

A thick dark blue vertical bar runs along the left edge of the page. A yellow arrow-shaped banner points to the right from this bar, containing the text '2023/2024'. In the bottom-left corner, several thin, curved lines in dark blue and light grey sweep upwards and to the right.

2023/2024

Processi stocastici e analisi di serie temporali

Relazione sul progetto finale

SOMMARIO

1.INTRODUZIONE	2
2.SIMULAZIONE DI PROCESSI ARMA	3
2.1.Simulazione di un processo WN	3
2.2.Simulazioni processi AR(1) e MA(1)	5
2.3.Trend e stagionalità	8
3.ANALISI DI SERIE TEMPORALI	19
3.1.Nascite in Italia(1961-2022)	19
3.2.Produzione mondiale di energia elettrica	24
3.3.Decessi mensili nel Lazio (2011-2023)	28

1.INTRODUZIONE

La presente relazione si propone di condurre un'analisi approfondita su tre diverse serie temporali utilizzando il linguaggio di programmazione R.

L'obiettivo principale è quello di mettere in pratica le conoscenze teoriche acquisite durante il corso ed esplorare le potenzialità di R nel simulare e analizzare dati temporali.

Si partirà con un'illustrazione dettagliata di come simulare processi ARMA utilizzando R. In seguito, si procederà all'analisi approfondita dei metodi per individuare e stimare trend e stagionalità all'interno delle 3 serie temporali scelte.

2.1. SIMULAZIONE DI UN PROCESSO WHITE-NOISE

Un processo stocastico X_t è considerato WN se presenta le seguenti caratteristiche:

- Media zero: $E(X_t)=0 \quad \forall t$
- Varianza costante: $\text{Var}(X_t)=\sigma^2$, dove σ^2 è una costante positiva $\forall t$
- Funzione di autocorrelazione: $R(t_1, t_2) = \begin{cases} 0 & \text{se } t_1 \neq t_2 \\ \sigma^2 & \text{se } t_1 = t_2 \end{cases}$

Per simulare un tale processo in R può essere utilizzato il comando:

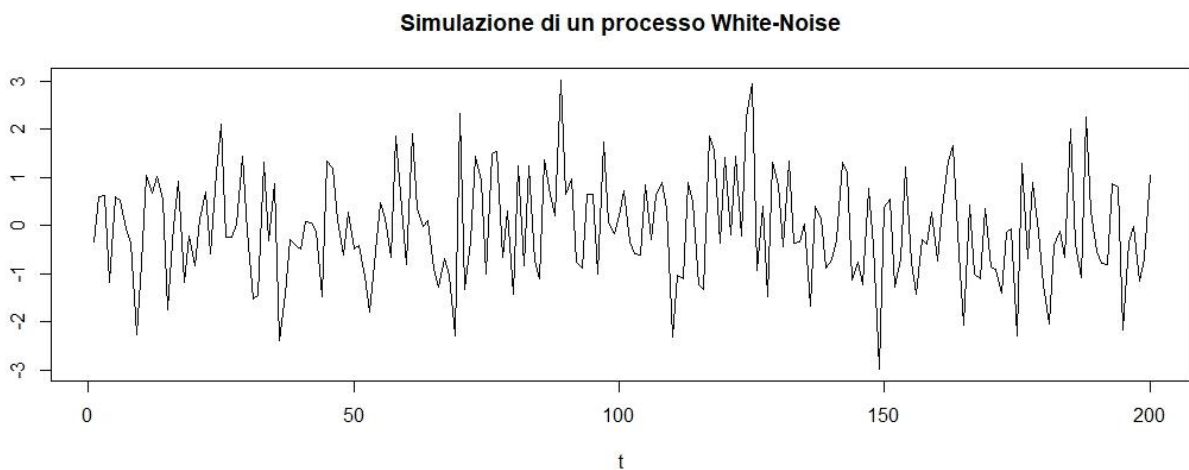
```
>> eps <- arima.sim(list(order=c(0,0,0)), n=200)
```

In questo caso `arima.sim()` è configurato con `order=c(0,0,0)`, il che significa che si sta simulando un processo ARIMA senza componente autoregressiva ($p=0$), senza componente di differenziazione ($d=0$) e senza componente a media mobile ($q=0$), che è equivalente ad un processo di rumore bianco.

La lunghezza della serie temporale è impostata su $n=200$.

Per visualizzare il grafico della serie storica generata da tale processo basta eseguire il seguente comando:

```
>> plot(eps, ylab="", xlab="t", main="Simulazione di un processo White-Noise")
```



Inoltre, è possibile generare i grafici relativi alle funzioni di autocorrelazione (ACF) e correlazione parziale (PACF) come riportato in seguito:

```
>> par(mfrow=c(2,1))
>> acf(eps, lag.max=20, main="ACF campionaria")
>> pacf(eps, lag.max=20, main="PACF campionaria")
```

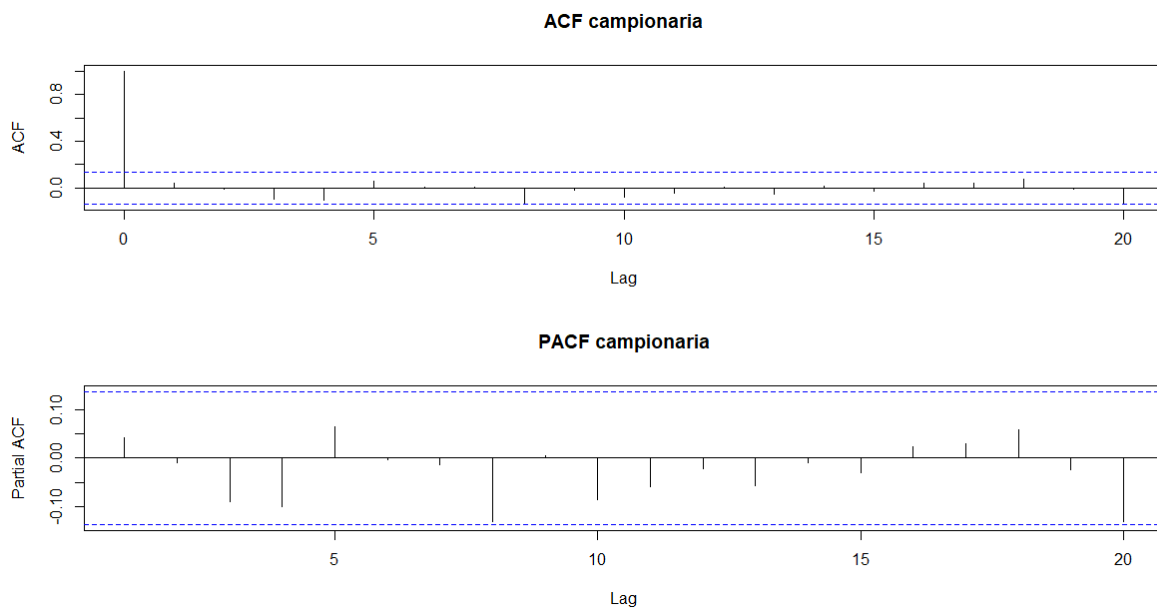
`par(mfrow = c(2,1))` imposta la disposizione dei grafici a 2 righe e 1 colonna per facilitare il confronto.

`acf()` genera il grafico dell'ACF mentre `pacf()` genera il grafico della PACF.

Entrambe le funzioni limitano i ritardi a un massimo di 20.

Questi grafici forniscono una rappresentazione visiva della correlazione tra le osservazioni a diversi

ritardi temporali.



Per valutare la significatività delle correlazioni osservate, è fondamentale prestare attenzione agli intervalli di confidenza (tratteggiati in blu).

In un processo di white noise ideale, ci si aspetta che ogni autocorrelazione, tranne quella a ritardo zero, cada all'interno degli intervalli di confidenza. Questo perché il white noise è caratterizzato dalla mancanza di correlazioni significative tra le osservazioni a diversi ritardi temporali.

Dall'analisi dei grafici sopra, emerge che non sono presenti picchi significativi al di fuori degli intervalli di confidenza, ad eccezione di un picco al ritardo 0 nel correlogramma dell'ACF. Tuttavia, questa osservazione è congruente con le aspettative.

2.2. SIMULAZIONI PROCESSI AR(1) E MA(1)

Un processo stocastico autoregressivo (AR) descrive una sequenza di osservazioni temporali in termini di relazioni lineari con le sue osservazioni passate.

L'equazione di un AR di ordine p può essere espressa come:

$$X_t = \mu + \varphi_1(X_{t-1} - \mu) + \dots + \varphi_p(X_{t-p} - \mu) + \varepsilon_t, \text{ dove:}$$

- ε_t è un processo white-noise
- X_{t-1}, \dots, X_{t-p} sono le osservazioni passate
- $\varphi_1, \dots, \varphi_p$ sono i coefficienti autoregressivi
- μ è una costante che in caso di stazionarietà coincide con la media del processo

Un processo a media mobile (MA), a differenza dei processi AR, descrive una sequenza di osservazioni temporali in termini di combinazioni lineari di white-noise.

L'equazione di un MA di ordine q può essere espressa come:

$$Y_t = \mu + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \dots + \theta_q \varepsilon_{t-q}, \text{ dove:}$$

- μ è la media del processo
- $\theta_1, \dots, \theta_q$ sono i coefficienti di media mobile

Procediamo a simulare $n=200$ realizzazioni del processo AR(1)

$$X_t = 0.2X_{t-1} + \varepsilon_t$$

e dei processi MA(1)

$$Y_t = \varepsilon_t + 0.5\varepsilon_{t-1}$$

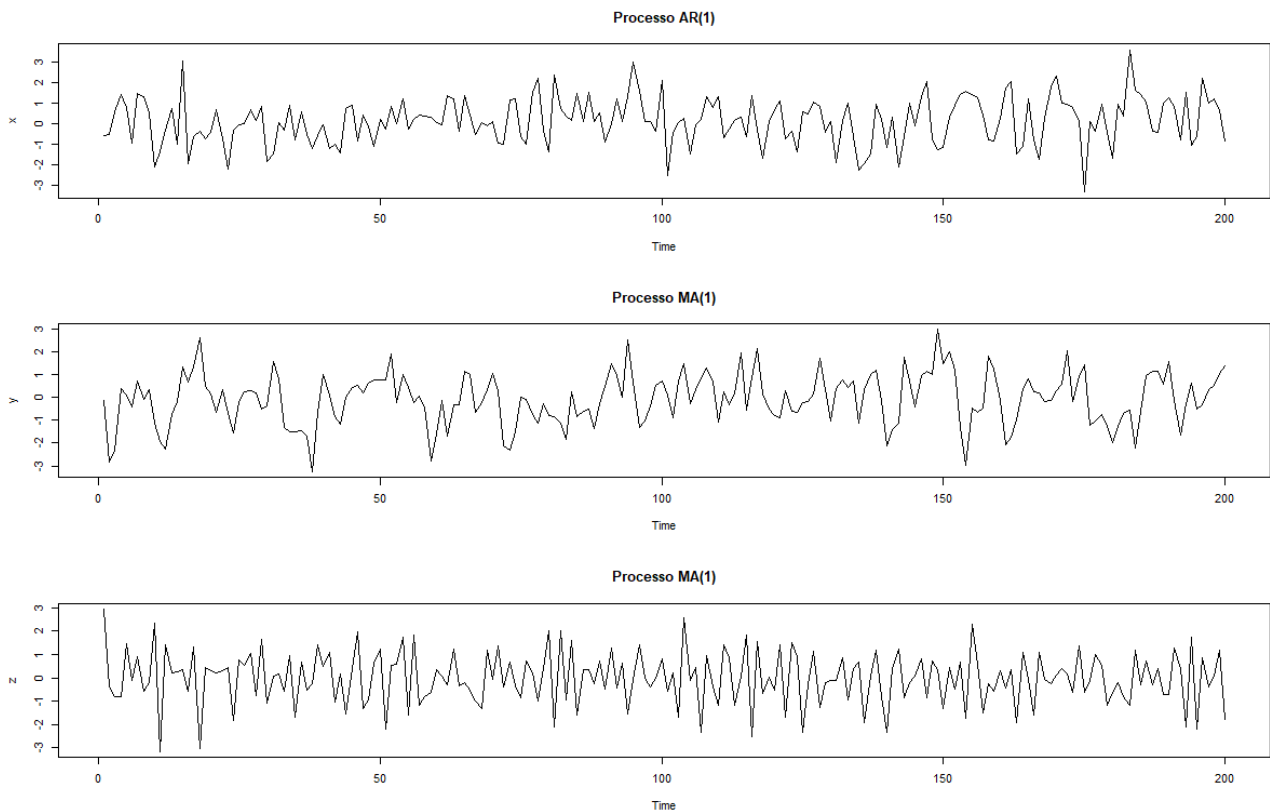
$$Z_t = \varepsilon_t - 0.5\varepsilon_{t-1}$$

La sequenza di istruzioni da inviare a R è:

```
>> x<-arma.sim(list(order=c(1,0,0), ar=0.2), n=200)
>> y<-arma.sim(list(order=c(0,0,1), ma=0.5), n=200)
>> z<-arma.sim(list(order=c(0,0,1), ma=-0.5), n=200)
```

e i grafici di tali serie sono stati generati tramite:

```
>> par(mfrow=c(3,1))
>> plot(x, type="l", main="Processo AR(1)")
>> plot(y, type="l", main="Processo MA(1)")
>> plot(z, type="l", main="Processo MA(1)")
```

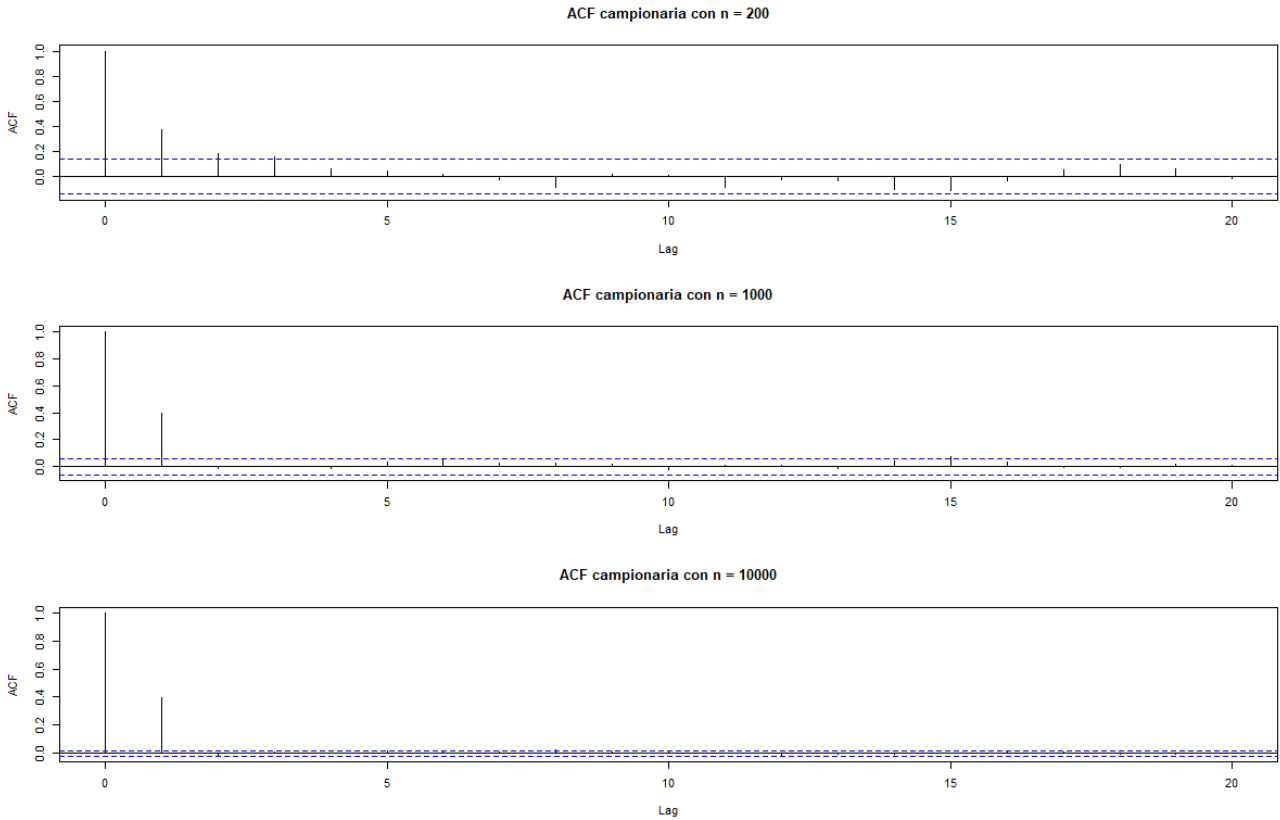


L'analisi dei grafici rivela una differenza marcata nel comportamento delle serie $\{Y_t\}$ e $\{Z_t\}$. La chiara discrepanza nei grafici suggerisce che il segno del coefficiente ε_{t-1} esercita un impatto sostanziale sulle dinamiche delle serie temporali, portando a una maggiore frequenza e rapidità di variazioni nella serie $\{Z_t\}$ rispetto a $\{Y_t\}$ e la causa di ciò si può comprendere dalle funzioni di autocorrelazione:

$$\rho_Y(h) = \begin{cases} 1 & \text{se } h = 0 \\ 0.4 & \text{se } h = 1 \\ 0 & \text{se } h = 2, 3, \dots \end{cases} \quad \rho_Z(h) = \begin{cases} 1 & \text{se } h = 0 \\ -0.4 & \text{se } h = 1 \\ 0 & \text{se } h = 2, 3, \dots \end{cases}$$

