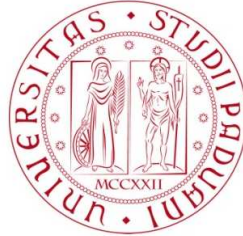


UNIVERSITÀ DEGLI STUDI DI PADOVA  
DIPARTIMENTO DI SCIENZE STATISTICHE  
CORSO DI LAUREA TRIENNALE IN  
STATISTICA PER LE TECNOLOGIE E LE SCIENZE



RELAZIONE FINALE  
**Metodologie bagging per le previsioni di  
serie storiche**

Relatrice Prof.ssa Luisa Bisaglia  
Dipartimento di Scienze Statistiche

Laureanda Alessia Adriani  
Matricola 2002490

Anno Accademico 2022/2023

# Indice

<b>Introduzione</b>	<b>1</b>
<b>1 Metodi di previsione a confronto</b>	<b>4</b>
1.1 Le serie storiche . . . . .	4
1.2 Il lisciamiento esponenziale . . . . .	5
1.2.1 Metodi di lisciamiento esponenziale . . . . .	5
1.2.2 Modelli di lisciamiento esponenziale . . . . .	6
1.3 Procedura bagging . . . . .	8
1.3.1 Trasformazione di Box-Cox . . . . .	9
1.3.2 Decomposizione STL . . . . .	9
1.3.3 Bootstrap dei residui e calcolo delle previsioni . . . . .	12
<b>2 Metodologie bootstrap per serie storiche</b>	<b>13</b>
2.1 Introduzione . . . . .	13
2.2 Bootstrap per dati indipendenti . . . . .	14
2.3 Bootstrap per serie storiche . . . . .	15
2.3.1 Bootstrap parametrico . . . . .	16
2.3.2 Bootstrap non parametrico . . . . .	18
2.3.3 Sieve bootstrap . . . . .	21
<b>3 Analisi dei dati e valutazione dei risultati</b>	<b>24</b>
3.1 M3-competition . . . . .	24
3.2 Indici per la valutazione degli errori di previsione . . . . .	26
3.3 Confronto delle metodologie bootstrap . . . . .	28
3.4 Risultati e conclusioni . . . . .	29
3.4.1 Risultati per le serie annuali . . . . .	30
3.4.2 Risultati per le serie trimestrali . . . . .	31
3.4.3 Risultati per le serie mensili . . . . .	32
3.4.4 Considerazioni finali . . . . .	33

<b>A</b>		<b>35</b>
A.1	Codice R utilizzato . . . . .	35
A.2	Ringraziamenti . . . . .	49

# Introduzione

Nell'ambito dell'analisi delle serie storiche, uno dei principali obiettivi è dato dalla capacità di poter fornire delle previsioni.

Fare previsione significa fare inferenza sui valori futuri, utilizzando le informazioni disponibili nella maniera più accurata possibile.

Le previsioni di un fenomeno sono necessarie per molteplici ragioni e in diversi contesti: esse fungono da supporto nei processi decisionali, facilitano la gestione del rischio e delle risorse, sono indispensabili per la prevenzione dei problemi e nel campo della pianificazione.

Ad oggi tra i metodi di previsione maggiormente utilizzati, soprattutto in azienda, rientrano quelli basati sul lisciamiento esponenziale (si veda, ad esempio, Goodwin, 2010) e le previsioni mediante i modelli autoregressivi integrati a media mobile (modelli ARIMA).

Tuttavia, negli ultimi anni sono state proposte ulteriori tecniche, tra cui il metodo basato sul *bagging* (*bootstrap aggregation*). Tale metodo consiste nel generare delle replicazioni *bootstrap* della serie originale, calcolare per ciascuna replicazione le previsioni  $h$  passi in avanti (previsioni *bootstrap*) ed utilizzare un'aggregazione (*bagging*) delle previsioni *bootstrap* come valore finale di previsione  $h$  passi in avanti per la serie in esame. Generalmente le previsioni *bootstrap* vengono aggregate utilizzando un indicatore di sintesi, come la media o la mediana.

L'obiettivo di questa tesi è quello di valutare l'efficacia della metodologia *bagging*, dimostrando come quest'ultima fornisca previsioni altrettanto accurate, e in alcuni casi in misura maggiore, rispetto a quelle ottenute utilizzando il metodo di lisciamiento esponenziale.

Il lavoro di tesi segue ed amplia l'approccio descritto da Bergmeir *et al.* (2016), che applica il *bagging* utilizzando una specifica metodologia *bootstrap*, il *moving block bootstrap*, per generare le replicazioni delle serie storiche. La procedura proposta viene rielaborata ed estesa, andando a considerare due ulteriori tipologie di *bootstrap*: il *sieve bootstrap* ed il *wild bootstrap*.

Nel primo capitolo viene fatto un richiamo dei concetti di base delle se-

rie storiche. Viene descritto il metodo classico di previsione basato sul lisciamiento esponenziale. In seguito, viene approfondita nel dettaglio la procedura sottostante la tecnica *bagging*.

Nel secondo capitolo verrà trattato il tema del *bootstrap*, con particolare attenzione al principio di funzionamento e verranno descritte le metodologie *bootstrap* utilizzate per generare le serie.

Il terzo capitolo è dedicato all'applicazione della tecnica *bagging* su dati reali: la validità della procedura viene dimostrata utilizzando i dati della *M3-competition*, contenente serie storiche di tipo mensile, trimestrale e annuale. Dopo una breve descrizione del dataset, verrà fornita una spiegazione degli indici utilizzati per valutare l'accuratezza delle previsioni e del test adoperato per confrontare le diverse tecniche *bootstrap* adattate. Infine, verranno riportati i risultati delle analisi, suddivisi in base al periodo delle serie.

Le analisi sono state eseguite con il software R nella versione 4.1.2. In appendice è riportato il codice R utilizzato per effettuare le analisi.

# Capitolo 1

## Metodi di previsione a confronto

Dopo una breve panoramica sui concetti alla base delle serie storiche, nel capitolo verranno spiegate nel dettaglio le due metodologie di previsione da analizzare: quella basata sui modelli di lisciamiento esponenziale e la procedura *bagging*.

### 1.1 Le serie storiche

Una serie storica è una sequenza di valori ordinati in base al tempo, generalmente indicata con la dicitura

$$\{y_t\}_{t=1}^n$$

dove  $y_t$  è il valore associato al particolare istante temporale  $t$  ed  $n$  è la lunghezza della serie. Ciò che caratterizza una serie storica è la dipendenza, legata al tempo, delle osservazioni successive. Quest'ultima viene assunta come realizzazione di un processo stocastico. Tale processo viene modellato mediante la somma di due componenti:

$$Y_t = f(t) + u_t, \quad (1.1)$$

dove  $f(t)$  rappresenta la componente sistematica della serie, mentre  $u_t$  costituisce la componente stocastica che obbedisce ad una determinata legge di probabilità.

Per le serie storiche generate dalla (1.1) si assume che la parte sistematica  $f(t)$  sia la somma di tre componenti non individualmente osservabili: il trend, il ciclo e la stagionalità. Il trend descrive l'andamento di lungo

periodo della serie, il ciclo è costituito da una sequenza di fluttuazioni generalmente collegate alle fasi di contrazione ed espansione del sistema economico, mentre la stagionalità è costituita dai movimenti che tendono a ripetersi in maniera analoga nei medesimi periodi di anni successivi. Tuttavia, di queste tre componenti si prenderanno in considerazione solo il trend e la stagionalità.

Per ulteriori approfondimenti si rimanda a Di Fonzo e Lisi (2005).

## 1.2 Il lisciamiento esponenziale

Il metodo del lisciamiento esponenziale è ancora oggi fortemente utilizzato per la sua semplicità e flessibilità d'uso. Il metodo venne ideato nel 1959 da Brown ed ampliato da Holt (1957) e Winters (1960). Tale metodologia parte dall'assunto che per effettuare una previsione le osservazioni recenti sono più rilevanti rispetto a quelle passate, ovvero hanno un peso maggiore. Ciò significa che la previsione della serie storica  $h$  passi in avanti,  $\hat{y}_{t+h|t}$ , viene calcolata facendo una media pesata delle osservazioni passate ( $y_1, y_2, \dots, y_t$ ) con pesi che decrescono esponenzialmente all'aumentare della distanza temporale dell'osservazione. Tale metodo è di natura non parametrica, dunque non è possibile calcolare intervalli di previsione.

### 1.2.1 Metodi di lisciamiento esponenziale

Sull'assunto che la serie consti di più componenti, dette componente di livello, componente tendenziale e componente stagionale, il lisciamiento esponenziale è di fatto un algoritmo che effettua una previsione puntuale sulla base di un'equazione di previsione e di un'equazione di lisciamiento per ciascuna delle componenti della serie storica osservata.

Esistono varie forme specifiche di lisciamiento esponenziale, ciascuna delle quali permette di effettuare previsioni su serie storiche con caratteristiche diverse. Tali caratteristiche dipendono dal numero di componenti che formano la serie stessa e come queste entrano nella serie, ovvero se entrano in forma additiva o moltiplicativa.

Ad esempio, il metodo di lisciamiento esponenziale semplice viene utilizzato per modellare serie temporali con assenza di trend e stagionalità (cioè con sola componente di livello); il metodo di Holt o lisciamen-

to esponenziale doppio permette di prevedere dati che presentano una componente di livello e una di trend; il metodo di Holt-Winters estende il precedente, andando a considerare serie storiche aventi tutte e tre le componenti (Hyndman e Athanasopoulos, 2018).

Le possibili combinazioni di queste componenti (Tabella 1.1) hanno portato alla creazione di quindici metodi di lisciamiento esponenziale.

<b>Trend</b>	<b>Stagionalità</b>		
	<i>Nessuna</i>	<i>Additiva</i>	<i>Moltiplicativa</i>
<i>Nessuno</i>	$N, N$	$N, A$	$N, M$
<i>Additivo</i>	$A, N$	$A, A$	$A, M$
<i>Additivo smorzato</i>	$A_d, N$	$A_d, A$	$A_d, M$
<i>Moltiplicativo</i>	$M, N$	$M, A$	$M, M$
<i>Moltiplicativo smorzato</i>	$M_d, N$	$M_d, A$	$M_d, M$

Tabella 1.1: Possibili combinazioni di trend e stagionalità per i metodi di lisciamiento esponenziale.

## 1.2.2 Modelli di lisciamiento esponenziale

Sulla base dei metodi sopra citati sono stati definiti i modelli di lisciamiento esponenziale (*ETS models*) utili per ottenere delle previsioni puntuali, ma che permettono, in aggiunta, la costruzione di intervalli di previsione.

Ogni modello consta di un'equazione di misura, che descrive i dati osservati, e di alcune equazioni di transizione che spiegano come le componenti di trend, livello e stagionalità, dette anche stati, cambiano nel tempo (Hyndman e Athanasopoulos, 2018). In totale si hanno trenta modelli di lisciamiento esponenziale, quindici con errore di tipo additivo (Tabella 1.2) e quindici con errore di tipo moltiplicativo (Tabella 1.3).



Trend	Stagionalità		
	$N$	$A$	$M$
$N$	$y_t = l_{t-1} + \epsilon_t$ $l_t = l_{t-1} + \alpha\epsilon_t$	$y_t = l_{t-1} + s_{t-m} + \epsilon_t$ $l_t = l_{t-1} + \alpha\epsilon_t$ $s_t = s_{t-m} + \gamma\epsilon_t$	$y_t = l_{t-1}s_{t-m} + \epsilon_t$ $l_t = l_{t-1} + \alpha\frac{\epsilon_t}{s_{t-m}}$ $s_t = s_{t-m} + \gamma\frac{\epsilon_t}{l_{t-1}}$
$A$	$y_t = l_{t-1} + b_{t-1} + \epsilon_t$ $l_t = l_{t-1} + b_{t-1} + \alpha\epsilon_t$ $b_t = b_{t-1} + \beta\epsilon_t$	$y_t = l_{t-1} + b_{t-1} + s_{t-m} + \epsilon_t$ $l_t = l_{t-1} + b_{t-1} + \alpha\epsilon_t$ $b_t = b_{t-1} + \beta\epsilon_t$ $s_t = s_{t-m} + \gamma\epsilon_t$	$y_t = (l_{t-1} + b_{t-1})s_{t-m} + \epsilon_t$ $l_t = l_{t-1} + b_{t-1} + \alpha\frac{\epsilon_t}{s_{t-m}}$ $b_t = b_{t-1} + \beta\frac{\epsilon_t}{s_{t-m}}$ $s_t = s_{t-m} + \gamma\frac{\epsilon_t}{(l_{t-1} + b_{t-1})}$
$A_d$	$y_t = l_{t-1} + \phi b_{t-1} + \epsilon_t$ $l_t = l_{t-1} + \phi b_{t-1} + \alpha\epsilon_t$ $b_t = b_{t-1} + \beta\epsilon_t$	$y_t = l_{t-1} + \phi b_{t-1} + s_{t-m} + \epsilon_t$ $l_t = l_{t-1} + \phi b_{t-1} + \alpha\epsilon_t$ $b_t = \phi b_{t-1} + \beta\epsilon_t$ $s_t = s_{t-m} + \gamma\epsilon_t$	$y_t = (l_{t-1} + \phi b_{t-1})s_{t-m} + \epsilon_t$ $l_t = l_{t-1} + \phi b_{t-1} + \alpha\frac{\epsilon_t}{s_{t-m}}$ $b_t = \phi b_{t-1} + \beta\frac{\epsilon_t}{s_{t-m}}$ $s_t = s_{t-m} + \gamma\frac{\epsilon_t}{(l_{t-1} + \phi b_{t-1})}$
$M$	$y_t = l_{t-1}b_{t-1} + \epsilon_t$ $l_t = l_{t-1}b_{t-1} + \alpha\epsilon_t$ $b_t = b_{t-1} + \beta\frac{\epsilon_t}{l_{t-1}}$	$y_t = l_{t-1}b_{t-1} + s_{t-m} + \epsilon_t$ $l_t = l_{t-1}b_{t-1} + \alpha\epsilon_t$ $b_t = b_{t-1} + \beta\epsilon_t$ $s_t = s_{t-m} + \gamma\epsilon_t$	$y_t = l_{t-1}b_{t-1}s_{t-m} + \epsilon_t$ $l_t = l_{t-1}b_{t-1} + \alpha\frac{\epsilon_t}{s_{t-m}}$ $b_t = b_{t-1} + \beta\frac{\epsilon_t}{s_{t-m}l_{t-1}}$ $s_t = s_{t-m} + \gamma\frac{\epsilon_t}{(l_{t-1} + b_{t-1})}$
$M_d$	$y_t = l_{t-1}b_{t-1}^\phi + \epsilon_t$ $l_t = l_{t-1}b_{t-1}^\phi + \alpha\epsilon_t$ $b_t = b_{t-1}^\phi + \beta\frac{\epsilon_t}{l_{t-1}}$	$y_t = l_{t-1}b_{t-1}^\phi + s_{t-m} + \epsilon_t$ $l_t = l_{t-1}b_{t-1}^\phi + \alpha\epsilon_t$ $b_t = b_{t-1}^\phi + \beta\epsilon_t$ $s_t = s_{t-m} + \gamma\epsilon_t$	$y_t = l_{t-1}b_{t-1}^\phi s_{t-m} + \epsilon_t$ $l_t = l_{t-1}b_{t-1}^\phi + \alpha\frac{\epsilon_t}{s_{t-m}}$ $b_t = b_{t-1}^\phi + \beta\frac{\epsilon_t}{s_{t-m}l_{t-1}}$ $s_t = s_{t-m} + \gamma\frac{\epsilon_t}{(l_{t-1} + b_{t-1}^\phi)}$

Tabella 1.2: Equazioni di misura e di transizione per le possibili combinazioni di trend e stagionalità dei modelli *ETS* con errore additivo.

