

## CAPITOLO 4

# Un'applicazione ai dati reali

In questa applicazione sono state considerate le tre società con il peso più elevato nell'indice S&P 500: *Apple* (AAP), *Amazon* (AMZN) e *NVIDIA* (NVDA). Le serie storiche, scaricate da *Yahoo Finance*, coprono il periodo dal 1° gennaio 2021 al 31 dicembre 2024.

Nella prima parte del caso studio, i modelli GARCH(1,1) ed EGARCH(1,1) verranno utilizzati per stimare la volatilità basandosi sui rendimenti *open-to-close*, ossia la variazione di prezzo tra l'apertura e la chiusura del mercato. In parallelo, i modelli RANGE-GARCH(1,1) e HAR saranno impiegati per stimare la volatilità utilizzando misure basate sul *range*, che sfruttano informazioni più ampie sulla dinamica dei prezzi *intraday*, come la differenza tra il massimo e il minimo.

L'obiettivo principale dell'analisi è valutare se l'introduzione di stimatori di volatilità basati sul *range* possa migliorare l'accuratezza previsionale rispetto ai modelli tradizionali basati esclusivamente sui rendimenti (*open-to-close*, *close-to-close*). Questo confronto consentirà di comprendere se l'uso di informazioni aggiuntive, come l'escursione giornaliera dei prezzi, permetta di catturare meglio la dinamica della volatilità e fornire previsioni più affidabili.

Ogni modello sarà sottoposto alle consuete diagnostiche sui residui per verificarne l'adeguatezza e valutare se soddisfa le ipotesi necessarie per una corretta modellizzazione. Successivamente, analizzeremo la capacità previsionale dei modelli. In particolare, valuteremo le loro prestazioni considerando campioni *out-of-sample* di diversa dimensione, ovvero 50, 100 e 200 giorni di negoziazione. Questo approccio consente di testare l'efficacia dei modelli su diversi orizzonti temporali: un campione *out – of – sample* più breve permette di valutare la precisione delle previsioni nel breve periodo, mentre uno più lungo evidenzia la capacità del modello di mantenere buone prestazioni nel tempo. Tuttavia, se la volatilità subisce cambiamenti strutturali, i modelli potrebbero adattarsi con velocità diverse a seconda dell'orizzonte di previsione considerato.

Per valutare le *performance*, verrà presa in considerazione la volatilità stimata attraverso lo stimatore di Parkinson, che fungerà da volatilità “vera”. La qualità delle previsioni sarà misurata tramite il *Root Mean Squared Error* (RMSE), che quantifica la deviazione media delle previsioni rispetto alla volatilità osservata.

Infine, per un confronto statistico tra le performance dei modelli, applicheremo il test di *Diebold – Mariano*, che consentirà di verificare se le previsioni di un modello risultano significativamente meno accurate rispetto a quelle di un altro. L'ipotesi nulla del test è che

le due previsioni abbiano la stessa accuratezza, mentre l'ipotesi alternativa è che il primo modello produca previsioni meno accurate rispetto al secondo. Questo approccio consentirà di analizzare in modo rigoroso l'affidabilità e la robustezza dei diversi modelli previsivi in scenari temporali differenti.

Nella seconda parte del caso studio, consideriamo una possibile estensione del modello *HAR* di base per la volatilità basata sul *range*, includendo la possibilità di una relazione non lineare tra la volatilità effettiva del *range* e i suoi valori ritardati nello stesso periodo  $V_{t-1}$ , nonché sui periodi a breve e lungo termine  $V_t^{(5)}$  e  $V_t^{(22)}$ .

Iniziamo con i modelli GARCH(1,1) ed EGARCH(1,1). Nella tabella 4.1 sono riportate le stime dei coefficienti accompagnate dai rispettivi p-value. Nel modello EGARCH(1,1) il coefficiente  $\gamma_1$ , che assume valori negativi, rappresenta l'effetto asimmetrico delle innovazioni sulla volatilità, in particolare esso permette di catturare l'effetto leverage.

