



## 4 - Lissage exponentiel et qualité des prévisions

ALAIN QUARTIER-LA-TENTE

# Objectifs de cette séquence

---

Présenter les méthodes basiques de prévision d'une série temporelle ainsi que le lissage exponentiel

# Questions de positionnement

---

Quelles sont les modèles de prévision les plus simples ?

Comment évaluer la qualité d'un modèle de prévision ?

Quelles sont les propriétés que doivent suivre les erreurs de prévision ?

Qu'est-ce que le lissage exponentiel ?

# Sommaire

---

## 1. Modèles de prévision simples

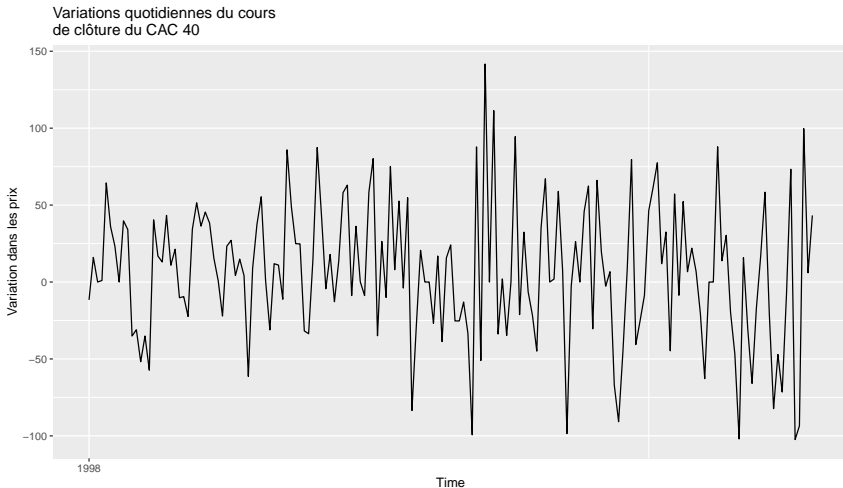
### 1.1 Exemples

## 2. Modèles exponentiels

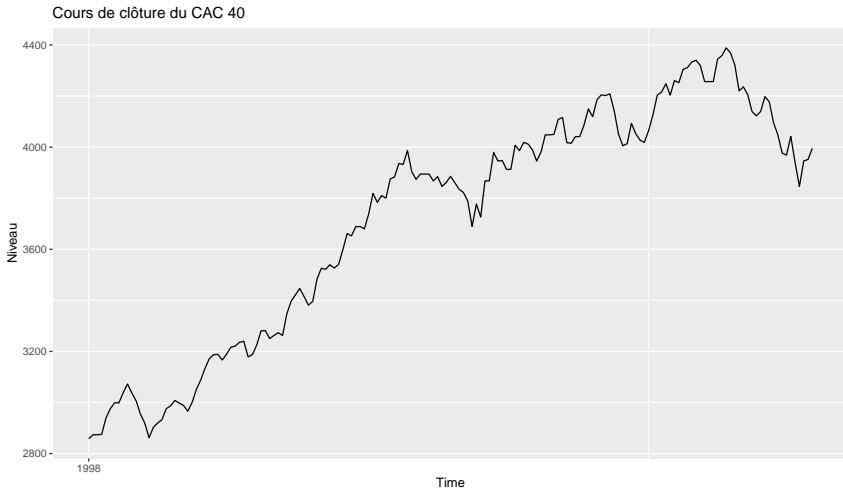
## 3. Résidus et qualité des prévisions

## 4. Conclusion

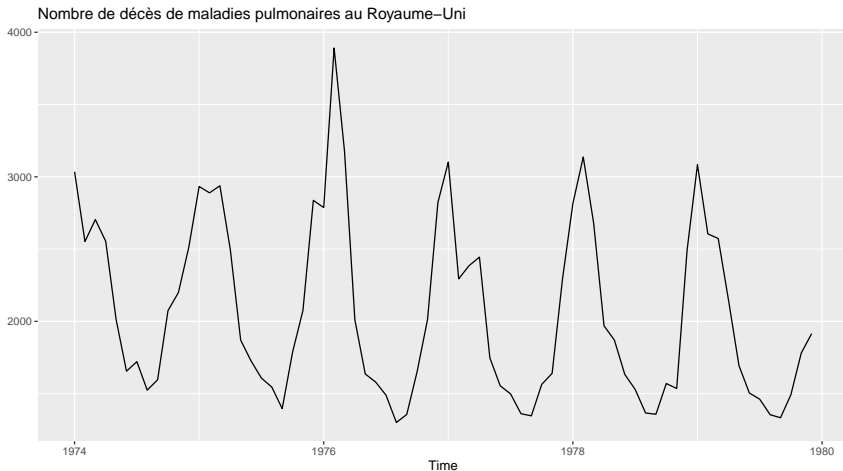
# Comment prévoir ces séries ? (1)



# Comment prévoir ces séries ? (2)



# Comment prévoir ces séries ? (3)



# Modèles de base

---

1. Moyenne de valeurs passées (`forecast::meanf()` ou `fable::MEAN()`):

$$\hat{y}_{T+h|T} = \bar{y} = \frac{1}{T} \sum_{i=1}^T y_t$$



# Modèles de base

---

1. Moyenne de valeurs passées (`forecast::meanf()` ou `fable::MEAN()`):

$$\hat{y}_{T+h|T} = \bar{y} = \frac{1}{T} \sum_{i=1}^T y_t$$

2. Dernière valeur connue (`forecast::naive()` ou `fable::NAIVE()`):

$$\hat{y}_{T+h|T} = y_T$$

Hypothèse du marché efficient

# Modèles de base

1. Moyenne de valeurs passées (`forecast::meanf()` ou `fable::MEAN()`):

$$\hat{y}_{T+h|T} = \bar{y} = \frac{1}{T} \sum_{i=1}^T y_t$$

2. Dernière valeur connue (`forecast::naive()` ou `fable::NAIVE()`):

$$\hat{y}_{T+h|T} = y_T$$

Hypothèse du marché efficient

3. Dernière valeur connue à la saison précédente (`forecast::snaive()` ou `fable::SNAIVE()`),  $m$  la période de saisonnalité :

$$\hat{y}_{T+h|T} = y_{t+h-m(k+1)} \quad \text{avec} \quad k = \lfloor (h-1)/m \rfloor$$

# Modèles de base

1. Moyenne de valeurs passées (`forecast::meanf()` ou `fable::MEAN()`):

$$\hat{y}_{T+h|T} = \bar{y} = \frac{1}{T} \sum_{i=1}^T y_t$$

2. Dernière valeur connue (`forecast::naive()` ou `fable::NAIVE()`):

$$\hat{y}_{T+h|T} = y_T$$

Hypothèse du marché efficient

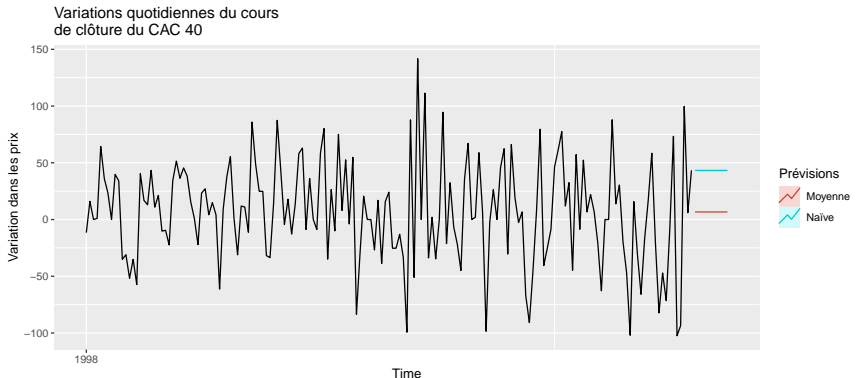
3. Dernière valeur connue à la saison précédente (`forecast::snaive()` ou `fable::SNAIVE()`),  $m$  la période de saisonnalité :

$$\hat{y}_{T+h|T} = y_{t+h-m(k+1)} \quad \text{avec} \quad k = \lfloor (h-1)/m \rfloor$$

