



Università
Ca' Foscari
Venezia

Corso di Laurea Magistrale in

Economia e Finanza

Tesi di Laurea

**Dati ad alta frequenza e volatilità:
i modelli ibridi HEAVY-ESN e HARQ-ESN**

Relatore

Claudio Pizzi

Correlatore

Marco Corazza

Laureando

Devis Ielo

Matricola 874918

Anno Accademico

2022 / 2023

INDICE

Indice delle figure.....	V
Indice delle tabelle.....	VII
Capitolo 1: Introduzione alle serie storiche e ai dati ad alta frequenza.....	1
Capitolo 2: La volatilità e i modelli per la volatilità.....	9
Paragrafo 2.1: I dati ad alta frequenza.....	9
Paragrafo 2.2: Gli stimatori non parametrici della volatilità.....	11
Paragrafo 2.3: Il modello HAR-RV di Corsi.....	16
Paragrafo 2.4: Il modello HAR-Q di Bollerslev.....	19
Paragrafo 2.5: Il modello HEAVY di Sheppard e Shephard.....	21
Paragrafo 2.6: Le echo state network (ESN).....	24
2.6.1 Breve introduzione alla reti neurali ricorrenti.....	24
2.6.2 La struttura delle echo state network.....	27
2.6.3 Una breve panoramica sulle numerose estensioni della ESN.....	34
Paragrafo 2.7: Gli algoritmi di ottimizzazione evolutivi.....	37
2.7.1 L'algoritmo del particle swarm optimization (PSO).....	38
2.7.2 L'algoritmo dell'artificial bee colony (ABC).....	41
Capitolo 3: Il modelli ibridi per la volatilità ad alta frequenza.....	45
Paragrafo 3.1: Lo scopo dei modelli ibridi per la volatilità.....	45
Paragrafo 3.2: Le serie storiche finanziarie oggetto di analisi.....	46
Paragrafo 3.3: L'applicativo R e i relativi pacchetti di risorse.....	51
Paragrafo 3.4: La scelta e la dimensione dei dati utilizzati.....	52
Paragrafo 3.5: Training set e validation set.....	53
Paragrafo 3.6: La tecnica della rolling window.....	54
Paragrafo 3.7: Definizione dei modelli ibridi.....	56
3.7.1 I modelli stocastici per la volatilità.....	56
3.7.2 La configurazione della echo state network.....	57
3.7.3 Gli iperparametri della rete.....	58

Paragrafo 3.8: Il confronto dei risultati dei modelli ibridi proposti.....	60
3.8.1 Il modello HAR-ESN-PSO.....	60
3.8.2 Il modello HEAVY-ESN-PSO.....	70
3.8.3 Il modello HARQ-ESN-PSO.....	77
3.8.4 Il confronto tra l' algoritmo PSO e l' algoritmo ABC.....	83
Capitolo 4: Conclusioni: il confronto finale sui modelli ibridi proposti.....	85
Capitolo 5: Appendice: i comandi utilizzati su R.....	89
Paragrafo 5.1: Il modello ibrido HAR-ESN-PSO.....	90
Paragrafo 5.2: Il modello ibrido HEAVY-ESN-PSO.....	95
Paragrafo 5.3: Il modello ibrido HARQ-ESN-PSO.....	102
Bibliografia.....	109
Sitografia.....	111

INDICE DELLE FIGURE

Figura numero 125

Figura numero 228

Figura numero 330

Figura numero 432

Figura numero 535

Figura numero 636

Figura numero 740

Figura numero 843

Figura numero 947

Figura numero 1048

Figura numero 1149

Figura numero 1250

Figura numero 1354

Figura numero 1462

Figura numero 1562

Figura numero 1664

Figura numero 1765

Figura numero 1866

Figura numero 1967

Figura numero 2067

Figura numero 2168

Figura numero 2270

Figura numero 2372

Figura numero 2473

Figura numero 2573

Figura numero 2674

Figura numero 2775

Figura numero 2875

Figura numero 2976

Figura numero 3079

Figura numero 3179

Figura numero 32.....	80
Figura numero 33.....	81
Figura numero 34.....	81
Figura numero 35.....	82

INDICE DELLE TABELLE

Tabella numero 161
Tabella numero 2.....61
Tabella numero 3.....61
Tabella numero 4.....61
Tabella numero 5.....69
Tabella numero 6.....72
Tabella numero 7.....72
Tabella numero 8.....73
Tabella numero 9.....73
Tabella numero 10.....76
Tabella numero 11.....77
Tabella numero 12.....77
Tabella numero 13.....78
Tabella numero 14.....78
Tabella numero 15.....82
Tabella numero 16.....83
Tabella numero 17.....84
Tabella numero 18.....84

1 Introduzione alle serie storiche e ai dati ad alta frequenza

Negli ultimi anni i mercati finanziari sono stati caratterizzati da un crescente grado di complessità a seguito dell'esplosiva crescita di innumerevoli fattori di natura geopolitica, ambientale e macroeconomica. L'azione congiunta di queste forze ha alimentato un forte clima di incertezza nel quale gli operatori finanziari tuttora operano. L'elevato grado di incertezza è conseguenza delle marcate fluttuazioni di prezzo che hanno riguardato qualsiasi portafoglio d'investimento, che sia esso detenuto da un semplice privato o da un'istituzione finanziaria.

Pertanto, negli anni recenti si è assistito allo sviluppo di modelli e algoritmi di *machine learning* in grado di prevedere e quantificare l'intensità delle oscillazioni in termini di volatilità. Tali modelli e algoritmi permettono di analizzare la volatilità e misurare il grado di dispersione degli asset di un portafoglio oltre ad essere utilizzata in fase di valutazione delle performance.

La volatilità, a differenza dei rendimenti, non è direttamente osservabile, perciò è necessario ricorrere ad appositi indicatori atti a misurare la variabilità o l'incertezza del rendimento.

A seconda della natura dei rendimenti (giornalieri/settimanali/mensili) l'indicatore della volatilità assume diverse accezioni sulla base dell'arco temporale di riferimento.

Gli studi condotti affermano che all'interno dei mercati finanziari agiscono,

contestualmente, diverse componenti di volatilità conseguenti della presenza di fattori eterogenei quali la quantità di patrimonio da investire, attitudini al rischio, orizzonti temporali di investimento e la disponibilità nel reperire le informazioni.

Ai fini del presente elaborato verranno prese in considerazione differenti stimatori con lo scopo di modellare correttamente la volatilità finanziaria.

Gli stimatori adottati vertono sull'utilizzo di dati ad alta frequenza e pertanto offrono una misura della volatilità dei rendimenti circoscritti nell'arco di una giornata.

La *realized variance* è il primo degli stimatori che viene implementato nel modello proposto e che ha conosciuto un diffuso utilizzo per la sua semplicità e versatilità. Parallelamente si è deciso di ricorrere ad altri stimatori quali la *realized quarticity* e la *realized bipower variation* che offrono spunti diversi in base alla natura dei dati impiegati all'interno delle analisi condotte.

Questo elaborato si prefigge lo scopo di illustrare un modello ibrido, perfezionato sulla base di studi recenti, atto a fornire una previsione quanto più accurata e fedele a servizio degli operatori che intendono investire nei mercati finanziari.

Le recenti pubblicazioni in tema di previsione delle serie storiche finanziarie indicano come i modelli più efficienti e meno dispersivi siano le strutture ibride che combinano più componenti al loro interno. Un esempio è costituito dal modello ARIMA-RNN che combina un modello autoregressivo al fine di studiare la componente di linearità con una rete neurale ricorrente destinata all'analisi della componente non lineare. Adottando siffatti modelli è possibile sopperire alle carenze dei singoli modelli sfruttandone appieno i pregi e i punti di forza.

Per quanto concerne questo elaborato il modello ibrido di riferimento è l'HAR-ESN-PSO ed è stato proposto dai ricercatori Gabriel Trierweiler Ribeiro, André Alves Portela Santos, Viviana Cocco Mariani e Leandro dos Santos Coelho all'interno dell'articolo di ricerca "Novel hybrid model based on echo state neural network applied to the prediction of stock price return volatility" pubblicato nel 2021.

Da quanto emerge dall'articolo appena menzionato questa struttura ibrida risulta più accurata e precisa se raffrontata con modelli stocastici quali l'ARIMA e con altri modelli ibridi quali il GS-ARIMA.

Prima dello sviluppo delle reti neurali artificiali lo studio della volatilità veniva prettamente condotto attraverso modelli stocastici di matrice autoregressiva come l'ARIMA e modelli stocastici della classe GARCH.

I modelli ARCH e GARCH, rispettivamente, conosciuti come modello autoregressivo a eteroschedasticità condizionale e modello autoregressivo a eteroschedasticità condizionale generalizzato, sono stati proposti con lo scopo di modellare la volatilità nel tempo rappresentando in modo fedele il fenomeno del *volatility clustering*.

Il *volatility clustering* è un fenomeno che caratterizza le serie storiche finanziarie determinando periodi ad alta volatilità intervallati da periodi a volatilità modesta.

La classe dei modelli GARCH è piuttosto estesa e comprende svariate suddivisioni sulla base delle finalità di utilizzo e natura delle serie storiche implementate. Una prima distinzione può riferirsi ai modelli GARCH univariati e ai modelli GARCH multivariati, con quest'ultimi in grado di modellare e rappresentare le dipendenze tra le serie storiche finanziarie. Per quanto concerne la natura dei dati utilizzati è possibile distinguere alcune estensioni della famiglia dei modelli GARCH tra cui risaltano i modelli APARCH e T-GARCH.

I modelli appena citati affondano le proprie radici nell'utilizzo di dati a bassa frequenza, tipicamente giornalieri, tuttavia esistono modelli stocastici che utilizzano dati ad alta frequenza con osservazioni rilevate ogni 5 minuti, ogni minuto o addirittura ogni 30/10 secondi.

L'opinione condivisa dai ricercatori e studiosi della materia è che i modelli per la volatilità che ricorrono a dati ad elevata frequenza performano mediamente meglio dei modelli per la volatilità che utilizzano dati a bassa frequenza.

Questa proprietà dei modelli ad alta frequenza dei dati è riscontrabile nel momento in cui l'orizzonte di previsione è piuttosto breve, mentre per previsioni di medio periodo le differenze si assottigliano enormemente.

A questo fine, sono stati sviluppati modelli stocastici per la volatilità che ricorrono ai *dati ad alta frequenza*. Il novero di tali modelli comprende, tra gli altri, il modello HAR proposto da F. Corsi nel 2009 con lo scopo di modellare la volatilità giornaliera realizzata. Tale modello si basa sul principio del comportamento asimmetrico della volatilità dove la volatilità realizzata è funzione delle volatilità realizzate osservate a cadenza mensile, settimanale e giornaliera.

Il modello di Corsi è stato fonte di ispirazione per numerose pubblicazioni successive al fine di approfondire lo studio della volatilità associato ai *dati ad alta frequenza*. Tale pubblicazioni vertono sulla proposta di estensioni al modello HAR di F. Corsi.

La famiglia dei modelli stocastici per la volatilità che ricorrono ai dati ad alta frequenza

non si esaurisce con i modelli appena menzionati. Difatti, vi è il modello HEAVY, noto come high-frequency based volatility model, proposto da N. Shephard e K. Sheppard (2010). Tale modello rappresenta un'evoluzione alternativa ai modelli della classe GARCH, i quali non sono in grado di sfruttare i dati ad elevata frequenza.

Parallelamente, ai classici modelli statistici si sono affiancati algoritmi di *machine learning* più avanzati e complessi in grado di cogliere i profili di non linearità che i modelli statistici non riuscivano a sintetizzare. Tali algoritmi di *machine learning* sono noti come reti neurali artificiali e si ispirano all'articolazione e composizione dei neuroni e delle sinapsi del cervello umano.

Le reti neurali si compongono di uno strato di input, di uno o più strati nascosti e di uno strato di output. Ciascuno strato è composto da nodi e le reti neurali più semplici, le cosiddette *feed-forward*, prevedono che solamente i nodi di strati differenti siano collegati attraverso un'opportuna ponderazione dei pesi. In seguito viene applicata una funzione di attivazione, sovente si ricorre alla funzione *sigmoide* o, alternativamente, alla funzione rettificatore (*ReLU*), funzionale a definire l'output dei nodi sulla base dell'insieme degli input.

La funzione *sigmoide* schiaccia i valori di output in un intervallo 0,1, mentre la funzione *ReLU* assume il valore massimo tra 0 e x, graficamente analoga a una bisettrice che taglia il primo quadrante.

In ambito economico-finanziario le reti *feed-forward* non sono in grado di offrire una corretta e puntuale stima predittiva dei rendimenti futuri in quanto perdono la memoria di quanto accade nei nodi antecedenti. A causa di questa loro caratteristica sono state sviluppate le reti neurali ricorrenti le quali ricorrono ai dati dei nodi di input per influenzare l'input e output dei nodi successivi creando legami di dipendenza tra i nodi stessi.

Le *recurrent neural network* sono una classe di reti neurali che ha conosciuto una forte diffusione e utilizzo nell'ambito della previsione delle serie storiche finanziarie caratterizzate per la presenza di forme di dipendenza, anche non lineari.

Difatti, mentre le reti neurali *feed-forward* non hanno memoria degli ingressi effettuati in precedenza, le RNN sono caratterizzate da cicli ove il valore di uscita di un neurone costituisce il valore di entrata di un neurone nello strato precedente o del neurone stesso che ha prodotto quel valore.

L'ideazione e lo sviluppo delle reti neurali ricorrenti è stato profuso nel corso degli anni

per contrastare il fenomeno dei gradienti che “esplodono” o “svaniscono” tipico delle RNN con un’architettura complessa composta da numerosi *hidden layer*. Il primo fenomeno è riscontrabile nelle RNN dove si ricorre alla funzione *ReLU* in quanto il ciclo prevede che gli output dei nodi vengano moltiplicati costituendo gli input dei nodi precedenti. Valori elevati della funzione *ReLU* moltiplicati tra loro causano un’esplosiva crescita dei valori di input. Il fenomeno analogo, ma contrapposto avviene quando la funzione di attivazione è il *sigmoide*.

La classe delle RNN è ampia e annovera tra le altre le *long-short-term-memory NN*, le *gated recurrent units* e le *echo state network*, sulla quale si fondano i presenti studi.

La *echo state network* è una particolare tipologia di RNN dove è possibile individuare lo strato composto dai nodi di input, il “serbatoio” nel quale sono contenuti numerosi nodi scarsamente connessi tra loro e lo strato costituito dai nodi di output.

La *echo state network* ha trovato ampia diffusione nell’ambito della previsione delle serie storiche finanziarie in quanto risolvono i sopracitati problemi di *vanishing* e *exploding gradient* grazie alla loro struttura a serbatoio. Inoltre, solamente i pesi che collegano i nodi del serbatoio e i nodi dello strato di output sono addestrati dunque la fase di training è più celere e snella.

Ai modelli sopracitati sono affiancati degli algoritmi di ottimizzazione in grado di affinare gli iperparametri che intervengono all’interno degli stessi modelli proposti. Gli algoritmi di meta-euristica di ispirazione al mondo animale ne costituiscono una branca che ha riscontrato un forte successo nel corso del tempo. Tra i principali algoritmi di ottimizzazione vi è il PSO, *particle swarm optimization*, ispirato al comportamento e alle interazioni riscontrate negli sciami di insetti o negli stormi di uccelli per procacciare il cibo.

La soluzione dell’algoritmo è costituita da una particella che può muoversi nello spazio parametrico ed a ogni particella viene assegnata una fitness. Il procedimento è iterativo fintanto che non viene raggiunto il risultato migliore.

Il novero degli algoritmi di base meta-euristica di ispirazione al mondo animale è particolarmente nutrito e comprende, tra gli altri, il *Genetic Algorithm (GA)*, l’*Artificial bee colony (ABC)*, il *Grey Wolf Optimizer (GWO)* e via discorrendo.

Dunque, il modello HAR-ESN-PSO è frutto della sinergia di tre distinte componenti scelte per le loro potenzialità e capacità predittive.

L’obiettivo dell’elaborato è proporre un’evoluzione del modello, utile a offrire una valida

alternativa sulla base dei dati a disposizione per la propria analisi e viene proposto da con l'intento di non intervenire in modo drastico sull'impianto complessivo.

La rete neurale ESN ricopre un ruolo centrale nel modello ibrido in quanto è stata dimostrata la sua efficienza e potenzialità nella previsione di serie storiche finanziarie. L'intento è intervenire, in prima battuta, sul primo blocco del modello ibrido HAR-ESN-PSO, ossia sul modello HAR di F.Corsi (2009). In seguito alla pubblicazione del modello HAR si sono susseguiti numerosi studi e ricerche che hanno condotto allo sviluppo di numerose estensioni del modello stesso.

All'interno del suddetto elaborato si farà ricorso ad una di queste estensioni al fine di ottenere performance migliori riducendo il valore degli indicatori statistici di errore.

Tale estensione è conosciuta con l'acronimo di modello HARQ proposto da T. Bollerslev nel 2016.

Inoltre, la ricerca e le pubblicazioni nell'ambito dello studio della volatilità delle serie finanziarie non si è conclusa e nel corso degli anni recenti sono stati pubblicati modelli stocastici per la volatilità sulla scorta delle proprietà dei modelli della famiglia GARCH, tra i quali risalta il modello HEAVY.

Il modello HEAVY, il cui acronimo sta per high-frequency-based volatility, è un modello particolarmente indicato per la previsione della volatilità di serie storiche finanziarie con un orizzonte temporale breve. Difatti, evidenze empiriche confermano che ottiene risultati in media superiori rispetto ai modelli della famiglia GARCH.

Per quanto concerne l'aspetto di ottimizzazione degli iperparametri adottati all'interno della rete neurale ESN, l'obiettivo è proporre algoritmi di meta-euristica alternativi rispetto al *particle swarm optimization*. L'alternativa proposta ricade sulla scelta dell'algoritmo *artificial bee colony* (ABC), un metodo di ottimizzazione che si ispira al comportamento collaborativo instaurato dalle api di un alveare. La motivazione di questa scelta risiede nel fatto che l'algoritmo ABC riesce a offrire una soluzione valida in termini di performance se comparato con il PSO e altri algoritmi di ottimizzazione.

L'articolazione della struttura ibrida è piuttosto semplice. Innanzitutto, si ripartisce il periodo di riferimento del titolo in training set e validation set.

Il training set costituisce il set di osservazioni utili all'addestramento della rete neurale, mentre per valutare la reale capacità predittiva del modello si ricorre al validation set, un set di osservazioni che non vengono inserite nella rete, ma che rappresentano il metro di confronto con i risultati ottenuti.

Il modello stocastico per la volatilità prescelto è funzionale a determinare l'input dei dati da inserire nella *echo state network*. La quantità dei dati che verranno impiegati all'interno della rete neurale dipende dal numero di variabili indipendenti stimate all'interno dei singoli modelli per volatilità. In seguito attraverso l'utilizzo degli algoritmi di ottimizzazione vengono affinati gli iperparametri che governano la struttura della rete neurale al fine di raggiungere la migliore soluzione in termini di efficienza.

Infine, attraverso la fase di apprendimento vengono ricalibrati i pesi dell'output e si giunge al risultato finale.

I risultati delle varie combinazioni proposte sono raffrontati sulla base del benchmark costituito dal modello originale di riferimento. L'indicatore statistico di errore utilizzato nelle presente trattazione è l'errore quadratico medio, conosciuto come *mean squared error* (MSE). Il modello più efficiente e prestazionale è decretato sulla base del minore risultato ottenuto dell'indicatore MSE considerato.

2 La volatilità e i modelli per la volatilità

2.1 I dati ad alta frequenza

La presente trattazione verte sull'utilizzo e la trasformazione dei *dati ad alta frequenza* al fine di offrire delle soluzioni efficienti per la previsione della volatilità di serie storiche finanziarie.

La raccolta di *dati ad alta frequenza* ha conosciuto una rapida espansione nei recenti decenni a seguito dell'innovazione tecnologica e al diffuso accesso a macchine capaci di elaborare una quantità di dati superiore rispetto al passato in tempi di elaborazione più ridotti.

I *dati ad alta frequenza* consentono di raccogliere osservazioni infra-giornaliere nell'ordine dei minuti, dei secondi o delle singole transazioni. L'adozione di un dataset informativo composto da osservazioni infra-giornaliere presenta numerosi vantaggi rispetto ai dati a bassa frequenza, in particolare per uno studio finalizzato all'analisi della volatilità di serie storiche finanziarie.

Difatti, i *dati ad alta frequenza* consentono di tracciare con più precisione l'andamento delle serie storiche finanziarie, di osservare i repentini cambiamenti di prezzo e di catturare l'alternanza dei momenti di volatilità. Inoltre, l'elevata frequenza delle osservazioni permette di individuare con maggiore precisione eventuali salti di prezzo del titolo esaminato.

I salti di prezzo causano una discontinuità nell'andamento della serie storica finanziaria e avvengono a seguito di determinati shock quali la diffusione di notizie e informazioni rilevanti o cambiamenti nel sentiment di mercato.

Tuttavia, nelle serie finanziarie costituite dai *dati ad alta frequenza* si può manifestare una forma di rumore, chiamata *microstructure noise*, che dipende da diversi fattori tra i quali il *bid-ask bounce*¹, negoziazioni asincrone e poco presenti e lenta risposta dei prezzi dalle grandi transazioni. Il disturbo causato *microstructure noise* rende instabili le stime ad alta frequenza di alcuni parametri quali, ad esempio, la *realized variance*. Gli *effetti* del *microstructure noise* sulle serie storiche finanziarie può essere mitigato ricorrendo ad opportuni accorgimenti in sede di raccolta dei dati.

In primo luogo, selezionare un titolo azionario il cui mercato è particolarmente liquido limitando il *bid-ask bounce* e favorendo la frequenza continua delle osservazioni lungo il periodo analizzato. Inoltre, la qualità e la frequenza dei dati svolgono un ruolo cruciale in quanto disporre di dati con osservazioni ogni 1 o 5 minuti rispetto a set di osservazioni raccolte ogni 10 minuti permette di ridurre il rumore nella serie storica finanziaria. La motivazione risiede nel fatto che disponendo di un dataset a più elevata frequenza vi è una minore perdita di informazioni e conseguentemente un miglioramento dell'efficienza.

Inoltre, le rilevazioni non sono intese come una media ponderata delle osservazioni all'interno dell'intervallo di tempo considerato, bensì come singole osservazioni.

Dato il crescente utilizzo di *dati ad alta frequenza*, in letteratura sono stati proposti per la loro analisi modelli e algoritmi di *machine learning* diversi da quelli tradizionali utilizzati per i dati a bassa frequenza. L'avvicendamento è favorito dal largo e diffuso accesso ai *dati ad alta frequenza* a una considerevole percentuale di popolazione mediante piattaforme digitali. Il fine della trattazione è esporre dei modelli alternativi il cui utilizzo è destinato all'analisi dei dati ad alta frequenza rispettando le accortezze e la metodologia brevemente illustrata.

¹ Il bid-ask bounce indica un fenomeno dove il prezzo di un titolo rimbalza all'interno di un intervallo limitato tra il prezzo bid e il prezzo ask. Questo avviene quando avvengono scambi sia sul prezzo bid che sul prezzo ask, ma nessun movimento reale di prezzo. Inoltre, la definizione di bid-ask bounce va distinta dalla definizione di bid-ask spread, la quale rappresenta la differenza tra il prezzo bid e il prezzo ask.

2.2 Gli stimatori non parametrici della volatilità

La volatilità rappresenta la misura dell'incertezza circa i futuri movimenti del prezzo di un bene o di una attività finanziaria. "Al crescere della volatilità cresce la probabilità che la performance risulti molto elevata oppure molto contenuta, ossia cresce la probabilità che i movimenti di prezzo siano molto ampi, sia in aumento che in diminuzione"². Questa è la definizione fornita dal sito della Borsa Italiana.

Dal punto di vista statistico-econometrico il concetto di volatilità è legato alla deviazione standard, una metrica che misura il grado di dispersione di una variabile casuale attorno alla sua media. Tanto più la popolazione di dati di una variabile aleatoria è sparsa diffusamente intorno alla media tanto più la deviazione standard è elevata.

La definizione economica di volatilità fornita coincide con il concetto di volatilità storica e si contrappone alla volatilità implicita che rappresenta una misura delle aspettative del mercato relative alla volatilità futura.

Ai fini del presente elaborato viene presa in considerazione la definizione di volatilità storica.

La volatilità può essere stimata mediante numerosi indicatori in base alla metodologia adottata e ai fini del suo utilizzo.

Nella presente trattazione vengono esposti alcuni stimatori della volatilità che ricorrono all'utilizzo dei *dati ad alta frequenza*. Lo scopo di questi stimatori è calcolare la volatilità giornaliera sulla base dei *rendimenti intra-day*. Il novero di stimatori della volatilità che utilizzano high-frequency data è piuttosto ampio e ai fini della trattazione è stato ritenuto opportuno analizzare e implementare tre di questi stimatori, selezionati in base alle loro caratteristiche e pregi.

Le *realized measure* sono stimatori che ricostruiscono la volatilità *ex-post* sulla base dei rendimenti ad alta frequenza osservati in precedenza.

L'intuizione fondamentale fu fornita da Merton (1980) che notò che la varianza su un

² Definizione presa dal Glossario Finanziario a cura del sito di Borsa Italiana. Il riferimento è "<https://www.borsaitaliana.it/borsa/glossario/volatilita-.html>"

intervallo fisso può essere stimata abbastanza accuratamente come la somma dei rendimenti al quadrato, a condizione che i rendimenti siano disponibili con una frequenza di campionamento sufficientemente elevata. Dunque, con lo sviluppo e la diffusione di macchine sempre più potenti in grado di elaborare dati ad alta frequenza è stato possibile operare sotto le condizioni imposte dall'intuizione di Merton.

Dalla seconda metà degli anni '90 le *realized measure* fondate sui dati ad alta frequenza hanno conosciuto una vasta applicazione in materia.

Formalizzando in modo sintetico l'intuizione di Merton si supponga che il prezzo di un asset rispetti il seguente processo definito attraverso una *semi-martingala*=

$$dp(t) = \mu(t)dt + \sigma(t)dW(t) \quad (2.1)$$

Dove $W(t)$ rappresenta un *moto Browniano standard*.

Il rendimento continuo composto sull'intervallo temporale $t-k$ è il seguente=

$$R(t, k) = p(t) - p(t - k) = \int_{t-k}^t \mu(\tau)d\tau + \int_{t-k}^t \sigma(\tau)dW(\tau) \quad (2.2)$$

L'ultimo termine può essere interpretato come la radice quadrata della varianza integrata, ossia la *quadratic variation*.

Nel momento in cui il numero di osservazioni tende idealmente a infinito, ossia si hanno un numero particolarmente consistente di osservazioni giornaliere, allora lo stimatore della *realized variance* risulta essere uno stimatore robusto e consistente della *quadratic variation*.

E' opportuno segnalare che ai fini della fattibilità del presente elaborato il numero di osservazioni giornaliere si limita a rilevazioni di prezzo ogni 5 minuti.

La prima *realized measure* sistematicamente studiata e impiegata allo scopo di modellare e prevedere la volatilità è stata la *realized variance*, proposta da T.G. Andersen, L. Benzoni e L. Lund nel 1998³.

³ T.G Andersen, L. Benzoni e L. Lund sono gli autori dell'articolo "Estimating Jump-Diffusions for Equity Returns" pubblicato nel 1998. Tale pubblicazione approfondisce i processi di salto, una tipologia di processo stocastico caratterizzato da movimenti discreti, e le serie storiche finanziarie. Forniscono per la prima volta la definizione di *realized variance*.

La *realized variance*, abbreviata RV, è uno stimatore calcolato attraverso la somma dei rendimenti infra-giornalieri elevati al quadrato come riportato dalla seguente formula.

$$RV_{t,k} = \sum_{j=1}^m R_{t-k+j/m}^2, \quad j = 1, \dots, m, \quad (2.3)$$

Dove m rappresenta il numero di osservazioni intraday.

La *realized variance* è nota per la sua semplicità interpretativa nonché per la sua versatilità. Sovente, tale stimatore è stato utilizzato anche per il calcolo della volatilità per periodi temporali più ampi, ad esempio una settimana o un mese, ricorrendo ai dati a bassa frequenza.

La *realized variance* ha sperimentato una vasta applicazione in ambito di modellazione e previsione di serie storiche finanziarie, in particolare all'interno dei modelli della classe HAR, predisposti per lo studio e analisi della volatilità.

In presenza di salti (o *jumps*) nella serie storica finanziaria la *quadratic variation* non corrisponde perfettamente alla radice della varianza integrata.

La QV è data dalla seguente formula=

$$QV_{t,k} = \int_{t-k}^t \sigma^2(\tau) d\tau + \sum_{t-k \leq \tau \leq t} J^2(\tau) \quad (2.4)$$

Nell'equazione 2.4 il simbolo $J^2(\tau)$ rappresenta il termine che misura la magnitudo dei salti di prezzo. Tale termine è elevato alla seconda data la natura positiva dei salti di prezzo.

Sebbene la *realized variance* risulti essere uno stimatore robusto, presenta dei limiti legati all'eventuale presenza di *price jumps* all'interno della serie storica finanziaria. Difatti, le manifestazioni di salti di prezzo nella serie storica implicano che la definizione di *quadratic variation* non è più perfettamente sovrapponibile al concetto di *integrated variance*.

A questo fine è stato proposto dapprima lo stimatore della *realized power variation* e in seguito la sua evoluzione, la *realized bi-power variation*, che ha riscontrato un diffuso successo e una vasta applicazione in materia di modellazione della volatilità.

La *realized bi-power variation* è uno stimatore della volatilità proposto da O. E.

Barndorff-Nielsen e N. Shephard nel 2004⁴ allo scopo di ottenere una misura della volatilità robusta ai salti di prezzo e che fosse in grado di quantificare l'entità dei salti stessi.

La formula della *realized bi-power variation* è così definita=

$$BV(t, k, h, m) = \frac{\pi}{2} \sum_{j=h+1}^{mk} \left| R\left(t - k + \frac{jk}{m}, \frac{1}{m}\right) \right| \left| R\left(t - k + \frac{(j-h)k}{m}, \frac{1}{m}\right) \right| \quad (2.5)$$

Lo stimatore risulta essere robusto rispetto ai salti di prezzo in quanto considera i rendimenti tra periodi consecutivi riducendo l'impatto dei salti isolati e inoltre incorpora la varianza integrata, una misura della persistenza della volatilità nel tempo. Inoltre, è possibile determinare in termini quantitativi l'entità dei salti di prezzo attraverso la differenza tra la serie della *realized variance* e la serie della *realized bi-power variation*. La seguente formula formalizza il concetto appena espresso=

$$RV(t, k, m) - BV(t, k, m) = \sum_{t-k \leq \tau \leq t} J^2(\tau) \quad (2.6)$$

L'analisi di una serie storica finanziaria non può prescindere dalla valutazione dei salti di prezzo relativi alla medesima serie pertanto è uno stimatore che ha conosciuto un diffuso successo nei modelli che utilizzano high-frequency data.

La *realized quarticity* (RQ) è la terza *realized measure* della volatilità presente all'interno dei modelli proposti. Alla stregua delle precedenti due misure della volatilità, anche la *realized quarticity* risulta essere uno stimatore consistente della volatilità.

La *realized quarticity* giornaliera si ottiene attraverso la somma dei *rendimenti intraday* elevati alla quarta come suggerito dalla formula 2.7=

$$RQ_t = \frac{M}{3} \sum_{i=1}^M r_{t,i}^4 \quad (2.7)$$

⁴ O.E. Barndorff-Nielsen e N. Shephard sono gli autori della pubblicazione "Realized power variation and stochastic volatility models" rilasciata nel 2003. Tale articolo di ricerca approfondisce la presenza dei salti all'interno dei processi stocastici sulla scorta delle teorie di Merton (1980) relative al pricing delle opzioni. Contestualmente, i due ricercatori hanno proposto lo stimatore della realized power variation.

La *realized quarticity* è stata proposta allo scopo di stimare con precisione la *integrated quarticity (IQ)*, definita all'interno della formula 2.8.

$$IQ = \int_0^1 \sigma_u^4 du \quad (2.8)$$

La formula dell'*integrated quarticity* richiama la definizione di curtosi, la cui formula è definita dalla seguente specificazione=

$$\text{Curtosi in eccesso} = E \left[\left(\frac{x-\mu}{\sigma} \right)^4 \right] - 3 \quad (2.9)$$

Difatti, la *realized quarticity* è legata al momento quarto della distribuzione normale, la *curtosi*.