In R è possibile anche stimare queste funzioni di autocorrelazione. Ad esempio, si può provare a stimare la funzione di autocorrelazione campionaria $\hat{\rho}_Y$ della serie $\{Y_t\}$ con campioni di ampiezza crescente e verificare se si avvicina alla vera funzione di autocorrelazione ρ_Y . Per eseguire questa analisi, la sequenza di comandi in R è la seguente:

```
>> y1 <- arima.sim(list(order=c(0,0,1), ma=0.5), n=200)
>> y2 <- arima.sim(list(order=c(0,0,1), ma=0.5), n=1000)
>> y3 <- arima.sim(list(order=c(0,0,1), ma=0.5), n=10000)
>> par(mfrow=c(3,1))
>> acf(y1, lag.max=20, main="ACF campionaria con n = 200")
>> acf(y2, lag.max=20, main="ACF campionaria con n = 1000")
>> acf(y3, lag.max=20, main="ACF campionaria con n = 10000")
```



Esaminando i grafici, si osserva come le stime della funzione di autocorrelazione campionaria $\hat{\rho}_Y$ si avvicinano alla vera funzione di autocorrelazione ρ_Y all'aumentare della dimensione del campione.

Con R è anche possibile valutare esattamente la funzione di autocorrelazione di un qualsiasi processo ARMA. Ad esempio, riprendendo il processo Y_t e lanciando il seguente comando:

```
>> rho <-ARMAacf(ar=0, ma=c(0.5), lag.max=20)
```

otteniamo i seguenti risultati che corrispondono a quelli analitici:

[illegible]

2.3. TREND E STAGIONALITÀ

Le serie storiche possono presentare varie componenti, tra cui trend e stagionalità, che contribuiscono a renderle non stazionarie. La stazionarietà è una proprietà importante nelle analisi delle serie temporali, poiché semplifica l'interpretazione dei modelli e consente di fare previsioni più affidabili.

Stima del trend

L'uso di filtri lineari, come le medie mobili semplici e ponderate, è un metodo comune per stimare il trend in una serie temporale. La stima del trend in t mediante una media mobile semplice di ordine m coincide con

$$\hat{T}_t = \frac{1}{2m+1} \sum_{k=-m}^m x_{t+k}$$

Per dati aventi periodicità stagionale, la medesima stima è data da

$$\hat{T}_t = \frac{1}{4m} (x_{t-m} + 2x_{t-m+1} + \dots + 2x_{t+m-1} + x_{t+m})$$

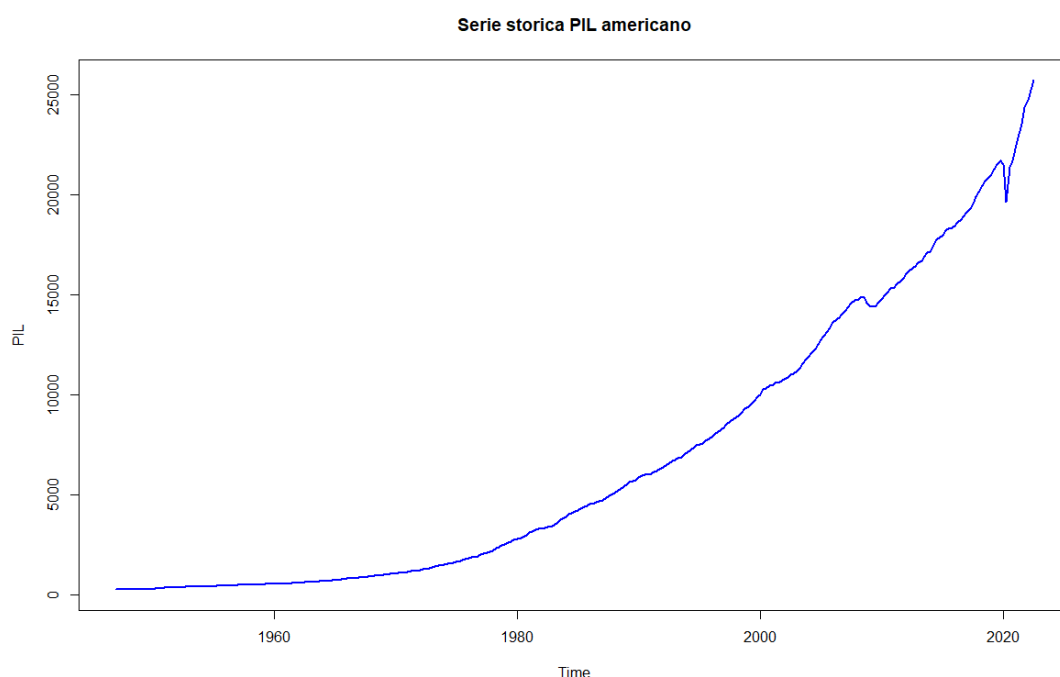
Per definire e valutare questi filtri lineari in R occorre usare la funzione `filter()`.

Supponiamo, ora, di applicare una media mobile semplice di ordine $m = 2$ per stimare il trend della serie storica di dati (trimestrali) sul PIL americano contenenti dati dal 1947 ad oggi. Carichiamo il dataset in R ed estrapoliamo i dati di interesse tramite le seguenti istruzioni:

```
>> pil.us <- read.csv(".\GDP.csv", sep = ",")
>> PIL <- ts(pil.us$GDP, start=c( 1947 ,1), freq= 4)
```

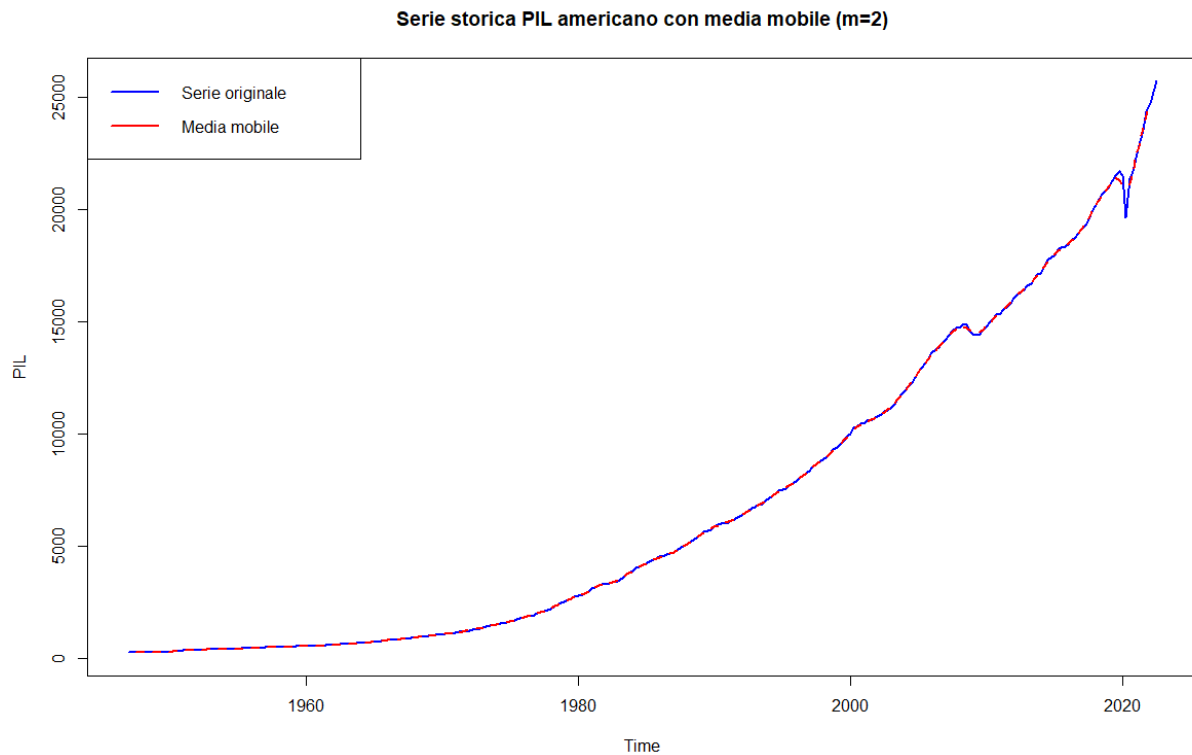
ed osserviamo l'andamento tramite il seguente comando

```
>> plot(PIL, main = "Serie storica PIL americano", col="blue",lwd=2)
```



Come evidenziato nel grafico, la serie presenta un trend crescente nel tempo. Per stimare il trend, come detto in precedenza, utilizziamo una media mobile centrata di ordine $m=2$ generata nel modo seguente:

```
>> m=2
>> gamma<-c(1/8,1/4,1/4,1/4,1/8)
>> ma<-filter(PIL, gamma, method="convolution", sides=2, circular=FALSE)
>> lines(ma, col="red", lty=2, lwd=2)
>> legend("topleft", legend = c("Serie originale", "Media mobile"), col = c("blue", "red"), lwd = 2)
```

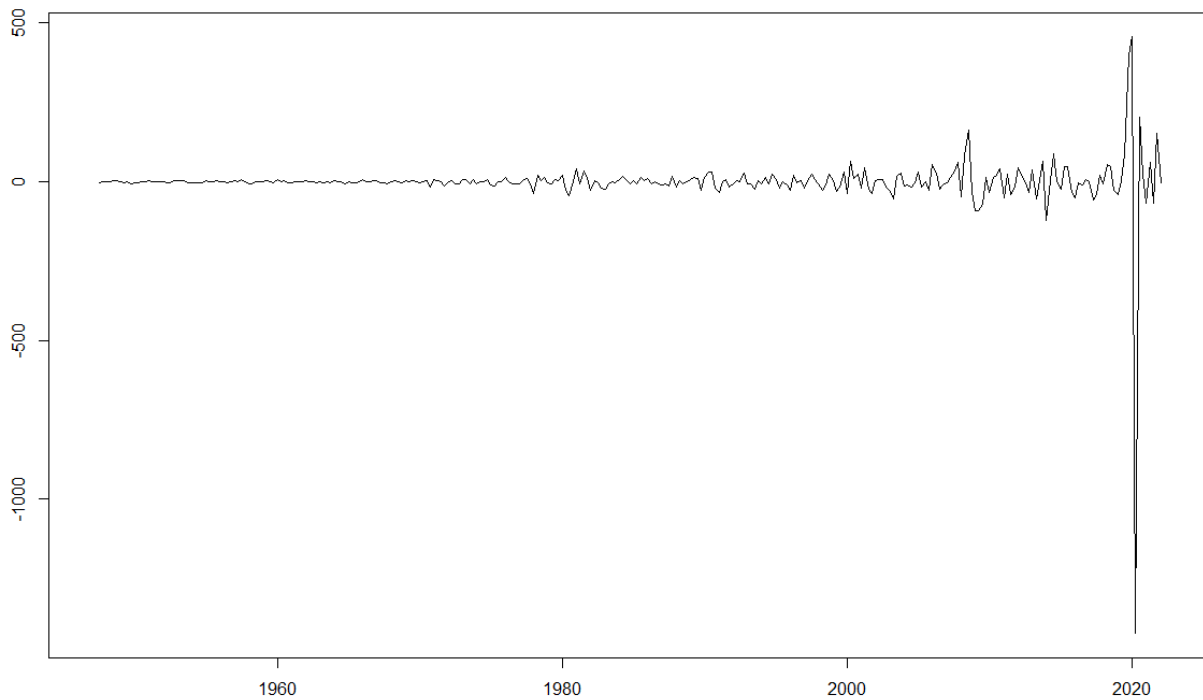


Per valutare la serie detrendizzata, supponiamo che la serie originaria sia descritta mediante un modello additivo. In questo caso, la serie detrendizzata è ottenuta mediante l'operazione $\hat{X}_t = X_t - \hat{T}_t$.

Le istruzioni eseguite in R sono le seguenti:

```
>> detrendx<-PIL-ma
>> plot(detrendx, ylab="",xlab="",main="Serie storica PIL americano detrendizzata")
```

Serie storica PIL americano detrendizzata-modello additivo

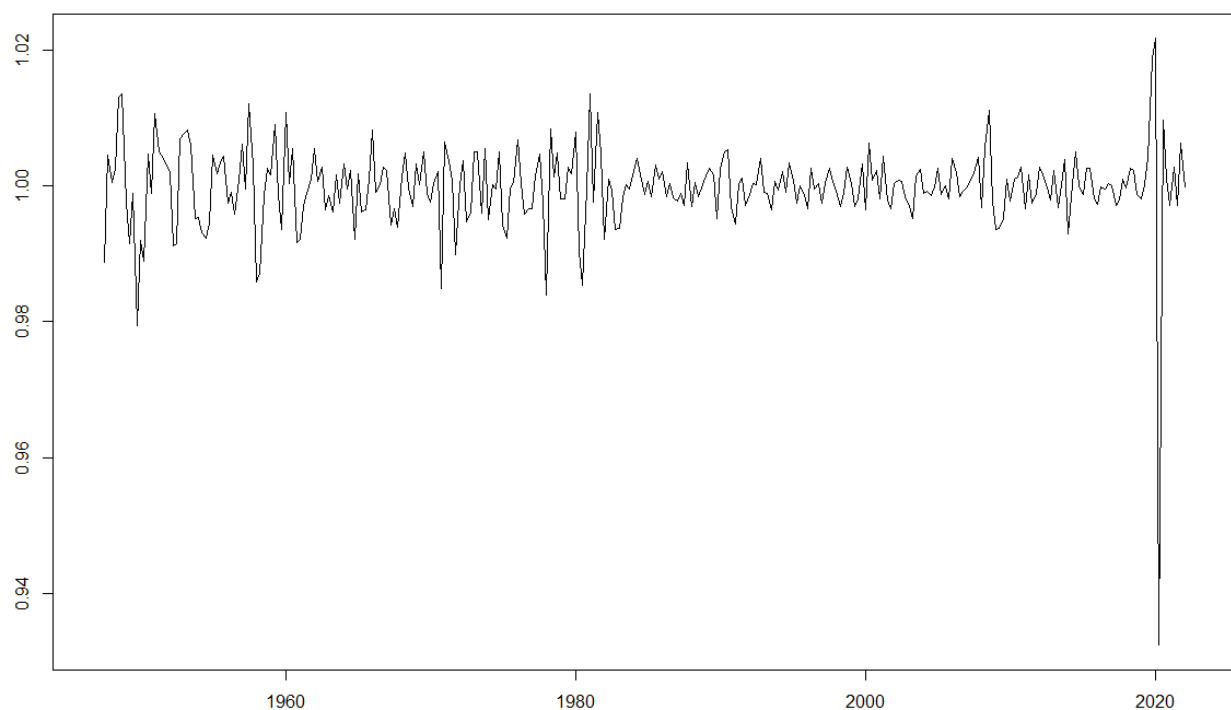


Il grafico risultante mostra che, mentre la media del processo oscilla intorno allo 0, la varianza sembra essere ancora in funzione del tempo, per cui tale processo non si può ritenere stazionario.