Trend	Stagionalità		
	$N$	$A$	$M$
$N$	$y_t = l_{t-1} + \epsilon_t$ $l_t = l_{t-1} + \alpha\epsilon_t$	$y_t = l_{t-1} + s_{t-m} + \epsilon_t$ $l_t = l_{t-1} + \alpha\epsilon_t$ $s_t = s_{t-m} + \gamma\epsilon_t$	$y_t = l_{t-1}s_{t-m} + \epsilon_t$ $l_t = l_{t-1} + \alpha\frac{\epsilon_t}{s_{t-m}}$ $s_t = s_{t-m} + \gamma\frac{\epsilon_t}{l_{t-1}}$
$A$	$y_t = l_{t-1} + b_{t-1} + \epsilon_t$ $l_t = l_{t-1} + b_{t-1} + \alpha\epsilon_t$ $b_t = b_{t-1} + \beta\epsilon_t$	$y_t = l_{t-1} + b_{t-1} + s_{t-m} + \epsilon_t$ $l_t = l_{t-1} + b_{t-1} + \alpha\epsilon_t$ $b_t = b_{t-1} + \beta\epsilon_t$ $s_t = s_{t-m} + \gamma\epsilon_t$	$y_t = (l_{t-1} + b_{t-1})s_{t-m} + \epsilon_t$ $l_t = l_{t-1} + b_{t-1} + \alpha\frac{\epsilon_t}{s_{t-m}}$ $b_t = b_{t-1} + \beta\frac{\epsilon_t}{s_{t-m}}$ $s_t = s_{t-m} + \gamma\frac{\epsilon_t}{(l_{t-1} + b_{t-1})}$
$A_d$	$y_t = l_{t-1} + \phi b_{t-1} + \epsilon_t$ $l_t = l_{t-1} + \phi b_{t-1} + \alpha\epsilon_t$ $b_t = b_{t-1} + \beta\epsilon_t$	$y_t = l_{t-1} + \phi b_{t-1} + s_{t-m} + \epsilon_t$ $l_t = l_{t-1} + \phi b_{t-1} + \alpha\epsilon_t$ $b_t = \phi b_{t-1} + \beta\epsilon_t$ $s_t = s_{t-m} + \gamma\epsilon_t$	$y_t = (l_{t-1} + \phi b_{t-1})s_{t-m} + \epsilon_t$ $l_t = l_{t-1} + \phi b_{t-1} + \alpha\frac{\epsilon_t}{s_{t-m}}$ $b_t = \phi b_{t-1} + \beta\frac{\epsilon_t}{s_{t-m}}$ $s_t = s_{t-m} + \gamma\frac{\epsilon_t}{(l_{t-1} + \phi b_{t-1})}$
$M$	$y_t = l_{t-1}b_{t-1} + \epsilon_t$ $l_t = l_{t-1}b_{t-1} + \alpha\epsilon_t$ $b_t = b_{t-1} + \beta\frac{\epsilon_t}{l_{t-1}}$	$y_t = l_{t-1}b_{t-1} + s_{t-m} + \epsilon_t$ $l_t = l_{t-1}b_{t-1} + \alpha\epsilon_t$ $b_t = b_{t-1} + \beta\epsilon_t$ $s_t = s_{t-m} + \gamma\epsilon_t$	$y_t = l_{t-1}b_{t-1}s_{t-m} + \epsilon_t$ $l_t = l_{t-1}b_{t-1} + \alpha\frac{\epsilon_t}{s_{t-m}}$ $b_t = b_{t-1} + \beta\frac{\epsilon_t}{s_{t-m}l_{t-1}}$ $s_t = s_{t-m} + \gamma\frac{\epsilon_t}{(l_{t-1} + b_{t-1})}$
$M_d$	$y_t = l_{t-1}b_{t-1}^\phi + \epsilon_t$ $l_t = l_{t-1}b_{t-1}^\phi + \alpha\epsilon_t$ $b_t = b_{t-1}^\phi + \beta\frac{\epsilon_t}{l_{t-1}}$	$y_t = l_{t-1}b_{t-1}^\phi + s_{t-m} + \epsilon_t$ $l_t = l_{t-1}b_{t-1}^\phi + \alpha\epsilon_t$ $b_t = b_{t-1}^\phi + \beta\epsilon_t$ $s_t = s_{t-m} + \gamma\epsilon_t$	$y_t = l_{t-1}b_{t-1}^\phi s_{t-m} + \epsilon_t$ $l_t = l_{t-1}b_{t-1}^\phi + \alpha\frac{\epsilon_t}{s_{t-m}}$ $b_t = b_{t-1}^\phi + \beta\frac{\epsilon_t}{s_{t-m}l_{t-1}}$ $s_t = s_{t-m} + \gamma\frac{\epsilon_t}{(l_{t-1} + b_{t-1}^\phi)}$

Tabella 1.3: Equazioni di misura e di transizione per le possibili combinazioni di trend e stagionalità dei modelli *ETS* con errore moltiplicativo.

Per ciascun metodo di lisciamento esponenziale corrispondono due mo-

delli: uno con errore additivo ed uno con errore moltiplicativo. Fissati i valori dei parametri di lisciamiento, i due modelli forniranno le stesse previsioni puntuali. Tuttavia, daranno origine ad intervalli di previsione diversi (Hyndman e Athanasopoulos, 2018).

La procedura di previsione basata sul lisciamiento esponenziale opera nel seguente modo: a partire dalla serie storica osservata, vengono adattati tutti i possibili modelli di lisciamiento esponenziale. Per ciascuno di questi modelli viene valutata la bontà di adattamento tramite un opportuno criterio, scegliendo così il modello che si adatta meglio ai dati. A questo punto viene calcolata la previsione  $h$  passi in avanti sulla base dell'equazione di previsione specificata dal modello.

### 1.3 Procedura bagging

La corretta identificazione delle componenti di trend e stagionalità, necessaria per poter separare la parte deterministica da quella stocastica, è il punto cardine su cui si basa la metodologia *bagging*, il cui obiettivo è quello di prevedere i valori futuri al netto delle variazioni accidentali, evitando così di riprodurre le fluttuazioni casuali osservate nella serie storica.

A differenza del metodo basato sul lisciamiento esponenziale, che stima direttamente un modello, questa metodologia segue un approccio totalmente diverso: dopo aver effettuato una trasformazione di Box-Cox alla serie osservata, quest'ultima viene suddivisa nelle componenti di trend, stagionalità e residuale attraverso una decomposizione STL (*Seasonal and Trend Decomposition using Loess*). Successivamente ci si focalizza sulla serie dei residui, che in assenza di componenti stocastiche risultano essere incorrelati, e su di essa si applica un metodo *bootstrap* per generare un numero elevato,  $B$ , di serie residuali. Ciascuna di queste serie viene sommata alle componenti stagionale e tendenziale trovate al passo precedente e viene effettuato l'inverso della trasformazione di Box-Cox: si ottiene così una nuova serie pressoché simile a quella originaria, chiamata serie *bootstrap*. Per ciascuna serie *bootstrap* viene adattato un modello di lisciamiento esponenziale e, sulla base di questo modello, vengono calcolate le previsioni puntuali  $h$  passi in avanti. La previsione finale  $\hat{y}_{t+h|t}$  sarà data dalla mediana delle  $B$  previsioni puntuali ottenute dalle serie *bootstrap*.

Se la serie non presenta stagionalità la procedura rimane la stessa, cam-

bia solo il metodo di decomposizione adattato, detto metodo *loess*, che di fatto non va a considerare la componente stagionale, e che divide la serie nelle sole componenti tendenziale e residuale.

### 1.3.1 Trasformazione di Box-Cox

Al fine di ridurre la variabilità dei dati in una serie storica, possono essere effettuate delle trasformazioni della serie. In questo modo si hanno dei pattern più semplici che rendono le previsioni più accurate.

La trasformazione di Box-Cox (Box e Cox 1964) è una classe di trasformazioni parametriche dipendenti da un parametro  $\lambda$ , definita come segue:

$$\omega_t = \begin{cases} \log(y_t) & \lambda = 0 \\ \frac{(y_t^\lambda - 1)}{\lambda} & \lambda \neq 0 \end{cases} .$$

In quest'analisi il valore assunto dal parametro  $\lambda$  verrà scelto utilizzando il metodo proposto da Guerrero (1993) che fornisce un valore  $\lambda$  compreso nell'intervallo  $[0, 1]$ . Il metodo opera suddividendo la serie in tante sottoserie di lunghezza pari alla stagionalità, o di lunghezza due se la serie non è stagionale. Per ciascuna sottoserie, viene calcolata la media campionaria  $\mu$  e la deviazione standard  $\sigma$ ; il valore ottimale  $\hat{\lambda}$  viene scelto in maniera tale che sia minimo il coefficiente di variazione  $\frac{\sigma}{\mu^{(1-\lambda)}}$  tra le sottoserie.

### 1.3.2 Decomposizione STL

Per andare a scomporre la serie nelle varie componenti vengono adoperati due metodi: il metodo *loess* (Cleveland, 1992) utilizzato per decomporre serie che non presentano stagionalità ed il metodo STL per decomporre serie stagionali (Cleveland *et al.*, 1990).

Il metodo *loess* (*local regression*) è un metodo di lisciamiento basato sulla regressione locale. Per prima cosa si sceglie un numero intero positivo  $q$  e per ogni valore osservato  $y_t$  ( $t = 1, \dots, n$ ) vengono presi i  $q$  valori più vicini. Ciascuno di questi valori  $y_1^*, y_2^*, \dots, y_q^*$  viene pesato mediante un peso che decresce all'aumentare della distanza di tale valore da  $y_t$ .

Data la funzione tricubica  $\mathcal{W}$ :

$$\mathcal{W}(u) = \begin{cases} (1 - u^3)^3 & 0 \leq u < 1 \\ 0 & u \geq 1 \end{cases},$$

il peso associato a  $y_i^*$  viene calcolato mediante la formula:

$$v_i(y_t) = \mathcal{W}\left(\frac{|y_i^* - y_t|}{\lambda_q(y_t)}\right) \quad i = 1, \dots, q, \quad (1.2)$$

dove  $\lambda_q(y_t)$  è la distanza del  $q$ -mo punto più lontano da  $y_t$ . Se  $q > n$ , nella formula (1.2) si pone  $\lambda_q(y_t) = \lambda_n(y_t)\frac{q}{n}$ , dove  $\lambda_n(y_t)$  è la distanza del punto più lontano da  $y_t$ .

Il secondo passaggio consiste nello stimare un polinomio  $g(y)$  di grado  $d$ , passante per i valori pesati, dove in genere viene fissato un valore  $d = 1, 2$ , cioè  $g(y)$  è una curva lineare o quadratica.

La stima della componente tendenziale è data da  $\hat{g}(y_t)$ . All'aumentare di  $q$ ,  $\hat{g}(y_t)$  diventa più liscia. Per  $q$  tendente all'infinito  $\hat{g}(y_t)$  coincide col polinomio stimato ai minimi quadrati di grado  $d$ .

Il metodo STL è un'estensione del metodo descritto in precedenza: il lisciatore *loess* viene applicato sequenzialmente al fine di poter scomporre la serie nelle componenti di trend ( $T_t$ ), stagionalità ( $S_t$ ) e residuale ( $R_t$ ).  
Data la serie:

$$Y_t = T_t + S_t + R_t,$$

la procedura consta di due operazioni ricorsive, un ciclo interno annidato in un ciclo esterno, al termine delle quali si otterranno le componenti:  $T_t = T_t^{(k+1)}$ ,  $S_t = S_t^{(k+1)}$ ,  $R_t = R_t^{(k+1)}$ , dove  $k$  corrisponde al numero di iterazioni eseguite nel ciclo interno.

Nella prima operazione ricorsiva vengono aggiornate le componenti di trend e stagionalità, mentre nel ciclo esterno viene eseguito un ciclo interno e vengono corrette le componenti risultanti dai passaggi precedenti.

La prima componente ad essere calcolata nel ciclo interno è quella stagionale: dopo aver detrendizzato la serie ( $Y_t - T_t^k$ ), si considerano le cosiddette sottoserie cicliche, ovvero le serie composte dalle sole osservazioni rilevate in uno specifico periodo della serie (supponendo di avere una serie di periodo  $m$ , si avranno  $m$  sottoserie cicliche). A ciascuna di queste sottoserie viene applicata la procedura *loess* ponendo  $d = 1$ . L'insieme dei valori lisciati di ciascuna sottoserie ciclica determina una nuova serie temporanea,  $C_t^{(k+1)}$ . A questa nuova serie, viene applicata una media mobile di ordine pari al periodo della serie,  $m$ , seguita da

un'ulteriore media mobile di ordine  $m$ . Successivamente viene eseguita un'altra operazione di lisciamiento tramite *loess* che utilizza un polinomio di grado  $d = 1$ , che genera la serie  $L_t^{(k+1)}$ . La componente stagionale è dunque ottenuta facendo:  $S_t^{(k+1)} = C_t^{(k+1)} - L_t^{(k+1)}$ .

