

PW Deep Learning on Temporal Data

Luca Ruocco & Roberta Luocco

2025-06-05

```
library(knitr)
library(ggplot2)
library(dplyr)
library(tidyr)
library(fpp2)
library(forecast)
library(seastests)
library(tseries)
library(fBasics)
library(ggplot2)
library(NTS)
```

1 Introduzione

Il presente lavoro si concentra sull'analisi di una serie storica generata artificialmente a scopo didattico. Sebbene non sia nota la natura reale del fenomeno rappresentato, l'obiettivo dell'elaborato è quello di applicare le principali tecniche di analisi delle serie temporali, al fine di comprenderne la struttura interna, verificarne la stazionarietà, individuare eventuali componenti deterministiche (trend, stagionalità) e stimare modelli in grado di descriverla e prevederne l'evoluzione futura.

L'approccio metodologico seguito integra più prospettive complementari, rifacendosi ai principali paradigmi dell'analisi delle serie storiche:

- Approccio classico (descrittivo): basato sulla scomposizione della serie nelle sue componenti fondamentali (trend, ciclo, stagionalità e residuo), analizzate sia graficamente che attraverso scomposizioni additiva e moltiplicativa. Tale analisi consente una prima lettura qualitativa del comportamento della serie e delle sue regolarità interne.
- Approccio Box-Jenkins (modelli ARIMA/SARIMA): fondato su una visione stocastica del processo generativo, attraverso la stima di modelli autoregressivi e a media mobile su serie rese stazionarie. In questo contesto si fa ricorso a strumenti diagnostici rigorosi, come ACF, PACF, test di stazionarietà e analisi dei residui, per identificare e validare i modelli.
- Modelli predittivi neurali (NNAR): come estensione moderna ai modelli lineari, si considerano reti neurali autoregressive non lineari (Neural Network AutoRegressive), che permettono di cogliere pattern complessi e relazioni non lineari all'interno delle serie. Sebbene non siano approfonditi in questa fase, tali modelli rappresentano una prospettiva interessante nell'analisi predittiva avanzata.

2 Analisi della serie temporale

2.1 Analisi visiva e test statistici della serie temporale

```
serie_raw <- load("Serie1_sim.csv")
```

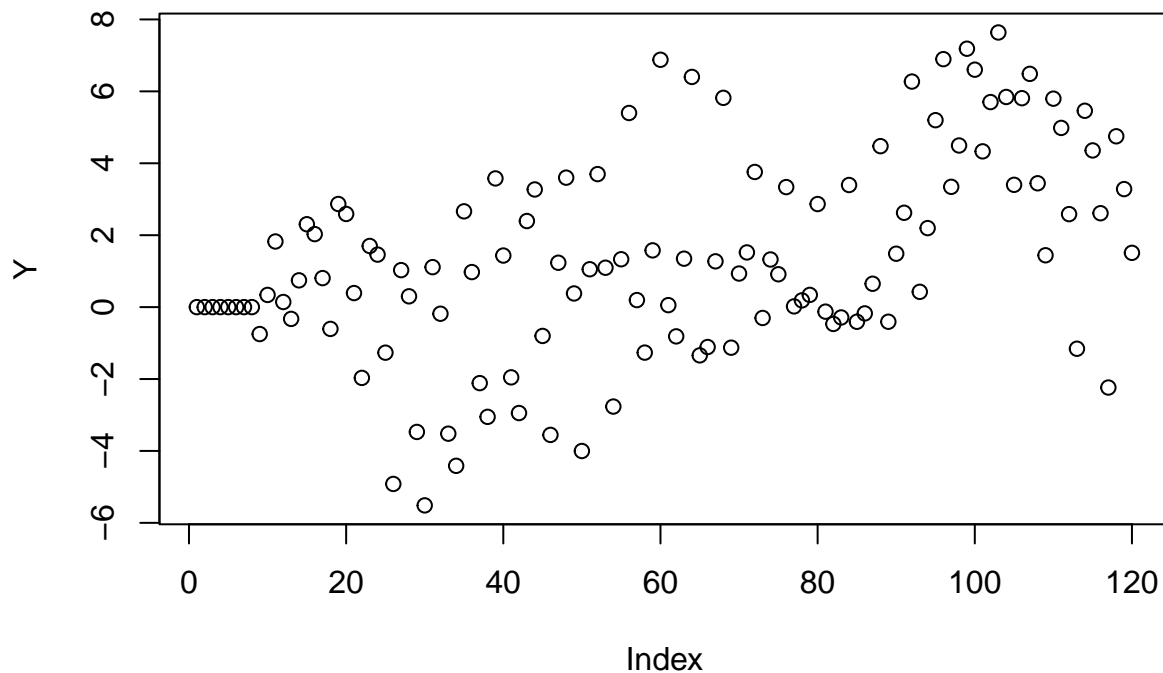
```
str(Y)
```

```
## num [1:120] 0 0 0 0 0 ...
```

```
summary(Y)
```

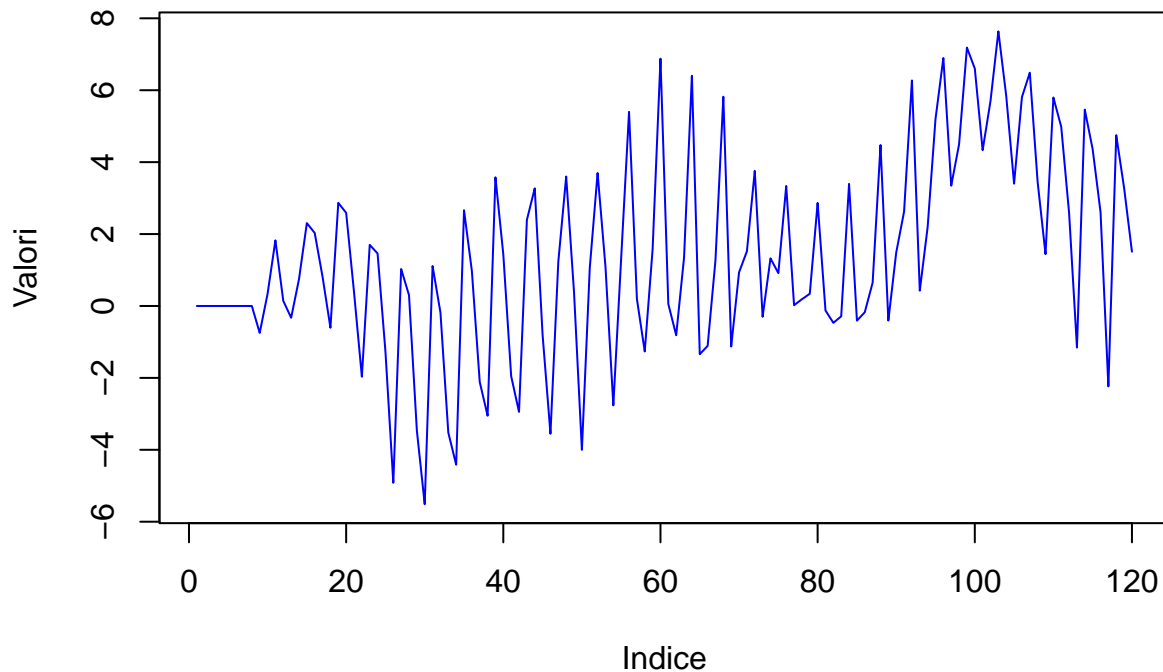
```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -5.5138 -0.2099   1.0739   1.3818   3.3396   7.6371
```

```
plot(Y)
```



```
plot(Y, type = "l", col = "blue", main = "Serie temporale",  
ylab = "Valori", xlab = "Indice")
```

Serie temporale



Analisi visiva preliminare della serie temporale: Nel contesto dell'approccio classico all'analisi delle serie storiche, si è inizialmente proceduto con la rappresentazione grafica della serie al fine di esaminare visivamente la presenza di componenti fondamentali quali trend, stagionalità, cicli e componente irregolare (o rumore).

Il grafico ottenuto evidenzia chiaramente un andamento crescente del livello medio della serie per un ampio intervallo temporale (indicativamente tra le osservazioni 30 e 100), suggerendo la presenza di una componente di trend strutturata.

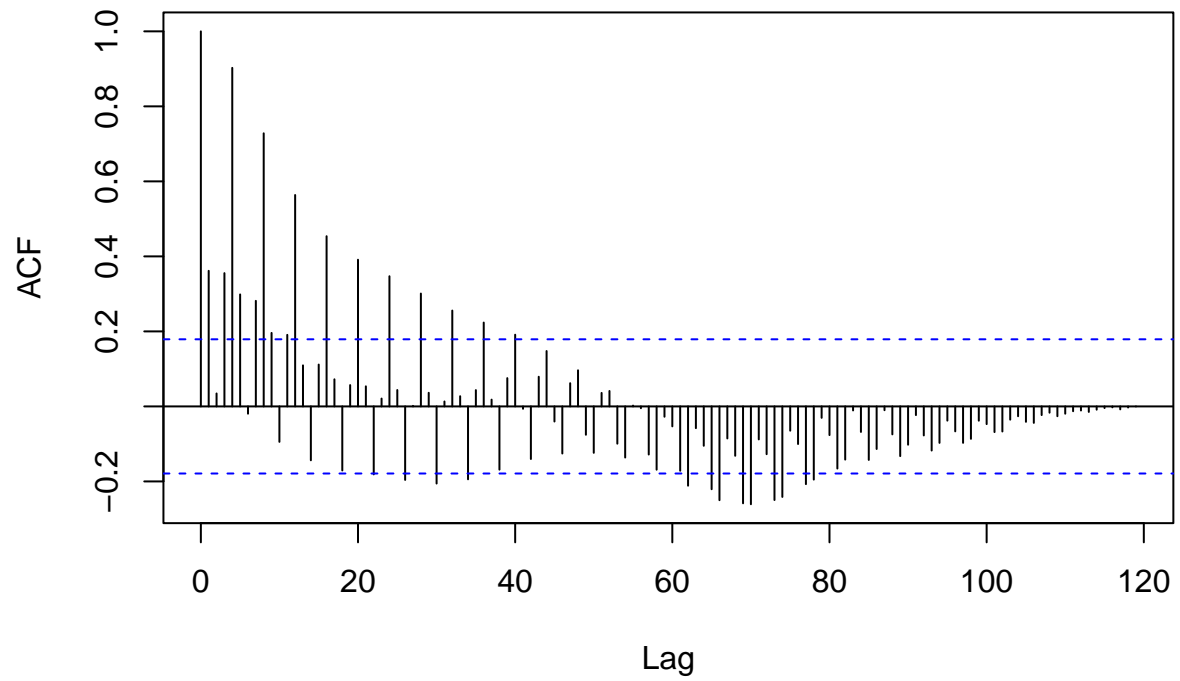
Si osservano inoltre oscillazioni regolari e periodiche, distribuite con una certa stabilità lungo l'asse temporale, la cui frequenza apparente è compatibile con un intervallo stagionale. Questo elemento indica la probabile presenza di una componente stagionale. Tali oscillazioni risultano inoltre non costanti in ampiezza, ma piuttosto crescenti con l'incremento del livello della serie, circostanza che suggerisce l'opportunità di adottare una scomposizione di tipo moltiplicativo.

Accanto al trend e alla stagionalità, si rileva anche una componente ciclica, caratterizzata da movimenti ampi e irregolari del livello medio, e una componente aleatoria residua, rappresentata da fluttuazioni imprevedibili e non sistematiche attorno al valore atteso.

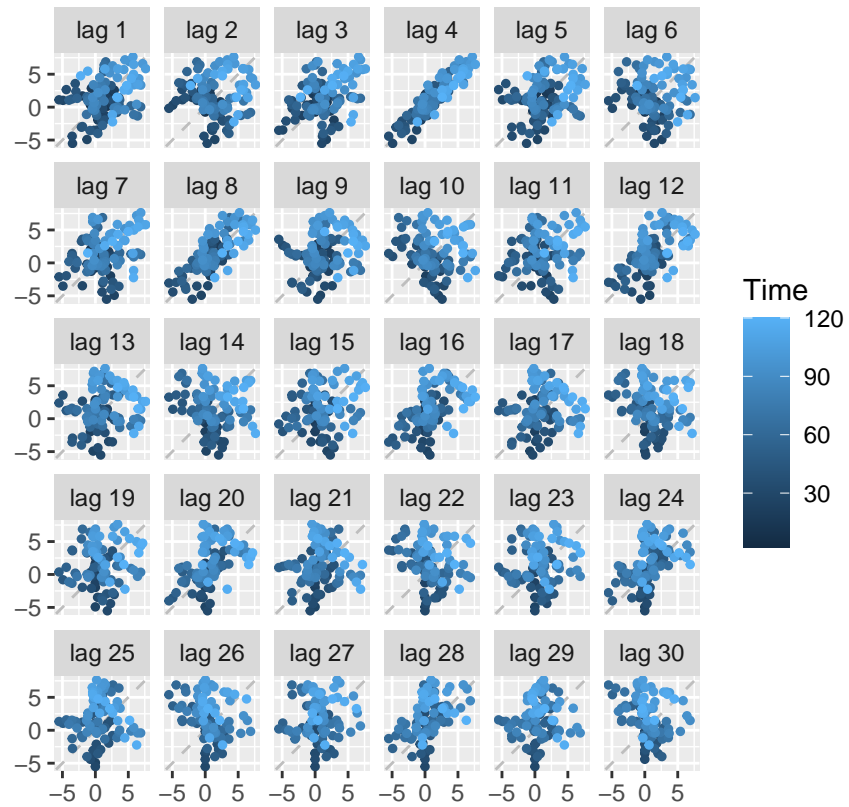
Pertanto, sulla base dell'analisi visiva, la serie in esame appare costituita da una combinazione additiva/moltiplicativa delle componenti principali e risulta idonea a una successiva scomposizione classica moltiplicativa per l'identificazione formale delle strutture interne.

```
acf(Y, lag.max = 120, main = "ACF - Ricerca stagionalità")
```