La *curtosi* è un indice che offre la misura dello “spessore” delle code di una funzione di densità. Tale indice riveste un'importanza centrale all'interno dello studio di distribuzioni casuali in quanto lo “spessore” delle code influenza il comportamento di numerose statistiche.

Analogamente, i medesimi ragionamenti sono applicati nell'ambito dell'analisi di serie storiche finanziarie.

E' possibile distinguere tre differenti casistiche. Per quanto concerne le serie storiche finanziarie caratterizzate da un valore positivo della *curtosi* è riscontrabile una concentrazione delle osservazioni intorno alla sua media. La funzione di distribuzione gaussiana è definita *leptocurtica*, la cui conformazione della curva risulta essere molto appuntita con code particolarmente snelle.

Parimenti, alcune serie storiche finanziarie presentano un valore negativo della *curtosi* determinando una distribuzione poco incentrata intorno alla sua media e caratterizzata dalla presenza di code molto spesse e pesanti. La curva della funzione di distribuzione è definita *platicurtica* e risulta più piatta e schiacciata rispetto alla tipica curva a campana di una distribuzione normale.

A completamento, il coefficiente di *curtosi* pari a zero indica una distribuzione definita *normocurtica*, ossia caratterizzata attraverso la curva con configurazione a campana tipica della distribuzione gaussiana. I valori sono normalmente concentrati intorno alla loro media. Quest'ultima eventualità è la meno comune per quanto concerne la

distribuzione di variabili aleatorie legate a serie storiche finanziarie.

La *realized quarticity* è stata implementata e affiancata alla *realized variance* sfruttando l'alta correlazione tra i processi di RV e RQ. La presenza di correlazione positiva tra la volatilità e la volatilità della volatilità è stata documentata anche da parte di J. Da Fonseca e W. Zhang in "Volatility of volatility is (also) rough".

2.3 Il modello HAR-RV di Corsi

Il modello HAR è stato sviluppato e proposto da Fulvio Corsi nel 2009⁵ con lo scopo di prevedere e modellare il comportamento della volatilità giornaliera realizzata, ottenuta attraverso lo stimatore della *realized variance*.

Il modello HAR è funzione di tre distinte variabili: la *realized variance* osservata al tempo $t-1$, dunque al tempo immediatamente precedente, il valore atteso della *realized variance* settimanale e il valore atteso della *realized variance* mensile.

L'intuizione economica sottostante la scelta delle tre variabili indipendenti segue una precisa logica che governa i mercati finanziari.

Numerosi studi e ricerche condotte nel corso degli anni hanno evidenziato che nei mercati finanziari agiscono diversi fattori eterogenei.

Queste ricerche sono attribuibili a autori quali Peters (1994) in "Fractal Market Analysis"⁶, Lux e Marchesi (1999), Alfarano e Lux (2007), LeBaron (2006).

Questi fattori eterogenei sono riconducibili alla quantità di patrimonio da investire, attitudine al rischio, collocazione geografica e orizzonte temporale d'investimento.

In questa sede particolare importanza riveste il concetto di orizzonte temporale di

⁵ Fulvio Corsi è l'autore dell'articolo scientifico "A Simple Approximate Long-Memory Model of Realized Volatility" pubblicato nel 2009. Tale articolo è incentrato sulla proposta del modello HAR e sulle sue numerose proprietà. Difatti, lo stesso autore F. Corsi afferma che il modello HAR è in grado di riprodurre le principali caratteristiche empiriche dei rendimenti finanziari (memoria a lungo termine, code grasse e autosimilarità) in modo parsimonioso e efficace.

⁶ "Fractal Market Analysis", pubblicato nel 1994, è un libro a cura di E. Peters. Il libro fornisce numerosi strumenti utili all'analisi del comportamento e dei movimenti dei prezzi dei titoli azionari.

investimento in quanto i diversi operatori del mercato finanziario possono effettuare investimenti nel breve, nel medio o nel lungo periodo.

All'interno di un contesto così variegato, ove operano individui e enti con finalità e orizzonti temporali diversi, si creano tre distinte componenti di volatilità: di breve, medio o lungo termine causate da investimenti, rispettivamente, giornalieri, o più frequenti, settimanali e di una o più mensilità.

La presenza di tre distinte componenti temporali di volatilità ha alimentato ulteriori studi e ricerche in merito con lo scopo di definire le relazioni e gli effetti sugli investimenti.

I risultati ottenuti da Müller et al. (1997), Arneodo, et al. (1998), Gençay e Selçuk (2004) e Lynch e Zumbach (2003) hanno dimostrato la volatilità a lungo termine ha un'influenza maggiore della volatilità a breve termine. Pertanto il comportamento della volatilità viene definito asimmetrico.

Dal punto di vista economico tali risultati trovano fondamento in quanto le scelte di investimento con un orizzonte temporale breve vengono condizionate tanto dalla volatilità a breve termine quanto dalla volatilità a lungo termine che offre un'importante indicazione dei rischi e della variazioni prezzi futuri, mentre coloro che investono a lungo termine ignorano la volatilità a breve termine.

Per comprendere la specificazione del modello HAR è necessario introdurre il concetto di volatilità latente parziale non osservata $\tilde{\sigma}_t^{(c)}$ generata da una determinata componente di mercato.

Coerentemente con quanto esposto in precedenza il modello HAR di Corsi fa riferimento alla natura asimmetrica della volatilità e pertanto la volatilità parziale non osservata $\tilde{\sigma}_t^{(c)}$ viene valutata in funzione della sua realizzazione al tempo precedente, t-1, e al valore atteso delle volatilità parziali settimanale e mensile.

Le componenti mensile, settimanale e giornaliera sono specificate nelle seguenti formule.

$$\tilde{\sigma}_{t+1m}^{(m)} = c^{(m)} + \varphi^{(m)}RV_t^{(m)} + \tilde{\omega}_{t+1m}^{(m)} \quad (2.10)$$

$$\tilde{\sigma}_{t+1w}^{(w)} = c^{(w)} + \varphi^{(w)}RV_t^{(w)} + \gamma^{(w)} E_t[\tilde{\sigma}_{t+1m}^{(m)}] + \tilde{\omega}_{t+1w}^{(w)} \quad (2.11)$$

$$\tilde{\sigma}_{t+1d}^{(d)} = c^{(d)} + \varphi^{(d)}RV_t^{(d)} + \gamma^{(d)} E_t[\tilde{\sigma}_{t+1w}^{(w)}] + \tilde{\omega}_{t+1d}^{(d)} \quad (2.12)$$

$RV_t(d)$, $RV_t(w)$ e $RV_t(m)$ rappresentano, rispettivamente, la *realized variance giornaliera*, la *realized variance settimanale* e la *realized variance mensile*.

$\tilde{\omega}_{t+1m}^{(m)}$, $\tilde{\omega}_{t+1w}^{(w)}$ e $\tilde{\omega}_{t+1d}^{(d)}$ rappresentano le tre componenti di innovazione, o *shock*, della volatilità.

Inoltre, affinché sia garantita la validità del modello le tre forme di *realized variance* devono essere contemporaneamente e serialmente indipendenti con media pari a zero e varianza positiva tali da soddisfare il vincolo di positività della volatilità parziale.

Operando una serie di sostituzioni si giunge alla seguente formula.

$$\tilde{\sigma}_{t+1d}^{(d)} = c + \beta^{(d)}RV_t^{(d)} + \beta^{(w)}RV_t^{(w)} + \beta^{(m)}RV_t^{(m)} + \tilde{\omega}_{t+1d}^{(d)} \quad (2.13)$$

Attraverso l'equazione soprastante viene espresso un modello per la volatilità a tre fattori stocastici corrispondenti alle tre *realized variance* di orizzonti temporali distinti.

Tuttavia va applicata un'ulteriore sostituzione per pervenire alla formulazione definitiva del modello HAR.

Il processo della volatilità latente risulta definito dalla seguente specificazione=

$$\tilde{\sigma}_{t+1d}^{(d)} = RV_{t+1d}^{(d)} + \omega_{t+1d}^{(d)} \quad (2.14)$$

Dunque, sostituendo il termine $\tilde{\sigma}_{t+1d}^{(d)}$ all'interno della formula numero 13 si perviene all'equazione formale del modello HAR di Corsi=

$$RV_{t+1d}^{(d)} = c + \beta^{(d)}RV_t^{(d)} + \beta^{(w)}RV_t^{(w)} + \beta^{(m)}RV_t^{(m)} + \omega_{t+1d} \quad (2.15)$$

$$\text{dove } \omega_{t+1d} = \omega_{t+1d}^{(d)} + \tilde{\omega}_{t+1d}^{(d)} \quad (2.16)$$

Il modello HAR è ascrivibile alla classe dei modelli autoregressivi (o AR) fintanto che può essere rappresentato attraverso un modello AR di ordine 22.

Inoltre, il modello HAR ha riscontrato popolarità e successo presso gli studiosi della materia per merito della sua semplicità applicativa e della sua parsimonia.

Il generale apprezzamento per il modello sviluppato da F. Corsi ha alimentato ulteriori studi e approfondimenti della materia da parte degli studiosi e dei ricercatori conducendo a numerose estensioni del modello originale

Le successive estensioni proposte si avvalgono di *realized measure* differenti, mentre parallelamente sono stati proposti modelli HAR con triplete di valori della RV che si discostano dai canonici valori 1,5 e 22 proposti da Corsi.

All'interno della presente trattazione il modello HAR è utilizzato al fine di stimare le tre differenti *realized variance* distinte per orizzonte temporale di riferimento. Le tre *realized variance* costituiranno le variabili di input della *echo state network*, la rete neurale artificiale prescelta per i modelli ibridi proposti.

2.4 Il modello HAR-Q di Bollerslev

L'HARQ model, proposto da T. Bollerslev nel 2016, è ascrivibile al novero di estensioni del modello HAR di Corsi proposte e pubblicate negli ultimi anni.

La struttura del modello HARQ prevede le medesime variabili presenti all'interno del modello originale di F. Corsi. Dunque, sono presenti le tre variabili definite come *realized variance* giornaliera, *realized variance* settimanale e *realized variance* mensile.

La logica e l'intuizione economica sottostante al modello HARQ è la medesima del modello HAR di Corsi. L'elemento di novità introdotto all'interno di questa estensione riguarda lo stimatore della *realized quarticity*.

Esistono due distinte versioni del modello di Bollerslev (2016).

La prima versione conosciuta come full HARQ model o HARQ-F model, la cui specificazione è la seguente=

$$RV_t = \beta_0 + (\beta_1 + \beta_{1Q}\sqrt{RQ_{t-1}^d})RV_{t-1}^d + (\beta_2 + \beta_{2Q}\sqrt{RQ_{t-1}^w})RV_{t-1}^w + (\beta_3 + \beta_{3Q}\sqrt{RQ_{t-1}^m})RV_{t-1}^m + u_t \quad (2.17)$$

Sono evidenti le analogie con il modello di F. Corsi date dalla presenza della RV

giornaliera, RV settimanale e RV mensile.

Inoltre, T. Bollerslev ha proposto una versione semplificata particolarmente adatta per le previsioni a breve termine, la cui formula è definita=

$$RV_t = \beta_0 + (\beta_1 + \beta_{1Q}\sqrt{RQ_{t-1}^d})RV_{t-1}^d + \beta_2RV_{t-1}^w + \beta_3RV_{t-1}^m + u_t. \quad (2.18)$$

Tale versione semplificata è stata proposta a seguito delle difficoltà di stima dei coefficienti β_Q relativi agli orizzonti temporali superiori, ossia settimanale e mensile.

Inoltre, l'errore di stima relativo alle variabili della *realized variance* settimanale e della *realized variance* mensile è sensibilmente inferiore rispetto a quello relativo alla *realized variance* giornaliera. Dunque, la sola presenza della RQ giornaliera è in grado di attenuare considerevolmente l'errore di previsione del modello HARQ.

Pertanto, ai fini della presente trattazione verrà utilizzata quest'ultima versione del modello HARQ in quanto, sovente, i coefficienti relativi alle *realized quarticity* di orizzonti temporali superiori risultano non significativi. In aggiunta, è opportuno precisare che il termine β_{1Q} è presumibilmente negativo. Questo comporta una minore reazione (persistenza) rispetto alla *realized variance* a un ritardo.

In sintesi, il modello HARQ è stato proposto da T. Bollerslev col fine di calmierare una problematica legata al tradizionale modello HAR di Corsi. Difatti, lo stesso economista danese ha dimostrato che i coefficienti delle tre variabili del modello HAR contengono errori di misurazione che possono condurre a previsioni non puntuali della volatilità. Tuttavia è opportuno precisare che all'aumentare dell'orizzonte di previsione tale errore viene gradualmente mitigato.

Attraverso l'introduzione di una quarta variabile, ossia la *realized quarticity*, legata al momento quarto della distribuzione dei rendimenti, il modello è in grado di ridurre l'errore di stima e condurre a previsioni più accurate.

All'interno dello stesso articolo T. Bollerslev ha condotto numerose simulazioni con differenti orizzonti temporali di previsione e ha dimostrato la superiorità del modello HARQ.

Per quanto concerne il modello ibrido adottato nella presente trattazione le quattro variabili, ossia la RV1, RV5, RV22 e la RQ1 costituiranno i dati di input da immettere all'interno della *echo state network*.

2.5 Il modello HEAVY di Sheppard e Shephard

Il modello High-frequency based Volatility, abbreviato con il termine HEAVY, è stato sviluppato e proposto da Sheppard e Shephard⁷ nel 2010 con lo scopo di realizzare un modello stocastico per la previsione della volatilità alternativo alla famiglia dei modelli GARCH. Un modello appartenente alla classe GARCH è un modello autoregressivo generalizzato atto a catturare i raggruppamenti di volatilità dei rendimenti giornalieri attraverso la varianza condizionata.

Si individui con $\varepsilon_t = h_t \eta_t$ la notazione dei rendimenti corretti per la media (o *demeaned returns*), allora un modello GARCH(m,s) è definito attraverso la seguente specificazione.

$$h_t^2 = \omega + \sum_{i=1}^m \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^s \beta_j h_{t-j}^2 \quad (2.19)$$

Dove ω rappresenta la varianza condizionata.

Il termine h_t è strettamente positivo, mentre i coefficienti alfa e beta sono positivi e la loro somma deve risultare inferiore a uno, condizione necessaria affinché la varianza condizionata sia finita e positiva.

Inoltre, la risultanza della somma dei coefficienti di alfa e beta viene definita persistenza di h_t .

Il modello GARCH è in grado di rappresentare il fenomeno di *volatility clustering*.

Dal punto di vista formale questo fenomeno è spiegato dal fatto che sebbene i rendimenti logaritmici siano *incorrelati* tra loro, per converso le loro trasformazioni positive (il valore assoluto e il quadrato) presentano autocorrelazione positiva e lentamente decrescente.

⁷ N. Shephard e K. Sheppard sono gli autori dell'articolo di ricerca "Realising the future: forecasting with high-frequency-based volatility (HEAVY) models" pubblicato nel 2010. La pubblicazione è incentrata sulla proposta del modello HEAVY e sul suo comportamento con riferimento agli effetti della crisi creditizia del 2008. Infine, i risultati ottenuti attraverso il modello HEAVY sono stati raffrontati con i risultati ottenuti dai modelli GARCH tradizionali.

Valori elevati di ε_t o di h_t inducono valori elevati di h_t^2 e di conseguenza di ε_t^2 innescando un processo “esplosivo”.

Il *volatility clustering* è un fenomeno riscontrabile nelle serie storiche finanziarie per il quale a periodi caratterizzati da volatilità piuttosto elevata si alternano periodi dove la volatilità assume valori modesti.

Il modello HEAVY è una derivazione del modello della classe GARCH con il quale condivide numerosi aspetti, nonché le medesime intuizioni di natura economica e statistica. Pertanto questa sintetica esposizione si focalizzerà sui profili che contraddistinguono i due modelli piuttosto che sugli elementi in comune.

Innanzitutto, il modello HEAVY differisce dai modelli della classe GARCH per la tipologia dei dati utilizzati.

I modelli GARCH ricorrono all'utilizzo di dati a bassa frequenza raccolti, tipicamente, giornalmente, mentre il modello HEAVY si avvale delle *realized measure*, stimatori non-parametrici della volatilità calcolati *ex-post* sulla base dei dati ad alta frequenza.

Il novero di *realized measure* utilizzate all'interno del modello sviluppato da Sheppard e Shephard comprende, tra gli altri, la *realized variance* e la *realized bi-power variation*, stimatori illustrati nei paragrafi precedenti. Come affermato dagli stessi autori il modello HEAVY è stato costruito con l'intento di proporre un modello per la previsione della volatilità capace di tenere in considerazione fenomeni tipici dei mercati finanziari quali la *mean reversion* e il *momentum*.

La *mean reversion* si riferisce a una teoria economica che afferma che la volatilità dei prezzi di un titolo e i relativi rendimenti si riallineeranno alla media osservata nel lungo periodo.

Il *momentum* è un fenomeno riscontrabile nei mercati finanziari per il quale i titoli in rapida crescita sono destinati a salire, mentre i titoli poco appetibili continuano a perdere. La strategia sul *momentum* suggerisce di acquistare i titoli in fase di rapida ascesa e vendere i titoli la cui crescita è lenta, o ,addirittura, negativa. L'orizzonte temporale di tale strategia è circoscritto a un periodo piuttosto breve.

A suffragio di quanto esposto, Sheppard e Shephard hanno dimostrato empiricamente che il modello HEAVY è in grado di performare meglio rispetto ai modelli GARCH su orizzonti temporali molto brevi, tipicamente della durata di qualche giorno, mentre su orizzonti più estesi i due modelli hanno capacità predittive analoghe.

Il modello HEAVY si dimostra un modello a risposta relativamente breve in grado di

performare meglio in periodi brevi dove il livello della volatilità o di correlazione della stessa sono soggetti a bruschi cambiamenti. Difatti, è noto per la sua adattabilità alle rapide e improvvise variazioni di volatilità nel breve termine.

I dati di input necessari per la costruzione del modello HEAVY sono due:

- i rendimenti giornalieri, indicati con la notazione r_1, r_2, \dots, r_T ;
- la corrispondente sequenza di *realized measure* giornaliere indicate con la notazione RM_1, RM_2, \dots, RMT .

Ovviamente la scelta della *realized measure* da adottare è condizionata anche dal fenomeno del *microstructural noise*, una forma di rumore tipica dei dataset con osservazioni ad alta frequenza.

Il modello è costituito da un sistema definito da una duplice equazione. Le due equazioni sono stimate separatamente all'interno del modello come affermato dagli stessi autori, K. Sheppard e N. Shephard.

Le due equazioni che costituiscono il modello HEAVY richiamano fortemente la struttura dei modelli della famiglia GARCH=

$$E[r_t^2 | \mathcal{F}_{t-1}^{HF}] := h_t = c_h + b_h h_{t-1} + a_h v_{t-1}; \quad (2.20)$$

$$E[v_t | \mathcal{F}_{t-1}^{HF}] := m_t = c_m + b_m m_{t-1} + a_m v_{t-1}. \quad (2.21)$$

Dove la notazione \mathcal{F}_{t-1}^{HF} indica il set di dati ad alta-frequenza.

La prima equazione viene definita "*variance equation*", la quale utilizza la *realized measure* come dato di input per stimare h_t , ossia la varianza condizionata. Parallelamente, il modello GARCH utilizza i rendimenti quadratici per stimare h_t . E' stato osservato e provato che le *realized measure* forniscono una stima più precisa e puntuale della volatilità rispetto ai rendimenti quadratici giornalieri.

La seconda equazione è conosciuta come "*realized measure equation*" ed è utilizzata per la previsione multi-step di m_t , la quale rappresenta una misura della varianza condizionata.

Nel presente elaborato, la varianza condizionata riferita alla seconda equazione del sistema del modello HEAVY costituirà la variabile di input da immettere nella rete artificiale ESN.

2.6 Le echo state network (ESN)

La *echo state network* (ESN) è una rete neurale artificiale facente parte della classe delle reti neurali ricorrenti conosciute per la loro efficacia in ambito di previsione di serie storiche finanziarie. Inoltre, la ESN presenta una qualità peculiare nel panorama delle reti neurali ricorrenti: data la sua natura di rete neurale artificiale è considerato un modello non-lineare, tuttavia gli unici pesi modificati durante la fase di addestramento riguardano le connessioni con i neuroni facenti parte lo strato di output. Per tale ragione, la funzione di errore è quadratica rispetto al vettore dei parametri e può essere facilmente differenziata in un sistema lineare.

Alla luce di quanto descritto, si evince la versatilità e la semplicità applicativa delle *echo state network*.

2.6.1 Breve introduzione alle reti neurali ricorrenti

Innanzitutto, le reti neurali artificiali sono modelli computazionali composti da “neuroni” artificiali che richiamano la struttura di una rete neurale biologica.

La prima e la più semplice tipologia di reti neurali proposta e applicata è la rete neurale *feed-forward* ove le connessioni tra i nodi non formano cicli. Le informazioni fluiscono verso una sola direzione, dai nodi d’ingresso passano attraverso i nodi nascosti fino ai nodi d’uscita.

Le reti neurali *feed-forward* non hanno memoria degli input avvenuti a tempi precedenti pertanto l’output è determinato dall’attuale input. Questa caratteristica delle RN *feed-*

forward implica la loro inadeguatezza a modellare e prevedere serie storiche finanziarie, ove le osservazioni sono caratterizzate da un legame di dipendenza temporale.

Al fine di modellare e simulare il comportamento di serie storiche è stata concepita e sviluppata una variante delle reti neurali artificiali, le reti neurali ricorrenti, o *recurrent neural network*.

Le *recurrent neural network* è una rete neurale costituita da cicli ove i valori di un nodo di uno strato di livello superiore, più vicino all'uscita, sono utilizzati come input di un nodo di uno strato inferiore, vicino all'ingresso. Dunque, si distinguono per la loro "memoria" in quanto le informazioni degli input precedenti influenzano i valori di input e output attuali.

Questa tipologia di reti è indicata per lo studio di dati sequenziali o serie temporali, caratterizzate da legami di dipendenza temporale.

Ovviamente, la struttura delle RNN si fonda sui medesimi presupposti di una rete neurale *feed-forward*. Ciascuno strato è composto da neuroni e ciascuno dei neuroni è tipicamente collegato a tutti i neuroni dello strato successivo (e antecedenti data la ciclicità delle connessioni della RNN) attraverso una connessione pesata. Il neurone è il risultato della somma ponderata di tutti i neuroni a esso collegato a cui si aggiunge un *bias*.

Infine, il valore viene riparametrizzato all'interno di un intervallo di valori, tipicamente tra 0 e 1 o tra -1 e 1, attraverso una funzione detta di attivazione che può assumere diverse forme tra cui la *ReLU* e la funzione *sigmoidale*.

L'immagine seguente sintetizza la struttura di una semplice RNN composta dallo strato di input, lo strato nascosto e lo strato di output.

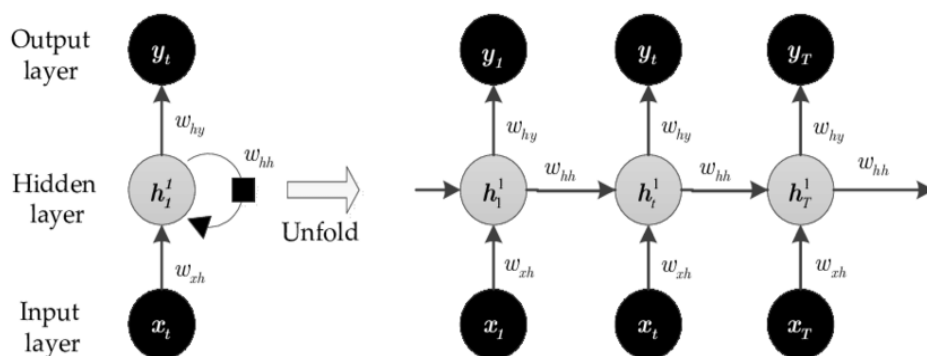


Figura n.1: Esempio di rete neurale ricorrente

Fonte: "Day-Ahead Solar Irradiance Forecasting for Microgrids Using a Long Short-Term Memory Recurrent Neural Network: A Deep Learning Approach"

E' possibile osservare come l'output dello strato nascosto vada a costituire l'input dello stesso determinando una ciclicità dei dati. A destra viene raffigurata la RNN nella sua forma "distesa".

Al fine di ottenere un risultato quanto più in linea a quello desiderato le reti neurali artificiali sono sottoposte a un addestramento attraverso l'utilizzo di appositi algoritmi. Uno degli algoritmi di apprendimento più diffusi è la retropropagazione dell'errore (in inglese *backpropagation*) che ha lo scopo di modificare i pesi sinaptici della rete in modo da avvicinare i valori di output ai valori attuali.

Per le RNN si utilizza una variante dell'algoritmo di retropropagazione dell'errore, conosciuto come *backpropagation through time* (BPTT). A seconda del numero dei passaggi temporali tale algoritmo può risultare piuttosto oneroso dal punto di vista computazionale. Difatti, l'algoritmo BPTT ricorre all'utilizzo di derivate per calcolare e accumulare gli errori per ogni fase temporale per poi aggiornare i pesi iterando questo processo finché l'output della rete non è prossimo al risultato desiderato.

L'addestramento di una RNN con una mole di dati molto elevata può sostanziarsi in un processo piuttosto lento, oltre che generare del rumore.

Attraverso un semplice ciclo è possibile consentire ai neuroni della rete di avere memoria degli ingressi passati e quindi di modellare il comportamento dinamico temporale dipendente dalle informazioni ai tempi precedenti.

Le reti neurali ricorrenti sono note per la loro vasta applicazione in ambito di previsione di serie storiche, anche dati storiche di diversa natura.

La loro efficacia in ambito di previsione di serie finanziarie ha contribuito notevolmente alla letteratura in materia. Ad esempio, è possibile addestrare la rete utilizzando *time series* con una sequenza di dati nota fino al tempo t per prevedere poi i possibili risultati al tempo $t+1$, $t+2$ e via scorrendo.

Le prime RNN hanno sperimentato una problematica legata ai valori dei gradienti ottenuti dalle funzioni di attivazione. Tale fenomeno riguarda la fase di addestramento col metodo della *back-propagation* ove vengono aggiornati i parametri dei neuroni (pesi e *bias*) in modo proporzionale alla derivata parziale della *loss-function* rispetto al parametro stesso.

In base alle diverse tipologie di funzioni d'attivazione implementate all'interno della rete si può manifestare il problema del gradiente che scompare (*vanishing gradient*) o del

gradiente che esplode (*exploding gradient*).

Il primo caso riguarda una funzione di tipo sigmoideale o tangente iperbolica il cui valore dei gradienti è compreso nell'intervallo 0-1. L'algoritmo di BPTT prevede che i gradienti vengano moltiplicati in successione seguendo i *layer* e risulta evidente che il prodotto di un considerevole numero di gradienti, il cui valore è inferiore a 1, porta il valore complessivo a diminuire rapidamente lungo la catena di neuroni.

Il secondo caso si manifesta con le funzioni della tipologia *ReLU* o lineare pura, ove il valore dei gradienti può essere superiore a 1. Il prodotto di una serie di valori superiori a 1 genera un risultato complessivo che cresce enormemente lungo la catena dei neuroni.

Nel corso degli anni sono state sviluppate numerose tipologie di reti neurali ricorrenti con l'obiettivo di arginare e limitare gli effetti dei fenomeni appena descritti.

Da una struttura semplice e molto intuitiva delle prime RNN si è giunti a strutture molto articolate, complesse e onerose a livello computazionale come le *long-short term memory* (LSTM) e le *gated recurrent unit* (GRU). Queste tipologie di reti neurali ricorrenti rappresentano un'evoluzione delle prime reti ricorrenti, tuttavia è stato osservato che quando la mole di dati di input è molto significativa è possibile che emergano problematiche legate al *vanishing gradient*.

Parallelamente, si è affermata una metodologia alternativa per la progettazione e l'allenamento delle reti neurali ricorrenti: il *reservoir computing*.

In sostanza, il *reservoir computing* distingue tra una parte ricorrente (o dinamica), tipica delle reti neurali ricorrenti, e una parte non ricorrente. L'elemento di dinamicità è costituito dal *reservoir* o serbatoio che contiene al suo interno un elevato numero di neuroni connessi tra loro.

2.6.2 La struttura delle echo state network

Le *echo state network* rappresentano una specifica tipologia di reservoir computing che ha conosciuto un diffuso successo in ambito di previsione di serie storiche per merito della loro efficacia e semplicità di calcolo. La prima versione di *echo state*

network è stata proposta da Jaeger nel 2002⁸, mentre solamente nel 2004 è stata sperimentata in ambito di previsioni di serie storiche, con un preciso focus sulle serie storiche caratterizzate dalla presenza di componenti di caos e rumore.

Al pari delle reti LSTM e GRU, la ESN non soffre della problematica legata ai fenomeni del *vanishing* ed *exploding gradient*.

Inoltre, la ESN è nota per avere una fase di addestramento particolarmente snella e poco dispendiosa a livello computazionale, in antitesi rispetto alle articolate e complesse reti neurali ricorrenti brevemente esposte in precedenza.

La struttura di una *echo state network* standard è sintetizzata nelle seguenti immagini.

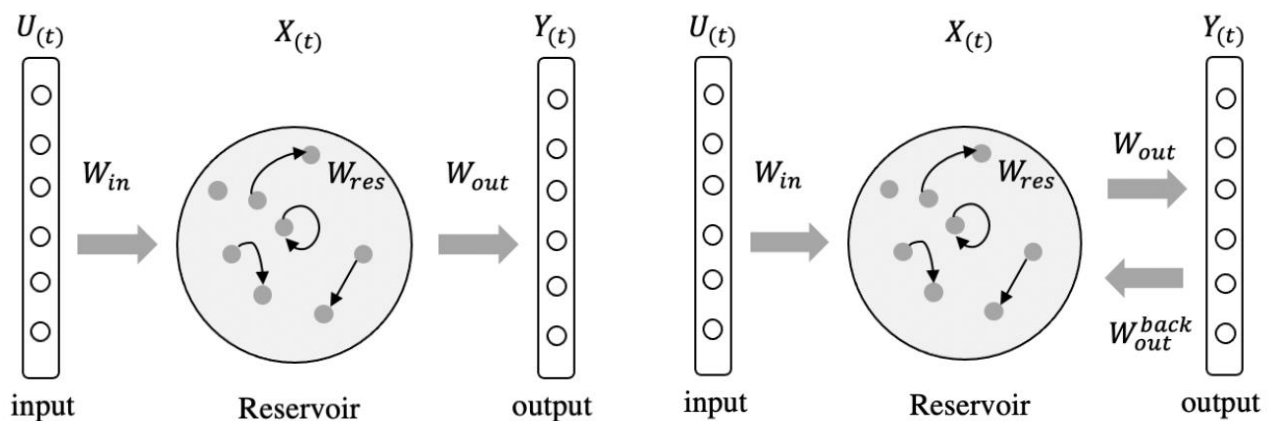


Figura n.2 , la struttura della echo state network standard.

Fonte: "A Review of Designs and Applications of Echo State Networks"

L'immagine a destra rappresenta una prima integrazione della *echo state network* ove si distinguono due distinte tipologie di connessioni tra serbatoio e strato di output.

Da quanto si evince dall'immagine a sinistra, $u(t)$ rappresenta i valori di input della rete al tempo t . La notazione $x(t)$ rappresenta lo stato del serbatoio al tempo t , il quale è composto da un elevato numero di nodi connessi in modo sparso e casuale tra loro.

Infine, $y(t)$ indica lo strato di output della rete al tempo t .

Le funzioni di transizione di stato sono definite dalle equazioni 2.22 e 2.23, dove f rappresenta una funzione non lineare, tipicamente una *sigmoide* o una *tan*.

⁸ Herbert Jaeger è l'autore dell'articolo scientifico "The "echo state" approach to analysing and training recurrent neural networks-with an erratum note" rilasciato nel 2002. La pubblicazione fornisce un'approfondita e dettagliata analisi delle echo state network, delle condizioni sufficienti e necessarie affinché sia garantita l'intera struttura e della fase di addestramento.