Il y a également d'autres combinaisons possibles (*drift*...), voir <https://otexts.com/fpp3/simple-methods.html>

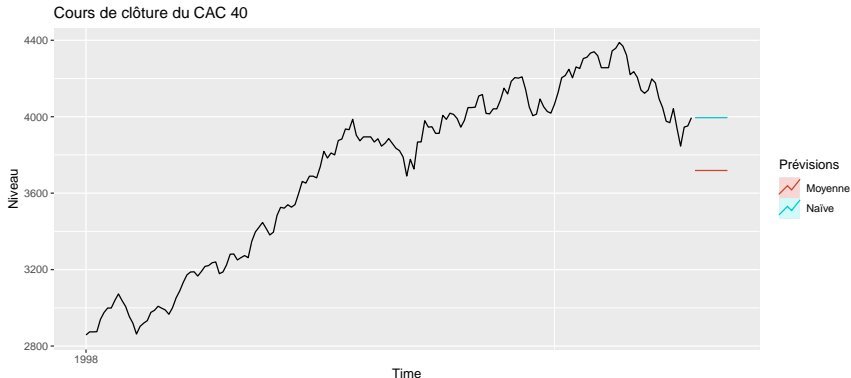
# Retour sur les exemples (1)

```
y = window(diff(EuStockMarkets[, "CAC"]), start = 1998)
autoplot(y, y = "Variation dans les prix",
         main = "Variations quotidiennes du cours\nde clôture du CAC 40") +
  autolayer(meanf(y, h=10), PI=FALSE, series="Moyenne") +
  autolayer(naive(y, h=10), PI=FALSE, series="Naïve") +
  guides(colour=guide_legend(title="Prévisions"))
```



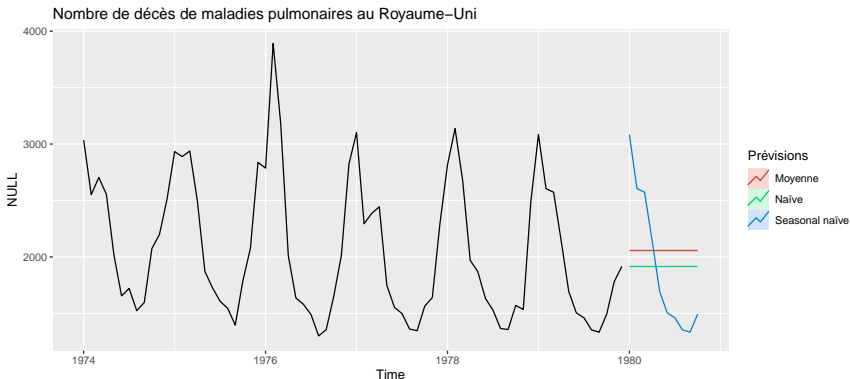
## Retour sur les exemples (2)

```
y = window(EuStockMarkets[, "CAC"], start = 1998)
autoplot(y, y = "Niveau",
         main = "Cours de clôture du CAC 40") +
  autolayer(meanf(y, h=10), PI=FALSE, series="Moyenne") +
  autolayer(naive(y, h=10), PI=FALSE, series="Naïve") +
  guides(colour=guide_legend(title="Prévisions"))
```



# Retour sur les exemples (3)

```
autoplot(ldeaths, y = "NULL",  
         main = "Nombre de décès de maladies pulmonaires au Royaume-Uni") +  
  autolayer(meanf(ldeaths, h=10), PI=FALSE, series="Moyenne") +  
  autolayer(naive(ldeaths, h=10), PI=FALSE, series="Naïve") +  
  autolayer(snaive(ldeaths, h=10), PI=FALSE, series="Seasonal naïve") +  
  guides(colour=guide_legend(title="Prévisions"))
```



# Sommaire

---

1. Modèles de prévision simples
- 2. Modèles exponentiels**
3. Résidus et qualité des prévisions
4. Conclusion

# Modèle exponentiel simple (SES)

---

Méthode entre prévision naïve et moyenne lorsqu'il n'y a pas de tendance ni de saisonnalité claire



# Modèle exponentiel simple (SES)

---

Méthode entre prévision naïve et moyenne lorsqu'il n'y a pas de tendance ni de saisonnalité claire

Idée : un poids décroissant est associée aux valeurs passées

$$\hat{y}_{T+1|T} = \alpha y_t + \alpha(1 - \alpha)y_{t-1} + \alpha(1 - \alpha)^2 y_{t-2} + \dots$$

# Modèle exponentiel simple (SES)

Méthode entre prévision naïve et moyenne lorsqu'il n'y a pas de tendance ni de saisonnalité claire

Idée : un poids décroissant est associée aux valeurs passées

$$\hat{y}_{T+1|T} = \alpha y_t + \alpha(1 - \alpha)y_{t-1} + \alpha(1 - \alpha)^2 y_{t-2} + \dots$$

Peut s'écrire sous forme espace-état :

$$\begin{cases} \hat{y}_{T+h|T} &= l_t \\ l_t &= \alpha y_t + (1 - \alpha)l_{t-1} \end{cases} \iff \begin{cases} y_t &= l_{t-1} + \varepsilon_t \\ l_t &= l_{t-1} + \alpha \varepsilon_t \end{cases}$$

$l_t$  représente le niveau de la série.

# Modèle exponentiel simple (SES)

Méthode entre prévision naïve et moyenne lorsqu'il n'y a pas de tendance ni de saisonnalité claire

Idée : un poids décroissant est associée aux valeurs passées

$$\hat{y}_{T+1|T} = \alpha y_t + \alpha(1 - \alpha)y_{t-1} + \alpha(1 - \alpha)^2 y_{t-2} + \dots$$

Peut s'écrire sous forme espace-état :

$$\begin{cases} \hat{y}_{T+h|T} &= l_t \\ l_t &= \alpha y_t + (1 - \alpha)l_{t-1} \end{cases} \iff \begin{cases} y_t &= l_{t-1} + \varepsilon_t \\ l_t &= l_{t-1} + \alpha \varepsilon_t \end{cases}$$

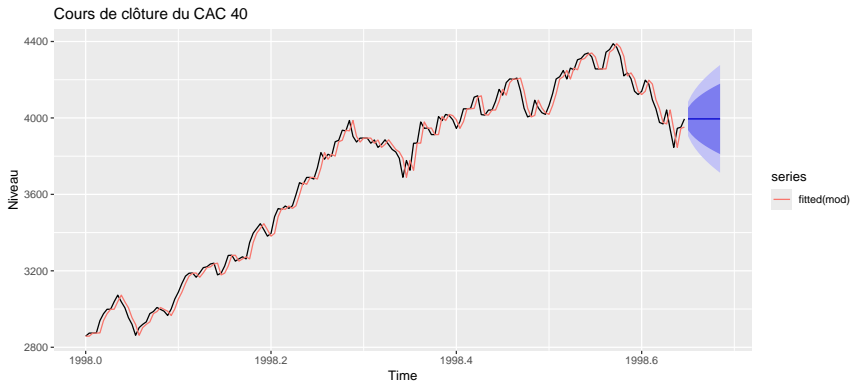
$l_t$  représente le niveau de la série.

