

Projet Série Temporelle

Kuassi Pierre DOVODJI - Lucas Debauche

2024-12-18

Contents

1	Introduction	2
2	Objectifs	2
3	Description de la base de données	2
3.1	Statistique descriptive	3
4	Identification du modèle	4
4.1	Analyse de la série	4
4.2	Modele ARIMA	7
4.3	Le modèle SARIMA	15
4.4	Résumé des modèles	22
5	Conclusion	22
6	Annexe : Code Source	23

1 Introduction

Les exportations de viande et de produits carnés sont un pilier de l'économie agroalimentaire française, notamment pour le bœuf, le porc et la volaille. Cependant, elles sont influencées par divers facteurs comme la demande mondiale, les réglementations sanitaires et les tendances de consommation.

2 Objectifs

L'objectif principal de ce projet est d'appliquer les techniques de prévision des séries temporelles pour anticiper l'évolution des exportations de viande et de produits carnés en France. Pour cela nous allons :

- Analyser les tendances passées des exportations à partir des données historiques.
- Identifier les effets saisonniers qui influencent les exportations.
- Comparer différentes méthodes de prévision : **ARIMA**, **SARIMA**.
- Évaluer la précision des modèles avec l'AIC.

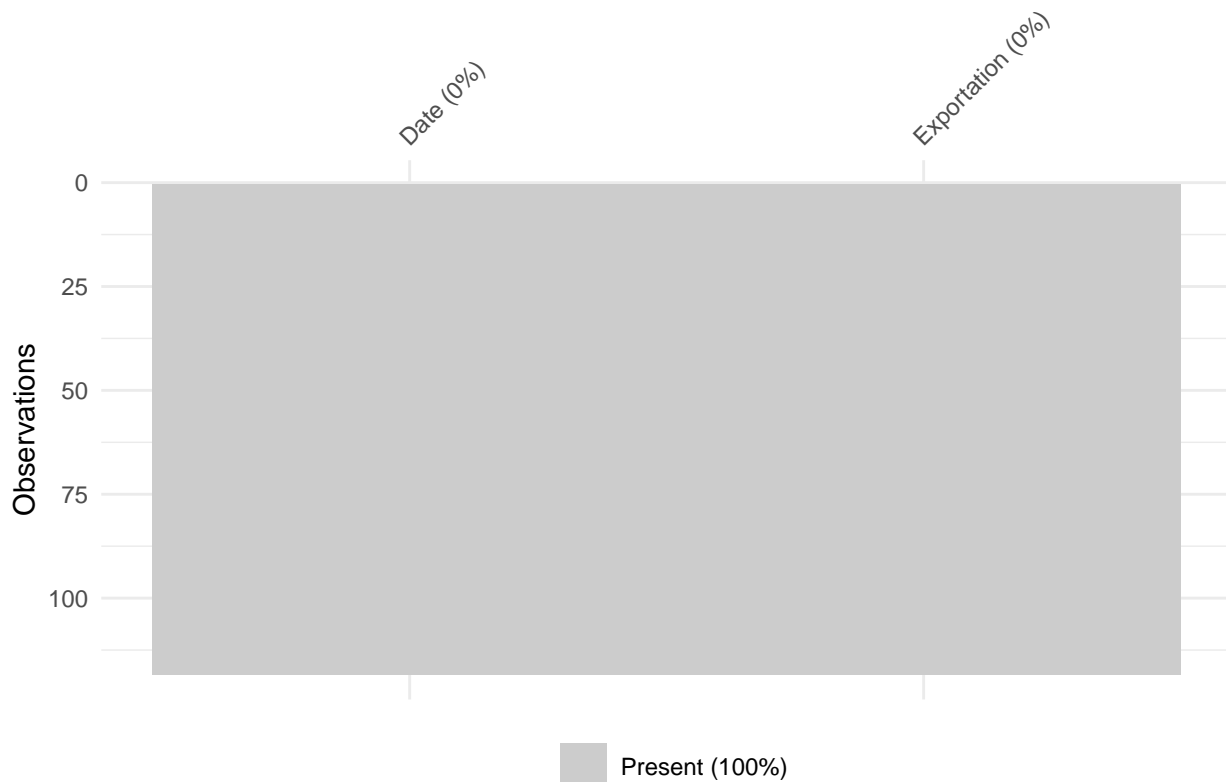
3 Description de la base de données

La base de données utilisée concerne les exportations de viande et de produits carnés en France. Il s'agit d'une série temporelle classique qui représente les volumes mensuels d'exportations.

La base de données sur les exportations de viande et de produits carnés contient 118 observations et deux variables. Les variables sont:

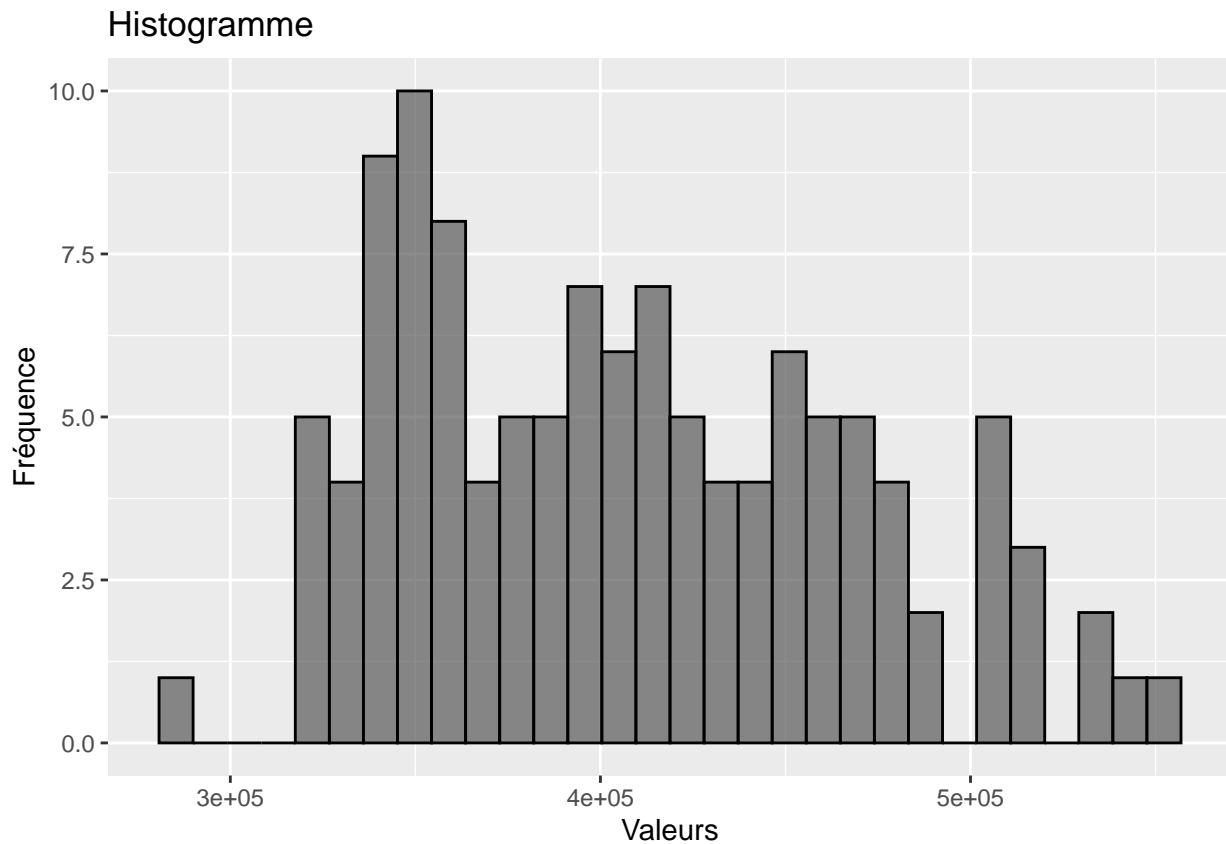
- **Date:** La date (mois et année) de chaque observation.
- **Exportation:** Le volume des exportations en tonne de viande et de produits carnés pour chaque mois.

Les données couvrent la période de janvier 2015 à octobre 2024, soit presque 10 années de données mensuelles.



On constate que la base de données ne contient pas de valeur manquante.

3.1 Statistique descriptive



Les exportations ne présentent pas une distribution naturelle.

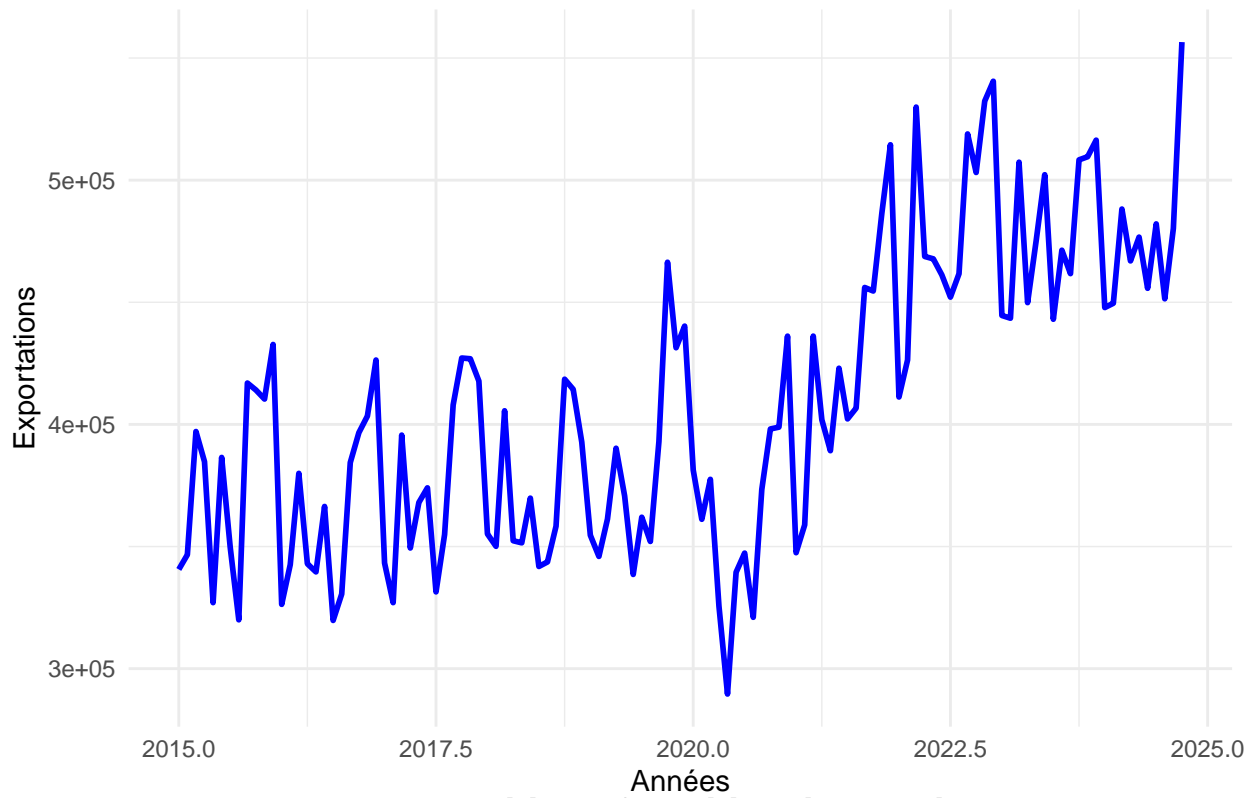
##	Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
##	289602	354708	402088	408123	451938	556534

Les données montrent une répartition asymétrique, avec des valeurs majoritairement situées entre 354708 et 451938, mais une grande variabilité due à des valeurs extrêmes. Cette variabilité pourrait être liée à des fluctuations saisonnières ou conjoncturelles, influencées par les cycles de production ou la demande internationale. La valeur maximale élevée (556,534) pourrait également correspondre à un événement exceptionnel, tel qu'une forte demande ou un changement réglementaire. Cependant, la tendance générale reste positive, comme le suggèrent la médiane et la moyenne relativement élevées, indiquant une activité exportatrice significative sur l'ensemble de la période. Puisque le volume d'exportation est grande nous utiliserons dans la suite de l'étude le logarithme de la série.