La prima funzione fa riferimento alle connessioni pesate stabilite all'interno del serbatoio, o *reservoir*, mentre la seconda alle connessioni pesate tra il serbatoio e lo strato di output, conosciuto come *readout*.

$$x(t) = f(W_{in}u(t) + W_{res}x(t - 1)) \quad (2.22)$$

$$y(t) = W_{out}x(t) \quad (2.23)$$

Le connessioni stabilite tra i diversi *layer* sono classificabili nell'ordine di tre (quattro) distinte tipologie.

Le connessioni pesate stabilite tra i differenti *layer* sono esprimibili attraverso delle matrici di diverse dimensioni.

Innanzitutto, con $W_{in} \in R^{N \times D}$ si riferisce alla connessione tra lo strato di input e il serbatoio, mentre $W_{res} \in R^{N \times N}$ rappresenta le connessioni stabilite tra i nodi del serbatoio. Tipicamente, i nodi del serbatoio di una *ESN* sono scarsamente connessi, nell'ordine del 10% circa.

I pesi delle connessioni appena descritte sono casuali, entro determinate condizioni, e definiti durante la fase di inizializzazione della rete. Durante l'addestramento della rete tali pesi non vengono allenati, alla stregua di parametri fissi.

W_{out} rappresenta le connessioni intercorrenti tra il serbatoio e l'output *layer*.

La figura 2, a destra, specifica due distinti flussi di connessione stabiliti tra il serbatoio e lo strato di output. Tali flussi sono definiti attraverso le notazioni di W_{out}^{back} e di W_{out} .

Il termine W_{out}^{back} rappresenta le connessioni pesate che collegano lo strato di output al serbatoio della rete. Questi pesi sono definiti durante la fase di inizializzazione della rete. I pesi denominati W_{out} sono calcolati durante la fase di training della rete e connettono il serbatoio allo strato di output. L'implementazione di connessioni pesate W_{out}^{back} favorisce la stabilità complessiva della rete.

La fase di *training* viene espletata ricorrendo ad algoritmi molto snelli e celeri quale, ad esempio, la *ridge regression*, una forma di regressione lineare atta a stimare i parametri di modelli costituiti da variabili indipendenti altamente correlate tra loro.

La fase di inizializzazione della rete è raffigurata dalla figura 3.

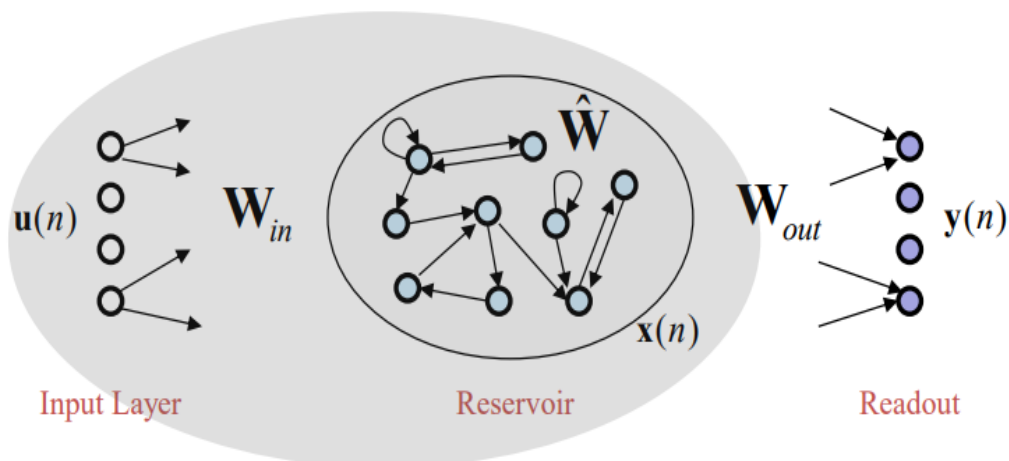


Figura n.3: la fase di inizializzazione della echo state network
 Fonte: "An introduction to echo state network"

Per quanto concerne la simbologia adottata: la notazione W_{res} relativa all'immagine 2 è equivalente alla notazione \hat{W} relativa all'immagine 3, mentre il termine (t) dell'immagine 2 è equivalente al termine (n) dell'immagine 3. Quest'ultimi termini si riferiscono agli stati dei layer che compongono la rete. Nel presente paragrafo si utilizza esclusivamente la notazione W_{res} al fine di riferirsi alle connessioni pesate che riguardano i neuroni presenti all'interno del serbatoio.

Nella fase di inizializzazione è necessario determinare il valore degli iperparametri della rete, ossia parametri non allenabili che regolano il processo di addestramento.

I principali iperparametri coinvolti sono tre:

- L'*input-scaling*, il cui simbolo è w^{in} che definisce l'intervallo di W_{in} all'interno del quale gli elementi in $[-w^{in}, w^{in}]$ sono comunemente inizializzati;
- Il *coefficiente* α che rappresenta lo sparsity parameter di W_{res} , ossia quanto "scarsamente" sono connessi i pesi nel serbatoio. Restituisce la proporzione degli zeri all'interno della matrice dei pesi del serbatoio.
- Lo *spectral radius* dei W_{res} . Pertanto, W_{res} è inizializzato partendo da una matrice \hat{W}_{random} dove gli elementi di \hat{W}_{random} sono generati casualmente nell'intervallo $[-1, 1]$.

Oltre, ai suddetti iperparametri sono definiti il coefficiente λ_r , l'input bias e il *settling time*.

Il valore degli iperparametri può essere definito arbitrariamente sulla base dell'esperienza e dei pregressi studi in materia. Ad esempio, l'articolo "Effects of *spectral radius* and *settling time* in the performance of *echo state networks*" pubblicato dal ricercatore Venayagamoorthy et al.⁹ è incentrato sulla ricerca del valore ottimale dello *spectral radius* w^{in} . A seguito di numerosi tentativi condotti è stato individuato un valore ottimale dello *spectral radius* pari a 0.8 in grado di fornire le migliori performance predittive.

Tuttavia, è frequente che gli iperparametri vengano affinati ricorrendo ad algoritmi di ottimizzazione quali, ad esempio, il *Genetic Algorithm* e il *Particle swarm optimization* facenti parte della classe degli algoritmi evolutivi.

La procedura di inizializzazione prevede che la matrice dei pesi del serbatoio W_{res} sia inizializzata sulla base di una matrice \hat{W}_{random} generata in modo casuale. Successivamente la matrice viene ridimensionata per rispettare le condizioni imposte dalla ESP, ossia la proprietà fondamentale delle *echo state network*, oggetto di approfondimento nelle pagine successive.

La matrice W_{res} è generata attraverso la seguente formula=

$$W_{res} = \hat{W}_{random} \frac{\rho_{desired}}{\rho(\hat{W}_{random})} \quad (2.24)$$

La fase di *training* coinvolge esclusivamente le connessioni pesate W_{out} come rappresentato dalla figura 4.

⁹ G. K. Venayagamoorthy è l'autore dell'articolo "Effects of spectral radius and settling time in the performance of echo state networks" pubblicato nel 2009. Tale articolo analizza l'importanza dello spectral radius e del parametro del settling time, nonché la loro influenza sull'intera struttura della echo state network. Gli autori eseguono numerosi esperimenti giungendo a una duplice conclusione. Lo spectral radius di 0.8 fornisce le migliori performance predittive e al contempo elevati valori del settling time tendono a inficiare negativamente sulle prestazioni della rete neurale.

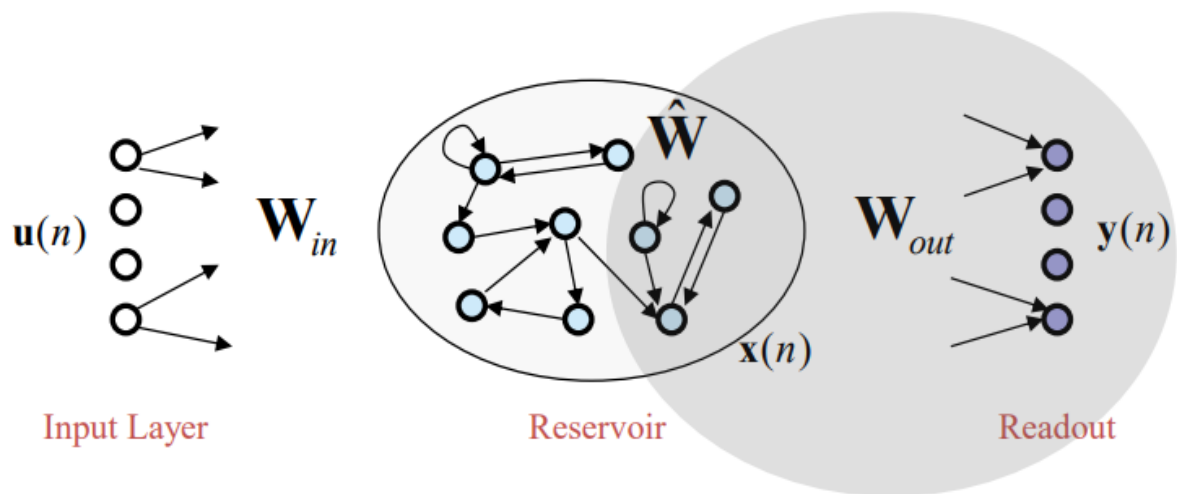


Figura n.4, la fase di apprendimento della echo state network.

Fonte: "An introduction to echo state network"

L'obiettivo della fase di *training* è minimizzare il valore della funzione di costo che esprime la differenza tra il risultato previsto e quello effettivo, ossia desiderato. Innanzitutto, si raccolgono gli stati del serbatoio e i valori effettivi. I dati sono raccolti per ciascuno degli n considerati.

$$\mathbf{X} = [\mathbf{x}(1) \dots \mathbf{x}(N)] \quad \mathbf{Y}_{target} = [\mathbf{y}(1) \dots \mathbf{y}(N)] \quad (2.25)$$

Si esegue l'addestramento delle connessioni pesate tra il serbatoio e il *readout*. La seguente funzione rappresenta la ricerca del minimo tra i valori stimati dalla rete e i valori target di riferimento.

$$\min \|\mathbf{W}_{out} \mathbf{X} - \mathbf{Y}_{target}\|_2^2 \quad (2.26)$$

La fase di *training* viene espletata, come menzionato in precedenza, prediligendo la scelta di algoritmi non dispendiosi e dalla semplice e intuitiva applicazione. I due algoritmi più diffusi sono la *pseudo-inversa* di Moore-Penrose e la *ridge regression*.

L'algoritmo che ricorre alla *pseudo-inversa* di Moore-Penrose è definito dalla seguente specificazione=

$$\mathbf{W}_{out} = \mathbf{Y}_{target} \mathbf{X}^+ \quad (2.27)$$

La *pseudo-inversa* è rappresentata da \mathbf{X}^+ , ossia la generalizzazione della matrice inversa di \mathbf{X} , quest'ultima definita come matrice non quadrata.

La *ridge regression* è sintetizzata nella formula 2.28=

$$\mathbf{W}_{out} = \mathbf{Y}_{target} \mathbf{X}^T (\mathbf{X} \mathbf{X}^T + \lambda_r \mathbf{I})^{-1} \quad (2.28)$$

Dove λ_r rappresenta il *coefficiente di regolarizzazione* (tipicamente inferiore a 1).

Entrambi gli algoritmi proposti sono ascrivibili alla categoria dell'*offline training*, la più comune nell'ambito dell'addestramento di *echo state network*.

Oltre alle tecniche esposte, esistono altri algoritmi di *training* attraverso i quali può essere condotto l'addestramento della rete.

Difatti, è possibile definire la categoria dell'*online training*, un metodo di apprendimento automatico che permette di lavorare con dati disponibili in ordine sequenziale aggiornando dinamicamente il migliore predittore disponibile. Questa tipologia di training è meno frequente e comprende, tra le altre, tecniche quali l'algoritmo dei minimi quadrati ricorsivi.

L'analisi della struttura della *echo state network* non può prescindere dal concetto di *echo state property* e delle sue implicazioni. Innanzitutto, la *echo state property* rappresenta la condizione fondamentale per garantire la stabilità e la funzionalità dell'intera struttura ricorrente e dinamica.

L'ESP descrive che gli stati del serbatoio dovrebbero dipendere asintoticamente solo dal segnale in ingresso derivante dallo strato di input e contemporaneamente l'effetto delle condizioni iniziali dovrebbe svanire progressivamente nel tempo.

Jaeger, all'interno delle sue ricerche, afferma che ogni vettore \mathbf{x} dello stato del serbatoio è determinato in modo univoco per ogni sequenza di input u . Questo implica che le *echo state* vicine hanno input simili.

La definizione precisa e puntuale di *echo state property* recita:

una ESN con un'equazione di transizione di stato F rispetta la *echo state property* se per ogni sequenza di input $U = [u(1), u(2), \dots, u(N)]$ e per tutte le coppie di stati iniziali x, x' , essa soddisfa la condizione definita dall'equazione 2.29.

$$\|F(U, x) - F(U, x')\| \rightarrow 0 \quad \text{as } N \rightarrow \infty \quad (2.29)$$

L'esistenza della ESP è verificata da una duplice condizione, una necessaria e una sufficiente. La condizione necessaria riguarda lo *spectral radius* $\rho(W_{res})$, mentre la condizione sufficiente afferisce al valore $\bar{\sigma}(W_{res}) = \|W\|_2 =$

$$\text{echo state property} \Rightarrow \rho(W_{res}) < 1 \quad (2.30)$$

$$\bar{\sigma}(W_{res}) < 1 \Rightarrow \text{echo state property} \quad (2.31)$$

Recenti studi condotti in materia di *echo state network*, hanno definito e stabilito l'esistenza di condizioni supplementari.

Ad esempio, il ricercatore Michael R. Buehner¹⁰, all'interno delle sue pubblicazioni, ha definito un'ulteriore condizione al fine di garantire la stabilità asintotica della rete.

La condizione è rappresentata dalla seguente specificazione=

$$\bar{\sigma}(D(W_{res}D^{-1})) < 1 \Rightarrow \text{echo state property} \quad (2.32)$$

2.6.3 Una breve panoramica sulle numerose estensioni della ESN

La letteratura in materia di *reservoir computing* è in continua evoluzione e numerose estensioni della ESN standard sono state proposte nel corso degli anni.

Una semplice *echo state network* dall'architettura alternativa è rappresentata da una rete con connessioni dirette tra lo strato di input e quello di output come raffigurato all'interno della figura 5.

¹⁰ M. Buehner e P. Young sono gli autori dell'articolo scientifico "A tighter bound for the echo state property" pubblicato nel 2006. Tale articolo propone un'ulteriore e rigorosa condizione volta a favorire la stabilità asintotica della echo state network. I risultati sono stati dimostrati attraverso una serie di simulazioni.

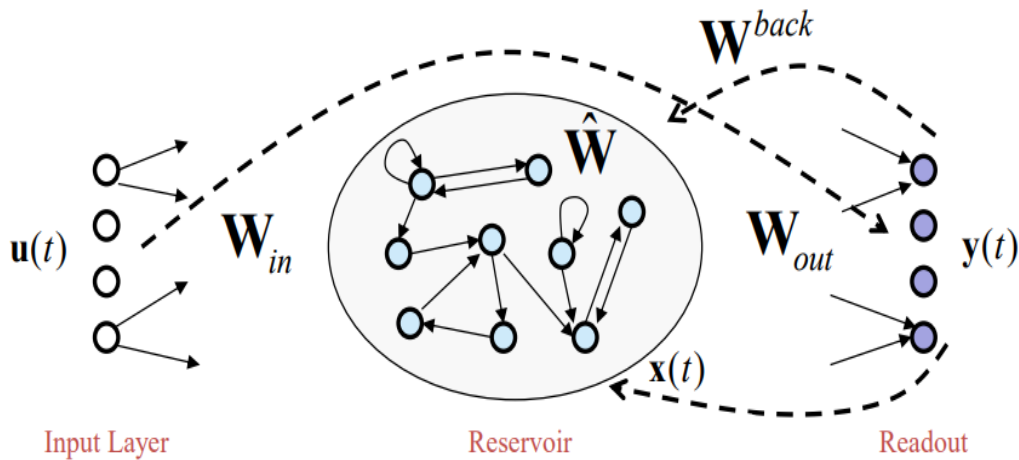


Figura n.5, una semplice estensione della echo state network.
 Fonte: "An introduction to echo state network"

In questa configurazione, tra il serbatoio e lo strato di output viene stabilita una connessione pesata per favorire la funzionalità della rete.

In seguito alle prime pubblicazioni di Harbert Jaeger, colui che ha proposto le *echo state network* nel 2002, sono state condotte ulteriori ricerche in materia e proposte delle architetture alternative alla prima ESN. I numerosi lavori sono incentrati sul design del serbatoio, considerato l'elemento caratterizzante dell'intera architettura della rete.

Parte delle ricerche hanno approfondito la connettività e la modularità del serbatoio, mentre parallelamente sono stati condotti studi sull'addestramento delle connessioni pesate relative ai neuroni presenti nel serbatoio. Le numerose pubblicazioni hanno contribuito ad arricchire la letteratura in materia di echo state network. Tra le proposte più interessanti e di successo emerge la *critical echo state network* (CESN) di Ajnal et al. Tale rete neurale artificiale reimposta la definizione di W_{res} in \hat{W}_{res} .

\hat{W}_{res} incorpora le matrici delle connessioni pesate W_{in} e W_{out} .

$$\hat{W}_{res} = W_{res} + W_{in}W_{out} \quad (2.33)$$

Di seguito, due immagini che illustrano, a scopo esemplificativo, due architetture alternative sviluppate sulla base della *echo state network* di Jeager (2002).

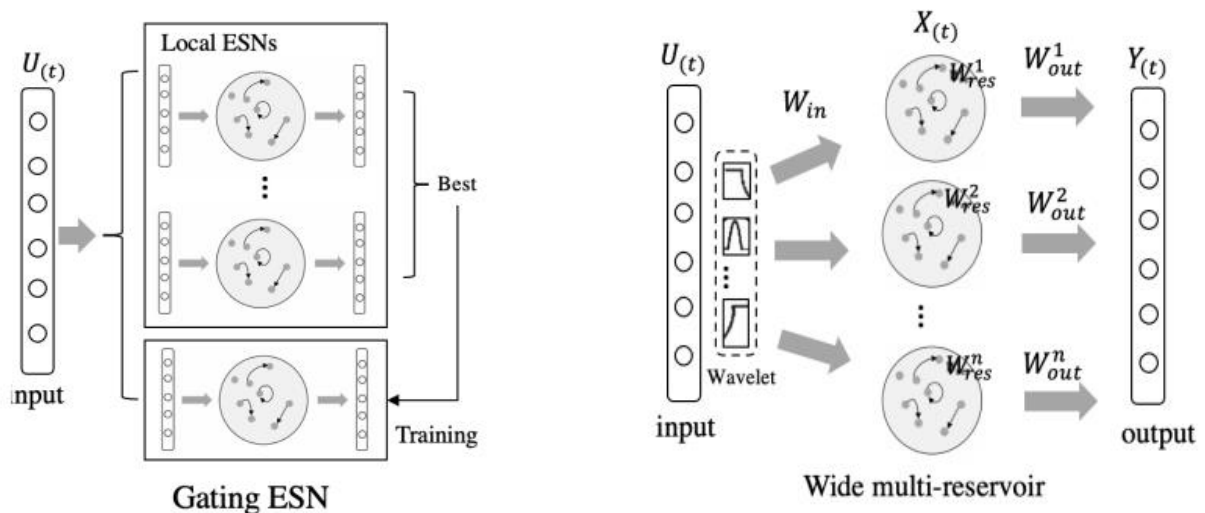


Figura n.6: la Gating ESN e la ESN multi-serbatoio.
 Articolo "A Review of Designs and Applications of Echo State Networks"

L'immagine 6, a sinistra, raffigura una *echo state network* integrata con un meccanismo di *gating*. Tale estensione è stata proposta da Babinec et al. combinando varie *local ESN* a cui sono associate differenti *spectral radius*. La *local ESN* che consegue il risultato migliore è utilizzata per addestrare il modulo del *gating ESN*.

La figura 6, a destra, rappresenta una rete costituita da più serbatoi, distinti per periodo temporale. L'output $Y(t)$ è ottenuto dai molteplici serbatoi, dal tempo 1 a t . Tuttavia, questa estensione presenta dei limiti a causa della natura lineare dell'output *layer* che non permette di esplorare pienamente le informazioni contenute nei serbatoi.

Infine, lo sviluppo del *deep learning* ha contribuito all'evoluzione della materia del *reservoir computing*. Gli studi condotti hanno portato allo sviluppo delle cosiddette *Deep-ESN*, delle architetture molto complesse che si articolano in più *echo state network* sfruttando le innovazioni del *deep-learning* e della *stacking architecture*. Difatti, le *Deep-ESN* sono state proposte col fine di combinare i punti di forza delle *ESN* e del *deep learning*.

Trattasi di una materia molto complessa che non viene affrontata all'interno del presente elaborato, tuttavia è utile per comprendere come l'universo del *reservoir*

computing si stia progressivamente integrando con le recenti innovazioni in ambito di intelligenza artificiale e machine learning.

Ai fini della presente trattazione i modelli ibridi implementati vertono sulla *echo state network* standard, dunque la versione proposta da Jaeger nel 2002.

La scelta ricade su questa soluzione in quanto rappresenta un'ottima soluzione per lo studio e l'analisi di serie storiche finanziarie oltre che per la tecnologia a disposizione in termini di software e hardware.

Nel caso del modello ibrido HAR-ESN i dati di input si riferiscono alle tre variabili *realized variance* a un giorno, a una settimana e un mese. Dunque, $u(t)$ è costituita dalle tre variabili al tempo t . E' possibile distinguere T stati sulla base del numero di ritardi considerati.

La scelta della *echo state network* è ricaduta sulla configurazione "base", raffigurata dall'immagine 3 del presente capitolo.

Con $Y(t)$ viene rappresentato il valore della *realized variance*, ossia la misura da stimare attraverso questo modello ibrido. L'allenamento della rete coinvolge solo le connessioni pesate contraddistinte come w_{out} . La previsione viene condotta in seguito all'addestramento fornendo i dati di input delle tre variabili menzionate con lo scopo di modellare il comportamento della *realized variance* su più orizzonti temporali.

Infine, la selezione degli iperparametri da adottare è eseguita ricorrendo ad algoritmi di ottimizzazione tra i quali vi è il *particle swarm optimization*.

2.7 Gli algoritmi di ottimizzazione evolutivi

L'algoritmo di ottimizzazione è una procedura logico-matematica di risoluzione di un problema di ottimizzazione. All'algoritmo di ottimizzazione spetta il compito di trovare una o più soluzioni al problema di ottimizzazione di riferimento. L'efficacia dei suddetti algoritmi dipende da due definizioni fondamentali.

La robustezza di un algoritmo di ottimizzazione rappresenta la capacità di trovare il

massimo assoluto della funzione obiettivo.

L'accuratezza di un algoritmo di ottimizzazione è intesa come la capacità di avvicinarsi al vero valore del massimo assoluto della funzione obiettivo.

E' solito distinguere gli algoritmi di ottimizzazione sulla base dell'adozione o meno del gradiente della funzione obiettivo. La presenza del gradiente favorisce l'aspetto dell'accuratezza, per converso l'assenza del gradiente implica una maggiore robustezza. La scelta dell'algoritmo è condizionata dalla tipologia del problema oggetto di ottimizzazione.

I problemi di ottimizzazione riguardanti i parametri di reti neurali artificiali trovano, tipicamente, soluzione attraverso l'adozione degli algoritmi evolutivi, una famiglia di algoritmi di ispirazione al fenomeno dell'evoluzione delle specie. Tali algoritmi non prevedono l'adozione del gradiente, sebbene nel recente passato siano stati proposti innovativi algoritmi ibridi che combinano i punti di forza degli algoritmi evolutivi alla funzionalità del gradiente.

Ai fini del presente elaborato vengono presi in considerazione solamente alcuni algoritmi evolutivi che non prevedono l'adozione del gradiente.

2.7.1 L'algoritmo del particle swarm optimization (PSO)

L'algoritmo *particle swarm optimization* conosciuto come PSO è un algoritmo di ottimizzazione la cui applicazione è estesa a vari campi, in particolare all'intelligenza artificiale. Rientra nella famiglia degli algoritmi evolutivi, ossia algoritmi euristici che si ispirano al principio dell'evoluzione delle specie e del regno animale. Inoltre, tali algoritmi si basano su popolazioni iniziali casuali (*soluzioni candidate*) che mutano e migliorano ogni generazione successiva, lavorando verso una soluzione da inserire nell'altro.

L'algoritmo del PSO è stato proposto da due famosi sociobiologi, Kennedy ed Eberhart, nel 1995. Come rivelato nella loro ricerca gli uccelli appartenenti a uno stormo si muovono in gruppo e sono in grado di condividere le esperienze con gli altri membri allo scopo di ottimizzare la caccia.

Dunque, il PSO simula il comportamento degli stormi di uccelli in cerca di cibo.

E' definito algoritmo euristico in quanto la soluzione fornita non è dimostrabile essere

la soluzione ottimale globale che non può essere individuata. Sovente, la soluzione fornita dal PSO si avvicina all'ottimo globale.

Ogni soluzione proposta (una combinazione di parametri) è una particella che può muoversi all'interno di uno *spazio parametrico*. Per ciascuna particella si valuta un valore un fitness.

L'algoritmo prevede due distinte fasi. La prima fase viene definita di inizializzazione, mentre la seconda di aggiornamento, iterata fino al raggiungimento dell'ottimo o di un criterio d'arresto.

Innanzitutto, per ogni particella $\mathbf{x}_i : i = 1, \dots, S$

La fase di inizializzazione è articolata in quattro distinti passaggi:

1. Si definisce la distribuzione di ciascuna particella (solitamente si ricorre a una distribuzione uniforme);
2. Si calcola la fitness di ciascuna particella;
3. Si assegna al personal best la fitness calcolata allo step precedente
4. Si assegna la migliore posizione in base alla *fitness* $\mathbf{g} = \max_i(\mathbf{p}_i)$ in seguito si inizializza la velocità di ciascuna particella

$$\mathbf{v}_i \sim U(-|\mathbf{b}_{up} - \mathbf{b}_{lo}|, |\mathbf{b}_{up} - \mathbf{b}_{lo}|)$$

La fase di aggiornamento consta di sei ulteriori passaggi per ogni particella e per ogni parametro $\mathbf{x}_i : i = 1, \dots, S$ rispetto al $d : d = 1, \dots, m$ quale si cerca l'ottimo:

1. Si generano due numeri casuali;
2. Si aggiorna la velocità della particella i-ma rispetto al parametro d ;

$$v_{i,d} = wv_{i,d} + \phi_p r_p (p_{i,d} - x_{i,d}) + \phi_g r_g (g_d - x_{i,d}) \quad (2.34)$$
3. Si aggiorna la posizione della particella;
4. Si calcola la *fitness* $f(\mathbf{x}_i)$ della particella nella nuova posizione;
5. Se $f(\mathbf{x}_i) > f(\mathbf{p}_i)$ allora $\mathbf{p}_i \leftarrow \mathbf{x}_i$ si aggiorna la migliore posizione della particella i-esima, mentre se $f(\mathbf{x}_i) > f(\mathbf{g})$ allora $\mathbf{g} \leftarrow \mathbf{x}_i$ la migliore posizione globale è già aggiornata;

6. Si ripetono tutti i passaggi precedenti della fase di aggiornamento fino al raggiungimento del criterio d'arresto.

Di seguito, un semplice diagramma di flusso che riepiloga in modo sintetico e fruibile i macro-passaggi in cui si snoda l'algoritmo PSO.

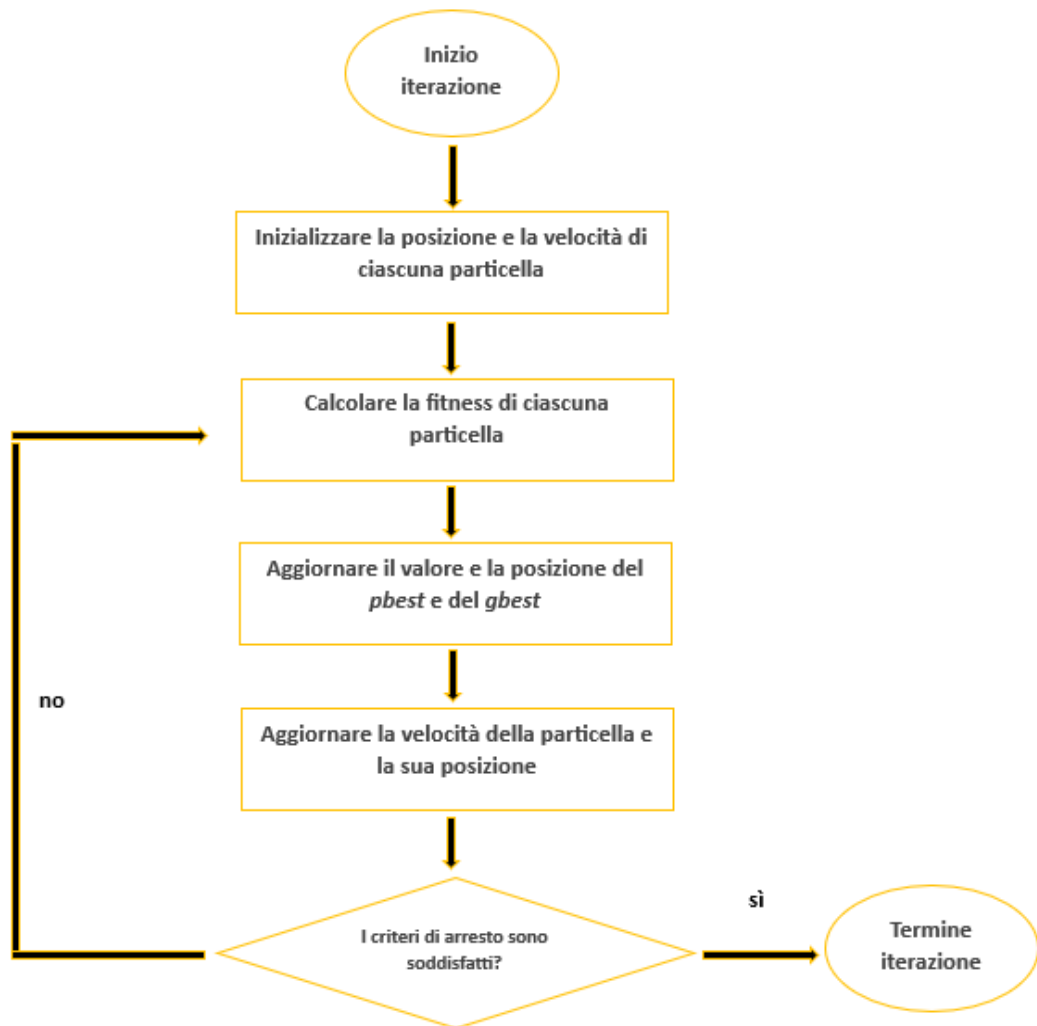


Figura n.7: diagramma di flusso raffigurante l'algoritmo PSO
Fonte: Rielaborazione personale

Nella presente trattazione l'algoritmo del *particle swarm optimization* è stato implementato allo scopo di ottimizzare gli iperparametri che definiscono la *echo state network*.

Il risultato fornito da tale algoritmo dà luogo a valori stocastici, dunque è opportuno

ripetere il procedimento al fine di giungere a un sufficiente numero di prove che rendono attendibili i risultati del processo di ottimizzazione.

Nel contesto dei modelli ibridi esposti tale algoritmo è stato utilizzato con lo scopo di affinare gli iperparametri che caratterizzano la *echo state network* quali, ad esempio, il *leaking rate* e lo *spectral radius*.

2.7.2 L'algoritmo dell'artificial bee colony (ABC)

L'*artificial bee colony* è un recente algoritmo di ottimizzazione proposto da Basturk e Karaboga nel 2007. Al pari del PSO rientra nella famiglia degli algoritmi evolutivi di ispirazione al regno animale.

L'algoritmo è noto per la sua estrema flessibilità di applicazione in numerosi ambiti, oltre che per ricorrere a meno parametri di controllo rispetto ad altri algoritmi quali il PSO e il GA. Infine, è conosciuto per la sua robustezza e per l'efficacia nell'ambito di funzioni di costo di tipo stocastico.