Paramètres à estimer :  $l_0$  et  $\alpha$  par minimisation des erreurs de prévision :

$$SSE = \sum_{t=1}^T e_t^2 = \sum_{t=1}^T (y_t - \hat{y}_{t|t-1})^2$$

# Exemple

```
mod <- ses(y)
mod2 <- ets(y, model = "ANN") # autre option
autoplot(mod) +
  autolayer(fitted(mod)) +
  labs(y = "Niveau",
       title = "Cours de clôture du CAC 40")
```



# Exemple

```
mod2
```

```
ETS(A,N,N)
```

```
Call:
```

```
ets(y = y, model = "ANN")
```

```
Smoothing parameters:
```

```
alpha = 0.9999
```

```
Initial states:
```

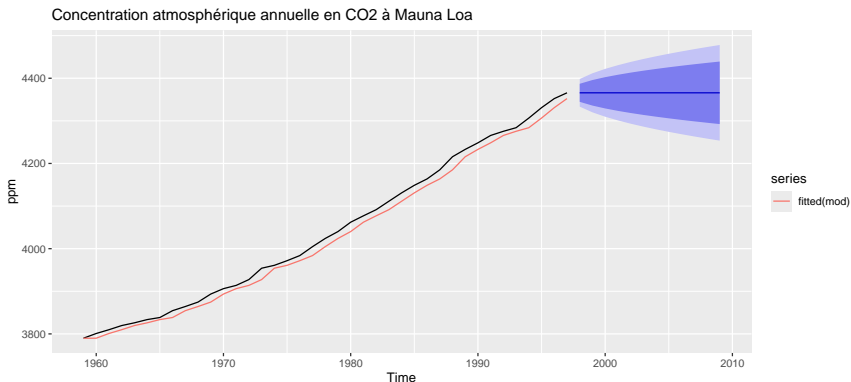
```
l = 2857.9761
```

```
sigma: 45.4825
```

AIC	AICc	BIC
2161.198	2161.343	2170.587

## Exemple 2

```
mod <- ses(aggregate(co2), h = 12)
autoplot(mod) +
  autolayer(fitted(mod)) +
  labs(y = "ppm",
       title = "Concentration atmosphérique annuelle en CO2 à Mauna Loa")
```



# Lissage exponentiel double : Holt (1957)

Le SES peut être étendu pour ajouter une prévision de la tendance :

$$\begin{cases} \hat{y}_{T+h|T} &= l_t + hb_t \\ l_t &= \alpha y_t + (1 - \alpha)(l_{t-1} + b_{t-1}) \\ b_t &= \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1} \end{cases} \iff \begin{cases} y_t &= l_{t-1} + b_{t-1} + \varepsilon_t \\ l_t &= l_{t-1} + b_{t-1} + \alpha\varepsilon_t \\ b_t &= b_{t-1} + \beta\varepsilon_t \end{cases}$$

# Lissage exponentiel double : Holt (1957)

Le SES peut être étendu pour ajouter une prévision de la tendance :

$$\begin{cases} \hat{y}_{T+h|T} &= l_t + hb_t \\ l_t &= \alpha y_t + (1 - \alpha)(l_{t-1} + b_{t-1}) \\ b_t &= \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1} \end{cases} \iff \begin{cases} y_t &= l_{t-1} + b_{t-1} + \varepsilon_t \\ l_t &= l_{t-1} + b_{t-1} + \alpha\varepsilon_t \\ b_t &= b_{t-1} + \beta\varepsilon_t \end{cases}$$

On a encore  $l_t = \alpha y_t + (1 - \alpha)\hat{y}_{t|t-1}$ .



## *Damped trend*/tendance amortie : Gardner & McKenzie (1985)

---

La méthode de Holt prévoit une tendance croissante de manière indéfinie : tend à sur-estimer les prévisions sur horizon longue.

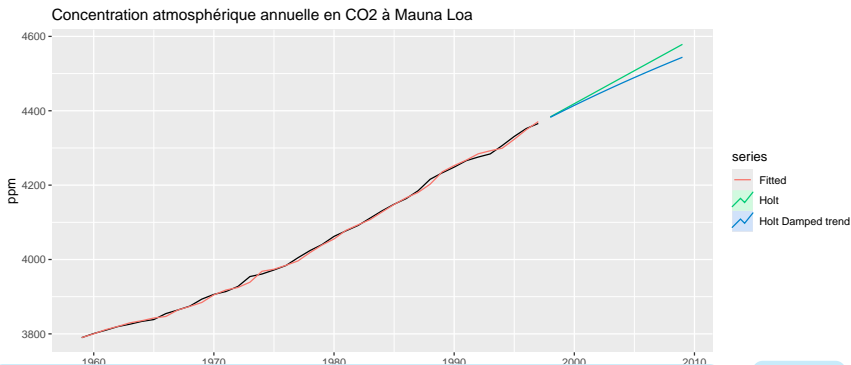


Amortir la tendance avec le temps

$$\begin{cases} \hat{y}_{T+h|T} &= l_t + (\phi + \phi^2 + \dots + \phi^h)b_t \\ l_t &= \alpha y_t + (1 - \alpha)(l_{t-1} + \phi b_{t-1}) \\ b_t &= \beta^*(l_t - l_{t-1}) + (1 - \beta^*)\phi b_{t-1} \end{cases} \iff \begin{cases} y_t &= l_{t-1} + \phi b_{t-1} + \varepsilon_t \\ l_t &= l_{t-1} + \phi b_{t-1} + \alpha \varepsilon_t \\ b_t &= \phi b_{t-1} + \beta \varepsilon_t \end{cases}$$

# Exemple

```
mod <- holt(aggregate(co2), h = 12)
mod_damped <- holt(aggregate(co2), damped = TRUE, h = 12)
mod2 <- ets(aggregate(co2), model = "AAN") # autre option
autoplot(aggregate(co2)) + autolayer(mod, PI = FALSE, series = "Holt") +
  autolayer(mod_damped, PI = FALSE, series = "Holt Damped trend") +
  autolayer(fitted(mod), series = "Fitted") +
  labs(y = "ppm",
       title = "Concentration atmosphérique annuelle en CO2 à Mauna Loa")
```



# Exemple

---

```
mod2
```

```
ETS(A,A,N)
```

```
Call:
```

```
ets(y = aggregate(co2), model = "AAN")
```

```
Smoothing parameters:
```

```
alpha = 0.9999
```

```
beta  = 0.1785
```

```
Initial states:
```

```
l = 3779.61
```

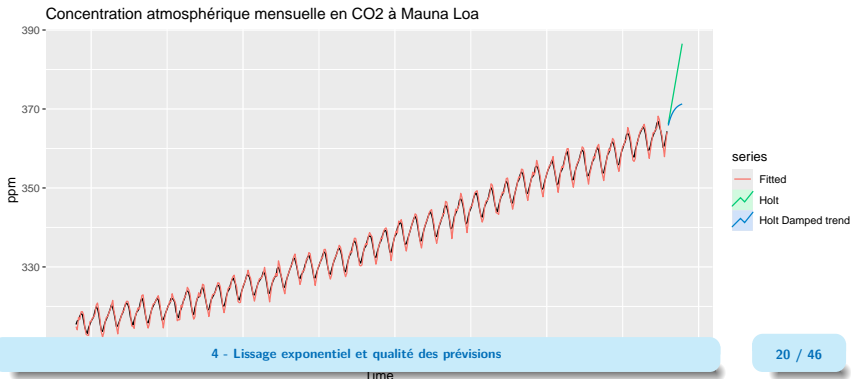
```
b = 10.2955
```

```
sigma: 5.8373
```

	AIC	AICc	BIC
	286.2718	288.0900	294.5896

## Exemple 2

```
mod <- holt(co2, h = 12)
mod_damped <- holt(co2, damped = TRUE, h = 12)
autoplot(co2) + autolayer(mod, PI = FALSE, series = "Holt") +
  autolayer(mod_damped, PI = FALSE, series = "Holt Damped trend") +
  autolayer(fitted(mod), series = "Fitted") +
  labs(y = "ppm",
       title = "Concentration atmosphérique mensuelle en CO2 à Mauna Loa")
```



# Holt-Winter

---

On ajoute une composante saisonnière !

$$\left\{ \begin{array}{l} \hat{y}_{T+h|T} = l_t + hb_t + s_{t+h-m(k+1)} \\ l_t = \alpha(y_t - s_{t-m}) + (1 - \alpha)(l_{t-1} + b_{t-1}) \\ b_t = \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1} \\ s_t = \gamma(y_t - l_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m} \end{array} \right. \iff \left\{ \begin{array}{l} y_t = l_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t \\ l_t = l_{t-1} + b_{t-1} + \alpha\varepsilon_t \\ b_t = b_{t-1} + \beta\varepsilon_t \\ s_t = s_{t-m} + \gamma\varepsilon_t \end{array} \right.$$