ACF – Ricerca stagionalità

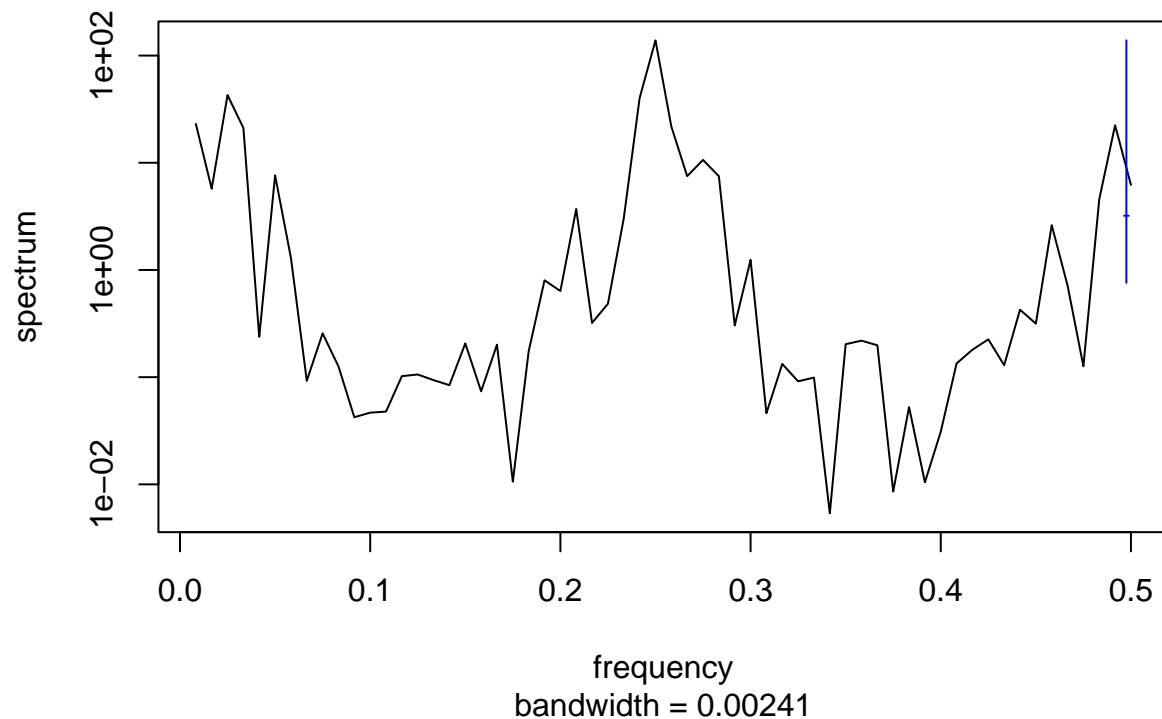


```
gglagplot(Y, lags = 30, do.lines = FALSE)
```



```
spectrum(Y, main = "Spettro di potenza (Fourier)")
```

Spettro di potenza (Fourier)



Per investigare la presenza di componente stagionale nella serie, sono stati impiegati tre strumenti grafici complementari: funzione di autocorrelazione (ACF), scatterplot dei ritardi (gglagplot) e spettro di potenza.

Funzione ACF Il grafico dell'autocorrelazione, calcolato fino a lag 120, mostra una chiara struttura regolare, con picchi decrescenti visibili a intervalli costanti. In particolare, si osservano valori significativamente positivi a lag 4, 8, 12, ..., che suggeriscono una stagionalità con periodicità pari a 4. La lenta decadenza dell'autocorrelazione conferma inoltre la non stazionarietà della serie.

Scatterplot dei ritardi – gglagplot() La rappresentazione della correlazione tra i valori osservati e i loro ritardi (fino a lag 30) conferma visivamente il comportamento ciclico. Si notano pattern ricorrenti e strutture lineari marcate a lag multipli di 4 (es. lag 4, lag 8, lag 12), dove i punti tendono a disporsi lungo una linea diagonale, indicando forte dipendenza lineare stagionale tra le osservazioni distanti di 4 unità.

Spettro di potenza (Analisi di Fourier) Il grafico dello spettro di potenza mostra un picco evidente per una frequenza prossima a 0.25, che corrisponde a un periodo di 4. Questo conferma la presenza di una componente ciclica dominante a frequenza 4, coerente con quanto osservato nei grafici precedenti.

```
Y_ts4 <- ts(Y, frequency = 4)

# Test di Weibel-Olleek
wo_test <- isSeasonal(Y_ts4, test = "wo")
print(wo_test)

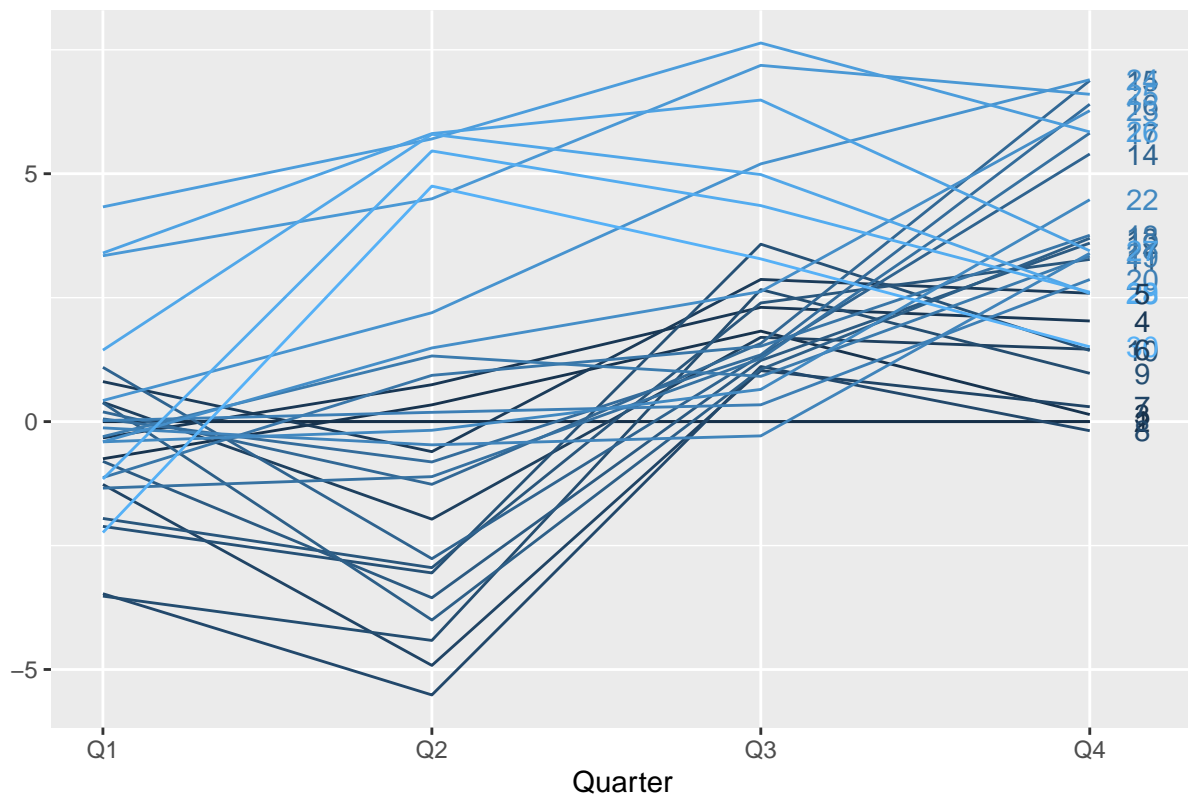
## [1] TRUE

# Test di Kruskal-Wallis
kw_test <- isSeasonal(Y_ts4, test = "kw")
print(kw_test)

## [1] TRUE
```

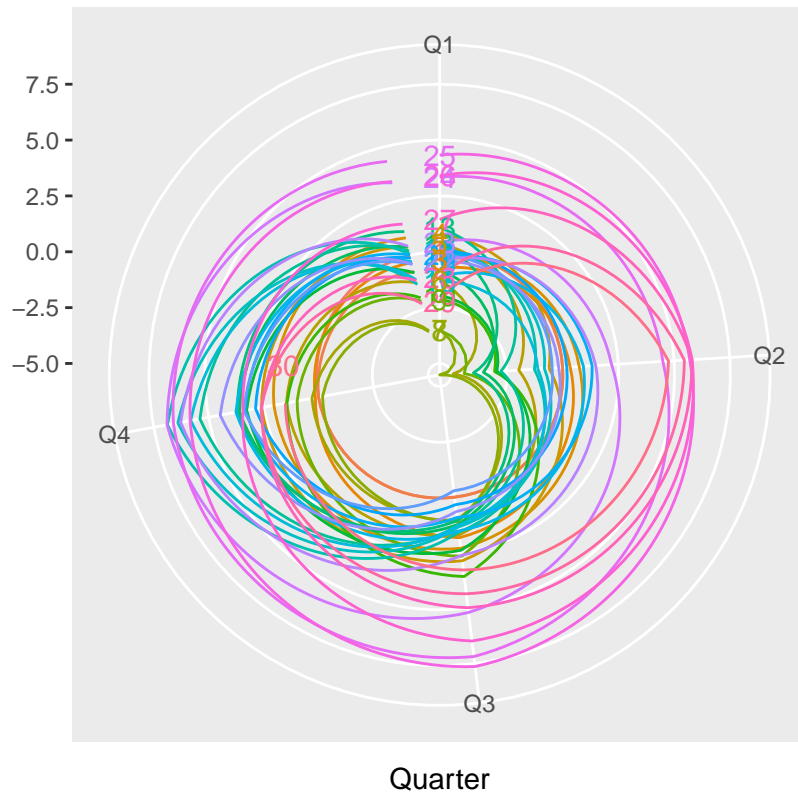
```
ggseasonplot(Y_ts4, year.labels = TRUE, continuous = TRUE)
```

Seasonal plot: Y_ts4



```
# Polar season plot
ggseasonplot(Y_ts4, year.labels = TRUE, polar = TRUE)
```

Seasonal plot: Y_ts4



Dopo l'osservazione esplorativa e la valutazione grafica dell'ACF, la presenza di una componente stagionale con frequenza 4 è stata formalmente confermata attraverso test statistici e rappresentazioni grafiche di tipo stagionale. Sono stati condotti due test statistici per la verifica della stagionalità:

- Il test di Webel-Olleck (WO-test), basato su tecniche di regressione stagionale, ha restituito valore TRUE.
- Il test non parametrico di Kruskal-Wallis (KW-test), che valuta differenze significative tra sottogruppi stagionali, ha anch'esso restituito TRUE.

Entrambi i risultati confermano con significatività statistica la presenza di stagionalità regolare con lag=4.

Grafici stagionali: - Rappresentazione lineare Il grafico mostra l'andamento medio della serie ripartito in gruppi di 4 periodi consecutivi per ciascun "anno" simulato. Le traiettorie dei diversi cicli si dispongono con andamenti simili, evidenziando la presenza di una dinamica stagionale regolare e ripetuta ogni 4 osservazioni.

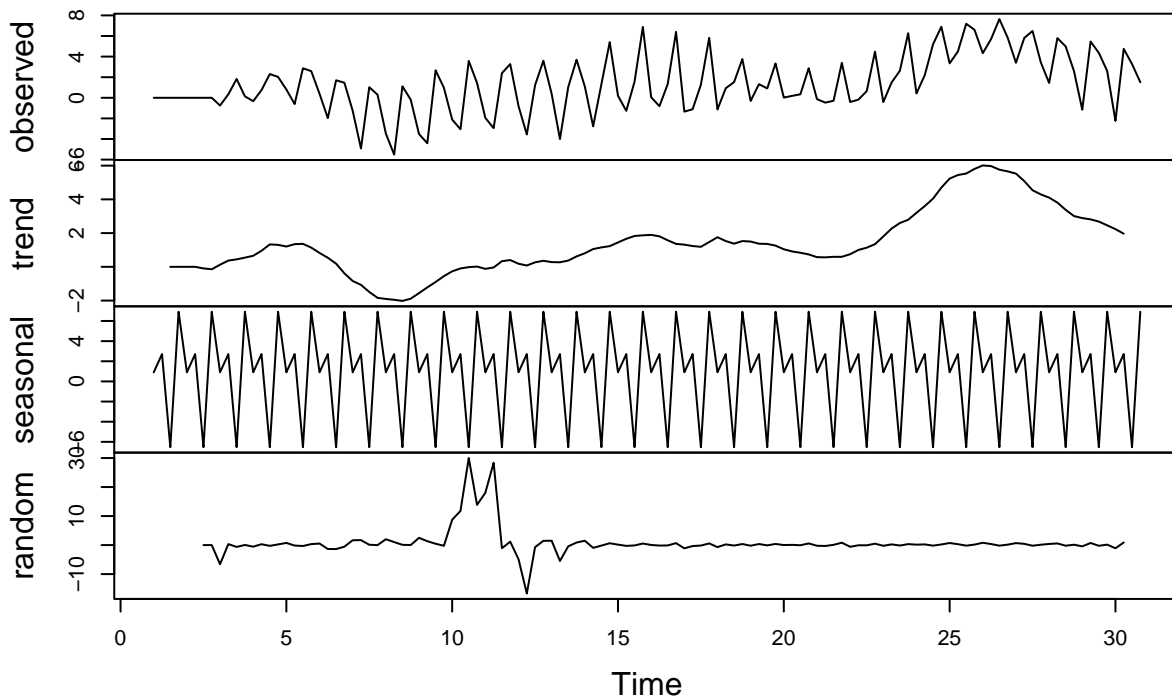
- Rappresentazione polare Il grafico in coordinate polari rafforza visivamente la ciclicità, mostrando curve chiuse e concentriche che seguono lo stesso schema per ogni ciclo. La regolarità delle traiettorie nei 4 settori conferma la presenza di una componente stagionale ben definita con frequenza pari a 4.

Tutti gli strumenti utilizzati – test statistici e rappresentazioni grafiche – convergono nell'evidenziare una stagionalità ben strutturata e ripetitiva con periodicità di 4 periodi, che sarà modellata nelle fasi successive tramite specifici modelli.

2.2 Approccio classico

```
decomp_m <- decompose(Y_ts4, type = "multiplicative")
plot(decomp_m)
```


Decomposition of multiplicative time series



In seguito all'evidenza grafica e statistica di una chiara stagionalità con periodicità pari a 4 osservazioni, si è proceduto con l'applicazione della scomposizione classica di tipo moltiplicativo, secondo il modello $y_t = \text{trend} \times \text{seasonal} \times \text{random}$. Tale approccio consente di isolare e analizzare separatamente le tre componenti fondamentali della serie: il trend, la stagionalità e la componente irregolare (residuale).

L'analisi del grafico risultante mostra che la serie osservata è caratterizzata da fluttuazioni periodiche regolari, sovrapposte a un'evoluzione strutturale del livello medio. La componente di trend, estratta tramite medie mobili, evidenzia un andamento non lineare: si osserva una fase iniziale relativamente stabile, seguita da un periodo di crescita progressiva fino a circa il venticinquesimo ciclo, per poi decrescere leggermente nella parte finale della serie. Questo comportamento suggerisce la presenza di dinamiche a lungo termine, potenzialmente riconducibili a effetti strutturali del fenomeno generatore della serie.