L'algoritmo ABC si ispira all'intelligente comportamento collaborativo adottato dalle api mellifere all'interno dell'alveare e cerca di trovare soluzioni ottimali a problemi di ottimizzazione complessi. Tali problemi di ottimizzazione riguardano diversi ambiti spaziando dall'ingegneria fino all'economia.

All'interno dell'algoritmo della colonia d'api, le fonti alimentari rappresentano la soluzione del problema. Ogni fonte produce nettare in misura proporzionale alla fitness.

L'algoritmo ABC distingue tra tre differenti tipologie di api. A ciascuna tipologia vengono attribuiti compiti e funzioni ben distinte.

Innanzitutto, le api operaie alla ricerca di nuove fonti di cibo con più nettare rispetto a quelle precedenti. La ricerca è casuale, ma sfrutta le informazioni delle altre fonti alimentari. Le api spettatrici osservano i movimenti e i viaggi delle api operaie e apprendono le posizioni delle fonti di cibo, ossia le soluzioni del problema. Infine, le api esploratrici scelgono le fonti di cibo in modo casuale per sostituire le fonti di cibo abbandonate che non possono essere migliorate su un numero determinato di percorsi.

Le fasi in cui si articola l'algoritmo ABC richiamano la sequenzialità dell'algoritmo dello sciame di particelle.

Innanzitutto, vi è una fase di inizializzazione dell'algoritmo ove viene generata una popolazione casuale $(X_i), i = 1, 2, 3, 4, \dots$ per ciascuno elemento della popolazione viene calcolato il relativo valore di fitness. In seguito viene memorizzato il valore (X_{Best}) che rappresenta la soluzione migliore nella popolazione studiata.

E' necessario impostare l'iterazione corrente ($t=1$).

La fase di aggiornamento si compone di numerosi step successivi:

1. generare nuove soluzioni per le *api operaie* (v_i) dalle precedenti soluzioni (X_i) e successivamente calcolare il relativo valore di fitness;
2. scegliere la migliore soluzione tra quella corrente e quella candidata;
3. calcolare la probabilità (P_i) per la soluzione (X_i) ;
4. generare le nuove soluzioni (v_i) relative alle *api spettatrici* dalle precedenti soluzioni (X_i) dipendenti da P_i ;
5. calcolare i valori di fitness delle nuove soluzioni all'interno della popolazione;
6. determinare la soluzione di arresto se esiste e sostituirla con una casuale soluzione X_i ;
7. mantenere la soluzione migliore nella popolazione;

Infine, si imposta la nuova iterazione come $t=t+1$. Dunque, l'ultimo passaggio riguarda la ripetizione dell'intero processo finché non si giunge alla condizione di arresto o alternativamente al risultato ottimale.

Il seguente diagramma di flusso sintetizza le principali fasi dell'algoritmo.

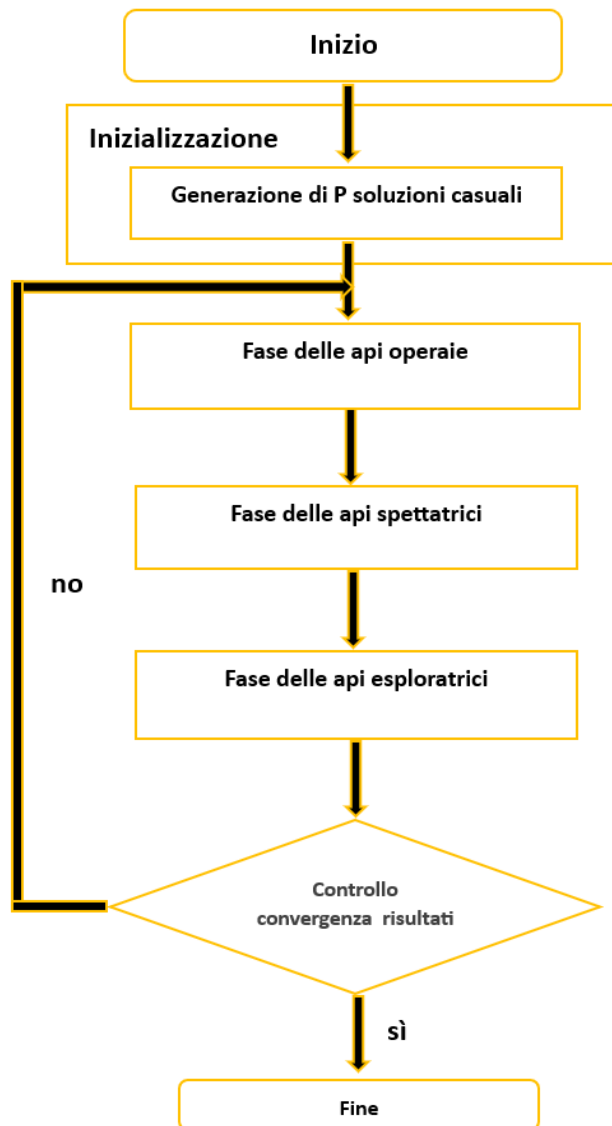


Figura n.8: diagramma di flusso raffigurante l'algoritmo ABC

Fonte: Rielaborazione personale

All'interno della presente trattazione l'algoritmo ABC è stato utilizzato al fine di ottimizzare gli iperparametri coinvolti nella *echo state network* implementata. Analogamente a quanto avviene per l'algoritmo PSO, l'algoritmo ABC dà luogo a valori stocastici dunque è opportuno ripetere il procedimento fino a raggiungere un numero sufficiente di prove.

3 I modelli ibridi ad alta frequenza per la volatilità

3.1 Lo scopo dei modelli ibridi per la volatilità

L'obiettivo del presente elaborato è costruire dei modelli ibridi per lo studio e l'analisi della volatilità delle serie storiche finanziarie ricorrendo ai dati ad alta frequenza. I modelli ibridi proposti sono l'HEAVY-ESN-PSO e l'HARQ-ESN-PSO.

Tali modelli sono stati sviluppati sulle conoscenze del modello HAR-ESN-PSO di Gabriel Trierweiler Ribeiro et al. Il fine è proporre delle strutture ibride alternative in grado di fornire ottime performance predittive per quanto concerne l'analisi della volatilità di serie storiche finanziarie. Le performance sono valutate in termini di errore quadratico medio, una misura d'errore che misura lo scarto quadratico medio tra valori osservati e valori previsti dai modelli.

Inoltre, all'algoritmo del *particle swarm optimization* è affiancato l'algoritmo dell'*artificial bee colony* con lo scopo di offrire una soluzione alternativa per quanto riguarda l'affinamento degli iperparametri della *echo state network*.

E' fondamentale ribadire che la *echo state network* risulta essere l'elemento centrale di ogni modello ibrido analizzato in quanto è stata provata la sua efficacia in ambito di analisi e previsione di serie storiche finanziarie. Dunque, i modelli alternativi sono stati

proposti per sfruttare pienamente le potenzialità della *echo state network*.

3.2 Le serie storiche finanziarie oggetto di analisi

L'analisi è stata condotta su tre differenti titoli azionari, i medesimi oggetto di studio all'interno dell'articolo "Novel hybrid model based on echo state neural network applied to the prediction of stock price return volatility " pubblicato da Gabriel Trierweiler Ribeiro, Andrè Alvers Portela Santos, Viviana Cocco Mariani e Leandro Dos Santos Coelho.

Il primo titolo azionario è riferito alla multinazionale Ebay Inc., un sito di aste online fondato a San Jose, California, il 3 settembre 1995. E' una piattaforma web equiparabile a un sito di e-commerce dove agli utenti è concessa la possibilità di vendere e comprare oggetti sia nuovi che usati in qualsiasi momento, con diverse modalità di pagamento e da qualunque dispositivo che disponga di una connessione internet. La prima quotazione di Ebay presso la borsa valori del NASDAQ risale al 21 settembre del 1998, dopo che il sito contava già oltre un milione di utenti solo negli Stati Uniti. L'espansione di Ebay in Italia è avvenuta nel 2001, mentre l'anno successivo ha acquistato il sistema di pagamento elettronico PayPal. Nel 2005 è stata annunciata l'acquisizione del sistema di comunicazione VoIP Skype, poi ceduto a Microsoft nel 2011.

Il fatturato per l'anno 2020 ammonta a 10,800 miliardi di dollari, mentre l'utile netto conseguito è pari a 1,786 miliardi di dollari. Stando ai dati del 2021, il numero complessivo di dipendenti è pari a 12700.

Di seguito, il grafico del titolo Ebay nel periodo compreso tra il 14 ottobre 2019 e il 16 ottobre 2019. Il grafico permette di cogliere la presenza di micro-oscillazioni di prezzo, nonché di raffigurare i movimenti più consistenti di prezzo avvenuti nell'arco di tre giornate di contrattazioni.

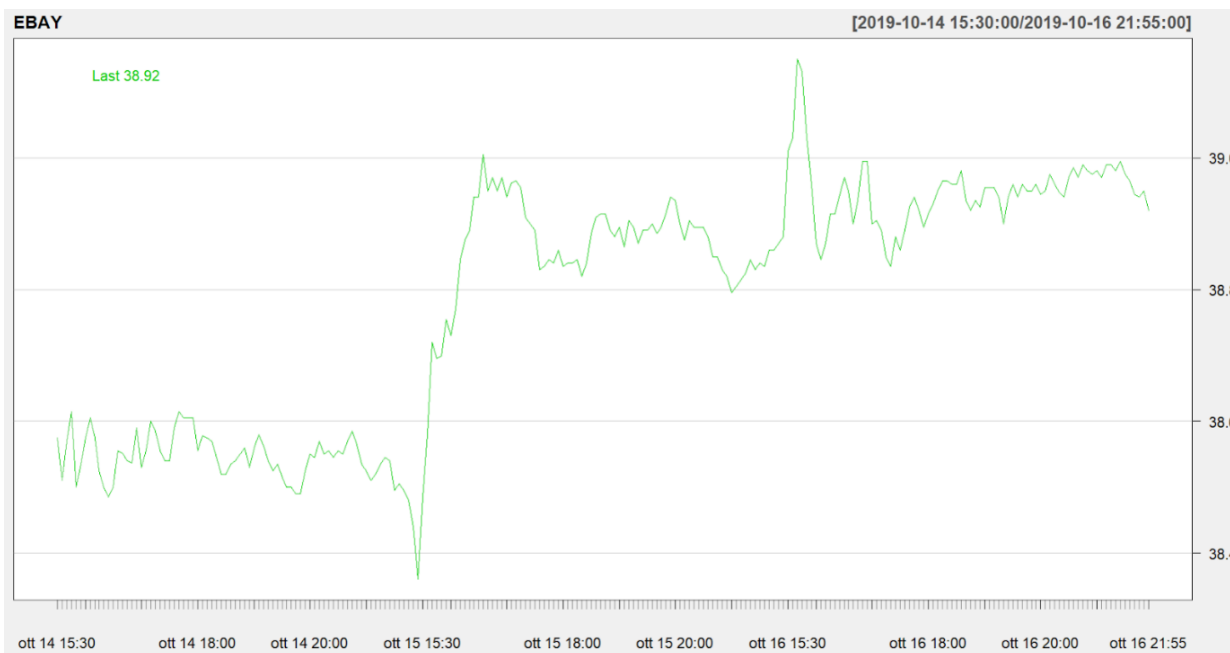


Figura n.9: grafico dei prezzi intra-giornalieri di EBAY
 Fonte: rielaborazione personale

Il secondo titolo azionario si riferisce alla Caterpillar Inc., nota a livello internazionale con l'abbreviazione di CAT. Fondata nel 1925 in California, attualmente la sede principale risiede a Deerfield, in Illinois. Caterpillar Inc. è l'azienda produttrice di macchinari più grande del mondo. Il core business è incentrato sulla produzione di veicoli pesanti, tipicamente mezzi d'opera e macchinari per l'edilizia e l'agricoltura. La prima quotazione del titolo azionario di Caterpillar risale al 2 dicembre 1929. A differenza del titolo EBAY, il titolo CAT è quotato presso la borsa valori del NYSE. La presenza di Caterpillar nel mercato italiano è limitata a due stabilimenti la cui produzione è dedicata alle macchine asfaltatrici e ai rulli per sottocarro.

Al termine dell'anno 2021 il fatturato di Caterpillar risulta essere pari a 59,427 miliardi di dollari, mentre l'utile conseguito ammonta a 6,489 miliardi di dollari.

Il grafico sottostante rappresenta l'evoluzione del prezzo del titolo Caterpillar all'interno dell'orizzonte temporale definito dall'intervallo compreso tra il 14 ottobre 2019 e il 16 ottobre 2019.



Figura n.10: grafico dei prezzi intra-giornalieri di CAT
 Fonte: rielaborazione personale

Il terzo titolo azionario è riferito alla Microsoft Corporation, la nota multinazionale statunitense operante nel settore dell'informatica con sede principale a Redmond, Washington.

E' stata fondata da Bill Gates e Paul Allen il 4 aprile 1975 a Albuquerque. Il momento della quotazione presso la borsa valori del NASDAQ risale al 13 marzo 1986. Microsoft è una delle principali aziende al mondo in termini di capitalizzazione azionaria, il cui valore è approssimativamente pari a 2288 miliardi di dollari secondo l'ultima rilevazione del 2022. Microsoft è presente fisicamente in Italia a partire dagli anni '80 con sede a Milano nella Microsoft House e a Roma nel quartiere dell'EUR. Nel nostro paese, la multinazionale annovera collaborazioni fino a 25000 aziende alleate. Attualmente Microsoft sviluppa, produce e supporta software, personal computer e elettronica di consumo, oltre che fornire servizi legati all'informatica.

Il fatturato registrato al termine dell'anno 2021 ammonta a 161 miliardi di dollari con un utile netto pari a 60,6 miliardi di dollari. Il numero dei dipendenti, il cui dato è stato raccolto nel 2021, è pari a 181000.

Il seguente grafico raffigura le rilevazioni intra-giornaliere di prezzo afferenti al titolo Microsoft riferite all'intervallo di tempo osservato tra il 14 ottobre 2019 e 16 ottobre

2019.

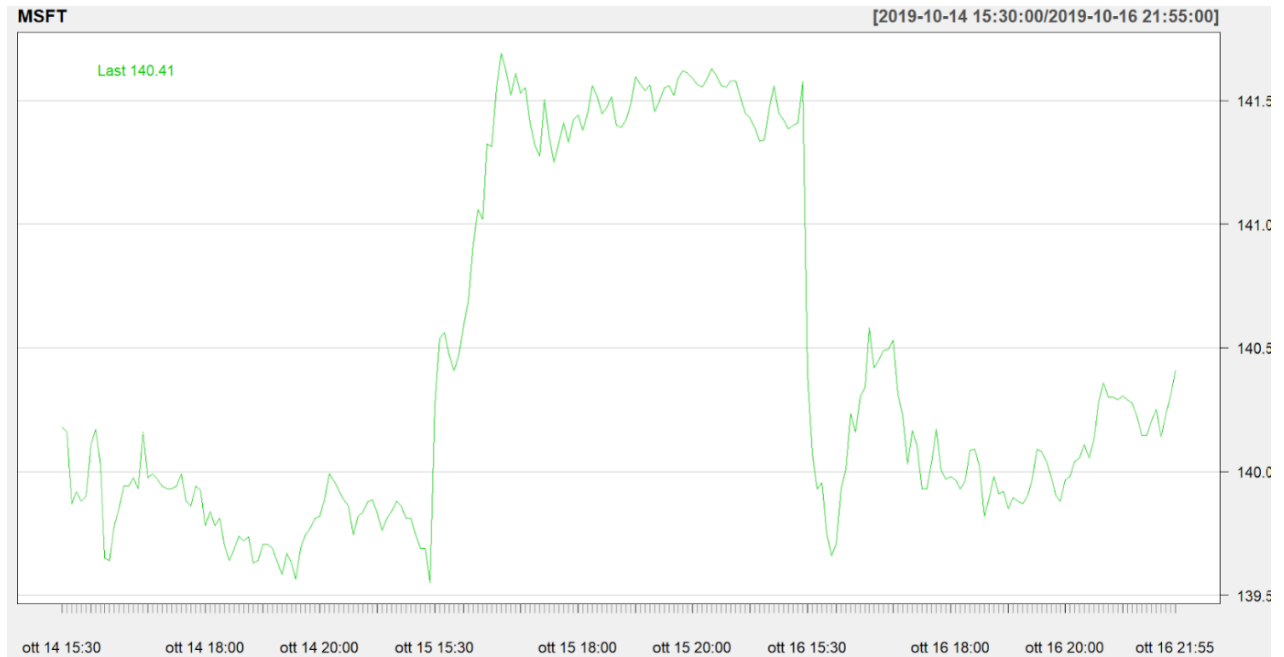


Figura n.11: grafico dei prezzi infra-giornalieri di MSFT
Fonte: rielaborazione personale

Infine, oltre ai tre suddetti titoli azionari, l'analisi della volatilità è stata condotta anche con riferimento all'indice azionario Standard & Poor 500, noto come S&P 500. Tale indice è il più importante benchmark azionario relativo ai titoli quotati presso Wall Street.

E' stato introdotto nel 1957 e si avvale un paniere azionario che rappresenta le 500 aziende statunitensi a capitalizzazione maggiore.

Rientrano nel novero dell'S&P 500 le azioni delle maggiori aziende contrattate presso i mercati azionari del New York Stock Exchange (NYSE), American Stock Exchange (AMEX) e NASDAQ.

I criteri di selezione per l'inserimento nell'S&P 500 sono di duplice natura, sia qualitativa che quantitativa. Innanzitutto, l'azienda deve provenire dagli Stati Uniti, essere quotata presso la borsa di New York o iscritta presso il NASDAQ, osservare le norme in materia di trasparenza di bilancio e rispettare determinati requisiti relativi alla forma giuridica dell'azienda e alla natura della stessa. I criteri quantitativi prevedono che la capitalizzazione di mercato dell'azienda debba essere superiore a 11,2 miliardi di dollari, la quota di azioni liberamente circolanti pari ad almeno il 50% e il controvalore

mensile scambiato deve essere superiore a 250.000 azioni per ognuno dei sei mesi antecedenti alla data di valutazione.

L'indice S&P 500 è composto, tra gli altri, da titoli azionari relativi alle più note e importanti multinazionali statunitensi quali, ad esempio, la Apple, la Microsoft, Amazon e Google.

Il seguente grafico riporta le osservazioni infra-giornaliere di prezzo riferite all'indice azionario S&P 500 all'interno dell'intervallo definito tra il 14 ottobre 2019 e il 16 ottobre 2019.

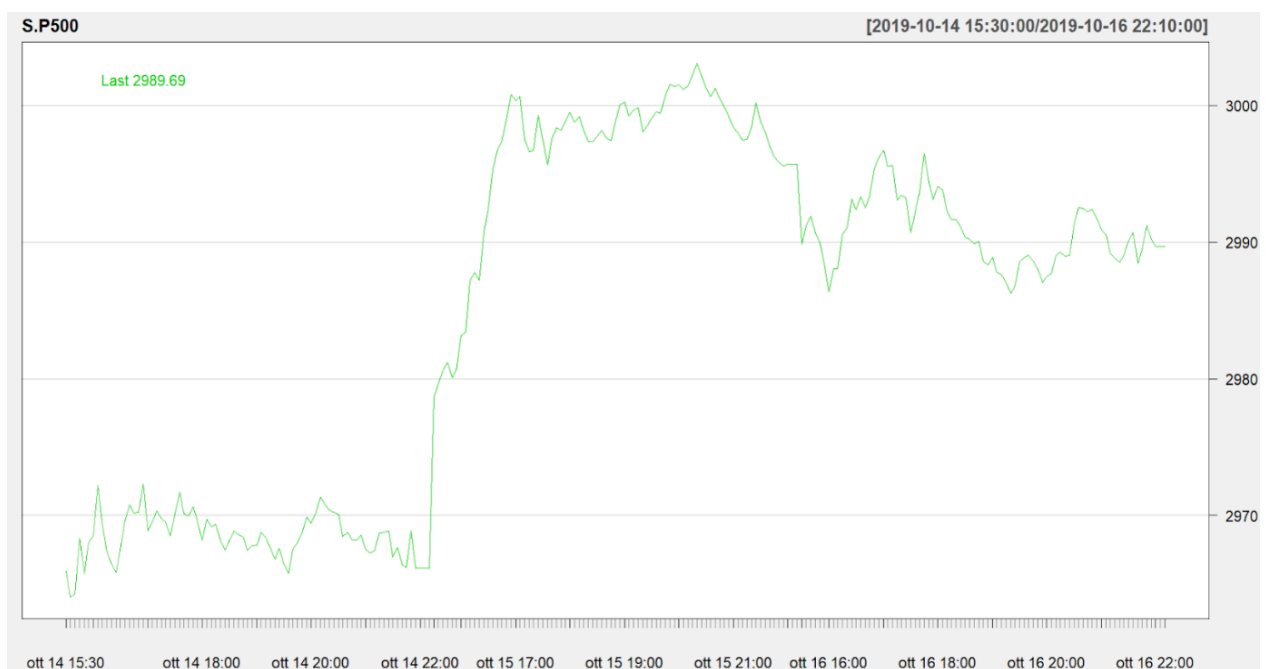


Figura n.12: grafico dei prezzi infra-giornalieri di S&P 500
Fonte: rielaborazione personale

Il periodo di analisi delle quattro serie storiche finanziarie è compreso tra le date 01/01/2015 e 31/12/2019. Le osservazioni sono raccolte ogni 5 minuti per un ammontare di circa 80 rilevazioni giornaliere. I dati sono stati scaricati dai database Bloomberg, il cui accesso è stato effettuato attraverso il Bloomberg Finance Lab situato presso la biblioteca universitaria BEC del campus di San Giobbe.

3.3 L'applicativo R e i relativi pacchetti di risorse

Le analisi effettuate all'interno della presente trattazione sono state condotte ricorrendo al linguaggio di programmazione "R" visualizzato all'interno dell'interfaccia grafica "R Studio". I principali pacchetti utilizzati all'interno della ricerca condotta sono tre, articolati con riferimento a ciascuna componente principale dei modelli ibridi proposti.

Il pacchetto "*highfrequency*", scaricabile presso la repository ("CRAN") di "R", è stato creato da Kris Boudt et al. Questo pacchetto permette di implementare numerose funzioni per lo studio e l'analisi di serie storiche finanziarie disponendo di dati ad alta frequenza. Difatti, il pacchetto permette di calcolare e stimare un ampio ventaglio di *realized measure* comprendente, tra le altre, la *realized variance*, la *realized bi-power variation* e la *realized quarticity*.

A supporto dell'analisi della volatilità sono presenti numerosi test volti ad individuare la presenza di salti di prezzo all'interno delle serie storiche finanziarie. A titolo esemplificativo sono presenti lo Jiang e Oomen test (2008) e l'Ait-Sahalia e Jacod test (2009).

Inoltre, questo pacchetto permette di esplorare e implementare alcuni modelli per lo studio della volatilità di serie storiche finanziarie afferenti alla famiglia dei modelli HAR. Difatti, è possibile eseguire il classico modello HAR di Corsi (2009), nonché alcune delle sue numerose estensioni proposte nel corso degli anni.

Oltre al modello HAR, è presente anche il modello high frequency based volatility (HEAVY) di Sheppard e Shephard preposto allo studio della varianza condizionata nelle serie storiche finanziarie. A questi modelli sono associati i tipici comandi essenziali per lo studio delle serie storiche quali le funzioni *plot* e *predict*.

Il pacchetto "*rESN*" creato da Jared Huling con lo scopo di fornire una soluzione intuitiva e pragmatica per lo studio delle *echo state network*.

Tale pacchetto è stato scaricato presso il sito web *Github*, una piattaforma che offre un servizio di hosting per progetti software. In questa piattaforma è possibile ampliare la gamma di pacchetti, con le relative funzioni, a disposizione sul proprio programma di "R". Difatti, è stato necessario ricorrere a tale piattaforma in quanto nella repository di "R" non è stato reperito alcun pacchetto incentrato sul tema del

reservoir computing. Tale pacchetto risulta essere particolarmente fruibile, sebbene non permetta di personalizzare e sviluppare strutture di *echo state network* particolarmente articolate.

Il pacchetto "*MetaheuristicOpt*", creato da Lala Septem Riza et al e scaricabile dalla repository di "R", permette di implementare 21 differenti algoritmi di ottimizzazione. Gli algoritmi di ottimizzazione proposti sono facenti parte della classe degli algoritmi evolutivi quali il *genetic algorithm*, il *particle swarm optimization*, l'*artificial bee colony*, il *grey wolf optimizer* e via discorrendo.

3.4 La scelta e la dimensione dei dati utilizzati

Le serie storiche finanziarie relative ai titoli Ebay, Caterpillar, Microsoft e all'indice S&P 500 sono considerabili come processi stocastici che si distribuiscono su scale di valori molto differenti. E' sufficiente consultare le immagini numero 9 e numero 12 relative, rispettivamente, al titolo EBAY e all'indice S&P 500 per cogliere le principali differenze in termini di quotazione. Difatti, mentre il prezzo del titolo EBAY presenta un andamento oscillatorio che si attesta intorno ai 38 dollari di quotazione, il valore dell'indice azionario S&P 500 è prossimo ai 3000 punti di quotazione. Insomma, appare evidente la differente grandezza di dimensione dei dati.

Questa differenza di scala rende ardua l'analisi comparata dei risultati ottenuti e pertanto si è soliti ricorrere a delle tecniche di ridimensionamento e riorganizzazione dei dati. Le due tecniche di ridimensionamento dei dati più diffuse sono la standardizzazione e la normalizzazione. Entrambe riscontrano una vasta applicazione in ambito di analisi e studio di serie storiche finanziarie.

La standardizzazione presuppone che la distribuzione dei prezzi di un titolo venga ricondotta a una distribuzione "standard" con valore atteso pari a 0 e varianza pari a 1. Il procedimento prevede di sottrarre alla variabile aleatoria la sua media e dividere il tutto per la deviazione standard.

La normalizzazione è una tecnica che limita l'escursione dei valori di una variabile

aleatoria all'interno di un intervallo predefinito, tipicamente tra il valore 0 e il valore 1. Entrambe le tecniche hanno lo scopo di favorire l'analisi comparata tra le diverse serie storiche finanziarie espresse su scale differenti.

Tuttavia, all'interno del presente elaborato non si ricorre ad alcuna tecnica di ridimensionamento dei dati in quanto ai fini della previsione degli stimatori della volatilità adottati non si considerano i prezzi dei titoli, bensì i *rendimenti infra-giornalieri*. Quest'ultimi rappresentano una misura che limita notevolmente l'escursione dei dati concentrando i valori ottenuti intorno allo zero. Per tale ragione è possibile raffrontare le differenti serie storiche finanziarie sulla base di scale di valore non eccessivamente dissimili fra loro.

3.5 Training set e validation set

I dataset delle quattro serie storiche finanziarie oggetto di analisi nel presente elaborato sono composti da circa 97000 osservazioni cadauno.

Al fine di modellare e prevedere il comportamento della volatilità dei titoli analizzati è opportuno distinguere il set di dati in due differenti raggruppamenti: il training set e il validation set.

Il training set rappresenta la porzione di dataset utilizzata per la fase di inizializzazione e addestramento dell'intero modello statistico.

In queste fasi il modello viene allenato con lo scopo di apprendere la relazione tra le variabili X , ossia le variabili di input, e la variabile Y , la variabile dipendente.

Una volta addestrato, il modello può incorrere nel fenomeno conosciuto come *overfitting*. Tale termine si riferisce alla capacità del modello di predire perfettamente il comportamento dei dati utilizzati nella fase di training, ma di non riuscire a generalizzare i nuovi dati.

Il validation set rappresenta l'insieme dei nuovi dati da trasmettere al modello con lo scopo di verificare la bontà predittiva del modello adottato. Il confronto tra la Y predetta e la Y reale avviene utilizzando una serie di misure di errore quali l' R quadro e il MSE.

Il rapporto che determina la ripartizione del dataset in training set e validation set prevede che siano definiti un intervallo compreso tra il 70-80% per le osservazioni destinate al training set e un intervallo compreso tra il 20-30% per le osservazioni riferite al validation set.

Tale proporzione è giustificata dal fatto che l'addestramento della rete necessita di una mole corposa di dati e contestualmente è opportuno che il validation set sia sufficientemente ampio da riuscire a verificare le capacità predittive del modello.

Nella presente trattazione la scelta è ricaduta sulla ripartizione 80-20%.

3.6 La tecnica della rolling window

L'analisi delle serie storiche finanziarie adottate prevede l'implementazione della tecnica statistica della *rolling window*. Attraverso tale tecnica è possibile suddividere la serie storica in sottoperiodi temporali con una duplice finalità.

Innanzitutto, è possibile valutare la stabilità dei modelli statistici e l'andamento dei relativi coefficienti nel tempo. Inoltre, tale tecnica permette di confrontare l'accuratezza della previsione dei modelli nel corso del tempo.



Figura n.13: esempio di una serie storica suddivisa in finestre mobili.

Fonte: Rielaborazione personale sulla base del sito web "stackoverflow.com".

L'immagine 13 rappresenta la tecnica della *rolling window* associata alle definizioni di training set e validation set.

La *rolling window*, conosciuta come finestra mobile, è una tecnica statistica ampiamente diffusa nell'ambito dell'analisi e della previsione delle serie storiche finanziarie. Tale tecnica prevede di suddividere l'intera serie storica in sottoperiodi temporali costituendo delle finestre temporali di un definito numero di periodi. Le finestre mobili slittano progressivamente di periodo in periodo mantenendo costante la loro ampiezza.

Al concetto di *rolling window* è contrapposto il concetto di *expanding window*. Tale tecnica prevede di definire un'iniziale finestra temporale composta da un numero definito di sottoperiodi e in seguito estendere il numero di sottoperiodi delle successive finestre temporali ampliandone progressivamente l'ampiezza. La scelta della tecnica statistica da implementare dipende dalla natura della serie storica generatrice dei dati. Nel caso in cui il processo osservato fosse costante nel tempo la scelta ideale ricadrebbe sull'adozione della tecnica della *rolling window*, mentre se i parametri sono mutevoli nel tempo allora è maggiormente indicata l'adozione della *expanding window*. Nella presente trattazione la scelta è ricaduta sulla tecnica statistica della *rolling window*, coerentemente con quanto proposto da Gabriel Trierweiler Ribeiro, Andrè Alvers Portela Santos, Viviana Cocco Mariani e Leandro Dos Santos Coelho (2021). L'elaborato prevede la ripartizione delle serie storiche finanziarie in 6 distinte finestre mobili la cui ampiezza corrisponde a 30 mesi di osservazioni infragiornaliere. Le finestre mobili slittano di 6 mesi in 6 mesi coprendo l'intero periodo di rilevazioni costituito da 60 mesi di osservazioni infragiornaliere. Considerato un rapporto di 80-20% per la distinzione in training set e validation set, i primi 24 mesi di ciascuna finestra mobile costituiscono i dati necessari all'addestramento della rete, mentre i successivi 6 mesi sono destinati al set per la previsione e misurazione delle performance dei modelli approntati.

La tecnica della *rolling window* è implementata in modo tale da evitare sovrapposizioni dei periodi di previsione favorendo il confronto temporale dei risultati ottenuti.

La scelta di costituire delle *rolling window* piuttosto estese favorisce la costruzione dei modelli stocastici per la volatilità, nonché beneficia la coerenza e la significatività dei relativi coefficienti.

3.7 Definizione dei modelli ibridi

La volatilità può essere stimata attraverso un'ampia selezione di misure e stimatori in base alla natura dei dati a disposizione e alle finalità della ricerca. Nel presente elaborato, lo studio della volatilità è stato condotto con riguardo a serie storiche finanziarie costituite da dati ad alta-frequenza ricorrendo alle cosiddette *realized measure*, stimatori non parametrici della volatilità.

I modelli ibridi, basati sulle *realized measure*, si articolano lungo due strutture principali: il modello stocastico per la volatilità prescelto e la *echo state network*.

3.7.1 I modelli stocastici per la volatilità

I modelli stocastici per la volatilità sono stati selezionati sulla base delle *realized measure* adottate, differenti per caratteristiche e metodo di stima.

