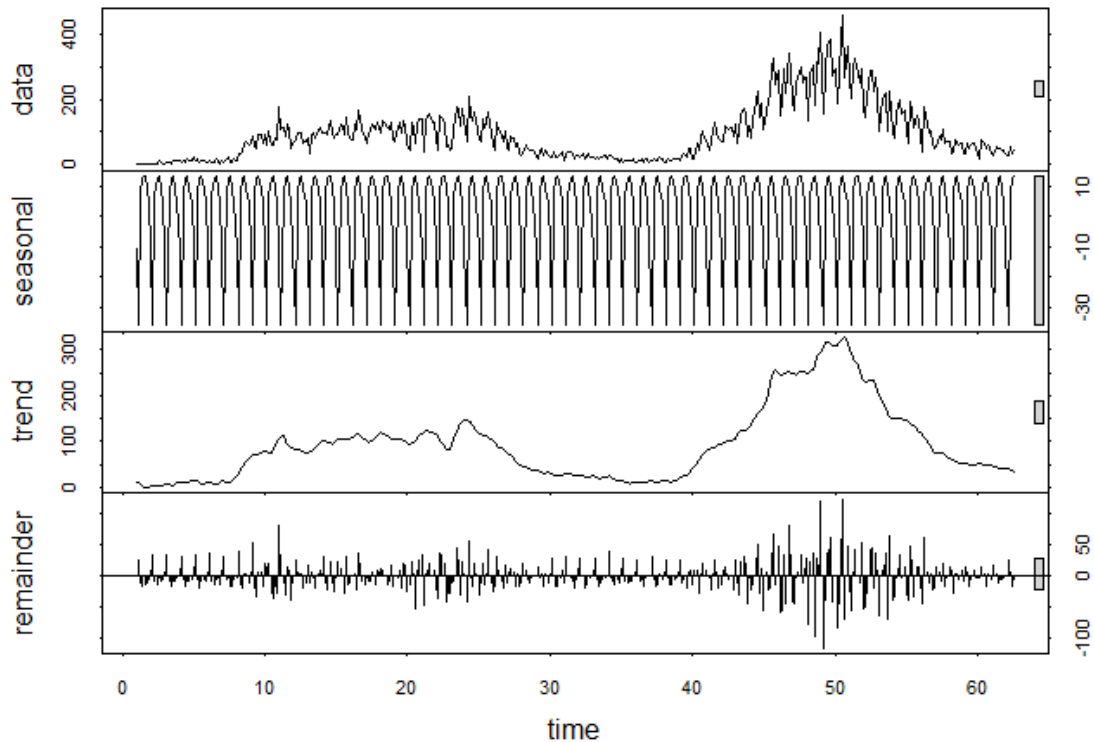


```
decomposition <- stl(cas_ts, s.window="periodic")
```

```
plot(decomposition)
```



```
# Visualisation de la série avec sa moyenne mobile
```

```
plot(cas_ts, main="Cas Positifs avec Moyenne Mobile",
```

```
      ylab="Nombre de cas", xlab="Temps")
```

```
lines(mm_7, col="red", lwd=2)
```

```
legend("topleft", legend=c("Données brutes", "Moyenne mobile"),
```

```
      col=c("black", "red"), lty=1)
```

```
# Ajustement d'un modèle linéaire pour la tendance
```

```
temps <- 1:length(cas_ts)
```

