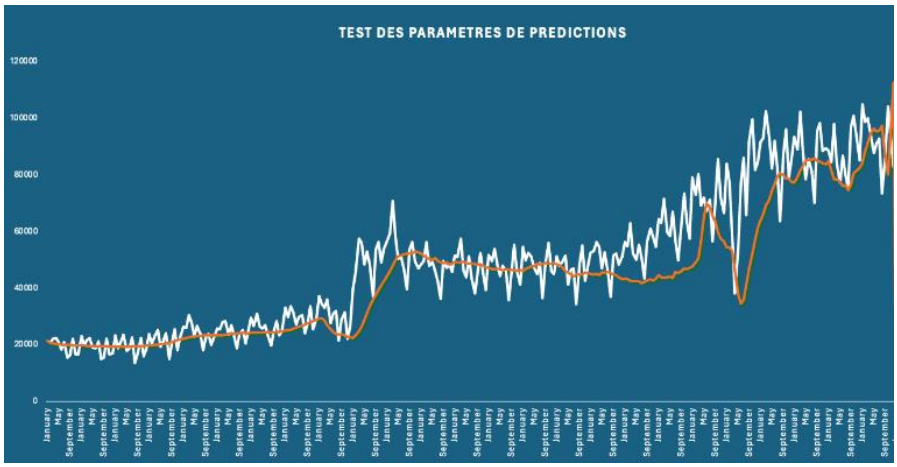
Pour affiner les prévisions et grâce aux graphiques précédent nous remarquons que nos données suivent une tendance positive ce qui nous à mener à appliquer un lissage exponentiel double, qui permet de modéliser des séries avec une tendance évolutive. Les étapes sont les suivantes :

1. **Création des colonnes pour le niveau et la tendance** :
   a. **Niveau (L)** : Représente la moyenne pondérée des valeurs actuelles et des niveaux précédents.
   b. **Tendance (T)** : Mesure la progression ou l'évolution de la tendance.

2. **Formules utilisées** :

$$L_t = \propto . Y_t + (1 - \propto) . (L_{t-1} + T_{t-1})$$
$$T_t = \beta . (L_t - L_{t-1}) + (1 - \beta) . T_{t-1}$$

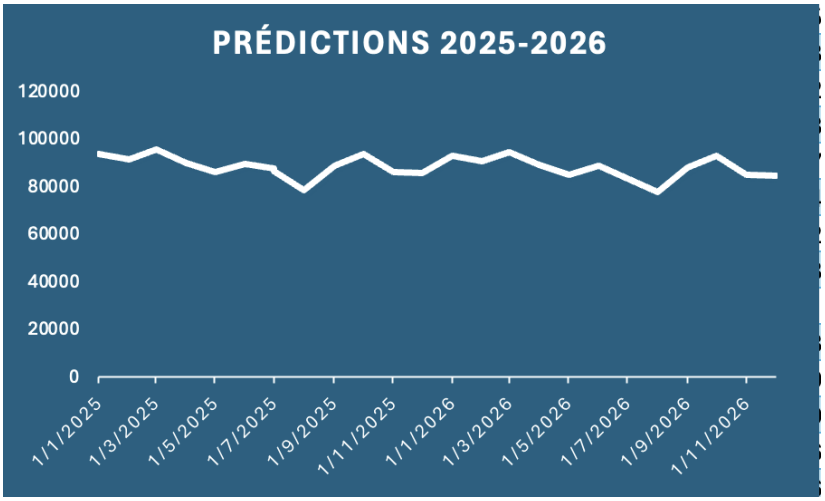
   où α=0,1 et β=0,9 ont été choisis graphiquement pour minimiser l'écart entre les observations et les prévisions comme montré dans le graphique ci-dessous qui représente la courbe d'observation réelle en blanc et la courbe de prévision (sur des données observées) en orange réalisée avec ces deux paramètres pour tester leur efficacité.