Le *realized measure* prescelte sono la *realized variance*, la *realized quarticity* e la *realized bi-power variation*. Tali stimatori rappresentano la variabile dipendente Y che i modelli ibridi proposti intendono prevedere. Sviluppando i suddetti modelli è possibile ricavare le relative variabili dipendenti da immettere all'interno della struttura della *echo state network*. I tre distinti modelli considerati per l'analisi della volatilità sono:

- il modello HAR-RV di F. Corsi: tale modello fornisce una previsione della *realized variance* distinguendo tre differenti definizioni di *realized variance* sulla base dell'orizzonte temporale considerato;
- il modello HARQ di T. Bollerslev: tale modello rappresenta un'estensione del modello HAR di Corsi integrando lo stimatore della *realized quarticity*. Alla versione originale del modello HARQ si affianca la versione semplificata. Ai fini del presente elaborato è stato ritenuto opportuno presentare la sola versione semplificata;
- il modello HEAVY di K. Sheppard e N. Shephard: tale modello è formato da un sistema di due equazioni atte a stimare la varianza condizionata. La seconda equazione del sistema, definita "*realized measure equation*" è utilizzata per

effettuare previsioni sulla *realized measure* adottata nel modello. Lo stimatore della volatilità prescelto è la *realized bi-power variation*.

Le variabili ottenute sviluppando tali modelli rappresentano le variabili dipendenti X, la cui numerosità è suscettibile in base al modello prescelto.

Ad esempio, per quanto concerne il modello HAR-RV di F. Corsi la variabile della *realized variance* rappresenta la variabile dipendente Y, mentre le tre componenti di volatilità realizzata giornaliera, volatilità realizzata settimanale e volatilità realizzata mensile rappresentano le variabili indipendenti.

3.7.2 La configurazione della echo state network

La configurazione della *echo state network* adottata si fonda sulla struttura originale proposta da Jaeger (2022). La scelta ha una duplice motivazione legata alla funzionalità e alla disponibilità di risorse.

Gli autori del modello ibrido HAR-ESN-PSO, ossia il modello di riferimento del presente elaborato, hanno dimostrato la bontà e l'efficacia predittiva della *echo state network* "base" conseguendo ottime performance se raffrontate con modelli alternativi quali, ad esempio, le classiche reti neurali ricorrenti.

Inoltre, il pacchetto "*rESN*" permette di definire strutture piuttosto standard per la *echo state network* scevri della possibilità di intervenire massivamente sulla configurazione del serbatoio e sulla connettività dei neuroni presenti nei vari strati.

Alla luce di queste considerazioni la *echo state network* adottata è composta da tre distinti strati:

- lo strato di input, definita come *layer* $u(t)$. Tale *layer* è costituito da una matrice ove le righe rappresentano le osservazioni disposte sulla base dell'ordine temporale, mentre le colonne si riferiscono alle distinte variabili indipendenti;
- il serbatoio è tipicamente definito attraverso una matrice di grandi dimensioni, quali possono essere una 50x50 o una 100x100;

- lo strato di output, definito come *layer* $y(t)$. Le righe rappresentano le osservazioni raccolte in base all'ordine temporale di riferimento. Il valore in colonna rappresenta la variabile dipendente che il modello adottato intende stimare.

La personalizzazione della ESN verte interamente sulla corretta selezione degli iperparametri, ossia parametri che regolano il processo di configurazione e addestramento della rete. Tali parametri possono incidere notevolmente sulla struttura della rete determinando risultati molto differenti in base ai valori immessi e attribuiti agli stessi iperparametri in fase di inizializzazione delle rete.

3.7.3 Gli iperparametri della rete

I sei iperparametri regolabili all'interno del comando di inizializzazione della rete sono:

- Il parametro "*n.neurons*", concernente la numerosità dei neuroni all'interno del serbatoio, deve risultare piuttosto elevato per favorire la funzionalità e la stabilità della rete;
- Il parametro "*density*", concernente la numerosità delle connessioni non-nulle all'interno del serbatoio della rete. Tale parametro è compreso tra 0 e 1 pertanto più il valore si avvicina a 1 più risulta fitta la rete di connessioni che vengono stabilite all'interno del serbatoio. Tipicamente, i neuroni presenti nel reservoir sono scarsamente connessi;
- Il parametro "*back.density*". Analogamente a quanto detto per il parametro "*density*", tale valore è compreso tra 0 e 1 e si riferisce alla densità delle connessioni W_{out} stabilite tra il serbatoio e lo strato di output;
- Il parametro "*leak.rate*", anch'esso compreso tra 0 e 1, rappresenta il tasso della *leaky integrator*, un'equazione differenziale specifica tipica delle *echo state network*;
- Il parametro "*lambda*" corrisponde al parametro che regola la *ridge regression*. Tipicamente, valori alti di tale parametro penalizzano il processo di addestramento.

- Il parametro “*spectral radius*” relativo alle connessioni dei neuroni presenti all’interno del serbatoio. Tale parametro è responsabile del processo di inizializzazione delle connessioni pesate W_{res} del serbatoio.

I parametri “*n.neurons*”, “*density*” e “*back.density*” sono stati definiti sulla base degli studi pregressi e delle conoscenze acquisite in materia di *echo state network*. Tali parametri non sono stati individuati ricorrendo ad appositi algoritmi di ottimizzazione in quanto trattasi di parametri che intervengono notevolmente sulla struttura della *echo state network* e pertanto è consigliabile selezionare valori coerenti con quanto indicato dalle ricerche in materia. Numerose pubblicazioni suggeriscono di impostare un elevato numero di neuroni all’interno del serbatoio le cui connessioni pesate devono risultare piuttosto rade e scarsamente diffuse.

I valori prescelti per i parametri “*n.neurons*”, “*density*” e “*back.density*” sono, rispettivamente, 50L (stante per una matrice 50x50), 0.1 e 0.

Inoltre, la scelta di determinare in modo arbitrario i valori dei suddetti parametri è conseguenza della limitata capacità di calcolo del terminale in sede di elaborazione degli algoritmi di ottimizzazione.

I restanti iperparametri sono stati affinati, modello per modello, attraverso l’utilizzo di algoritmi di ottimizzazione quali il *particle swarm optimization* e l’*artificial bee colony*. L’ottimizzazione degli iperparametri è stata condotta con riferimento a degli intervalli di valori suggeriti dalla letteratura in tema di *echo state network*. Difatti, il parametro “*leak.rate*” è definito all’interno dell’intervallo [0,1], il parametro “*lambda*” definito tra i valori [0,1] e il parametro “*spectral radius*” tra i valori [0.1,1.5].

Definiti gli iperparametri coinvolti nel comando di inizializzazione delle rete, la *echo state network* risulta essere approntata e operativa. Attraverso il comando “*train*” è possibile effettuare l’addestramento delle connessioni pesate W_{out} .

Infine, al termine di questo processo è possibile trasmettere i dati del validation set alla rete effettuando la previsione della variabile dipendente interessata. L’ultimo passaggio consiste nel verificare le capacità predittive del modello attraverso un riscontro grafico e ricorrendo a misure di calcolo d’errore quale il MSE.

3. 8 Il confronto dei risultati dei modelli ibridi proposti

Il presente paragrafo è destinato al confronto dei modelli ibridi proposti. Il confronto si basa sulla comparazione grafica delle previsioni ottenute da ciascun modello e sulla valutazione dell'errore medio quadratico, o MSE, che misura la discrepanza quadratica tra i valori osservati e i valori stimati.

Innanzitutto, si procede con l'esposizione dei risultati ottenuti attraverso il modello ibrido HAR-ESN-PSO, mentre in seguito si effettua il confronto con i modelli HEAVY-ESN-PSO e HARQ-ESN-PSO.

Infine, vi è un paragrafo dedicato al confronto dei risultati ottenuti dai modelli ibridi affinati attraverso l'algoritmo del *particle swarm optimization* e i modelli ibridi affinati attraverso l'algoritmo dell'*artificial bee colony*.

3.8.1 Il modello HAR-ESN-PSO

Il modello HAR-ESN-PSO si articola in due strutture principali: il modello statistico HAR di F. Corsi e la *echo state network*, una forma di rete neurale ricorrente.

Il modello HAR distingue tre forme di volatilità a seconda dell'orizzonte temporale considerato. Di seguito le tabelle riassuntive dei coefficienti di ciascun modello distinte in base alla serie storica finanziaria di riferimento.

Il coefficiente Beta 0 rappresenta la costante del modello lineare HAR, mentre il coefficiente Beta 1 è associato alla variabile della *realized variance* a un ritardo, il coefficiente Beta 2 è associato alla *realized variance* settimanale e, infine, il coefficiente Beta 3 è associato alla *realized variance* mensile.

E' opportuno notare che i coefficienti contrassegnati dal corsivo sono significativi al 5%.

Modello HAR titolo EBAY						
Coefficienti	1° finestra mobile	2° finestra mobile	3° finestra mobile	4° finestra mobile	5° finestra mobile	6° finestra mobile
Beta 0 (Int.)	<i>6,01E-03</i>	<i>5,61E-03</i>	<i>3,23E-03</i>	<i>3,33E-03</i>	<i>3,83E-03</i>	<i>4,10E-03</i>
Beta 1	<i>1,94E-01</i>	<i>1,90E-01</i>	<i>2,35E-01</i>	<i>3,18E-01</i>	<i>2,53E-01</i>	<i>2,50E-01</i>
Beta 2	<i>2,15E-01</i>	<i>2,31E-01</i>	<i>2,43E-01</i>	<i>3,33E-01</i>	<i>2,43E-01</i>	<i>2,50E-01</i>
Beta 3	<i>2,35E-01</i>	<i>2,41E-01</i>	<i>2,45E-01</i>	<i>9,63E-02</i>	<i>2,69E-01</i>	<i>2,44E-01</i>

Tabella n.1: i coefficienti del modello HAR titolo EBAY

Fonte: rielaborazione personale

Modello HAR titolo MSFT						
Coefficienti	1° finestra mobile	2° finestra mobile	3° finestra mobile	4° finestra mobile	5° finestra mobile	6° finestra mobile
Beta 0 (Int,)	<i>4,63E-05</i>	<i>3,556E-05</i>	<i>1,472E-05</i>	<i>1,91E-05</i>	<i>1,61E-05</i>	<i>2,42E-05</i>
Beta 1	<i>2,78E-01</i>	<i>2,80E-01</i>	<i>4,11E-01</i>	<i>5,33E-01</i>	<i>6,08E-01</i>	<i>6,04E-01</i>
Beta 2	<i>1,28E-01</i>	<i>1,17E-01</i>	<i>3,82E-02</i>	<i>1,74E-01</i>	<i>1,13E-01</i>	<i>1,45E-01</i>
Beta 3	<i>2,33E-01</i>	<i>3,02E-01</i>	<i>3,25E-01</i>	<i>8,92E-02</i>	<i>1,83E-01</i>	<i>8,53E-02</i>

Tabella n.2, i coefficienti del modello HAR titolo MSFT

Fonte: rielaborazione personale

Modello HAR titolo CAT						
Coefficienti	1° finestra mobile	2° finestra mobile	3° finestra mobile	4° finestra mobile	5° finestra mobile	6° finestra mobile
Beta 0 (Int,)	<i>2,59E-05</i>	<i>3,11E-05</i>	<i>2,34E-05</i>	<i>2,80E-05</i>	<i>2,44E-05</i>	<i>3,37E-05</i>
Beta 1	<i>1,97E-01</i>	<i>1,82E-01</i>	<i>2,11E-01</i>	<i>2,80E-01</i>	<i>2,96E-01</i>	<i>2,92E-01</i>
Beta 2	<i>3,87E-01</i>	<i>3,85E-01</i>	<i>2,33E-01</i>	<i>1,63E-01</i>	<i>1,94E-01</i>	<i>2,16E-01</i>
Beta 3	<i>2,53E-01</i>	<i>2,48E-01</i>	<i>3,20E-01</i>	<i>3,60E-01</i>	<i>4,07E-01</i>	<i>3,27E-01</i>

Tabella n.3, i coefficienti del modello HAR titolo CAT

Fonte: rielaborazione personale

Modello HAR indice S&P 500						
Coefficienti	1° finestra mobile	2° finestra mobile	3° finestra mobile	4° finestra mobile	5° finestra mobile	6° finestra mobile
Beta 0 (Int,)	<i>1,22E-05</i>	<i>1,06E-05</i>	<i>3,36E-06</i>	<i>5,21E-06</i>	<i>5,46E-06</i>	<i>8,29E-06</i>
Beta 1	<i>3,11E-01</i>	<i>3,12E-01</i>	<i>4,83E-01</i>	<i>5,74E-01</i>	<i>6,00E-01</i>	<i>5,85E-01</i>
Beta 2	<i>2,43E-01</i>	<i>2,37E-01</i>	<i>1,12E-01</i>	<i>1,32E-01</i>	<i>1,30E-01</i>	<i>1,90E-01</i>
Beta 3	<i>1,46E-01</i>	<i>1,80E-01</i>	<i>1,82E-01</i>	<i>1,04E-01</i>	<i>1,73E-01</i>	<i>5,56E-02</i>

Tabella n.4, i coefficienti del modello HAR indice S&P 500

Fonte: rielaborazione personale

La significatività dei coefficienti fornisce informazioni circa la presenza di correlazione positiva o negativa tra ciascuna variabile indipendente e la variabile dipendente studiata. Per quanto riguarda il modello HAR, i coefficienti Beta 1, Beta 2 e Beta 3 risultano sempre positivi se significativi.

La tabella numero 1 restituisce, sinteticamente, un modello HAR ove la quasi totalità dei coefficienti risulta essere significativa. Le successive tre tabelle presentano alcuni coefficienti Beta 2 e Beta 3 non significativi, dunque la cui correlazione non è statisticamente significativa. La presenza di alcuni coefficienti non significativi è da imputare parzialmente all'ampiezza temporale delle finestre considerate. Un dataset molto più ampio permetterebbe una maggiore accuratezza dei modelli proposti, tuttavia a causa di limitate capacità computazionali non è stato possibile integrarlo.

I seguenti grafici forniscono un'indicazione circa l'andamento dei coefficienti nel corso delle finestre mobili selezionate.

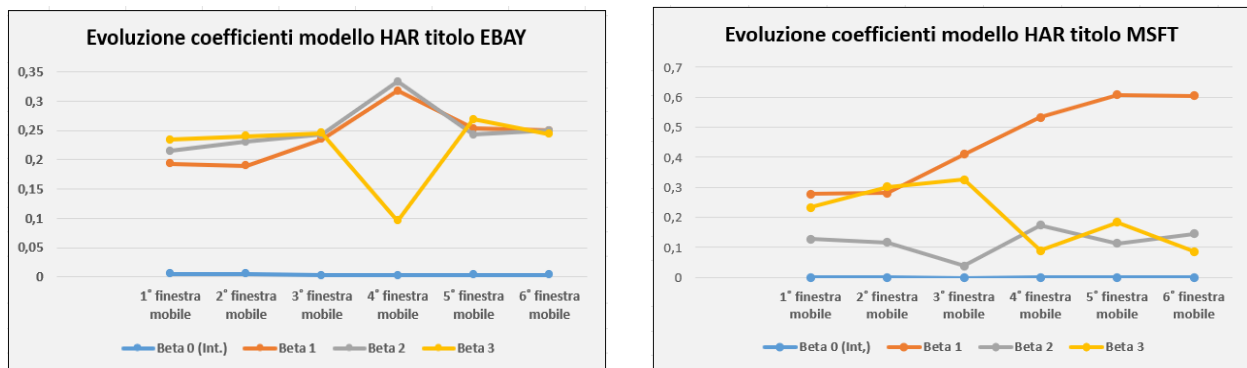


Figura n.14: grafico dell'andamento dei coefficienti del modello HAR relativo ai titoli EBAY e MSFT
Fonte: rielaborazione personale

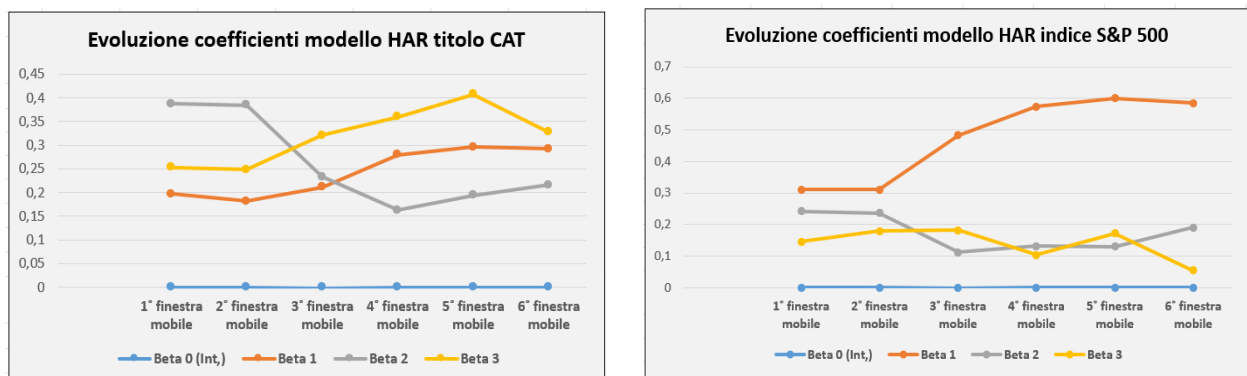


Figura n.15, grafico dell'andamento dei coefficienti del modello HAR relativo ai titoli CAT e S&P 500
Fonte: rielaborazione personale

Per quanto concerne il modello *HAR* relativo al titolo EBAY è apprezzabile la coerenza dei valori dei singoli coefficienti a eccezione della finestra mobile numero 4 ove è presente un significativo scostamento dei valori. I modelli *HAR* costruiti sul titolo MSFT e sull'indice S&P 500 presentano dei coefficienti piuttosto stabili nel tempo a eccezione del coefficiente Beta 1 che ha un marcato andamento evolutivo. Infine, il grafico numero 11, relativo al titolo CAT, raffigura un andamento piuttosto ondivago dei coefficienti analizzati.

La coerenza temporale dei coefficienti contribuisce a garantire la stabilità dei modelli statistici implementati.

Le tre variabili dipendenti formano il set di dati di input da immettere all'interno della *echo state network*. Ciascuna *echo state network* è stata approntata secondo le soluzioni fornite dall'algoritmo di ottimizzazione del *particle swarm optimization*.

I tre iperparametri coinvolti nella procedura di ottimizzazione sono: il *leaking rate*, il *lambda* e lo *spectral radius*. L'obiettivo della procedura di ottimizzazione è minimizzare lo scarto quadratico medio, conosciuto come MSE, ossia il mean squared error riferito alla rete neurale artificiale.

L'algoritmo del PSO è stato impostato con i seguenti parametri=

```
numPopulation=40 maxIter=10 Vmax=2 ci=1.5 cg=1.5 w=0.7
```

Il termine "*numPopulation*" si riferisce alla numerosità della popolazione di particelle, il termine "*maxIter*" si riferisce al numero massimo di iterazioni, il termine "*Vmax*" rappresenta la velocità massima della particella, i termini "*ci*" e "*cg*" rappresentano, rispettivamente, il parametro dell'apprendimento individuale e il parametro dell'apprendimento collettivo e infine il termine "*w*" costituisce il parametro inerziale della velocità.

L'algoritmo del PSO fornisce una soluzione stocastica, dunque aleatoria e pertanto è opportuno ripetere la procedura per un cospicuo numero di prove. La procedura è stata eseguita 50 volte.

I seguenti grafici rappresentano le funzioni di distribuzione riferite, rispettivamente, ai parametri del *leaking rate*, del *lambda* e dello *spectral radius* del modello *HAR-ESN-PSO* riferito al titolo EBAY.

TITOLO EBAY MODELLO HAR-ESN-PSO

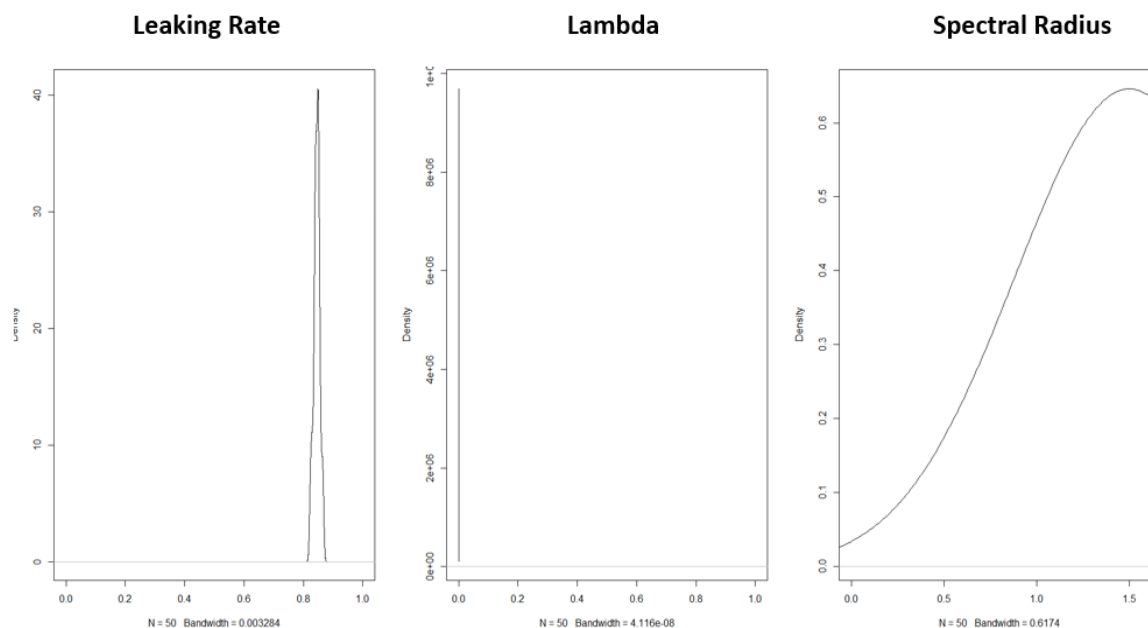


Figura n.16: distribuzione della funzione di densità degli iperparametri *leaking rate*, *lambda* e *spectral radius* relativi al titolo EBAY modello HAR-ESN-PSO
Fonte: rielaborazione personale

Il risultato della procedura di ottimizzazione è piuttosto indicativo. Per quanto concerne il parametro del *leaking rate* la curva della funzione di densità si concentra attorno al valore di $\sim 0,85$, mentre il parametro *lambda* assume esclusivamente valore $1e-07$. Infine, il parametro dello *spectral radius* presenta una distribuzione che richiama il versante sinistro di una distribuzione a campana. Il valore più osservato è di 1,5. La medesima procedura è stata ripetuta per tutti i modelli ibridi riferiti a ciascuna serie storica finanziaria.

Affinati gli iperparametri la rete è approntata per eseguire la previsione della *realized variance*, uno degli stimatori della volatilità presenti all'interno della trattazione.

Di seguito, i quattro grafici che rappresentano la previsione condotta attraverso i modelli ibridi. I grafici sono stati costruiti unendo i periodi di previsione di ciascuna *rolling window* formando un orizzonte di previsione di 36 mesi.

Inoltre, ai fini della fruibilità del grafico i risultati sono stati moltiplicati per 100 e dunque espressi in misura percentuale.

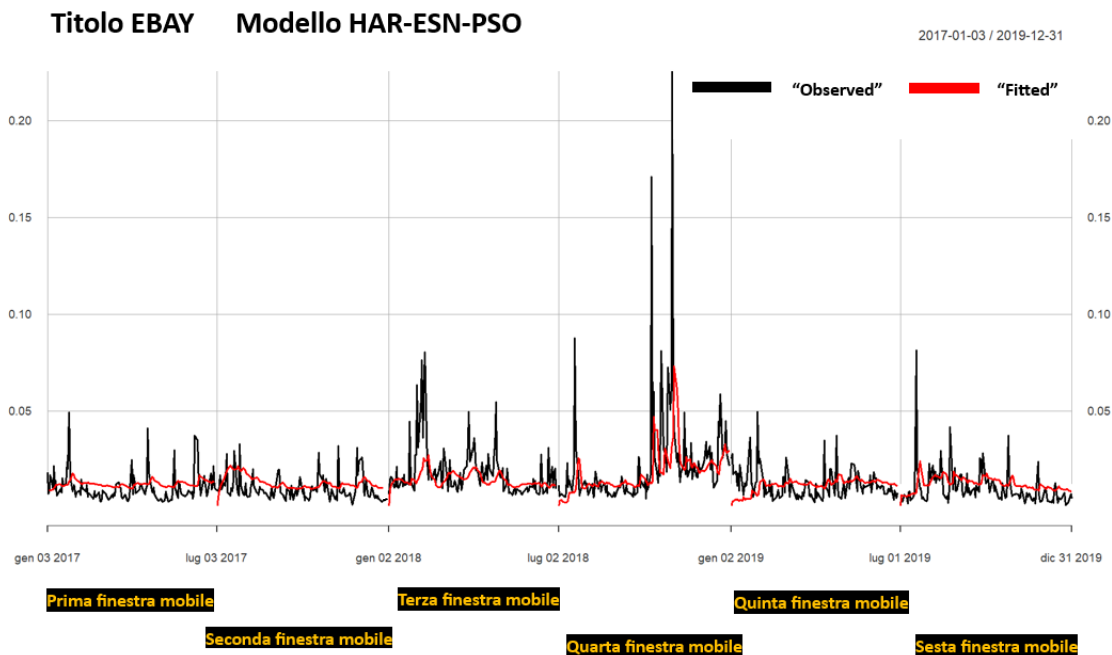


Figura n.17: grafico comparativo realized variance osservata e realized variance predetta titolo EBAY modello HAR-ESN-PSO
 Fonte: rielaborazione personale

Osservando il grafico relativo alla previsione della volatilità del titolo EBAY si evince come i primi valori previsti tendono a discostarsi dai valori osservati, mentre per i successivi valori la previsione risulta essere piuttosto fedele rispetto ai valori osservati. In particolare, i primi valori della quinta finestra mobile si allontanano notevolmente dai valori osservati. La motivazione potrebbe risiedere nella presenza di elevati picchi di volatilità nelle settimane antecedenti evidenziando il fenomeno del *volatility clustering*. Tale fenomeno si sostanzia nell'avvicendamento tra i periodi ad alta volatilità e i periodi a volatilità più contenuta generando dei raggruppamenti di volatilità. Dal grafico si evince che il modello ibrido è particolarmente indicato per cogliere i *cluster* di volatilità prevedendoli in modo piuttosto preciso.

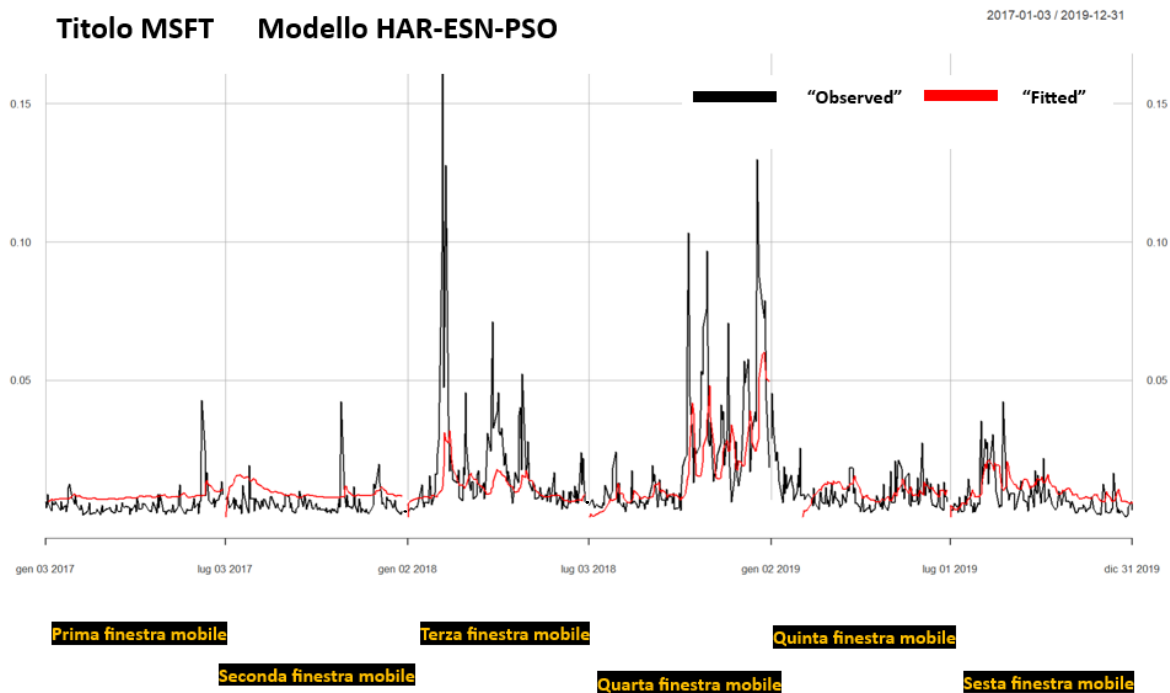


Figura n.18: grafico comparativo realized variance osservata e realized variance predetta titolo MSFT modello HAR-ESN-PSO
 Fonte: rielaborazione personale

Per quanto concerne il titolo MSFT è opportuno indagare sulla previsione all'interno della *rolling window* numero 5. Difatti i primi 10 valori assumono segno negativo non fornendo una previsione coerente e attendibile e pertanto non sono stati rappresentati in quanto privi di significato. La motivazione potrebbe risiedere nella presenza di salti di prezzo piuttosto marcati. Inoltre, ribadendo quanto affermato per il titolo EBAY, tali valori susseguono un periodo di diverse settimane caratterizzato da elevati picchi di volatilità piuttosto concentrati.

Il seguente grafico estratto a partire dai primi giorni dell'anno 2019 rappresenta evidenti salti di prezzo nella serie storica che potrebbero spiegare la difficoltà del modello a rappresentare correttamente la serie della *realized variance* nel periodo di gennaio 2019.

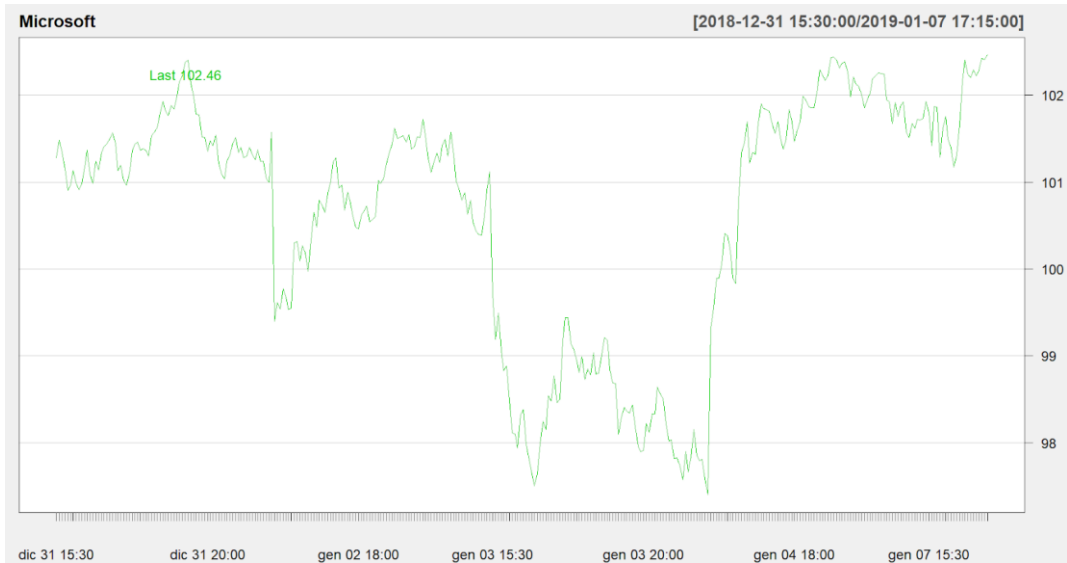


Figura n.19: grafico prezzo titolo MSFT periodo 31-12-2018/07-01-2019
 Fonte: rielaborazione personale

I principali salti di prezzo sono avvenuti tra la giornata del 31 dicembre 2018 e il 1 gennaio 2019, tra il 2 gennaio 2019 e il 3 gennaio 2019 e, infine, tra il 3 gennaio 2019 e il 4 gennaio 2019. Tali salti non sono limitati al titolo MSFT, ma si estendono alle altre serie storiche analizzate quali il titolo CAT e l'indice S&P 500.