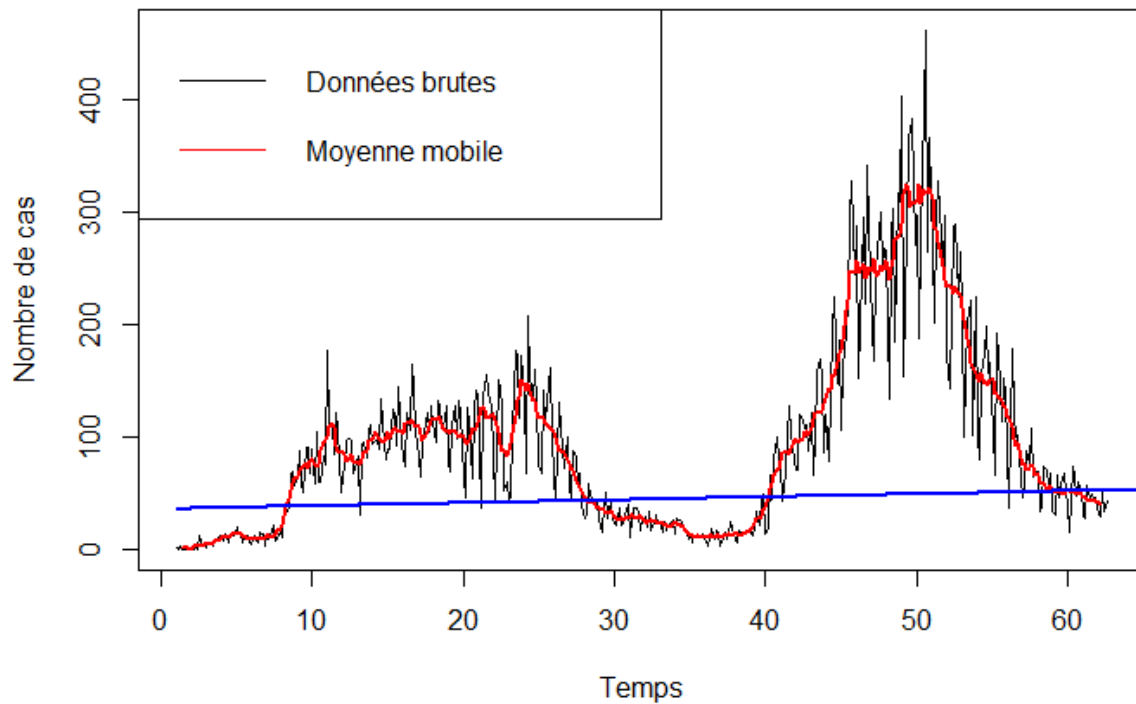
```
modele_lineaire <- lm(as.vector(cas_ts) ~ temps)
```

```
summary(modele_lineaire)
```

```
# Ajout de la tendance linéaire au graphique précédent
```

```
lines(temps, predict(modele_lineaire), col="blue", lwd=2)
```

Cas Positifs avec Moyenne Mobile



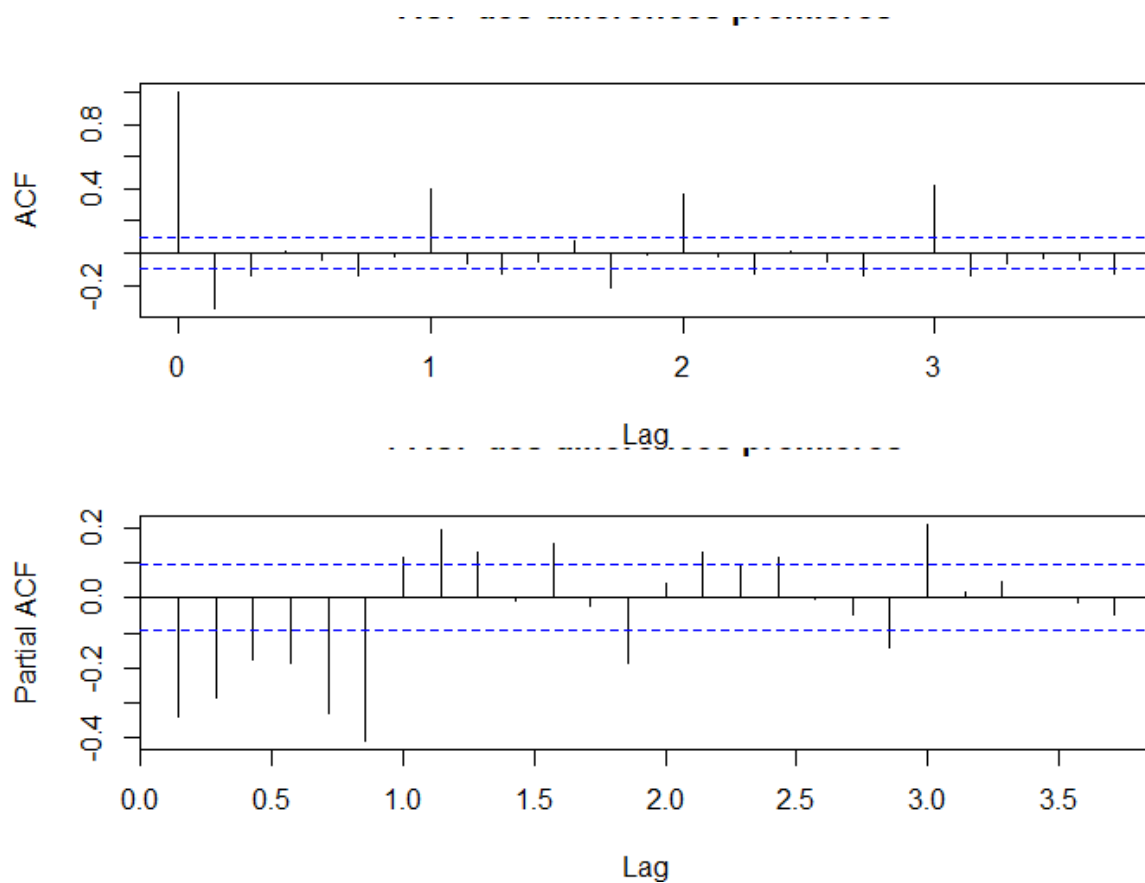
Analyse des fonctions d'autocorrélation sur la série stationnaire

```
par(mfrow=c(2,1))
```

```
par(mar=c(4,4,2,2))
```

```
acf(cas_diff, main="ACF des différences premières")
```

```
pacf(cas_diff, main="PACF des différences premières")
```