Sulla serie destagionalizzata  $Y_t^{d(k+1)} = Y_t - S_t^{(k+1)}$  viene applicato il metodo *loess* ponendo  $d=1$ , ottenendo così la stima della componente tendenziale  $T_t^{(k+1)}$  e terminando il ciclo interno.

A questo punto viene calcolata la componente residuale  $R_t^{(k+1)} = Y_t - S_t^{(k+1)} - L_t^{(k+1)}$  e a ciascun residuo viene attribuito un peso pari a :

$$\rho_t = \mathcal{B}\left(\frac{|R_t|}{6\text{median}(|R_t|)}\right),$$

dove

$$\mathcal{B} = \begin{cases} (1 - u^2)^2 & 0 \leq u < 1 \\ 0 & u \geq 1 \end{cases}.$$

In questo modo i valori anomali avranno peso prossimo allo zero. Questi pesi verranno utilizzati nell'iterazione successiva del ciclo interno per effettuare il lisciamiento delle serie  $C_t^{(k+2)}$  e  $Y_t^{d(k+2)}$ , ottenendo così delle stime di trend e stagionalità più robuste. Al passo  $k = 0$  tutti i pesi sono pari ad 1.

Per poter eseguire la procedura è necessario stabilire a priori il numero di passaggi nel ciclo interno ( $n_i$ ) ed il numero di passaggi nel ciclo esterno ( $n_e$ ). Se da un'analisi esplorativa dei dati emergono molteplici valori anomali o un chiaro comportamento non gaussiano che genera forte variabilità, allora è necessario eseguire almeno un ciclo esterno; in caso contrario è possibile eseguire solo il ciclo interno.

Il numero di iterazioni  $n_i$  deve essere tale per cui l'aggiornamento delle componenti di trend e stagionalità converga. Tuttavia, la convergenza viene raggiunta velocemente dall'algoritmo e la letteratura conferma che un valore di  $n_i = 2$  è più che sufficiente.

Qualora si avesse un comportamento dei dati fortemente variabile è necessario individuare il numero di iterazioni esterne necessarie affinché le stime di trend e stagionalità possano considerarsi robuste. Una regola generale suggerisce che se  $n_i = 1$ , un valore  $n_e = 10$  garantisce quasi certamente la convergenza e la robustezza delle stime (in alcuni casi anche un valore  $n_e = 5$  può andare bene) (Cleveland *et al.*, 1990).

La procedura STL risulta essere particolarmente veloce dal punto di vista computazionale, permettendo così di trattare anche serie storiche

particolarmente lunghe. In aggiunta, è possibile trattare serie con valori mancanti, dal momento che la procedura di lisciamento *loess* genera un polinomio  $\hat{g}(y)$  definito non solamente per i valori osservati  $y_t$ .

### 1.3.3 Bootstrap dei residui e calcolo delle previsioni

Il principale vantaggio della decomposizione STL risiede nel fatto che la stima delle componenti di trend e stagionalità che viene eseguita nel ciclo interno, è completamente indipendente dalla componente residuale. La possibilità di trattare le componenti separatamente permette alla procedura *bagging* di andare a lavorare direttamente sulla serie dei residui, più facile da simulare, essendo costituita da valori che per assunzione sono incorrelati. Dopo aver simulato le serie residuali tramite delle tecniche *bootstrap* (Capitolo 2), è possibile sommarle alle restanti componenti, ricomponendo così il fenomeno nella sua interezza e, soprattutto, non alterandone le caratteristiche.

Al fine di poter ottenere delle previsioni su scala originaria è necessario effettuare l'inverso della trasformazione di Box-Cox.

Tale trasformazione è data da:

$$y_t = \begin{cases} \exp(\omega_t) & \lambda = 0 \\ (\lambda\omega_t + 1)^{\frac{1}{\lambda}} & \lambda \neq 0 \end{cases} \quad (1.3)$$

A questo punto è possibile ultimare la procedura: avendo a disposizione le serie *bootstrap* nella stessa scala di quella originaria, si adattano tutti i modelli ETS per ciascuna di queste, scegliendo il modello migliore sulla base del criterio d'Akaike (Akaike, 1974) e calcolando le previsioni puntuali per ogni orizzonte temporale. La previsione puntuale finale sarà data dalla mediana delle previsioni puntuali ottenute dalle serie *bootstrap*.

In quest'analisi è stata scelta la mediana come strumento di aggregazione poiché è robusta ai valori anomali; tuttavia è possibile utilizzare altri indicatori di sintesi quali la media o la media troncata.

## Capitolo 2

# Metodologie bootstrap per serie storiche

### 2.1 Introduzione

Il metodo *bootstrap* venne introdotto per la prima volta nel 1979 da Bradley Efron con l'obiettivo di fornire una misura dell'accuratezza di una statistica di interesse mediante un approccio non parametrico.

Tale tecnica nasce come evoluzione del metodo *jackknife* (Tukey, 1958), che aveva l'obiettivo di stimare la varianza di uno stimatore attraverso una procedura di ricampionamento senza reinserimento che generava nuovi campioni, detti campioni *jackknife*, costruiti escludendo un'unica osservazione dal campione originario.

Entrambi i metodi sono detti *computer intensive* poiché ripetono delle operazioni un numero elevato di volte e in maniera del tutto automatica. Queste operazioni richiedono, nella maggior parte dei casi, una laboriosa implementazione.

L'utilizzo dei metodi non parametrici ha permesso di poter risolvere molti problemi. Infatti, quest'ultimi possono essere sempre utilizzati quando non è possibile verificare (o non sono valide) le assunzioni sottostanti i modelli parametrici e permettono sia di valutare diverse proprietà di uno stimatore, come la correttezza, ma anche di ricavare altre caratteristiche, come ad esempio la varianza, gli intervalli di confidenza e gli errori di previsione. Malgrado ciò, questi metodi forniscono dei risultati in forma approssimata poiché si basano su simulazioni.

Ad oggi questa procedura viene utilizzata per ulteriori scopi, tra cui problemi di verifica di ipotesi e la stima delle distribuzioni delle statistiche test (Efron e Tibshirani, 1993).

## 2.2 Bootstrap per dati indipendenti

Sia  $\mathbf{x} = (x_1, x_2, \dots, x_n)$  un campione di dati indipendenti ed identicamente distribuiti (*i.i.d*) estratto da una popolazione con una distribuzione  $P$  ignota. Si vuole fare inferenza sul parametro ignoto  $\theta = t(P)$ , stimato mediante lo stimatore  $\hat{\theta} = s(\mathbf{x})$ , dove  $s(\cdot)$  è una statistica, ovvero una funzione dei dati (Efron, 1992).

Sulla base del campione osservato viene stimata la distribuzione della popolazione  $\hat{P}$  tramite la funzione di ripartizione empirica  $\widehat{P}(x) = \sum_{k=1}^n I(x_i \leq x)$ . Completato questo passaggio, la procedura consiste nell'estrarre casualmente da  $\hat{P}$ , un numero elevato,  $B$ , di campioni della stessa numerosità  $n$ , ottenendo così i campioni *bootstrap*  $\mathbf{x}_b^* = (x_1^*, x_2^*, \dots, x_n^*)$  con  $b = 1, \dots, B$ . Quest'operazione equivale a compiere un'estrazione con reinserimento dal campione dei dati osservati, dove ogni osservazione ha la stessa probabilità di essere estratta pari a  $\frac{1}{n}$ .

Su ciascun campione *bootstrap* viene calcolato il parametro di interesse  $\hat{\theta}_b^*$ , che rappresenta la stima *bootstrap*. L'insieme dei valori  $\hat{\theta}_b^*$  costituisce la distribuzione *bootstrap* dello stimatore  $\hat{\theta}$ , ovvero un'approssimazione della reale distribuzione campionaria del parametro di interesse, che può essere adoperata ai fini già descritti.

La procedura *bootstrap* identifica due mondi: un mondo reale ed un mondo *bootstrap*. Se nel mondo reale, ovvero il mondo che ha generato il campione osservato, tutto è ignoto, nel mondo *bootstrap* tutto è noto a priori, sia la popolazione che il parametro, ed è proprio la conoscenza dei veri valori dei parametri che permette di poter valutare le caratteristiche dello stimatore.

Il passaggio dal mondo reale al mondo *bootstrap* avviene tramite il principio del *plug-in*, ovvero utilizzando  $\hat{P}$  e  $\hat{\theta}$ , come veri valori del mondo *bootstrap* (Tabella 2.1). L'unico aspetto che accomuna i due mondi è il comportamento dello stimatore, si dice dunque che i due mondi hanno lo stesso comportamento relativo: se lo stimatore *bootstrap*  $\hat{\theta}_b^*$  è corretto, allora lo sarà anche  $\hat{\theta}$ .



<b>Mondo Reale</b>	<b>Mondo Bootstrap</b>
$X \sim P$ con $\theta = t(P)$	$X^* \sim \widehat{P}$ con $\widehat{\theta} = t(\widehat{P})$
$\Downarrow$	$\Downarrow$
$x_1, \dots, x_n \sim i.i.d.(P)$	$x_1^*, \dots, x_n^* \sim i.i.d.(\widehat{P})$
$\Downarrow$	$\Downarrow$
$\widehat{\theta}$ stima $\theta$	$\widehat{\theta}^*$ stima $\widehat{\theta}$

Tabella 2.1: Principio del *plug-in*.

Estensioni del *bootstrap* non parametrico sono date dal *bootstrap* semi-parametrico e parametrico dove si assume che i dati provengano da una specifica distribuzione o che esista una particolare relazione tra essi, esprimibile mediante un opportuno modello. Se la distribuzione o il modello sono completamente specificati, allora si parla di *bootstrap* parametrico, se invece lo sono parzialmente si parla di *bootstrap* semiparametrico. Di conseguenza, se è nota la distribuzione da cui provengono i dati non è più necessario ottenere i campioni *bootstrap* dalla funzione di ripartizione empirica, ma basterà generare dei campioni dalla distribuzione teorica.

Nonostante gli innumerevoli vantaggi, questa procedura non può essere applicata nel caso vi sia dipendenza tra dati, come nelle serie storiche, in quanto il campione *bootstrap* simulato con le tecniche appena descritte non è in grado di replicare la struttura di dipendenza tra i dati. Per ovviare al problema sono stati proposti svariati metodi. Nel prosieguo verranno illustrati nel dettaglio solo tre di essi: il *moving block bootstrap*, il *sieve bootstrap* ed il *wild bootstrap*.

Per ulteriori approfondimenti sui metodi e le proprietà del *bootstrap* nel caso di dati indipendenti ed identicamente distribuiti si rimanda a Efron e Tibshirani (1993), Davison e Hinkley (1997). Per approfondimenti in merito ai metodi *bootstrap* per serie storiche si rimanda a Härdle *et al.* (2003) e Horowitz (2001).

## 2.3 Bootstrap per serie storiche

I metodi *bootstrap* utilizzati in questa tesi si dividono in due categorie principali: metodi parametrici e non parametrici. Per i primi è necessario assumere una distribuzione per il processo generatore dei dati e il *wild bootstrap* rientra in questa categoria. Al contrario, i metodi non parametrici non necessitano di alcuna assunzione sul meccanismo gene-

ratore dei dati. In quest'ultima categoria rientrano il *bootstrap* a blocchi ed il *sieve bootstrap*.

### 2.3.1 Bootstrap parametrico

Il *bootstrap* parametrico definito anche come *model-based resampling* è stato sviluppato da Efron e Tibshirani (1993) ed assume che il campionamento venga fatto sui residui di un modello, che risultano essere indipendenti ed identicamente distribuiti se il modello è correttamente specificato.

Affinché si possa eseguire il campionamento è necessario ipotizzare che il processo generatore dei dati (PGD) sia stazionario, invertibile e modellabile mediante un modello *ARIMA*.

Si assuma, ad esempio, che il processo  $\{X_t\}$  possa essere modellato mediante un modello *AR*( $p$ ) stazionario, ovvero presenta una forma del tipo:

$$X_t - \mu = \sum_{i=1}^p \phi_i (X_{t-i} - \mu) + \epsilon_t, \quad t = 1, \dots, n,$$

che equivale a scrivere:

$$\Phi(B)(X_t - \mu) = \epsilon_t, \quad t = 1, \dots, n,$$

dove  $\Phi(B) = (1 - \phi_1 B - \dots - \phi_p B^p)$  è l'operatore polinomiale autoregressivo con  $\phi_i$  ( $i = 1, \dots, p$ ) parametri costanti,  $\mu = E(X_t)$  è la media del processo e  $\epsilon_t$  corrispondono ai residui, assunti *i.i.d.* con media nulla e varianza  $\sigma_\epsilon^2$ . Affinché il processo si possa considerare stazionario, tutte le  $p$  radici dell'equazione caratteristica  $\Phi(B) = 0$  devono essere in modulo maggiori di uno.

Dopo aver identificato l'ordine  $p$  del modello, è possibile procedere con la stima dei parametri ignoti. La media del processo viene stimata mediante la media campionaria:

$$\hat{\mu} = \frac{1}{n} \sum_{t=1}^n X_t.$$

La stima dei parametri autoregressivi  $\hat{\phi}_i$  può essere eseguita mediante il metodo della massima verosimiglianza, attraverso le equazioni di Yule-Walker, oppure con il metodo dei minimi quadrati lineari. A questo punto è possibile ottenere una stima dei valori residuali, data da:

$$\hat{\epsilon}_t = X_t - \hat{\mu} - \sum_{i=1}^p \hat{\phi}_i (X_{t-i} - \hat{\mu}) \quad t = 1 + p, \dots, n.$$

Le innovazioni *bootstrap*  $\epsilon_t^*$  vengono così ottenute a partire dai residui centrati (Lahiri, 2003), ricavati attraverso la seguente formula:

$$\tilde{\epsilon}_t = \hat{\epsilon}_t - \frac{1}{n-p} \sum_{t=1+p}^n \hat{\epsilon}_t.$$

I valori  $\epsilon_t^*$  vengono estratti esattamente come nel *bootstrap* per dati indipendenti, ovvero dalla funzione di ripartizione empirica di  $\tilde{\epsilon}_t$ .