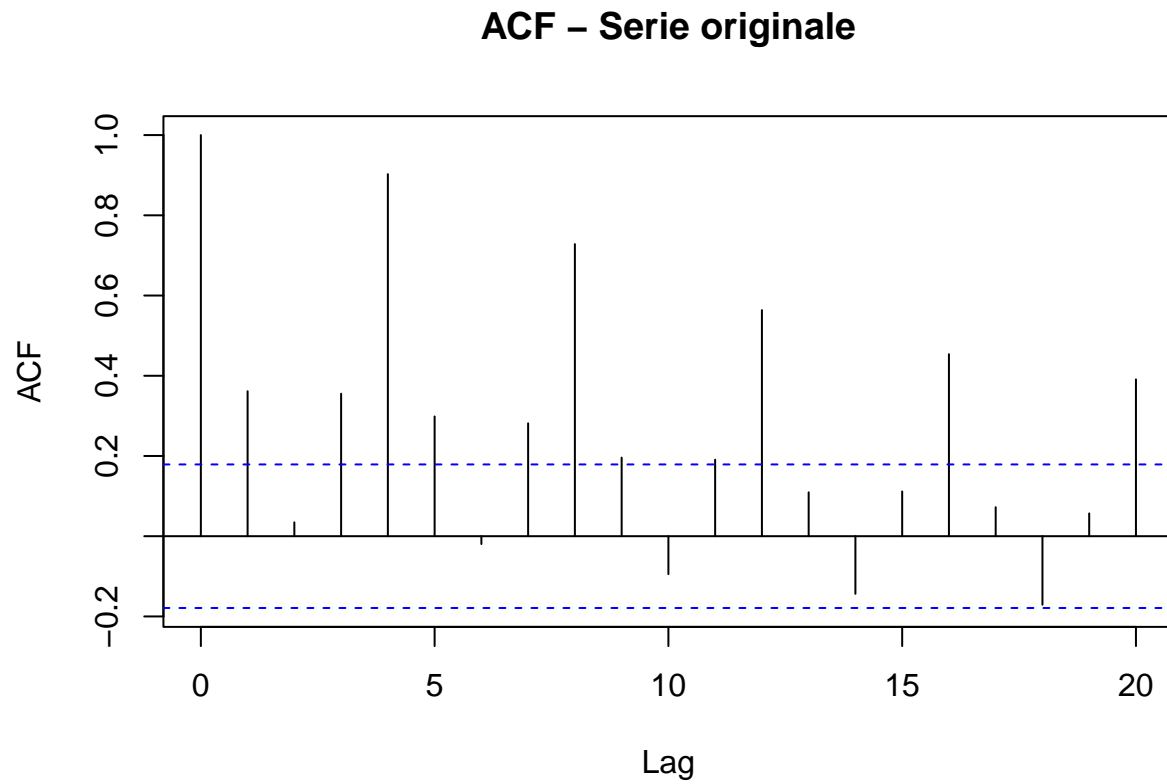
La stagionalità stimata si presenta marcatamente regolare e stabile: i cicli si ripetono con precisione ogni quattro osservazioni, confermando quanto già rilevato in fase esplorativa e nei test formali. L'ampiezza dei picchi stagionali è coerente con la natura moltiplicativa della scomposizione, in quanto risulta proporzionale al livello della serie nei diversi intervalli temporali.

Infine, la componente irregolare risulta in gran parte contenuta e distribuita attorno al valore neutro, con l'eccezione di un'anomalia localizzata intorno alla dodicesima osservazione, che potrebbe indicare la presenza di un outlier o di un evento non ricorrente. A parte questo episodio isolato, i residui appaiono compatibili con una distribuzione casuale, priva di struttura sistematica.

Nel complesso, la scomposizione classica ha permesso di confermare e quantificare la presenza di trend e stagionalità nella serie, offrendo una lettura chiara e separata delle componenti che ne influenzano l'evoluzione nel tempo.

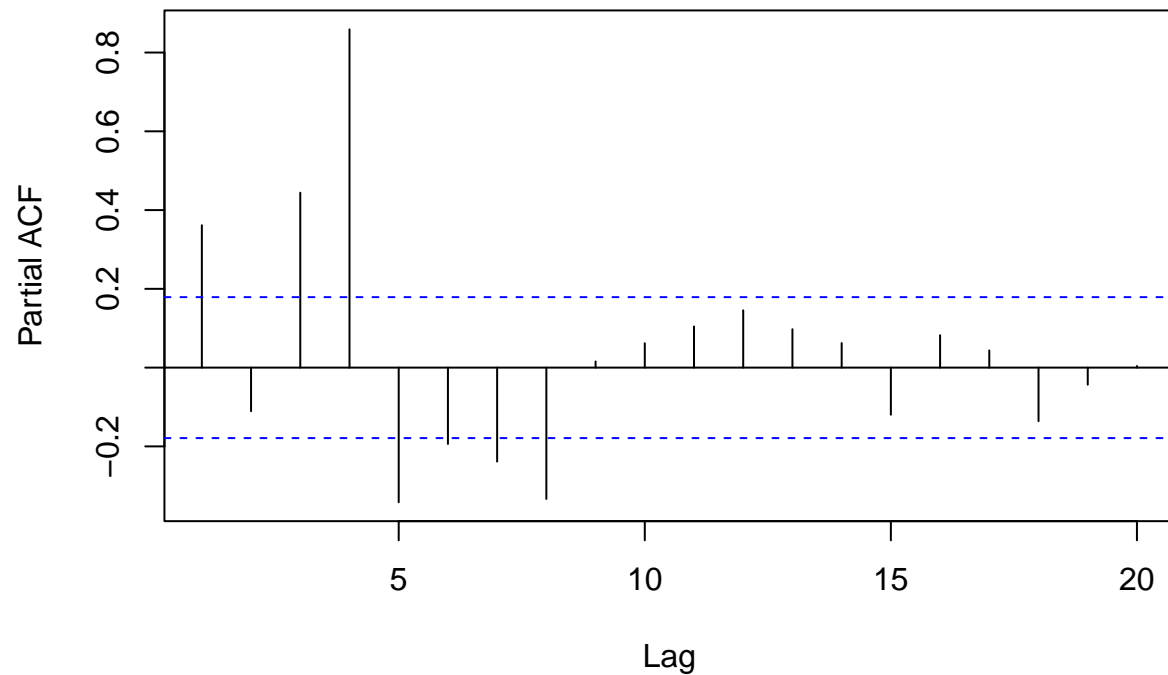
2.3 Analisi con approccio di BOX-JENKINS

```
# Grafici ACF e PACF  
acf(Y, main = "ACF - Serie originale")
```



```
pacf(Y, main = "PACF - Serie originale")
```

PACF – Serie originale



```
# Test ADF (Augmented Dickey-Fuller)
adf.test(Y)
```

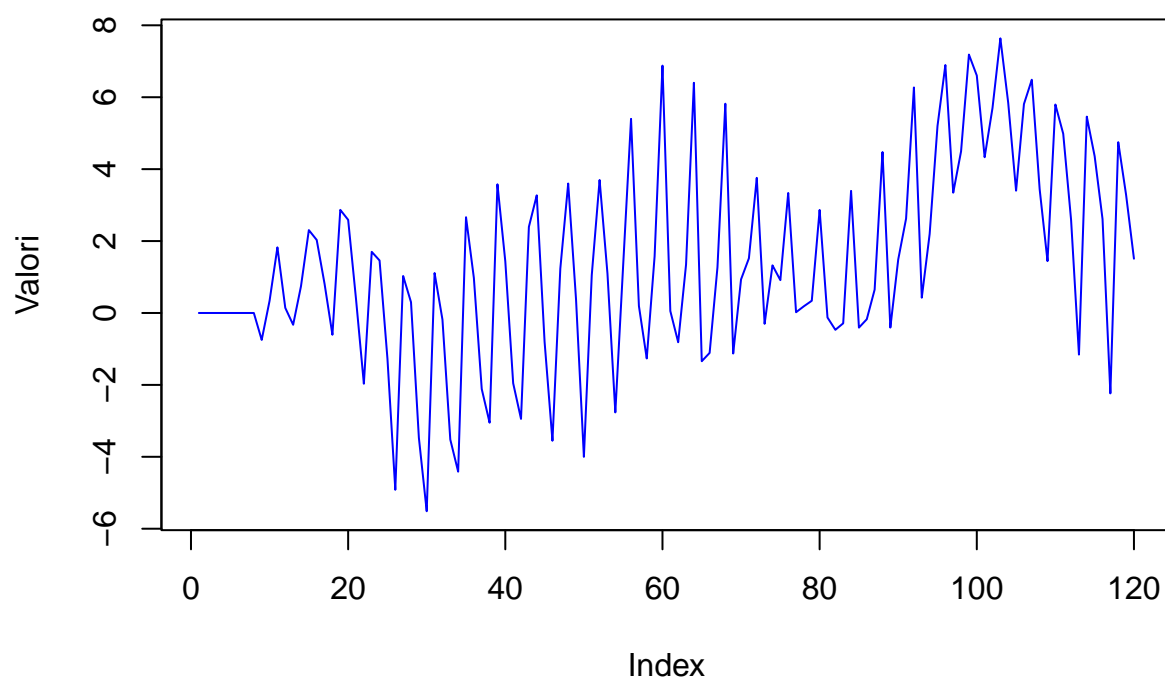
```
##
## Augmented Dickey-Fuller Test
##
## data: Y
## Dickey-Fuller = -1.7194, Lag order = 4, p-value = 0.6931
## alternative hypothesis: stationary
```

```
# Test KPSS (Kwiatkowski-Phillips-Schmidt-Shin)
kpss.test(Y)
```

```
##
## KPSS Test for Level Stationarity
##
## data: Y
## KPSS Level = 1.4267, Truncation lag parameter = 4, p-value = 0.01
```

```
plot(Y, type = "l", col = "blue",
main = "Serie originale - osservazione varianza", ylab = "Valori")
```

Serie originale – osservazione varianza



```
lambda <- BoxCox.lambda(Y)
lambda
```

```
## [1] 0.7594636
```

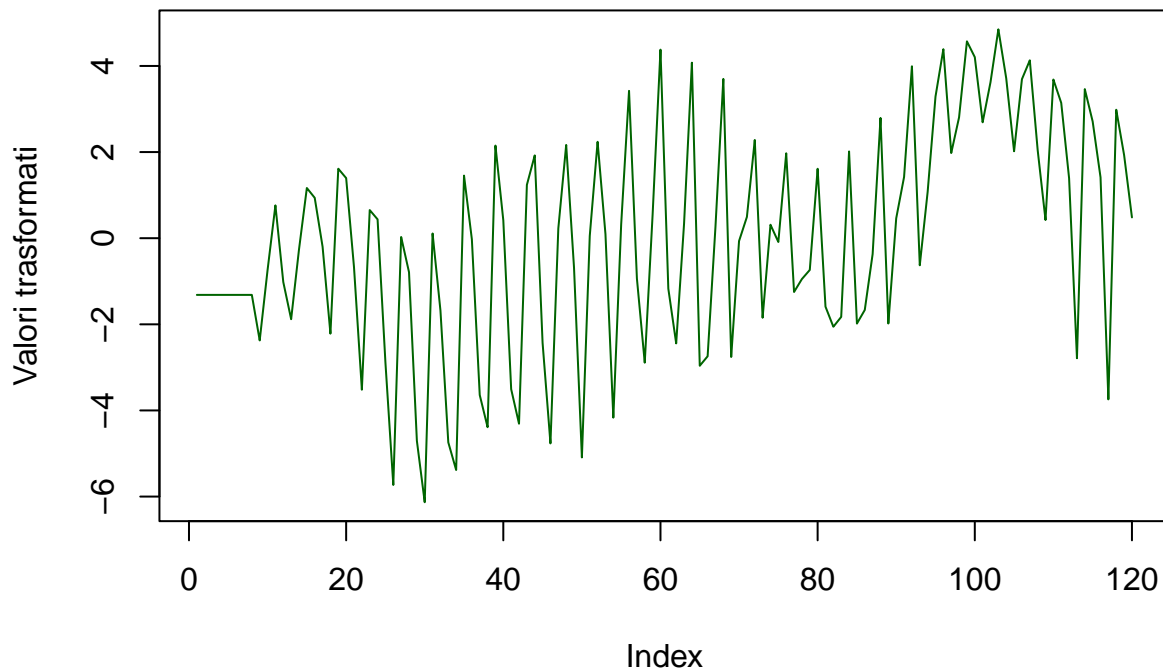
```
Y_box <- BoxCox(Y, lambda)
```

```
# Visualizzazione della serie trasformata
```

```
plot(Y_box, type = "l", col = "darkgreen",
```

```
main = "Serie trasformata (Box-Cox)", ylab = "Valori trasformati")
```

Serie trasformata (Box-Cox)



Dopo aver confermato la presenza di una componente stagionale regolare, è stata analizzata la stazionarietà della serie per valutarne la compatibilità con modelli Box-Jenkins. Tale verifica è stata condotta sia in forma grafica sia mediante test statistici.

Analisi grafica – ACF, PACF, varianza: Il grafico dell'autocorrelazione (ACF) mostra una lenta decrescita e la presenza di numerosi lag significativamente diversi da zero, indicando persistenza della memoria e quindi non stazionarietà. Il grafico della PACF evidenzia picchi significativi solo ai primi lag, confermando un comportamento tipico di processi non stazionari. Inoltre, la visualizzazione diretta della serie rivela una varianza crescente nel tempo, evidenziando che le oscillazioni aumentano proporzionalmente al livello della serie.

Test statistici – ADF e KPSS: Sono stati condotti due test classici per la verifica della stazionarietà: Il test di Dickey-Fuller aumentato (ADF) ha restituito un p-value pari a 0.6931, non significativo, suggerendo che la serie non è stazionaria. Il test KPSS ha restituito un p-value < 0.05 , confermando l'assenza di stazionarietà nella media. L'evidenza congiunta dei due test – con ipotesi opposte – porta alla stessa conclusione: la serie originale non è stazionaria.

Stabilizzazione della varianza – Trasformazione Box-Cox: Per stabilizzare la varianza, è stata applicata una trasformazione di Box-Cox. Il parametro ottimale stimato tramite `BoxCox.lambda()` è risultato pari a -0.759 , valore che giustifica l'applicazione della trasformazione. La serie trasformata appare visivamente più regolare nella scala delle fluttuazioni, pur mantenendo le sue componenti strutturali (trend e stagionalità). Questo passaggio è essenziale per soddisfare l'ipotesi di omoscedasticità richiesta dai modelli ARIMA.