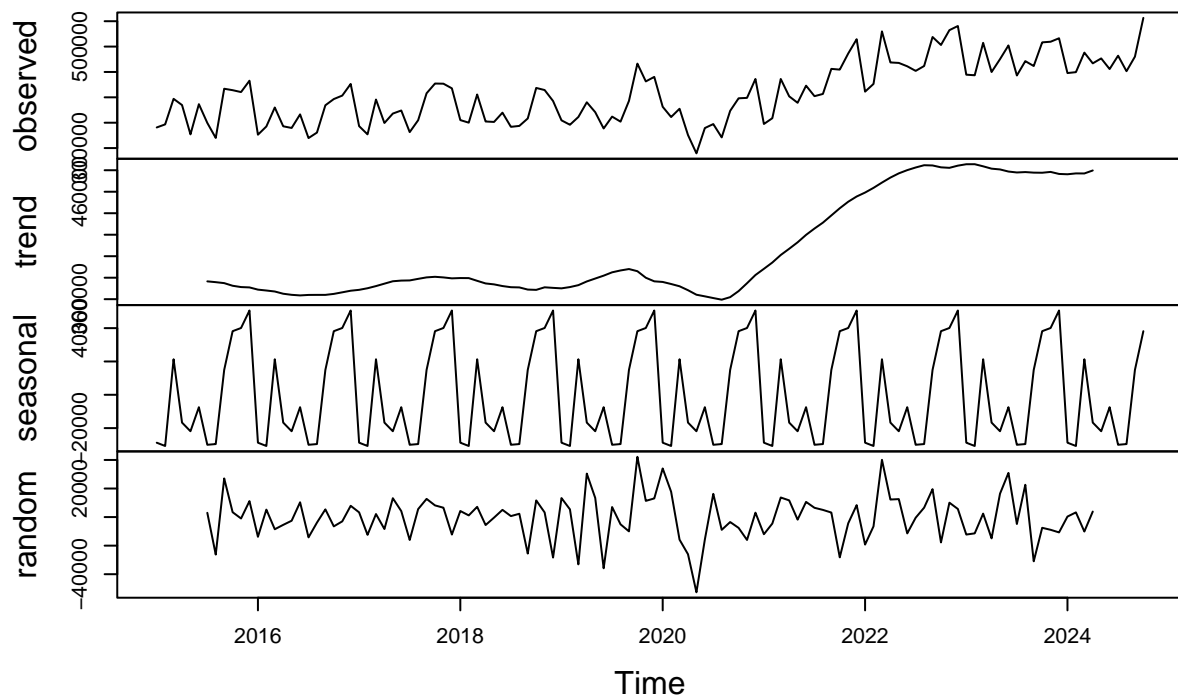
4 Identification du modèle

4.1 Analyse de la série

Exportations de viande et produits carnés en France



Decomposition of additive time series

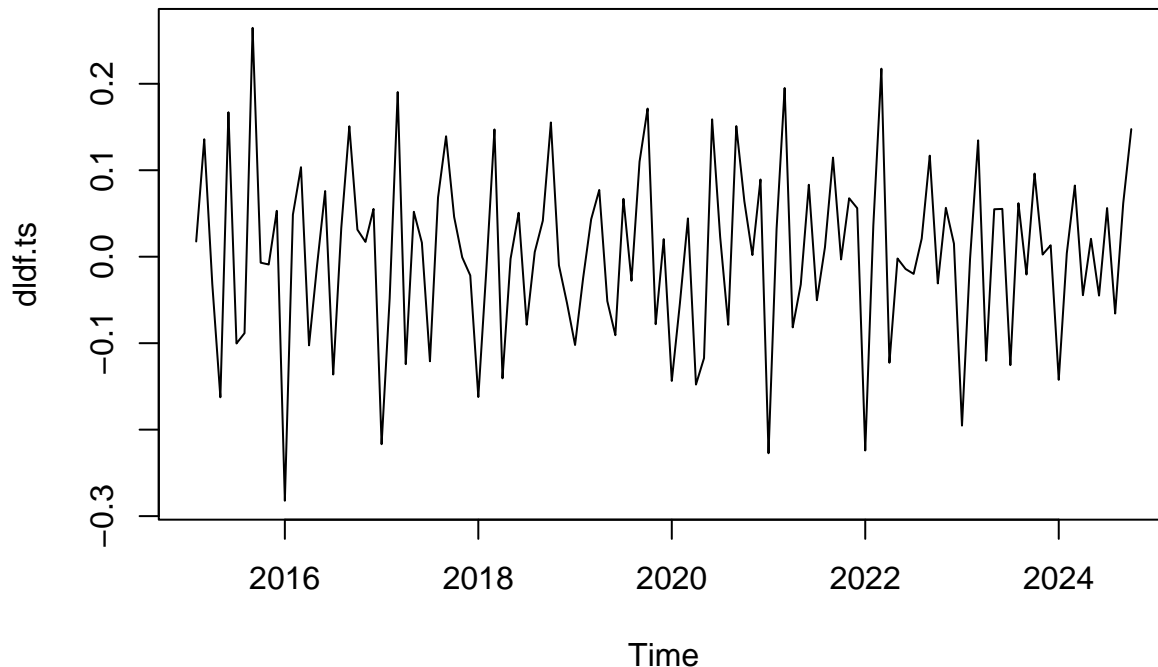


Entre 2015 et 2019, la tendance est relativement stable, montrant peu de croissance. À partir de 2020, une augmentation marquée est observable. La variabilité semble augmenter légèrement avec le temps, ce qui pourrait indiquer une hétéroscédasticité et une cyclicité saisonnière est visible, probablement annuelle. On peut déduire que notre série n'est pas stationnaire. Pour confirmer cela nous allons effectuer le test statistique de Dickey-Fuller augmenté.

Métrique	Valeur
Valeur de test	-2.787
P-valeur	0.25
Conclusion	On ne peut pas affirmer que la série est stationnaire

Le test confirme que la série est non stationnaire. Nous allons à présent transformer et différencier la série pour obtenir une série stationnaire.

- Dans un premier temps nous allons étudier la stationnarité de la série qui est le log de notre série initiale.



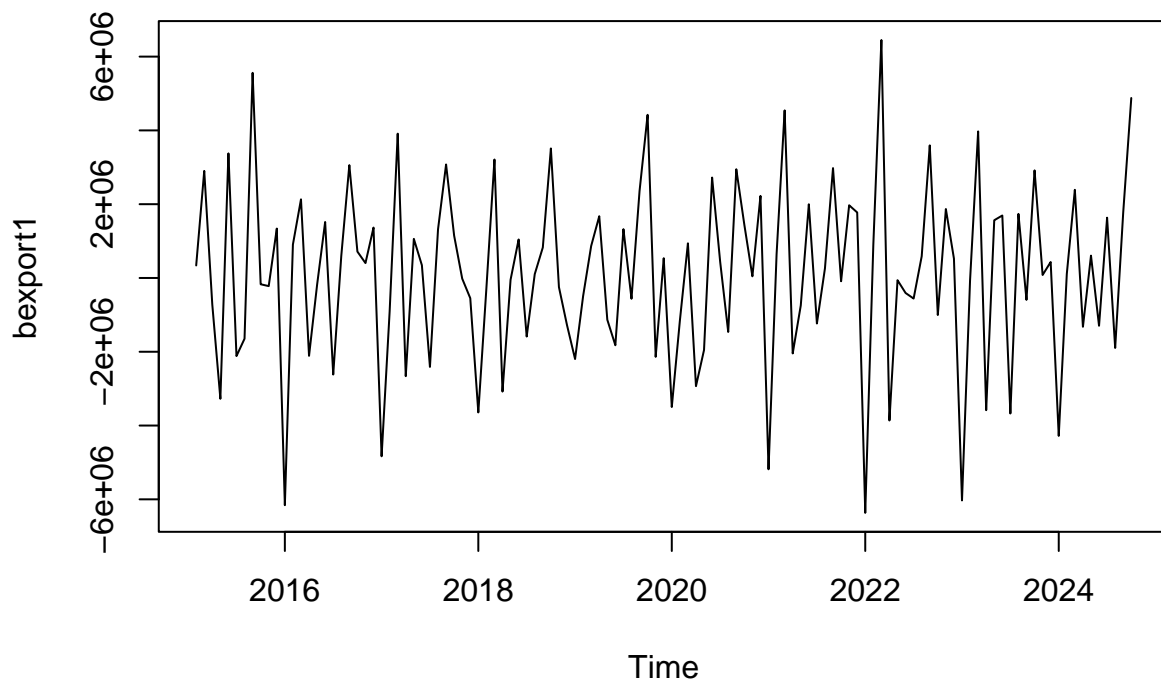
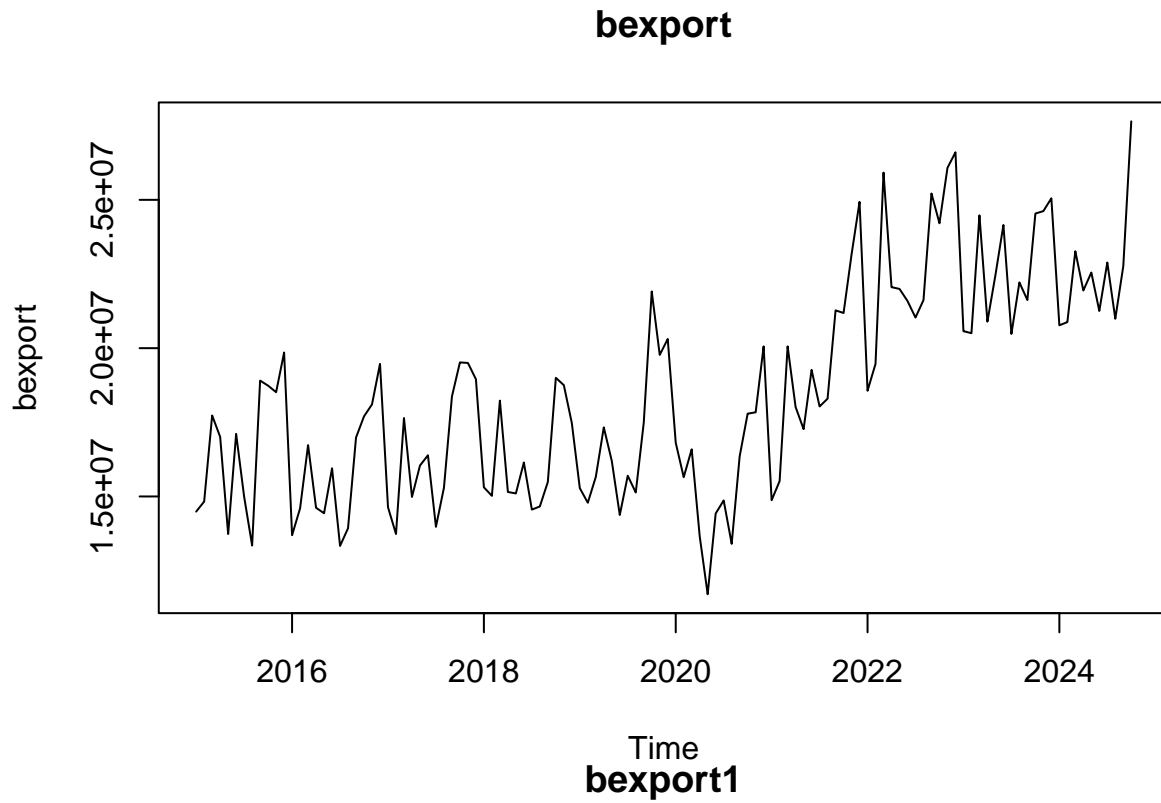
La moyenne de la nouvelle série obtenue après transformation est 0.004196346

Ce graphique nous montre la nouvelle série différenciée d'ordre 1 qui semble stationnaire, car elle oscille autour d'une moyenne constante quasi nulle et ne montre pas de tendance ni de saisonnalité évidente. Cette observation sera confirmée à l'aide d'un test statistique comme le test de Dickey-Fuller augmenté (ADF).