3. **Calcul des prévisions** : Les prévisions ont été calculées selon la formule :

$$P_{t+h} = L_t + h . T_t$$

   Où h est le pas par rapport à l'indice de la dernière observation. Dans notre cas, nous avons 300 lignes dans la base de données. Pour faire la prédiction de la 301ème ligne on choisit h = 1 et pour la 302ème Ligne, on choisit h = 2 etc . Ces prévisions incluent à la fois le niveau et la tendance et ont permis de produire des résultats fiables pour 2025 et 2026.



Ce graphique, basé sur le lissage exponentiel double (LED), présente des prévisions pour les années 2025 et 2026, avec une intégration explicite de la saisonnalité et une modélisation de la tendance évolutive. Les fluctuations saisonnières sont bien visibles, montrant une correspondance étroite avec les données historiques. Cependant, cette méthode met davantage l'accent sur les détails locaux, avec des ajustements plus précis au fil du temps, notamment pour les périodes de transition entre les mois.

Conclusion

En résumé, cette analyse a permis de répondre aux trois questions principales du projet :

1. **Choix de la série chronologique** : La série des créations mensuelles d'entreprises a été sélectionnée pour sa pertinence économique et sa saisonnalité marquée.
2. **Série CVS et prévisions** : Les données ont été ajustées saisonnièrement, et des prévisions ont été réalisées à l'aide d'une courbe de tendance exponentielle et du lissage exponentiel double.
3. **Analyse de la saisonnalité** : Les composantes saisonnières ont été isolées, analysées et interprétées. Cela a mis en évidence des cycles récurrents, influencés par des facteurs économiques et administratifs.

Cette approche a fourni des résultats solides et des outils d'analyse pour une meilleure compréhension des dynamiques des créations d'entreprises en France.

Partie 2 :

Dans cette partie, nous allons utiliser le modèle ARIMA pour faire des comparaisons entre des valeurs réelles et des valeurs prévisionnelles. Voici donc l'ensemble du code Python que nous avons réalisé suivi des commentaires explicatifs permettant la compréhension des différentes étapes du projet :

Entrée [2]:
```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)


path = '/Users/almabompard/Downloads/dataset.csv'

df = pd.read_csv(path)

df.head()
```

Out[2]:

	date	value
0	1991-07-01	3.526591
1	1991-08-01	3.180891
2	1991-09-01	3.252221
3	1991-10-01	3.611003
4	1991-11-01	3.565869

Ce code Python utilise Pandas, une librairie Python, pour charger et manipuler des données depuis un fichier CSV. Le chemin spécifié mène à un fichier contenant deux colonnes : date