La serie *bootstrap*  $X_1^*, \dots, X_n^*$  è quindi ottenuta come:

$$X_t^* - \hat{\mu} = \sum_{i=1}^p \hat{\phi}_i(X_{t-i}^* - \hat{\mu}) + \epsilon_t^*, \quad t = p+1, \dots, n. \quad (2.1)$$

Si osservi come la formula (2.1) non permetta di stimare i primi  $p$  valori  $X_1^*, \dots, X_p^*$  (non è possibile ottenere i valori  $X_{1-p}^*, \dots, X_0^*$ ).

Per sopperire al problema, la letteratura propone due soluzioni. La prima suggerisce di stimare la serie *bootstrap* utilizzando la formula (2.1), ponendo i primi  $p$  valori uguali ai dati osservati. In alternativa, si può assumere  $X_i^* = \hat{\mu}$  per  $i \leq -p$ , applicare la formula (2.1) per  $t = -p, \dots, n$  e conservare gli ultimi  $n$  valori (Berkowitz e Kilian, 2000).

A questo punto, dato lo stimatore di interesse  $\hat{\theta} = s(X_1, \dots, X_n)$  è possibile costruire la versione *bootstrap* mediante il principio del *plug-in*, ottenendo:

$$\hat{\theta}^* = s(X_1^*, \dots, X_n^*).$$

Si sottolinea, inoltre, che nel caso in cui sia possibile assumere che  $\tilde{\epsilon}_t \sim N(0, \sigma_{\tilde{\epsilon}}^2)$ , le innovazioni *bootstrap* avranno distribuzione  $\epsilon_t^* \sim N(0, \hat{\sigma}_{\tilde{\epsilon}}^2)$  e potranno essere generate a partire dalla stessa.

La procedura può essere estesa anche al caso più generico in cui si assume un modello *ARMA*( $p, q$ ).

Per ulteriori approfondimenti si rimanda a Delaney e Chatterjee (1986).

## Wild bootstrap

Il *wild bootstrap* è stato sviluppato da Wu (1986) e Liu (1988) e viene utilizzato nel caso in cui i residui presentano eteroschedasticità, ovvero quando il modello specificato per il PDG non è corretto. In una situazione di questo tipo l'eteroschedasticità non ha forma nota e non è possibile replicarla nel mondo *bootstrap*. Per ovviare al problema, i due autori hanno ideato questo nuovo metodo che di fatto segue tutti i passaggi del metodo *model-based resampling* con la differenza che i valori residuali stimati  $\hat{\epsilon}_t$  sono tali per cui  $E(\hat{\epsilon}_t) = 0$  e  $Var(\hat{\epsilon}_t) = \sigma_t^2$ . Le innovazioni *bootstrap* vengono generate nel seguente modo:

$$\epsilon_t^* = \hat{\epsilon}_t \cdot \zeta_t^* \quad t = 1, \dots, n,$$

dove  $\zeta_t^*$  è una variabile casuale con media zero e varianza unitaria ed  $\hat{\epsilon}_t$  è un valore estratto casualmente dagli  $n - p$  residui campionati.

Sono state proposte diverse distribuzioni per la variabile casuale  $\zeta_t^*$ , sia discrete che continue. In particolare, le distribuzioni maggiormente utilizzate sono la distribuzione Normale Standard, la distribuzione di Rademacher, definita come:

$$\zeta_t^* = \begin{cases} 1, & \text{con probabilità } 1/2 \\ -1, & \text{con probabilità } 1/2 \end{cases},$$

oppure la distribuzione di Mammen:

$$\zeta_t^* = \begin{cases} \frac{-\sqrt{5}-1}{2}, & \text{con probabilità } p = (\sqrt{5} + 1)/2\sqrt{5} \\ \frac{\sqrt{5}+1}{2}, & \text{con probabilità } 1 - p \end{cases}.$$

Quest'ultima distribuzione è caratterizzata dal momento terzo pari ad uno, che garantisce una migliore performance del metodo, come dimostra Liu (1988).

### 2.3.2 Bootstrap non parametrico

Quando non si hanno conoscenze a priori sulla natura del processo, i metodi parametrici non possono essere utilizzati.

Carlstein (1986) e Hall *et al.* (1995) introducono per la prima volta un nuovo schema di campionamento, oggi noto come *non-overlapping block bootstrap* (NBB), applicabile a dati dipendenti senza alcun'assunzione parametrica. Anziché campionare con reinserimento una singola osservazione per volta (come avviene per il *bootstrap* per dati *i.i.d.*), il NBB campiona blocchi non sovrapposti di osservazioni consecutive per volta, dove tutti i blocchi sono di lunghezza  $l$  prefissata. In questo modo la struttura di dipendenza delle osservazioni originarie è mantenuta all'interno di ciascun blocco. Inoltre, la lunghezza comune dei blocchi aumenta con la dimensione campionaria (Lahiri, 2003).

Dopo questa pubblicazione sono seguiti numerosi studi che hanno portato alla nascita di numerose varianti del NBB che riguardavano la sovrapposizione dei blocchi (Künsch, 1989; Hall *et al.*, 1995; Politis e Romano, 1994) e considerazioni sulla la lunghezza di questi, cioè fissa (Bühlmann, 2002) o variabile (Politis e Romano, 1994). La variante che ad oggi è maggiormente utilizzata è quella del *moving block bootstrap* (MBB).

## Moving block bootstrap

Il *moving block bootstrap* è stato proposto da Künsch (1989) e Liu *et al.* (1992) e si distingue dal NBB perchè permette la sovrapposizione dei blocchi, che hanno lunghezza fissa. La sovrapposizione fa sì che le osservazioni possano presentarsi in più di un blocco.

Sia  $\{X_t\}_{t \in \mathbb{N}}$  un processo stazionario e sia  $\mathbf{X}_n = \{X_1, \dots, X_n\}$  la serie storica osservata. Sia  $l$  un valore intero tale che  $1 \leq l < n$ . Si definiscono i blocchi sovrapposti  $\mathbb{B}_1 \dots \mathbb{B}_N$  di lunghezza  $l$  contenuti in  $\mathbf{X}_n$  come:

$$\begin{aligned}\mathbb{B}_1 &= (X_1, \dots, X_l) \\ \mathbb{B}_2 &= (X_2, \dots, X_l, X_{l+1}) \\ &\dots \quad \dots \\ \mathbb{B}_N &= (X_{n-l+1}, \dots, X_n),\end{aligned}$$

dove  $N = n - l + 1$ .

Per semplicità si supponga che  $n$  sia multiplo di  $l$ . Per generare i campioni MBB, si selezionano con reinserimento  $b$  blocchi dal campione  $(\mathbb{B}_1, \dots, \mathbb{B}_N)$ , con  $b = \frac{n}{l}$ .

I blocchi campionati genereranno  $b \cdot l = n$  osservazioni *bootstrap*  $X_1^*, \dots, X_n^*$ . A questo punto se la variabile di interesse è nella forma  $T_n = t_n(\mathbf{X}_n; \theta(P_n))$ , con  $P_n$  distribuzione di probabilità congiunta di  $\mathbf{X}_n$ , è possibile ottenere la versione MBB definita come

$$T_n^* = t_n(X_1^*, \dots, X_n^*; \theta(\hat{P}_n)),$$

dove  $\hat{P}_n$  è la distribuzione di probabilità condizionata di  $X_1^*, \dots, X_n^*$  dato  $\mathbf{X}_n$ . Le figure 2.1 e 2.2 riportano graficamente la differenza tra il MBB ed il NBB nella selezione dei blocchi.

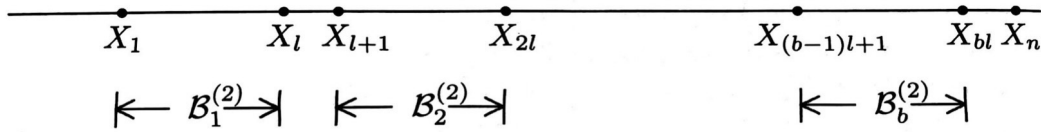


Figura 2.1: L'insieme dei blocchi non sovrapposti  $\{B_1, \dots, B_n\}$  nel NBB.

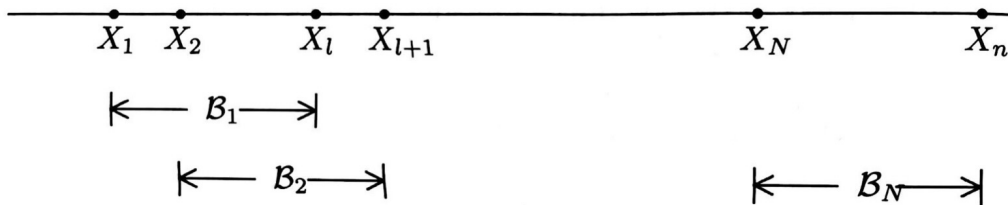


Figura 2.2: L'insieme dei blocchi sovrapposti  $\{B_1, \dots, B_n\}$  nel MBB.

Nel caso in cui  $l = 1$  il MBB si riduce al metodo di Efron (1992). Se  $n$  non è multiplo di  $l$  la letteratura suggerisce di campionare  $b = b_0$  blocchi, con  $b_0 = \min \{k \geq 1 : kl \geq n\}$  e conservare i primi  $n$  valori campionati per definire le replicazioni *bootstrap* di  $T_n$  (Lahiri, 2003). La lunghezza del blocco, il numero di valori osservati e la struttura di dipendenza del processo sono degli aspetti cruciali, che influenzano la performance del MBB. In particolare, se i blocchi sono troppo corti i campioni *bootstrap* non captano correttamente la struttura di dipendenza, che si interrompe ogni qualvolta termina un blocco ed inizia il successivo. Allo stesso tempo, si richiede che i blocchi non siano eccessivamente lunghi perché questo comporterebbe la presenza di un minor numero di blocchi con dei campioni *bootstrap* che tenderanno ad essere molto simili (Berkowitz e Kilian, 2000). Per poter bilanciare questo *trade-off*, una numerosità campionaria  $n$  molto grande aiuta, poiché permette di trovare con più facilità un valore  $l$  opportuno. Operativamente, si richiede che  $l$  sia tale per cui

$$l^{-1} + n^{-1} = o(1) \quad \text{per } n \rightarrow \infty. \quad (2.2)$$

Alcune scelte per  $l$  sono  $l = Cn^{\frac{1}{k}}$ , con  $k = 3, 4$  e  $C \in \mathbb{R}$  costante.

Un ulteriore svantaggio associato a questo schema di campionamento risiede nel fatto che il MBB assegna peso minore alle osservazioni all'inizio e alla fine del campione osservato  $\mathbf{X}_n$ . Infatti, per  $l \leq v \leq n - l$  la  $v$ -ma osservazione  $X_v$  compare esattamente in  $l$  blocchi, mentre per  $1 \leq v \leq l - 1$ ,  $X_v$  e  $X_{n-v+1}$  appaiono solo in  $v$  blocchi. Poiché non esiste alcuna osservazione prima di  $X_1$  e dopo  $X_n$  non è possibile definire nuovi blocchi per eliminare tale effetto. Politis e Romano (1991) hanno suggerito un nuovo metodo che suppone che i dati siano disposti attorno ad un cerchio ed utilizzano queste "osservazioni circolari" per creare i blocchi. Tale tecnica è nota come *circular block bootstrap* (CBB).

Per ovviare al problema relativo alla lunghezza dei blocchi Politis e Romano (1994) hanno ideato il cosiddetto *stationary block bootstrap* in cui le lunghezze dei blocchi sono casuali e realizzazione di una distribuzione geometrica con valore atteso  $l$  e che soddisfano (2.2).

Infine, altre due varianti particolarmente utilizzate sono quelle del *matched block bootstrap* (MaBB) e *tapered block bootstrap* (TBB). Il primo usa un meccanismo stocastico per risolvere il problema di interruzione della dipendenza nel passare da un blocco al successivo, mentre il TBB fa convergere i valori iniziali dei blocchi verso un valore comune, come la media campionaria, per ridurre la medesima discontinuità.

È stato dimostrato che i due metodi forniscono degli stimatori più accurati, sia in termini di distorsione che di varianza, rispetto a quelli forniti dal MBB. A causa della loro complessità, MaBB e TBB sono stati definiti dei metodi *bootstrap* di seconda generazione e non verranno trattati in questa tesi. Per ulteriori approfondimenti riguardo a questi metodi e ad altri si rimanda a Lahiri (2003); Carlstein *et al.* (1998); Paparoditis e Politis (2001).

### 2.3.3 Sieve bootstrap

La procedura sottostante il *sieve bootstrap* fu proposta da Bühlmann (1997) e combina, in un certo senso, le caratteristiche del *bootstrap* parametrico e non parametrico.

Tale approccio si basa sul metodo dei setacci (*method of sieves*) descritto da Grenander (1981) che viene utilizzato nell'ambito della statistica non parametrica per trovare degli stimatori di parametri infinito dimensionali. In una situazione di questo tipo, il metodo della massima verosimiglianza o il metodo dei minimi quadrati, non forniscono stimatori accurati. Pertanto Grenander suggerisce di effettuare l'ottimizzazione (massimizzazione della verosimiglianza, o minimizzazione della som-

ma dei quadrati degli errori) all'interno di un sottoinsieme dello spazio parametrico, detto setaccio (*sieve*), che aumenta al crescere della dimensione campionaria. Il setaccio ottimale viene in seguito individuato mediante criteri basati sulla bontà di adattamento.

Nell'ambito del *sieve bootstrap* si assume che il processo sottostante  $\{\mathbf{X}_t\}_{t \in \mathbb{Z}}$  sia stazionario, ergodico ed invertibile, dunque modellabile mediante un modello AR di ordine infinito:

$$X_t - \mu = \sum_{i=1}^{\infty} \phi_i (X_{t-i} - \mu) + \epsilon_t, \quad t \in \mathbb{Z},$$

dove  $E(X_t) = \mu$  e  $\sum_{i=1}^{\infty} \phi_i^2 < \infty$ .