# Holt-Winter

On ajoute une composante saisonnière !

$$\begin{cases} \hat{y}_{T+h|T} &= l_t + hb_t + s_{t+h-m(k+1)} \\ l_t &= \alpha(y_t - s_{t-m}) + (1 - \alpha)(l_{t-1} + b_{t-1}) \\ b_t &= \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1} \\ s_t &= \gamma(y_t - l_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m} \end{cases} \iff \begin{cases} y_t &= l_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t \\ l_t &= l_{t-1} + b_{t-1} + \alpha\varepsilon_t \\ b_t &= b_{t-1} + \beta\varepsilon_t \\ s_t &= s_{t-m} + \gamma\varepsilon_t \end{cases}$$

On peut aussi réécrire

$$s_t = \gamma^*(y_t - l_t) + (1 - \gamma^*)s_{t-m}$$

# Holt-Winter

On ajoute une composante saisonnière !

$$\begin{cases} \hat{y}_{T+h|T} &= l_t + hb_t + s_{t+h-m(k+1)} \\ l_t &= \alpha(y_t - s_{t-m}) + (1 - \alpha)(l_{t-1} + b_{t-1}) \\ b_t &= \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1} \\ s_t &= \gamma(y_t - l_{t-1} - b_{t-1}) + (1 - \gamma)s_{t-m} \end{cases} \iff \begin{cases} y_t &= l_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t \\ l_t &= l_{t-1} + b_{t-1} + \alpha\varepsilon_t \\ b_t &= b_{t-1} + \beta\varepsilon_t \\ s_t &= s_{t-m} + \gamma\varepsilon_t \end{cases}$$

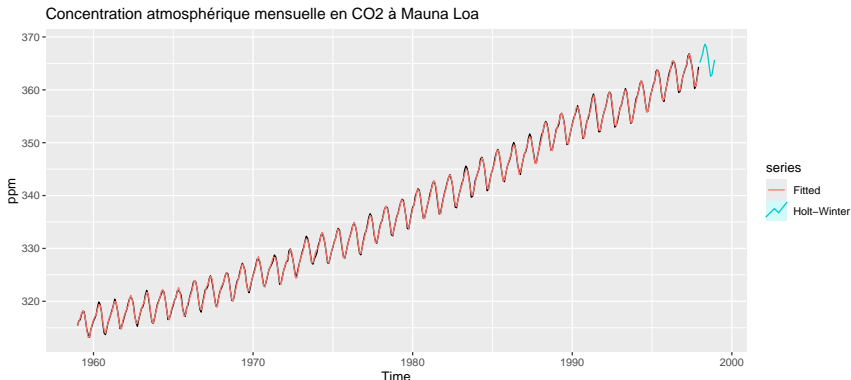
On peut aussi réécrire

$$s_t = \gamma^*(y_t - l_t) + (1 - \gamma^*)s_{t-m}$$

Même idée avec tendance amortie

# Exemple

```
mod <- hw(co2, h=12)
mod2 <- ets(co2, model = "AAA") # autre option
autoplot(co2) + autolayer(mod, PI = FALSE, series = "Holt-Winter") +
  autolayer(fitted(mod), series = "Fitted") +
  labs(y = "ppm",
       title = "Concentration atmosphérique mensuelle en CO2 à Mauna Loa")
```





# Exemple

```
mod2
```

```
ETS(A,A,A)
```

```
Call:
```

```
ets(y = co2, model = "AAA")
```

```
Smoothing parameters:
```

```
  alpha = 0.5785
```

```
  beta  = 0.0061
```

```
  gamma = 0.1373
```

```
Initial states:
```

```
  l = 315.3303
```

```
  b = 0.0801
```

```
  s = -0.8174 -1.836 -3.024 -2.7715 -1.2671 0.7784  
      2.1746 2.702 2.1571 1.1912 0.6693 0.0433
```

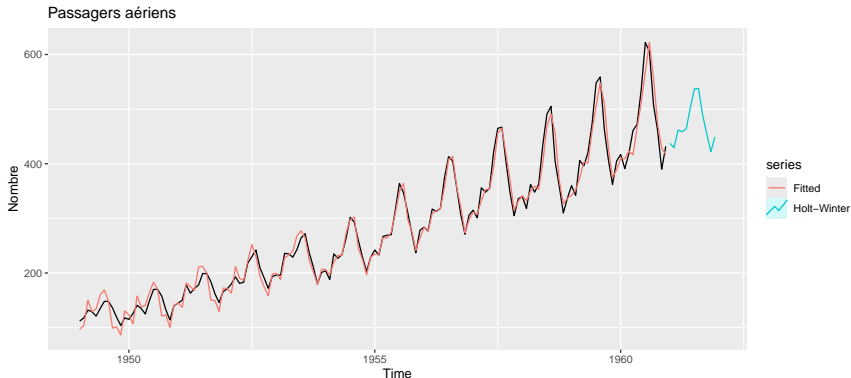
```
sigma: 0.294
```

```
      AIC      AICc      BIC
```

```
1749.350 1750.710 1819.874
```

# Et maintenant ?

```
mod <- hw(AirPassengers, h=12)
mod2 <- ets(AirPassengers, model = "AAA") # autre option
autoplot(AirPassengers) + autolayer(mod, PI = FALSE, series = "Holt-Winter") +
  autolayer(fitted(mod), series = "Fitted") +
  labs(y = "Nombre",
       title = "Passagers aériens")
```



# Saisonnalité multiplicative

$$\begin{cases} \hat{y}_{T+h|T} &= (l_t + hb_t) + s_{t+h-m(k+1)} \\ l_t &= \alpha \frac{y_t}{s_{t-m}} + (1 - \alpha)(l_{t-1} + b_{t-1}) \\ b_t &= \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1} \\ s_t &= \gamma \frac{y_t}{l_{t-1} + b_{t-1}} + (1 - \gamma)s_{t-m} \end{cases}$$

$$\Longleftrightarrow \begin{cases} y_t &= (l_{t-1} + b_{t-1})s_{t-m} + \varepsilon_t \\ l_t &= l_{t-1} + b_{t-1} + \alpha\varepsilon_t \\ b_t &= b_{t-1} + \beta\varepsilon_t/s_{t-m} \\ s_t &= s_{t-m} + \gamma\varepsilon_t/(l_{t-1} + b_{t-1}) \end{cases}$$

# Saisonnalité multiplicative

$$\begin{cases} \hat{y}_{T+h|T} &= (l_t + hb_t) + s_{t+h-m(k+1)} \\ l_t &= \alpha \frac{y_t}{s_{t-m}} + (1 - \alpha)(l_{t-1} + b_{t-1}) \\ b_t &= \beta^*(l_t - l_{t-1}) + (1 - \beta^*)b_{t-1} \\ s_t &= \gamma \frac{y_t}{l_{t-1} + b_{t-1}} + (1 - \gamma)s_{t-m} \end{cases}$$

$$\Longleftrightarrow \begin{cases} y_t &= (l_{t-1} + b_{t-1})s_{t-m} + \varepsilon_t \\ l_t &= l_{t-1} + b_{t-1} + \alpha\varepsilon_t \\ b_t &= b_{t-1} + \beta\varepsilon_t/s_{t-m} \\ s_t &= s_{t-m} + \gamma\varepsilon_t/(l_{t-1} + b_{t-1}) \end{cases}$$

L'erreur aussi peut être multiplicative !