Métrique	Valeur
Valeur de test	-7.425
P-valeur	0.01
Conclusion	La série est stationnaire

Le test de Dickey-Fuller augmenté, avec une valeur de test de -7.425 et une p-valeur de 0.01, confirme que la série différenciée est stationnaire. Donc on prend $d = 1$

- Nous allons maintenant essayer la transformation de BoxCox.



La moyenne de la nouvelle série obtenue après transformation de BoxCox et différencié est: 112457.7

La méthode ci dessus doit nous donner la transformation optimale (pas forcément le log), la transformation est de la forme :

$$f_{\lambda}(x) = \frac{x^{\lambda}-1}{\lambda}, \lambda > 0, x > 0$$

On obtient également une série qui semble stationnaire.

Nous allons donc utiliser deux modèles, dans un premier temps nous utiliserons un modèle ARIMA sur les deux transformations de notre série et dans un second temps le modèle SARIMA.

4.2 Modèle ARIMA

Le modèle ARIMA (AutoRegressive Integrated Moving Average) est une méthode couramment utilisée pour modéliser et prévoir les séries temporelles. Il combine les composantes de l'autorégression (AR) et de la moyenne mobile (MA) avec une différenciation (I) pour prendre en compte les tendances et les comportements saisonniers dans les données.

4.2.1 Composantes du modèle ARIMA

- **AR (Autorégression)** : L'AR fait référence à la régression linéaire des valeurs actuelles sur les valeurs précédentes de la série temporelle. Il capture les effets de dépendance linéaire à partir des valeurs passées.
- **MA (Moyenne mobile)** : Le MA utilise la moyenne mobile des erreurs précédentes pour modéliser la relation entre les résidus et les observations actuelles de la série temporelle. Il capture les effets de dépendance entre les résidus.
- **I (Différenciation)** : La différenciation est utilisée pour rendre les données stationnaires en supprimant les tendances et les comportements saisonniers. Elle consiste à prendre la différence entre les observations consécutives jusqu'à obtenir une série stationnaire.

4.2.2 Ordres du modèle ARIMA

Le modèle ARIMA est défini par trois paramètres: p , d et q .

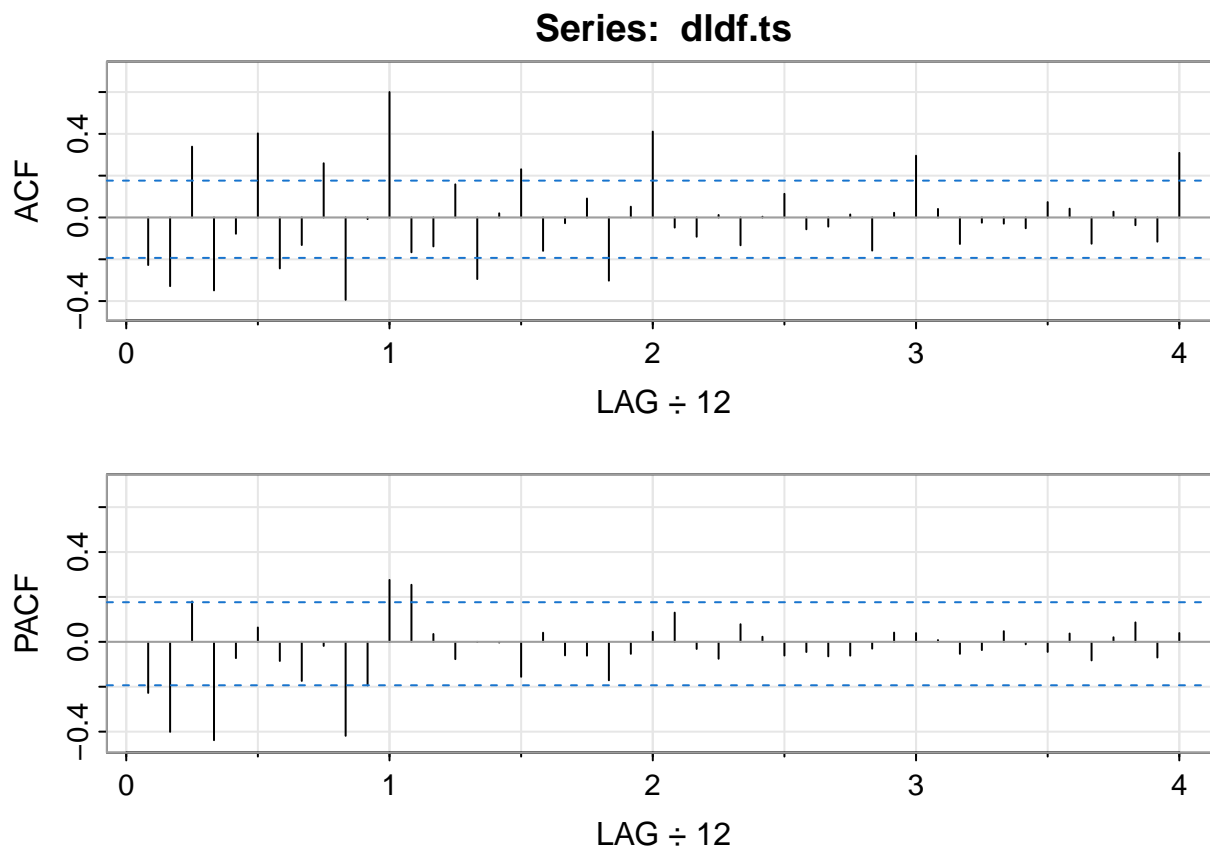
- **p (Ordre AR)** : L'ordre AR spécifie le nombre de termes autorégressifs à inclure dans le modèle. Il indique combien de valeurs passées sont utilisées pour prédire la valeur actuelle.
- **d (Ordre de différenciation)** : L'ordre de différenciation indique combien de fois la série doit être différenciée pour rendre les données stationnaires.
- **q (Ordre MA)** : L'ordre MA spécifie le nombre de termes de la moyenne mobile à inclure dans le modèle. Il indique combien de résidus passés sont utilisés pour prédire l'observation actuelle.

La méthodologie de Box-Jenkins, utilisée pour modéliser et prévoir les séries temporelles, suit trois grandes étapes :

- **Identification du modèle**: On analyse les données pour repérer tendances et saisonnalités, puis on applique une différenciation si nécessaire pour les rendre stationnaires. Les graphiques ACF et PACF aident à déterminer les paramètres p , d et q du modèle ARIMA.
- **Estimation** : On calcule les paramètres du modèle à l'aide de méthodes comme les moindres carrés.
- **Vérification** : On vérifie que les résidus (erreurs) du modèle ressemblent à un bruit blanc (sans structure ni corrélation). Si ce n'est pas le cas, on ajuste le modèle et on répète les étapes. Enfin, on valide le modèle en testant ses prédictions.

C'est une approche itérative pour trouver le modèle ARIMA le plus adapté, capable de capturer les caractéristiques importantes de la série temporelle et de fournir des prévisions fiables.

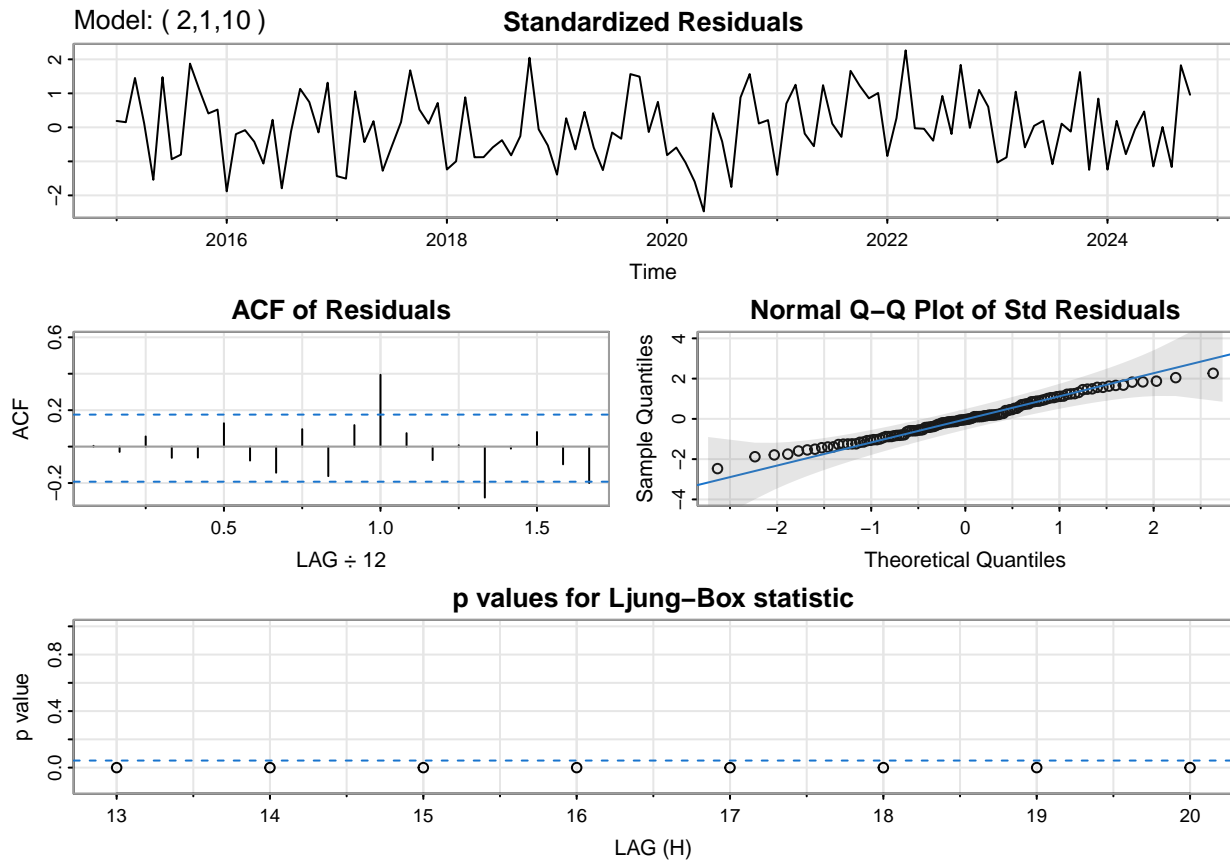
- Dans cette partie nous allons appliquer le modèle ARIMA sur la série transformée avec \log .
 - **d** : Nous avons obtenu une série stationnaire en appliquant une différenciation une seule fois sur la série transformée, ce qui implique que le paramètre de différenciation $d = 1$.
 - **p et q** Nous allons donc déterminer p et q à partir des autocorrélogrammes de la série différenciée.