Sia  $\mathbf{X}_n = \{X_1, \dots, X_n\}$  la serie storica osservata, realizzazione del processo appena descritto. Sulla base della teoria dei setacci, si approssima il processo mediante un modello autoregressivo di ordine  $p$ , dove  $p = p(n)$  è funzione della dimensione campionaria con  $p(n) \rightarrow \infty$ ,  $p(n) = o(n)$  ( $n \rightarrow \infty$ ).

Scelto l'ordine  $p$  ottimale mediante il criterio di Akaike, la procedura è simile a quella del *bootstrap* parametrico. Si stima  $\mu$  mediante la media campionaria  $\hat{\mu} = \bar{X} = \frac{1}{n} \sum_{t=1}^n X_t$ , mentre la stima dei coefficienti  $\hat{\phi}_1, \dots, \hat{\phi}_p$  può essere ottenuta utilizzando le equazioni di Yule-Walker, il metodo della massima verosimiglianza, oppure il metodo dei minimi quadrati (Brockwell *et al.*, 1987).

È possibile costruire i residui

$$\hat{\epsilon}_t = \sum_{i=0}^{p(n)} \hat{\phi}_i (X_i - \bar{X}), \quad \hat{\phi}_0 = 1, \quad t = p+1, \dots, n.$$

Per generare le innovazioni *bootstrap* si centrano i residui in zero

$$\tilde{\epsilon}_t = \hat{\epsilon}_t - \frac{1}{n-p} \sum_{i=p+1}^n \hat{\epsilon}_i \quad t = 1+p, \dots, n,$$

e si costruisce la funzione di ripartizione empirica

$$\widehat{F}_{\tilde{\epsilon}}(x) = \frac{1}{n-p} \sum_{i=p+1}^n I(\tilde{\epsilon}_i \leq x).$$

Le innovazioni *bootstrap* si ottengono assumendo  $X_t^* = \hat{\mu}$  per  $t = -\tau, \dots, -\tau + p - 1$  con  $\tau$  valore intero grande positivo e campionando  $\epsilon_t^* \text{ i.i.d.} \sim \widehat{F}_{\tilde{\epsilon}}$ , ottenendo così

$$X_t^* - \bar{X} = \sum_{i=0}^p \hat{\phi}_i (X_i^* - \bar{X}) + \epsilon_t^* \quad \hat{\phi}_0 = 1, \quad t = -\tau + p, \dots, 0, \dots, n,$$

conservando solo gli ultimi  $n$  valori  $X_1^*, \dots, X_n^*$ .

Supponendo che  $\hat{\theta}_n = s(X_1, \dots, X_n)$  sia lo stimatore di interesse, il principio del *plug-in* permette di definire lo stimatore *bootstrap*:



$$\hat{\theta}_n^* = s(X_1^*, \dots, X_n^*).$$

Sotto condizione di regolarità del parametro  $p$  e delle variabili  $\{\epsilon_t\}_{t \in \mathbb{Z}}$ , Bühlmann (1997) ha dimostrato la consistenza del metodo *sieve*.

Dai risultati si evince che il metodo fornisce degli stimatori della varianza più accurati rispetto a quelli ottenuti attraverso le procedure MBB e NBB. Tuttavia, la consistenza di tali stimatori è valida per una classe più ristretta di processi rispetto ai metodi basati sui blocchi. Per ulteriori approfondimenti si rimanda a Choi e Hall (2000).

Rispetto al *bootstrap* a blocchi, questa procedura genera un campione stazionario, che non è un sottoinsieme del campione originale. L'aver assunto un processo  $AR(\infty)$  rende il metodo di più generale applicazione rispetto alle procedure *model-based*, e al tempo stesso più accurato rispetto al *block bootstrap* se il processo sottostante ha la suddetta forma. La rappresentazione autoregressiva di ordine infinito permette inoltre di includere tutti i modelli  $ARMA(p, q)$ , con polinomio MA invertibile. Inoltre, tale approccio permette di gestire con maggiore facilità i dati mancanti, sostituendoli con dei valori stimati dal modello, mentre nel *bootstrap* a blocchi è necessario eliminare il blocco o assumere una lunghezza diversa. Al tempo stesso, però, ci sono diversi svantaggi, tra cui la difficoltà nel verificare la linearità o nel testare l'ipotesi di ordine infinito (Bickel e Bühlmann, 1997). Analogamente al problema di scegliere la lunghezza del blocco ottimale per il *block bootstrap*, anche la scelta del valore  $p$  dipende dalla vera natura del processo sottostante, dalla statistica che deve essere esaminata e dallo scopo per cui il *bootstrap* è usato. Ciononostante questo valore è di facile interpretazione e semplice da testare, attraverso procedure grafiche basate sui residui. Al contrario, nel *block bootstrap* non è possibile interpretare la lunghezza del blocco né tanto meno testarla facilmente. Studi empirici hanno dimostrato che la scelta dell'ordine non incide particolarmente sulla performance del *sieve bootstrap*, mentre lo stesso non si può dire per l'altra procedura (Bühlmann, 2002).

Per cercare di captare gli aspetti positivi di entrambi i metodi, Davison e Hinkley (1997) hanno proposto una nuova procedura, chiamata *post-blackening*. L'idea è quella di effettuare un'operazione di "sbiancamento" della serie storica mediante un modello o un'approssimazione basata sulla teoria dei setacci e poi applicare il *bootstrap* a blocchi ai residui che dovrebbero essere meno dipendenti e procedere nella stima della serie *bootstrap* come descritto nei paragrafi precedenti.

# Capitolo 3

## Analisi dei dati e valutazione dei risultati

### 3.1 M3-competition

Al fine di applicare i metodi precedentemente menzionati, sono stati utilizzati i dati relativi alla *M3-competition*. Le *M-competitions*, o competizioni di Makridakis, rappresentano una serie di sfide ideate da Makridakis che dal 1982 hanno l'obiettivo di coinvolgere numerosi esperti nell'ambito delle previsioni delle serie storiche al fine di scoprire nuovi metodi predittivi e confrontarne le performance con quelle dei metodi tradizionali. Le prime due competizioni, denominate *M1-competition* ed *M2-competition*, si sono svolte rispettivamente negli anni 1982 e 1993. In queste due occasioni, sono state analizzate 1001 e 29 serie, da cui sono state tratte quattro precise conclusioni (Makridakis e Hibon, 2000).

1. I metodi statistici particolarmente complessi non necessariamente forniscono previsioni più accurate rispetto ai metodi più semplici.
2. A seconda della misura di accuratezza utilizzata per valutare le previsioni, può cambiare la classifica relativa alle prestazioni dei metodi.
3. La lunghezza dell'orizzonte temporale di previsione influenza l'accuratezza dei metodi.
4. La combinazione di più metodi porta, in media, ad avere un'accuratezza superiore rispetto a quando i metodi vengono considerati singolarmente.

La terza competizione, la *M3-competition*, avvenne nel 2000, con l'intento di replicare ed estendere i risultati emersi dalle competizioni passate. In particolare, si voleva replicare quanto fatto nelle edizioni precedenti utilizzando un numero maggiore di serie, 3003, mentre si voleva estendere quanto già fatto andando a considerare nuovi metodi legati all'area delle reti neurali e sistemi esperti.

Le serie analizzate sono di diverso tipo e di diverso intervallo temporale, per un totale di 645 serie annuali, 756 serie trimestrali, 1428 serie mensili e 174 serie di altro tipo.

Per garantire un numero sufficientemente adeguato di dati per poter applicare un modello di previsione, si è deciso di fissare un numero minimo di osservazioni per ogni tipologia di serie. Nello specifico, è stato fissato un minimo di 14 osservazioni per le serie annuali (che hanno una lunghezza mediana di 19 osservazioni), 16 per i dati trimestrali (che hanno una lunghezza mediana di 44 osservazioni) e 48 per le serie mensili (che hanno una lunghezza mediana di 115 osservazioni). La tabella 3.1 riporta la classificazione delle serie. Tutti i dati sono strettamente positivi.

Tipologia della serie storica	Intervallo temporale				
	Annuale	Trimestrale	Mensile	Altro	Totale
<i>Microeconomia</i>	146	204	474	4	828
<i>Industria</i>	102	83	334	0	519
<i>Macroeconomia</i>	83	336	312	0	731
<i>Finanza</i>	58	76	145	29	308
<i>Demografia</i>	245	57	111	0	413
<i>Altro</i>	11	0	52	141	204
<i>Totale</i>	645	756	1428	174	3003

Tabella 3.1: Classificazione delle 3003 serie utilizzate nella *M3-competition*.

Ai partecipanti è stato chiesto di effettuare le previsioni sei passi in avanti per le serie annuali, otto per quelle trimestrali, diciotto per quelle mensili e otto per le serie della categoria "Altro". Le previsioni che hanno fornito un valore negativo sono state sostituite con lo zero. Esse sono state in seguito confrontate con i valori reali (questi valori non erano stati resi noti agli esperti, quindi non sono stati adoperati per sviluppare i vari modelli).

Per valutare le prestazioni dei metodi sono stati utilizzati cinque indici di accuratezza: l'errore assoluto medio percentuale simmetrico (SMAPE), la media dei ranghi, l'errore assoluto mediano percentuale simmetrico

(*median* APE), l'errore relativo mediano percentuale simmetrico (*median* RAE) e la percentuale di miglioramento.

Complessivamente, sono state esaminate 24 procedure, suddivise in sei categorie: metodi semplici, modelli per il trend, metodi basati sulla decomposizione, modelli ARIMA e ARARMA (*AutoRegressive AutoRegressive Moving Average*, Parzen (1982)), sistemi esperti e reti neurali. Tutti questi metodi erano già stati testati nelle edizioni precedenti a meno dei metodi appartenenti alle categorie delle reti neurali, sistemi esperti e decomposizione.

La competizione ha confermato i risultati già ottenuti nelle due edizioni precedenti. In particolare, è stato dimostrato che i metodi semplici (come i metodi basati sul lisciamiento esponenziale) hanno prestazioni migliori rispetto ai metodi complessi (come i metodi basati sui modelli ARIMA e ARARMA) e che il metodo *Theta* (Assimakopoulos e Nikolopoulos, 2000) risulta essere uno dei migliori tra quelli proposti, in quanto ha ottime prestazioni per tutte le tipologie e periodi della serie storica in esame, indipendentemente dall'indice di accuratezza scelto e dall'orizzonte temporale (Makridakis e Hibon, 2000).

Le competizioni hanno proseguito anche in questi ultimi anni e ad oggi è in corso la sesta edizione, in cui si confrontano, oltre ai tradizionali metodi, anche nuovi approcci basati sul *deep learning* e *machine learning*.

## 3.2 Indici per la valutazione degli errori di previsione

In virtù delle conclusioni emerse dalla *M3-competition*, sono stati utilizzati tre diversi indici per valutare l'accuratezza delle quattro procedure adoperate in questa tesi.

Il primo tra questi è l'indice SMAPE, definito come:

$$\text{SMAPE} = \frac{100}{h} \sum_{j=1}^h \frac{|y_{t+j} - \hat{y}_{t+j}|}{(|y_{t+j}| + |\hat{y}_{t+j}|)/2},$$

dove  $y_{t+j}$  è il vero valore della serie al tempo  $t+j$ ,  $\hat{y}_{t+j}$  è la rispettiva previsione e  $j$  è l'orizzonte temporale. L'indicatore misura la percentuale di errore medio tra le previsioni e i valori osservati, ed è simmetrico perché pesa allo stesso modo gli errori di previsione negativi ( $\hat{y}_{t+j} > y_{t+j}$ ) e gli errori positivi ( $\hat{y}_{t+j} < y_{t+j}$ ). Tale indice ha un limite inferiore (0%) ed uno superiore (200%). Un valore SMAPE basso indica una maggiore

accuratezza delle previsioni.

Un altro indice utilizzato è il MASE (errore scalare medio assoluto) proposto da Hyndman e Koehler (2006). Tale indice calcola la media degli errori assoluti tra i valori osservati e le rispettive previsioni, scalando per l'errore medio assoluto che si otterrebbe utilizzando il metodo di previsione *naive* che assume come valore previsto l'ultimo valore osservato, tenendo conto della stagionalità.

Siano  $n$  il numero totale di osservazioni della serie,  $j = 1, \dots, h$  l'orizzonte temporale e  $m$  il periodo della serie, il MASE è definito come:

$$\text{MASE} = \frac{\frac{1}{h} \sum_{j=1}^h |y_{t+j} - \hat{y}_{t+j}|}{\frac{1}{n-m} \sum_{t=1+m}^n |y_t - y_{t-m}|},$$

dove  $m$  vale 1 se la serie è annuale, 4 se i dati sono trimestrali e 12 per le serie mensili. Questo indice è simmetrico e gode della proprietà di invarianza di scala, e può essere utilizzato sia per mettere a confronto diversi metodi previsivi su una stessa serie, ma anche per paragonare l'accuratezza previsiva di uno specifico metodo utilizzando più serie. In aggiunta, questo indice può essere impiegato anche nelle situazioni in cui le serie storiche contengono molti valori prossimi o esattamente pari a zero, poiché l'indice non diverge verso l'infinito né si avvicina a valori eccessivamente elevati, a differenza di quanto accade con l'indice SMAPE quando sia la previsione che il valore reale tendono a zero. Quando il MASE viene utilizzato per confrontare vari metodi previsivi, viene considerato più accurato il metodo con il MASE più basso.

Infine, l'ultimo indice utilizzato è l' RMSE (radice dell'errore quadratico medio):

$$\text{RMSE} = \sqrt{\frac{1}{h} \sum_{j=1}^h (y_{t+j} - \hat{y}_{t+j})^2}.$$

Anche in questo caso, più un metodo è accurato e più il valore RMSE sarà basso. Questo indice dipende dalla scala di misura dei dati ed è sensibile ai valori anomali, ma nella pratica viene frequentemente utilizzato grazie alla sua semplicità di interpretazione, essendo espresso nella stessa scala dei dati originali (Chai e Draxler, 2014).