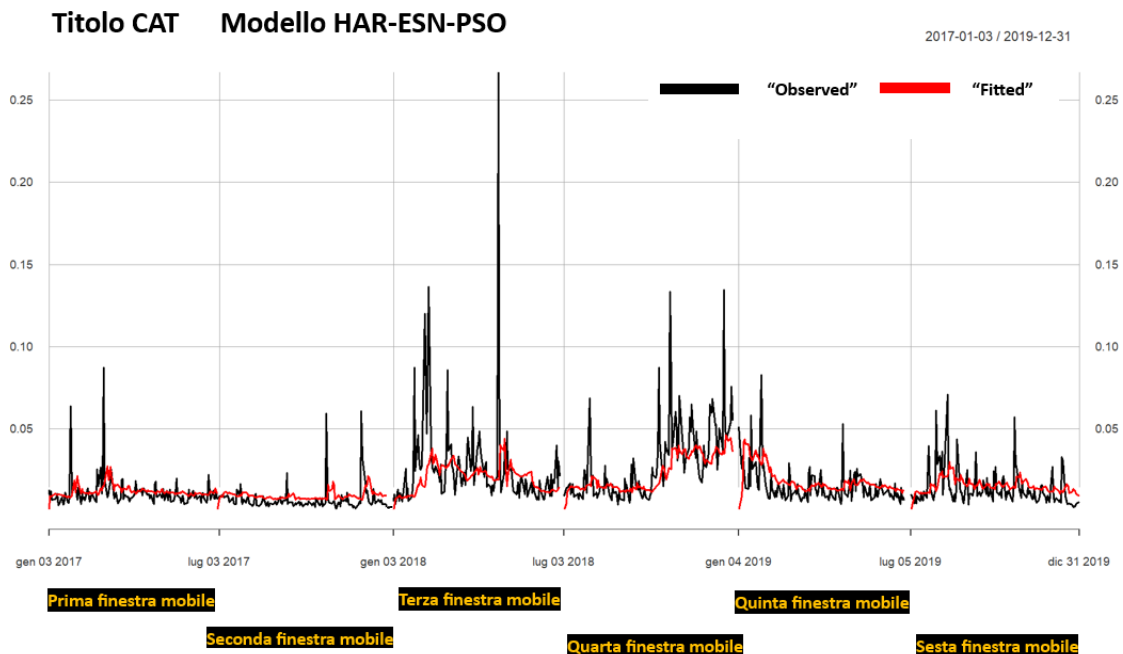


Figura n.20: grafico comparativo realized variance osservata e realized variance predetta titolo CAT modello HAR-ESN-PSO
 Fonte: rielaborazione personale

La previsione della *realized variance* relativa al titolo CAT presenta il medesimo profilo di criticità osservato all'interno della finestra mobile numero 5. Difatti, la presenza di salti di prezzo ha, presumibilmente, inciso sulla bontà della previsione condotta. Infine, appare evidente il fenomeno della *volatility clustering*. In particolare all'interno della terza e della quarta finestra mobile si può apprezzare la presenza di elevata persistenza della volatilità.

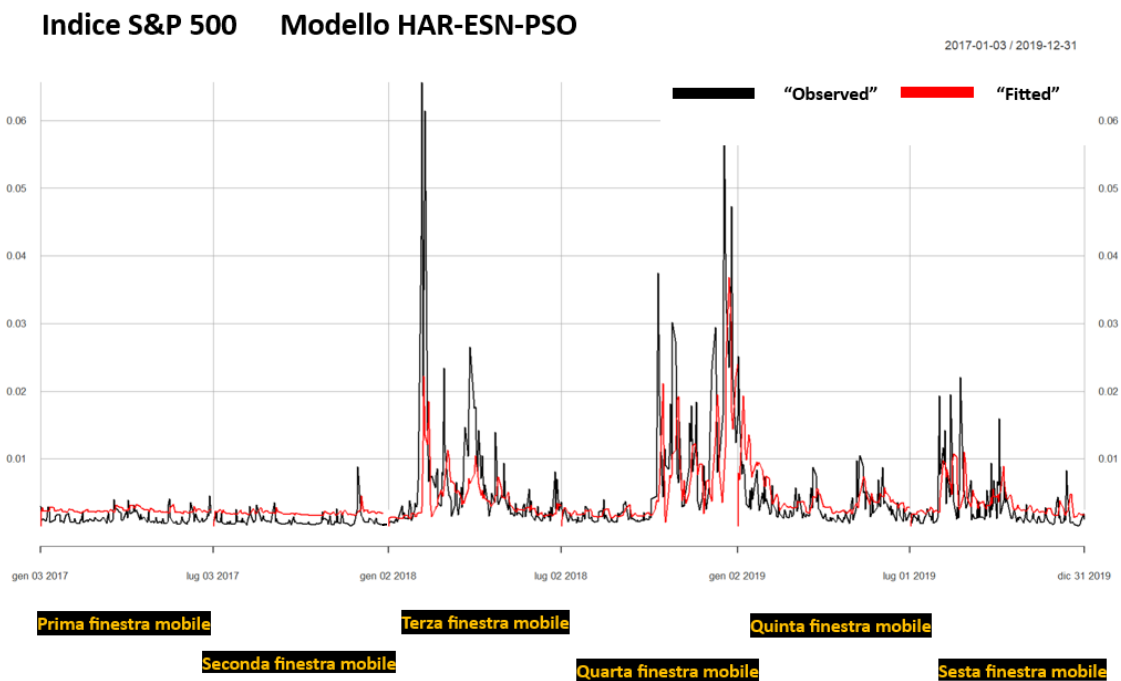


Figura n.21: grafico comparativo realized variance osservata e realized variance predetta indice S&P 500 modello HAR-ESN-PSO
 Fonte: rielaborazione personale

Analogamente a quanto affermato per i titoli analizzati in precedenza, il grafico della *realized variance* relativa all'indice S&P 500 presenta un forte scostamento dei valori predetti e osservati durante i primi giorni della quinta finestra mobile. Tale periodo succede a una fase caratterizzata dalla presenza di picchi considerevoli di volatilità. Il fenomeno del *volatility clustering* è piuttosto apprezzabile con riferimento all'indice S&P 500 in quanto è possibile individuare periodi contraddistinti da raggruppamenti di volatilità alternarsi con periodi a volatilità ridotta. In particolare tale fenomeno emerge

in relazione alle finestre mobili numero 3 e numero 4.

Inoltre, da una breve analisi comparata dei quattro titoli analizzati è possibile evincere come i cluster di elevata volatilità si concentrino nei medesimi periodi con particolare riferimento ai mesi iniziali del 2018 e dei mesi a cavallo tra gli anni 2018 e 2019. Tale comparazione grafica fornisce spunti di discussione in merito alla natura sistemica dei mercati finanziari americani.

Infine, appare evidente come per l'indice S&P 500 il fenomeno del *volatility clustering* risulti amplificato e di portata superiore data la sua natura di paniere azionario.

Al fine di determinare le performance del modello ibrido HAR-ESN-PSO si ricorre alla misura dello scarto quadratico medio. Di seguito le tabelle riassuntive che espongono gli MSE distinti in base alla serie storica finanziaria considerata e alla *rolling window* di riferimento. Inoltre, è stata calcolata la media semplice riferita ad ognuno dei titoli. L'ampiezza delle singole finestre differisce al massimo di 2 valori pertanto è sufficiente adottare la tecnica della media semplice, piuttosto che della media ponderata.

MSE modello HAR-ESN-PSO				
Finestre mobili	Titolo EBAY	Titolo MSFT	Titolo CAT	Indice S&P 500
<i>I</i>	6,19E-09	2,83E-09	1,19E-06	1,92E-10
<i>II</i>	5,34E-09	4,04E-09	9,49E-07	2,11E-10
<i>III</i>	1,67E-08	4,58E-08	1,20E-05	8,14E-09
<i>IV</i>	7,10E-08	4,17E-08	1,31E-05	8,14E-09
<i>V</i>	7,36E-09	1,83E-08	1,76E-06	1,40E-09
<i>VI</i>	9,89E-09	4,61E-09	1,61E-06	1,41E-09
Media	1,94E-08	1,96E-08	5,11E-06	3,25E-09

Tabella n.5: risultati mse del modello HAR-ESN-PSO

Fonte: rielaborazione personale

Le migliori performance, in media, sono state raggiunte con riferimento all'indice S&P 500. Per quanto concerne il titolo CAT l'errore quadratico medio assume valori tendenzialmente superiori rispetto a quelli riscontrati per le altre serie storiche finanziarie analizzate.

I risultati andranno comparati con quanto risulterà dai modelli alternativi proposti nella presente trattazione.

3.8.2 Il modello HEAVY-ESN-PSO

L'articolazione del modello ibrido HEAVY-ESN-PSO poggia su basi analoghe rispetto al modello ibrido HAR-ESN-PSO. Difatti, è possibile distinguere due strutture portanti: il modello stocastico per la volatilità HEAVY di K. Sheppard e N. Shephard e la echo state network.

Il modello HEAVY rappresenta un'estensione della famiglia dei modelli GARCH che utilizza dati ad alta frequenza. Il modello HEAVY si articola in un sistema a duplice equazione: l'equazione dei rendimenti e l'equazione della *realized measure*. Ai fini del presente elaborato è considerata la sola equazione della *realised measure*.

La *realized measure* considerata è la *realized bi-power variation*, anziché la *realized variance* presente nei modelli HAR-ESN-PSO e HARQ-ESN-PSO.

Il seguente grafico fornisce un confronto tra le due *realized measure*, relativamente al titolo EBAY.

Titolo EBAY

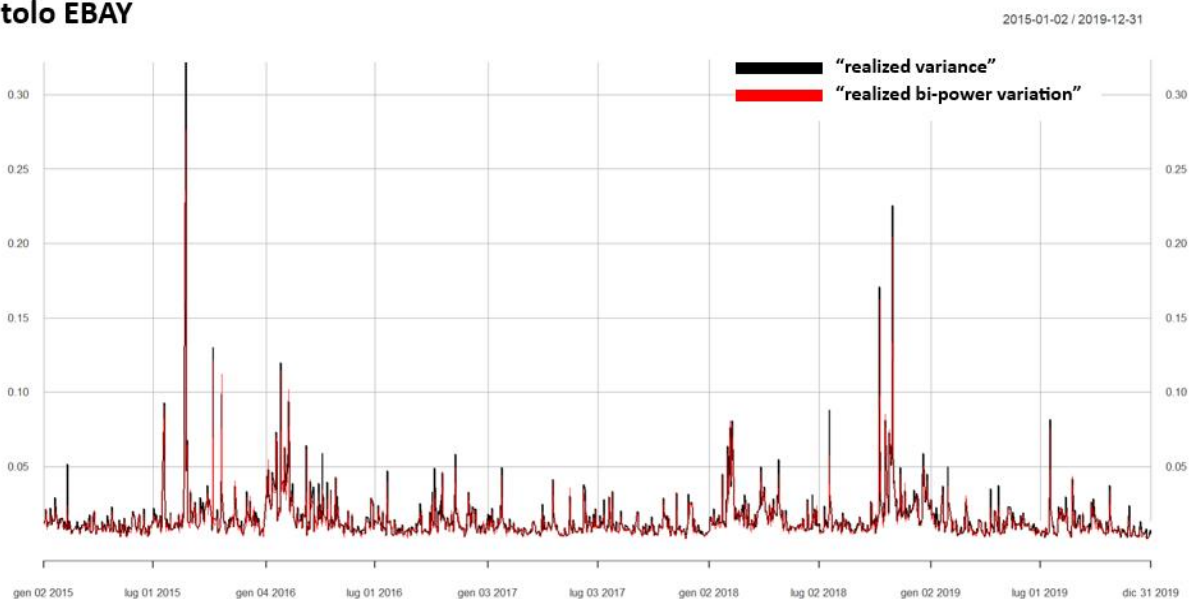


Figura n.22: grafico comparativo realized variance e realized bi-power variation titolo EBAY
Fonte: rielaborazione personale

Le curve dalle *realized variance* e della *realized bi-power variation* si sovrappongono quasi perfettamente lungo tutto il periodo considerato all'interno della presente trattazione. Nonostante le differenti modalità di stima entrambe le misure sono in grado di stimare *l'integrated variance*.

La differenza grafica è data dalla componente di salto di prezzo presente nella serie storica EBAY. Difatti, mentre *la realized variance* non è in grado di sintetizzare gli effetti dei salti di prezzo, la *realized bi-power variation* presenta una robustezza maggiore alla presenza dei salti.

Di seguito le tabelle riassuntive dei coefficienti distinte in base alle serie storiche finanziarie. I primi tre coefficienti fanno riferimento all'equazione dei rendimenti, mentre i secondi tre, ossia Omega RM, Alpha RM e Beta RM afferiscono all'equazione delle *realized measure*.

Modello HEAVY titolo EBAY						
Coefficienti	1° finestra mobile	2° finestra mobile	3° finestra mobile	4° finestra mobile	5° finestra mobile	6° finestra mobile
Omega	2,68E+00	2,38E+00	1,46E-01	2,01E+00	3,99E-01	4,52E-01
Alpha	3,01E-01	4,55E-01	7,95E-01	2,54E-01	2,06E-02	1,91E-02
Beta	2,75E-09	1,25E-06	3,32E-08	9,65E-08	8,30E-01	8,16E-01
Omega RM	3,62E-01	3,39E-01	1,68E-01	2,59E-01	2,65E-01	2,43E-01
Alpha RM	6,04E-01	6,38E-01	3,82E-01	4,28E-01	6,06E-01	5,20E-01
Beta RM	1,87E-01	1,74E-01	4,87E-01	3,61E-01	2,32E-01	3,26E-01

Tabella n.6: i coefficienti del modello HEAVY titolo EBAY
Fonte: rielaborazione personale

Modello HEAVY titolo MSFT						
Coefficienti	1° finestra mobile	2° finestra mobile	3° finestra mobile	4° finestra mobile	5° finestra mobile	6° finestra mobile
Omega	1,03E+00	4,95E-01	1,24E-02	4,05E-01	2,02E-01	3,03E-01
Alpha	7,79E-02	5,31E-01	1,42E-02	2,31E-01	2,05E-01	1,96E-01
Beta	5,29E-01	3,84E-01	9,75E-01	4,71E-01	7,03E-01	6,74E-01
Omega RM	1,70E-01	9,23E-02	8,47E-02	9,73E-02	6,75E-02	9,47E-02
Alpha RM	6,25E-01	6,33E-01	4,54E-01	5,29E-01	6,09E-01	5,64E-01
Beta RM	2,56E-01	3,10E-01	4,37E-01	3,53E-01	3,49E-01	3,66E-01

Tabella n.7: i coefficienti del modello HEAVY titolo MSFT
Fonte: rielaborazione personale

Modello HEAVY titolo CAT						
Coefficienti	1° finestra mobile	2° finestra mobile	3° finestra mobile	4° finestra mobile	5° finestra mobile	6° finestra mobile
Omega	1,39E-01	1,39E-01	1,22E+00	7,88E-01	5,28E-01	6,77E-02
Alpha	3,25E-02	3,25E-02	3,33E-01	1,44E-01	2,79E-01	6,37E-02
Beta	9,15E-01	9,15E-01	1,87E-01	5,10E-01	5,79E-01	9,16E-01
Omega RM	1,28E-01	1,95E-01	1,22E-01	1,69E-01	1,32E-01	1,09E-01
Alpha RM	4,81E-01	4,77E-01	3,58E-01	5,39E-01	5,28E-01	4,56E-01
Beta RM	4,46E-01	4,07E-01	5,36E-01	3,52E-01	4,22E-01	5,04E-01

Tabella n.8: i coefficienti del modello HEAVY titolo CAT

Fonte: rielaborazione personale

Modello HEAVY indice S&P 500						
Coefficienti	1° finestra mobile	2° finestra mobile	3° finestra mobile	4° finestra mobile	5° finestra mobile	6° finestra mobile
Omega	8,74E-02	8,74E-02	8,74E-02	8,74E-02	8,74E-02	8,86E-02
Alpha	1,95E-01	1,95E-01	1,95E-01	1,95E-01	1,95E-01	1,96E-01
Beta	6,94E-01	6,94E-01	6,94E-01	6,94E-01	6,94E-01	6,92E-01
Omega RM	4,28E-02	3,44E-02	2,47E-02	2,18E-02	1,71E-02	1,47E-02
Alpha RM	7,98E-01	8,33E-01	7,04E-01	6,96E-01	7,55E-01	6,91E-01
Beta RM	1,34E-01	1,10E-01	1,94E-01	2,33E-01	2,34E-01	3,09E-01

Tabella n.9: i coefficienti del modello HEAVY indice S&P 500

Fonte: rielaborazione personale

Per ciascuna serie storica analizzata è possibile denotare che i coefficienti Omega RM, Alpha RM e Beta RM sono sovente significativi al 5%.

Di seguito dei sintetici grafici che raffigurano l'evoluzione dei coefficienti del modello heavy ripartiti a seconda della serie storica finanziaria.

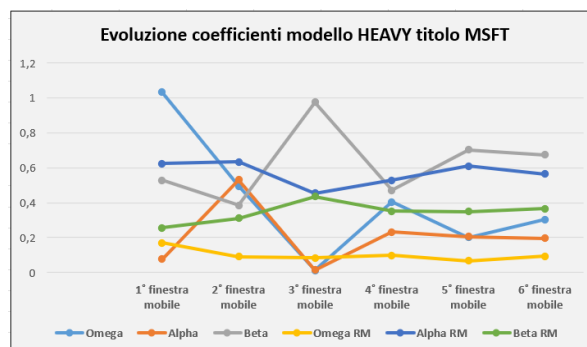
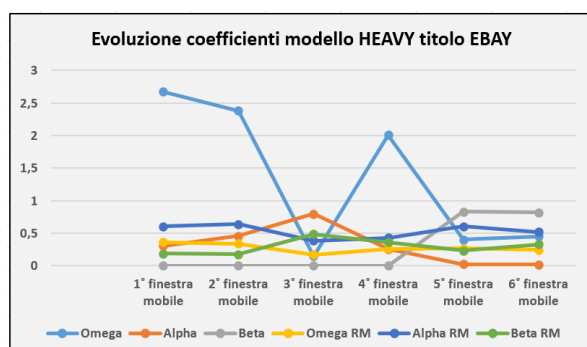


Figura n.23: grafico dell'andamento dei coefficienti del modello HEAVY relativo ai titoli EBAY e MSFT

Fonte: rielaborazione personale

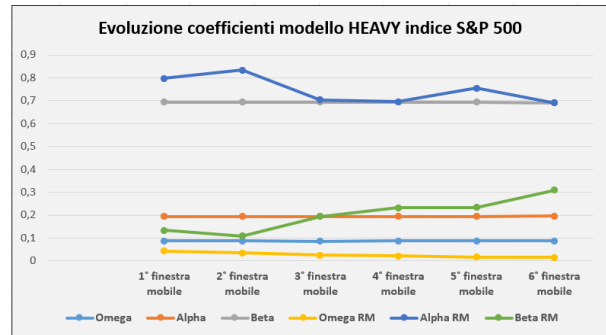
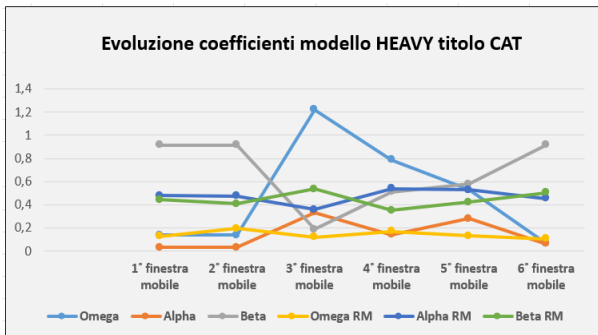


Figura n.24: grafico dell'andamento dei coefficienti del modello HEAVY relativo ai titoli CAT e S&P 500
 Fonte: rielaborazione personale

Le tabelle numero 6 e numero 9, relative, rispettivamente al titolo EBAY e all'indice S&P 500 raffigurano un andamento piuttosto costante dei coefficienti Omega, Alpha e Beta in contrapposizione a quanto si evince dalle tabelle 7 e 8 . Difatti i coefficienti dei modelli HEAVY attinenti al titolo MSFT e al titolo CAT presentano un andamento piuttosto oscillatorio e non costante nel tempo.

La non coerenza temporale dei coefficienti può inficiare la stabilità dei modelli adottati. Il seguente grafico, estratto dal modello HEAVY, fornisce un'importante indicazione circa il fenomeno del *volatility clustering*.

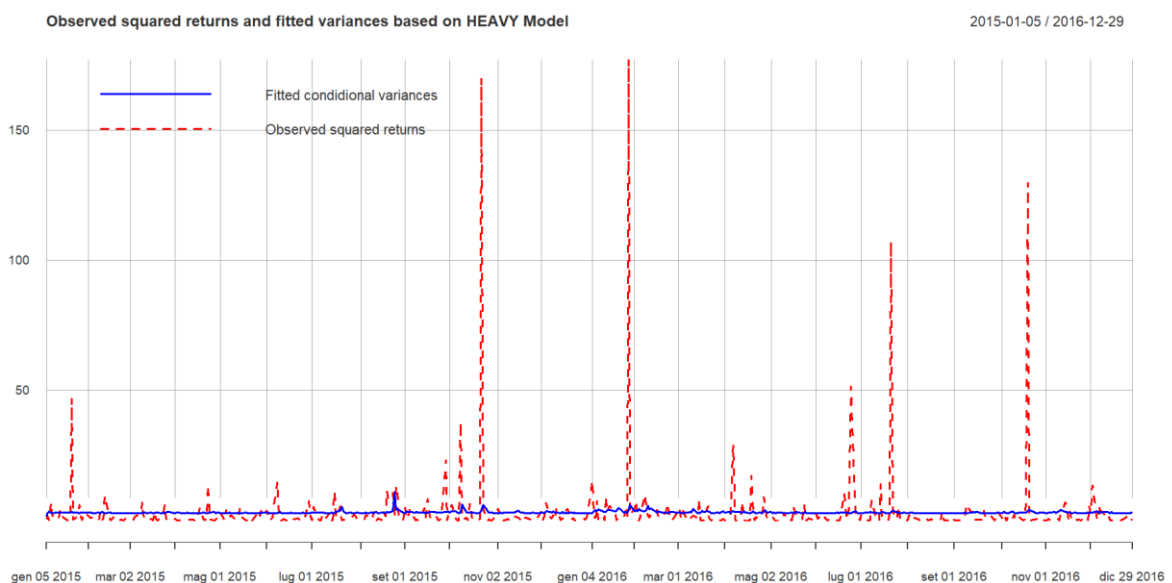


Figura n.25: grafico della varianza condizionata del modello HEAVY titolo EBAY
 Fonte: rielaborazione personale

Il modello HEAVY, attraverso la varianza condizionata, riesce a cogliere solamente i periodi di persistenza della volatilità, come è possibile osservare per il periodo di gennaio-marzo 2018. Per converso, se vi sono picchi isolati di volatilità il modello fatica a darne rappresentazione.

La varianza condizionata relativa all'equazione della *realized measure* rappresenta il dato di input da immettere all'interno della *echo state network*.

L'ottimizzazione degli iperparametri della *echo state network* segue la medesima procedura indicata nel paragrafo precedente e dunque una volta terminata, la rete neurale è funzionante e pronta per effettuare la previsione della *realized bi-power variation*. La previsione dello stimatore della *realized bi-power variation* ha restituito i seguenti riscontri grafici.

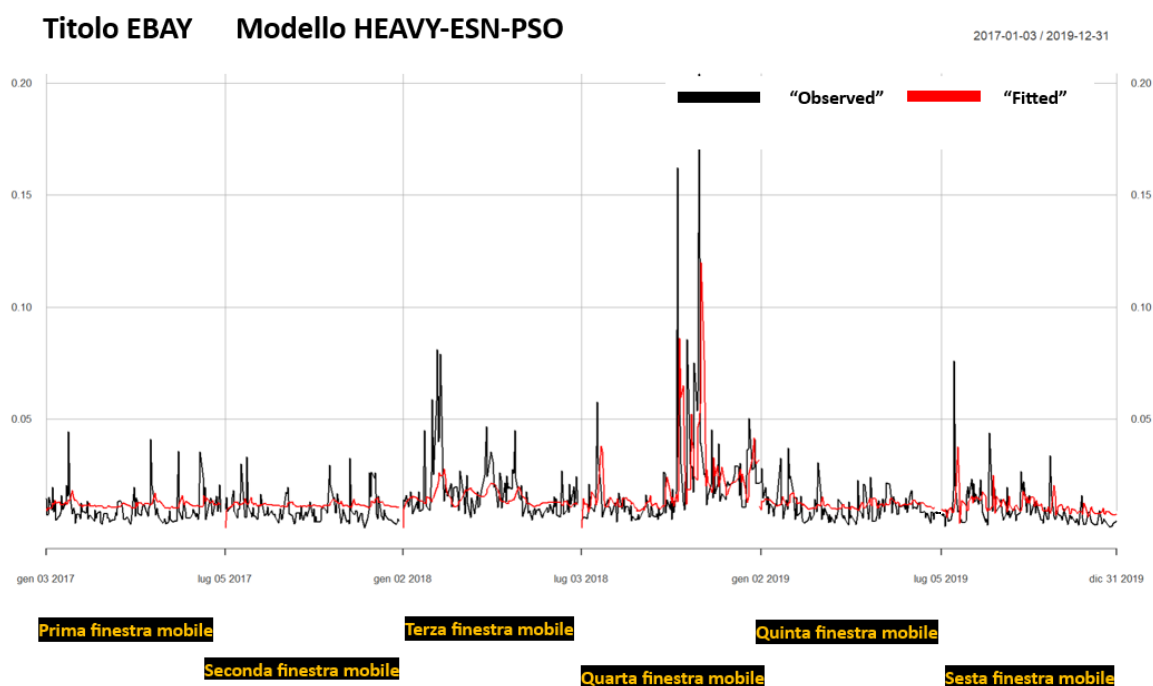


Figura n.26: grafico comparativo realized variance osservata e realized variance predetta titolo EBAY modello HEAVY-ESN-PSO

Fonte: rielaborazione personale

Titolo MSFT Modello HEAVY-ESN-PSO

2017-01-03 / 2019-12-31

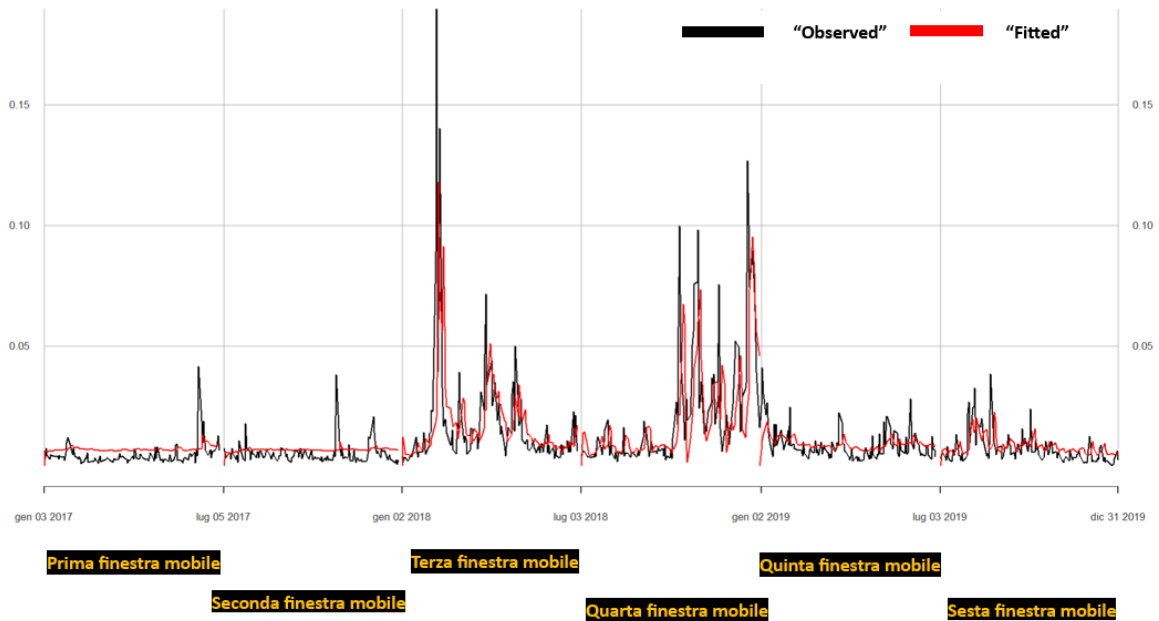


Figura n.27: grafico comparativo realized variance osservata e realized variance predetta titolo MSFT modello HEAVY-ESN-PSO

Fonte: rielaborazione personale

Titolo CAT Modello HEAVY-ESN-PSO

2017-01-03 / 2019-12-31

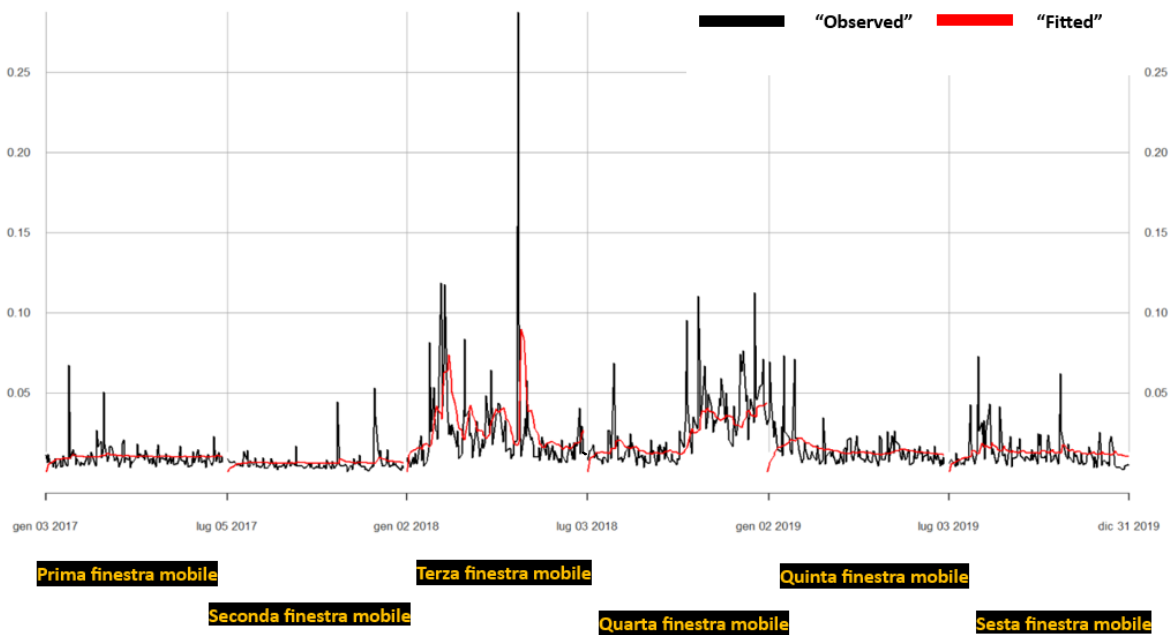


Figura n.28: grafico comparativo realized variance osservata e realized variance predetta titolo CAT modello HEAVY-ESN-PSO

Fonte: rielaborazione personale

Indice S&P 500 Modello HEAVY-ESN-PSO

2017-01-03 / 2019-12-31

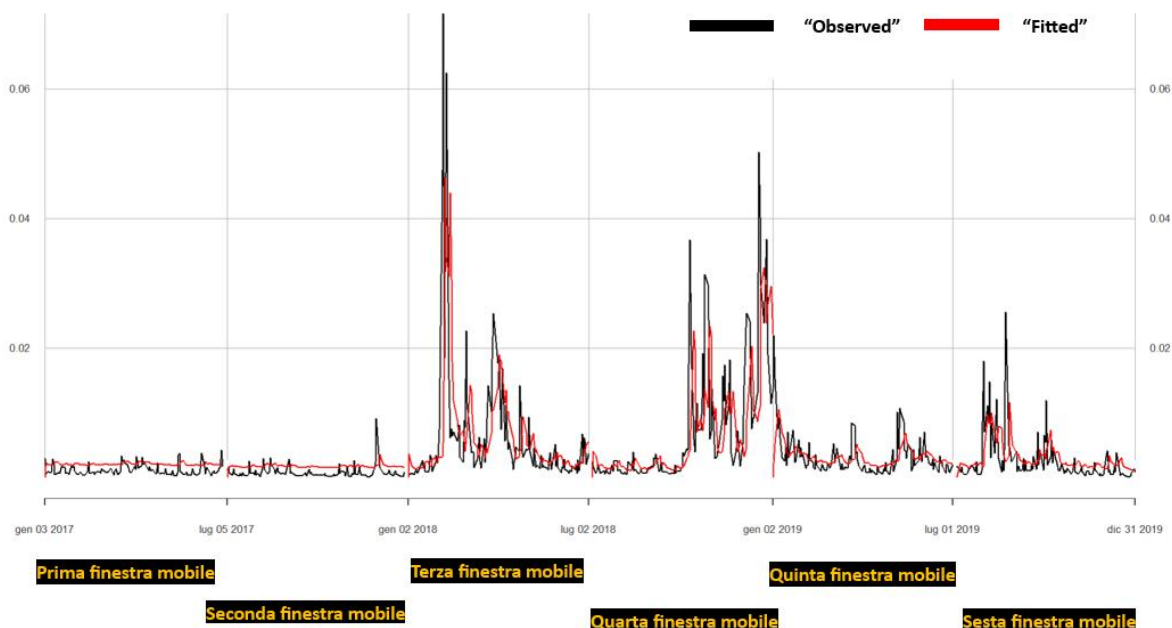


Figura n.29: grafico comparativo realized variance osservata e realized variance predetta indice S&P 500 modello HEAVY-ESN-PSO

Fonte: rielaborazione personale

Al pari del precedente modello, la previsione fornita dal modello ibrido HEAVY-ESN-PSO risulta essere piuttosto fedele e coerente rispetto ai valori osservati. La finestra temporale numero cinque presenta i medesimi profili di criticità evidenziati all'interno del modello ibrido precedente.