Modélisation ARIMA sur la série différenciée (stationnaire)

```
> modele_auto <- auto.arima(cas_diff, seasonal=TRUE)
```

```
> summary(modele_auto)
```

Series: cas_diff

ARIMA(0,0,1)(1,0,1)[7] with zero mean

Coefficients:

ma1 sar1 sma1

-0.7391 0.8982 -0.6020

s.e. 0.0321 0.0284 0.0529

sigma^2 = 962.1: log likelihood = -2092.59

AIC=4193.18 AICc=4193.27 BIC=4209.44

Training set error measures:

ME RMSE MAE MPE MAPE MASE ACF1

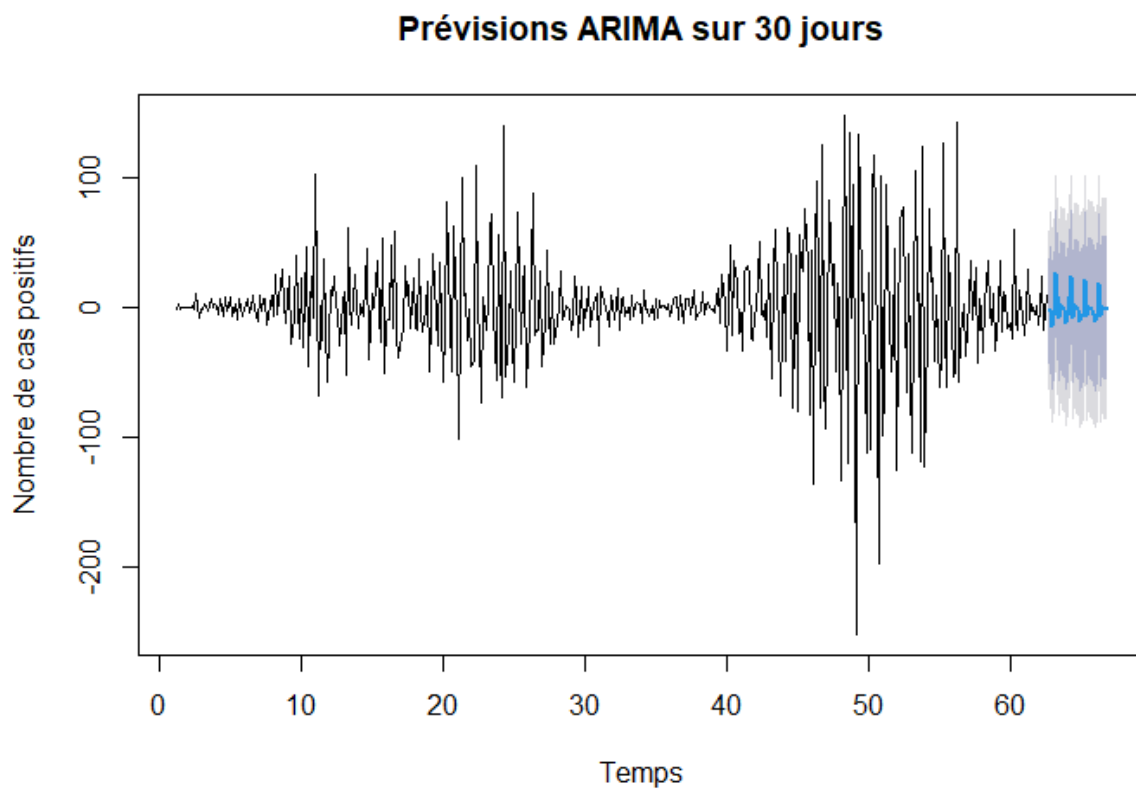
Training set -0.05961689 30.90945 20.50475 NaN Inf 0.6203363 -0.006537767

Prévisions sur 30 jours

```
prev_arima <- forecast(modele_auto, h=30)
```

4.2 Visualisation des prévisions ARIMA

```
plot(prev_arima, main="Prévisions ARIMA sur 30 jours",  
      xlab="Temps", ylab="Nombre de cas positifs")
```



Modèle de lissage exponentiel

```
modele_ets <- ets(cas_ts)
```

```
prev_ets <- forecast(modele_ets, h=30)
```