# Taxonomie des modèles ETS

---

## Notations Générales

ETS : ExponenTial SMOOTHing  
↑    ↑    ↑  
Erreur Tendance Saisonnalité

# Taxonomie des modèles ETS

## Notations Générales

E T S : ExponenTial Smoothing  
 Erreur Tendance Saisonnalité

- Erreur : Additive ("A") ou multiplicative ("M")
- Tendance : Sans tendance ("N"), additive ("A"), multiplicative ("M") ou amortie ("Ad" ou "Md")
- Saisonnalité : Sans saisonnalité ("N"), additive ("A") ou multiplicative ("M")

"Z" pour une sélection automatique

# Erreurs additives

Trend	Seasonal		
	N	A	M
<b>N</b>	$y_t = \ell_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha \varepsilon_t$	$y_t = \ell_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha \varepsilon_t$ $s_t = s_{t-m} + \gamma \varepsilon_t$	$y_t = \ell_{t-1} s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \alpha \varepsilon_t / s_{t-m}$ $s_t = s_{t-m} + \gamma \varepsilon_t / \ell_{t-1}$
<b>A</b>	$y_t = \ell_{t-1} + b_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha \varepsilon_t$ $b_t = b_{t-1} + \beta \varepsilon_t$	$y_t = \ell_{t-1} + b_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha \varepsilon_t$ $b_t = b_{t-1} + \beta \varepsilon_t$ $s_t = s_{t-m} + \gamma \varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1}) s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha \varepsilon_t / s_{t-m}$ $b_t = b_{t-1} + \beta \varepsilon_t / s_{t-m}$ $s_t = s_{t-m} + \gamma \varepsilon_t / (\ell_{t-1} + b_{t-1})$
<b>A<sub>d</sub></b>	$y_t = \ell_{t-1} + \phi b_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha \varepsilon_t$ $b_t = \phi b_{t-1} + \beta \varepsilon_t$	$y_t = \ell_{t-1} + \phi b_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha \varepsilon_t$ $b_t = \phi b_{t-1} + \beta \varepsilon_t$ $s_t = s_{t-m} + \gamma \varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1}) s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha \varepsilon_t / s_{t-m}$ $b_t = \phi b_{t-1} + \beta \varepsilon_t / s_{t-m}$ $s_t = s_{t-m} + \gamma \varepsilon_t / (\ell_{t-1} + \phi b_{t-1})$
<b>M</b>	$y_t = \ell_{t-1} b_{t-1} + \varepsilon_t$ $\ell_t = \ell_{t-1} b_{t-1} + \alpha \varepsilon_t$ $b_t = b_{t-1} + \beta \varepsilon_t / \ell_{t-1}$	$y_t = \ell_{t-1} b_{t-1} + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} b_{t-1} + \alpha \varepsilon_t$ $b_t = b_{t-1} + \beta \varepsilon_t / \ell_{t-1}$ $s_t = s_{t-m} + \gamma \varepsilon_t$	$y_t = \ell_{t-1} b_{t-1} s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} b_{t-1} + \alpha \varepsilon_t / s_{t-m}$ $b_t = b_{t-1} + \beta \varepsilon_t / (s_{t-m} \ell_{t-1})$ $s_t = s_{t-m} + \gamma \varepsilon_t / (\ell_{t-1} b_{t-1})$
<b>M<sub>d</sub></b>	$y_t = \ell_{t-1} b_{t-1}^\phi + \varepsilon_t$ $\ell_t = \ell_{t-1} b_{t-1}^\phi + \alpha \varepsilon_t$ $b_t = b_{t-1}^\phi + \beta \varepsilon_t / \ell_{t-1}$	$y_t = \ell_{t-1} b_{t-1}^\phi + s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} b_{t-1}^\phi + \alpha \varepsilon_t$ $b_t = b_{t-1}^\phi + \beta \varepsilon_t / \ell_{t-1}$ $s_t = s_{t-m} + \gamma \varepsilon_t$	$y_t = \ell_{t-1} b_{t-1}^\phi s_{t-m} + \varepsilon_t$ $\ell_t = \ell_{t-1} b_{t-1}^\phi + \alpha \varepsilon_t / s_{t-m}$ $b_t = b_{t-1}^\phi + \beta \varepsilon_t / (s_{t-m} \ell_{t-1})$ $s_t = s_{t-m} + \gamma \varepsilon_t / (\ell_{t-1} b_{t-1}^\phi)$

Source : Hyndman, R.J., & Athanasopoulos, G. (2018)

# Erreurs multiplicatives

Trend	Seasonal		
	N	A	M
<b>N</b>	$y_t = \ell_{t-1}(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}(1 + \alpha\varepsilon_t)$	$y_t = (\ell_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1} + \alpha(\ell_{t-1} + s_{t-m})\varepsilon_t$ $s_t = s_{t-m} + \gamma(\ell_{t-1} + s_{t-m})\varepsilon_t$	$y_t = \ell_{t-1}s_{t-m}(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}(1 + \alpha\varepsilon_t)$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$
<b>A</b>	$y_t = (\ell_{t-1} + b_{t-1})(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1} + b_{t-1} + \alpha(\ell_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t$ $b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t$ $s_t = s_{t-m} + \gamma(\ell_{t-1} + b_{t-1} + s_{t-m})\varepsilon_t$	$y_t = (\ell_{t-1} + b_{t-1})s_{t-m}(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = b_{t-1} + \beta(\ell_{t-1} + b_{t-1})\varepsilon_t$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$
<b>A<sub>d</sub></b>	$y_t = (\ell_{t-1} + \phi b_{t-1})(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + \phi b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = \phi b_{t-1} + \beta(\ell_{t-1} + \phi b_{t-1})\varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1} + \phi b_{t-1} + \alpha(\ell_{t-1} + \phi b_{t-1} + s_{t-m})\varepsilon_t$ $b_t = \phi b_{t-1} + \beta(\ell_{t-1} + \phi b_{t-1} + s_{t-m})\varepsilon_t$ $s_t = s_{t-m} + \gamma(\ell_{t-1} + \phi b_{t-1} + s_{t-m})\varepsilon_t$	$y_t = (\ell_{t-1} + \phi b_{t-1})s_{t-m}(1 + \varepsilon_t)$ $\ell_t = (\ell_{t-1} + \phi b_{t-1})(1 + \alpha\varepsilon_t)$ $b_t = \phi b_{t-1} + \beta(\ell_{t-1} + \phi b_{t-1})\varepsilon_t$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$
<b>M</b>	$y_t = \ell_{t-1}b_{t-1}(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}b_{t-1}(1 + \alpha\varepsilon_t)$ $b_t = b_{t-1}(1 + \beta\varepsilon_t)$	$y_t = (\ell_{t-1}b_{t-1} + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}b_{t-1} + \alpha(\ell_{t-1}b_{t-1} + s_{t-m})\varepsilon_t$ $b_t = b_{t-1} + \beta(\ell_{t-1}b_{t-1} + s_{t-m})\varepsilon_t/\ell_{t-1}$ $s_t = s_{t-m} + \gamma(\ell_{t-1}b_{t-1} + s_{t-m})\varepsilon_t$	$y_t = \ell_{t-1}b_{t-1}s_{t-m}(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}b_{t-1}(1 + \alpha\varepsilon_t)$ $b_t = b_{t-1}(1 + \beta\varepsilon_t)$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$
<b>M<sub>d</sub></b>	$y_t = \ell_{t-1}b_{t-1}^\phi(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}b_{t-1}^\phi(1 + \alpha\varepsilon_t)$ $b_t = b_{t-1}^\phi(1 + \beta\varepsilon_t)$	$y_t = (\ell_{t-1}b_{t-1}^\phi + s_{t-m})(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}b_{t-1}^\phi + \alpha(\ell_{t-1}b_{t-1}^\phi + s_{t-m})\varepsilon_t$ $b_t = b_{t-1}^\phi + \beta(\ell_{t-1}b_{t-1}^\phi + s_{t-m})\varepsilon_t/\ell_{t-1}$ $s_t = s_{t-m} + \gamma(\ell_{t-1}b_{t-1}^\phi + s_{t-m})\varepsilon_t$	$y_t = \ell_{t-1}b_{t-1}^\phi s_{t-m}(1 + \varepsilon_t)$ $\ell_t = \ell_{t-1}b_{t-1}^\phi(1 + \alpha\varepsilon_t)$ $b_t = b_{t-1}^\phi(1 + \beta\varepsilon_t)$ $s_t = s_{t-m}(1 + \gamma\varepsilon_t)$

Source : Hyndman, R.J., & Athanasopoulos, G. (2018)



# Sous R

---

Pour les objets `ts` : `forecast::ets()` avec paramètre `damped = FALSE` ou `damped = TRUE`.