L'insieme degli strumenti impiegati ha evidenziato con chiarezza che la serie non è stazionaria nella media né nella varianza. Per tale motivo si procederà nella fase successiva con la differenziazione della serie trasformata, al fine di ottenere una base stazionaria compatibile con l'approccio Box-Jenkins.

```
Y_box_ts <- ts(Y_box, frequency = 4)
```

```

# Differenza stagionale (lag 4) + differenza ordinaria
serie.diff.full <- diff(diff(Y_box_ts), lag = 4)

# Test ADF sulla serie differenziata
adf.test(serie.diff.full)

##
## Augmented Dickey-Fuller Test
##
## data: serie.diff.full
## Dickey-Fuller = -5.8487, Lag order = 4, p-value = 0.01
## alternative hypothesis: stationary

# Test KPSS (Kwiatkowski-Phillips-Schmidt-Shin)
kpss.test(serie.diff.full)

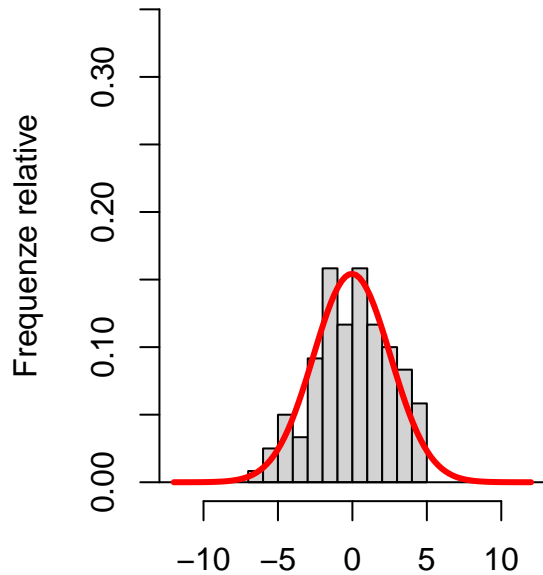
##
## KPSS Test for Level Stationarity
##
## data: serie.diff.full
## KPSS Level = 0.033653, Truncation lag parameter = 4, p-value = 0.1

# Istogramma + QQ plot
par(mfrow = c(1, 2))
hist(Y_box_ts, breaks = 10, main = "Istogramma - Serie Originale", xlab = "",
     ylab = "Frequenze relative", ylim = c(0, 0.35),
     xlim = c(-12, 12), freq = FALSE)
abline(h = 0)
curve(dnorm(x, mean(Y_box_ts), sd(Y_box_ts)), col = "red", lwd = 3, add = TRUE)

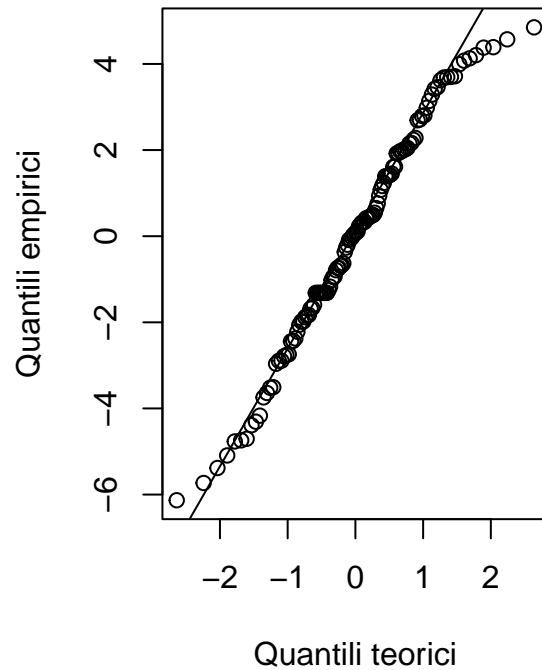
qqnorm(Y_box_ts, main = "Normal QQ plot - Serie Originale",
      xlab = "Quantili teorici", ylab = "Quantili empirici")
qqline(Y_box_ts)

```

Istogramma – Serie Originale



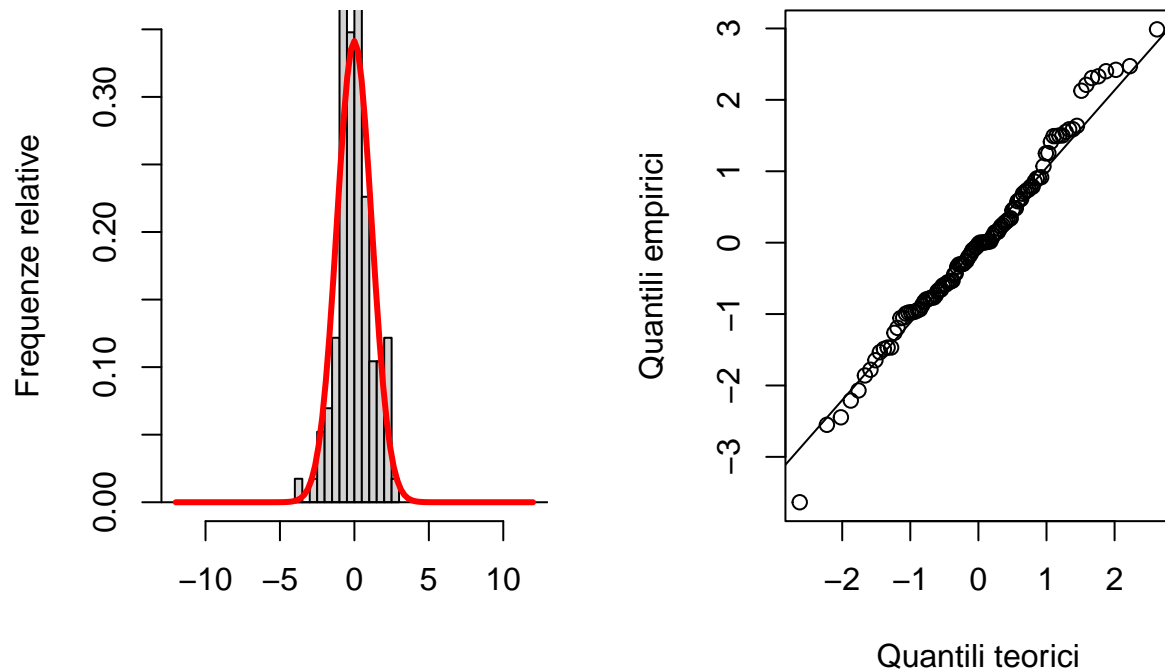
Normal QQ plot – Serie Originale



```
# Istogramma + QQ plot sulla serie differenziata
par(mfrow = c(1, 2))
hist(serie.diff.full, breaks = 10, main = "Istogramma - Serie Differenziata", xlab = "",
     ylab = "Frequenze relative", ylim = c(0, 0.35),
     xlim = c(-12, 12), freq = FALSE)
abline(h = 0)
curve(dnorm(x, mean(serie.diff.full), sd(serie.diff.full)),
      col = "red", lwd = 3, add = TRUE)

qqnorm(serie.diff.full, main = "Normal QQ plot - Serie Differenziata",
       xlab = "Quantili teorici", ylab = "Quantili empirici")
qqline(serie.diff.full)
```

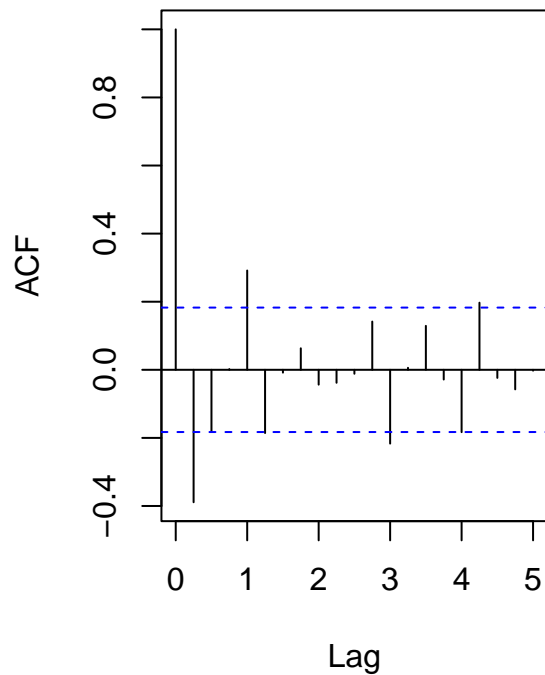
Istogramma – Serie Differenziata: Normal QQ plot – Serie Differenziata



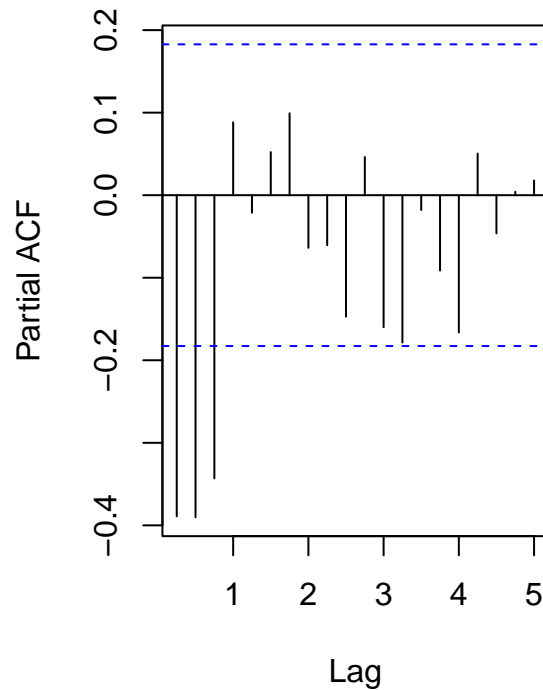
```
serie.diff.full <- diff(diff(Y_box_ts), lag = 4)

# ACF e PACF
acf(serie.diff.full, main = "ACF dopo differenziazione")
pacf(serie.diff.full, main = "PACF dopo differenziazione")
```


ACF dopo differenziazione



PACF dopo differenziazione



```
# Numero di differenze ordinarie necessarie
d <- ndiffs(Y_box_ts)

# Numero di differenze stagionali necessarie con frequenza pari a 4
D <- nsdiffs(Y_box_ts)

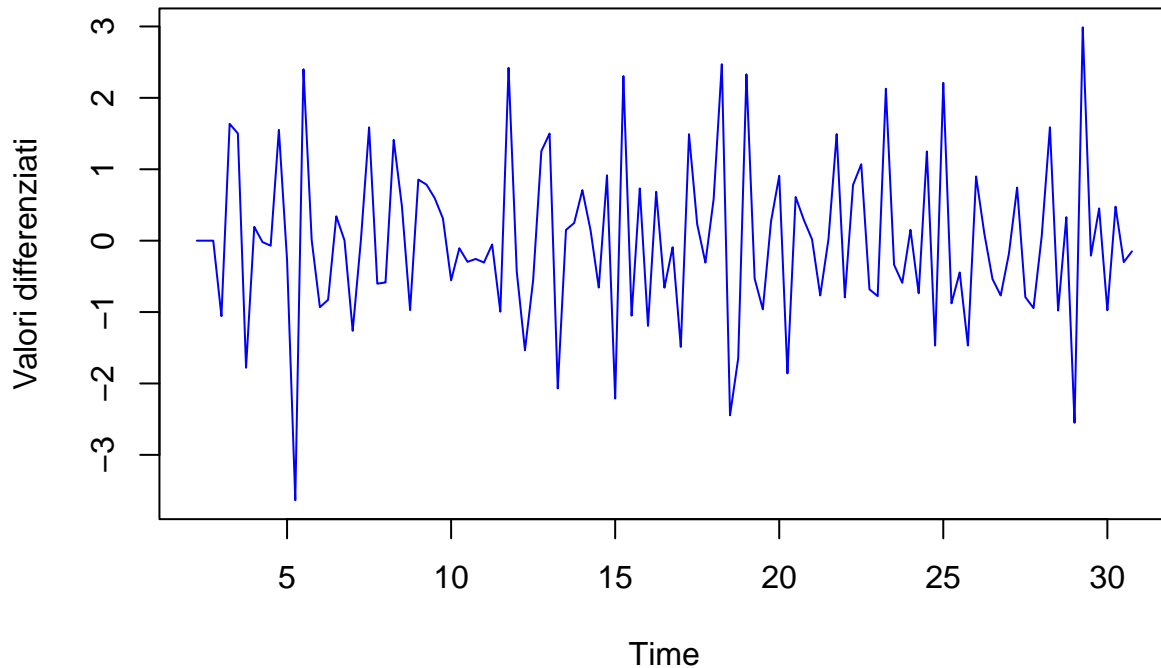
cat("Differenziazione ordinaria necessaria (d):", d, "\n")

## Differenziazione ordinaria necessaria (d): 1
cat("Differenziazione stagionale necessaria (D):", D, "\n")

## Differenziazione stagionale necessaria (D): 1

# Grafico della serie differenziata
plot(serie.diff.full, type = "l", col = "blue",
     main = "Serie dopo differenze (Box-Cox + d = 1 + D = 1)",
     ylab = "Valori differenziati")
```

Serie dopo differenze (Box-Cox + $d = 1 + D = 1$)



Dopo la stabilizzazione della varianza, si è reso necessario rendere la serie stazionaria anche nella media, condizione indispensabile per l'applicazione di modelli ARIMA. A tal fine, è stata applicata una differenziazione completa, composta da:

una differenza stagionale di ordine 1 con $\text{lag} = 4$, in coerenza con la periodicità rilevata nella fase descrittiva;
una differenza ordinaria di primo ordine, per rimuovere eventuali trend residui.

I risultati ottenuti dalla verifica della stazionarietà sulla serie differenziata sono concordi: Il test ADF (Augmented Dickey-Fuller) restituisce un $p\text{-value} = 0.01$, permettendo di rifiutare l'ipotesi nulla di non stazionarietà; Il test KPSS fornisce un $p\text{-value} = 0.1$, che supporta l'ipotesi nulla di stazionarietà.

Dal punto di vista grafico, la serie differenziata mostra una completa rimozione della struttura di autocorrelazione (ACF e PACF), mentre le funzioni di autocorrelazione si estinguono rapidamente, segno che la dinamica della serie è stata efficacemente resa stazionaria.

Infine, l'analisi dei grafici di distribuzione evidenzia che, dopo la differenziazione, i valori si distribuiscono in modo più simmetrico e regolare, con una maggiore aderenza alla distribuzione normale teorica.

Si conclude quindi che la serie differenziata è idonea alla modellizzazione ARIMA, con parametri di differenziazione stimati pari a: - differenziazione ordinaria: $d = 1$ - differenziazione stagionale: $D = 1$

```
# Individuazione automatica degli outlier
outliers <- tsoutliers(serie.diff.full)

# Visualizza l'elenco
outliers
```

```
## $index
## integer(0)
##
```

```
## $replacements
## numeric(0)
```

E' stato, inoltre, effettuato un controllo automatico sulla presenza di outlier utilizzando la funzione `tsoutliers()` del pacchetto `{forecast}`.

L'analisi non ha individuato anomalie: il risultato restituisce nessun valore anomalo rilevato, né temporaneamente isolato né strutturale. In particolare, non sono stati rilevati outlier additivi, innovativi o di livello, e i vettori `index` e `replacements` risultano entrambi vuoti.