La precisione e l'attendibilità del modello ibrido HEAVY-ESN-PSO è stata misurata attraverso la misura del MSE. Di seguito i risultati.

MSE modello HEAVY-ESN-PSO				
Finestre mobili	Titolo EBAY	Titolo MSFT	Titolo CAT	Indice S&P 500
I	5,22E-09	2,07E-09	5,89E-09	9,83E-11
II	4,23E-09	2,06E-09	4,66E-09	1,76E-10
III	1,35E-08	4,74E-08	1,09E-07	9,18E-09
IV	6,56E-08	3,99E-08	3,13E-08	6,62E-09
V	4,08E-09	3,60E-09	1,41E-08	9,32E-10
VI	8,79E-09	3,61E-09	1,08E-08	1,15E-09
Media	1,69E-08	1,64E-08	2,92E-08	3,03E-09

Tabella n.10: risultati mse del modello HEAVY-ESN-PSO

Fonte: rielaborazione personale

Comparando i risultati ottenuti è possibile desumere che le performance ottenute, con riferimento ai titoli EBAY, MSFT e all'indice S&P 500, sono piuttosto vicine tra loro. Non vi è uno scostamento sensibile delle performance.

Con riferimento al titolo CAT, il modello ibrido HEAVY-ESN-PSO è in grado di ottenere performance sensibilmente migliori rispetto al modello ibrido HAR-ESN-PSO facendo registrare un MSE di $2,92e-08$ a fronte di un MSE di $5,11e-06$.

3.8.3 Il modello HARQ-ESN-PSO

La struttura del modello ibrido HARQ-ESN-PSO prevede due blocchi principali: il modello HAR-Q di T. Bollerslev, un'estensione del modello HAR di F. Corsi che introduce la *realized quarticity*, e la *echo state network*.

Ai fini della presente trattazione è stata adottata la versione "semplificata" del modello HARQ ove sono le presenti le tre *realized variance* distinte per orizzonte temporale e la *realized quarticity* a un ritardo temporale.

Innanzitutto, sono stati sviluppati i modelli HARQ.

Modello HAR-Q titolo EBAY						
Coefficienti	1° finestra mobile	2° finestra mobile	3° finestra mobile	4° finestra mobile	5° finestra mobile	6° finestra mobile
Beta 0 (Int.)	$4,66E-05$	$4,32E-05$	$1,64E-05$	$1,75E-05$	$1,73E-05$	$1,90E-05$
Beta 1	$4,75E-01$	$4,77E-01$	$4,04E-01$	$5,58E-01$	$6,00E-01$	$5,87E-01$
Beta 2	$1,19E-01$	$1,29E-01$	$2,28E-01$	$2,70E-01$	$1,14E-01$	$1,30E-01$
Beta 3	$1,44E-01$	$1,48E-01$	$2,26E-01$	$5,68E-02$	$2,00E-01$	$1,80E-01$
Beta 4	$-6,68E+01$	$-6,68E+01$	$-2,74E+02$	$-3,90E+02$	$-1,15E+02$	$-1,13E+02$

Tabella n.11: i coefficienti del modello HARQ titolo EBAY

Fonte: rielaborazione personale

Modello HAR-Q titolo MSFT						
Coefficienti	1° finestra mobile	2° finestra mobile	3° finestra mobile	4° finestra mobile	5° finestra mobile	6° finestra mobile
Beta 0 (Int,)	$1,93E-05$	$1,35E-05$	$1,86E-05$	$1,07E-05$	$1,07E-05$	$1,54E-05$
Beta 1	$8,05E-01$	$8,51E-01$	$3,08E-01$	$8,12E-01$	$8,31E-01$	$7,85E-01$
Beta 2	$8,52E-02$	$1,06E-01$	$5,79E-02$	$9,15E-02$	$1,48E-02$	$8,34E-02$
Beta 3	$6,84E-02$	$8,80E-02$	$3,50E-01$	$1,27E-02$	$1,15E-01$	$4,98E-02$
Beta 4	$-1,38E+02$	$-1,46E+02$	$-1,62E+02$	$-1,98E+02$	$-1,54E+02$	$-1,38E+02$

Tabella n.12: i coefficienti del modello HARQ titolo MSFT

Fonte: rielaborazione personale

Modello HAR-Q titolo CAT						
Coefficienti	1° finestra mobile	2° finestra mobile	3° finestra mobile	4° finestra mobile	5° finestra mobile	6° finestra mobile
Beta 0 (Int,)	1,08E-05	1,57E-05	2,49E-05	2,07E-05	1,83E-05	2,49E-05
Beta 1	5,29E-01	4,89E-01	4,73E-01	4,55E-01	5,13E-01	4,73E-01
Beta 2	2,70E-01	2,82E-01	1,33E-01	9,13E-02	9,25E-02	1,33E-01
Beta 3	1,55E-01	1,57E-01	2,88E-01	3,22E-01	3,35E-01	2,88E-01
Beta 4	-1,58E+02	-1,50E+02	-8,20E+01	-7,11E+01	-9,46E+01	-8,20E+01

Tabella n.13: i coefficienti del modello HARQ titolo CAT

Fonte: rielaborazione personale

Modello HAR-Q indice S&P 500						
Coefficienti	1° finestra mobile	2° finestra mobile	3° finestra mobile	4° finestra mobile	5° finestra mobile	6° finestra mobile
Beta 0 (Int,)	5,05E-06	5,58E-08	1,78E-06	3,21E-06	4,04E-06	6,01E-06
Beta 1	8,99E-01	1,04E+00	4,49E-01	8,29E-01	8,05E-01	7,56E-01
Beta 2	4,15E-02	6,02E-02	2,00E-01	7,31E-02	3,71E-02	1,19E-01
Beta 3	2,05E-02	1,51E-02	2,24E-01	1,84E-02	1,17E-01	3,02E-02
Beta 4	-4,24E+02	-4,44E+02	-2,13E+02	-4,28E+02	-2,95E+02	-2,65E+02

Tabella n.14: i coefficienti del modello HARQ indice S&P 500

Fonte: rielaborazione personale

Da quanto si evince dalle tabelle il numero di coefficienti non significativi è superiore rispetto a quelli del modello HAR di F. Corsi e del modello HEAVY di K. Sheppard e N. Sheppard. Sovente, i coefficienti significativi risultano essere Beta 0, Beta 1 e Beta 4, quest'ultimo riferito alla *realized quarticity*. I coefficienti Beta 2 e Beta 3 presentano alcuni valori non significativi, per i quali non è possibile affermare l'esistenza di correlazione statisticamente significativa tra la variabile dipendente e le relative variabili indipendenti.

Inoltre, è interessante osservare come il coefficiente Beta 4 assuma sempre un valore negativo per tutte le serie storiche e le finestre temporali considerate.

Di seguito i grafici che rappresentano l'evoluzione dei coefficienti nel tempo.

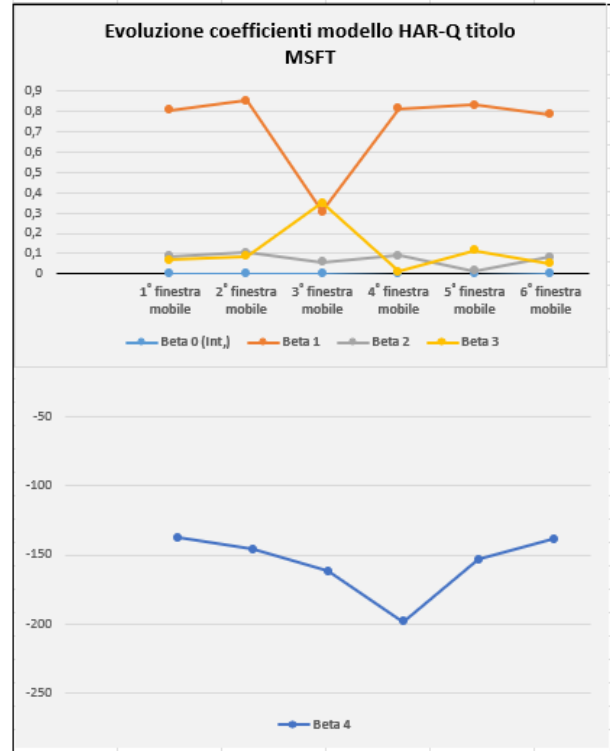
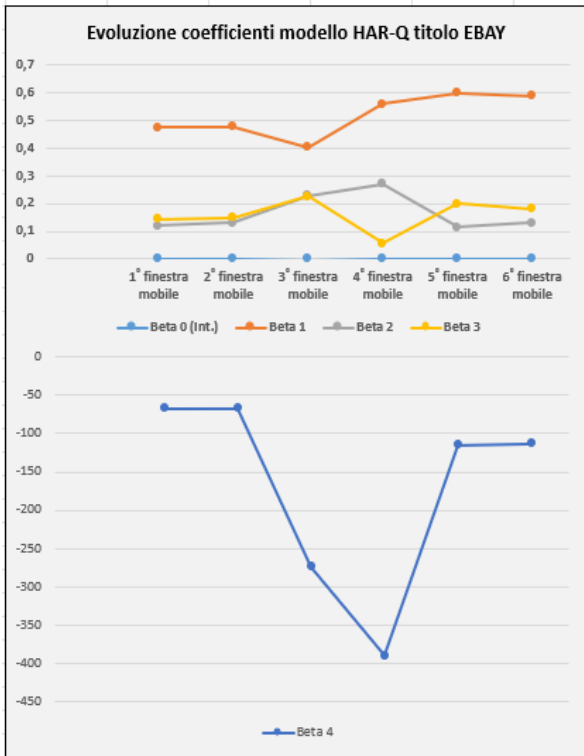


Figura n.30: grafico dell'andamento dei coefficienti del modello HARQ relativo ai titoli EBAY e MSFT

Fonte: rielaborazione personale

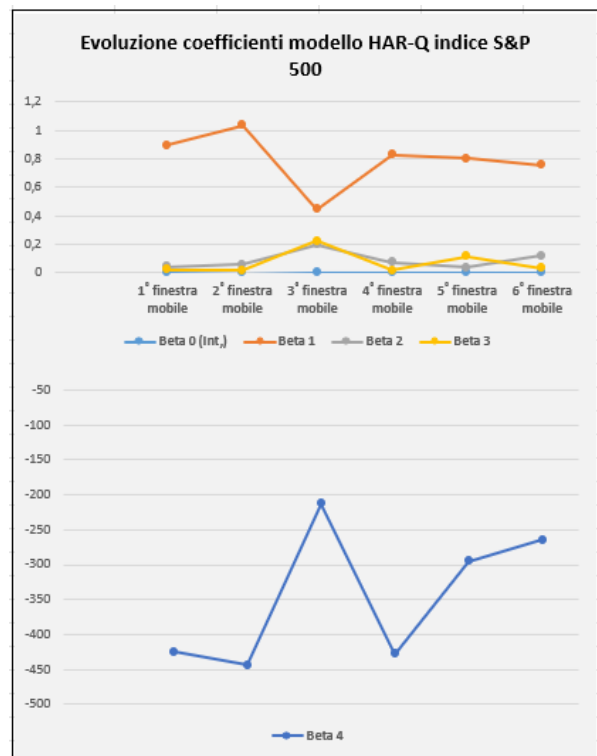
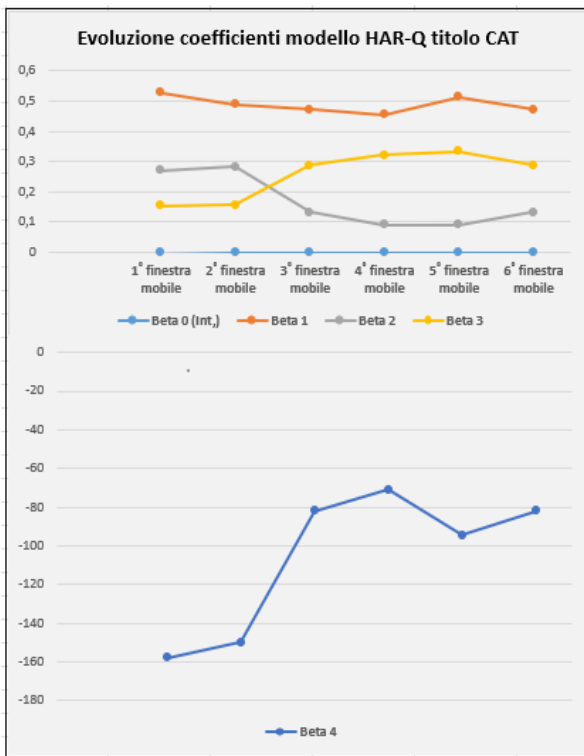


Figura n.31: grafico dell'andamento dei coefficienti del modello HARQ relativo ai titoli CAT e S&P 500

Fonte: rielaborazione personale

L'andamento dei coefficienti beta 0, beta 1, beta 2 e beta 3 appare piuttosto lineare nel tempo sebbene vi siano oscillazioni piuttosto intense relativamente al titolo MSFT e all'indice S&P 500 in coincidenza della finestra temporale numero 3. Per quanto concerne il coefficiente beta 4, associato alla *realized quarticity*, questo presenta un andamento piuttosto ondivago nel tempo.

I grafici rappresentano la bontà della previsione rispetto ai valori osservati nel periodo luglio 2017-dicembre 2019.

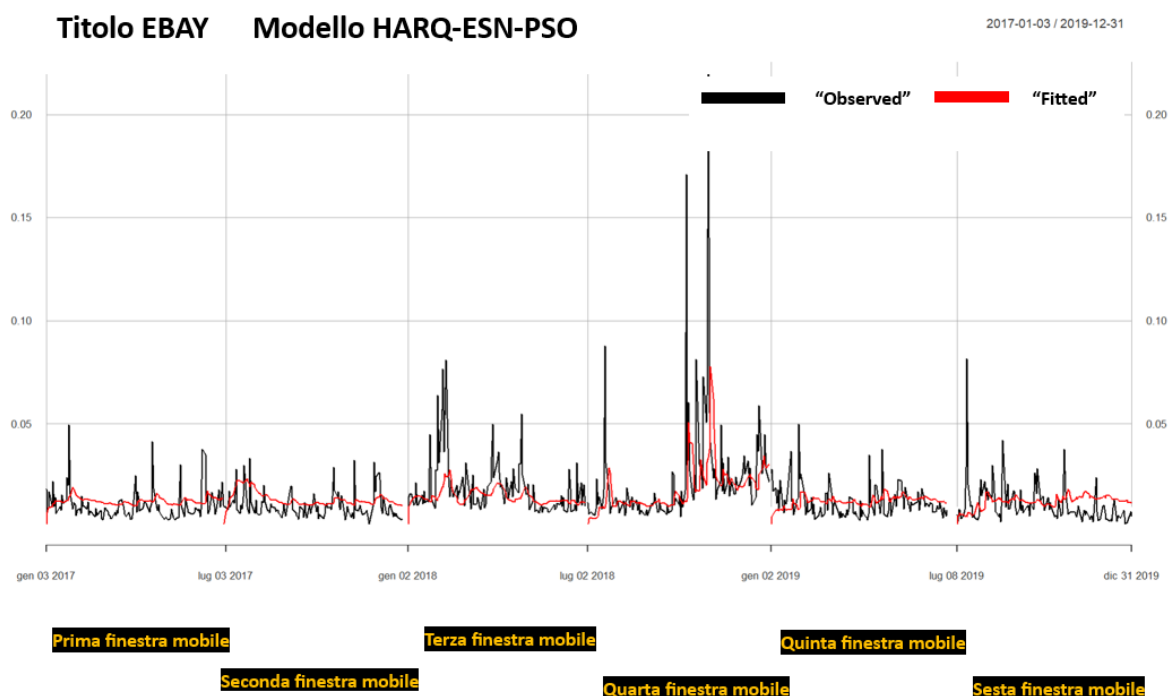


Figura n.32: grafico comparativo realized variance osservata e realized variance predetta titolo EBAY modello HARQ-ESN-PSO
 Fonte: rielaborazione personale

Titolo MSFT Modello HARQ-ESN-PSO

2017-01-03 / 2019-12-31

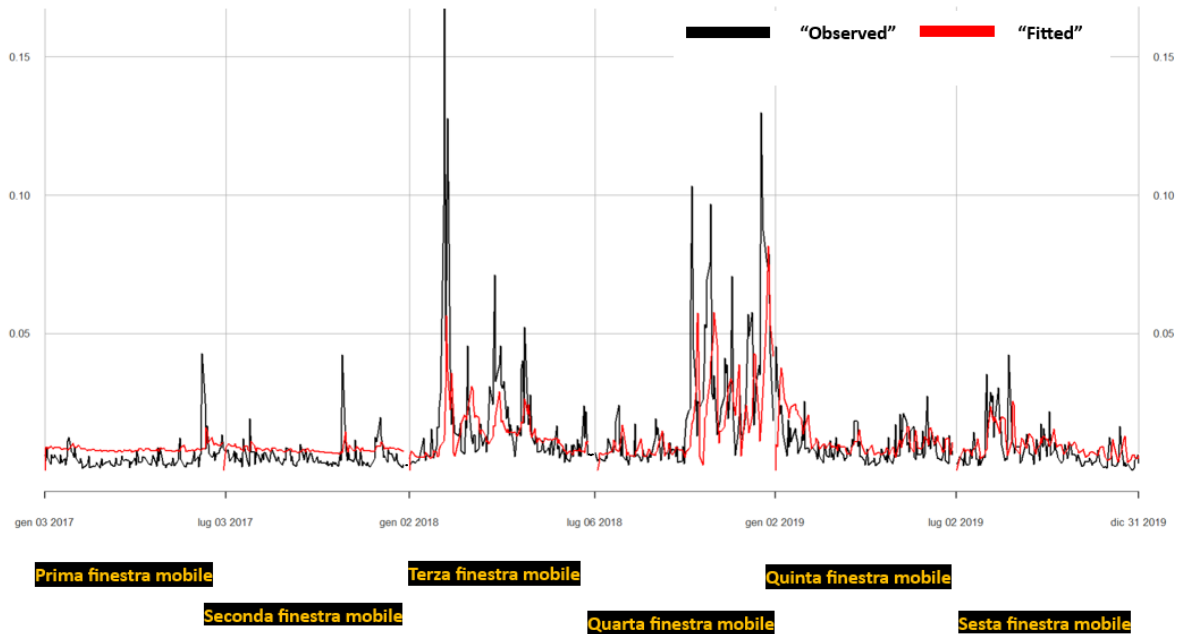


Figura n.33: grafico comparativo realized variance osservata e realized variance predetta titolo CAT modello HARQ-ESN-PSO
Fonte: rielaborazione personale

Titolo CAT Modello HARQ-ESN-PSO

2017-01-03 / 2019-12-31

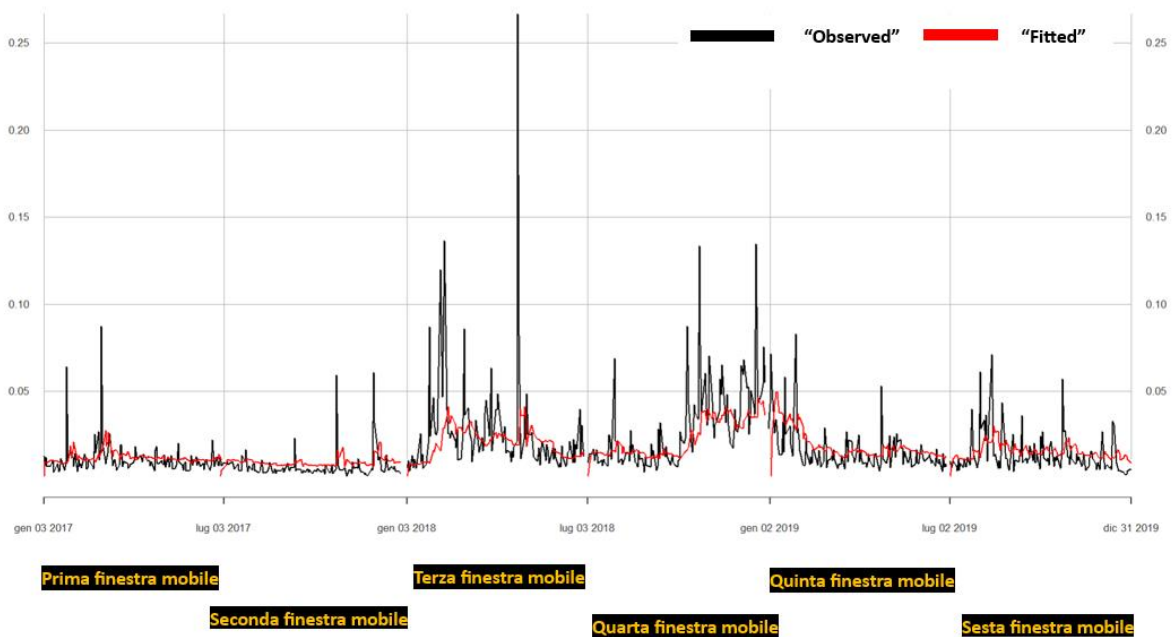


Figura n.34: grafico comparativo realized variance osservata e realized variance predetta titolo MSFT modello HARQ-ESN-PSO
Fonte: rielaborazione personale

Indice S&P 500 Modello HARQ-ESN-PSO

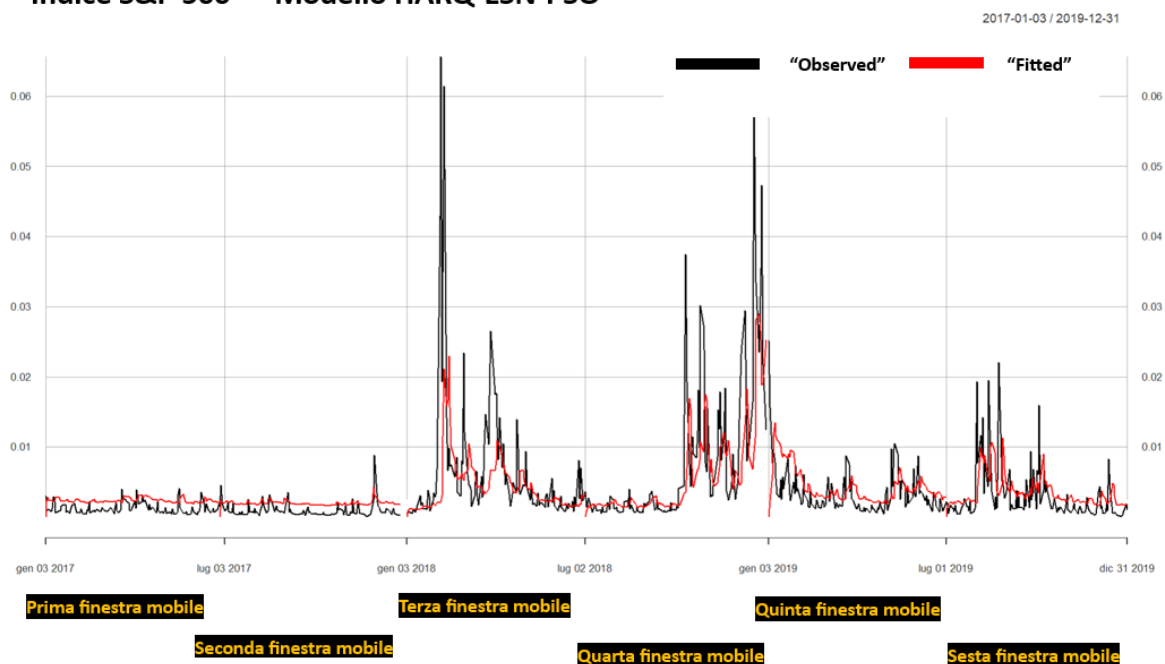


Figura n.35: grafico comparativo realized variance osservata e realized variance predetta indice S&P 500 modello HARQ-ESN-PSO

Fonte: rielaborazione personale

I riscontri grafici delle tabelle appena esposte confermano quanto affermato in precedenza riguardo i modelli ibridi HAR-ESN-PSO e HEAVY-ESN-PSO, ossia che il periodo che presenta le criticità più evidenti risulta essere il mese di gennaio 2019 caratterizzato dalla presenza di numerosi salti di prezzo all'interno delle serie storiche. Di seguito, la tabella riepilogativa che riassume i risultati degli scarti medi quadratici associati a ciascun titolo con riferimento alle sei finestre mobili considerate.

MSE modello HARQ-ESN-PSO				
Finestre mobili	Titolo EBAY	Titolo MSFT	Titolo CAT	Indice S&P 500
I	6,29E-09	3,39E-07	9,55E-09	1,91E-10
II	5,29E-09	3,23E-07	6,94E-09	1,82E-10
III	1,74E-08	5,27E-06	9,66E-08	8,36E-09
IV	7,09E-08	6,07E-06	3,93E-08	7,54E-09
V	7,16E-09	7,85E-07	2,02E-08	1,38E-09
VI	1,18E-08	5,86E-07	1,30E-08	1,36E-09
Media	1,98E-08	2,23E-06	3,09E-08	3,17E-09

Tabella n.15: risultati mse del modello HARQ-ESN-PSO

Fonte: rielaborazione personale

Le performance ottenute dal modello HARQ-ESN-PSO risultano essere coerenti con quanto visto con i precedenti due modelli. Tuttavia le prestazioni relative al titolo MSFT si sostanziano in un errore medio quadratico pari a $2,23e-06$, mentre i modelli ibridi HAR-ESN-PSO e HEAVY-ESN-PSO fanno registrare, rispettivamente, valori di MSE pari a $1,96e-08$ e $1,64e-08$.

Con riferimento ai titoli analizzati nella presente trattazione è possibile affermare che il modello ibrido dal comportamento migliore è il modello HEAVY-ESN-PSO.

3.8.4 Il confronto tra l'algoritmo PSO e l'algoritmo ABC

All'algoritmo del *particle swarm optimization* è stato affiancato l'algoritmo dell'*artificial bee colony*. Difatti, i tre modelli ibridi appena esposti sono stati replicati sostituendo l'algoritmo del PSO con l'algoritmo dell'ABC. Attraverso quest'ultimo algoritmo sono stati affinati i medesimi iperparametri della *echo state network*.

La personalizzazione dell'algoritmo delle api è più scarna rispetto a quella prevista per il metodo di ottimizzazione dello sciame di particelle.

L'algoritmo è stato implementato secondo tale impostazione=

`maxIter= 10 cycleLimit= 120`

Dove il termine "*cycleLimit*" indica il numero di volte consentite affinché la soluzione candidata non si muova.

I risultati ottenuti dall'implementazione dell'algoritmo dell'*artificial bee colony* sono sintetizzati nelle seguenti tabelle.

MSE modello HAR-ESN-ABC				
Finestre mobili	Titolo EBAY	Titolo MSFT	Titolo CAT	Indice S&P 500
I	6,41E-09	2,05E-09	1,19E-06	1,19E-10
II	4,91E-09	2,67E-09	9,36E-07	1,42E-10
III	1,52E-08	4,30E-08	1,19E-05	9,65E-09
IV	7,09E-08	3,93E-08	4,97E-06	6,46E-09
V	6,20E-09	5,05E-09	1,97E-06	6,47E-09
VI	9,73E-09	4,44E-09	1,62E-06	1,11E-09
Media	1,89E-08	1,61E-08	3,77E-06	3,99E-09

Tabella n.16: risultati mse del modello HAR-ESN-ABC

Fonte: rielaborazione personale

MSE modello HEAVY-ESN-ABC				
Finestre mobili	Titolo EBAY	Titolo MSFT	Titolo CAT	Indice S&P 500
<i>I</i>	5,11E-09	2,35E-09	6,09E-09	1,87E-10
<i>II</i>	4,21E-09	1,99E-09	5,20E-09	2,06E-10
<i>III</i>	1,37E-08	5,08E-08	1,18E-07	8,07E-09
<i>IV</i>	6,18E-08	4,13E-08	3,73E-08	5,91E-09
<i>V</i>	4,46E-09	4,16E-09	1,20E-08	1,26E-09
<i>VI</i>	8,23E-09	3,44E-09	1,20E-08	1,35E-09
Media	1,63E-08	1,73E-08	3,18E-08	2,83E-09

Tabella n.17: risultati mse del modello HEAVY-ESN-ABC

Fonte: rielaborazione personale

MSE modello HARQ-ESN-ABC				
Finestre mobili	Titolo EBAY	Titolo MSFT	Titolo CAT	Indice S&P 500
<i>I</i>	4,84E-09	3,79E-07	9,59E-09	8,75E-10
<i>II</i>	5,60E-09	4,22E-07	6,93E-09	9,54E-10
<i>III</i>	2,24E-08	1,16E-06	9,71E-08	1,16E-08
<i>IV</i>	7,16E-08	6,04E-06	3,98E-08	1,43E-08
<i>V</i>	7,00E-09	2,05E-06	2,02E-08	7,32E-10
<i>VI</i>	9,80E-09	5,54E-07	1,29E-08	1,88E-09
Media	2,02E-08	1,77E-06	3,11E-08	5,05E-09

Tabella n.18: risultati mse del modello HARQ-ESN-ABC

Fonte: rielaborazione personale

L'adozione dell'algoritmo dell'*artificial bee colony* con riferimento al modello ibrido HAR-ESN ha condotto a un leggero miglioramento delle performance per quanto concerne i titoli EBAY, MSFT e CAT, mentre a un lieve peggioramento delle prestazioni concernente l'indice S&P 500. Difatti, il valore dell'indicatore MSE relativo all'indice borsistico S&P 500 passa da 3,25e-09 a 3,99e-09 con riferimento al modello HAR-ESN-ABC. In ogni caso, il ragionamento è contestualizzato in termini piuttosto contenuti, ossia nell'ordine di differenze quantificabili in nove o dieci decimali.

Per quanto riguarda il modello HEAVY-ESN-ABC è possibile cogliere un leggero miglioramento delle performance per tutti i titoli considerati rispetto al modello HEAVY-ESN-ABC. Il miglioramento dell'indicatore MSE è piuttosto contenuto e valutabile nell'ordine dei nove o dieci decimali.

Infine, il modello HARQ-ESN-ABC presenta performance del tutto equiparabili rispetto al modello HARQ-ESN-PSO, si eccettua l'indicatore MSE relativo all'indice S&P 500 che passa da un valore di 3,17e-09 a 5,05e-09 con riferimento al modello HARQ-ESN-ABC.

4 Conclusioni: il confronto finale sui modelli ibridi proposti

Il modello ibrido HAR-ESN-PSO di Gabriel Trierweiler Ribeiro et al. rappresenta il benchmark rispetto al quale valutare le performance dei modelli ibridi proposti, l'HEAVY-ESN-PSO e l'HARQ-ESN-PSO. La comparazione dei modelli è stata effettuata sulla base della misura d'errore dello scarto quadratico medio, conosciuto con la dicitura di MSE.