Pour les objets `tsibble` : `fable::ETS()` avec fonctions `error()`, `trend()` et `season()`

```
library(fable)
as_tsibble(USAccDeaths) %>%
  model(ETS(value ~ season("A")))
```

```
# A mable: 1 x 1
  `ETS(value ~ season("A"))`
      <model>
1      <ETS(A,N,A)>
```

# Sommaire

---

## 1. Modèles de prévision simples

## 2. Modèles exponentiels

## 3. Résidus et qualité des prévisions

### 3.1 Analyse des résidus

### 3.2 Analyse des prévisions

### 3.3 Critères d'information

## 4. Conclusion

# Analyse des résidus

---

On distingue deux types de prévisions :

- Prévisions *in-sample*, dans l'échantillon, *fitted values* : paramètres estimés sur l'ensemble des données
- Prévisions *out-of-sample*, hors échantillon : on reproduit le processus de prévision  $\hat{y}_{t+h|t}$  permet de vérifier les problèmes de sur-ajustement

# Prévisions *in-sample*

---

Résidus :  $e_t = y_t - \hat{y}_t$

Hypothèses :

- $(e_t)$  non corrélés (sinon il reste de l'information qui auraient dû être prise en compte dans la prévision)
- $(e_t)$  de moyenne nulle : sinon prévisions biaisées

# Prévisions *in-sample*

---

Résidus :  $e_t = y_t - \hat{y}_t$

Hypothèses :

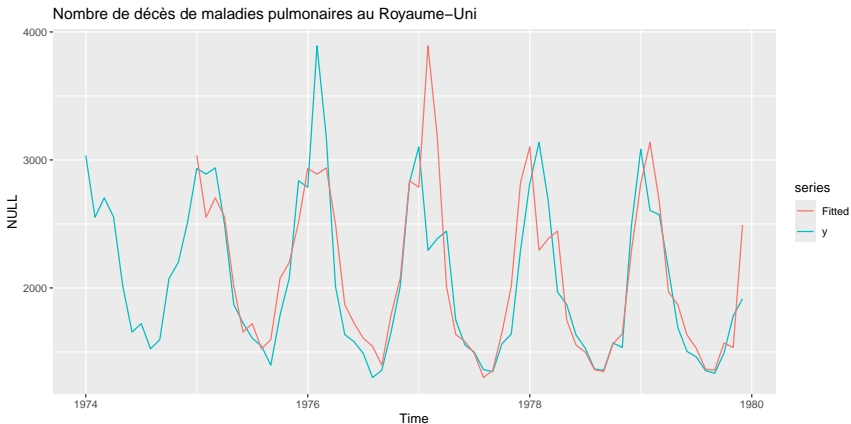
- $(e_t)$  non corrélés (sinon il reste de l'information qui auraient dû être prise en compte dans la prévision)
- $(e_t)$  de moyenne nulle : sinon prévisions biaisées

Hypothèses utiles pour la construction d'intervalles de confiance

- $(e_t)$  ont une variance constante
- $(e_t)$  suivent une loi normale

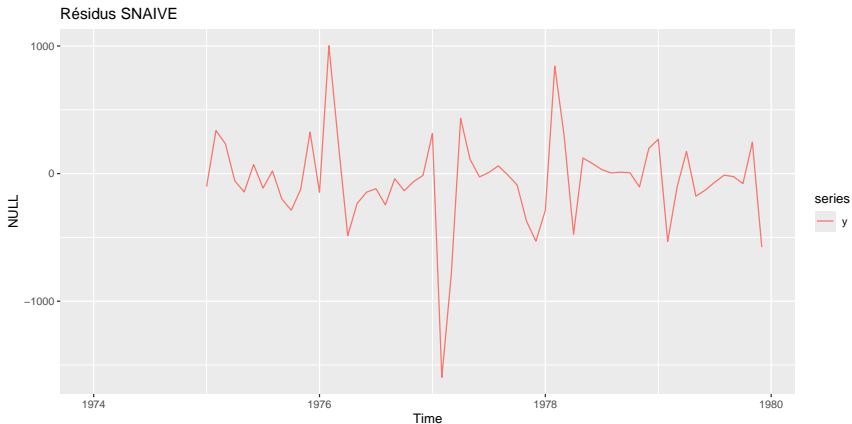
# Exemple

```
autoplot(ldeaths, y = "NULL",  
         main = "Nombre de décès de maladies pulmonaires au Royaume-Uni",  
         series = "y") +  
  autolayer(fitted(snaive(ldeaths)), series = "Fitted")
```



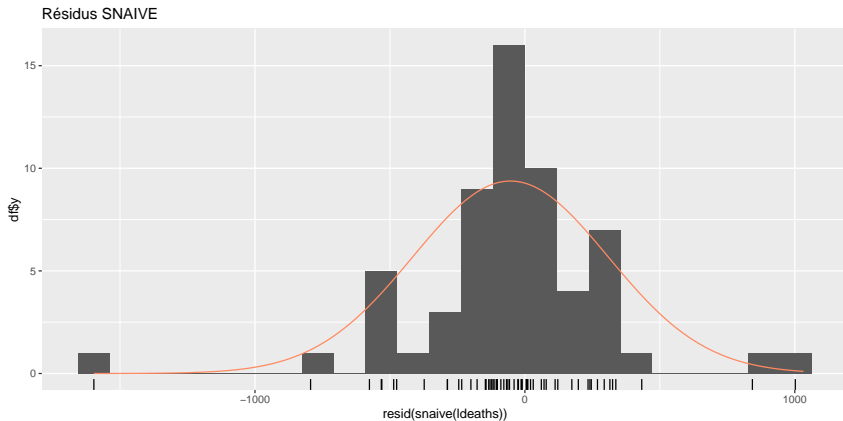
# Exemple

```
autoplot(resid(snaive(ldeaths)), y = "NULL",  
         main = "Résidus SNAIVE",  
         series = "y")
```



# Exemple

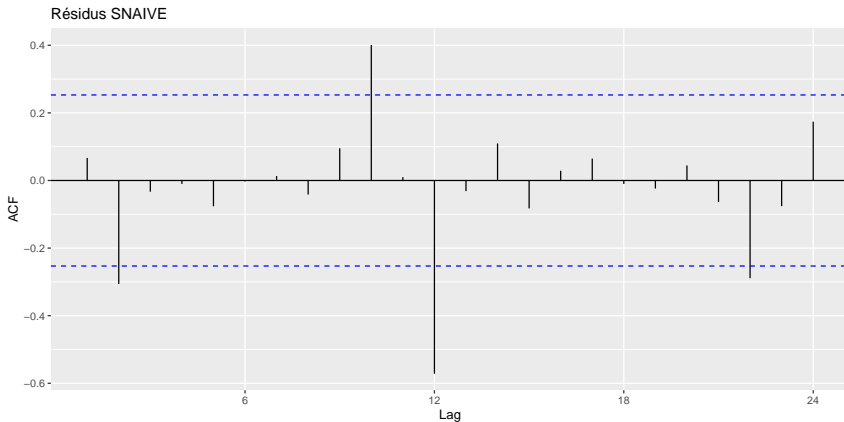
```
gghistogram(resid(snaive(ldeaths)), add.normal = TRUE) +  
  labs(title = "Résidus SNAIVE")
```





# Exemple

```
ggAcf(resid(snaive(ldeaths)), add.normal = TRUE) +  
  labs(title = "Résidus SNAIVE")
```



## J'ai un pic sur l'ACF ma série est donc autocorrélée ?

L'ACF est un outils graphique simple pour vérifier ont les mêmes propriétés qu'un bruit blanc.

## J'ai un pic sur l'ACF ma série est donc autocorrélée ?

L'ACF est un outils graphique simple pour vérifier ont les mêmes propriétés qu'un bruit blanc.