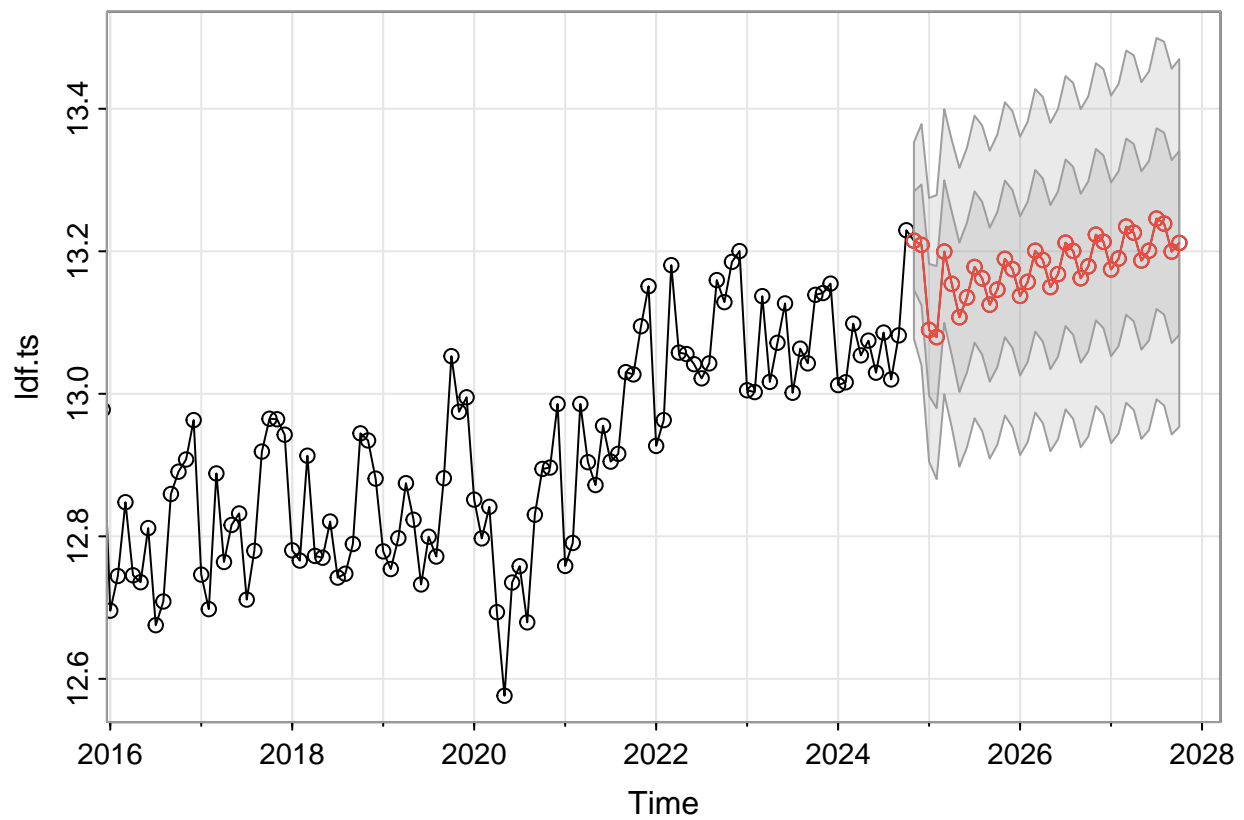
En analysant ce graphique, on remarque que :

- Le graphique ACF (autocorrélation) devient proche de 0 à partir du décalage (lag) 10, ce qui suggère que $q=10$.
- Le graphique PACF (autocorrélation partielle) devient proche de 0 à partir du décalage 2 ou 4, ce qui indique que $p = 2$ ou $p = 4$.

4.2.3 Modèle 1 : $p = 2, d = 1, q = 10$

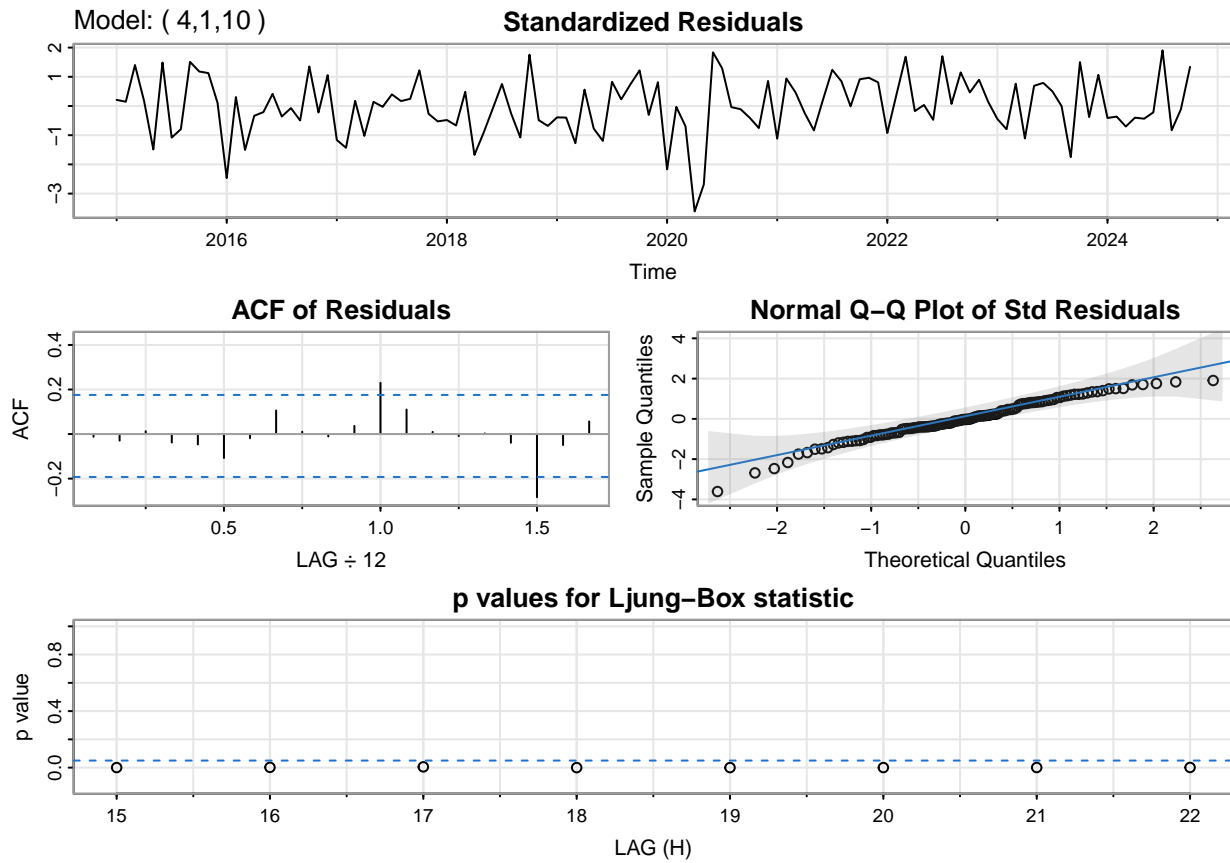


On voit que les résidus n'ont pas vraiment l'allure d'un bruit blanc. Les estimations des ρ_k sont presque tous significativement nuls donc c'est bien. Les résidus standardisés suivent une distribution normale. Les p-values pour la statistique de Ljung-Box sont toutes faibles donc on rejette H_0 , ce n'est pas idéal.



D'après le modèle, les valeurs devraient rester plutôt stables après 2024. Plus on avance dans le temps, plus l'incertitude augmente, ce qui est habituel en prévision.

4.2.4 Modèle 2 : $p = 4, d = 1, q = 10$

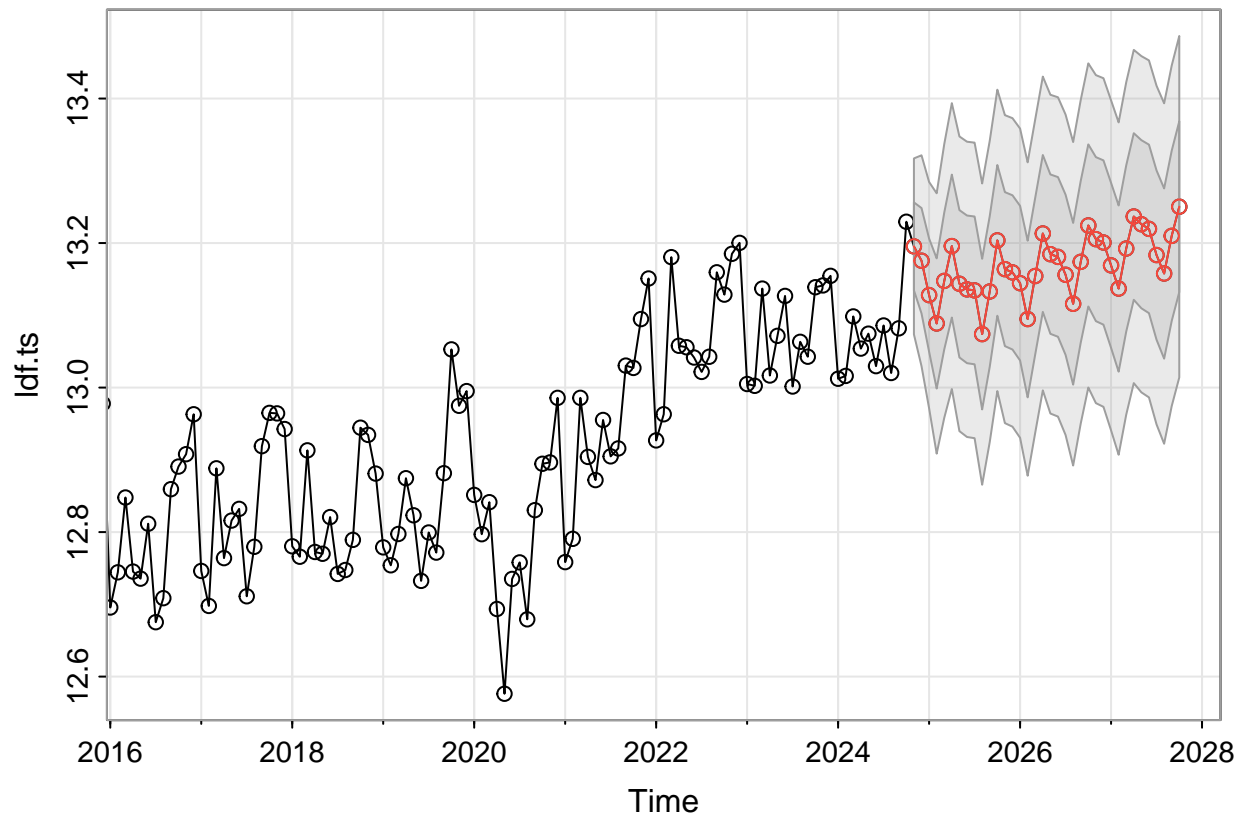


Les résidus ne présentent pas véritablement les caractéristiques d'un bruit blanc.

Les estimations des ρ_k sont presque toutes significativement nulles, ce qui est un bon signe.

Les résidus standardisés suivent une distribution normale.