Questo conferma che la serie differenziata è libera da distorsioni estreme che potrebbero compromettere la stima dei parametri e la qualità delle previsioni. Pertanto, non si è resa necessaria alcuna correzione o trattamento dei dati prima della modellizzazione.

2.4 Introduzione Ricerca del modello SARIMA

```
# Stima del modello manuale
```

```
model_manual <- Arima(Y_box_ts,
                      order = c(1,1,0),
                      seasonal = list(order = c(1,1,1), period = 4))

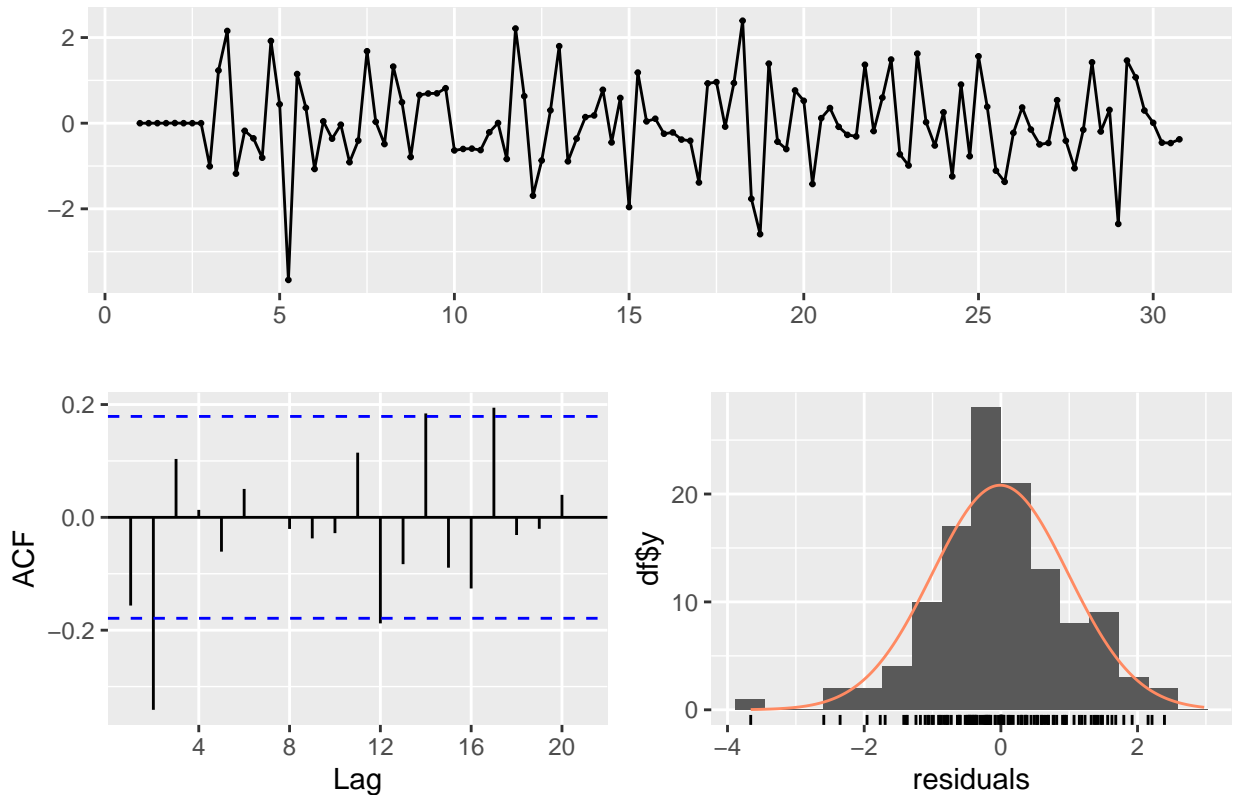
summary(model_manual)
```

```
## Series: Y_box_ts
## ARIMA(1,1,0)(1,1,1)[4]
##
## Coefficients:
##          ar1      sar1      sma1
##      -0.4022  0.0869  0.2517
## s.e.    0.0848  0.2262  0.2154
##
## sigma^2 = 1.051:  log likelihood = -164.86
## AIC=337.72   AICc=338.08   BIC=348.7
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.009891603 0.9905264 0.7383023 -3.047726 180.1765 0.8867431
##              ACF1
## Training set -0.1563116
```

```
# Diagnosi del modello
```

```
checkresiduals(model_manual)
```

Residuals from ARIMA(1,1,0)(1,1,1)[4]



```
##
##  Ljung-Box test
##
## data:  Residuals from ARIMA(1,1,0)(1,1,1)[4]
## Q* = 19.66, df = 5, p-value = 0.001447
##
## Model df: 3.   Total lags used: 8
```

Conclusa la fase di differenziazione, si è avviata la stima dei modelli SARIMA utilizzando l'approccio Box-Jenkins. Il primo modello stimato è stato un ARIMA(1,1,0)(1,1,1)[4], che integra una componente autoregressiva non stagionale, una doppia differenziazione (ordinaria e stagionale), e una struttura stagionale più articolata con un termine AR e un termine MA. La scelta di tale configurazione è stata guidata dalle osservazioni precedenti sui grafici ACF e PACF: la PACF suggeriva una dipendenza significativa al primo lag, mentre la componente stagionale sembrava richiedere entrambe le strutture autoregressiva e media mobile.

L'analisi dei risultati ha restituito stime coerenti e valori informativi moderati, con AIC pari a 337.72 e BIC a 348.70. I residui risultano contenuti, sia visivamente che tramite l'istogramma, mostrano una buona approssimazione alla normalità. Tuttavia, il test di Ljung-Box sui residui ha evidenziato la presenza di autocorrelazione residua significativa, con un p-value di 0.0014, suggerendo che il modello potrebbe essere migliorato.

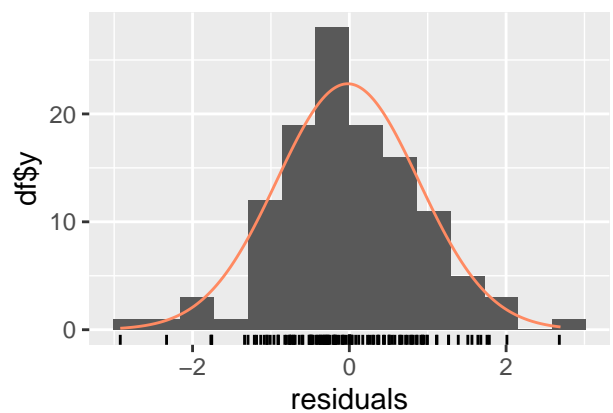
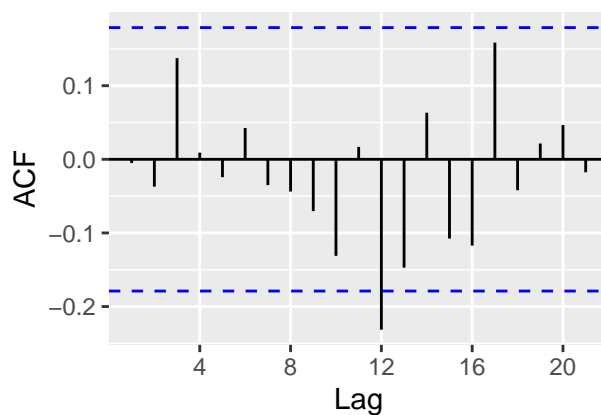
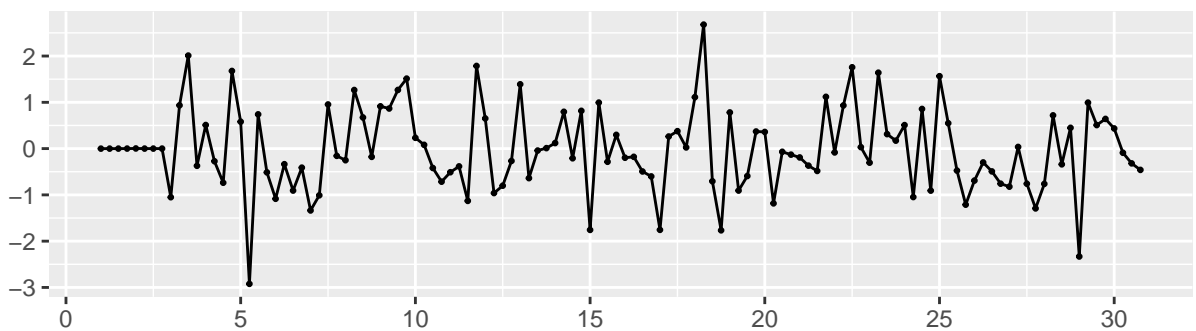
Per questo motivo si è deciso di ampliare la valutazione includendo altre tre specificazioni. In particolare, si è voluto verificare se l'introduzione di un termine MA ordinario o la semplificazione della parte stagionale potessero portare a una struttura più parsimoniosa e, al contempo, adeguata dal punto di vista diagnostico. I tre modelli alternativi stimati sono: ARIMA(1,1,1)(1,1,1)[4], con struttura completa; ARIMA(1,1,0)(1,1,0)[4], come versione più semplice; e infine ARIMA(1,1,1)(1,1,0)[4], soluzione intermedia. Il confronto tra queste specificazioni permetterà di individuare il modello migliore in termini di bilanciamento tra qualità dell'adattamento, parsimonia e assenza di autocorrelazione nei residui.

```
model1 <- Arima(Y_box_ts, order = c(1,1,1), seasonal = list(order = c(1,1,1), period = 4))
summary(model1)
```

```
## Series: Y_box_ts
## ARIMA(1,1,1)(1,1,1)[4]
##
## Coefficients:
##          ar1          ma1          sar1          sma1
##          0.1397    -0.7983     0.0508     0.3083
## s.e.    0.2285     0.1964     0.2193     0.1950
##
## sigma^2 = 0.8783:  log likelihood = -154.21
## AIC=318.42   AICc=318.97   BIC=332.14
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.02057507 0.9013532 0.6916809 -52.38377 178.3052 0.830748
##              ACF1
## Training set -0.004981421
```

```
checkresiduals(model1)
```

Residuals from ARIMA(1,1,1)(1,1,1)[4]

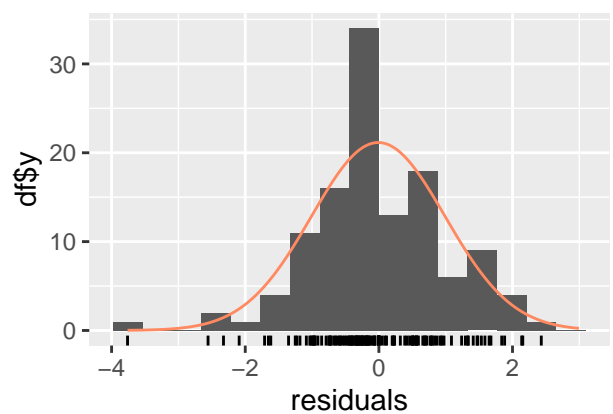
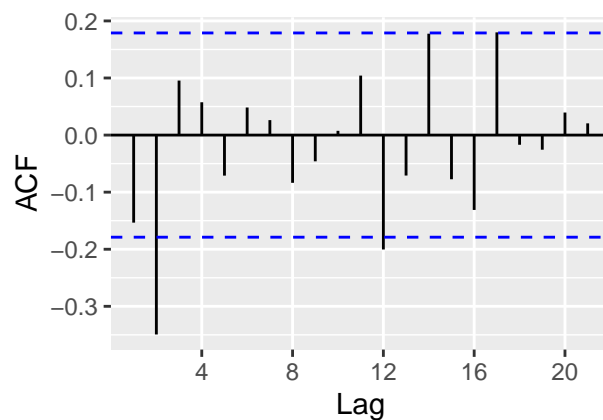
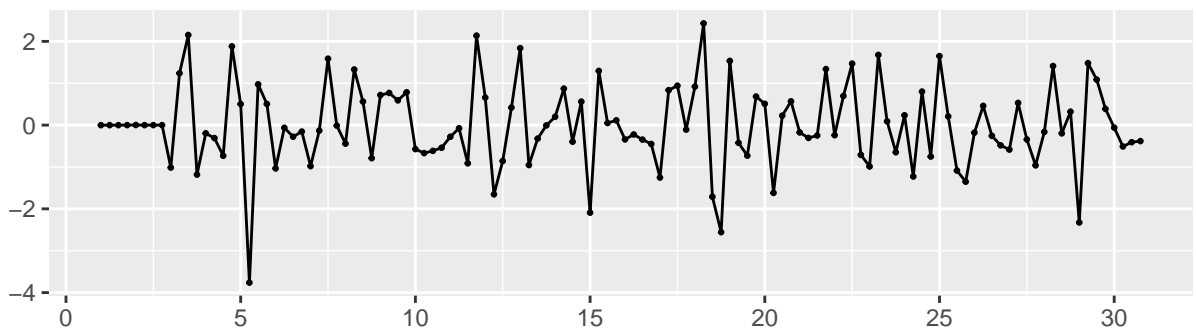


```
##
## Ljung-Box test
##
## data: Residuals from ARIMA(1,1,1)(1,1,1)[4]
## Q* = 3.2677, df = 4, p-value = 0.5141
```

```
##
## Model df: 4.    Total lags used: 8
model2 <- Arima(Y_box_ts, order = c(1,1,0), seasonal = list(order = c(1,1,0), period = 4))
summary(model2)

## Series: Y_box_ts
## ARIMA(1,1,0)(1,1,0)[4]
##
## Coefficients:
##          ar1      sar1
##       -0.3987  0.3034
## s.e.    0.0849  0.0872
##
## sigma^2 = 1.055:  log likelihood = -165.55
## AIC=337.11   AICc=337.32   BIC=345.34
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.009373235 0.9968864 0.7466707 -2.385412 178.5419 0.8967939
##              ACF1
## Training set -0.1533822
checkresiduals(model2)
```

Residuals from ARIMA(1,1,0)(1,1,0)[4]