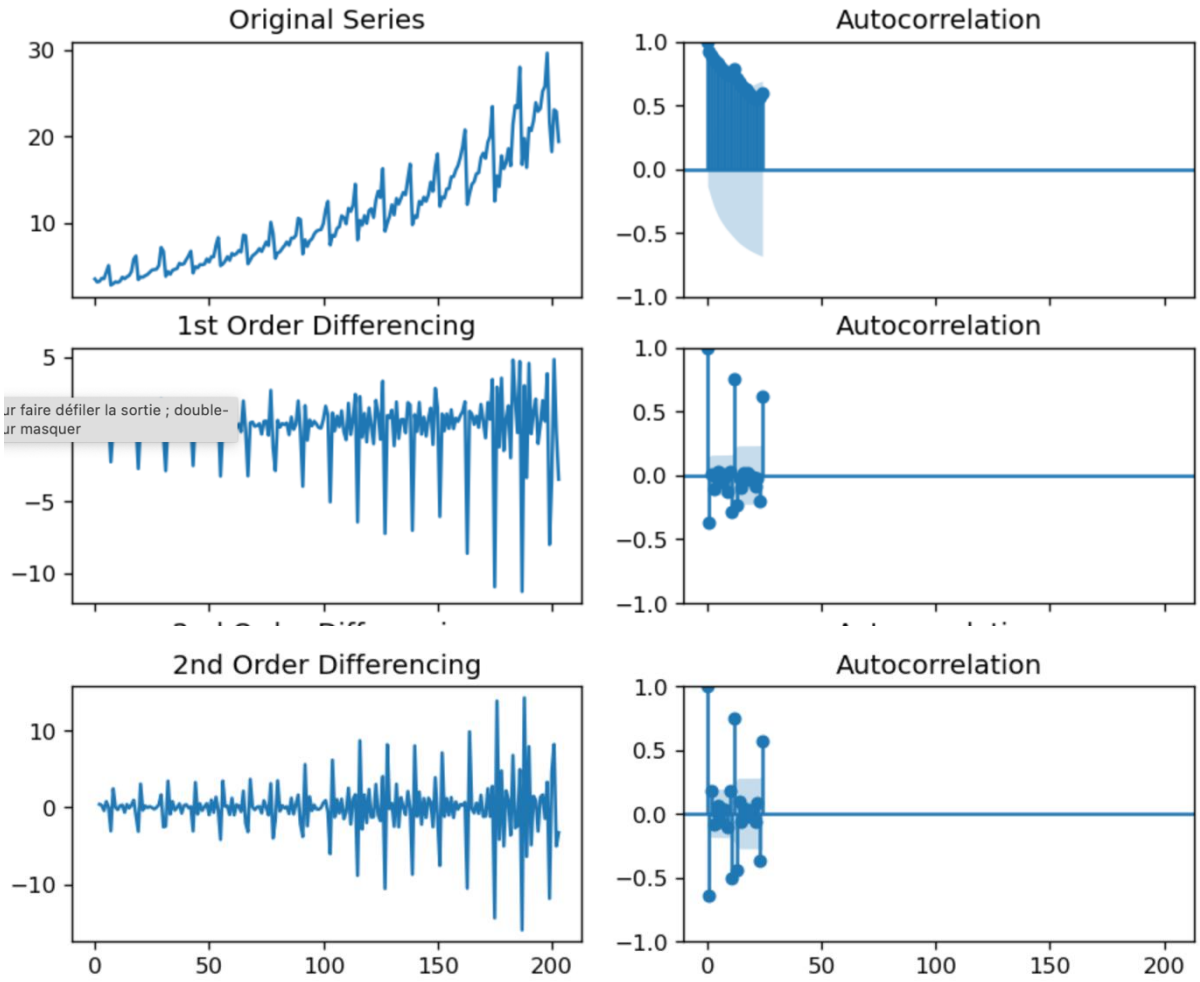
(dates au format YYYY-MM-DD) et value (valeurs numériques). La commande df.head() affiche les cinq premières lignes, révélant une série chronologique.

Entrée [8]:
```
from statsmodels.tsa.stattools import adfuller
from numpy import log
from statsmodels.graphics.tsaplots import plot_pacf

result = adfuller(df.value.dropna())
print('ADF Statistic: %f' % result[0])
print('p-value: %f' % result[1])
```

```
ADF Statistic: 3.145186
p-value: 1.000000
```

Ce code réalise un test d'Augmented Dickey-Fuller (ADF) pour évaluer la stationnarité de la série chronologique value. La méthode dropna() supprime les valeurs manquantes avant le test. Le résultat affiche une statistique ADF de 3.145186 et une p-value de 1.0, indiquant que la série n'est pas stationnaire, car l'hypothèse nulle (non-stationnarité) ne peut pas être rejetée.

Entrée [11]:
```
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf
import matplotlib.pyplot as plt
plt.rcParams.update({'figure.figsize':(9,7), 'figure.dpi':120})


# Original Series
fig, axes = plt.subplots(3, 2, sharex=True)
axes[0, 0].plot(df.value); axes[0, 0].set_title('Original Series')
plot_acf(df.value, ax=axes[0, 1])



# 1st Differencing
axes[1, 0].plot(df.value.diff()); axes[1, 0].set_title('1st Order Differencing')
plot_acf(df.value.diff().dropna(), ax=axes[1, 1])

# 2nd Differencing
axes[2, 0].plot(df.value.diff().diff()); axes[2, 0].set_title('2nd Order Differencing')
plot_acf(df.value.diff().diff().dropna(), ax=axes[2, 1])

plt.show()
```



Ce document analyse la série temporelle `value` en visualisant successivement la série originale, la première différenciation, la deuxième différenciation, ainsi que leurs fonctions d'autocorrélation (ACF). L'objectif est de déterminer le niveau de différenciation nécessaire pour stabiliser la série et la rendre stationnaire, une condition essentielle pour de nombreux modèles de séries temporelles.