Tabella 4.1: **Confronto tra modelli GARCH(1,1) ed EGARCH(1,1) per i ticker selezionati**

Ticker	GARCH(1,1)	EGARCH(1,1)
Apple (AAP)	$\omega = 0.00$	$\omega = -0.16$
	p-value = 0.40	p-value = 0.00
	$\alpha_1 = 0.04$	$\alpha_1 = 0.10$
	p-value = 0.00	p-value = 0.00
Microsoft (MSFT)	$\beta_1 = 0.94$	$\beta_1 = 0.98$
	p-value = 0.00	p-value = 0.00
		$\gamma_1 = -0.06$
		p-value = 0.01
NVIDIA (NVDA)	$\omega = 0.00$	$\omega = -0.07$
	p-value = 0.30	p-value = 0.00
	$\alpha_1 = 0.02$	$\alpha_1 = 0.05$
	p-value = 0.00	p-value = 0.00
	$\beta_1 = 0.96$	$\beta_1 = 0.99$
	p-value = 0.00	p-value = 0.00
		$\gamma_1 = -0.04$
		p-value = 0.00
	$\omega = 0.00$	$\omega = -0.27$
	p-value = 0.54	p-value = 0.00
	$\alpha_1 = 0.04$	$\alpha_1 = 0.12$
	p-value = 0.01	p-value = 0.00
	$\beta_1 = 0.94$	$\beta_1 = 0.96$
	p-value = 0.00	p-value = 0.00
		$\gamma_1 = -0.04$
		p-value = 0.00

I residui di un modello sono fondamentali per valutare se le informazioni presenti nei dati sono state catturate in modo adeguato. Se i residui presentano autocorrelazione, significa che nel modello è rimasta informazione inutilizzata, suggerendo la possibilità di migliorare le previsioni. Per questo motivo, i residui devono risultare incorrelati.

Una prima verifica dell'assenza di autocorrelazione può essere effettuata esaminando il grafico dell'*Autocorrelation Function* (ACF). Tuttavia, oltre all'analisi visiva dell'ACF, è

possibile adottare un approccio più formale attraverso un test statistico che consideri l'intero insieme di valori  $r_k$ , piuttosto che analizzarli singolarmente. Ricordiamo che  $r_k$  rappresenta l'autocorrelazione al *lag*  $k$ . Quando valutiamo il grafico dell'ACF per verificare se ogni picco rientra nei limiti di confidenza, stiamo implicitamente conducendo test multipli di ipotesi, ognuno con una certa probabilità di generare un falso positivo. Se il numero di test è elevato, è probabile che almeno uno di essi restituisca un falso positivo, portandoci a concludere erroneamente che i residui siano autocorrelati anche quando non lo sono.

Per evitare questo problema, è opportuno verificare se le prime  $h$  autocorrelazioni differiscono in modo significativo da quelle attese in un *white noise*. A tal fine, si utilizza un *test portmanteau*, che analizza un insieme di autocorrelazioni anziché singoli valori. Un esempio di *test portmanteau* è il *Ljung-Box test*.

Dall'analisi della Tabella 4.2 e delle Figure (4.0.1-4.0.2), possiamo concludere che i residui si comportano come un *white noise*, confermando l'adeguatezza della specificazione del modello.

Tabella 4.2: **Box-Ljung test tra modelli GARCH(1,1) ed EGARCH(1,1) per i ticker selezionati**

Ticker	GARCH(1,1)	EGARCH(1,1)
Apple (AAP)	$x - squared = 3.5$	$x - squared = 4.4$
	$df = 10$	$df = 10$
	$p - value = 0.96$	$p - value = 0.92$
Microsoft (MSFT)	$x - squared = 10.4$	$x - squared = 11.3$
	$df = 10$	$df = 10$
	$p - value = 0.40$	$p - value = 0.33$
NVIDIA (NVDA)	$x - squared = 11.4$	$x - squared = 12.0$
	$df = 10$	$df = 10$
	$p - value = 0.32$	$p - value = 0.28$

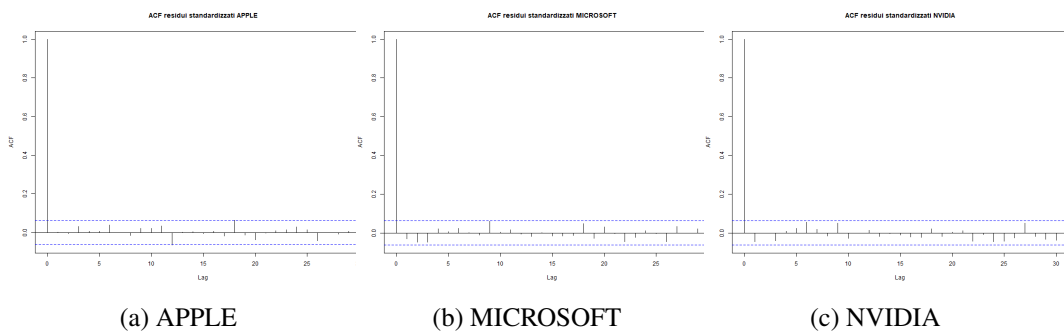


Figura 4.0.1: ACF dei residui standardizzati per il modello GARCH(1,1)