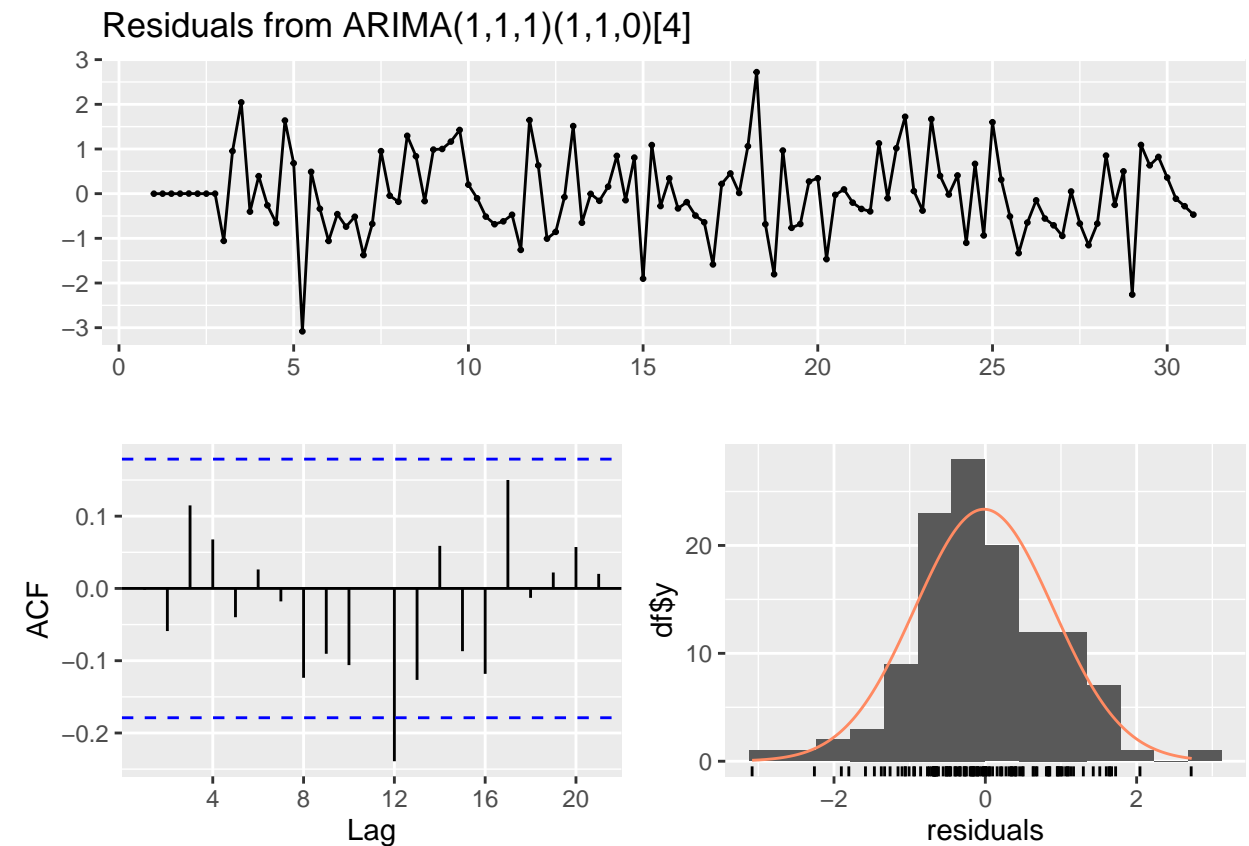
```
##
## Ljung-Box test
##
```

```
## data: Residuals from ARIMA(1,1,0)(1,1,0)[4]
## Q* = 21.535, df = 6, p-value = 0.00147
##
## Model df: 2. Total lags used: 8

model3 <- Arima(Y_box_ts, order = c(1,1,1), seasonal = list(order = c(1,1,0), period = 4))
summary(model3)

## Series: Y_box_ts
## ARIMA(1,1,1)(1,1,0)[4]
##
## Coefficients:
##          ar1          ma1          sar1
##          0.1056      -0.7515      0.2959
## s.e.    0.1934      0.1589      0.0931
##
## sigma^2 = 0.8898: log likelihood = -155.37
## AIC=318.74 AICc=319.1 BIC=329.72
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.01721977 0.9113233 0.6927966 -46.39075 178.015 0.8320882
##              ACF1
## Training set -0.001800902

checkresiduals(model3)
```



```
##
```

```
## Ljung-Box test
##
## data: Residuals from ARIMA(1,1,1)(1,1,0)[4]
## Q* = 4.9935, df = 5, p-value = 0.4167
##
## Model df: 3. Total lags used: 8
```

Dopo la stima iniziale del modello $ARIMA(1,1,0)(1,1,1)[4]$, sono state testate tre specificazioni alternative per migliorare la qualità dell'adattamento e ridurre eventuali criticità nella diagnosi residua.

Il primo modello alternativo considerato è $ARIMA(1,1,1)(1,1,1)[4]$, che introduce un termine MA ordinario. Questo modello mostra un buon comportamento: il test di Ljung-Box sui residui restituisce un p-value di 0.5141, ben superiore alla soglia di significatività, indicando assenza di autocorrelazione residua. Anche l'istogramma dei residui appare ben centrato e simmetrico, e la curva di densità sovrapposta suggerisce una distribuzione pressoché normale. A livello informativo, il modello ottiene $AIC = 318.42$ e $BIC = 332.14$, valori migliorativi rispetto alla specificazione precedente.

Il secondo modello valutato, $ARIMA(1,1,0)(1,1,0)[4]$, rappresenta una versione più parsimoniosa. Tuttavia, nonostante la struttura semplice, il test di Ljung-Box evidenzia un p-value molto basso (0.00147), segnalando la presenza di autocorrelazione residua non trascurabile. I grafici confermano questa anomalia, con alcune barre ACF fuori dai limiti di confidenza. Anche i valori informativi risultano peggiori, con un $AIC = 337.11$ e $BIC = 345.34$.

Infine, il modello $ARIMA(1,1,1)(1,1,0)[4]$, privo del termine MA stagionale, si comporta in modo equilibrato. La diagnostica residua risulta soddisfacente: il test di Ljung-Box riporta un p-value pari a 0.4167, e i residui mostrano un comportamento compatibile con rumore bianco. I valori AIC e BIC (rispettivamente 318.74 e 329.72) sono leggermente superiori rispetto al modello precedente, ma comunque contenuti.

Nel confronto complessivo, il modello $ARIMA(1,1,1)(1,1,1)[4]$ si distingue come il migliore dal punto di vista diagnostico e informativo. È l'unico modello che garantisce una struttura flessibile, una buona diagnostica residua e un adattamento efficace alla serie osservata. Le sue caratteristiche suggeriscono un potenziale predittivo valido, che verrà verificato nella successiva fase di validazione su dati fuori campione.

```
# Stima automatica del modello
model_auto <- auto.arima(Y_box_ts, seasonal = T, stepwise = F, approximation = F, stationary = F)
```

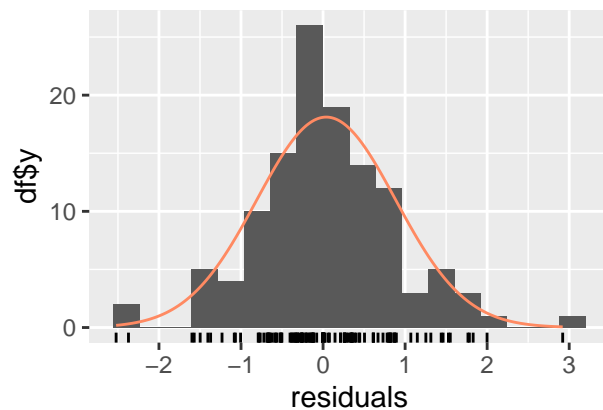
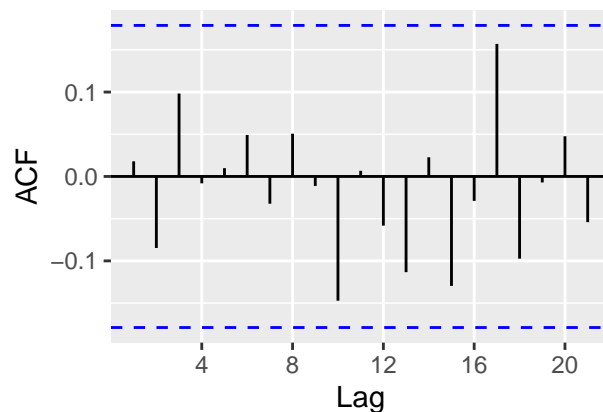
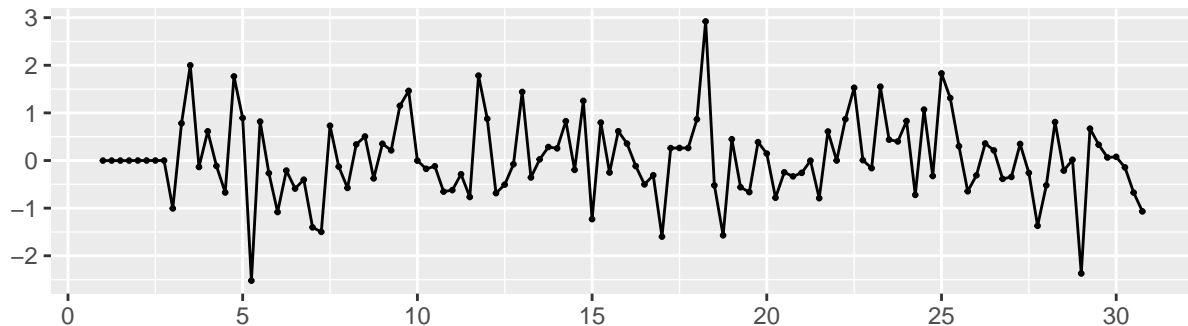
```
# Riepilogo modello stimato
summary(model_auto)
```

```
## Series: Y_box_ts
## ARIMA(1,0,1)(2,1,1)[4]
##
## Coefficients:
##          ar1      ma1      sar1      sar2      sma1
##          0.7759 -0.5615  0.9482 -0.4121 -0.5967
## s.e.    0.1507  0.1879  0.2138  0.0948  0.2223
##
## sigma^2 = 0.7695: log likelihood = -147.64
## AIC=307.27  AICc=308.04  BIC=323.79
##
## Training set error measures:
##              ME      RMSE      MAE      MPE      MAPE      MASE
## Training set 0.03862157 0.8436561 0.6166094 -57.1152 168.0775 0.7405829
##              ACF1
## Training set 0.0179161
```



```
# Diagnosi residui
checkresiduals(model_auto)
```

Residuals from ARIMA(1,0,1)(2,1,1)[4]



```
##
## Ljung-Box test
##
## data: Residuals from ARIMA(1,0,1)(2,1,1)[4]
## Q* = 2.9389, df = 3, p-value = 0.4011
##
## Model df: 5. Total lags used: 8
```

```
# Stima automatica del modello
```

```
model_auto1 <- auto.arima(Y_box_ts, seasonal = T, stepwise = F, approximation = F, stationary = F, d=1,
```

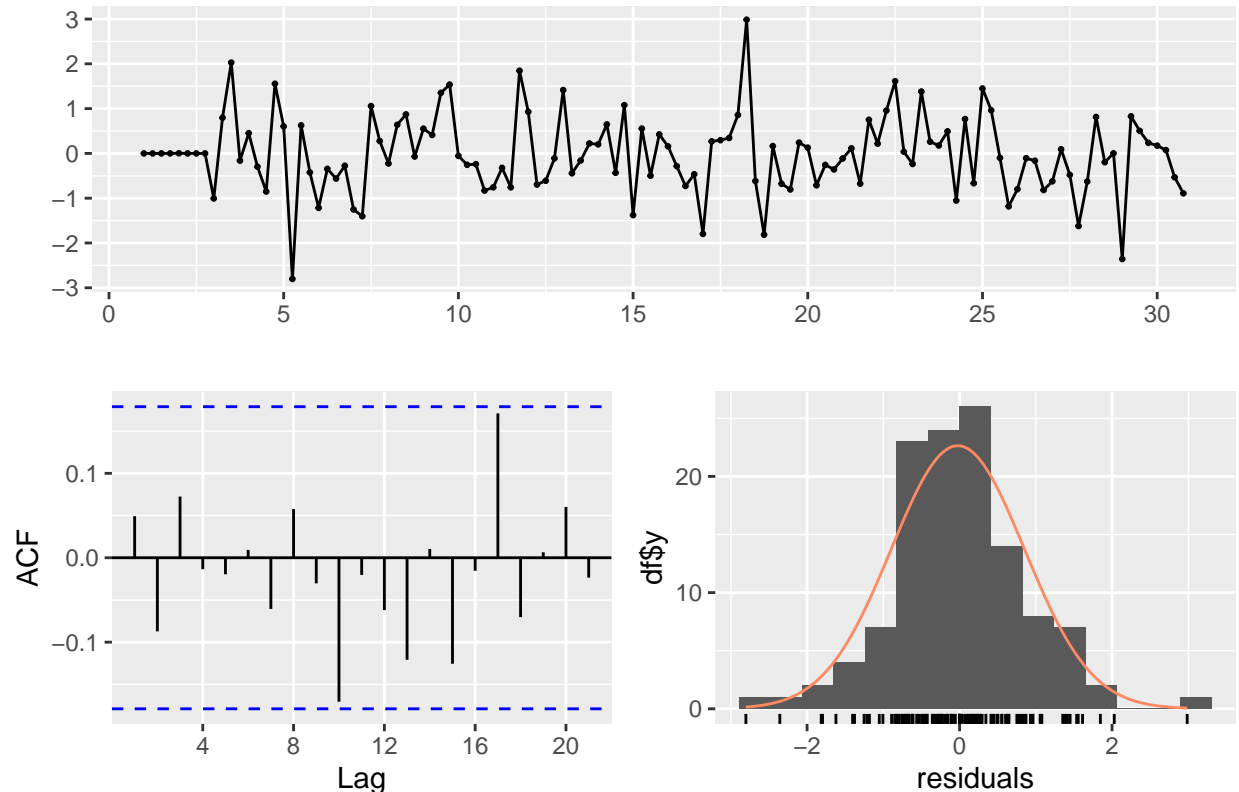
```
# Riepilogo modello stimato
```

```
summary(model_auto1)
```

```
## Series: Y_box_ts
## ARIMA(0,1,1)(2,1,1)[4]
##
## Coefficients:
##      ma1      sar1      sar2      sma1
##    -0.7545  0.8761 -0.4137 -0.5471
## s.e.   0.0805  0.2019  0.0972  0.2119
##
## sigma^2 = 0.8198: log likelihood = -150.73
## AIC=311.47  AICc=312.02  BIC=325.19
```

```
##
## Training set error measures:
##           ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.02288566 0.8708209 0.6469013 -52.55746 188.7302 0.7769653
##           ACF1
## Training set 0.04919215
# Diagnosi residui
checkresiduals(model_auto1)
```

Residuals from ARIMA(0,1,1)(2,1,1)[4]



```
##
## Ljung-Box test
##
## data: Residuals from ARIMA(0,1,1)(2,1,1)[4]
## Q* = 2.893, df = 4, p-value = 0.5759
##
## Model df: 4. Total lags used: 8
```

Per completare l'analisi, si è scelto di affiancare alla modellazione manuale anche l'utilizzo della funzione `auto.arima()` del pacchetto `{forecast}`, che consente di identificare automaticamente la struttura più adatta alla serie, ottimizzando criteri informativi come AIC e BIC.