1. **Série originale**
   La série montre une forte tendance croissante, indiquant une non-stationnarité. Cela est confirmé par la fonction d'autocorrélation (ACF), où les valeurs restent élevées et décroissent lentement.
2. **Première différenciation**
   En soustrayant chaque valeur par la précédente (fonction `diff()`), la tendance est atténuée, rendant la série plus stable. Toutefois, l'ACF montre encore des corrélations significatives, ce qui indique que la série n'est pas complètement stationnaire.
3. **Deuxième différenciation**
   Une deuxième différenciation (fonction `diff().diff()`) stabilise davantage la série. L'ACF est désormais concentrée autour de zéro, ce qui montre que la série est proche de la stationnarité.

Entrée [12]:

```python
from statsmodels.tsa.arima.model import ARIMA

# 1,1,2 ARIMA Model
model = ARIMA(df.value, order=(1,1,2))
model_fit = model.fit()
print(model_fit.summary())
```

```
                               SARIMAX Results
==============================================================================
Dep. Variable:                  value   No. Observations:                  204
Model:                 ARIMA(1, 1, 2)   Log Likelihood                -424.570
Date:                Mon, 27 Jan 2025   AIC                            857.140
Time:                        11:13:33   BIC                            870.393
Sample:                             0   HQIC                           862.502
                                  - 204
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.4178      0.356      1.174      0.240      -0.280       1.115
ma.L1         -0.9546      0.377     -2.531      0.011      -1.694      -0.215
ma.L2          0.0969      0.272      0.356      0.722      -0.437       0.631
sigma2         3.8259      0.269     14.209      0.000       3.298       4.354
===================================================================================
Ljung-Box (L1) (Q):                   0.46   Jarque-Bera (JB):               135.61
Prob(Q):                              0.50   Prob(JB):                         0.00
Heteroskedasticity (H):               9.82   Skew:                            -0.80
Prob(H) (two-sided):                  0.00   Kurtosis:                         6.67
===================================================================================
```

Ce document présente l'implémentation d'un modèle ARIMA(1,1,2) pour modéliser la série temporelle `value`. Le modèle spécifie une composante autorégressive (AR) d'ordre 1, une différenciation d'ordre 1 pour stabiliser la série, et deux composantes de moyenne mobile (MA). Après ajustement à l'aide de la méthode `fit()`, le résumé des résultats fournit les coefficients estimés pour chaque paramètre ainsi que leurs significativités statistiques. Les p-values indiquent que `ma.L1` et `sigma2` sont significatifs ($p < 0.05$), tandis que `ar.L1` et `ma.L2` ne le sont pas.

Les critères **AIC** et **BIC**, utilisés pour comparer plusieurs modèles, affichent des valeurs respectives de **857.140** et **870.393**. Ces indicateurs peuvent être utilisés pour guider la sélection d'un modèle optimal.

En revanche, les tests statistiques inclus, tels que le test **Ljung-Box** pour vérifier l'indépendance des résidus et le test **Jarque-Bera** pour leur normalité, révèlent certains problèmes. Notamment, la probabilité associée au test de normalité (Prob(JB) = 0.00) montre que les résidus ne suivent pas une distribution normale, suggérant que le modèle pourrait ne pas capturer parfaitement la dynamique de la série.