Per detrendizzare la serie in presenza di un modello moltiplicativo occorre effettuare la seguente divisione $\hat{X}_t = \frac{X_t}{\hat{\mu}_t}$ e in R viene specificato il comando

```
>> detrendxx<-PIL/ma
>> plot(detrendxx, ylab="", xlab="", main="Serie del PIL americano detrendizzata-modello moltiplicativo")
```

Serie del PIL americano detrendizzata-modello moltiplicativo



In R esiste anche una procedura automatizzata per detrendizzare e destagionalizzare una serie e si basa sulla funzione `decompose()`. Questa funzione, applicata ad una qualsiasi serie storica, restituisce la stima delle componenti di trend e stagionalità e la componente di errore residua che nel modello additivo è data da

$$\hat{E}_t = X_t - \hat{T}_t - \hat{S}_t$$

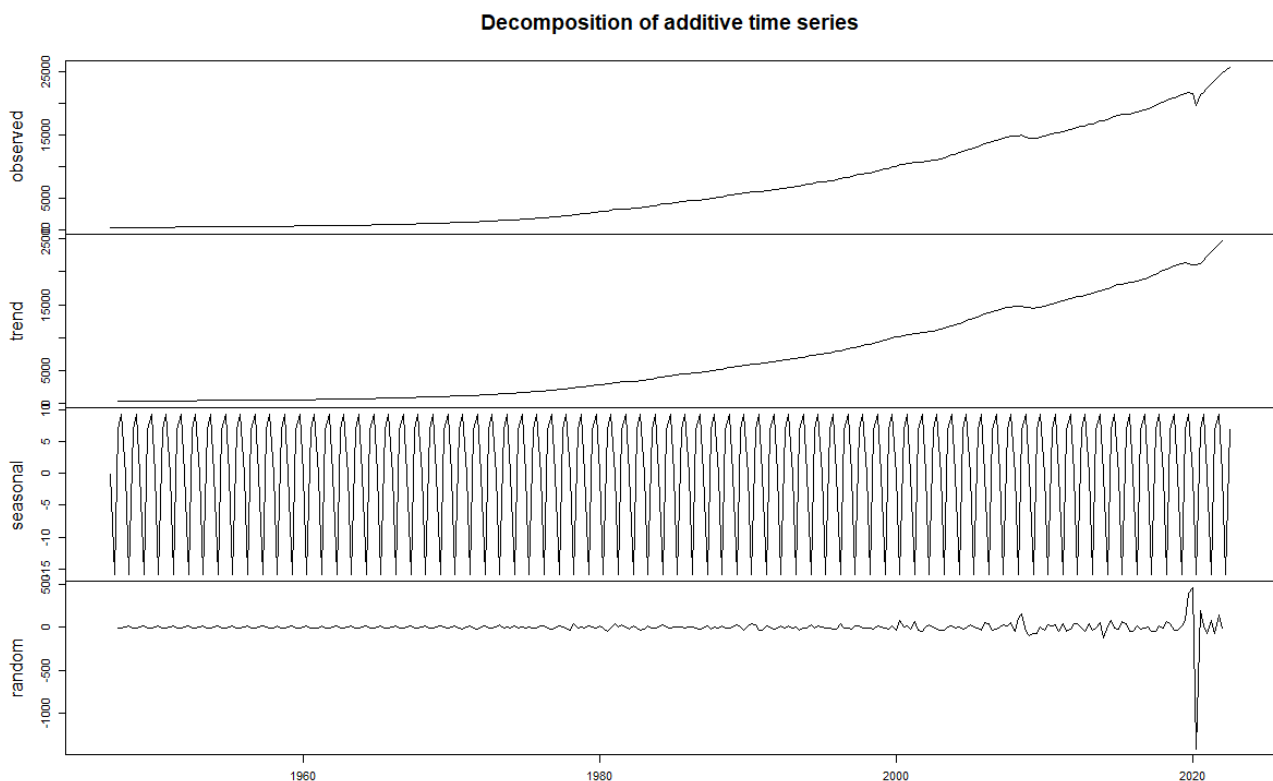
mentre nel modello moltiplicativo coincide con

$$\hat{E}_t = \frac{X_t}{\hat{T}_t \hat{S}_t}.$$

La decomposizione della serie storica del PIL statunitense mediante un modello additivo è realizzata in R mediante l'istruzione:

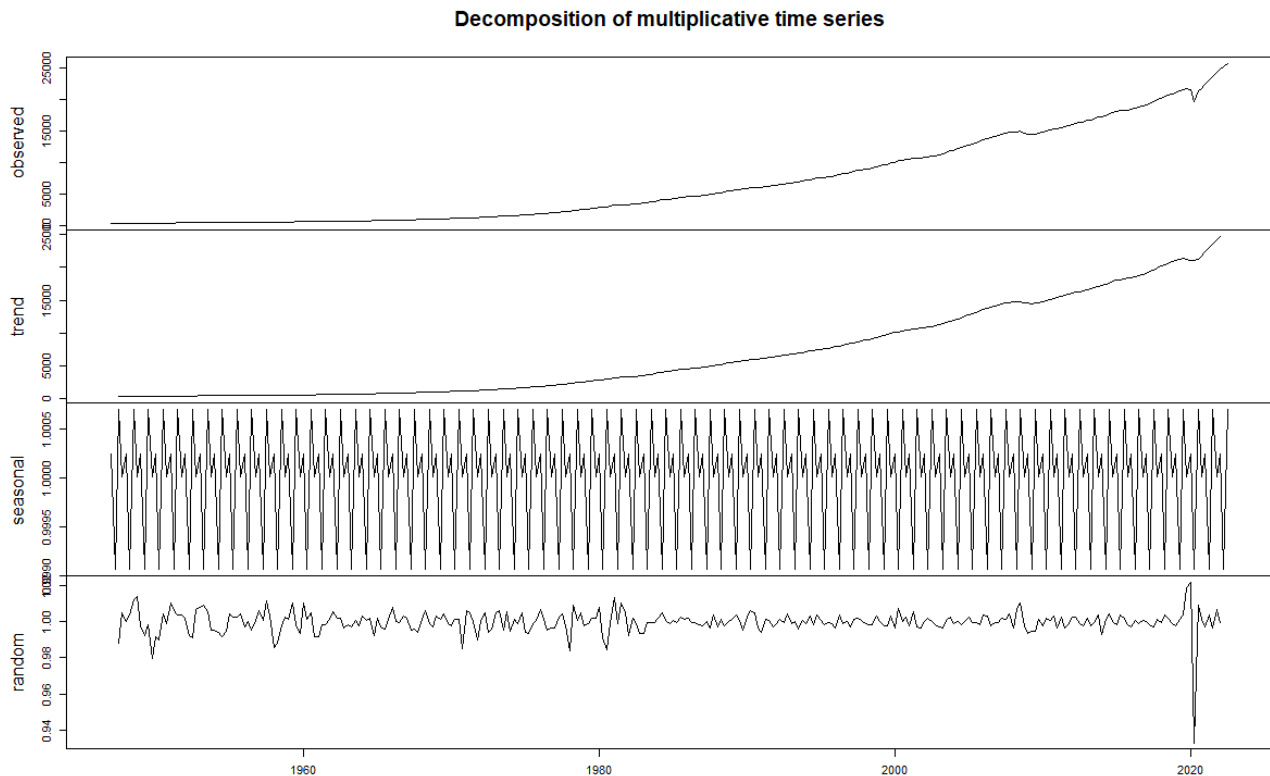
```
>> d.pil<-decompose(PIL, type="additive")
>> plot(d.pil)
```

L'istruzione `plot(d.pil)` fornisce 4 grafici, uno per ogni componente, compresa la serie originale.



Se invece abbiamo un modello moltiplicativo, le istruzioni da eseguire sono:

```
>> d.pil<-decompose(PIL, type="multiplicative")
>> plot(d.pil)
```

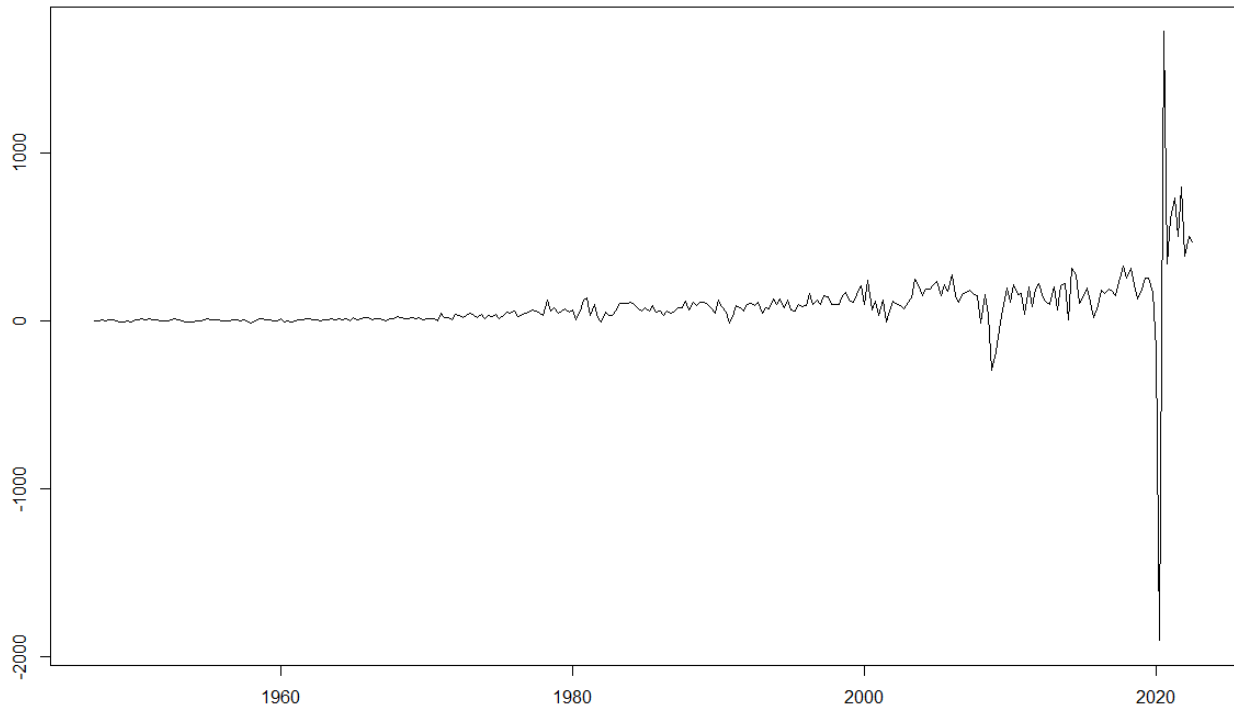


La detrendizzazione (o la destagionalizzazione) di una serie temporale può essere effettuata utilizzando anche il metodo della differenziazione, ossia applicando gli operatori differenza $\nabla^d = (1-B)^d$ e $\nabla_p^D = (1-B^p)^D$, dove B è l'operatore di ritardo e p la periodicità stagionale. Il comando che consente in R di generare una serie differenziata è `diff()`.

Riprendiamo la serie temporale del PIL americano e supponiamo che il trend nella serie sia lineare. Se si vuole detrendizzare, si può applicare l'operatore differenza di ordine $d = 1$ e l'istruzione da inviare a R:

```
>> dxt <- diff(PIL, lag=1, differences=1) #lag rappresenta la potenza da applicare a B
>> plot(dxt, main="Serie del PIL americano detrendizzata")
```

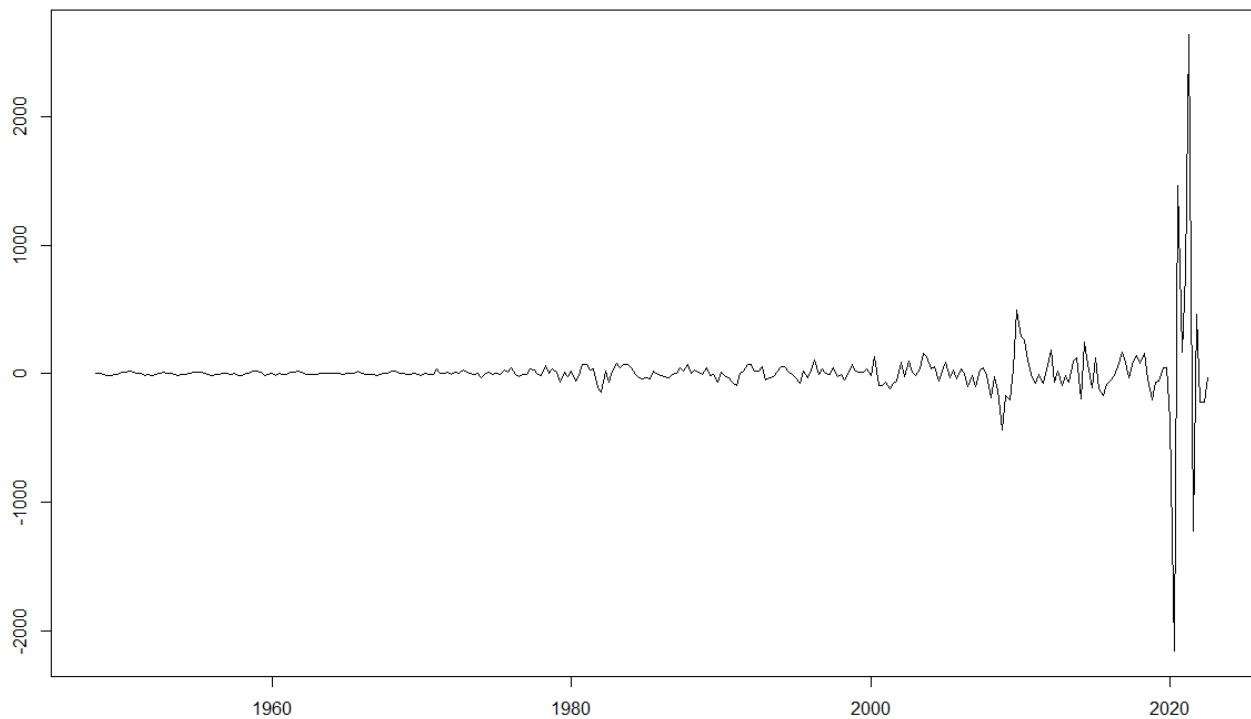
Serie del PIL americano detrendizzata



Se, oltre a detrendizzare, si intende anche destagionalizzare la serie si può procedere come segue

```
>> ddx<-diff(dxt, lag=4, differences=1)
```

Serie del PIL americano detrendizzata e destagionalizzata



Stima della stagionalità

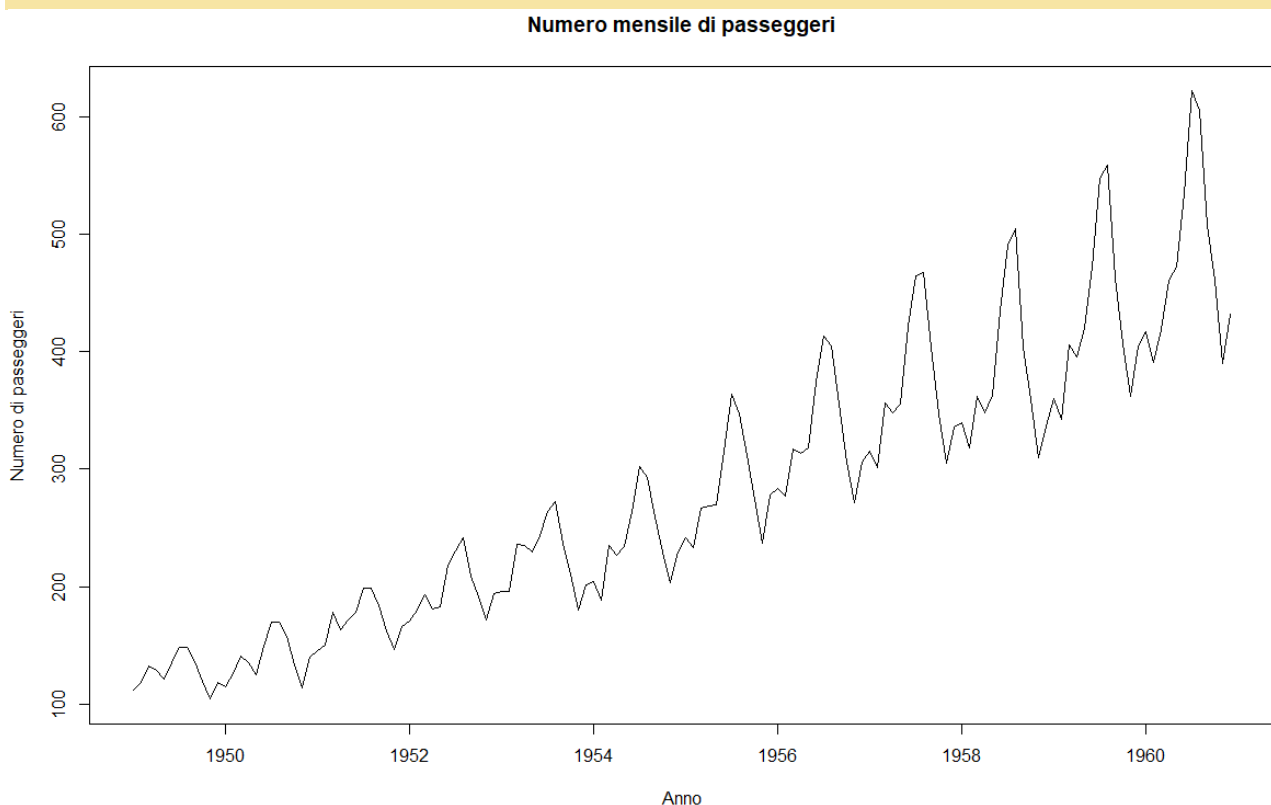
R mette a disposizione diverse funzioni per studiare a livello grafico le stagionalità di una serie, in particolare:

- **Monthplot():** questa funzione è comunemente utilizzata per visualizzare la distribuzione dei dati mensili, mettendo in evidenza eventuali tendenze o pattern stagionali.
- **Lag.plot():** consente di creare un grafico di dispersione per visualizzare la relazione tra un'osservazione e le sue versioni ritardate (lags) nella serie temporale. Può essere utile per esaminare la correlazione tra i valori della serie a diversi ritardi temporali.
- **Seasonal plot():** crea un grafico stagionale che può aiutare a identificare modelli stagionali nella serie temporale.