Création du dataframe pour la comparaison des deux méthodes

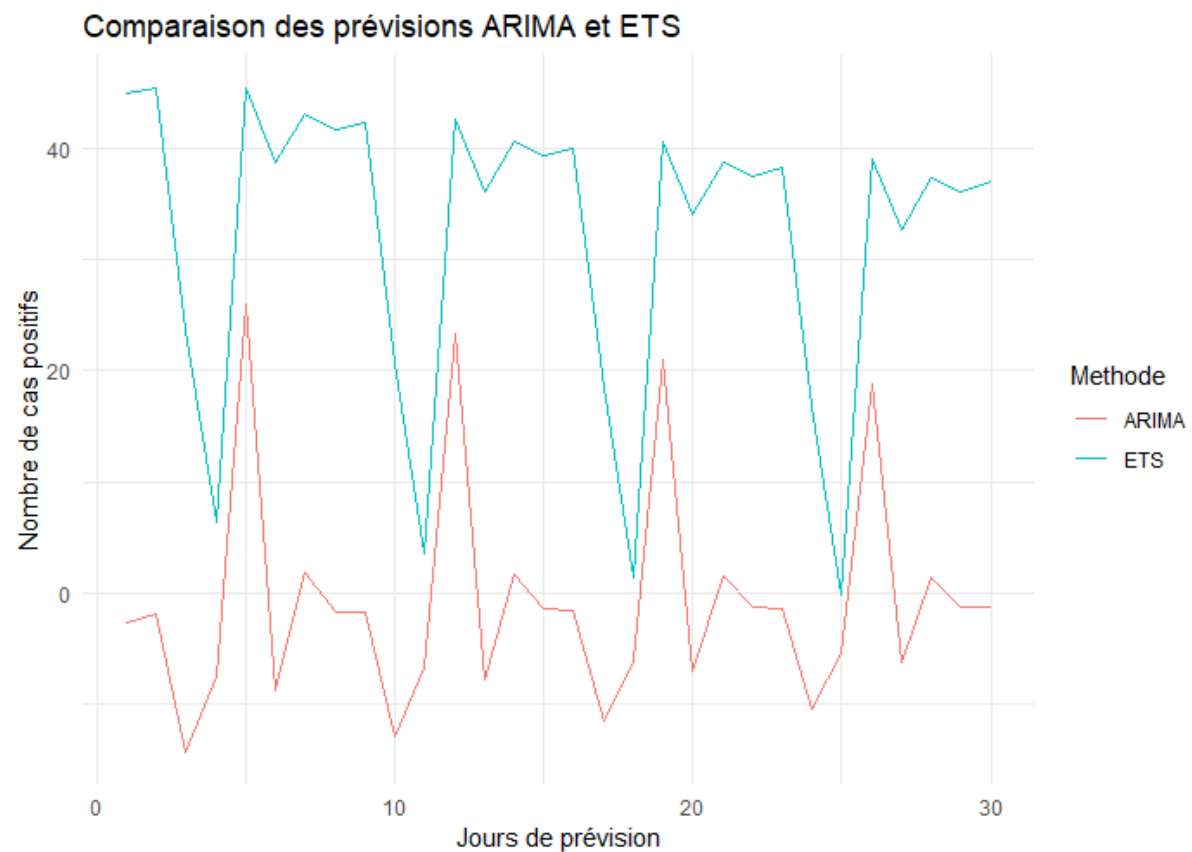
```

df_comparaison <- data.frame(
  Date = 1:30,
  ARIMA = as.numeric(prev_arima$mean), # Utilisation des prévisions dé-différenciées
  ETS = as.numeric(prev_ets$mean)
)

# Conversion en format long pour ggplot
df_long <- pivot_longer(df_comparaison,
  cols = c("ARIMA", "ETS"),
  names_to = "Methode",
  values_to = "Prevision")

# Visualisation comparative des deux méthodes
ggplot(df_long, aes(x = Date, y = Prevision, color = Methode)) +
  geom_line() +
  labs(title = "Comparaison des prévisions ARIMA et ETS",
    x = "Jours de prévision",
    y = "Nombre de cas positifs") +
  theme_minimal()

```



Évaluation de la précision des modèles

```
> print("Précision du modèle ARIMA :")
```

```
[1] "Précision du modèle ARIMA :"
```

```
> accuracy(prev_arima)
```

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	-0.05961689	30.90945	20.50475	NaN	Inf	0.6203363	-0.006537767

```
> print("Précision du modèle ETS :")
```

```
[1] "Précision du modèle ETS :"
```

```
> accuracy(prev_ets)
```

	ME	RMSE	MAE	MPE	MAPE	MASE	ACF1
Training set	0.2317715	31.35175	21.72128	NaN	Inf	0.8129221	0.03148064