Da quanto si è evinto nel precedente capitolo le performance migliori sono state ottenute dal modello ibrido HEAVY-ESN-PSO, sebbene non vi sia stata una differenza sensibile nei confronti degli altri due modelli ibridi.

Per quanto concerne la scelta dell'algoritmo di ottimizzazione, entrambi gli algoritmi, il *particle swarm optimization* e l'*artificial bee colony*, hanno condotto a risultati piuttosto soddisfacenti e non dissimili fra loro.

Inoltre, la velocità di elaborazione e l'onerosità computazionale associate a entrambi gli algoritmi sono le medesime. Dunque, in termini di dispendio di risorse l'algoritmo dello sciame di particelle e l'algoritmo delle api sono equiparabili.

Alla luce delle considerazioni appena esposte, è possibile ribadire la bontà del modello

ibrido HAR-ESN-PSO, nonché dei modelli proposti HEAVY-ESN-PSO e HARQ-ESN-PSO.

I tre modelli ibridi adottati si sono rivelati particolarmente efficienti nella previsione delle *realized measure* giornaliere riuscendo a cogliere con discreta precisione l'andamento della volatilità nel tempo.

Con riferimento a ognuno dei tre modelli ibridi tratti è stata rilevata una principale criticità riguardante un periodo di tempo circoscritto riferito a tutte e quattro le serie storiche finanziarie considerate. Difatti, in tale periodo la previsione si discosta sensibilmente dai valori osservati a causa, presumibilmente, di una duplice motivazione.

Innanzitutto, tale periodo succede a una fase di persistenza di elevata volatilità, ossia riconducibile al fenomeno del *volatility clustering*. Inoltre, relativamente al periodo considerato tutte le serie storiche finanziarie sono state caratterizzate dal fenomeno del *jump pricing*, ossia salti di prezzo che causano discontinuità all'interno del grafico delle quotazioni. Questi elementi hanno condotto a una minore precisione dei modelli ibridi con riferimento al periodo a cavallo tra dicembre 2018 e gennaio 2019.

I modelli proposti presentano una struttura a tre blocchi principali condividendo la medesima ispirazione, nonché simili capacità predittive. Inoltre, i modelli ibridi analizzati sono equiparabili dal punto di vista dell'onerosità computazionale. Tale aspetto non è trascurabile nel momento in cui si intende comparare dei modelli alternativi.

La struttura ibrida proposta è particolarmente replicabile e personalizzabile sulla base della vasta selezione dei modelli stocastici per la volatilità. Difatti, oltre al modello HAR di F. Corsi e all'estensione HARQ di T. Bollerslev sono stati proposti numerosi modelli alternativi per lo studio della volatilità. Tra le numerose estensioni pubblicate risaltano i modelli HARQJ, CHARQ e HARCJ. Tali modelli affondano le proprie radici sulle conoscenze del modello HAR integrando il numero di *realized measure* coinvolte. Le stesse *realized measure* rappresentano un elemento estremamente declinabile e diversificabile. Difatti, nel corso degli anni, la letteratura ha proposto una corposa scelta di stimatori della volatilità fondati sui dati ad alta frequenza. La natura di tali stimatori è suscettibile in base alle finalità di utilizzo e alla struttura delle serie storiche finanziarie analizzate.

Vi sono, ad esempio, serie storiche caratterizzate dalla presenza di rumore o di salti di

prezzo per le quali sono indicate determinate *realized measure*, quali la *realized kernel* e la *realized bi-power variation*.

Per quanto concerne l'elemento della *echo state network* è possibile affermare che trattasi di una rete neurale artificiale particolarmente indicata per la previsione di serie storiche finanziarie, tuttavia l'impiego di eventuali estensioni e architetture alternative (brevemente illustrate durante il capitolo 2, paragrafo 2.7) potrebbe rappresentare un'interessante evoluzione dei modelli ibridi illustrati nella presente trattazione.

Infine, per quanto concerne l'algoritmo di ottimizzazione utile all'affinamento degli iperparametri è fondamentale ribadire quanto sia ampio il novero dei metodi di ottimizzazione implementabili. Difatti, la letteratura in materia ha proposto un ampio ventaglio di algoritmi genetici ed evolutivi che trovano diffusa applicazione in ambito di *machine learning* e *deep learning*. L'inarrestabile proliferazione di tecniche di apprendimento automatico rappresenta un contributo fondamentale alla ricerca in materia di intelligenza artificiale applicata alla finanza.

Alla luce di queste considerazioni, è possibile affermare quanto i modelli proposti possano rappresentare un punto di partenza per l'analisi in materia di volatilità di serie storiche finanziarie, piuttosto che un semplice punto d'arrivo. Lo sviluppo delle nuove conoscenze in tema di machine learning, le recenti innovazioni tecnologiche e la diffusione di dati a frequenze sempre più elevate rappresentano un'opportunità per la ricerca e lo sviluppo di modelli ibridi volti allo studio e all'analisi di volatilità di serie storiche finanziarie. L'implementazione di questi fattori garantirà un successivo avanzamento nel perfezionamento di tali modelli raggiungendo prestazioni sempre migliori e previsioni sempre più precise.

5 Appendice: I comandi utilizzati su R

Il presente capitolo è dedicato alla trasposizione dei comandi presenti all'interno dello script di R utilizzato con riferimento al titolo EBAY. Gli script riferiti alle serie storiche finanziarie di Microsoft, Caterpillar e S&P 500 sono del tutto analoghi.

La preparazione dei dati prevede il caricamento dei pacchetti utili ai fini dei calcoli e delle funzioni implementate, nonché il caricamento dei dati infra-giornalieri riferiti al titolo EBAY.

```
library(quantmod)
library(HARModel)
library(metaheuristicOpt)
library(rESN)
library(highfrequency)
library(data.table)
library(xts)
```

```
EBAY
EBAY=as.data.table(EBAY)
EBAY=EBAY[,c(1,5)]
EBAY
EBAY=EBAY[157:97907]
```

```
EBAY
colnames(EBAY)=c("DT","PRICE")
head(EBAY)
```

```
EBAY.d=diff(log(EBAY$PRICE)[-1])
EBAY.d
```

5.1 Il modello HAR-ESN-PSO

Stima della realized variance

```
EBAYrv=rRVar(rData=EBAY[,list(DT, PRICE)], alignBy = "minutes", alignPeriod = 5,
makeReturns = TRUE)
EBAYrv
EBAYrv=as.xts(EBAYrv)
```

Creazione di 6 rolling window dalla durata di 30 mesi cadauna

```
EBAYm1rv=EBAYrv[1:629]
EBAYm2rv=EBAYrv[125:755]
EBAYm3rv=EBAYrv[253:880]
EBAYm4rv=EBAYrv[378:1006]
EBAYm5rv=EBAYrv[505:1130]
EBAYm6rv=EBAYrv[630:1258]
```

```
EBAYrv=EBAYm1rv
EBAYrv=EBAYm2rv
EBAYrv=EBAYm3rv
EBAYrv=EBAYm4rv
EBAYrv=EBAYm5rv
EBAYrv=EBAYm6rv
```

Distinzione in training set e validation set della realized variance

```
train_proportion=0.8
1006*0.8
split_index=round(train_proportion * length(EBAYrv))
#training set
training_set=EBAYrv[1:split_index,]
#validation set
validation_set=EBAYrv[(split_index + 1):length(EBAYrv),]
training_set
validation_set
```

Training della rete

```
EBAYHAR=HARmodel(data = training_set , periods = c(1,5,22), RVest =
c("rCov"),type = "HAR", h = 1, transform = NULL, inputType = "RM")
EBAYHAR$model
summary(EBAYHAR)
```

Preparazione dati di input da immettere nella rete

```
input=as.xts(EBAYHAR$model[-1])
input
dataU=matrix(input,nrow=480,ncol=3,byrow=F)
dataU

dataY=EBAYHAR$model$y
dataY=matrix(dataY,nrow=480,ncol=1)
dataY
```

Inizializzazione della ESN

```
EBAYESN=newESN(dataY,dataU,n.neurons = 50L, density = 0.1, back.density =  
0,leak.rate = 0.3 , lambda = 1e-07, spectral.radius = 1.15)
```

```
EBAYESN
```

```
EBAYESN=train(EBAYESN)
```

```
EBAYESN
```

Previsione della rete

```
#validation set per le 3 varianze (giornaliera, settimanale e mensile)
```

```
validset=EBAYrv[481:648]
```

```
validset
```

```
validset_data=HARmodel(data = validset, periods = c(1,5,22), RVest = c("rCov"),type  
= "HAR", h = 1, transform = NULL, inputType = "RM")
```

```
Vdataset=validset_data$model
```

```
input1=as.xts(Vdataset[,-1])
```

```
input1
```

```
dataUU=matrix(input1,nrow=125,ncol=3,byrow=F)
```

```
dataUU
```

```
ESNpredict=predict(EBAYESN,dataUU)
```

```
ESNpredict
```

```
actual=EBAYrv[525:629]
```

```
length(actual)
```

```
mse=mean((actual -ESNpredict)^2)
```

mse

Grafico comparativo della previsione

```
ESNpred=as.data.table(ESNpredict)
ESNpred=cbind(actual,ESNpred$V1)
ESNpred=ESNpred[,-1]
ESNpred=as.xts(ESNpred)
ESNpred
```

```
plot(actual,type="l",col="black")
lines(ESNpred,lwd=2,col="red")
```

Ottimizzazione con l' algoritmo PSO

```
rangeVar3=matrix(c(1e-07,0.9999999,1e-07,1,0.1,1.5), ncol = 3, nrow = 2, byrow =
FALSE)
```

```
rangeVar3
```

```
numVar3=3
```

```
numVar3
```

```
fitness=function(x){
  Leak_Rate=x[1]
  Lambda=abs(x[2])
  Spec.rad=abs(x[3])
```

```
EBAYESN=newESN(dataY, dataU, n.neurons = 50L, density = 0.1, back.density =
0, leak.rate = Leak_Rate, lambda = Lambda, spectral.radius = Spec.rad)
```

```
EBAYESNtrained=train(EBAYESN)
```

```
ESNpred=predict(EBAYESNtrained, dataUU)
```

```
mse=mean((actual-ESNpred)^2, na.rm = TRUE)
```

```

return(mse)
}

result_PSO3_EBAY=matrix(NA,nrow=50,ncol=3)
for(i in 1:50){
  result_PSO3_EBAY[i,]=PSO(fitness, optimType ="MIN", numVar = numVar3,
numPopulation = 40, maxIter = 50, rangeVar = rangeVar3, Vmax = 2, ci = 1.5, cg =
1.5, w = 0.7)
}

print(result_PSO3_EBAY)
EBAY_PSO3=print(result_PSO3_EBAY)

par(mfrow = c(1, 3))
plot(density(EBAY_PSO3[,1]),xlim=c(0,1))
plot(density(EBAY_PSO3[,2]),xlim=c(0,1))
plot(density(EBAY_PSO3[,3]),xlim=c(0.1,1.6))

```

Ottimizzazione con l'algoritmo ABC

```

fitness=function(x){
  Leak_Rate=x[1]
  Lambda=x[2]
  Spec.rad=x[3]

  EBAYESN=newESN(dataY, dataU, n.neurons = 50L, density = 0.1, back.density =
0, leak.rate = Leak_Rate, lambda = Lambda, spectral.radius = Spec.rad)
  EBAYESNtrained=train(EBAYESN)
  ESNpred=predict(EBAYESNtrained, dataUU)
  mse=mean((actual - ESNpred)^2, na.rm = TRUE)
  return(mse)
}

```

```

result_ABC3_EBAY=matrix(NA,nrow=50,ncol=3)
for(i in 1:50){
  result_ABC3_EBAY[i,]=ABC(fitness, optimType = "MIN", numVar = numVar3,
numPopulation = 40, maxIter = 10, rangeVar = rangeVar3)
}

print(result_ABC3_EBAY)

EBAY_ABC3=result_ABC3_EBAY
plot(density(EBAY_ABC3[,1]),xlim=c(0,1))
plot(density(EBAY_ABC3[,2]),xlim=c(0,1))
plot(density(EBAY_ABC3[,3]),xlim=c(0,1.6))

```

5.2 Modello ibrido HEAVY-ESN-PSO

Stima della realized bi-power variation

```

EBAYbpv=rBPCov(rData = EBAY[, list(DT, PRICE)], alignBy = "minutes", alignPeriod
= 5, makeReturns = TRUE)
is.Date(EBAYbpv$DT)
BPVdt=as.Date(EBAYbpv$DT)
BPVdt=as.xts(BPVdt)
EBAYbpv=cbind(BPVdt,EBAYbpv$BPV)
EBAYbpv=as.xts(EBAYbpv)
EBAYbpv

```

Rolling window della realized bi-power variation

```
EBAYm1bpv=EBAYbpv[1:629]
EBAYm2bpv=EBAYbpv[127:755]
EBAYm3bpv=EBAYbpv[252:880]
EBAYm4bpv=EBAYbpv[378:1006]
EBAYm5bpv=EBAYbpv[502:1130]
EBAYm6bpv=EBAYbpv[630:1258]
```

```
EBAYbpv=EBAYm1bpv
EBAYbpv=EBAYm2bpv
EBAYbpv=EBAYm3bpv
EBAYbpv=EBAYm4bpv
EBAYbpv=EBAYm5bpv
EBAYbpv=EBAYm6bpv
```

Calcolo prezzi giornalieri del titolo EBAY

```
CHIUSURA=data.table(aggregate(PRICE~cut(DT, breaks = "1 day"),EBAY,last))
colnames(CHIUSURA)=c("DT","PRICE")
CHIUSURA=as_tibble(CHIUSURA)
CHIUSURA=as.data.table(CHIUSURA)
CHIUSURAdt=as.Date(CHIUSURA$DT)
CHIUSURA=cbind(CHIUSURAdt,CHIUSURA)
CHIUSURA=CHIUSURA[,-2]
CHIUSURA=as.xts(CHIUSURA)
CHIUSURA
```

Rolling window dei prezzi di chiusura EBAY


```
CHIUSURA1=CHIUSURA[1:629]
CHIUSURA2=CHIUSURA[127:755]
CHIUSURA3=CHIUSURA[252:880]
CHIUSURA4=CHIUSURA[378:1006]
CHIUSURA5=CHIUSURA[502:1130]
CHIUSURA6=CHIUSURA[630:1258]
```

```
CHIUSURA=CHIUSURA1
CHIUSURA=CHIUSURA2
CHIUSURA=CHIUSURA3
CHIUSURA=CHIUSURA4
CHIUSURA=CHIUSURA5
CHIUSURA=CHIUSURA6
```

Suddivisione in training set e validation set della bi-power variation

```
train_proportion=0.8
```

```
split_index=round(train_proportion * length(EBAYbpv))
```

```
#training set
```

```
training_setBPV=EBAYbpv[1:split_index,]
```

```
#validation set
```

```
validation_setBPV=EBAYbpv[(split_index + 1):length(EBAYbpv),]
```

```
training_setBPV
```

```
validation_setBPV
```

```
DATA=EBAYbpv$DT
```

Suddivisione in training set e validation set dei prezzi di chiusura

```
train_proportion=0.8
```

```
629*0.8
```

```
training_setCHIUSURA=CHIUSURA[1:503]
```

```
validation_setCHIUSURA=CHIUSURA[504:629]
```

Preparazione dati da immettere nella ESN

```
logRetTrain=100*makeReturns(training_setCHIUSURA)[-1]
```

```
DATAtrain=DATA[1:503]
```

```
DATAtrain=as.Date(DATAtrain)
```

```
length(logRetTrain)
```

```
length(DATAtrain)
```

```
length(training_setBPV$EBAY_bpv.BPV)
```

```
dataEBAYtrain=xts::xts(cbind(logRetTrain,training_setBPV$EBAY_bpv.BPV[-1]*10000),order.by=DATAtrain[-1])
```

```
EBAYHEAVYtrain=HEAVYmodel(dataEBAYtrain)
```

```
EBAYHEAVYtrain$coefficients
```

```
EBAYHEAVYtrain$RMCondVariances
```

```
plot(EBAYHEAVYtrain)
```

Training della rete

```
dataU_HEAVY=(EBAYHEAVYtrain$RMCondVariances*(1/10000))
dataU_HEAVY=matrix(dataU_HEAVY,ncol=1,nrow=502)
```

```
dataY_HEAVY=training_setBPV$EBAY_bpv.BPV[-1]
dataY_HEAVY=matrix(dataY_HEAVY,ncol=1,nrow=502)
```

```
HEAVYESN=newESN(dataY_HEAVY,dataU_HEAVY,n.neurons = 50L, density = 0.1,
back.density = 0,leak.rate = 0.0000001, lambda = 1e-07, spectral.radius = 1.3)
HEAVYESN=train(HEAVYESN)
HEAVYESN
```

Previsione della rete

```
logRetVal=100*makeReturns(validation_setCHIUSURA$PRICE)[-1]
```

```
DATAval=DATA[504:629]
DATAval
```

```
length(logRetVal)
length(validation_setBPV)
length(DATAval)
```

```
dataEBAYval=xts::xts(cbind(logRetVal,validation_setBPV$EBAY_bpv.BPV[-
1]*10000),order.by=DATAval[-1])
EBAYHEAVYval=HEAVYmodel(dataEBAYval)
EBAYHEAVYval
```

```
dataUU_H=matrix(EBAYHEAVYval$RMCondVariances*(1/10000),ncol=1,nrow=125)
dataUU_H
```

```
HEAVYpredict=predict(HEAVYESN,u=dataUU_H)
HEAVYpredict
```

```
actualHEAVY=EBAYbpv[505:629]
actualHEAVY
```

```
mseHEAVY=mean((actualHEAVY-HEAVYpredict)^2)
mseHEAVY
```

Grafico comparativo della previsione

```
HEAVYpred=as.data.table(HEAVYpredict)
HEAVYpred=cbind(actualHEAVY,HEAVYpred$V1)
HEAVYpred=HEAVYpred[,-1]
HEAVYpred=as.xts(HEAVYpred)
HEAVYpred
```

```
plot(actualHEAVY,type="l",col="black")
lines(HEAVYpred,lwd=2,col="red")
```

Ottimizzazione con l' algoritmo PSO

```
rangeVar3=matrix(c(1e-07,0.999999,1e-07,1,0.1,1.5), ncol = 3, nrow = 2, byrow =
FALSE)
```

```
rangeVar3
```

```
numVar3=3
```

```
numVar3
```

```
fitness=function(x){
  Leak_Rate=x[1]
  Lambda=x[2]
```

```
Spec.rad=x[3]
```

```
HEAVYESN=newESN(dataY_HEAVY,dataU_HEAVY, n.neurons = 50L, density  
=0.1, back.density = 0, leak.rate = Leak_Rate, lambda = Lambda, spectral.radius =  
Spec.rad)  
HEAVYESNtrained=train(HEAVYESN)  
HEAVYESNpred=predict(HEAVYESNtrained, dataUU_H)  
mse=mean((actualHEAVY-HEAVYESNpred)^2, na.rm = TRUE)  
return(mse)  
}
```

```
HEAVYresult_PSO3=matrix(NA,nrow=50,ncol=3)  
for(i in 1:200){  
  HEAVYresult_PSO3[i,]=PSO(fitness, optimType ="MIN", numVar = numVar3,  
numPopulation = 40, maxIter =10, rangeVar = rangeVar3, Vmax = 2, ci = 1.5, cg =  
1.5, w = 0.7)  
}
```

```
print(HEAVYresult_PSO3)  
HEAVY_PSO3=HEAVYresult_PSO3
```

```
plot(density(HEAVY_PSO3[,1]),xlim=c(0,1))  
plot(density(HEAVY_PSO3[,2]),xlim=c(0,1))  
plot(density(HEAVY_PSO3[,3]),xlim=c(0,1.6))
```

Ottimizzazione con l'algoritmo ABC

```
fitness=function(x){  
  
  Leak_Rate=x[1]  
  Lambda=x[2]
```

```
Spec.rad=x[3]
```

```
HEAVYESN=newESN(dataY_HEAVY,dataU_HEAVY, n.neurons = 50L, density  
=0.1, back.density = 0, leak.rate = Leak_Rate, lambda = Lambda, spectral.radius =  
Spec.rad)
```

```
HEAVYESNtrained=train(HEAVYESN)
```

```
HEAVYESNpred=predict(HEAVYESNtrained, dataUU_H)
```

```
mse=mean(( actualHEAVY-HEAVYESNpred)^2, na.rm = TRUE)
```

```
return(mse)
```

```
}
```

```
HEAVYresult_ABC3=matrix(NA,nrow=50,ncol=3)
```

```
for(i in 1:50){ HEAVYresult_ABC3[i,]=ABC(fitness, optimType ="MIN", numVar =
```

```
numVar3, numPopulation = 40, maxIter = 10, rangeVar = rangeVar3)
```

```
}
```

```
print(HEAVYresult_ABC3)
```

```
HEAVY_ABC3=HEAVYresult_ABC3
```

```
plot(density(HEAVY_ABC3[,1]))
```

```
plot(density(HEAVY_ABC3[,2]))
```

```
plot(density(HEAVY_ABC3[,3]))
```

5.3 Modello ibrido HARQ-ESN-PSO

Rolling window della realized variance

```
EBAYrv=rRVar(rData=EBAY[,list(DT, PRICE)], alignBy = "minutes", alignPeriod = 5 ,  
makeReturns = TRUE)
```

```
EBAYrv=as.xts(EBAYrv)
```

```
EBAYrv
```

```
EBAYm1rv=EBAYrv[1:629]
EBAYm2rv=EBAYrv[125:755]
EBAYm3rv=EBAYrv[253:880]
EBAYm4rv=EBAYrv[378:1006]
EBAYm5rv=EBAYrv[505:1130]
EBAYm6rv=EBAYrv[630:1258]
```

```
EBAYrv=EBAYm1rv
EBAYrv=EBAYm2rv
EBAYrv=EBAYm3rv
EBAYrv=EBAYm4rv
EBAYrv=EBAYm5rv
EBAYrv=EBAYm6rv
```

Calcolo dei rendimenti infra-giornalieri per stimare le realized measure

```
datQ=as.xts(EBAY[, makeReturns(PRICE), by = list(DATE = as.Date(DT))])
datQ
length(datQ)
is.na(datQ)
is.infinite(datQ)
is.nan(datQ)
datQ=na.omit(datQ)
datQ
```

Rolling window dei rendimenti infra-giornalieri

```
datQ1=datQ[1:48961]
datQ2=datQ[9673:58721]
datQ3=datQ[19589:68470]
```

```
datQ4=datQ[29339:78195]
datQ5=datQ[39212:87867]
datQ6=datQ[48962:97751]
```

```
datQ=datQ1
datQ=datQ2
datQ=datQ3
datQ=datQ4
datQ=datQ5
datQ=datQ6
```

Suddivisione in training set e validation set dei rendimenti infra-giornalieri

```
train_proportion=0.8
```

```
split_index=round(train_proportion * length(datQ))
```

```
#training set
```

```
training_setRQ=datQ[1:split_index,]
```

```
#validation set
```

```
validation_setRQ=datQ[(split_index + 1):length(datQ),]
```

```
training_setRQ
```

```
validation_setRQ
```

Preparazione dati da immettere nella ESN

```
EBAYHARQtrain=HARmodel(training_setRQ, periods = c(1,5,22), periodsJ =
c(1,5,22),periodsQ = c(1), RVest = c("rCov", "rQuar"),type="HARQ", inputType =
```



```
"returns")
summary(EBAYHARQtrain)
EBAYHARQtrain$coefficients
EBAYHARQtrain$model
```

Training della rete

```
input2=EBAYHARQtrain$model[,-1]
input2=as.xts(input2)

dataUQ=matrix(input2,nrow=480,ncol=4,byrow=F)

inputY=as.xts(EBAYHARQtrain$model)
inputY=inputY[,1]
dataYQ=matrix(inputY,nrow=480,ncol=1)
```

```
HARQESN=newESN(dataYQ,dataUQ,n.neurons = 50L, density = 0.1, back.density =
0,leak.rate = 0.8, lambda = 1e-07, spectral.radius = 1.5)
HARQESN=train(HARQESN)
HARQESN
```

Previsione della rete

```
validatsetRQ1=datQ[37495:48961]
validatsetRQ2=datQ[37495:48961]
validatsetRQ3=datQ[37403:48881]
validatsetRQ4=datQ[37383:48856]
validatsetRQ5=datQ[37271:48655]
validatsetRQ6=datQ[37323:48789]
```

```
EBAYHARQval=HARmodel(validatsetRQ1, periods = c(1,5,22), periodsJ =  
c(1,5,22),periodsQ = c(1), RVest = c("rCov", "rQuar"),type="HARQ", inputType =  
"returns")  
summary(EBAYHARQtrain)  
EBAYHARQval$model
```

```
dataUUQ=EBAYHARQval$model[,-1]  
dataUUQ=as.xts(dataUUQ)  
dataUUQ=matrix(dataUUQ,nrow=125,ncol=4,byrow=F)  
dataUUQ
```

```
QESNpredict=predict(HARQESN,dataUUQ)  
QESNpredict
```

```
actualHARQ=EBAYrv[505:629]  
actualHARQ
```

```
mseHARQ=mean((actualHARQ-QESNpredict)^2)  
mseHARQ
```

Grafico comparativo della previsione

```
QESNpred=as.data.table(QESNpredict)  
QESNpred=cbind(actualHARQ,QESNpred$V1)  
QESNpred=QESNpred[,-1]  
QESNpred=as.xts(QESNpred)  
QESNpred
```

```
plot(actualHARQ,type="l",col="black")  
lines(QESNpred,lwd=2,col="red")
```

Ottimizzazione con l'algoritmo PSO

```
rangeVar3=matrix(c(1e-07,0.999999,1e-07,1,0.1,1.5), ncol = 3, nrow = 2, byrow = FALSE)
```

```
rangeVar3
```

```
numVar3=3
```

```
numVar3
```

```
fitness=function(x){
```

```
    Leak_Rate=x[1]
```

```
    Lambda=x[2]
```

```
    Spec.rad=x[3]
```

```
HARQESN=newESN(dataYQ, dataUQ, n.neurons = 50L, density =0.1, back.density =0,leak.rate = Leak_Rate, lambda = Lambda, spectral.radius = Spec.rad)
```

```
HARQESNtrained=train(HARQESN)
```

```
HARQESNpred=predict(HARQESNtrained, dataUUQ)
```

```
mse=mean((actualHARQ-HARQESNpred)^2, na.rm = TRUE)
```

```
return(mse)
```

```
}
```

```
HARQresult_PSO3=matrix(NA,nrow=50,ncol=3)
```

```
for(i in 1:50){
```

```
    HARQresult_PSO3[i,]=PSO(fitness, optimType = "MIN", numVar = numVar3,
```

```
    numPopulation = 40, maxIter =10, rangeVar = rangeVar3, Vmax = 2, ci = 1.5, cg =
```

```
    1.5, w = 0.7)
```

```
}
```

```
print(HARQresult_PSO3)
```

```
HARQ_PSO3=HARQresult_PSO3
```

```
plot(density(HARQ_PSO3[,1]),xlim=c(0,1))
```

```
plot(density(HARQ_PSO3[,2]),,xlim=c(0,1))
plot(density(HARQ_PSO3[,3]),,xlim=c(0.1,1.6))
```

Ottimizzazione con l' algoritmo ABC

```
fitness=function(x){
  Leak_Rate=x[1]
  Lambda=x[2]
  Spec.rad=x[3]

  HARQESN=newESN(dataYQ, dataUQ, n.neurons = 50L, density =0.1, back.density
= 0, leak.rate = Leak_Rate, lambda = Lambda, spectral.radius = Spec.rad)
  HARQESNtrained=train(HARQESN)
  HARQESNpred=predict(HARQESNtrained, dataUUQ)
  mse=mean((actualHARQ-HARQESNpred)^2, na.rm = TRUE)
  return(mse)
}

HARQresult_ABC3=matrix(NA,nrow=50,ncol=3)
for(i in 1:50){
  HARQresult_ABC3[i,]=ABC(fitness, optimType ="MIN", numVar = numVar3,
numPopulation = 40, maxIter = 10, rangeVar = rangeVar3)
}

print(HARQresult_ABC3)

HARQ_ABC3=HARQresult_ABC3
plot(density(HARQ_ABC3[,1]))
plot(density(HARQ_ABC3[,2]))
plot(density(HARQ_ABC3[,3]))
```

Bibliografia

G. Trierweiler Ribeiro, A. Alves Portela Santos, V. Cocco Mariani, L. Dos Santos Coelho (2021). Novel hybrid model based on echo state neural network applied to the prediction of stock price return volatility. *Journal of econometrics*, vol 184/2021. Elsevier.

C. Sun, M. Song, S. Hong, H. Li (2020). *A review of design and applications of echo state networks*. Tratto dall'archivio della Cornell University.

C. Gallicchio, A. Micheli (2017). Echo state property of deep reservoir computing. *Cognitive Computation*, vol.9/2017.

F. Corsi (2009), A Simple Approximate Long-Memory Model of Realized Volatility. *Journal of financial econometrics*, vol.7/2009. Elsevier.

N. Shephard, K. Sheppard (2010). Realising the future: forecasting with high-frequency based volatility (HEAVY) models. *Journal of applied econometric*, vol.25/2010. Elsevier.

G. M. Gallo, B. Pacini (2002). *Metodi quantitativi per i mercati finanziari*. Carocci editore.

D. Karaboga, B. Basturk(2007). Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems. *Foundations of fuzzy logic and soft computing*, 2007. Springer.

T. Bollerslev, A. J. Pattona, R. Quaedvielig (2016). Exploiting the errors: A simple approach for improved volatility forecasting. *Journal of econometrics* 192/2016. Elsevier.

C. A.E. Goodhart, M. O'hara (1997). High frequency data in financial markets: Issues and applications. *Journal of empirical finance*, volume 4, 1997. Elsevier.

G. K. Venayagamoorthy B. ShishiR (2009). Effects of spectral radius and settling time in the performance of echo state networks. *Neural Networks* , volume 22, 2009. Elsevier.

N. Chouikhi, B. Ammar, A. M. Alimi (2018). *Genesis of Basic and Multi-Layer Echo State Network Recurrent Autoencoder for Efficient Data Representations*. Tratto dall'archivio della Cornell University.

M. E. Mancino, S. Sanfelici (2012). Estimation of Quarticity with High Frequency Data. *Quantitative Finance*, volume 12, 2012. Taylor&Francis Online.

H. Kawakatsu (2022). *Modeling Realized Variance with Realized Quarticity*. MDPI

A. Carlstein (2019). *Stochastic calculus: understanding brownian motion and quadratic variation*. Tratto dall'archivio dell'University of Chicago.

O. E. Barndorff-Nielsen, N. Shephard (2002). Estimating quadratic variation using realized variance. *Journal of applied econometrics*, volume 17, 2002. Wiley.

J. Da Fonseca, W. Zhang (2019). Volatility of volatility is (also) rough. *The journal of future markets*, volume 39, 2019. Wiley.

Y. Qiu, X. Zhang, T. Xie, S. Zhao (2019). Versatile HAR model for realized volatility: A least square model averaging perspective. *Journal of Management Science and Engineering*, volume 4, 2019. SciencePress.

F. Audrino, F. Corsi (2009). Modeling tick-by-tick realized correlations. *Computational Statistics and Data Analysis* volume 54, 2009. Elsevier.

Y. Ait-Sahalia, J. Yu (2009). High frequency market microstructure noise estimates

and liquidity measures. *The annals of applied statistics, volume 3, 2009*. Institute of mathematical statistics.

T. Lux, M. Marchesi (1999). Scaling and Criticality in a Stochastic Multiagent Model of a Financial Market. *Nature, volume 397, 1999*. Springer Nature

G. O. Zumbach, P. Lynch (2003). *Heterogeneous Volatility Cascade in Financial Markets*. Olsen & Associates.

Sitografia

Sito web di ScienceDirect (consultato in data 26/07): <https://www.sciencedirect.com/>.

Sito web di ResearchGate (consultato in data 20/07): <https://www.researchgate.net/>.

Sito web della Borsa Italiana (consultato in data 17/07):
<https://www.borsaitaliana.it/homepage/homepage.htm>.

Sito web di Bloomberg (consultato in data 10/07): <https://www.bloomberg.com/europe>.

Sito web di Github (consultato in data 14/04): <https://github.com/>.

Sito web di Rproject.(consultato in data 10/04): <https://www.r-project.org/>.

Sito web di Machine Learning Mastery (consultato in data 07/07):
<https://machinelearningmastery.com/>.