Per esaminare in dettaglio il loro funzionamento prendiamo come serie di riferimento AirPassengers già presente in R che si riferisce al numero mensile di passeggeri statunitensi imbarcati su voli di linea diretti verso l'estero tra gennaio 1949 e dicembre 1960.

Tale dataset è reperibile tramite il comando data():

```
>> data("AirPassengers")
>> plot(AirPassengers, main = "Numero mensile di passeggeri", xlab = "Anno", ylab = "Numero di passeggeri")
```

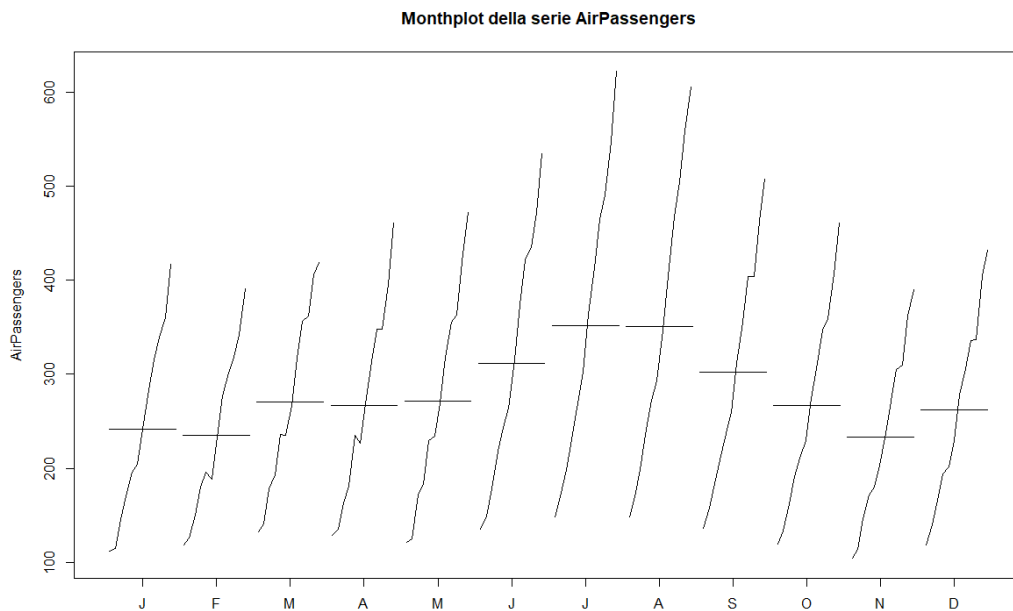


Dal grafico si notano un trend crescente e delle fluttuazioni cicliche stagionali.

Se si digita il comando

```
>> monthplot(AirPassengers, main="Monthplot della serie AirPassengers")
```

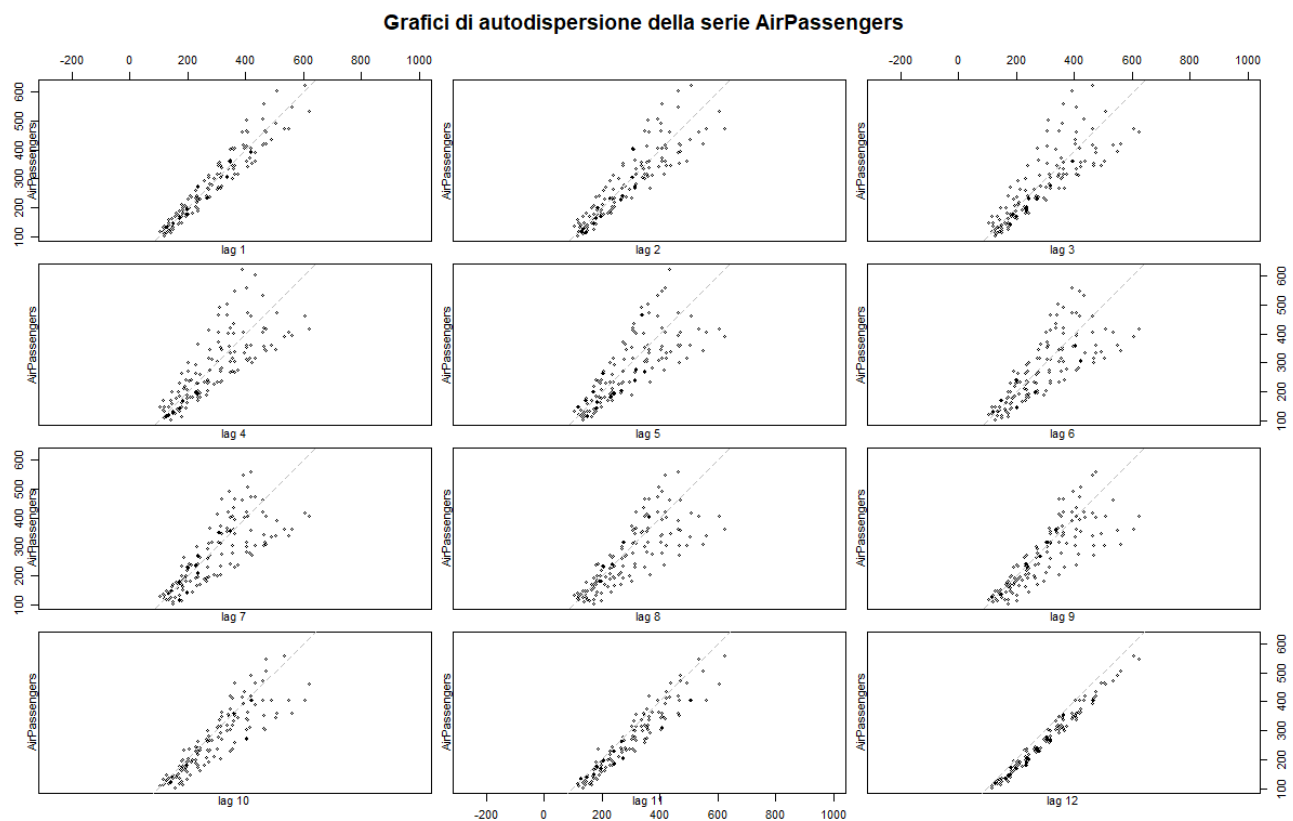
otteniamo il seguente grafico, dove le linee orizzontali corrispondono alle medie dei valori delle singole sottoserie corrispondenti a ciascuno dei 12 mesi dell'anno. Si può osservare che i valori medi sono più elevati nei mesi di Luglio e Agosto e si nota anche un deciso incremento del numero di passeggeri in ciascuno dei mesi dell'anno. Questa osservazione ci suggerisce che occorre applicare alla serie AirPassengers entrambi gli operatori di differenziazione.



Se si digita il comando

```
>> lag.plot(AirPassengers, set.lags=1:12, type="p", do.lines=FALSE, main="Grafici di autodispersione della serie AirPassengers")
```

con *set.lags* impostato da 1 a 12 verranno creati grafici di dispersione per ciascun lag specificato, mostrando la relazione tra i valori della serie originale e quelli della serie ritardata a diversi ritardi temporali (in questo caso, da 1 a 12 mesi). Questo è utile per identificare eventuali correlazioni a diversi ritardi e valutare la possibile presenza di pattern stagionali.



Ultimo strumento grafico che consente di estrapolare alcune informazioni utili circa la stagionalità di una serie storica è il *seasonal plot*.

Il seasonal plot è il grafico della serie storica limitata ad un singolo anno. Ciò consente di visualizzare il comportamento dei dati nel corso dell'anno e verificare se si manifestano regolarità che si ripetono nei vari anni di osservazione.

Lavorando sempre con il dataset AirPassengers, estrapoliamo le serie degli anni 1949, 1951, 1954, 1956, 1958 e 1960:

```
>> s1 <- window(AirPassengers, start=c(1949,1), end=c(1949,12))
>> s2 <- window(AirPassengers, start=c(1951,1), end=c(1951,12))
>> s3 <- window(AirPassengers, start=c(1954,1), end=c(1954,12))
>> s4 <- window(AirPassengers, start=c(1956,1), end=c(1956,12))
>> s5 <- window(AirPassengers, start=c(1958,1), end=c(1958,12))
>> s6 <- window(AirPassengers, start=c(1960,1), end=c(1960,12))
```

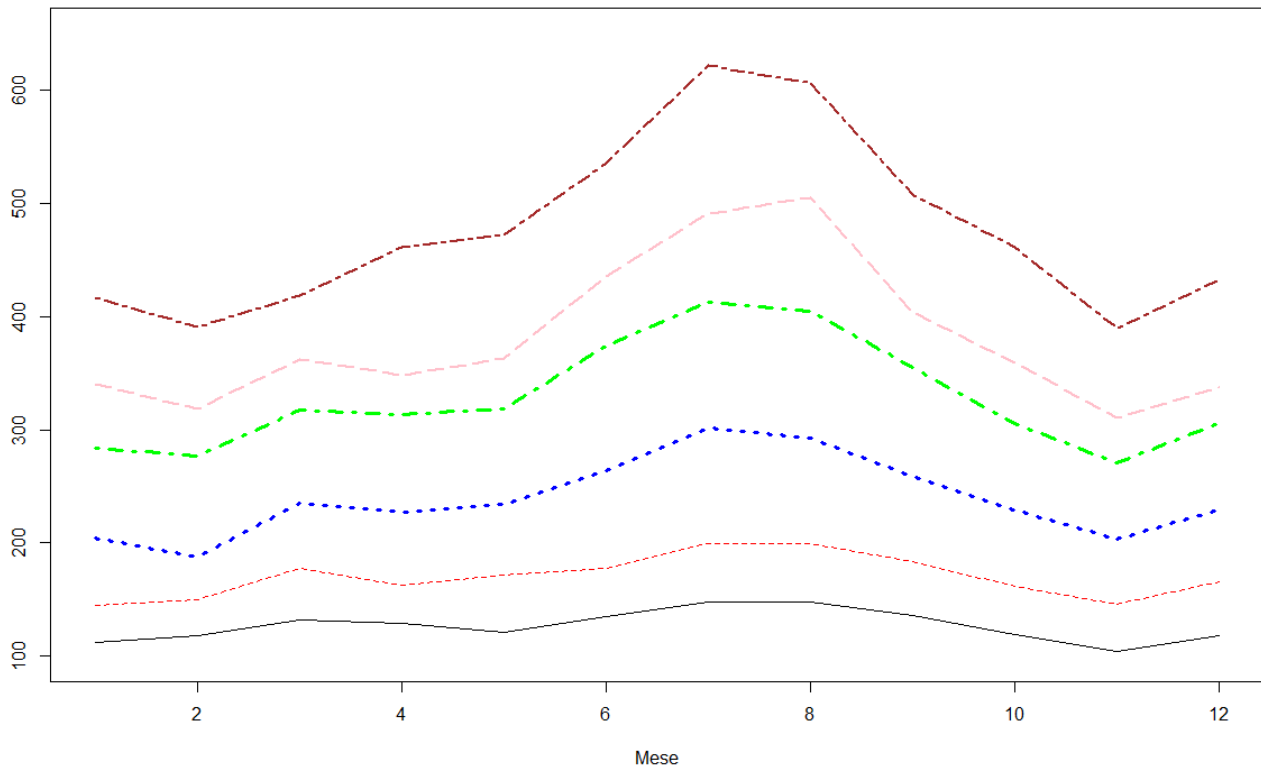
Le sei serie memorizzate da R sono costituite da 12 osservazioni ciascuna (corrispondenti ai 12 mesi dell'anno). Per generare il grafico occorre creare un vettore $x = (1, \dots, 12)$ che sarà utilizzato sull'asse delle ascisse. Dal momento che

```
>> range(AirPassengers)
Restituisce [1] 104 622
```

bisogna generare una finestra grafica che ricomprenda tutti i valori da 104 a 622 specificando in modo opportuno il parametro ylim:

```
x <- seq(1,12,1)
matplot(x, s1, type="l", ylim=c(100,650), ylab="", xlab="Mese", main="Seasonal plot della serie AirPassengers")
lines(x, s2, type="l", lty=2, col="red")
lines(x, s3, type="l", lty=3, col="blue", lwd=3)
lines(x, s4, type="l", lty=4, col="green", lwd=3)
lines(x, s5, type="l", lty=5, col="pink", lwd=2)
lines(x, s6, type="l", lty=6, col="brown", lwd=2)
```

Seasonal plot della serie AirPassengers



La linea continua in basso si riferisce ai 12 valori nella serie nell'anno 1949 e gli anni successivi si trovano su livelli sempre più elevati.

Il grafico suggerisce due conclusioni rilevanti: (a) la serie temporale manifesta una componente stagionale significativa che evidenzia picchi notevoli nei mesi di luglio e agosto per tutti gli anni osservati, con un andamento pressoché simile in ciascun anno; (b) i valori numerici della serie mostrano una tendenza crescente nel tempo, come evidenziato dal fatto che le curve tendono a collocarsi su livelli progressivamente più elevati con il passare degli anni. Inoltre, si nota un aumento dell'oscillazione tra i valori minimi e massimi della serie nel corso dell'anno, suggerendo la presenza di una componente di trend nella stagionalità.

A seguito di queste osservazioni occorre destagionalizzare e detrendizzare la serie. Per farlo si possono utilizzare i metodi visti in precedenza (`deff()` o `decompose()`).