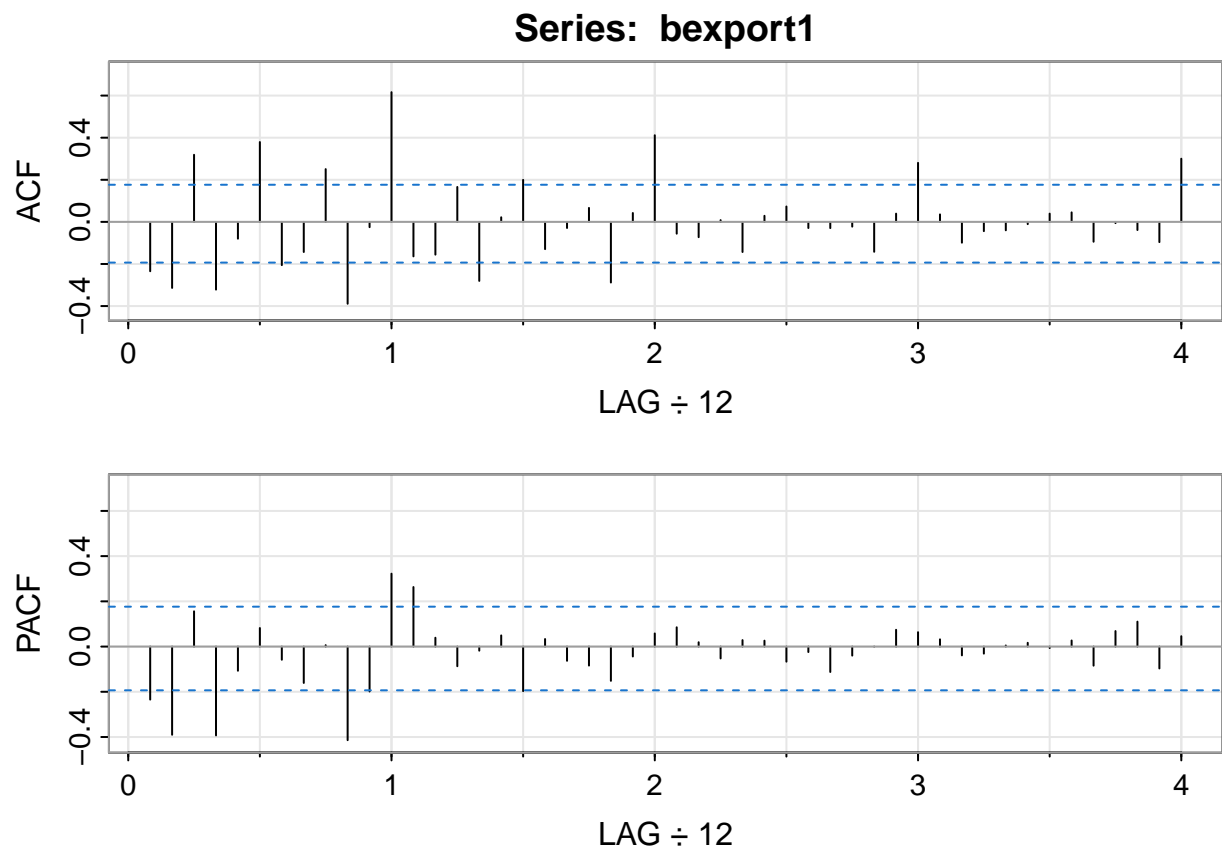
Cependant, les p-values de la statistique de Ljung-Box sont toutes faibles, ce qui conduit au rejet de l'hypothèse nulle, ce qui n'est pas optimal.



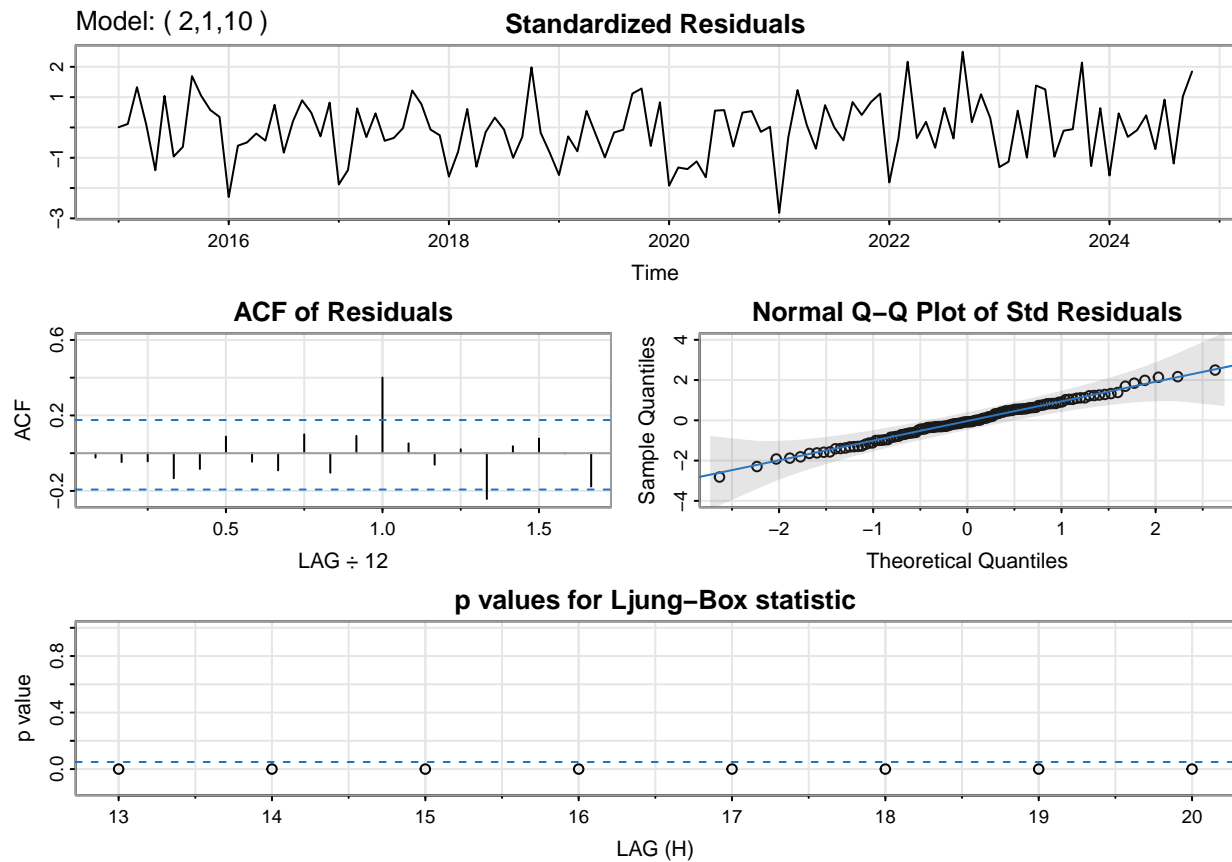
Le modèle prévoit que les valeurs resteront assez stables après 2024. L'incertitude augmente avec le temps, ce qui est normal pour une prévision. Il serait utile de comparer plusieurs modèles pour vérifier lequel donne les prévisions les plus fiables.

4.2.5 Modèle 3: Transformation BoxCox

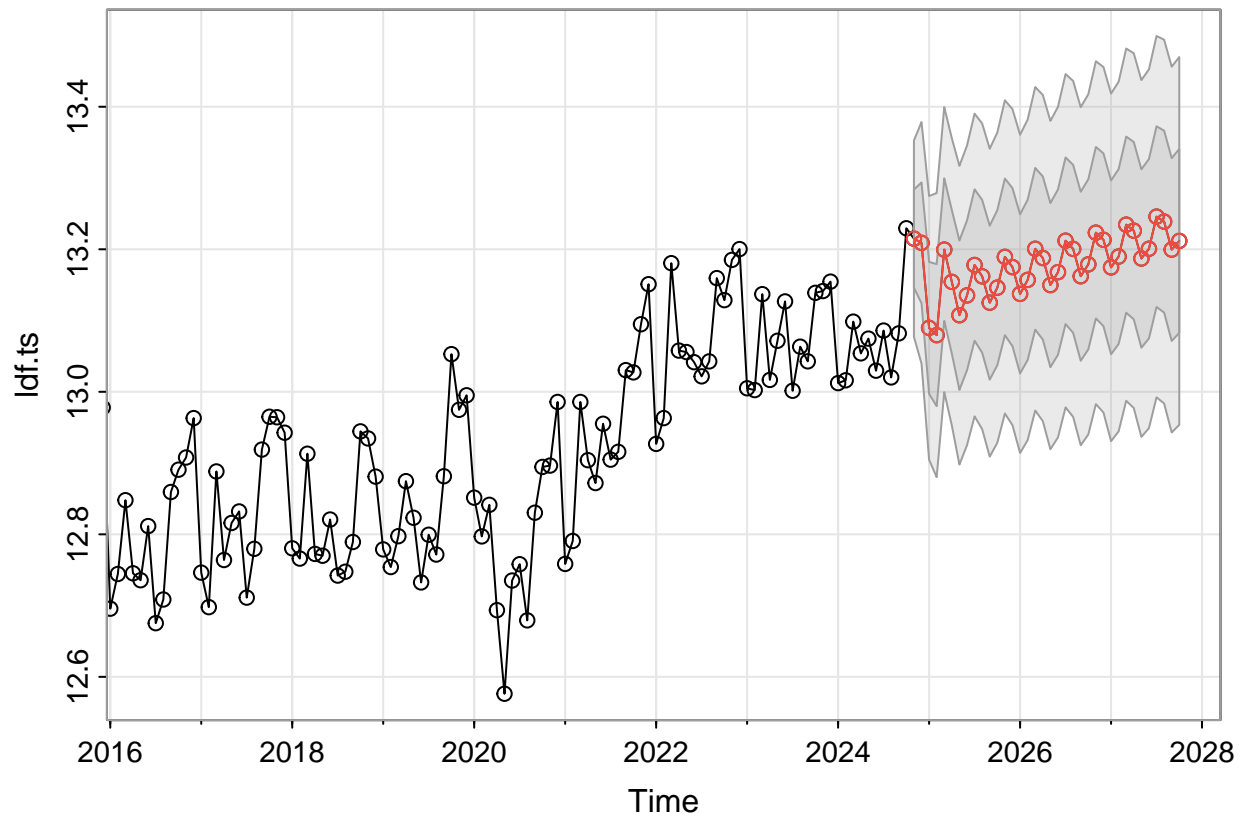
Déterminons les paramètres du modèle.



En analysant les graphiques d'autocorrélation et d'autocorrélation partielle, nous trouvons que les valeurs optimales pour les paramètres sont $\mathbf{p} = 2$ et $\mathbf{q} = 10$. En tenant compte de ce qui est fait précédemment sur la partie stationnarité, nous avons déterminé que le paramètre de différenciation est $\mathbf{d} = 1$.



Le commentaire reste le même, à l'exception du fait que les résidus semblent encore moins ressembler à un bruit blanc, et les résidus standardisés ne suivent pas vraiment une distribution normale.