Attention à ne surtout pas analyser la significativités de tous les retards des ACF et de conclure à l'autocorrélation dans le cas d'une présence d'un seul pic ?

## J'ai un pic sur l'ACF ma série est donc autocorrélée ?

L'ACF est un outils graphique simple pour vérifier ont les mêmes propriétés qu'un bruit blanc.



Attention à ne surtout pas analyser la significativités de tous les retards des ACF et de conclure à l'autocorrélation dans le cas d'une présence d'un seul pic ?

Rappel statistique : si l'on fait  $n$  tests indépendants au seuil  $\alpha$ , la probabilité qu'au moins un test soit erroné est :

$$1 - (1 - \alpha)^n$$

Pour  $n = 12$  et  $\alpha = 0,05$  on a  $1 - (1 - \alpha)^n \simeq 46 \%$ .

## J'ai un pic sur l'ACF ma série est donc autocorrélée ?

L'ACF est un outils graphique simple pour vérifier ont les mêmes propriétés qu'un bruit blanc.



Attention à ne surtout pas analyser la significativités de tous les retards des ACF et de conclure à l'autocorrélation dans le cas d'une présence d'un seul pic ?

Rappel statistique : si l'on fait  $n$  tests indépendants au seuil  $\alpha$ , la probabilité qu'au moins un test soit erroné est :

$$1 - (1 - \alpha)^n$$

Pour  $n = 12$  et  $\alpha = 0,05$  on a  $1 - (1 - \alpha)^n \simeq 46 \%$ .

Préférer les tests d'autocorrélation (test Portemanteau)

# ACF et tests Portemanteau (`Box.test()`)

Il existe également des tests d'autocorrélation :

- Box-Pierce

$$Q = T \sum_{k=1}^p \hat{\rho}(k)^2$$

- Ljung-Box (marche mieux sur petits échantillons)

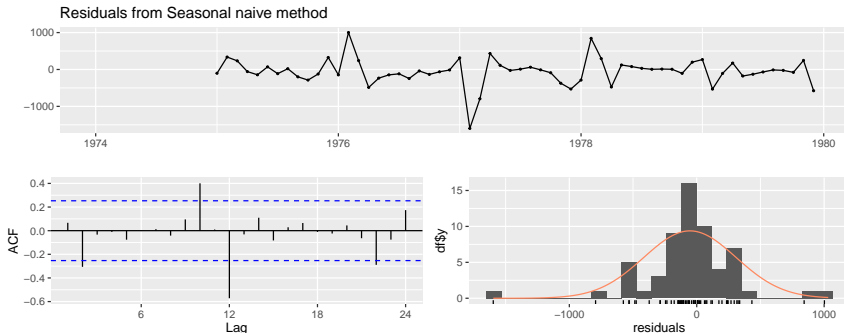
$$Q^* = T(T+1) \sum_{k=1}^p (T-k)^{-1} \hat{\rho}(k)^2$$

Paramètre  $p$  à choisir. Recommandation :  $p = 10$  pour séries non saisonnières,  $p = 2m$  sinon.

Sous ( $H_0$ ) ces quantités suivent  $\chi^2(p-K)$  avec  $K$  nb de paramètres dans le modèle

# Exemple

```
forecast::checkresiduals(snaive(ldeaths))
```



## Ljung-Box test

data: Residuals from Seasonal naive method

Q\* = 45.922, df = 14, p-value = 2.887e-05

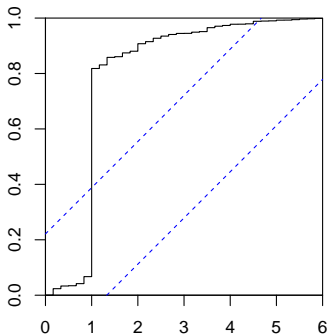
Model df: 0. Total lags used: 14

# Périodogramme cumulatif

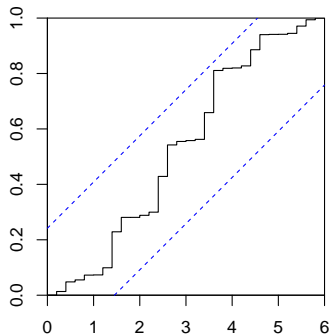
On peut aussi analyser les résidus avec le périodogramme cumulatif : proche d'une ligne droite pour un bruit blanc

```
par(mfrow = c(1,2))  
cpgram(ldeaths)  
cpgram(resid(snaive(ldeaths)))
```

Series: ldeaths



Series: resid(snaive(ldeaths))



frequency

frequency



# Mesure de la qualité de la prévisions

---

Plusieurs critères :

$$MAE = \text{moy}(|e_{T+h}|) \quad MSE = \text{moy}(e_{T+h}^2)$$

$$RMSE = \sqrt{\text{moy}(e_{T+h}^2)} \quad MAPE = 100 \text{mean}(|e_{T+h}|/|y_{T+h}|)$$

Les 3 premiers critères dépend de l'échelle mais pas le MAPE (mais valable si  $y_t \gg 0$ )

# Mesure de la qualité de la prévisions

Plusieurs critères :

$$MAE = moy(|e_{T+h}|) \quad MSE = moy(e_{T+h}^2)$$

$$RMSE = \sqrt{moy(e_{T+h}^2)} \quad MAPE = 100mean(|e_{T+h}|/|y_{T+h}|)$$

Les 3 premiers critères dépend de l'échelle mais pas le MAPE (mais valable si  $y_t \gg 0$ )

MASE proposé par Hyndman and Koehler (IJF, 2006) :

$$MASE = moy(|e_{T+h}|/Q) \quad \text{avec } Q \text{ une mesure stable de l'échelle de } y_t$$

$$\begin{cases} Q = \frac{1}{T-1} \sum_{t=2}^T |y_t - y_{t-1}| & \text{série non saisonnière} \\ Q = \frac{1}{T-m} \sum_{t=m+1}^T |y_t - y_{t-m}| & \text{série saisonnière} \end{cases}$$

# Validation croisée dans les séries temporelles

---

Prévisions en temps-réel : prévisions dynamiques en réactualisant les coefficients à chaque date.

➔ *leave-h-out cross-validation*

# Validation croisée dans les séries temporelles

Prévisions en temps-réel : prévisions dynamiques en réactualisant les coefficients à chaque date.

➔ *leave-h-out cross-validation*

On peut ensuite comparer les erreurs en utilisant un critère (e.g. RMSE) et un test (`forecast::dm.test()`).

Exemple LOOCV ( $h = 1$ ), modèle trimestriel :

2000T1
⋮
2021T4

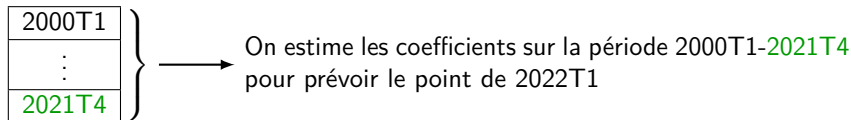
# Validation croisée dans les séries temporelles

Prévisions en temps-réel : prévisions dynamiques en réactualisant les coefficients à chaque date.

➔ *leave-h-out cross-validation*

On peut ensuite comparer les erreurs en utilisant un critère (e.g. RMSE) et un test (`forecast::dm.test()`).

Exemple LOOCV ( $h = 1$ ), modèle trimestriel :



On peut s'aider de `forecast::tsCV()`

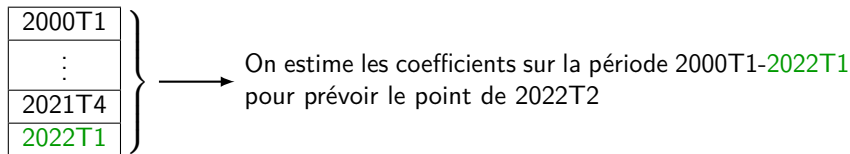
# Validation croisée dans les séries temporelles

Prévisions en temps-réel : prévisions dynamiques en réactualisant les coefficients à chaque date.