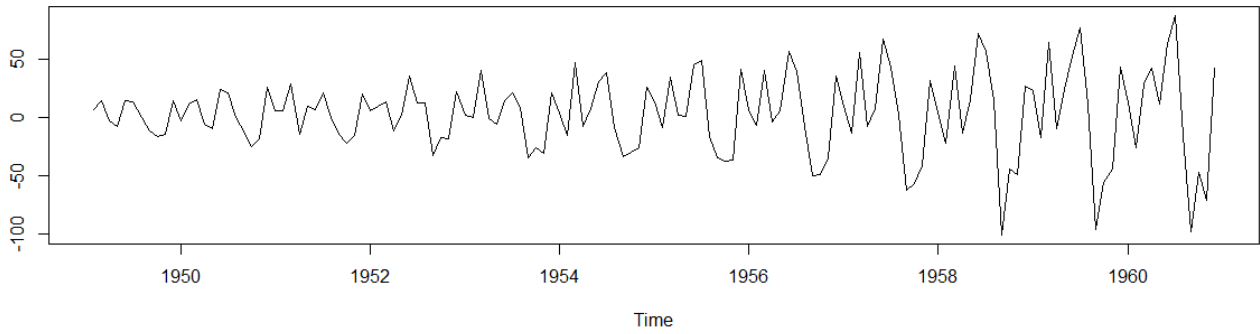
Scegliendo di differenziare la serie:

```
>> detrend <- diff(AirPassengers, lag=1, differences=1)
>> par(mfrow = c(2,1))
>> plot(detrend, main="Serie AirPassengers detrendizzata")
>> acf(detrend, lag.max = 20, main = "ACF campionaria serie AirPassengers detrendizzata")

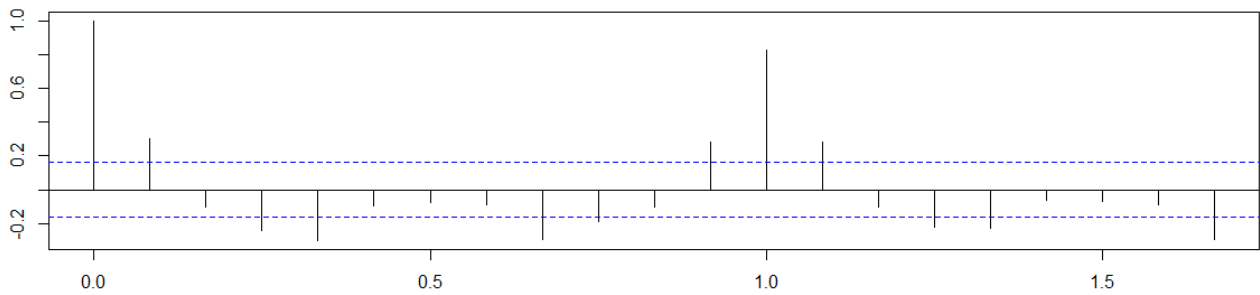
>> destag <- diff(detrend, lag=12, differences=1)
>> par(mfrow = c(2,1))
>> plot(destag, main="Serie AirPassengers detrendizzata e destagionalizzata")
>> acf(destag, lag.max = 20, main = "ACF campionaria serie AirPassengers detrendizzata e destagionalizzata")
```

otteniamo i seguenti grafici:

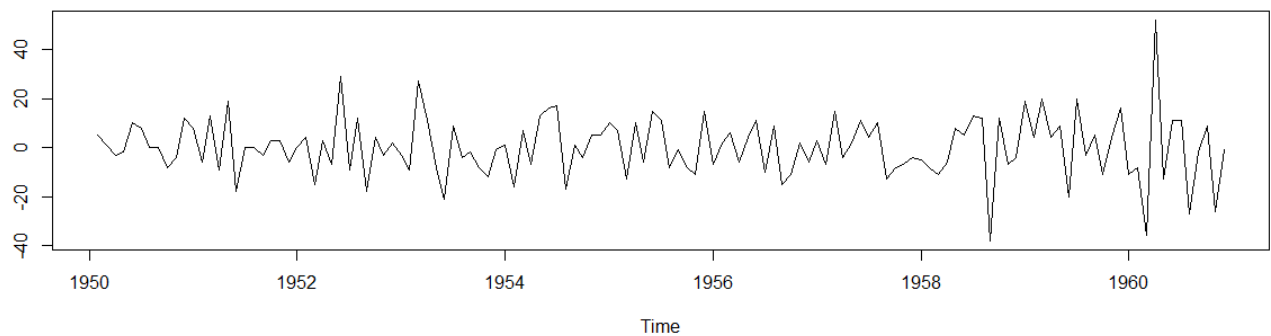
Serie AirPassengers detrendizzata



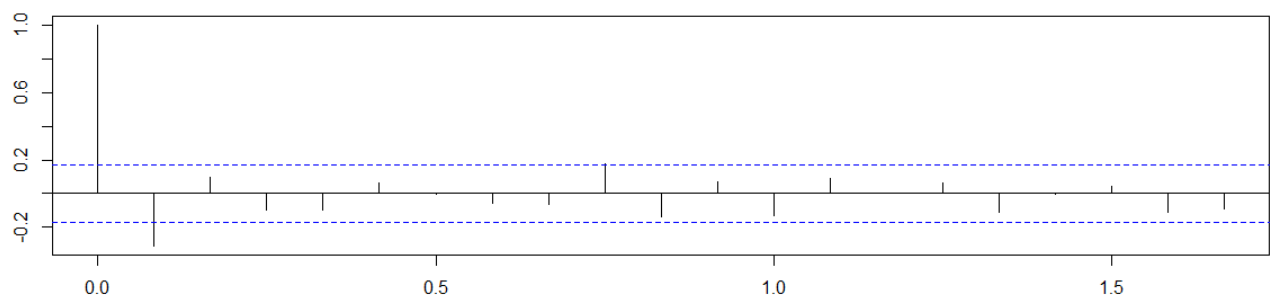
ACF campionaria serie AirPassengers detrendizzata



Serie AirPassengers detrendizzata e destagionalizzata



ACF campionaria serie AirPassengers detrendizzata e destagionalizzata



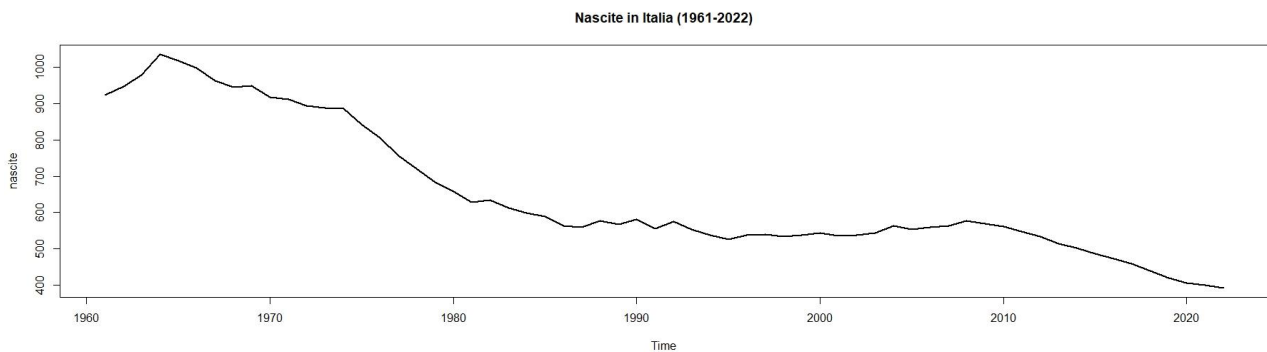
Il grafico della serie detrendizzata e destagionalizzata lascia presumere che il processo sia white-noise.

Nascite in Italia (1961-2022)

I dati sono tratti dal sito [serie storiche ISTAT](#)

Come evidenziato chiaramente nel seguente grafico, nei decenni recenti, l'Italia ha attraversato un significativo trend decrescente per le nascite.

L'asse delle ascisse delinea il periodo preso in considerazione dell'analisi (gli anni dal 1961 al 2022), mentre l'asse delle ordinate quantifica il numero di nascite in migliaia.



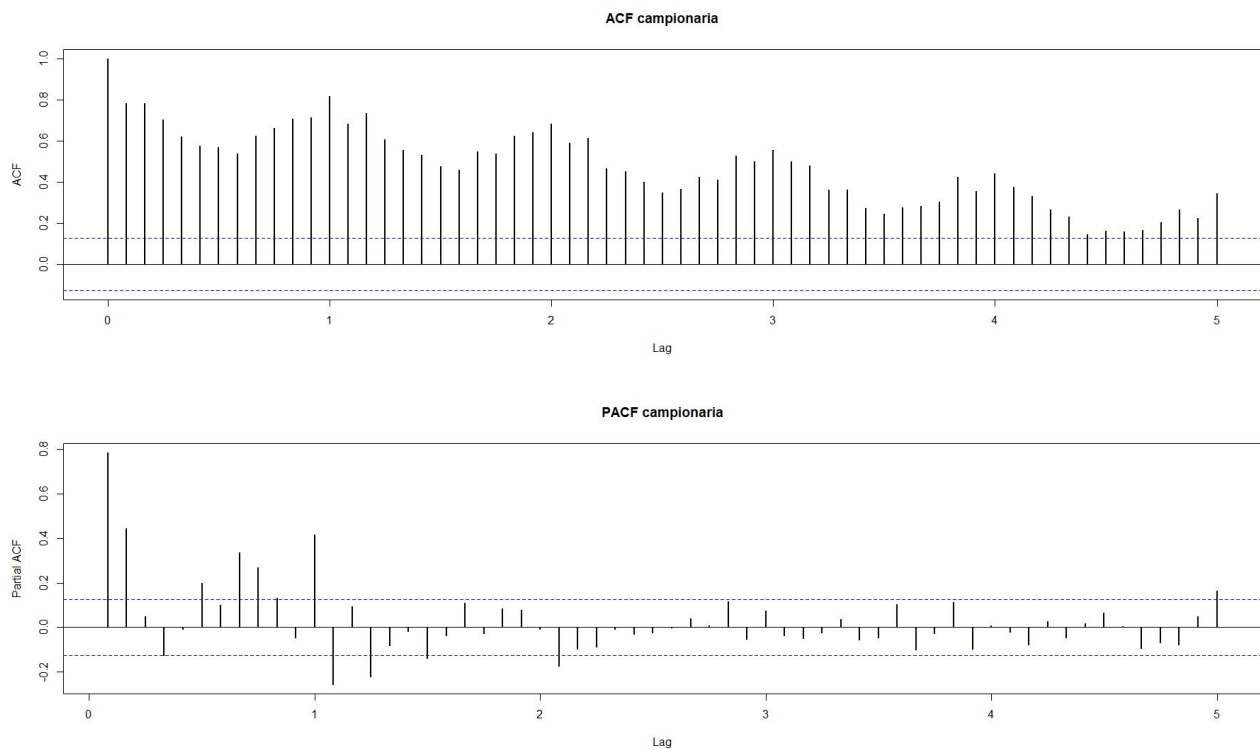
Nel 1964, l'Italia ha raggiunto un apice demografico con un notevole numero di nascite, totalizzando ben 1.035.207 unità. Tuttavia, questa fase positiva è stata seguita da un lungo periodo di declino demografico. Inizialmente, il declino è stato moderato fino al 1974, ma successivamente si è accentuato nel periodo dal 1975 al 1986.

Dopo questa fase di contrazione più marcata, l'Italia ha attraversato una successiva fase di variazioni demografiche ridotte con un minimo toccato nel 1995 (526.064) e un massimo nel 2008 (576.659).

Dal 2009, il Paese ha intrapreso una nuova fase di decrescita demografica, caratterizzata da una pendenza simile a quella osservata negli anni '60. Questo declino sembra persistere, e anno dopo anno si superano i record negativi di nascite. Nel 2022, il numero di nati è ulteriormente diminuito, scendendo a 392.598 rispetto ai 400.249 del 2021.

Spostiamo la nostra attenzione su un livello più dettagliato, adottando una prospettiva mensile attraverso l'analisi del grafico delle nascite avvenute dal 01/2003 al 11/2023.

```
>> nat<- read.csv(".\\nasciteMensiliItalia.csv", sep = ",")
>> nasciteMensili <- ts(nat$NASCITE,start=c(2003 , 1),end=c(2022,12), freq= 12)
>> acf(nasciteMensili, lag.max = 60, main = "ACF campionaria")
>> pacf(nasciteMensili, lag.max = 60, main= "PACF campionaria", lwd=2)
```



Analizzando la funzione di autocorrelazione, si osserva un chiaro andamento oscillatorio che indica la presenza di pattern periodici nei dati. Questo suggerisce la possibilità di una struttura temporale non casuale ed esclude la possibilità che la serie sia un white-noise.

Per confermare questa osservazione eseguiamo il test di Ljung-Box.

```
>> Box.test(nasciteMensili, lag=60, type = "Ljung-Box")
```

Box-Ljung test

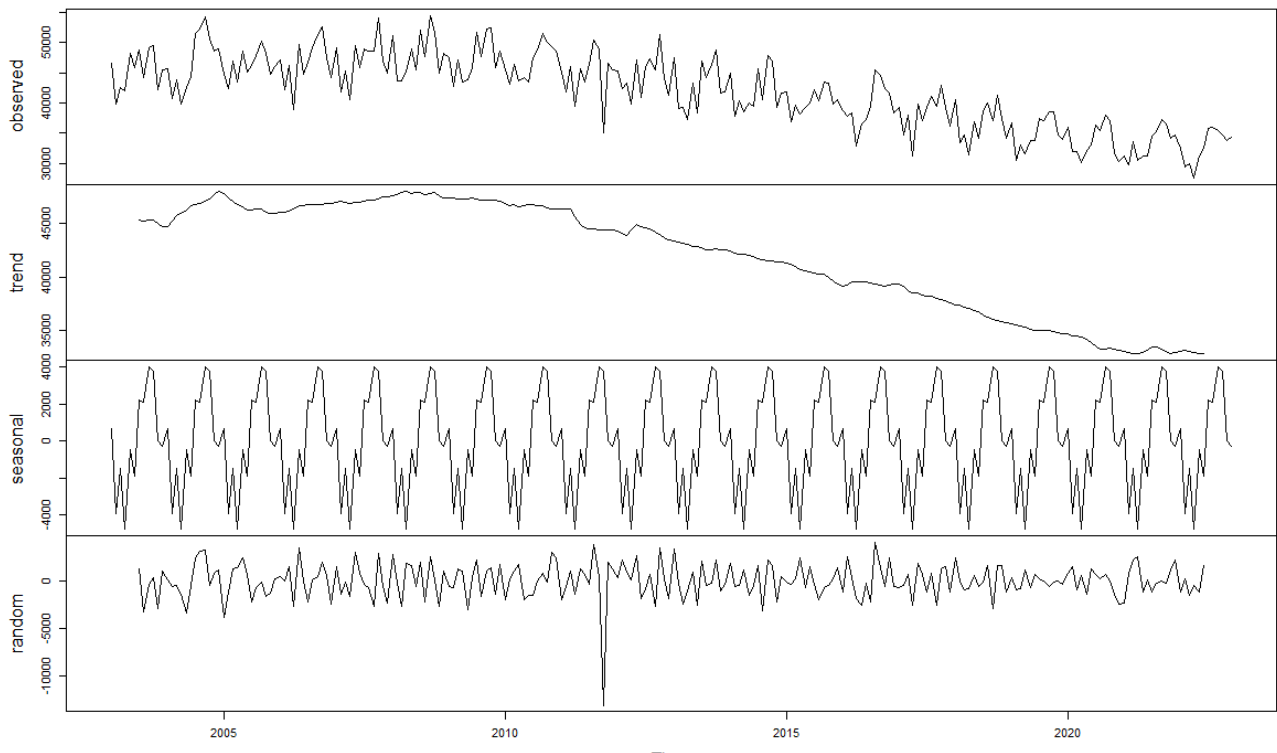
```
data: nasciteMensili
X-squared = 3975.8, df = 60, p-value < 2.2e-16
```

L'esecuzione del test di Ljung-Box restituisce un p-value notevolmente inferiore al livello di significatività di 0.05, rafforzando ulteriormente l'osservazione precedente.

Utilizzando la funzione `decompose()` in R andiamo a decomprimere la serie temporale delle nascite mensili e disegnarla insieme ai suoi componenti di tendenza, stagionalità e residui.