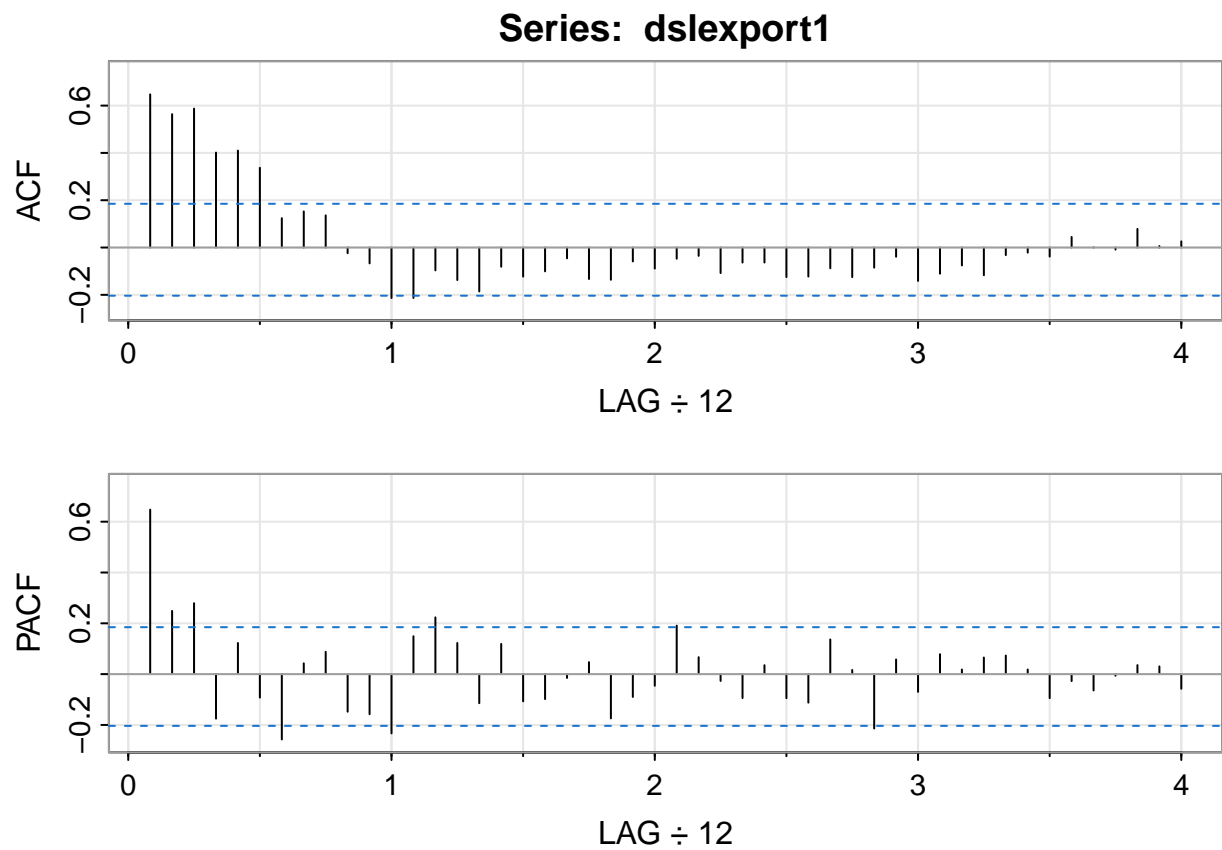
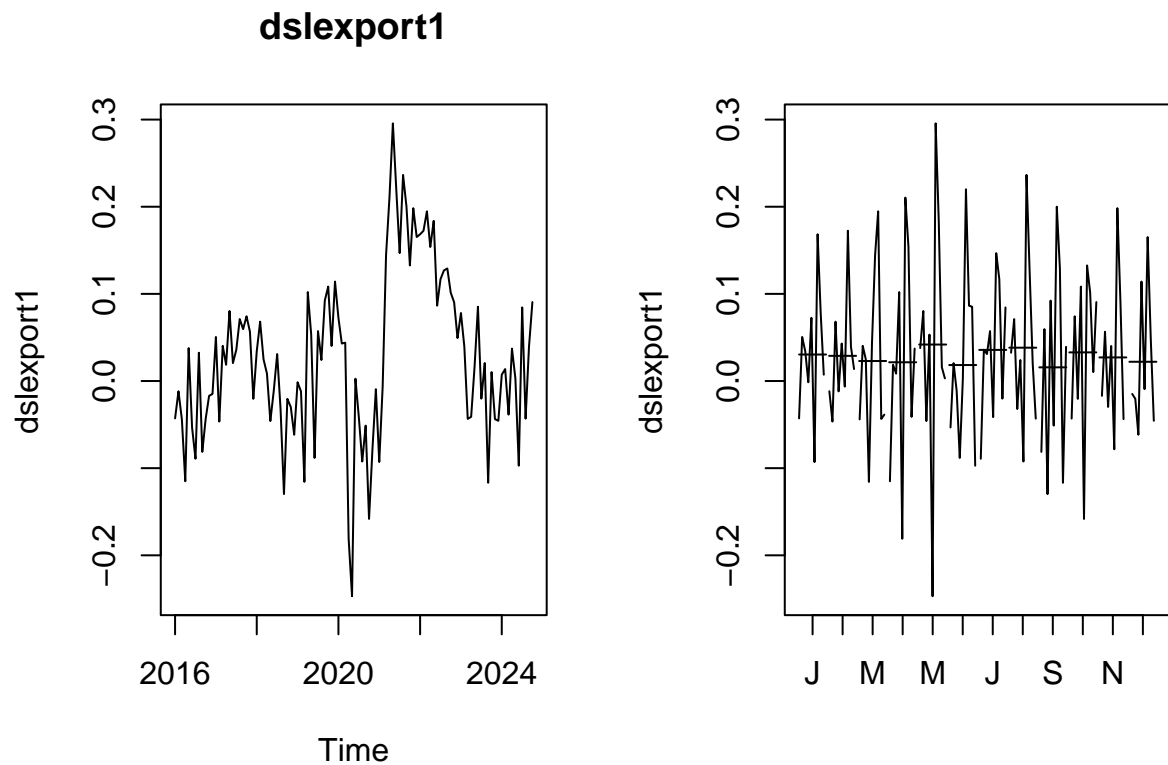
D'après le modèle, les valeurs devraient rester plutôt stables après 2024. Plus on avance dans le temps, plus l'incertitude augmente, ce qui est habituel en prévision.

4.3 Le modèle SARIMA

Le modèle SARIMA (Seasonal ARIMA) est une extension du modèle ARIMA qui inclut une composante saisonnière pour analyser les séries temporelles avec des motifs périodiques. Il utilise les paramètres saisonniers suivants en plus des paramètres du modèle ARIMA :

- q' : Ordre saisonnier de la moyenne mobile, déterminé à partir du graphique ACF saisonnier.
- p' : Ordre saisonnier autorégressif, obtenu grâce au graphique PACF saisonnier.
- d' : Différenciation saisonnière, utilisée pour rendre la série stationnaire sur les cycles saisonniers.

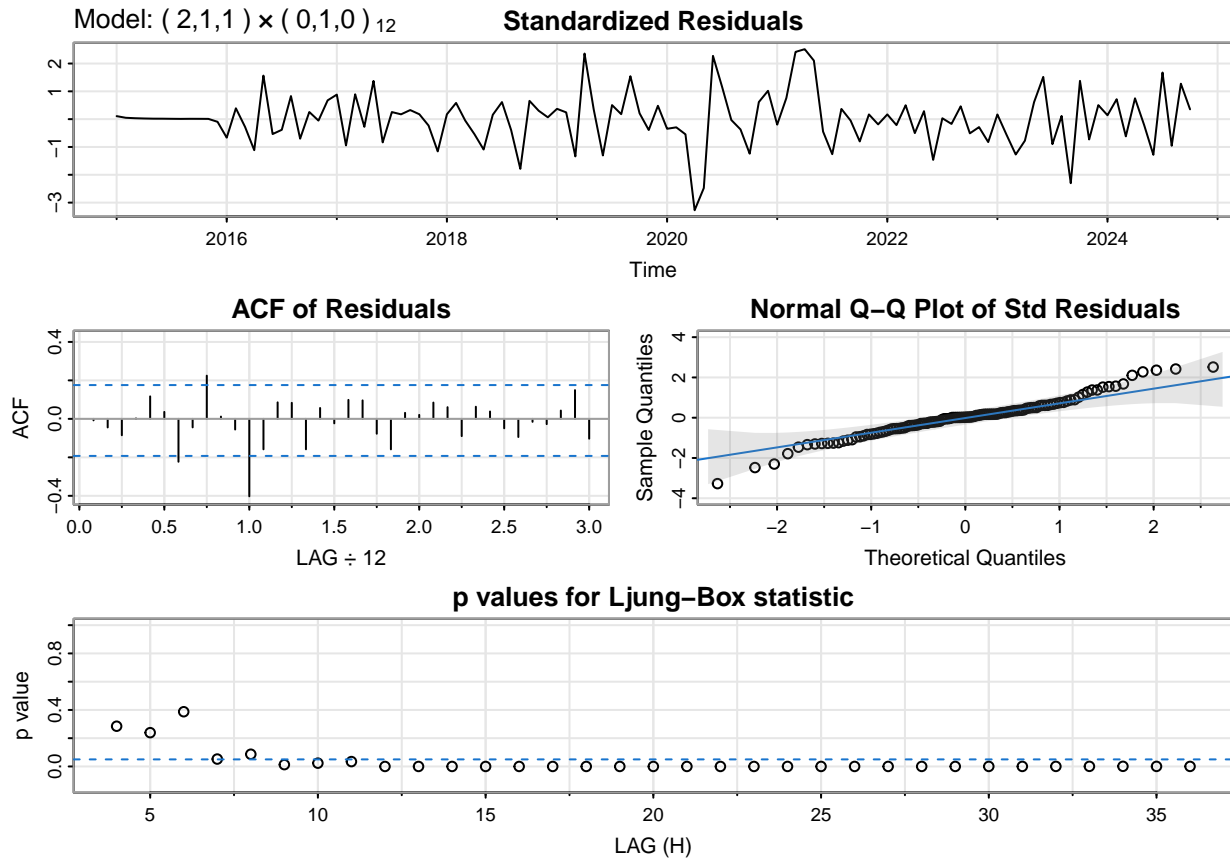
On va faire une différenciation saisonnière.



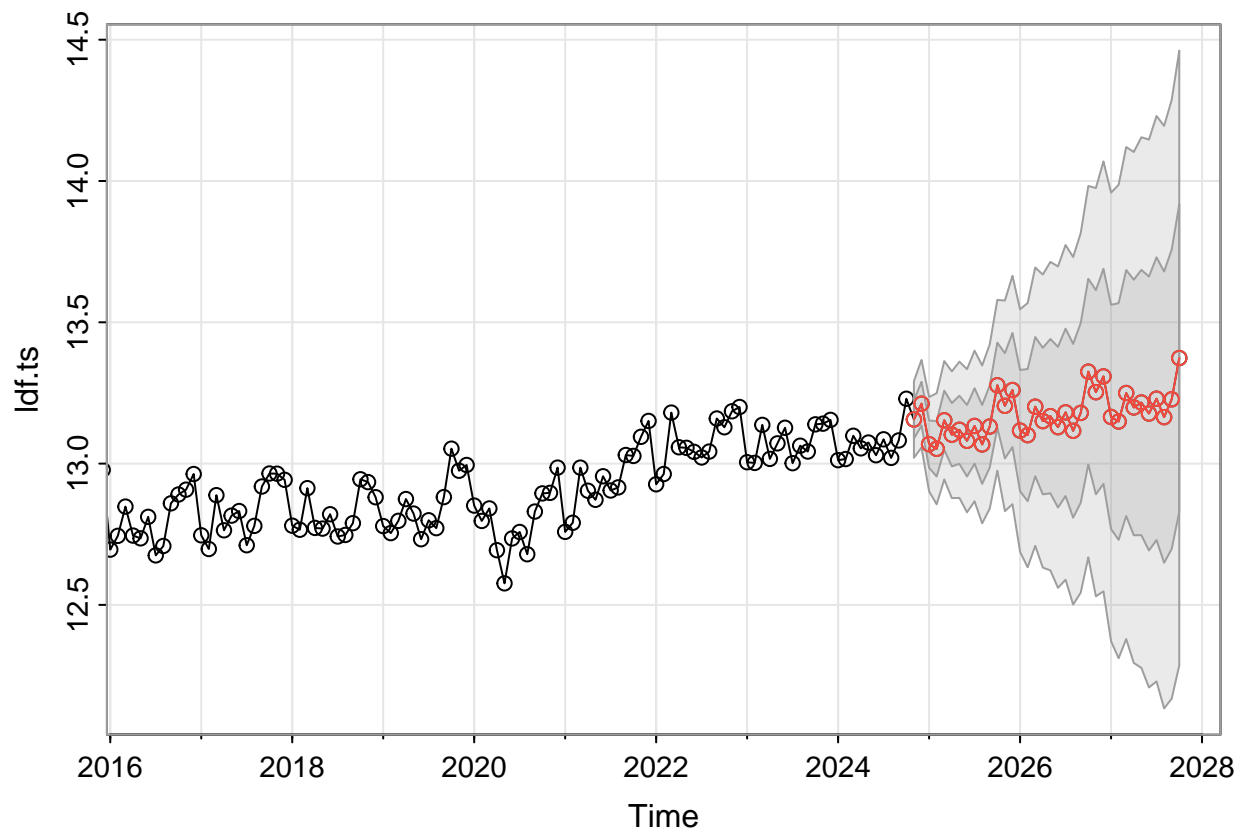
D'après l'analyse du graphique d'autocorrélation, on en déduit que le paramètre $q = 1$ et que $q' = 0$.