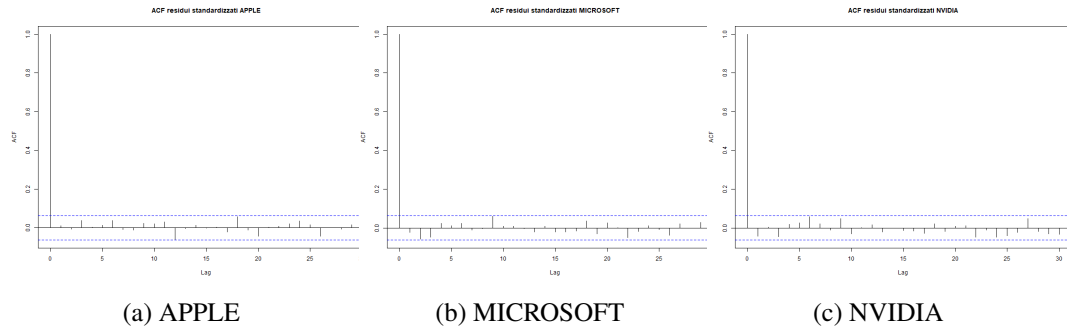


Figura 4.0.2: ACF dei residui standardizzati per il modello EGARCH(1,1)

La Tabella 4.3 riporta i risultati dell'*ARCH LM-test* per i modelli GARCH(1,1) ed EGARCH(1,1). Questo test verifica la presenza di effetti ARCH nei residui, valutando se la varianza condizionata sia stata modellata correttamente.

Sotto l'ipotesi nulla  $H_0$ , tutti i coefficienti da  $\alpha_1$  a  $\alpha_r$  sono uguali a zero, il che implica che i residui siano condizionatamente omoschedastici e che il modello sia ben specificato. Al contrario, sotto l'ipotesi alternativa  $H_1$ , almeno uno di questi coefficienti è diverso da zero, indicando la presenza di eteroschedasticità condizionata nei residui.

Se i residui risultano condizionatamente eteroschedastici, significa che l'informazione passata continua a influenzare la loro varianza, suggerendo una specificazione non adeguata del modello sulla varianza condizionata.

Dall'analisi dei risultati riportati nella Tabella 4.3, possiamo concludere che i residui non presentano effetti ARCH, confermando che la varianza condizionata è stata modellata in modo appropriato.

Tabella 4.3: ARCH LM-test tra modelli GARCH(1,1) ed EGARCH(1,1) per i ticker selezionati

Ticker	GARCH(1,1)	EGARCH(1,1)
Apple (AAP)	$Chi - squared = 6.49$	$Chi - squared = 7.05$
	$df = 12$	$df = 12$
	$p - value = 0.88$	$p - value = 0.85$
Microsoft (MSFT)	$Chi - squared = 6.7$	$Chi - squared = 7.0$
	$df = 12$	$df = 12$
	$p - value = 0.87$	$p - value = 0.85$
NVIDIA (NVDA)	$Chi - squared = 4.91$	$Chi - squared = 2.90$
	$df = 12$	$df = 12$
	$p - value = 0.96$	$p - value = 0.99$

Ora consideriamo il modello RANGE-GARCH(1,1) e il modello HAR, che incorpora informazioni sulla volatilità a diversi orizzonti temporali (giornaliero, settimanale e mensile). Questo modello è particolarmente efficace nel descrivere la persistenza della volatilità, sfruttando la dipendenza temporale delle sue componenti.

Nella tabella 4.4 sono riportate le stime dei coefficienti per entrambi i modelli, accompagnate dai rispettivi p-value, che indicano la significatività statistica di ciascun parametro.

Anche in questo caso, prima di procedere con la fase di previsione, è essenziale esaminare i residui generati dai due modelli per verificare se riescono a rappresentare adeguatamente la dinamica della volatilità. Questa analisi permette di individuare eventuali segnali di misspecificazione e di valutare se i modelli necessitano di miglioramenti.