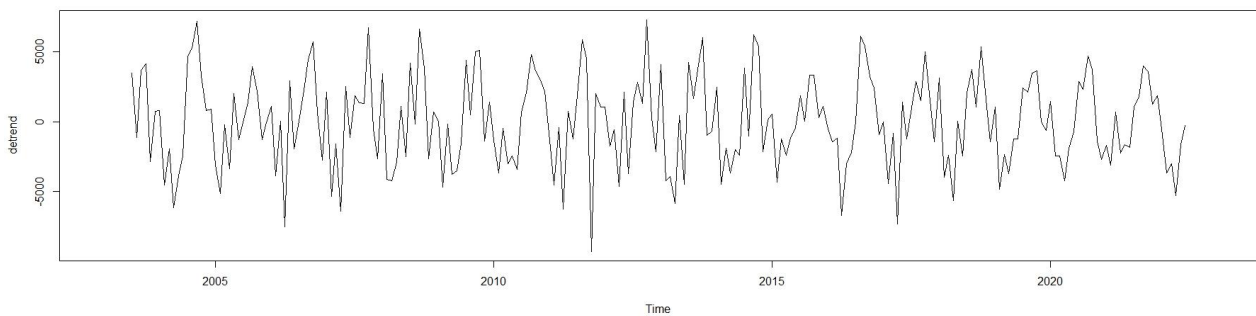
```
>> natMens<-decompose(nasciteMensili, type="additive")
>> plot(natMens)
```

Decomposition of additive time series

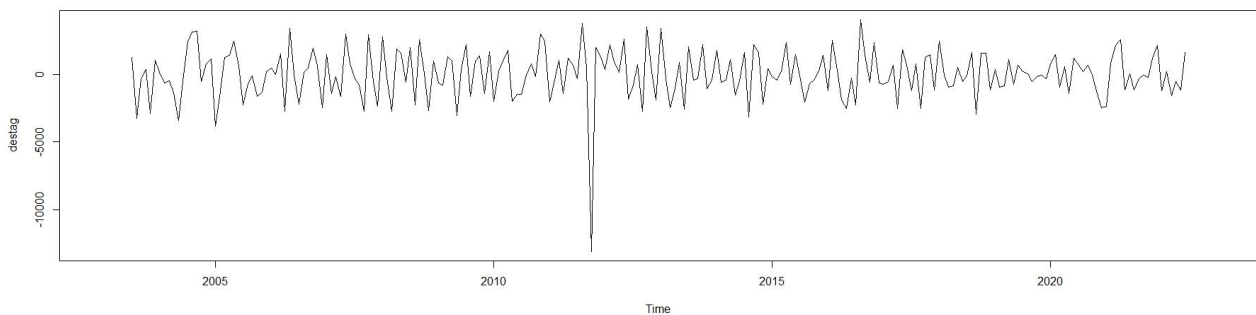


Andiamo a estrarre dalla serie le componenti di tendenza e di stagionalità.

```
>> detrend<-nasciteMensili - natMens$trend
>> plot(detrend)
```

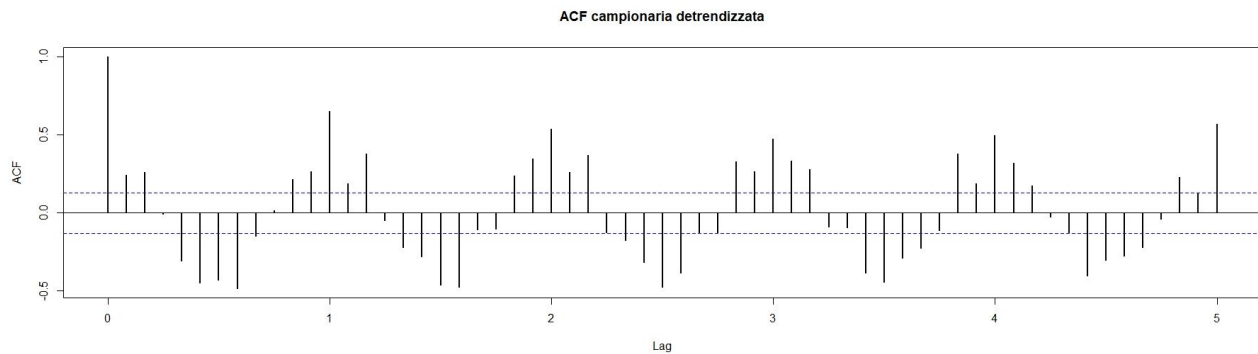


```
>> destag<-nasciteMensili - natMens$trend - natMens$seasonal
>> plot(destag)
```

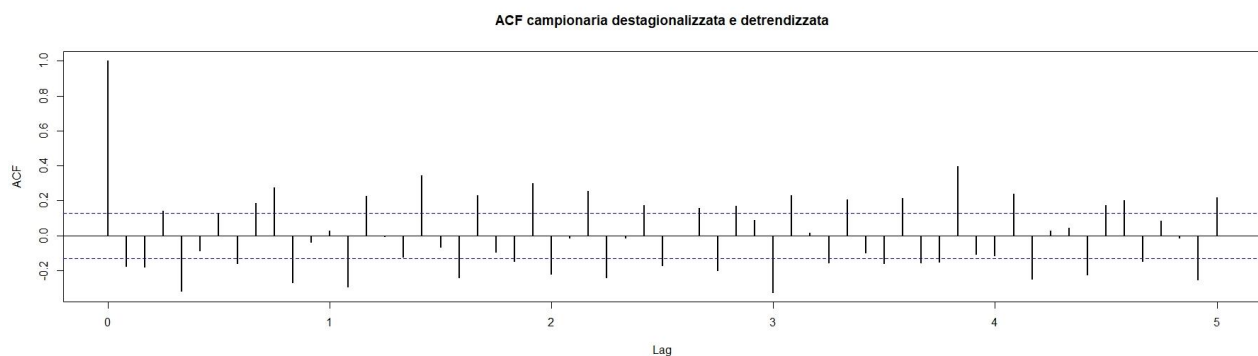


Esaminiamo ora le funzioni di autocorrelazione dopo aver applicato le due operazioni.

```
>> acf(detrend, lag.max = 60, main = "ACF campionaria detrendizzata",lwd=2)
```



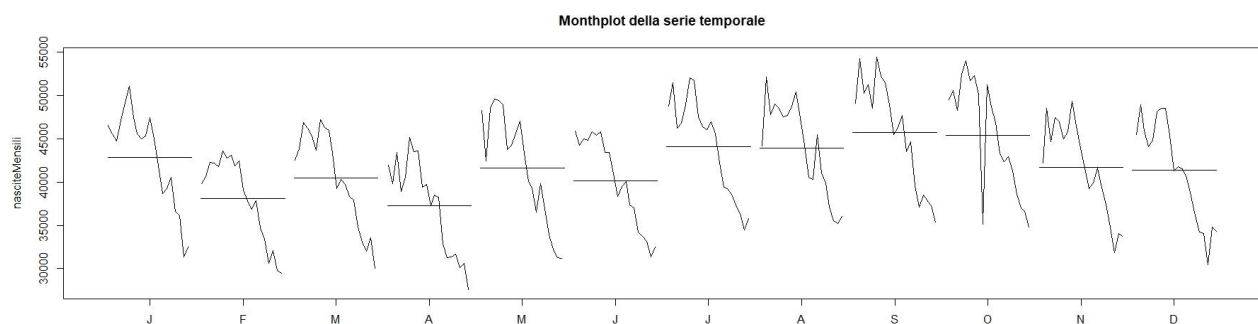
```
>> acf(destag, lag.max = 60, main = "ACF campionaria destagionalizzata e detrendizzata", lwd=2)
```



Dall'esame visivo di entrambi i grafici, è evidente che la serie, dopo aver subito il processo di destagionalizzazione e detrendizzazione, non può essere considerata come un white noise.

Possiamo vedere maggiori dettagli per quanto riguarda la stagionalità attraverso l'illustrazione del monthplot.

```
>> monthplot(nasciteMensili, main="Monthplot della serie temporale")
```



Si nota che il mese di settembre è il mese con più nascite, seguito da ottobre. Inoltre, si evince anche un decremento del numero di nascite in ciascuno dei mesi dell'anno.

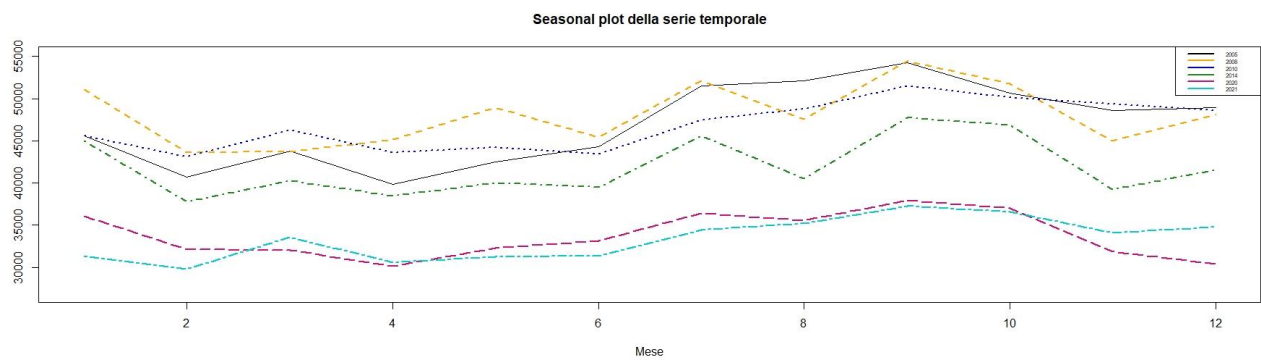
Tale informazione è captata anche dal seasonal plot.

```
>> s1 <- window(nasciteMensili, start=c(2004,1), end=c(2004,12))
>> s2 <- window(nasciteMensili, start=c(2008,1), end=c(2008,12))
>> s3 <- window(nasciteMensili, start=c(2010,1), end=c(2010,12))
>> s4 <- window(nasciteMensili, start=c(2014,1), end=c(2014,12))
>> s5 <- window(nasciteMensili, start=c(2020,1), end=c(2020,12))
>> s6 <- window(nasciteMensili, start=c(2021,1), end=c(2021,12))
>> range(nasciteMensili)
```

```

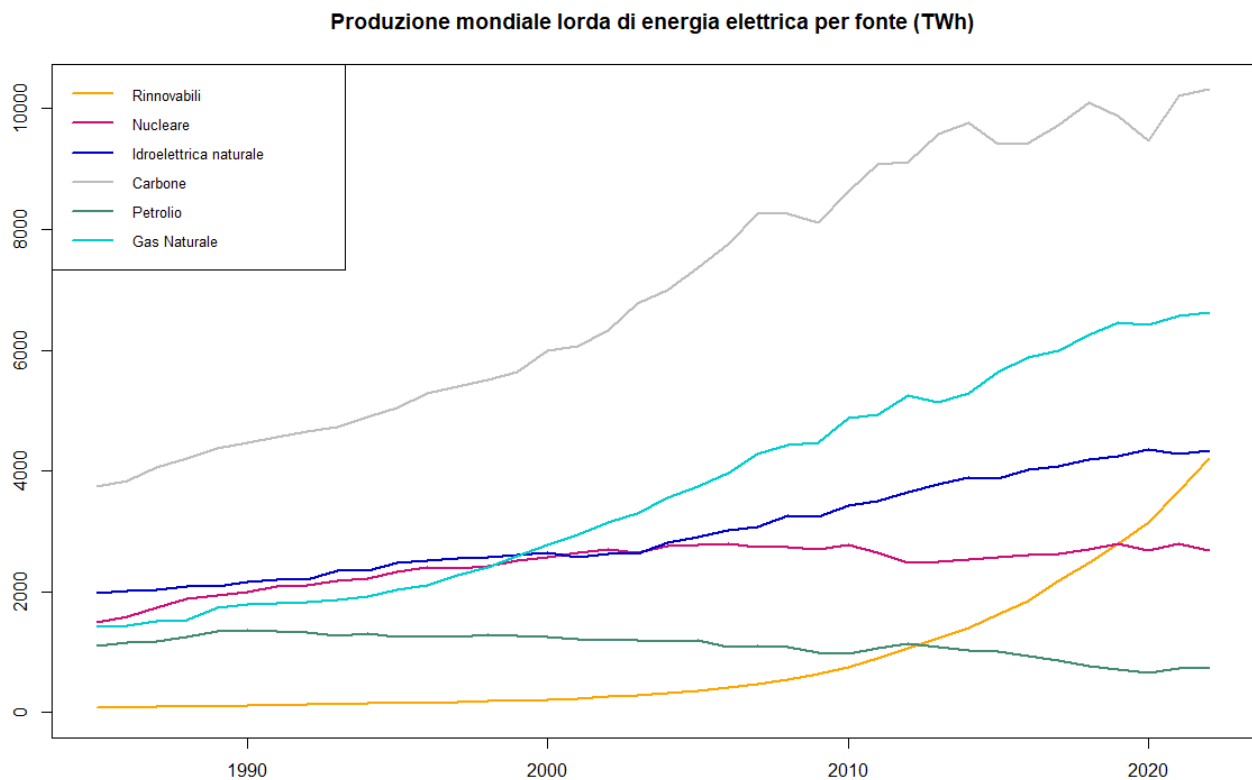
>> x <- seq(1,12,1)
>> matplot(x, s1, type="l", ylim=c(27000,55000), ylab="", xlab="Mese", main="Seasonal plot della serie
temporale")
>> lines(x, s2, type="l", lty=2, col="orange",lwd=2)
>> lines(x, s3, type="l", lty=3, col="blue3", lwd=2)
>> lines(x, s4, type="l", lty=4, col="forestgreen", lwd=2)
>> lines(x, s5, type="l", lty=5, col="deeppink3", lwd=2)
lines(x, s6, type="l", lty=6, col="cyan3", lwd=2)
>> legend("topright", legend = c("2005", "2008","2010","2014","2020","2021"), col = c("black",
"orange","blue3","forestgreen","deeppink3","cyan3"), lwd = 2,cex = 0.5)

```



Produzione mondiale lorda di energia elettrica per fonte(1985-2022)

I dati sono tratti dal sito web dell'[Energy Institute](#) sezione "Exploring Energy-> Statistical Review of World Energy" scaricando il foglio di calcolo "Statistical Review of World Energy Data"



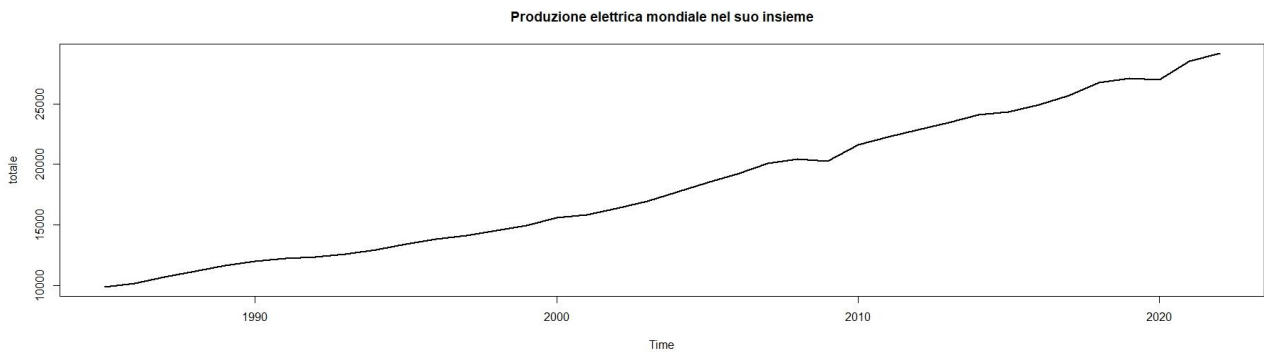
Per un lungo periodo, il carbone è stato la fonte predominante nella produzione globale di energia elettrica, caratterizzandosi come la principale causa di una crescita assoluta significativa nel settore. Tuttavia, a partire dal 1997, il gas naturale ha iniziato a emergere con una crescita più sostenuta, conquistando la posizione di seconda fonte di produzione più importante, subito dopo il carbone.

Nel tentativo di trovare alternative ai combustibili fossili e rispondere alle sfide ambientali, le nuove fonti rinnovabili, quali l'energia eolica, solare, e le biomasse, insieme ad altre meno diffuse, hanno registrato una rapida crescita. Questo fenomeno è caratterizzato da un andamento in continua ascesa, con una tendenza che appare lievemente esponenziale nel tempo.