D'après l'analyse du graphique d'autocorrélation partielle, on obtient $p = 2$ et $p' = 0$.

4.3.1 Modele 4

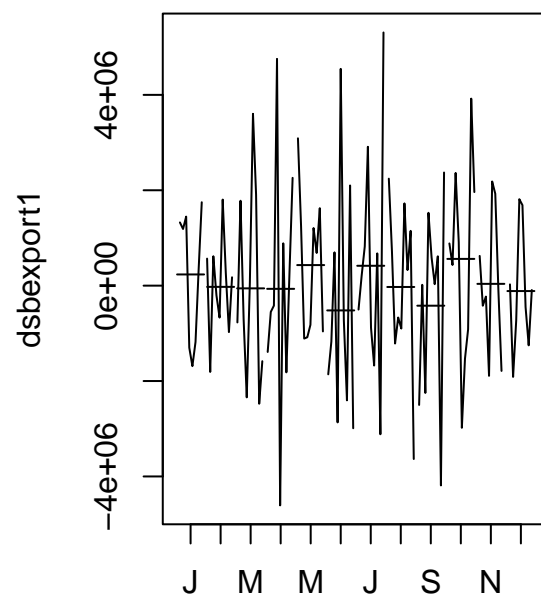
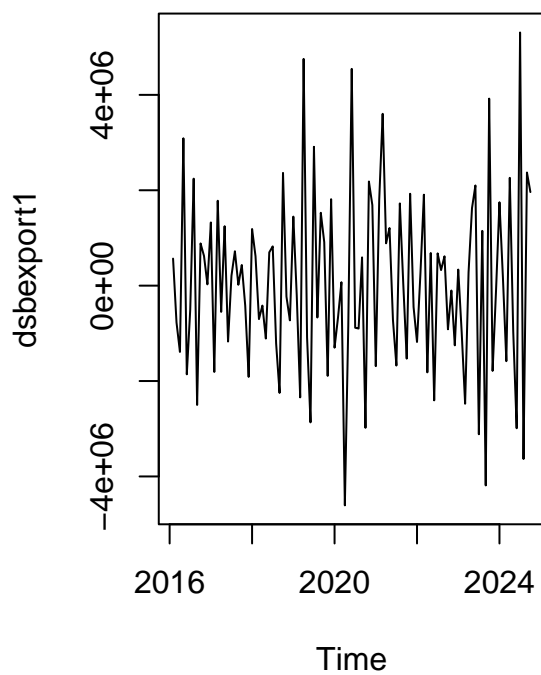


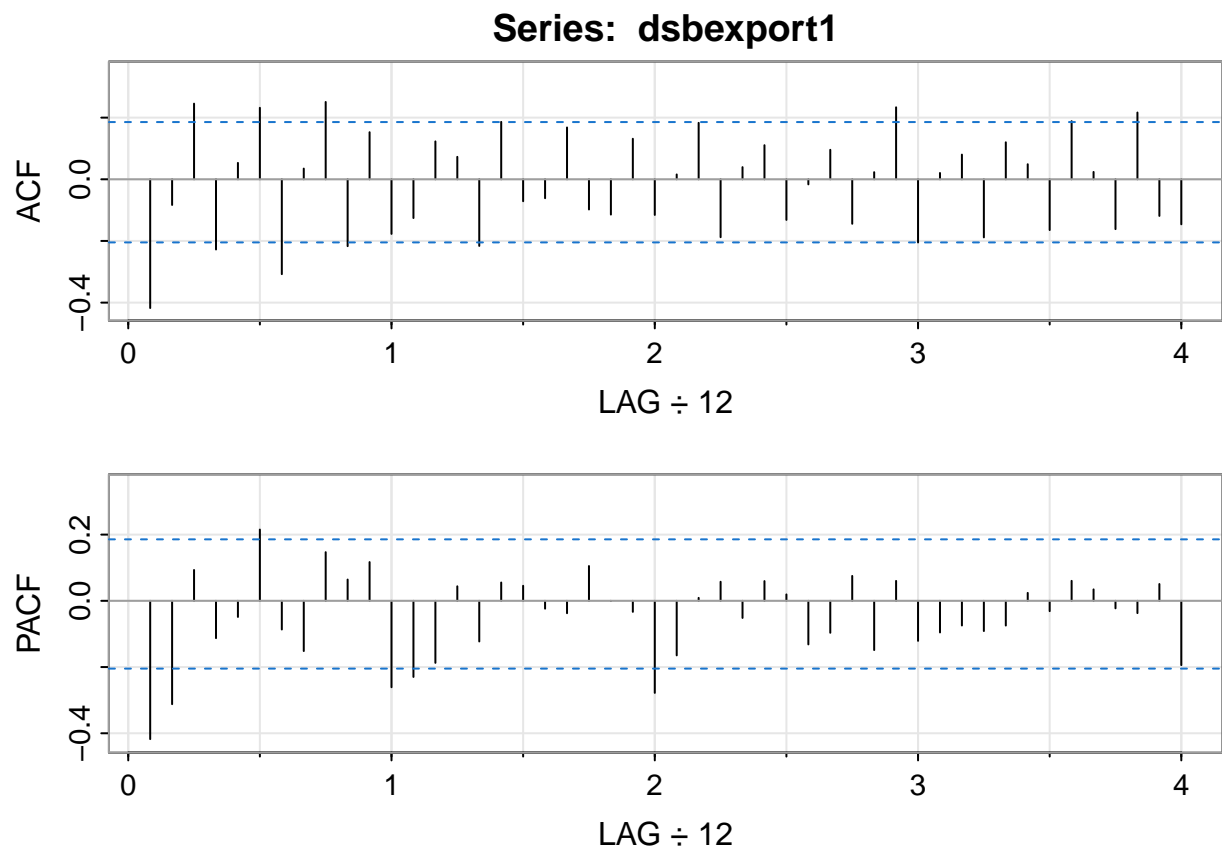
Les ρ_k sont presque tous significativement nuls, et bien que les p-values de la statistique de Ljung-Box ne soient pas toutes nulles, la situation n'est pas trop défavorable.



Plus on avance dans le temps, plus l'incertitude augmente, ce qui est habituel en prévision. Aussi on observe une stabilité des prévisions.