Con riferimento alle analisi svolte in questa tesi, fissato il periodo della serie, i tre indici sono stati calcolati facendo la media su tutti gli orizzonti per ciascuna serie. Successivamente, sono state calcolate le medie complessive di questi indici tra le diverse serie.

### 3.3 Confronto delle metodologie bootstrap

Tutte le conclusioni emerse dalla *M3-competition*, sono puramente descrittive e non è stato condotto alcun test statistico per valutare se ci fosse una differenza significativa nell'accuratezza previsiva dei ventiquattro metodi in gara. Pertanto, Koning *et al.* (2005) hanno affrontato la questione valutando quali metodi fossero significativamente diversi e, in aggiunta, hanno verificato la veridicità di ciascuna delle quattro conclusioni mediante opportuni test. Dalle analisi è emerso che ci sono metodi significativamente migliori di altri e che la performance di tali metodi non dipende né dall'orizzonte temporale, né dalla categoria della serie. Inoltre, hanno dimostrato che una combinazione dei metodi non conduce necessariamente a misurazioni più precise e che i risultati non variano a seconda della misura di accuratezza utilizzata.

Per valutare se ci fossero differenze significative tra i metodi è stato utilizzato il test di Friedman.

Il test di Friedman (Friedman, 1937, 1939) è un test non parametrico, basato sui ranghi, utilizzato per verificare se esistono differenze tra misurazioni quando le assunzioni alla base dei test parametrici sono violate, o più semplicemente, quando non è possibile verificarle.

Nel caso in esame, l'ipotesi nulla corrisponde all'uguaglianza dell'accuratezza delle previsioni ottenute con le diverse metodologie *bootstrap* adattate, contro l'ipotesi alternativa che almeno un metodo risulti essere diverso dagli altri.

A partire da un dataset  $n \times p$ , dove  $n$  corrisponde al numero di osservazioni della serie storica e  $p$  al totale dei metodi da confrontare, il test calcola i ranghi per ogni unità statistica  $r_{ij}$ , ( $i = 1, \dots, n$ ,  $j = 1, \dots, p$ ) e successivamente esegue la media dei ranghi per ciascun metodo utilizzato  $\bar{r}_j = \sum_{i=1}^n r_{ij}/n$ . Se i metodi sono uguali, allora la media dei ranghi sarà la stessa. L'ipotesi nulla è esprimibile come  $H_0 : \bar{r}_1 = \bar{r}_2 = \dots = \bar{r}_p$  e può essere testata mediante la statistica test

$$\chi = \frac{12n}{p(p+1)} \sum_{j=1}^p (\bar{r}_j - \frac{p+1}{2})^2.$$

Per verificare la significatività, è necessario confrontare il valore della statistica osservata con i valori tabulati. Sotto l'ipotesi nulla,  $\chi$  converge ad una distribuzione chi quadrato con  $p - 1$  gradi di libertà, quando  $n$  tende ad infinito.

Se l'ipotesi nulla non viene rifiutata, allora i metodi risultano essere uguali. In caso contrario si dice che il test è significativo, ed è necessario condurre un'analisi *post-hoc* per valutare quali metodi abbiano portato

al rifiuto di tale ipotesi.

L'analisi *post-hoc* consiste nell'effettuare tutti i possibili confronti a coppie tra i metodi e, successivamente, correggere i *p-value* ottenuti. La correzione è necessaria per poter controllare la probabilità di commettere un errore di primo tipo, che altrimenti risulterebbe essere troppo alta.

Per eseguire i confronti a coppie è stato utilizzato il test non parametrico dei segni, che verifica se due metodi sono uguali andando a calcolare le differenze tra i due punteggi per tutte le coppie di osservazioni. Se i metodi risultano essere uguali, allora il numero di differenze positive sarà pressoché uguale al numero di differenze negative ed avrà distribuzione binomiale con probabilità  $p = 0.5$ . L'ipotesi nulla associata a questo test assume che la distribuzione delle differenze abbia mediana pari a zero (Dixon e Mood, 1946).

La specificazione dell'ipotesi alternativa varia a seconda dell'interpretazione che si vuole ottenere. Specificando un'ipotesi alternativa bidirezionale si sta valutando nel complesso quali sono le coppie di metodi che hanno reso significativo il test di Friedman, ovvero le coppie per cui la mediana delle differenze dei punteggi è diversa da zero. Se invece si vuole andare a valutare se un metodo sia significativamente migliore degli altri è richiesta la specificazione di un'ipotesi alternativa unilaterale sinistra (mediana delle differenze minore di zero). In caso di rifiuto, il punteggio fornito dal metodo preso a riferimento è significativamente più basso dell'altro, ovvero il metodo è migliore.

Per correggere i *p-value* è stata utilizzata la procedura di Hochberg e Rom (1995).

Tutti i test sono stati effettuati utilizzando il punteggio SMAPE, con un livello di significatività  $\alpha = 0.05$ .

### 3.4 Risultati e conclusioni

Conformemente a quanto riportato nell'articolo di Bergmeir *et al.* (2016) per poter valutare le prestazioni dei metodi *bagging*, sono state utilizzate solo una parte delle 3003 serie della *M3-competition*, escludendo le serie appartenenti alla categoria "Altro", per un totale di 2829 serie analizzate.

Le analisi sono state eseguite dividendo le serie in base al periodo e calcolando le previsioni sei passi in avanti per le serie annuali, otto per

quelle trimestrali e diciotto per le serie mensili. Ottenute le previsioni, sono stati calcolati i tre indici di accuratezza.

Fissato un certo indice di accuratezza, per ogni serie esaminata si hanno quattro punteggi: un punteggio misura l'accuratezza delle previsioni ottenute mediante la tecnica del liscio esponenziale, mentre gli altri tre punteggi si riferiscono alle previsioni ottenute mediante la tecnica *bagging* (Capitolo 1, paragrafo 3), dove sono stati utilizzati i tre metodi *bootstrap* descritti nel Capitolo 2 per replicare le serie dei residui.

In particolare, per il *moving block bootstrap* è stato scelto di fissare la lunghezza del blocco  $l$  pari ad otto per le serie annuali e trimestrali, mentre per le serie mensili si è scelto di porre  $l = 24$ .

Per quanto riguarda l'applicazione del *wild bootstrap* si è scelto di assumere distribuzione Normale Standard per la variabile  $\zeta_t^*$ .

### 3.4.1 Risultati per le serie annuali

La tabella 3.2 riporta le medie dei punteggi per ciascuno dei quattro metodi utilizzati. Si può notare che il metodo col punteggio più basso cambia a seconda dell'indicatore che viene preso in considerazione: sulla base dei punteggi SMAPE e MASE, il metodo migliore risulta essere quello basato sul liscio esponenziale, mentre se si considera il punteggio RMSE prevale il metodo che utilizza il *sieve bootstrap*. Tuttavia, il test di Friedman non risulta essere significativo ( $p\text{-value} = 0.293$ ), pertanto i metodi risultano essere uguali.

Metodo	Punteggio medio		
	<i>SMAPE</i>	<i>MASE</i>	<i>RMSE</i>
<i>ETS</i>	<b>17.003</b>	<b>2.859</b>	1182.215
<i>MBB</i>	17.223	2.886	1233.511
<i>SIEVE</i>	17.898	2.999	<b>1155.179</b>
<i>WILD</i>	17.392	2.885	1248.527

Tabella 3.2: Risultati per le serie annuali.

Ciò è ulteriormente confermato dall'analisi *post-hoc*. Infatti, nel tentativo di valutare se il metodo ETS (che ha il punteggio SMAPE più basso) risulti essere superiore agli altri, si giunge alla stessa conclusione. L'ipotesi nulla non viene rifiutata per nessuna delle coppie (Tabella 3.3).



<b>Coppia</b>	$p_{Hoch}$
<i>ETS-MBB</i>	0.709
<i>ETS-SIEVE</i>	0.125
<i>ETS-WILD</i>	0.709

Tabella 3.3: Analisi *post-hoc* per le serie annuali. Valutazione della performance del metodo ETS. *P-value* corretti mediante il metodo di Hochberg e Rom.

### 3.4.2 Risultati per le serie trimestrali

Per quanto riguarda le serie trimestrali, il test di Friedman non risulta essere significativo, con un  $p\text{-value} = 0.0487$ .

Le tabelle 3.1 e 3.2 riportano rispettivamente i punteggi medi associati a ciascuno dei quattro metodi e i risultati derivanti dai confronti a coppie, eseguiti per identificare eventuali coppie di metodi con punteggi mediani diversi.

<b>Metodo</b>	<b>Punteggio medio</b>		
	<i>SMAPE</i>	<i>MASE</i>	<i>RMSE</i>
<i>ETS</i>	<b>9.684</b>	1.170	<b>598.734</b>
<i>MBB</i>	9.772	<b>1.157</b>	602.213
<i>SIEVE</i>	9.985	1.247	623.325
<i>WILD</i>	10.105	1.246	617.213

Figura 3.1: Risultati per le serie trimestrali.

<b>Coppia</b>	$p_{Hoch}$
<i>ETS-MBB</i>	0.726
<i>ETS-SIEVE</i>	0.534
<i>ETS-WILD</i>	0.159
<i>MBB-SIEVE</i>	0.534
<i>MBB-WILD</i>	0.227
<i>SIEVE-WILD</i>	0.799

Figura 3.2: Analisi *post-hoc*. Confronti tra le coppie di metodi. *P-value* corretti mediante il metodo di Hochberg e Rom.

Poiché l'applicazione del metodo ETS porta al minor punteggio SMAPE, è stata effettuata un'analisi *post-hoc* per valutare se questo metodo sia statisticamente migliore degli altri. Dall'analisi evince che tale metodo risulta essere leggermente migliore della procedura *bagging* che utilizza il *wild bootstrap*, ma non lo è rispetto alle altre metodologie *bagging* (Tabella 3.4).

Coppia	$p_{Hoch}$
<i>ETS-MBB</i>	0.837
<i>ETS-SIEVE</i>	0.156
<i>ETS-WILD</i>	0.04

Tabella 3.4: Analisi *post-hoc* per le serie trimestrali. Valutazione della performance del metodo ETS. *P-value* corretti mediante il metodo di Hochberg e Rom.

### 3.4.3 Risultati per le serie mensili

La tabella 3.3 riporta i punteggi medi per le serie annuali.

In questo caso il test di Friedman è fortemente significativo, con un *p-value*  $< 0.001$ . Facendo tutti i possibili confronti a coppie, si desume che tutte le coppie hanno portato al rifiuto dell'ipotesi nulla a meno delle coppie *MBB-SIEVE* e *ETS-WILD* (Tabella 3.4).

Metodo	Punteggio medio		
	<i>SMAPE</i>	<i>MASE</i>	<i>RMSE</i>
<i>ETS</i>	14.139	0.865	755.008
<i>MBB</i>	<b>13.704</b>	<b>0.834</b>	<b>742.022</b>
<i>SIEVE</i>	13.762	0.845	748.741
<i>WILD</i>	14.779	0.888	816.736

Figura 3.3: Risultati per le serie Mensili.

Coppia	$p_{Hoch}$
<i>ETS-MBB</i>	$<0.001$
<i>ETS-SIEVE</i>	$<0.001$
<i>ETS-WILD</i>	0.543
<i>MBB-SIEVE</i>	0.543
<i>MBB-WILD</i>	$<0.001$
<i>SIEVE-WILD</i>	$<0.001$

Figura 3.4: Analisi *post-hoc*. Confronti tra le coppie di metodi. *P-value* corretti mediante il metodo di Hochberg e Rom.

Rispetto ai due casi precedenti, il metodo che esegue il *bagging* utilizzando il *moving block bootstrap*, risulta essere quello con il punteggio *SMAPE* più basso. Da un'analisi descrittiva, sembra che le applicazioni del *moving block bootstrap* e del *sieve bootstrap* portino a previsioni migliori rispetto alle altre e due procedure.

Per testare le performance dei metodi sono state effettuate due valutazioni. La prima, coerente con le analisi fatte ai passi precedenti, valuta se il metodo *MBB* sia migliore degli altri (Tabella 3.5). La seconda, invece, mira a valutare la performance del metodo basato sul liscio esponenziale ma da un altro punto di vista. In particolare, si effettua un test dei segni con ipotesi alternativa unilaterale destra, ovvero si va

a verificare se il metodo basato sul lisciamiento esponenziale sia significativamente peggiore dei metodi *bagging* (Tabella 3.6).

Coppia	$p_{Hoch}$
<i>MBB-ETS</i>	<0.001
<i>MBB-SIEVE</i>	0.271
<i>MBB-WILD</i>	<0.001

Figura 3.5: Analisi *post-hoc* per le serie mensili. Valutazione della performance del metodo MBB. *P-value* corretti mediante il metodo di Hochberg e Rom.

Coppia	$p_{Hoch}$
<i>ETS-MBB</i>	<0.001
<i>ETS-SIEVE</i>	<0.001
<i>ETS-WILD</i>	0.746

Figura 3.6: Analisi *post-hoc* per le serie mensili. Valutazione della performance del metodo ETS. *P-value* corretti mediante il metodo di Hochberg e Rom.

Dai risultati ottenuti emerge che il metodo basato sul MBB presenta prestazioni paragonabili a quelle del metodo basato sul *sieve bootstrap* e risulta essere più accurato rispetto al metodo basato sul *wild bootstrap* e al metodo ETS.

D'altra parte, si osserva che la procedura tradizionale fornisce previsioni meno accurate rispetto alla procedura *bagging*, ma questa differenza si verifica solamente quando si utilizza il *moving block bootstrap* o il *sieve bootstrap*.

### 3.4.4 Considerazioni finali

Dall'insieme delle analisi condotte emerge che la procedura *bagging* costituisce un'alternativa valida al metodo tradizionale.

Nello specifico, per le serie annuali e trimestrali, le previsioni ottenute tramite *bagging* risultano essere altrettanto accurate di quelle ottenute con il metodo basato sul lisciamiento esponenziale, senza mostrare differenze significative tra le diverse metodologie *bootstrap* impiegate. Al contrario, questa procedura funziona meglio di quella tradizionale se si considerano le serie mensili, ma solo se si esclude il *wild bootstrap* dalla valutazione.