Entrée [13]:
```
# 1,1,1 ARIMA Model
model = ARIMA(df.value, order=(1,1,1))
model_fit = model.fit()
print(model_fit.summary())
```

```
                               SARIMAX Results
==============================================================================
Dep. Variable:                  value   No. Observations:                  204
Model:                 ARIMA(1, 1, 1)   Log Likelihood                -424.762
Date:                Mon, 27 Jan 2025   AIC                            855.524
Time:                        11:13:35   BIC                            865.463
Sample:                             0   HQIC                           859.545
                                  - 204
Covariance Type:                  opg
==============================================================================
                 coef    std err          z      P>|z|      [0.025      0.975]
------------------------------------------------------------------------------
ar.L1          0.3009      0.094      3.195      0.001       0.116       0.485
ma.L1         -0.8300      0.048    -17.204      0.000      -0.925      -0.735
sigma2         3.8327      0.259     14.790      0.000       3.325       4.341
===================================================================================
Ljung-Box (L1) (Q):                   0.72   Jarque-Bera (JB):               130.26
Prob(Q):                              0.40   Prob(JB):                         0.00
Heteroskedasticity (H):               9.98   Skew:                            -0.75
Prob(H) (two-sided):                  0.00   Kurtosis:                         6.63
===================================================================================
```

Ce document présente l'implémentation d'un modèle ARIMA(1,1,1) pour modéliser la série temporelle `value`. Ce modèle utilise une composante autorégressive (AR) d'ordre 1, une différenciation d'ordre 1 pour stabiliser la série, et une composante de moyenne mobile (MA) d'ordre 1. Les résultats montrent que les paramètres estimés, notamment `ar.L1` (0.3009), `ma.L1` (-0.8300) et `sigma2` (3.8327), sont significatifs, avec des p-values inférieures à 0.05, ce qui indique leur pertinence dans le modèle.

Les critères **AIC** (855.524) et **BIC** (865.463) suggèrent que ce modèle est légèrement meilleur que le modèle ARIMA(1,1,2) testé précédemment.

Cependant, les tests sur les résidus révèlent quelques limites : le test **Jarque-Bera** montre une probabilité nulle (Prob(JB) = 0.00), indiquant une non-normalité des résidus, bien que le test de **Ljung-Box** (Prob(Q) = 0.72) confirme l'indépendance des résidus.

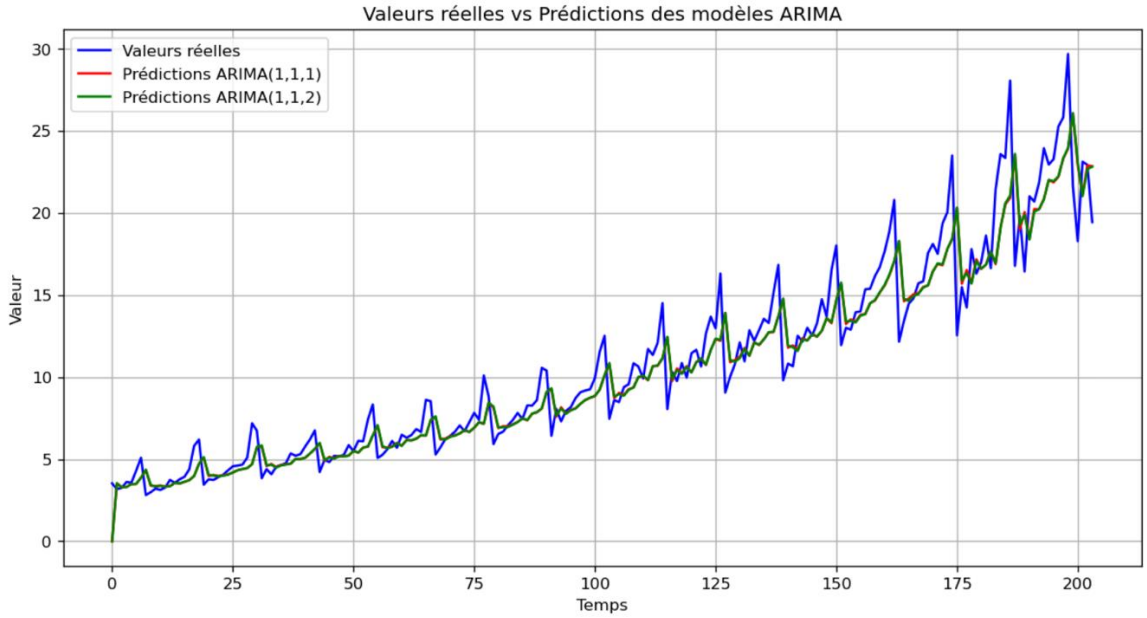
Entrée [14]:
```
# Plot residual errors
residuals = pd.DataFrame(model_fit.resid)
fig, ax = plt.subplots(1,2)
residuals.plot(title="Residuals", ax=ax[0])
residuals.plot(kind='kde', title='Density', ax=ax[1])
plt.show()
```