dsbexport1

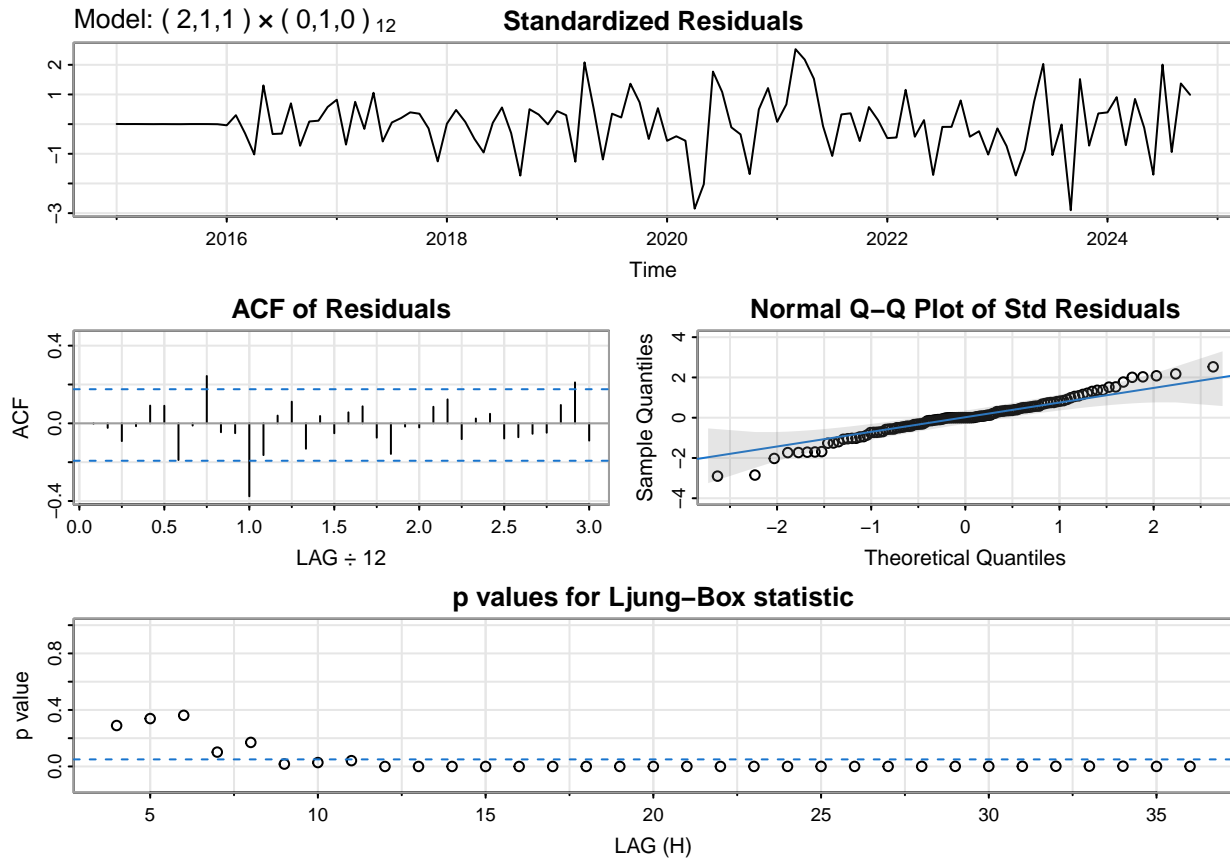




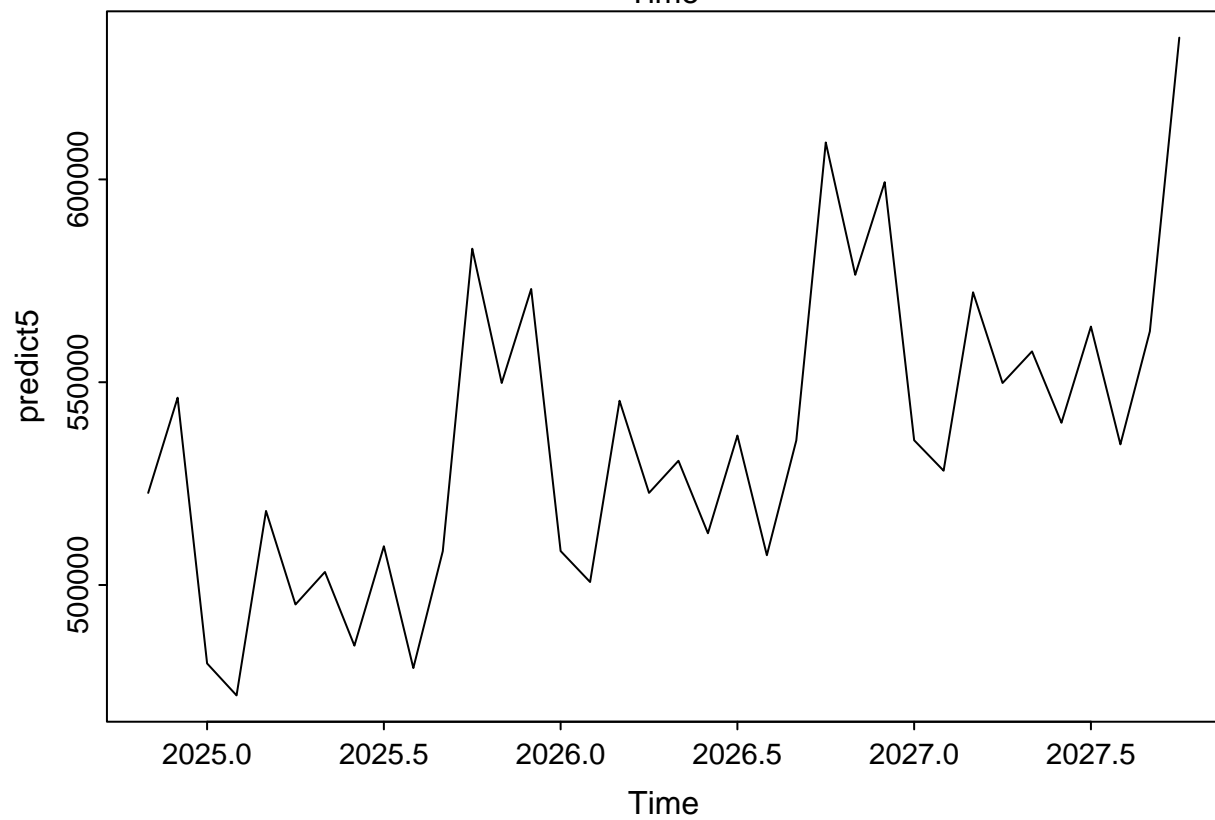
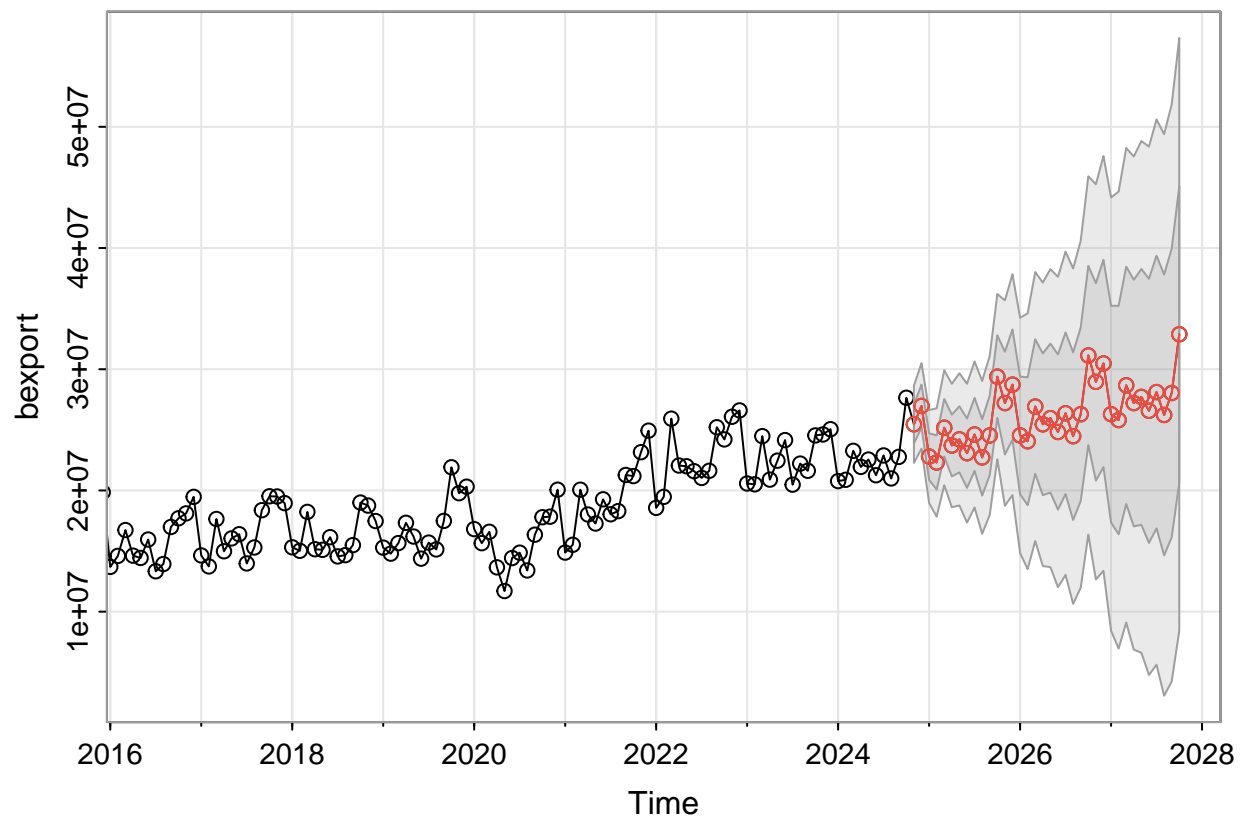
D'après l'analyse du graphique d'autocorrélation, on obtient $q = 1$ et $q' = 0$.

D'après l'analyse du graphique d'autocorrélation partielle, on déduit $p = 2$ et $p' = 0$.

4.3.2 Modele 5



Les resultats de ce modèles sont semblables a celle du modèle 4



Plus on avance dans le temps, plus l'incertitude augmente et nos prévision semble mieux sur la même allure que les observations précédentes.

4.4 Résumé des modèles

Nous allons évaluer la qualité des cinq modèles en utilisant les critères d'information d'Akaike (AIC) et bayésien (BIC), qui permettent de comparer leur performance et leur ajustement aux données.

	AIC	AICc	BIC
Modèle1	-2.186788	-2.156583	-1.856272
Modèle2	-2.408933	-2.368313	-2.031199
Modèle3	31.865063	31.895268	32.195579
Modèle4	-2.461054	-2.458791	-2.359951
Modèle5	31.480901	31.483164	31.582004

Modèles 1, 2 et 4 ont des valeurs négatives d'AIC, AICc et BIC, ce qui suggère qu'ils sont bien ajustés aux données. Modèles 3 et 5 ont des valeurs élevées et positives, ce qui indique un moins bon ajustement par rapport aux autres modèles.

5 Conclusion

Nous avons testé plusieurs modèle, avec et sans prise en compte de la saisonnalité. D'une part avec la transformation optimal de la série obtenue avec la fonction BoxCox, d'autre part avec une transformation logarithmique qui semblait justifiée ici.

Tous les modèles donnent des prédictions médiocre car intervalle de confiance assez large. Cependant les modèles prenant en compte la saisonnalité ont une prédiction beaucoup moins bonne car l'intervalle de confiance explose rapidement. Les modèles plus complexes donnent peut-être une meilleur prédiction mais avec peu de confiance, peut-être qu'avec plus de données les intervalles seraient plus étroits.

6 Annexe : Code Source

```
knitr::opts_chunk$set(echo=FALSE, warning = FALSE, message = FALSE, results = "hide")
library(naniar)
library(ggplot2)
library(tidyr)
library(tseries)
library(forecast)
library(astsa)
df <- read.table("F1069528899.csv", sep = ";", header = TRUE)
df.ts <- ts(df$Exportation, start = c(2015, 1), frequency = 12)
vis_miss(df)
ggplot(df, aes(x = Exportation)) +
  geom_histogram(color = "black", alpha = 0.7) +
  labs(title = "Histogramme", x = "Valeurs", y = "Fréquence")
summary(df$Exportation)

df <- data.frame(
  Date = time(df.ts),
  Exportation = as.numeric(df.ts)
)

ggplot(df, aes(x = Date, y = Exportation)) +
  geom_line(color = "blue", size = 1) +
  labs(
    title = "Exportations de viande et produits carnés en France",
    x = "Années",
    y = "Exportations"
  ) +
  theme_minimal()

plot(decompose(df.ts))
result <- adf.test(df.ts)

# Formater les résultats dans un tableau
table <- data.frame(
  Métrique = c("Valeur de test", "P-valeur", "Conclusion"),
  Valeur = c(
    round(result$statistic, 3),
    round(result$p.value, 3),
    ifelse(result$p.value < 0.05, "La série est stationnaire", "On ne peut pas affirmer que la série es
  )
)

knitr::kable(table, col.names = c("Métrique", "Valeur"), format = "markdown")
##passage au logarithme
ldf.ts <- log(df.ts)
dldf.ts <- diff(ldf.ts, 1)
plot(dldf.ts)

cat('La moyenne de la nouvelle série obtenue apres transformation est', mean(dldf.ts))

result1 <- adf.test(dldf.ts)
```

```

# Formater les résultats dans un tableau
table1 <- data.frame(
  Métrique = c("Valeur de test", "P-valeur", "Conclusion"),
  Valeur = c(
    round(result1$statistic, 3),
    round(result1$p.value, 3),
    ifelse(result1$p.value < 0.05, "La série est stationnaire", "On ne peut pas affirmer que la série est stationnaire")
  )
)

knitr::kable(table1, col.names = c("Métrique", "Valeur"), format = "markdown")

best_lambda_bc<-BoxCox.lambda(df.ts)
bexport<-BoxCox(df.ts, best_lambda_bc)
ts.plot(bexport, main='bexport')#pas stationnaire
bexport1 <- diff(bexport)
ts.plot(bexport1,main='bexport1')#semble stationnaire
#on a diff une fois donc d=1

cat('La moyenne de la nouvelle série obtenue après transformation de BoxCox et différencié est:', mean(bexport1))
acf2(dldf.ts)
model1 <- sarima(ldf.ts, p=2, d=1, q=10)
predict1 <- sarima.for( ldf.ts,n.ahead = 36,2,1,10)
predict1 <- sapply(predict1,exp)
model2 <- sarima(ldf.ts, p = 4, d = 1, q = 10)
predict2 <- sarima.for(ldf.ts,n.ahead = 36,4,1,10)
predict2 <- sapply(predict2,exp)
#ts.plot(predict2)
acf2(bexport1)
model3 <- sarima(bexport,2,1,10)
model3
predict3 <- sarima.for(ldf.ts,n.ahead = 36,2,1,10)
predict3 <- InvBoxCox(predict3$pred,lambda = best_lambda_bc)
#différentiation saisonnière
dslexport1 <- diff(ldf.ts,12)
par(mfrow=c(1,2))
ts.plot(dslexport1,main="dslexport1")
monthplot(dslexport1)
acf2(dslexport1)
model4 <- sarima(ldf.ts,2,1,1,0,1,0,12)
predict4 <- sarima.for(ldf.ts,n.ahead = 36,2,1,1,0,1,0,12)
predict4 <- sapply(predict4,exp)
#ts.plot(predict4)
#différentiation saisonnière
dsbexport1 <- diff(bexport1,12)
par(mfrow=c(1,2))
ts.plot(dsbexport1,main="dsbexport1")
monthplot(dsbexport1)
acf2(dsbexport1)
model5 <- sarima(bexport,2,1,1,0,1,0,12)
model5
predict5 <- sarima.for(bexport,n.ahead = 36,2,1,1,0,1,0,12)
predict5 <- InvBoxCox(predict5$pred,lambda = best_lambda_bc)

```



```
ts.plot(predict5)
resume <- rbind(model1$ICs,model2$ICs,model3$ICs, model4$ICs, model5$ICs)
rownames(resume) <- c("Modèle1", "Modèle2", "Modèle3", "Modèle4", "Modèle5")

knitr::kable(resume, format = "markdown")
```