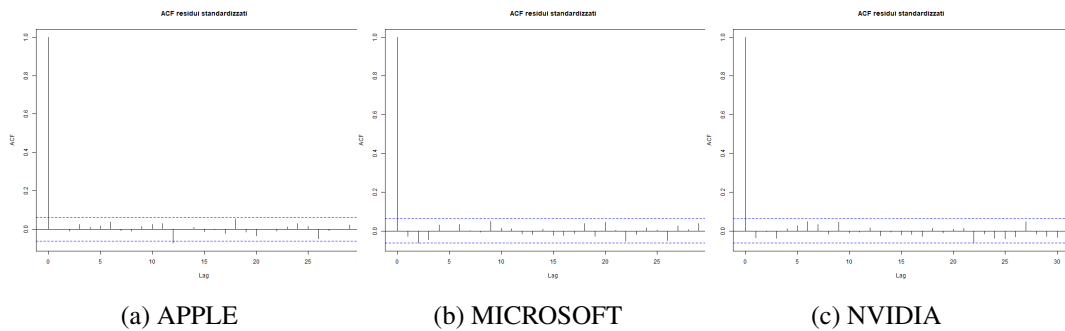
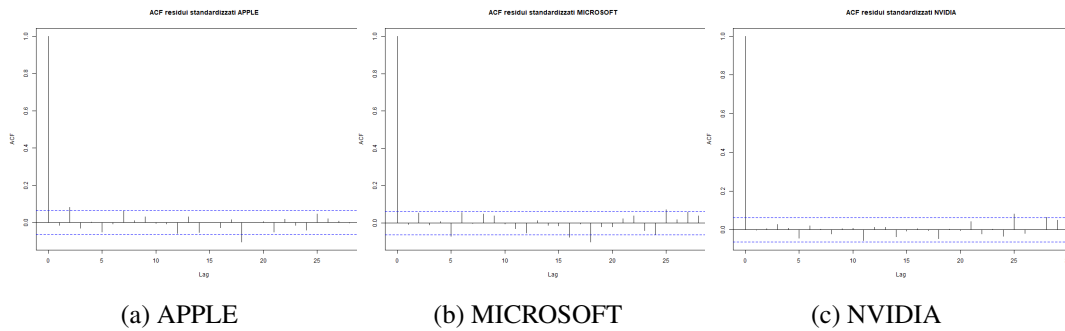
Tabella 4.4: **Confronto tra modelli RANGE-GARCH(1,1) ed HAR per i ticker selezionati**

Ticker	RANGE-GARCH(1,1)	HAR
Apple (AAP)	$\omega = 0.00$ $p - value = 0.02$	$\beta_0 = 0.00$ $p - value = 0.00$
	$\alpha_1 = 0.24$ $p - value = 0.00$	$\beta_1 = 0.19$ $p - value = 0.00$
	$\beta_1 = 0.69$ $p - value = 0.00$	$\beta_2 = 0.25$ $p - value = 0.00$
		$\beta_3 = 0.30$ $p - value = 0.00$
Microsoft (MSFT)	$\omega = 0.00$ $p - value = 0.09$	$\beta_0 = 0.00$ $p - value = 0.00$
	$\alpha_1 = 0.24$ $p - value = 0.01$	$\beta_1 = 0.024$ $p - value = 0.50$
	$\beta_1 = 0.71$ $p - value = 0.00$	$\beta_2 = 0.46$ $p - value = 0.00$
		$\beta_3 = 0.26$ $p - value = 0.00$
NVIDIA (NVDA)	$\omega = 0.00$ $p - value = 0.00$	$\beta_0 = 0.00$ $p - value = 0.00$
	$\alpha_1 = 0.23$ $p - value = 0.00$	$\beta_1 = 0.09$ $p - value = 0.00$
	$\beta_1 = 0.72$ $p - value = 0.00$	$\beta_2 = 0.37$ $p - value = 0.00$
		$\beta_3 = 0.28$ $p - value = 0.00$

Osservando i risultati riportati nella tabella 4.5 e nelle figure (4.0.3-4.0.4), possiamo constatare che i residui non presentano autocorrelazione significativa e si comportano come un *white noise*. Questo suggerisce che i modelli hanno catturato in modo soddisfacente le caratteristiche della volatilità e che le ipotesi di base risultano rispettate.