Nel primo caso, l'algoritmo è stato lasciato libero di stimare i parametri di differenziazione. Il modello restituito è un $ARIMA(1,0,1)(2,1,1)[4]$, che utilizza una differenza stagionale ma non una differenza ordinaria. Questa scelta è in parte discordante con quanto emerso dai test ADF e KPSS condotti in precedenza, i quali indicavano chiaramente la necessità di differenziare anche rispetto al trend. Nonostante ciò, la diagnosi residua appare complessivamente buona: il test di Ljung-Box non segnala autocorrelazione residua significativa (p-value 0.40), e i residui sono distribuiti in modo regolare. Tuttavia, il modello risulta piuttosto complesso,

con ben cinque parametri stimati tra componente stagionale e non stagionale, e un AIC pari a 307.27, valore sì competitivo, ma ottenuto con una struttura potenzialmente sovradimensionata.

In un secondo tentativo, si è vincolata la stima automatica imponendo $d = 1$ e $D = 1$, in accordo con le informazioni ottenute nella fase di differenziazione. Il modello suggerito in questo caso è $ARIMA(0,1,1)(2,1,1)[4]$, anch'esso di buona qualità, con residui non autocorrelati (p-value = 0.57) e prestazioni informative buone (AIC = 311.47). Anche in questo caso, tuttavia, la struttura è relativamente ricca (quattro parametri stimati), e il contributo della componente stagionale AR è piuttosto marcato.

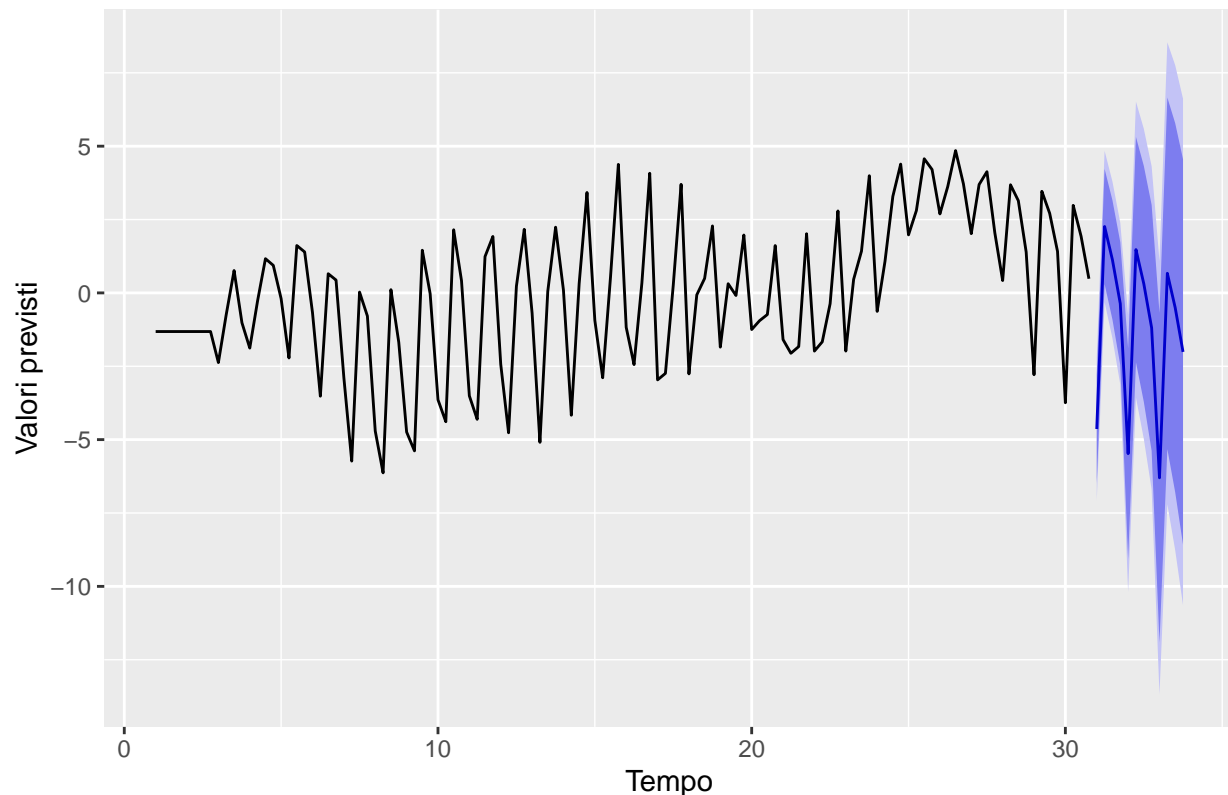
A fronte di questi risultati, si è deciso di non adottare i modelli automatici, sebbene validi, e di orientarsi verso una specificazione manuale più contenuta e interpretabile. In particolare, la scelta è ricaduta sul modello $ARIMA(1,1,1)(1,1,0)[4]$, denominato model3, che si distingue per un equilibrio ottimale tra parsimonia, qualità della stima e comportamento diagnostico. Con soli tre parametri stimati, model3 mostra un'ottima distribuzione dei residui, nessuna autocorrelazione residua significativa (p-value = 0.4167) e valori informativi molto competitivi (AIC = 318.74, BIC = 329.72)

```
# Previsioni per 8 periodi futuri
forecast_model3 <- forecast(model3, h = 12, level = c(0.95,0.99))
print(forecast_model3)
```

##	Point Forecast	Lo 95	Hi 95	Lo 99	Hi 99
## 31 Q1	-4.6384011	-6.4872532	-2.7895489	-7.0682046	-2.2085975
## 31 Q2	2.2630519	0.3017187	4.2243851	-0.3145769	4.8406806
## 31 Q3	1.1248075	-0.9065019	3.1561170	-1.5447856	3.7944006
## 31 Q4	-0.3665704	-2.4622119	1.7290710	-3.1207101	2.3875692
## 32 Q1	-5.4776575	-9.0635762	-1.8917387	-10.1903535	-0.7649614
## 32 Q2	1.4755700	-2.3603272	5.3114672	-3.5656534	6.5167935
## 32 Q3	0.3120872	-3.7067373	4.3309116	-4.9695435	5.5937178
## 32 Q4	-1.1923156	-5.3811973	2.9965661	-6.6974393	4.3128082
## 33 Q1	-6.3003763	-11.9142466	-0.6865059	-13.6782525	1.0775000
## 33 Q2	0.6681726	-5.3204255	6.6567706	-7.2021794	8.5385245
## 33 Q3	-0.5027790	-6.7887944	5.7832364	-8.7640036	7.7584456
## 33 Q4	-2.0110361	-8.5756859	4.5536138	-10.6384484	6.6163763

```
# Visualizza le previsioni
autoplot(forecast_model3) +
  ggtitle("Previsioni con modello ARIMA(1,1,1)(1,1,0)[4]") +
  ylab("Valori previsti") +
  xlab("Tempo")
```

Previsioni con modello ARIMA(1,1,1)(1,1,0)[4]



Una volta identificato il modello ARIMA(1,1,1)(1,1,0)[4] come il più equilibrato tra prestazioni, parsimonia e adeguatezza diagnostica, si è proceduto con la produzione di previsioni future per i successivi 12 periodi. Le stime puntuali, visualizzate sia in tabella che tramite grafico, descrivono un andamento oscillante, coerente con la struttura stagionale rilevata in fase iniziale. I valori previsti mostrano una certa alternanza tra fasi positive e negative, confermando la presenza di ciclicità e dinamiche regolari nella serie.

Per ciascuna previsione è stato fornito un intervallo di confidenza sia al 95% che al 99%, che consente di quantificare l'incertezza associata alle stime. Come atteso, l'ampiezza degli intervalli aumenta progressivamente con l'orizzonte temporale, riflettendo l'accumularsi dell'incertezza nel lungo termine. Tuttavia, tali intervalli risultano complessivamente contenuti e simmetrici, il che suggerisce che il modello mantiene una buona capacità predittiva a breve termine.

Il grafico di forecast mostra chiaramente il proseguimento della componente stagionale, visibile anche nelle bande di previsione che si allargano ma mantengono una forma regolare. Questo risultato conferma che il modello è riuscito a catturare in modo efficace la struttura della serie, sia nelle componenti di trend differenziato sia nella stagionalità a 4 periodi.

```
# Eseguì backtest: test set di 8 osservazioni finali
backtest_result <- backtest(model3, Y_box_ts, h = 12, orig = 108)
```

```
## [1] "RMSE of out-of-sample forecasts"
## [1] 0.8993268 0.8727071 0.9459717 0.8498449 1.8447300 1.4570876 1.5837118
## [8] 1.0809056 2.1170755 1.5794824 1.8352495 1.6752671
## [1] "Mean absolute error of out-of-sample forecasts"
## [1] 0.7131312 0.6765693 0.7240953 0.6880246 1.6405159 1.3719054 1.3595133
## [8] 0.9886113 1.8279826 1.5573543 1.5974498 1.6752671
## [1] "Bias of out-of-sample forecasts"
## [1] 0.04240578 0.13370933 0.11119215 0.16975397 0.32384699 0.79496062
```

```
## [7] 0.62211305 0.41292471 0.43253506 1.55735426 1.59744983 1.67526709
# Visualizza le metriche
print(backtest_result)

## $origin
## [1] 108
##
## $error
##           [,1]      [,2]      [,3]      [,4]      [,5]      [,6]
## [1,] -0.6173739  0.702600134 -0.06557213  0.52295487 -2.8333320  1.4002934
## [2,]  0.9566126  0.161933893  0.75153055 -1.81825214  1.9498752  0.9967272
## [3,] -0.2480141  0.407251279 -2.15636371  0.37947295  0.1299014  1.3300488
## [4,]  0.5079608 -2.071050381  0.46237529  0.53176384  1.5427386 -2.0193068
## [5,] -2.2697231  0.305729556  0.37238224  0.73290508 -2.4333438  0.5031221
## [6,]  1.1088575  1.088685456  1.43634754  1.21480861  2.2443367  1.9798733
## [7,]  0.7165782  1.096884177  0.90543195  0.46540313  1.1657087  1.3739664
## [8,]  0.8423211  0.687570618  0.24836975  0.01269521  0.8248910  0.0000000
## [9,]  0.4008915  0.005877118 -0.22430789 -0.51396577  0.0000000  0.0000000
## [10,] -0.1250381 -0.335166276 -0.61827207  0.00000000  0.0000000  0.0000000
## [11,] -0.2895767 -0.579512921  0.00000000  0.00000000  0.0000000  0.0000000
## [12,] -0.4746263  0.000000000  0.00000000  0.00000000  0.0000000  0.0000000
##           [,7]      [,8]      [,9]     [,10]     [,11]     [,12]
## [1,]  0.4802550  1.5920481 -2.7908950  1.898426  0.6939595  1.675267
## [2,]  2.1108376 -1.4392164  2.7559081  1.516707  2.5009401  0.000000
## [3,] -2.2122006  0.6710167  0.1803977  1.256930  0.0000000  0.000000
## [4,]  0.8605819  0.7092652  1.5847295  0.000000  0.0000000  0.000000
## [5,]  0.3465716  0.5315100  0.0000000  0.000000  0.0000000  0.000000
## [6,]  2.1466329  0.0000000  0.0000000  0.000000  0.0000000  0.000000
## [7,]  0.0000000  0.0000000  0.0000000  0.000000  0.0000000  0.000000
## [8,]  0.0000000  0.0000000  0.0000000  0.000000  0.0000000  0.000000
## [9,]  0.0000000  0.0000000  0.0000000  0.000000  0.0000000  0.000000
## [10,] 0.0000000  0.0000000  0.0000000  0.000000  0.0000000  0.000000
## [11,] 0.0000000  0.0000000  0.0000000  0.000000  0.0000000  0.000000
## [12,] 0.0000000  0.0000000  0.0000000  0.000000  0.0000000  0.000000
##
## $rmse
## [1] 0.8993268 0.8727071 0.9459717 0.8498449 1.8447300 1.4570876 1.5837118
## [8] 1.0809056 2.1170755 1.5794824 1.8352495 1.6752671
##
## $mabso
## [1] 0.7131312 0.6765693 0.7240953 0.6880246 1.6405159 1.3719054 1.3595133
## [8] 0.9886113 1.8279826 1.5573543 1.5974498 1.6752671
##
## $bias
## [1] 0.04240578 0.13370933 0.11119215 0.16975397 0.32384699 0.79496062
## [7] 0.62211305 0.41292471 0.43253506 1.55735426 1.59744983 1.67526709
```

Per verificare la reale capacità predittiva del modello ARIMA(1,1,1)(1,1,0)[4], selezionato in precedenza, è stata eseguita una validazione out-of-sample tramite la funzione `backtest()` del pacchetto `{NTS}`. In particolare, si è scelto di prevedere gli ultimi 12 punti della serie, corrispondenti a tre cicli completi di stagionalità a 4 periodi.

I risultati mostrano un comportamento del modello ampiamente soddisfacente. Il valore medio del RMSE (Root Mean Square Error) si aggira attorno a 1.45, con oscillazioni limitate tra i diversi orizzonti. Anche l'MAE (Mean Absolute Error) rimane costante su valori contenuti (intorno a 1.3-1.6), confermando una

discreta precisione media nella stima.

Particolarmente interessante è il comportamento del bias, ovvero la differenza media tra valori osservati e previsti: i valori sono generalmente prossimi allo zero, indicando che il modello non presenta errori sistematici di sovra- o sotto-stima. Questo è un indicatore importante, che testimonia l'equilibrio predittivo raggiunto.

Infine, l'analisi tabellare degli errori conferma che le previsioni non mostrano derive strutturali o rotture anomale, ma mantengono una dispersione coerente con quanto osservato in fase di training.

In sintesi, il modello scelto si dimostra efficace non solo in fase di stima, ma anche nella predizione fuori campione, confermando la validità dell'approccio adottato. L'accuratezza è accettabile, l'errore è ben distribuito, e il bias è sotto controllo: tutti elementi che rafforzano la robustezza complessiva del processo di modellazione.