➔ *leave-h-out cross-validation*

On peut ensuite comparer les erreurs en utilisant un critère (e.g. RMSE) et un test (`forecast::dm.test()`).

Exemple LOOCV ( $h = 1$ ), modèle trimestriel :



On peut s'aider de `forecast::tsCV()`

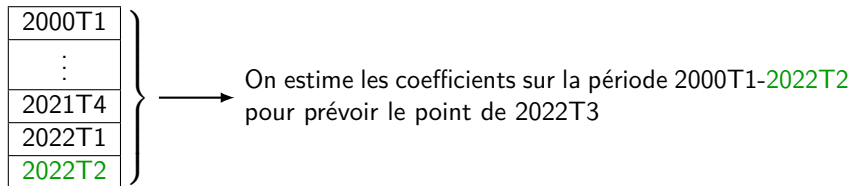
# Validation croisée dans les séries temporelles

Prévisions en temps-réel : prévisions dynamiques en réactualisant les coefficients à chaque date.

➔ *leave-h-out cross-validation*

On peut ensuite comparer les erreurs en utilisant un critère (e.g. RMSE) et un test (`forecast::dm.test()`).

Exemple LOOCV ( $h = 1$ ), modèle trimestriel :



On peut s'aider de `forecast::tsCV()`

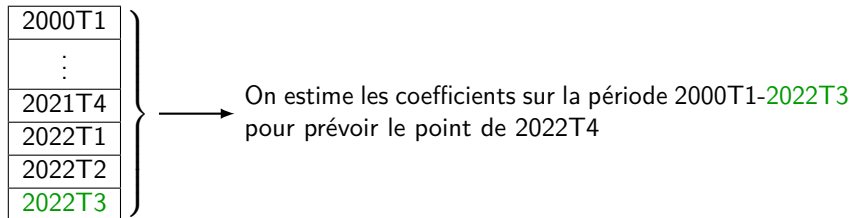
# Validation croisée dans les séries temporelles

Prévisions en temps-réel : prévisions dynamiques en réactualisant les coefficients à chaque date.

➔ *leave-h-out cross-validation*

On peut ensuite comparer les erreurs en utilisant un critère (e.g. RMSE) et un test (`forecast::dm.test()`).

Exemple LOOCV ( $h = 1$ ), modèle trimestriel :



On peut s'aider de `forecast::tsCV()`



# Sélection de modèles

---

Les modèles sont parfois sélectionnés en utilisant des critères d'information (AIC, BIC, HQ) qui sont des vraisemblances pénalisées. Retenir :

- Il faut les minimiser

# Sélection de modèles

---

Les modèles sont parfois sélectionnés en utilisant des critères d'information (AIC, BIC, HQ) qui sont des vraisemblances pénalisées. Retenir :

- Il faut les minimiser
- Ils sont définis à une constante additive/multiplicative près (qui peut changer en fonction des logiciels)

# Sélection de modèles

---

Les modèles sont parfois sélectionnés en utilisant des critères d'information (AIC, BIC, HQ) qui sont des vraisemblances pénalisées. Retenir :

- Il faut les minimiser
- Ils sont définis à une constante additive/multiplicative près (qui peut changer en fonction des logiciels)
- AICc effectue une correction lorsqu'il y a peu de données

# Sélection de modèles

---

Les modèles sont parfois sélectionnés en utilisant des critères d'information (AIC, BIC, HQ) qui sont des vraisemblances pénalisées. Retenir :

- Il faut les minimiser
- Ils sont définis à une constante additive/multiplicative près (qui peut changer en fonction des logiciels)
- AICc effectue une correction lorsqu'il y a peu de données
- Minimiser l'AIC est asymptotiquement équivalent à minimiser le LOOCV

# Sélection de modèles

---

Les modèles sont parfois sélectionnés en utilisant des critères d'information (AIC, BIC, HQ) qui sont des vraisemblances pénalisées. Retenir :

- Il faut les minimiser
- Ils sont définis à une constante additive/multiplicative près (qui peut changer en fonction des logiciels)
- AICc effectue une correction lorsqu'il y a peu de données
- Minimiser l'AIC est asymptotiquement équivalent à minimiser le LOOCV
- Minimiser le BIC est asymptotiquement équivalent à minimiser le L- $\nu$ -OCV avec  $\nu = T[1 - 1/(\log(T) - 1)]$  et à sélectionner le *true model*

# Sélection de modèles

---

Les modèles sont parfois sélectionnés en utilisant des critères d'information (AIC, BIC, HQ) qui sont des vraisemblances pénalisées. Retenir :

- Il faut les minimiser
- Ils sont définis à une constante additive/multiplicative près (qui peut changer en fonction des logiciels)
- AICc effectue une correction lorsqu'il y a peu de données
- Minimiser l'AIC est asymptotiquement équivalent à minimiser le LOOCV
- Minimiser le BIC est asymptotiquement équivalent à minimiser le L- $\nu$ -OCV avec  $\nu = T[1 - 1/(\log(T) - 1)]$  et à sélectionner le *true model*
- L'AIC a tendance à sur-ajuster le modèle et le BIC à le sous-ajuster

# Sélection de modèles

---

Les modèles sont parfois sélectionnés en utilisant des critères d'information (AIC, BIC, HQ) qui sont des vraisemblances pénalisées. Retenir :

- Il faut les minimiser
- Ils sont définis à une constante additive/multiplicative près (qui peut changer en fonction des logiciels)
- AICc effectue une correction lorsqu'il y a peu de données
- Minimiser l'AIC est asymptotiquement équivalent à minimiser le LOOCV
- Minimiser le BIC est asymptotiquement équivalent à minimiser le L- $\nu$ -OCV avec  $\nu = T[1 - 1/(\log(T) - 1)]$  et à sélectionner le *true model*
- L'AIC a tendance à sur-ajuster le modèle et le BIC à le sous-ajuster
- $\bar{R}^2$  a tendance à sélectionner trop de variables

# Sélection de modèles

Les modèles sont parfois sélectionnés en utilisant des critères d'information (AIC, BIC, HQ) qui sont des vraisemblances pénalisées. Retenir :

- Il faut les minimiser
- Ils sont définis à une constante additive/multiplicative près (qui peut changer en fonction des logiciels)
- AICc effectue une correction lorsqu'il y a peu de données
- Minimiser l'AIC est asymptotiquement équivalent à minimiser le LOOCV
- Minimiser le BIC est asymptotiquement équivalent à minimiser le L- $\nu$ -OCV avec  $\nu = T[1 - 1/(\log(T) - 1)]$  et à sélectionner le *true model*
- L'AIC a tendance à sur-ajuster le modèle et le BIC à le sous-ajuster
- $\bar{R}^2$  a tendance à sélectionner trop de variables
- Ne comparer les modèles avec critères d'information que s'ils sont

calculés sur les mêmes données ( ordre de différenciation ARIMA et ARIMA vs ETS)



# Sommaire

---

1. Modèles de prévision simples
2. Modèles exponentiels
3. Résidus et qualité des prévisions
- 4. Conclusion**

# Conclusions

---

- Dans beaucoup de cas les meilleurs modèles de prévision seront les plus simples : dernière valeur, valeur moyenne, valeur de la période précédente, etc.
- Le lissage exponentiel basé sur la description de la tendance et de la saisonnalité de la série
- La sélection d'un modèle peut se faire par un critère d'information ou par minimisation d'une statistique de validation croisée
- Les erreurs de prévision doivent être non corrélés et être de moyenne nulle. Pour la construction d'intervalles de confiance il faut en plus une variance constante et une loi normale

# Bibliographie

---

Hyndman, R.J., & Athanasopoulos, G. (2018) *Forecasting: principles and practice*, 2nd edition, OTexts: Melbourne, Australia. [OTexts.com/fpp2](https://otexts.com/fpp2). Accessed on june 2025.

Hyndman, R.J., & Athanasopoulos, G. (2021) *Forecasting: principles and practice*, 3rd edition, OTexts: Melbourne, Australia. [OTexts.com/fpp3](https://otexts.com/fpp3). Accessed on june 2025.