Tuttavia, ci sono degli aspetti da tenere in considerazione. Innanzitutto, il numero di replicazioni utilizzato per eseguire il *bagging* è stato impostato a 100, ed un valore maggiore potrebbe migliorare ulteriormente i risultati. In aggiunta, le serie mensili presentano un numero di osservazioni decisamente più alto rispetto alle serie trimestrali ed annuali e

questo potrebbe aver inciso sulla significatività dei test effettuati. Nell'articolo di Bergmeir *et al.* (2016) la procedura *bagging* che utilizza il *wild bootstrap* non è stata oggetto di studio, poiché gli autori ritengono che "per le serie storiche regolarmente distanziate, il *wild bootstrap* non raggiunge lo stesso ordine di accuratezza del MBB, quindi tale metodo non può essere messo in competizione con i metodi *bootstrap* basati sui blocchi". Sebbene in quest'affermazione non sia stato menzionato il *sieve bootstrap* (poiché neanche questo metodo è stato studiato dagli autori), potrebbe essere questo il motivo per cui tale procedura non abbia fornito risultati soddisfacenti.

Gli autori hanno altresì dimostrato che la procedura *bagging* fornisce previsioni migliori anche rispetto ad altri metodi che hanno preso parte alla competizione M3, indipendentemente dal periodo della serie osservata e che la procedura *bagging* che utilizza il *moving block bootstrap* e il *sieve bootstrap* risulta essere tanto accurata quanto il metodo *Theta* (metodo vincitore) per le serie mensili, confermando così l'efficacia della procedura.

# Appendice A

## A.1 Codice R utilizzato

Si riporta il codice R utilizzato per fare le analisi.

```
# Caricamento delle librerie necessarie
rm(list=ls())
library(fpp3)
library(tidyverse)
library(boot)
library(forecast)
library(Mcomp)
library(feasts)
library(tseriesEntropy)
library(nortsTest)
library(ggpubr)
library(rstatix)
library(readxl)

# Funzione che effettua la procedura bagging utilizzando
il sieve bootstrap per le serie annuali.

bld.sieve.res.year.bootstrap = function(y,nboot=100){
  #1. Si applica la trasformazione di Box-Cox
  l = guerrero(y, lower = 0, upper = 1, .period = frequency(y))
  y.cox = BoxCox(y, lambda = l)
  #2. Si applica la decomposizione loess alla serie
  T = length(y)
  index = 1:T
}
```

```

alfa = 6/T
loess = loess(y.cox ~ index, span=alfa)
trend.loess = ts(loess$fitted, start=start(y), frequency = 1)

#3. Applicazione del sieve bootstrap alla serie
    dei residui
residui = ts(loess$residuals, start=start(y), frequency = 1)
sieve.b.residui = surrogate.AR(residui, nsurr=nboot)$surr

#4. Ricomposizione della serie
serie.boot= apply(sieve.b.residui,2,
                 function(x) x + trend.loess)

#5. Trasformazione inversa di Box-Cox
Y.boot = apply(serie.boot,2,function(x) inv_box_cox(x,lambda = 1))

#6. Si trasforma la matrice Y.boot in una lista di nboot=100
# elementi dove ciascun elemento è un oggetto ts()
Y.boot.list = list()
Y.boot.list[[1]] = y
for(i in 2: nboot){
  Y.boot.list[[i]] = ts(Y.boot[,i],
    start = start(y), frequency = frequency(y))
}

Y.boot.list
}

# Funzione che esegue il bagging applicando il wild
bootstrap per le serie annuali.

bld.wild.year.bootstrap = function(y,nboot=100){
  #1. Trasformazione di Box-Cox
  l = guerrero(y, lower = 0, upper = 1, .period = frequency(y))
  y.cox = BoxCox(y, lambda = l)

  #2. Applicazione della decomposizione loess alla serie
  T = length(y)

```

```

index = 1:T
alfa = 6/T
loess = loess(y.cox ~ index, span=alfa)
trend.loess = ts(loess$fitted, start=start(y), frequency = 1)

#3. Applicazione del wild bootstrap alla serie
    dei residui
residui = ts(loess$residuals, start=start(y), frequency = 1)
wild.b.residui = residui.rammen.boot(residui,nboot)

#4. Ricomposizione della serie
serie.boot= apply(wild.b.residui,2,
                 function(x) x + trend.loess)
#5. Trasformazione inversa di Box-Cox
Y.boot = apply(serie.boot,2,function(x) inv_box_cox(x,lambda = 1))

#6. Trasformazione della matrice y.boot in una lista
    di nboot elementi
# dove ciascun elemento è un oggetto ts()
Y.boot.list = list()
Y.boot.list[[1]] = y
for(i in 2: nboot){
  Y.boot.list[[i]] = ts(Y.boot[,i],
    start = start(y), frequency = frequency(y))
}
Y.boot.list
}

residui.rammen.boot = function(y, nboot){
  # A partire dalla serie dei residui (y) genero una matrice (A)
  # che conterrà le nuove serie dei residui bootstrap.
  p = length(y)
  A = matrix(NA, nrow = p, ncol=nboot)

  for(i in 1: nboot){
    idx = sample(1:p, p ,replace = T)
    A[,i] = y[idx] * rrammen(p, construct = "normal-1")
  }
  A
}

```

```

}

rmammen <- function(n,
                    construct = c( "normal-1", "Mammen")){
  construct <- match.arg(construct)

  if (length(n) > 1) n <- length(n)

  if (construct == "Mammen"){
    .vals <- c(-(sqrt(5)-1)/2, (sqrt(5)+1)/2)
    .probs <- rev(abs(.vals)/sqrt(5))

    sample(.vals, size = n, replace = TRUE, prob = .probs)
  } else{
    .v <- rnorm(n)

    (.v)
  }
}

```

# Funzione che esegue il bagging applicando il sieve bootstrap per le serie mensili e trimestrali.

```

bld.sieve.bootstrap = function(y,nboot=100){
  #1. Trasformazione di Box-Cox
  l = guerrero(y, lower = 0, upper = 1, .period = frequency(y))
  y.cox = BoxCox(y, lambda = l)

  #2. Decomposizione STL della serie
  y.stl = y.cox %>%
    stl(t.window=13, s.window="periodic", robust=TRUE)

  #3. Applicazione del sieve bootstrap alla serie
  dei residui
  residui = y.stl$time.series[,3]
  sieve.b.residui = surrogate.AR(residui,nsurr = nboot)$surr

  #4. Ricomposizione delle componenti della serie
  serie.boot= apply(sieve.b.residui,2,
                   function(x) x + y.stl$time.series[,2]

```



```

+y.stl$time.series[,1])

#5. Trasformazione inversa di Box-Cox per avere i
# dati su scala originaria.
Y.boot = apply(serie.boot,2,function(x) inv_box_cox(x,lambda = 1))

#6. Trasformazione della matrice y.boot in una lista di
nboot=100
# elementi dove ciascun elemento è un oggetto ts()
Y.boot.list = list()
Y.boot.list[[1]] = y
for(i in 2: nboot){
  Y.boot.list[[i]] = ts(Y.boot[,i],
                        start = start(y), frequency = frequency(y))
}

Y.boot.list
}

# Funzione che esegue il baggining applicando il wild bootstrap
# per le serie mensili e trimestrali.

bld.wild.bootstrap = function(y,nboot=100){
  #1. Trasfromazione di Box-Cox
  l = guerrero(y, lower = 0, upper = 1, .period = frequency(y))
  y.cox = BoxCox(y, lambda = 1)

  #2. Decomposizione STL della serie
  y.stl = y.cox %>%
    stl(t.window=13, s.window="periodic", robust=TRUE)

  #3. Applicazione del wild bootstrap alla serie
  dei residui
  residui = y.stl$time.series[,3]
  wild.b.residui = residui.rammen.boot(residui,nboot)

  #4. Ricomposizione della serie
  serie.boot= apply(wild.b.residui,2,
                    function(x) x + y.stl$time.series[,2]+
                    y.stl$time.series[,1])

```

```

#5. Trasformazione inversa di Box-Cox
Y.boot = apply(serie.boot,2,function(x) inv_box_cox(x,lambda = 1))

#6. Trasformazione della matrice y.boot in una lista
    di nboot elementi
#  dove ciascun elemento è un oggetto ts().
Y.boot.list = list()
Y.boot.list[[1]] = y
for(i in 2: nboot){
  Y.boot.list[[i]] = ts(Y.boot[,i],
                        start = start(y), frequency = frequency(y))
}
Y.boot.list
}

```

```

# Applicazione della procedura per le serie annuali, trimestrali
# e mensili dell'M3-competition.

```

```

#1. Serie annuali
m3.year = subset(M3,"yearly")

```

```

sMAPE.mbb = c()
MASE.mbb = c()
RMSE.mbb = c()

```

```

sMAPE.ets = c()
MASE.ets = c()
RMSE.ets = c()

```

```

sMAPE.wild = c()
MASE.wild = c()
RMSE.wild = c()

```

```

sMAPE.sieve = c()
MASE.sieve = c()
RMSE.sieve = c()

```

```

for( i in 1:645){
  print(i)
  serie = m3.year[[i]]
  y = serie$x
  y.fut = serie$xx

  # MBB
  model.mbb = baggedETS(y,
                        bootstrapped_series =
                        bld.mbb.bootstrap(y,100, block_size = 8))

  #previsione bootstrap
  fcast.mbb = forecast(model.mbb,h=6)

  #accuratezza
  sMAPE.mbb[i] = mean(200*abs(y.fut-fcast.mbb$mean)/(abs(y.fut)
                                                    +abs(fcast.mbb$mean)))
  MASE.mbb[i] = accuracy(fcast.mbb, serie$xx)[2,6]
  RMSE.mbb[i] = accuracy(fcast.mbb, serie$xx)[2,2]

  # ETS
  fit <- ets(y)
  #previsione ets
  fc = fit %>% forecast(h=6)

  #accuratezza
  sMAPE.ets[i] = mean(200*abs(y.fut-fc$mean)/
                    (abs(y.fut)+abs(fc$mean)))
  MASE.ets[i] = accuracy(fc, serie$xx)[2,6]
  RMSE.ets[i] = accuracy(fc, serie$xx)[2,2]

  # SIEVE
  model.sieve = baggedETS(y,
                          bootstrapped_series =
                          bld.sieve.res.year.bootstrap(y,100))

  #previsione bootstrap

```

```

fcast.sieve = forecast(model.sieve,h=6)
#accuratezza
sMAPE.sieve[i] = mean(200*abs(y.fut-fcast.sieve$mean)/
                      (abs(y.fut)+abs(fcast.sieve$mean)))
MASE.sieve[i] = accuracy(fcast.sieve, serie$xx)[2,6]
RMSE.sieve[i] = accuracy(fcast.sieve, serie$xx)[2,2]

# WILD
model = baggedETS(y,
                  bootstrapped_series =
                  bld.wild.year.bootstrap(y,100))

#previsione bootstrap
fcast.wild = forecast(model,h=6)
#accuratezza
sMAPE.wild[i] = mean(200*abs(y.fut-fcast.wild$mean)/
                     (abs(y.fut)+abs(fcast.wild$mean)))
MASE.wild[i] = accuracy(fcast.wild, serie$xx)[2,6]
RMSE.wild[i] = accuracy(fcast.wild, serie$xx)[2,2]
}

```

```

#2. Serie trimestrali
m3.quart = subset(M3,"quarterly")

```

```

MASE.mbb = c()
MASE.ets = c()
MASE.wild = c()
MASE.sieve = c()

```

```

RMSE.mbb = c()
RMSE.ets = c()
RMSE.wild = c()
RMSE.sieve = c()

```

```

sMAPE.mbb = c()
sMAPE.ets = c()
sMAPE.wild = c()
sMAPE.sieve = c()

```

```

for( i in 1:756){
  print(i)
  serie = m3.quart[[i]]
  y = serie$x
  y.fut = serie$xx

  # ETS
  fit <- ets(y)
  #previsione ets
  fc = fit %>% forecast(h=8)
  #accuratezza
  sMAPE.ets[i] = mean(200*abs(y.fut-
                          fc$mean)/(abs(y.fut)+abs(fc$mean)))
  MASE.ets[i] = accuracy(fc, serie$xx)[2,6]
  RMSE.ets[i] = accuracy(fc, serie$xx)[2,2]

  #MBB
  model.mbb = baggedETS(y,
                        bootstrapped_series =
                          bld.mbb.bootstrap(y,100,block_size = 8))

  #previsione bootstrap
  fcast.mbb = forecast(model.mbb,h=8)

  #accuratezza
  sMAPE.mbb[i] = mean(200*abs(y.fut-fcast.mbb$mean)/
                      (abs(y.fut)+abs(fcast.mbb$mean)))
  MASE.mbb[i] = accuracy(fcast.mbb, serie$xx)[2,6]
  RMSE.mbb[i] = accuracy(fcast.mbb, serie$xx)[2,2]

  # SIEVE
  model.sieve = baggedETS(y,
                          bootstrapped_series =
                            bld.sieve.bootstrap(y,100))
  #previsione bootstrap
  fcast.sieve = forecast(model.sieve,h=8)

  #accuratezza

```

```

sMAPE.sieve[i] = mean(200*abs(y.fut-fcast.sieve$mean)/
                      (abs(y.fut)+abs(fcast.sieve$mean)))
MASE.sieve[i] = accuracy(fcast.sieve, serie$xx) [2,6]
RMSE.sieve[i] = accuracy(fcast.sieve, serie$xx) [2,2]

# WILD
model.wild = baggedETS(y,
                      bootstrapped_series =
                      bld.wild.bootstrap(y,100))

#previsione bootstrap
fcast.wild = forecast(model.wild,h=8)
#accuratezza
sMAPE.wild[i] = mean(200*abs(y.fut-fcast.wild$mean)/
                    (abs(y.fut)+abs(fcast.wild$mean)))
MASE.wild[i] = accuracy(fcast.wild, y.fut) [2,6]
RMSE.wild[i] = accuracy(fcast.wild, y.fut) [2,2]

}

```

```

#3. Serie mensili
m3.month = subset(M3,"monthly")

```

```

MASE.ets = c()
MASE.mbb = c()
MASE.wild = c()
MASE.sieve = c()

```

```

RMSE.mbb = c()
RMSE.ets = c()
RMSE.wild = c()
RMSE.sieve = c()