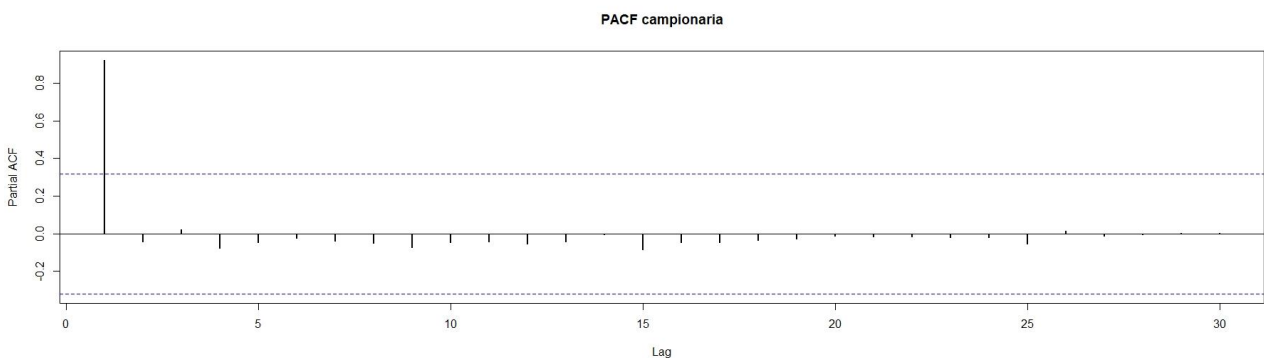
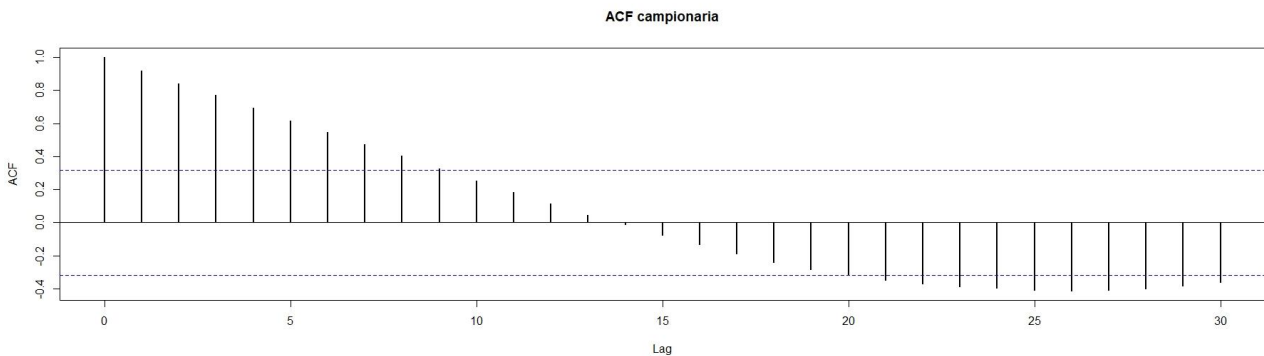
Se analizziamo le fonti energetiche raggruppate in macro categorie, emerge chiaramente che il settore continua a essere dominato dalle fonti termoelettriche basate su combustibili fossili.

Nonostante il crescente interesse verso le fonti rinnovabili, la transizione verso una produzione energetica più sostenibile e verde è ancora in corso, con le fonti fossili che mantengono un ruolo predominante nel panorama energetico mondiale.

Esaminiamo ora l'andamento complessivo della produzione elettrica mondiale attraverso un'analisi visiva del grafico.



Si rileva un chiaro trend in crescita. Tale osservazione trova ulteriore sostegno nel grafico della funzione di autocorrelazione, il quale evidenzia in modo inequivocabile la non stazionarietà della serie temporale.



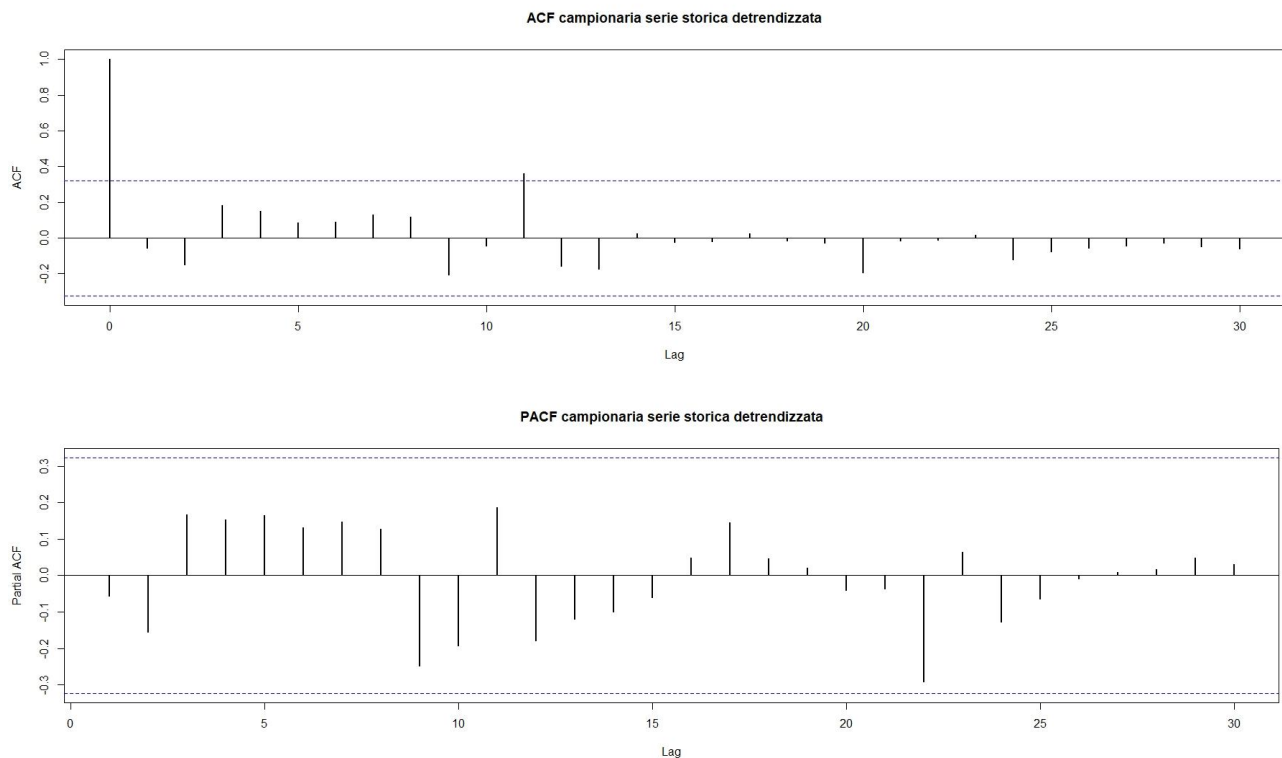
La presenza di autocorrelazioni significative è confermata anche dal test di Ljung-Box, il quale indica che la serie non si comporta come un white-noise.

Box-Ljung test

```
data: totale
x-squared = 392.53, df = 30, p-value < 2.2e-16
```

Procediamo con la detrendizzazione della serie temporale mediante l'approccio della differenza (diff() in R)

```
>> dxt<-diff(totale, lag=1, differences=1)
>> acf(dxt, lag.max = 30, main="ACF campionaria serie storica detrendizzata", lwd=2)
>> Box.test(dxt, lag=30, type="Ljung-Box")
```



Il grafico della funzione di autocorrelazione della serie detrendizzata suggerisce che la distribuzione delle autocorrelazioni assomigli a quella di un white noise.

Tuttavia, per confermare in modo statistico l'assenza di autocorrelazioni significative, eseguiamo il test di Ljung-Box.

Box-Ljung test

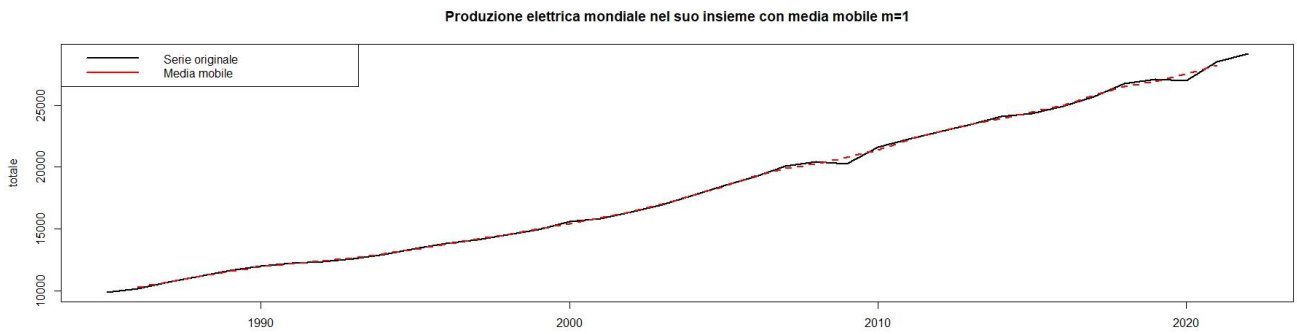
```
data: dxt
X-squared = 26.273, df = 30, p-value = 0.6611
```

I risultati del test indicano un p-value di 0.6611, che supera la soglia di significatività comune di 0.05. Di conseguenza, la serie è coerente con l'ipotesi di essere un white noise.

Esploriamo un diverso metodo di detrendizzazione utilizzando la media mobile di ordine m per stimare il trend della serie temporale. Il risultato di questo approccio di detrendizzazione verrà confrontato con quello ottenuto precedentemente, al fine di valutare la coerenza e la robustezza dei risultati ottenuti.

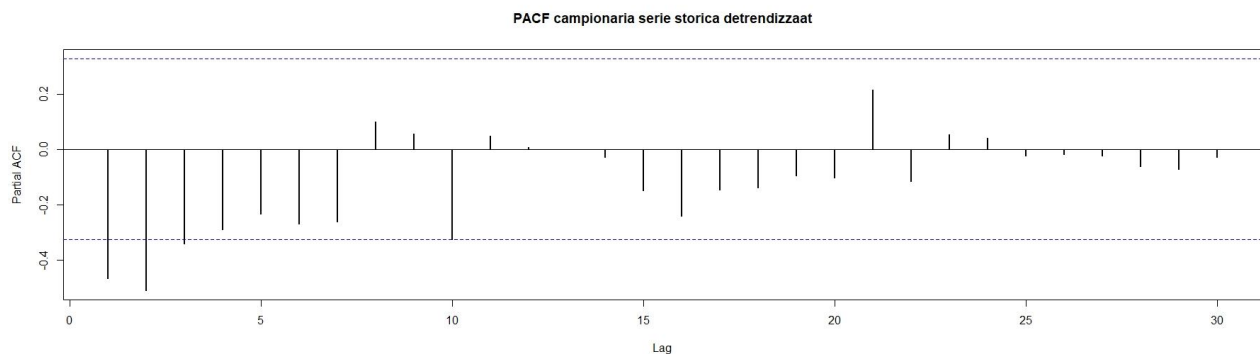
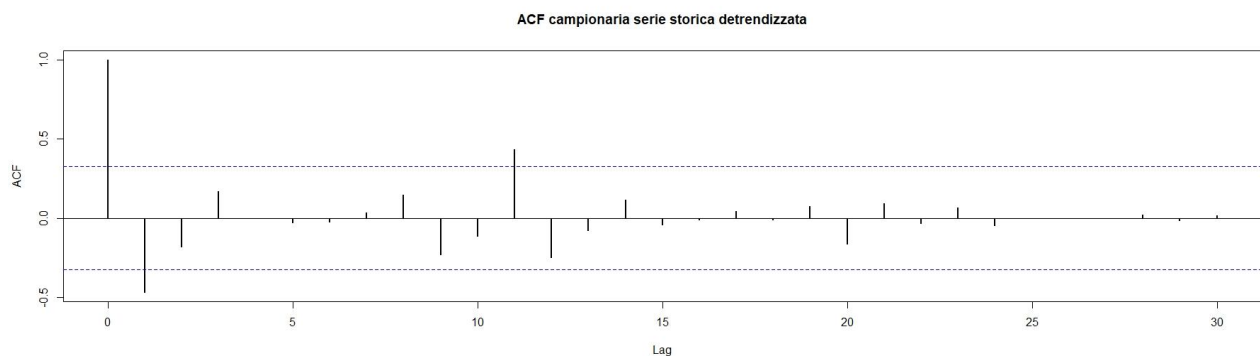
Poiché i dati sono raccolti annualmente, è stata utilizzata una media mobile di ordine $m=1$.

```
>> m=1
>> gamma<-rep(1/((2*m)+1),((2*m)+1))
>> ma<-filter(totale,gamma, method = "convolution",sides=2,circular = FALSE)
```



Successivamente, è stato eseguito il processo di detrendizzazione sottraendo la media mobile precedentemente calcolata dalla serie temporale.

```
>> detrend<-totale - ma
>> acf(detrend,lag.max = 30, main="ACF campionaria serie storica detrendizzata",lwd=2)
>> pacf(detrend,lag.max = 30,main="PACF campionaria serie storica detrendizzata",lwd=2)
>> Box.test(detrend,lag=30,type="Ljung-Box")
```



Box-Ljung test

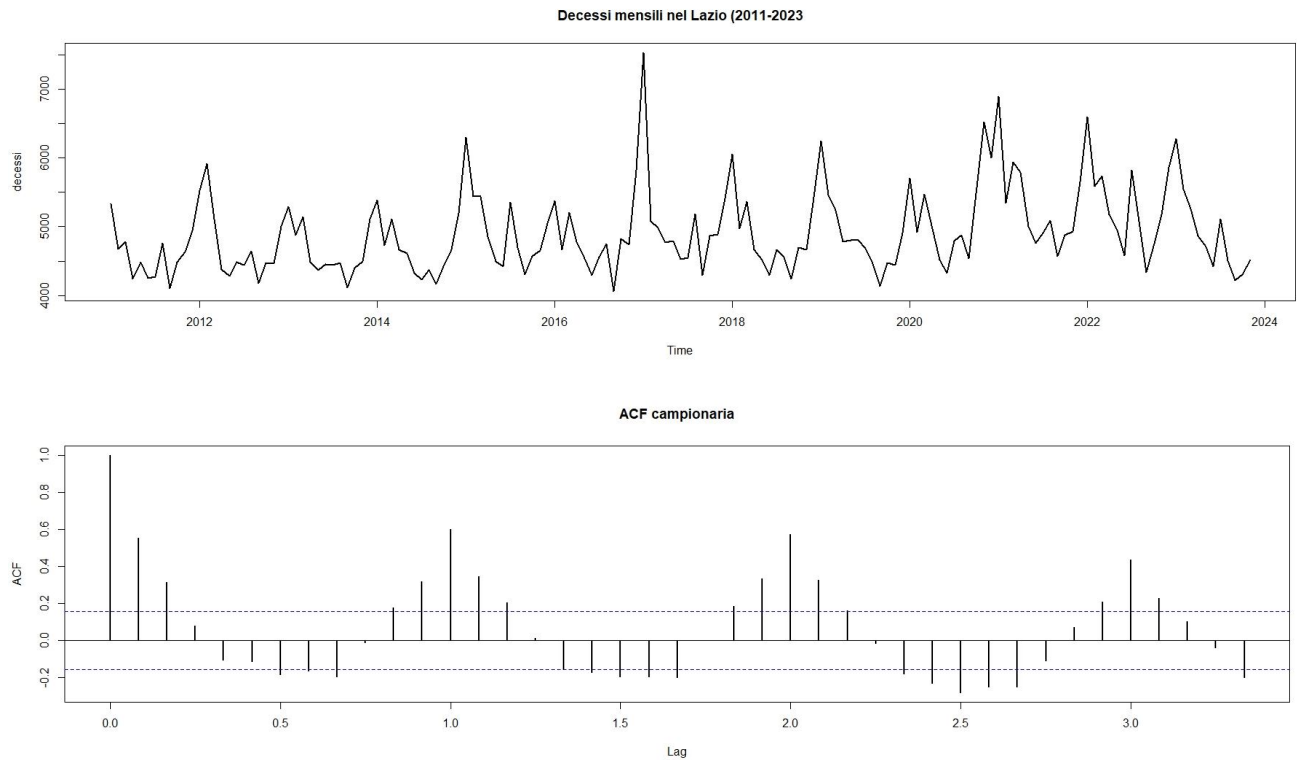
```
data: detrend
X-squared = 35.706, df = 30, p-value = 0.2179
```

Dall'esecuzione del test di Ljung-Box otteniamo un valore di p-value pari a 0.2179. Sebbene questo p-value sia inferiore rispetto a quello ottenuto precedentemente utilizzando l'operatore di differenza per la detrendizzazione, è comunque superiore al livello di significatività comune di 0.05. Di conseguenza, giungiamo alla conclusione che anche in questo caso la serie detrendizzata può essere considerata come un white-noise.