Ce code visualise les résidus du modèle ARIMA ajusté pour évaluer la qualité de l'ajustement. Deux graphiques sont générés : le premier illustre les résidus au fil du temps, tandis que le second montre leur densité estimée à l'aide d'une fonction noyau (KDE).

Le graphique des résidus met en évidence une variabilité relativement constante autour de zéro, bien qu'il soit possible de noter certains pics inhabituels qui pourraient indiquer des

anomalies ou des limitations dans le modèle. Le graphique de densité montre que les résidus sont globalement centrés sur zéro, mais la forme de la courbe suggère une légère asymétrie, ce qui confirme les résultats du test Jarque-Bera indiquant une non- normalité des résidus.

Entrée [15]:
```python
import matplotlib.pyplot as plt
from statsmodels.tsa.arima.model import ARIMA

# Modèle ARIMA(1,1,1)
model_111 = ARIMA(df.value, order=(1,1,1))
model_fit_111 = model_111.fit()


# Modèle ARIMA(1,1,2)
model_112 = ARIMA(df.value, order=(1,1,2))
model_fit_112 = model_112.fit()

# Prédictions pour les deux modèles
predictions_111 = model_fit_111.predict(start=0, end=len(df['value']) - 1, dynamic=False)
predictions_112 = model_fit_112.predict(start=0, end=len(df['value']) - 1, dynamic=False)

# Tracer les valeurs réelles et les prédictions
plt.figure(figsize=(12, 6))
plt.plot(df['value'], label="Valeurs réelles", color="blue")
plt.plot(predictions_111, label="Prédictions ARIMA(1,1,1)", color="red")
plt.plot(predictions_112, label="Prédictions ARIMA(1,1,2)", color="green")

# Ajouter des légendes et des titres
plt.title("Valeurs réelles vs Prédictions des modèles ARIMA")
plt.xlabel("Temps")
plt.ylabel("Valeur")
plt.legend()
plt.grid()
plt.show()
```



Ce graphique illustre une comparaison entre les valeurs réelles et les prédictions obtenues à l'aide de deux modèles ARIMA, à savoir ARIMA(1,1,1) et ARIMA(1,1,2). Les valeurs réelles, représentées en bleu, montrent une tendance générale à la hausse avec des variations saisonnières marquées par des pics et des creux réguliers. Les prédictions des deux modèles, en rouge pour ARIMA(1,1,1) et en vert pour ARIMA(1,1,2), suivent globalement cette tendance et capturent les fluctuations des données.

Les deux modèles semblent très proches en termes de performance, ce qui suggère que leur capacité à s'ajuster aux données est équivalente. Ils reproduisent efficacement les variations autour de la moyenne et dans les périodes plus stables. Cependant, des écarts légèrement plus importants

apparaissent lors des pics élevés, notamment vers la fin de la série (entre 175 et 200), où les valeurs réelles dépassent parfois légèrement les prédictions. Cela indique que les modèles pourraient sous-estimer ou avoir du mal à capturer certaines variations extrêmes.

En conclusion, les modèles ARIMA(1,1,1) et ARIMA(1,1,2) sont bien adaptés pour prédire la tendance et la saisonnalité des données, mais pourraient être optimisés pour mieux anticiper les fluctuations extrêmes. Une analyse statistique complémentaire, comme le calcul d'indicateurs de performance (ex. RMSE ou MAPE), permettrait de confirmer leur efficacité et de comparer leurs performances plus précisément.