```

```

sMAPE.mbb = c()
sMAPE.ets = c()
sMAPE.wild = c()
sMAPE.sieve = c()

```

```

for( i in 1:1428){
  print(i)
  serie = m3.month[[i]]
  y = serie$x
  y.fut = serie$xx

  # ETS
  fit <- ets(y)
  #previsione ets
  fc = fit %>% forecast(h=18)

  #accuratezza
  sMAPE.ets[i] = mean(200*abs(y.fut-fc$mean)/(abs(y.fut)+abs(fc$mean)))
  MASE.ets[i] = accuracy(fc, serie$xx)[2,6]
  RMSE.ets[i] = accuracy(fc, serie$xx)[2,2]

  # MBB
  model.mbb = baggedETS(y,
                        bootstrapped_series =
                        bld.mbb.bootstrap(y,100,block_size = 24))

  #previsione bootstrap
  fcast.mbb = forecast(model.mbb,h=18)
  #accuratezza
  sMAPE.mbb[i] = mean(200*abs(y.fut-fcast.mbb$mean)/
                      (abs(y.fut)+abs(fcast.mbb$mean)))
  MASE.mbb[i] = accuracy(fcast.mbb, serie$xx)[2,6]
  RMSE.mbb[i] = accuracy(fcast.mbb, serie$xx)[2,2]

  # SIEVE
  model.sieve = baggedETS(y,
                          bootstrapped_series =
                          bld.sieve.bootstrap(y,100))

  #previsione bootstrap
  fcast.sieve = forecast(model.sieve,h=18)

  #accuratezza
  sMAPE.sieve[i] = mean(200*abs(y.fut-fcast.sieve$mean)/
                        (abs(y.fut)+abs(fcast.sieve$mean)))

```

```

MASE.sieve[i] = accuracy(fcast.sieve, y.fut)[2,6]
RMSE.sieve[i] = accuracy(fcast.sieve, y.fut)[2,2]

# WILD
model.wild = baggedETS(y,
                      bootstrapped_series =
                      bld.wild.bootstrap(y,100))

#previsione bootstrap
fcast.wild = forecast(model.wild,h=18)
#accuratezza
sMAPE.wild[i] = mean(200*abs(y.fut-fcast.wild$mean)/
                    (abs(y.fut)+abs(fcast.wild$mean)))
MASE.wild[i] = accuracy(fcast.wild, y.fut)[2,6]
RMSE.wild[i] = accuracy(fcast.wild, y.fut)[2,2]

}

```

```

# VALUTAZIONE DELLA BONTÀ DEI METODI
# Test di Friedman ed analisi post-hoc.

# APPLICAZIONE: DATI ANNUALI, INDICE SMAPE.
Fried_ann <- read_excel("sMAPE_Bagged_annuali.xlsx")
Fried_ann = Fried_ann %>% mutate(id = 1:645)

# Trasformazione dei dati in formato lungo per poter
# applicare il test di Friedman.
Fried_a_long <- Fried_ann %>%
  gather(key = "Tipo", value = "value", ETS,SIEVE,MBB,WILD) %>%
  convert_as_factor(id, Tipo)

Fried_a_long %>% friedman_test(value ~ Tipo |id)

# Analisi post-hoc
# Confronti a coppie: ipotesi alternativa bidirezionale.
# Questo test viene eseguito per capire quali coppie hanno
# portato al rifiuto dell'ipotesi nulla.

```



```

Fried_a_long %>%
  sign_test(value ~ Tipo, p.adjust.method = "hochberg")

# Confronti a coppie: ipotesi alternativa unilaterale sinistra.
# Tale test permette di valutare se un metodo preso a
# riferimento sia significativamente migliore degli altri.
# Lo si applica prendendo ETS come metodo di riferimento,
# essendo quello con il punteggio più basso.

Fried_a_long %>%
  sign_test(value ~ Tipo, p.adjust.method = "hochberg",
            alternative = 'l', ref.group = "ETS")

# APPLICAZIONE: DATI TRIMESTRALI, INDICE SMAPE.
Fried_trim <- read_excel("sMape_Trimestrali.xlsx", range = "A1:D757")
Fried_trim = Fried_trim %>% mutate(id = 1:756)

# Trasformazione dei dati in formato lungo e applicazione del
# test di Friedman
Fried_t_long <- Fried_trim %>%
  gather(key = "Tipo", value = "value", ETS, MBB, SIEVE, WILD) %>%
  convert_as_factor(id, Tipo)

Fried_t_long %>% friedman_test(value ~ Tipo | id)

# Analisi post hoc.
# Ipotesi alternativa bidirezionale
Fried_t_long %>%
  sign_test(value ~ Tipo, p.adjust.method = "hochberg")

# Ipotesi alternativa unilaterale sinistra.
# Metodo di riferimento: MBB (come nell'articolo)
Fried_t_long %>%
  sign_test(value ~ Tipo, p.adjust.method = "hochberg",
            alternative = 'l', ref.group = "MBB")

```

```

# APPLICAZIONE: DATI MENSILI, INDICE SMAPE.
Fried_mensili <- read_excel("sMAPE_Mensili_riferimento.xlsx")
Fried_mensili <- Fried_mensili %>% mutate(id = 1:1428)

# Trasformazione dei dati in formato lungo e
# applicazione del test di Friedman.
Fried_m_long <- Fried_mensili %>%
  gather(key = "Tipo", value = "value", ETS, MBB, SIEVE, WILD) %>%
  convert_as_factor(id, Tipo)
Fried_m_long
Fried_m_long %>% friedman_test(value ~ Tipo | id)

# Analisi post-hoc.
# Confronti a coppie: ipotesi alternativa bidirezionale
Fried_m_long %>%
  sign_test(value ~ Tipo, p.adjust.method = "hochberg")

# Confronti a coppie: ipotesi alternativa unilaterale sinistra.
# Metodo di riferimento: MBB.
Fried_m_long %>%
  sign_test(value ~ Tipo, p.adjust.method = "hochberg",
            alternative = 'l', ref.group = "MBB")

```

## A.2 Ringraziamenti

Desidero dedicare questo spazio della mia tesi a tutte le persone che mi hanno supportata in questo percorso di crescita universitaria.

Un sentito grazie alla mia relatrice Luisa Bisaglia per la sua disponibilità e tempestività ad ogni mia richiesta. La ringrazio per avermi sempre indirizzata nei momenti di indecisione.

Grazie ai miei amichetti Andrea, Carolina, Emma, Federico, Filippo, Francesco, Irene, Luna, Martha e Sara per essere stati il mio punto di riferimento a Padova, ognuno di voi con le sue sfaccettature mi ha arricchita come persona e mi ha fatto passare momenti bellissimi. Sono orgogliosa di voi e felice di vedere come state raggiungendo i vostri obiettivi, vi voglio bene.

Un enorme grazie a Carolina e Luna per essere state un supporto costante, soprattutto in questi ultimi due anni. Vi ringrazio per essere sempre disponibili ad ascoltarmi, per le passeggiate delle sette, per le cene e per aver condiviso risate e pianti.

Grazie ad Alice, Martina ed Isotta per essere sempre state con me nonostante la lontananza. Grazie a Viviana per ascoltare quotidianamente le mie lamentele e per i messaggi motivazionali. Grazie a Cristiana per le pause tisana e per i suoi preziosi consigli. Grazie a Giulia per avermi sempre sollevata nei miei momenti no anche alle quattro di notte, sei sempre stata e sempre sarai il mio porto sicuro, anche se ci separano migliaia di chilometri.

Infine, il mio più grande ringraziamento va a Filippo, mamma e papà. Siete la mia colonna portante, e non potrei essere più grata di avervi come famiglia. Grazie per avermi dato la possibilità di studiare fuori, per aver creduto in me e per avermi supportata (e sopportata!) sempre. Senza il vostro amore, incoraggiamento e sacrificio, non sarei mai arrivata fin qui. Vi voglio bene oltre ogni parola e sono felice di poter condividere con voi ogni traguardo raggiunto e ogni successo futuro. Grazie di cuore per tutto.

# Bibliografia

- Akaike H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, **19**(6), 716–723.
- Assimakopoulos V.; Nikolopoulos K. (2000). The theta model: a decomposition approach to forecasting. *International journal of forecasting*, **16**(4), 521–530.
- Bergmeir C.; Hyndman R.; Benítez J. (2016). Bagging exponential smoothing methods using stl decomposition and box–cox transformation. *International Journal of Forecasting*, **32**, 303–312.
- Berkowitz J.; Kilian L. (2000). Recent developments in bootstrapping time series. *Econometric Reviews*, **19**(1), 1–48.
- Bickel P. J.; Bühlmann P. (1997). Closure of linear processes. *Journal of Theoretical Probability*, **10**, 445–479.
- Box G. E. P.; Cox D. R. (1964). An analysis of transformations (with discussion). *Journal of the Royal Statistical Society*, **B**(26), 211–252.
- Brockwell P. J.; Davis R. A.; Brockwell P. J.; Davis R. A. (1987). Model building and forecasting with arima processes. *Time series: Theory and methods*, pp. 265–319.
- Bühlmann P. (1997). Sieve bootstrap for time series. *Bernoulli*, **3**(2), 123 – 148.
- Bühlmann P. (2002). Bootstraps for Time Series. *Statistical Science*, **17**(1), 52 – 72.
- Carlstein E. (1986). The use of subseries values for estimating the variance of a general statistic from a stationary sequence. *The Annals of Statistics*, **14**(3), 1171–1179.
- Carlstein E.; Do K.-A.; Hall P.; Hesterberg T.; Künsch H. R. (1998). Matched-block bootstrap for dependent data. *Bernoulli*, **4**(3), 305–328.

- Chai T.; Draxler R. R. (2014). Root mean square error (rmse) or mean absolute error (mae)?—arguments against avoiding rmse in the literature. *Geoscientific model development*, **7**(3), 1247–1250.
- Choi E.; Hall P. (2000). Bootstrap confidence regions computed from autoregressions of arbitrary order. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, **62**(2), 461–477.
- Cleveland R. B.; Cleveland W. S.; McRae J.; Terpenning I. (1990). Stl: A seasonal-trend decomposition procedure based on loess. *Official Statistics.*, **6**, 3–73.
- Cleveland W. S. (1992). Local regression models. *Statistical Models in S*.
- Davison A. C.; Hinkley D. V. (1997). *Bootstrap Methods and their Application*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.
- Delaney N. J.; Chatterjee S. (1986). Use of the bootstrap and cross-validation in ridge regression. *Journal of Business & Economic Statistics*, **4**(2), 255–262.
- Di Fonzo T.; Lisi F. (2005). *Serie storiche economiche : analisi statistiche e applicazioni*. Carocci, Roma.
- Dixon W. J.; Mood A. M. (1946). The statistical sign test. *Journal of the American Statistical Association*, **41**(236), 557–566.
- Efron B. (1992). *Bootstrap Methods: Another Look at the Jackknife*, pp. 569–593. Springer New York, New York, NY.
- Efron B.; Tibshirani R. J. (1993). *An Introduction to the Bootstrap* in Monographs on Statistics and Applied Probability. Numero 57. Chapman & Hall/CRC, Boca Raton, Florida, USA.
- Friedman M. (1937). The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the american statistical association*, **32**(200), 675–701.
- Friedman M. (1939). A correction: The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association*, **34**(205), 109–109.
- Goodwin P. (2010). The holt-winters approach to exponential smoothing: 50 years old and going strong. *Foresight*, **19**(19), 30–33.

- Grenander U. (1981). *Abstract inference / Ulf Grenander*. Wiley New York.
- Guerrero V. (1993). Time-series analysis supported by power transformations. *Journal of Forecasting.*, **12**, 37–48.
- Hall P.; Horowitz J. L.; Jing B.-Y. (1995). On blocking rules for the bootstrap with dependent data. *Biometrika*, **82**(3), 561–574.
- Hochberg Y.; Rom D. (1995). Extensions of multiple testing procedures based on simes' test. *Journal of Statistical Planning and Inference*, **48**(2), 141–152.
- Horowitz J. L. (2001). Chapter 52 - the bootstrap. volume 5 di *Handbook of Econometrics*, pp. 3159–3228. Elsevier.
- Hyndman R. J.; Athanasopoulos G. (2018). *Forecasting : principles and practice*. OTexts, S.l, 2. ed. edizione.
- Hyndman R. J.; Koehler A. B. (2006). Another look at measures of forecast accuracy. *International journal of forecasting*, **22**(4), 679–688.
- Härdle W.; Horowitz J.; Kreiss J.-P. (2003). Bootstrap methods for time series. *International Statistical Review*, **71**(2), 435–459.
- Koning A. J.; Franses P. H.; Hibon M.; Stekler H. O. (2005). The m3 competition: Statistical tests of the results. *International journal of forecasting*, **21**(3), 397–409.
- Künsch H. R. (1989). The jackknife and the bootstrap for general stationary observations. *The Annals of Statistics*, **17**(3), 1217–1241.
- Lahiri S. N. (2003). *Resampling methods for dependent data*. Springer Science & Business Media.
- Liu R. Y. (1988). Bootstrap procedures under some non-iid models. *The annals of statistics*, **16**(4), 1696–1708.
- Liu R. Y.; Singh K. *et al.* (1992). Moving blocks jackknife and bootstrap capture weak dependence. *Exploring the limits of bootstrap*, **225**, 248.
- Makridakis S.; Hibon M. (2000). The m3-competition: results, conclusions and implications. *International journal of forecasting*, **16**(4), 451–476.
- Paparoditis E.; Politis D. N. (2001). Tapered block bootstrap. *Biometrika*, **88**(4), 1105–1119.

- Parzen E. (1982). Ararma models for time series analysis and forecasting. *Journal of Forecasting*, **1**(1), 67–82.
- Politis D. N.; Romano J. P. (1991). *A circular block-resampling procedure for stationary data*. Purdue University. Department of Statistics.
- Politis D. N.; Romano J. P. (1994). The stationary bootstrap. *Journal of the American Statistical association*, **89**(428), 1303–1313.
- Tukey J. (1958). The jackknife. abstract. *Annals of Mathematical Statistics*, **29**, 614.
- Wu C.-F. J. (1986). Jackknife, bootstrap and other resampling methods in regression analysis. *the Annals of Statistics*, **14**(4), 1261–1295.