Decessi mensili nel Lazio (01/2011-11/2023)

I dati sono tratti dal sito [Dati ISTAT](#)

```
>> dataset<- read.csv(".\\decessiMensiliLazio.csv", sep = ",")
>> decessi <- ts(dataset$MORTI,start=c(2011,1), freq= 12)
>> plot(decessi,lwd=2,main="Decessi mensili nel Lazio (2011-2023)")
>> acf(decessi,lag.max = 40, main="ACF campionaria")
```



La presenza di chiari picchi nell'ACF indica che la serie temporale non si comporta in modo casuale, come nel caso di un processo di white-noise.

Al contrario, suggerisce la presenza di correlazioni strutturate tra le osservazioni a varie distanze temporali.

Per confermare ulteriormente la presenza di correlazioni significative nella serie temporale, possiamo eseguire il test di Ljung-Box.

```
>> Box.test(decessi,lag=40,type = "Ljung-Box")
```

Box-Ljung test

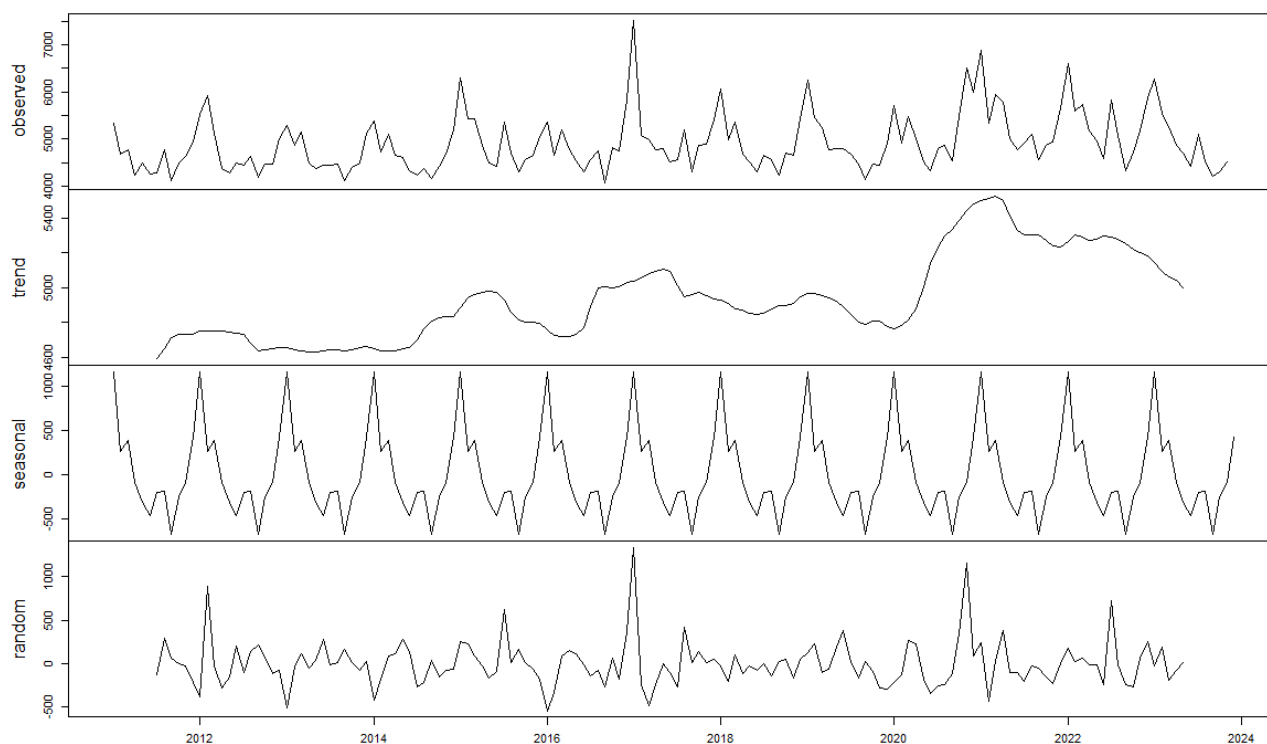
```
data: decessi
X-squared = 470.14, df = 40, p-value < 2.2e-16
```

Come si può notare, il test restituisce un p-value molto piccolo che conferma l'idea che la serie temporale non segue un processo di white-noise.

Per comprendere le dinamiche sottostanti alla serie temporale dei decessi nel Lazio, andiamo a effettuare la decomposizione additiva della serie temporale.

```
>> decessiMensili <-decompose(decessi, type = "additive")
```

Decomposition of additive time series



La presenza di trend e stagionalità contribuisce a rendere non stazionaria la serie temporale osservata ma prima di andare a depurare la serie di tali componenti vediamo maggiori dettagli sulla stagionalità.

Monthplot serie temporale dei decessi mensili nel Lazio

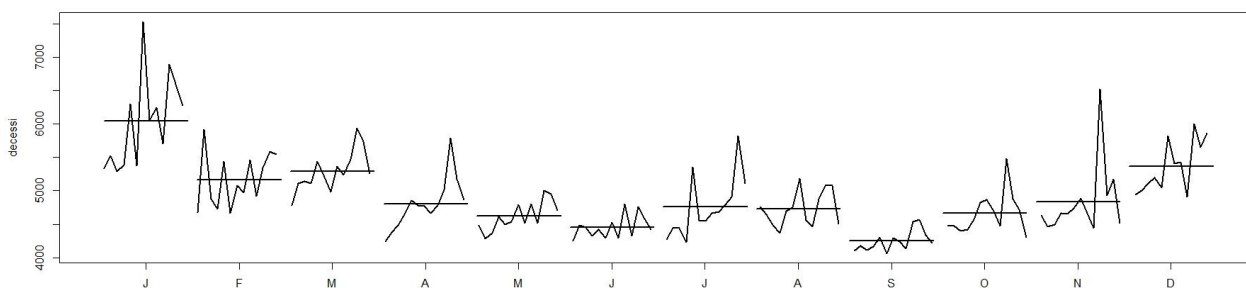
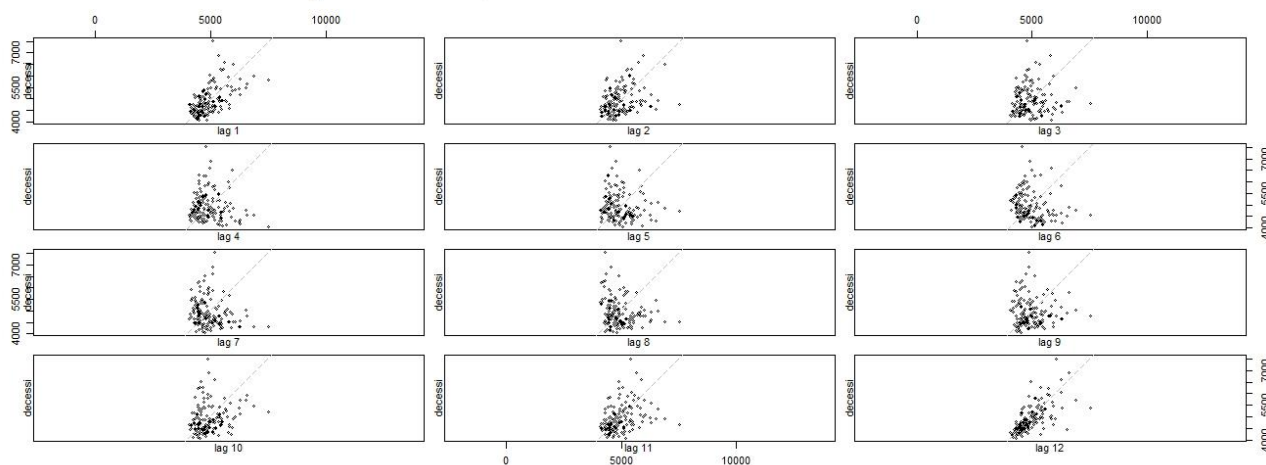
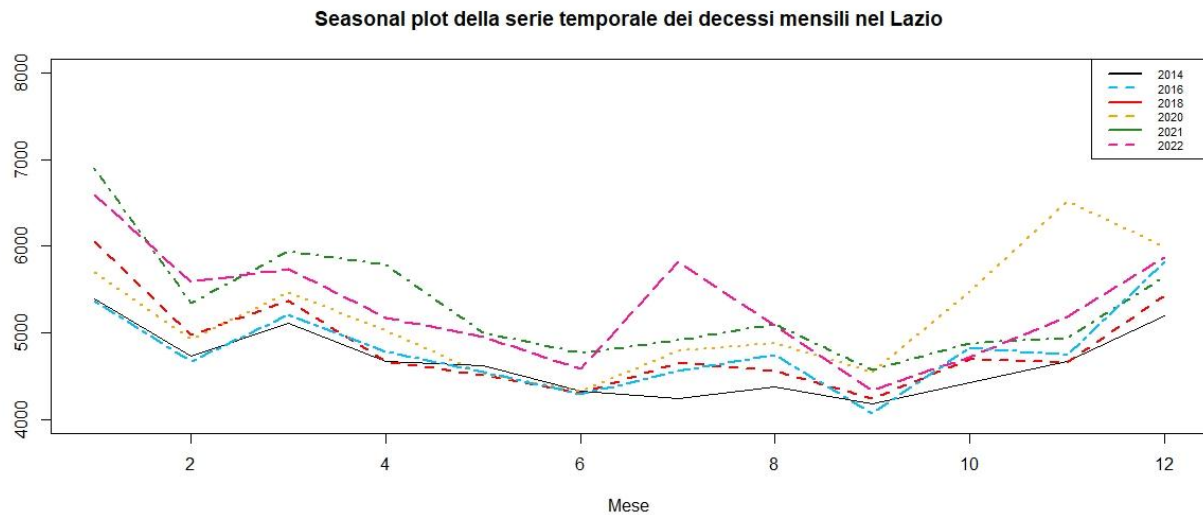


Diagramma di autodispersione della serie relativa ai decessi mensili nel Lazio





Dai grafici emerge un interessante andamento in cui il mese di gennaio presenta un valore più elevato rispetto agli altri mesi.

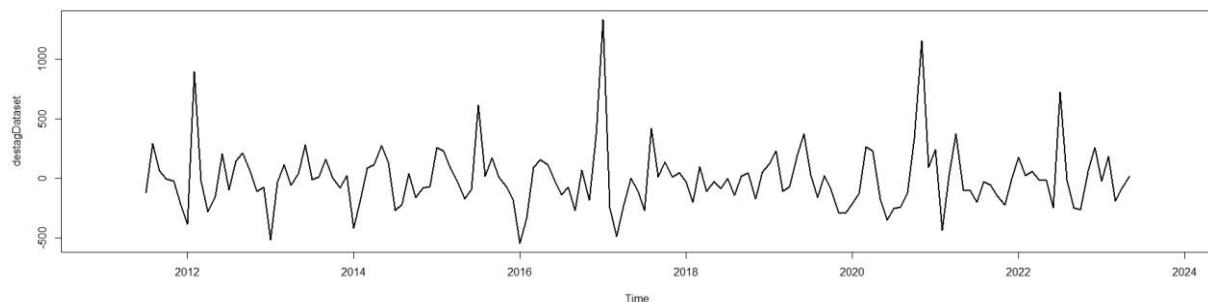
Si può notare come la mortalità sia maggiore durante i mesi più freddi, presumibilmente a causa della maggiore incisività delle tipiche malattie stagionali.

In aggiunta, notiamo un aumento anomalo del numero di decessi rispetto alla media nei mesi di aprile 2021, novembre 2020 e luglio 2022, chiaramente attribuibili agli impatti delle ondate di COVID.

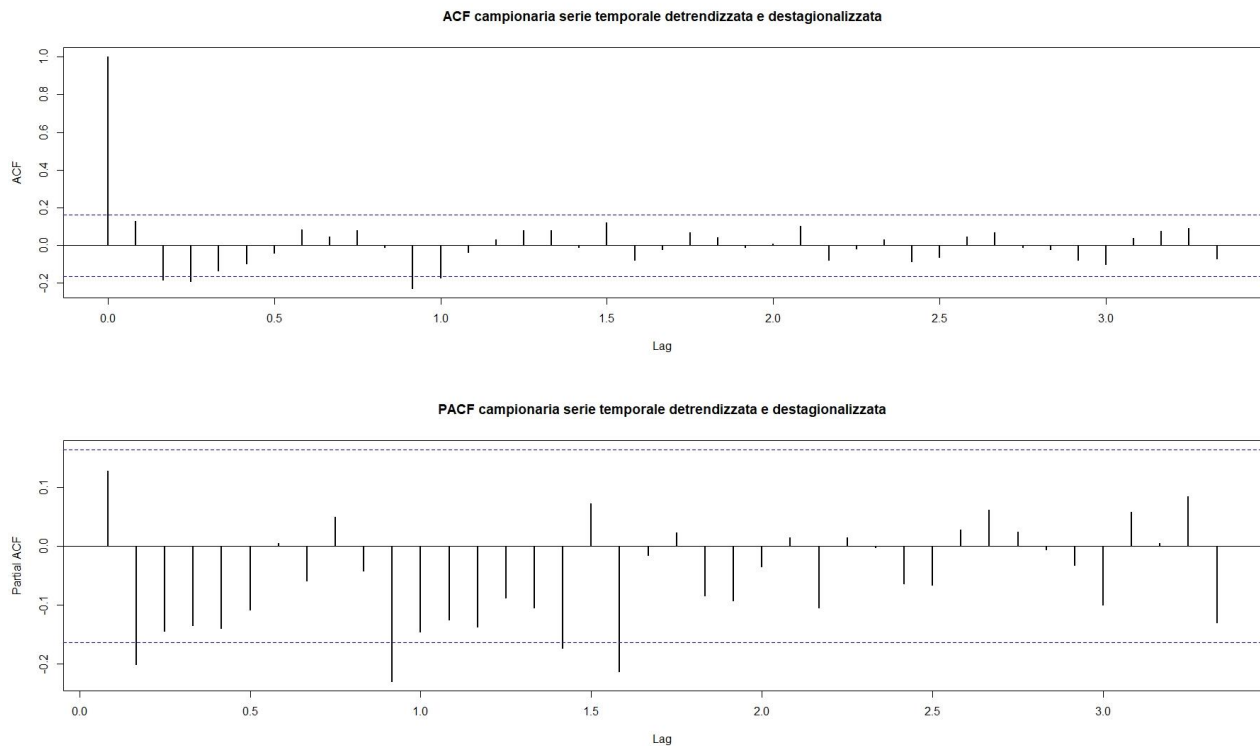
Passiamo a depurare la serie temporale dalla componente di trend e stagionalità.

```
>> trend_comp<-decessiMensili$trend
>> seasonal_comp<-decessiMensili$seasonal
>> detrendDataset<-decessi-trend_comp
>> destagDataset<-decessi - trend_comp - seasonal_comp
```

Il nuovo grafico osservato è il seguente:



e le funzioni di autocorrelazione e autocorrelazione parziale



Osservando il grafico della funzione di autocorrelazione, notiamo un comportamento simile a quello dei processi white-noise e ci poniamo la domanda se sia possibile associare la serie a un white noise o meno. Per rispondere a questa interrogativo, ci avvaliamo come al solito del test di Ljung-Box.

```
>> Box.test(ddxt,lag=40,type = "Ljung-Box")
```

Box-Ljung test

```
data: destagDataset
X-squared = 52.869, df = 40, p-value = 0.08364
```

Il p-value restituito è appena superiore a 0.05, ma non possiamo trarre una conclusione definitiva che la serie sia effettivamente un white-noise. L'unica affermazione che possiamo fare con certezza è che non disponiamo di sufficiente evidenza statistica per respingere l'ipotesi che la serie sia un white-noise.