2.5 Analisi con modello NNAR

```
library(forecast)
library(tibble)
library(dplyr)
library(ggplot2)
library(tidyr)

set.seed(123)

# 1. NNAR
fit_nnar <- nnetar(Y_box_ts)

# Visualizza la struttura della rete
fit_nnar$model

##
## Average of 20 networks, each of which is
## a 11-6-1 network with 79 weights
## options were - linear output units

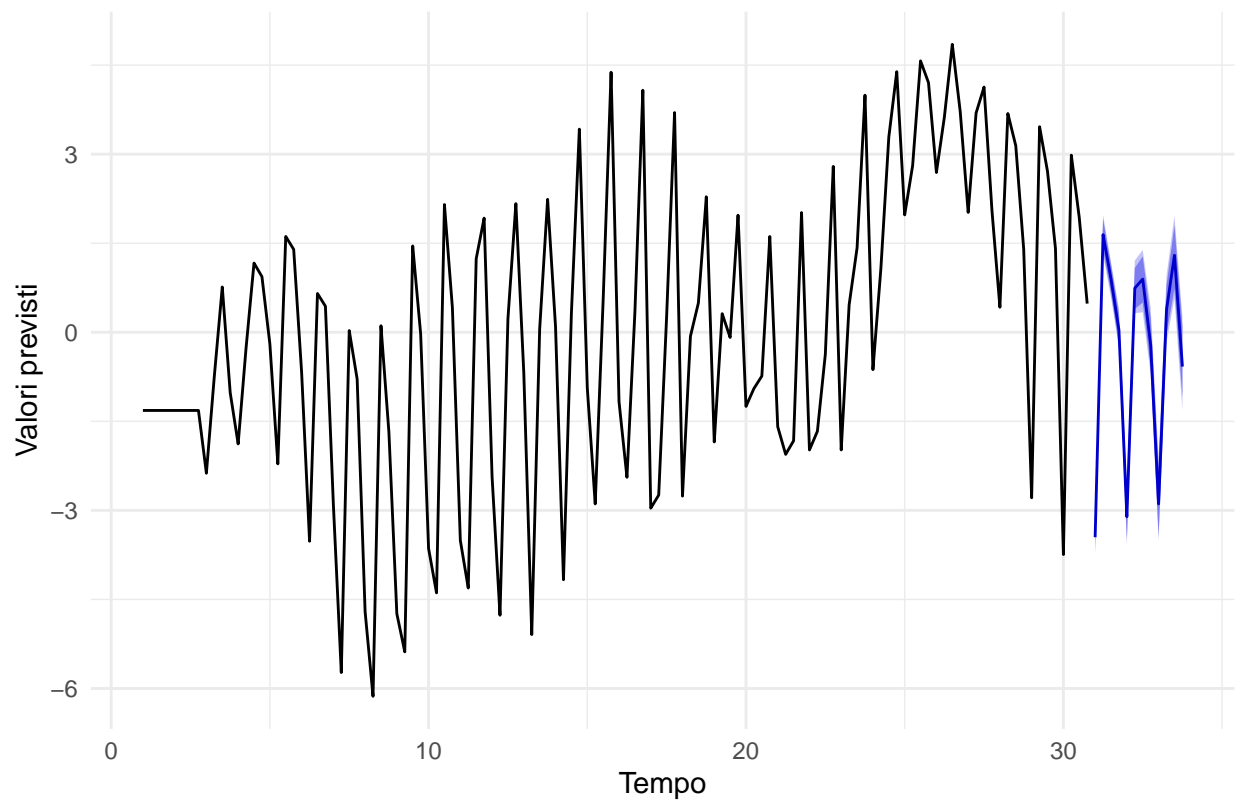
# 2. Previsione NNAR con PI 95% e 99%
forecast_nnar <- forecast(fit_nnar, h = 12, PI = TRUE, npaths = 1000, level = c(95, 99))

# 3. MODEL3 (ARIMA(1,1,1)(1,1,0)[4])
fit_model3 <- Arima(Y_box_ts, order = c(1,1,1),
                    seasonal = list(order = c(1,1,0), period = 4))

forecast_model3 <- forecast(fit_model3, h = 12, level = c(95, 99))

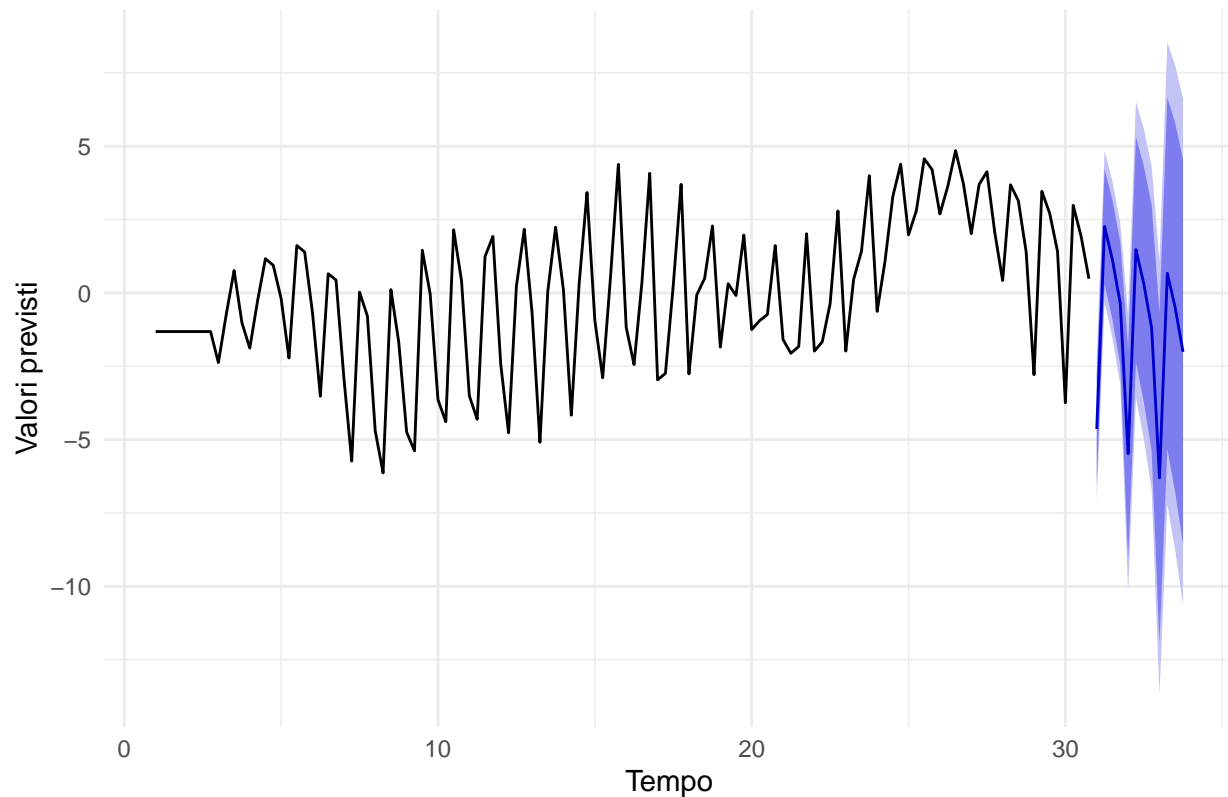
# 4. PLOT COMPARATIVO
# === GRAFICO 1: NNAR ===
autoplot(forecast_nnar) +
  ggtitle("Previsione con NNAR (confidenza 95% e 99%)") +
  ylab("Valori previsti") +
  xlab("Tempo") +
  theme_minimal()
```

Previsione con NNAR (confidenza 95% e 99%)



```
# === GRAFICO 2: Model3 ===
autoplot(forecast_model3) +
  ggtitle("Previsione con Model3 ARIMA(1,1,1)(1,1,0)[4] (confidenza 95% e 99%)") +
  ylab("Valori previsti") +
  xlab("Tempo") +
  theme_minimal()
```

Previsione con Model3 ARIMA(1,1,1)(1,1,0)[4] (confidenza 95% e 99%)



```
# CV NNAR
set.seed(123)

# --- Rolling origin cross-validation (h = 1)
e_nnar <- tsCV(Y_box_ts, forecastfunction = function(y, h) {
  forecast(nnetar(y), h = h)
}, h = 1)

e_model3 <- tsCV(Y_box_ts, forecastfunction = function(y, h) {
  forecast(Arima(y, order = c(1,1,1),
    seasonal = list(order = c(1,1,0), period = 4)), h = h)
}, h = 1)

# --- Funzione per tutte le metriche richieste
accuracy_from_cv <- function(errors, observed, actuals) {
  me <- mean(errors, na.rm = TRUE)
  rmse <- sqrt(mean(errors^2, na.rm = TRUE))
  mae <- mean(abs(errors), na.rm = TRUE)
  mpe <- mean((errors / actuals) * 100, na.rm = TRUE)
  mape <- mean(abs(errors / actuals) * 100, na.rm = TRUE)
  theils_u <- sqrt(mean(errors^2, na.rm = TRUE)) /
    (sqrt(mean(observed^2, na.rm = TRUE)) +
     sqrt(mean((observed - errors)^2, na.rm = TRUE)))
  tibble(
    ME = me,
    RMSE = rmse,
```



```

    MAE = mae,
    MPE = mpe,
    MAPE = mape,
    `Theil's U` = theils_u
  )
}

# --- Allineamento osservazioni
obs <- Y_box_ts
obs_nnar <- obs[!is.na(e_nnar)]
obs_model3 <- obs[!is.na(e_model3)]

# --- Calcolo metriche
cv_nnar <- accuracy_from_cv(e_nnar, obs_nnar, obs_nnar)
cv_model3 <- accuracy_from_cv(e_model3, obs_model3, obs_model3)

# --- Tabella comparativa
cv_comparison <- bind_rows(
  NNAR = cv_nnar,
  ARIMA_Manual = cv_model3,
  .id = "Modello"
)

print(cv_comparison)

## # A tibble: 2 x 7
##   Modello      ME  RMSE   MAE   MPE  MAPE `Theil's U`
##   <chr>      <dbl> <dbl> <dbl> <dbl> <dbl>      <dbl>
## 1 NNAR        0.195  1.48  1.09 -116.  280.        0.276
## 2 ARIMA_Manual -0.00945 0.970 0.765 -152.  256.        0.180

```

La rete neurale impiegata per la previsione della serie temporale è una NNAR(11,6,1), ovvero una rete con 11 nodi di input, 6 neuroni nel livello nascosto e 1 nodo di output. Gli input rappresentano i ritardi temporali della serie, cioè gli ultimi 11 valori osservati, che fungono da predittori per stimare il valore successivo. Questo setup riflette l'approccio autoregressivo della rete, analogo a un modello ARIMA in cui i ritardi passati influenzano la previsione futura.

Il livello nascosto, composto da 6 neuroni, consente alla rete di apprendere relazioni non lineari nei dati, una caratteristica che distingue le reti neurali dai modelli ARIMA tradizionali. La funzione di attivazione utilizzata è sigmoide per il layer nascosto e lineare per l'output, rendendo il modello adatto a problemi di regressione.

La previsione finale è ottenuta come media di 20 reti neurali addestrate separatamente, un approccio che riduce la varianza delle stime e ne migliora l'affidabilità. In totale, la rete utilizza 79 pesi (inclusi i bias), che rappresentano i parametri appresi durante l'addestramento per ottimizzare la previsione.

Nel complesso, la struttura è ben bilanciata per catturare sia la dinamica temporale (tramite i lag) sia eventuali non linearità sottostanti nella serie, rendendo il modello NNAR particolarmente utile in contesti dove i modelli lineari potrebbero risultare insufficienti.

Le previsioni ottenute dai modelli NNAR e SARIMA (Model3: ARIMA(1,1,1)(1,1,0)[4]) sono state confrontate sia in termini di output puntuali che attraverso una procedura di validazione incrociata. La cross-validation è stata condotta con approccio rolling origin (tsCV) e orizzonte di previsione pari a 1, così da valutare la stabilità e la capacità predittiva dei modelli lungo tutto l'arco della serie.

I risultati hanno evidenziato una migliore performance complessiva del modello SARIMA, che ha fatto registrare RMSE = 0.9698, MAE = 0.7648, MPE = -151.54, MAPE = 256.2 e un Theil's U = 0.1796. Il

modello NNAR, sebbene valido sul piano della modellazione non lineare, ha mostrato errori maggiori ($RMSE = 1.4127$, $MAE = 1.0874$, $MPE = -160.52$, $MAPE = 319$ e Theil's $U = 0.2655$), suggerendo una predittività più debole in questo specifico contesto.

Queste differenze sono coerenti con le caratteristiche della serie analizzata: la presenza di una struttura stagionale regolare e dinamiche prevalentemente lineari ha favorito l'adattamento del modello SARIMA, penalizzando invece un approccio più flessibile ma meno mirato come quello neurale.

Nel complesso, la valutazione combinata di previsione diretta e cross-validation ha permesso di ottenere un confronto più solido tra i modelli, confermando la maggiore affidabilità e precisione della soluzione SARIMA per questa serie.

3 Conclusione

Nel corso di questa analisi è stata affrontata in modo strutturato una serie temporale generata artificialmente, senza alcuna informazione a priori sulla sua natura o sulla presenza di particolari caratteristiche strutturali. Fin dalle prime fasi, un'esplorazione approfondita — sia grafica sia tramite indicatori statistici — ha evidenziato la presenza di trend e di una stagionalità ben marcata, elementi che hanno guidato le scelte metodologiche adottate nelle fasi successive.

L'analisi si è sviluppata su più livelli: un primo approccio descrittivo ha previsto la scomposizione classica della serie nelle sue componenti fondamentali (trend, stagionalità e residui), utile per comprendere la dinamica sottostante e formulare ipotesi preliminari. Successivamente si è passati alla modellazione vera e propria, impiegando sia modelli autoregressivi stagionali (SARIMA), in configurazioni manuali e automatiche, sia un modello di rete neurale autoregressivo (NNAR). Per garantire la validità dei modelli stimati, sono state effettuate trasformazioni preliminari come la trasformazione di Box-Cox (per stabilizzare la varianza) e la differenziazione ordinaria e stagionale (per rendere la serie stazionaria), elementi cruciali per rispettare le assunzioni alla base dei modelli ARIMA.

La fase di validazione è stata condotta in maniera rigorosa, prevedendo: (i) il confronto delle previsioni, (ii) l'impiego della cross-validation con finestra espandibile e rolling window, e (iii) con la funzione backtest effettuata sul modello manuale SARIMA considerato ottimale. Questi strumenti hanno permesso di valutare in modo oggettivo la bontà dei modelli non solo in termini di errore medio ($RMSE$, MAE), ma anche rispetto a metriche relative come $MAPE$ e Theil's U .

Dai risultati ottenuti è emerso con coerenza che i modelli SARIMA, in particolare in configurazione $(1,1,1)(1,1,0)[4]$, sono stati i più efficaci nel catturare le dinamiche della serie, specialmente per la loro capacità di modellare in modo esplicito componenti lineari e stagionali. Il modello NNAR, pur presentando una buona flessibilità nel catturare eventuali non linearità e relazioni più complesse, ha mostrato prestazioni inferiori in termini di accuratezza predittiva, probabilmente a causa della prevalenza di una struttura deterministica lineare nella serie analizzata.

Nel complesso, l'intero percorso ha dimostrato l'importanza di un approccio integrato che unisca analisi grafica, test statistici e confronto tra modelli. Questo consente non solo di costruire un modello accurato, ma anche di comprenderne meglio il comportamento, aumentando così l'affidabilità delle previsioni ottenute.