Tabella 4.5: **Box-Ljung test tra modelli RANGE-GARCH(1,1) ed HAR per i ticker selezionati**

Ticker	RANGE-GARCH(1,1)	HAR
Apple (AAP)	$x - squared = 3.46$	$x - squared = 14.95$
	$df = 10$	$df = 10$
	$p - value = 0.96$	$p - value = 0.13$
Microsoft (MSFT)	$x - squared = 11.43$	$x - squared = 15.27$
	$df = 10$	$df = 10$
	$p - value = 0.32$	$p - value = 0.12$
NVIDIA (NVDA)	$x - squared = 9.23$	$x - squared = 3.72$
	$df = 10$	$df = 10$
	$p - value = 0.50$	$p - value = 0.95$

Figura 4.0.3: **ACF dei residui standardizzati per il modello RANGE-GARCH(1,1)**Figura 4.0.4: **ACF dei residui standardizzati per il modello HAR**

Anche in questo caso, utilizziamo il *test ARCH-LM* per verificare la presenza di eteroschedasticità nei residui. Analizzando i risultati riportati nella tabella 4.6, possiamo concludere che i residui non presentano effetti ARCH, confermando così l'assenza di eteroschedasticità condizionata e la corretta specificazione dei modelli.

Tabella 4.6: **ARCH LM-test tra modelli RANGE-GARCH(1,1) e HAR per i ticker selezionati**

Ticker	RANGE-GARCH(1,1)	HAR
Apple (AAP)	$\chi^2 = 8.85$	$\chi^2 = 4.57$
	$df = 12$	$df = 12$
	$p\text{-value} = 0.71$	$p\text{-value} = 0.97$
Microsoft (MSFT)	$\chi^2 = 12.37$	$\chi^2 = 2.08$
	$df = 12$	$df = 12$
	$p\text{-value} = 0.41$	$p\text{-value} = 0.99$
NVIDIA (NVDA)	$\chi^2 = 9.99$	$\chi^2 = 8.93$
	$df = 12$	$df = 12$
	$p\text{-value} = 0.61$	$p\text{-value} = 0.70$

Ora testiamo le capacità di previsione dei modelli implementati. Come si osserva dalle tabelle 4.7 e 4.8, in ogni scenario di previsione i modelli RANGE-GARCH(1,1) e HAR mostrano una migliore capacità previsionale rispetto agli altri modelli considerati. Questo risultato è evidente dal valore più basso dell'RMSE (*Root Mean Squared Error*), indicatore dell'accuratezza delle previsioni.

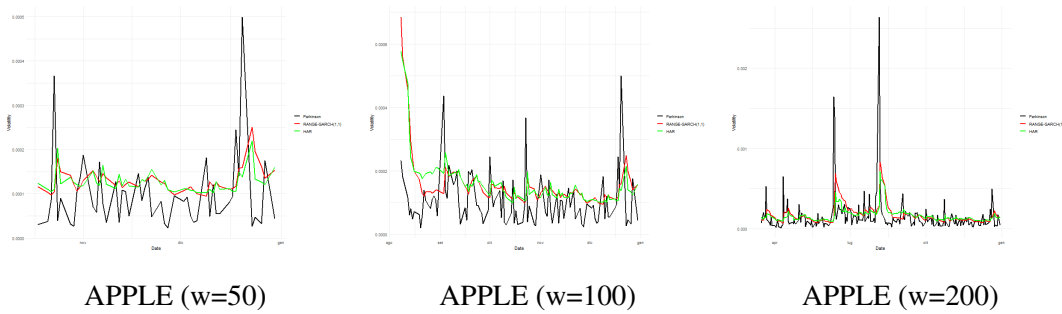
In particolare, il simbolo \*\*\* segnala che la differenza tra i modelli è statisticamente significativa a favore dei modelli RANGE-GARCH(1,1) e HAR, confermando la loro superiorità nella previsione della volatilità. Questi risultati evidenziano quindi la maggiore efficacia di tali modelli nel catturare la dinamica della volatilità rispetto alle alternative analizzate.

Tabella 4.7: **RMSE per GARCH(1,1) e RANGE-GARCH(1,1). \*\*\* indica quando la differenza è significativa. (p-value < 0.001).  $w$  indica l'orizzonte temporale.**

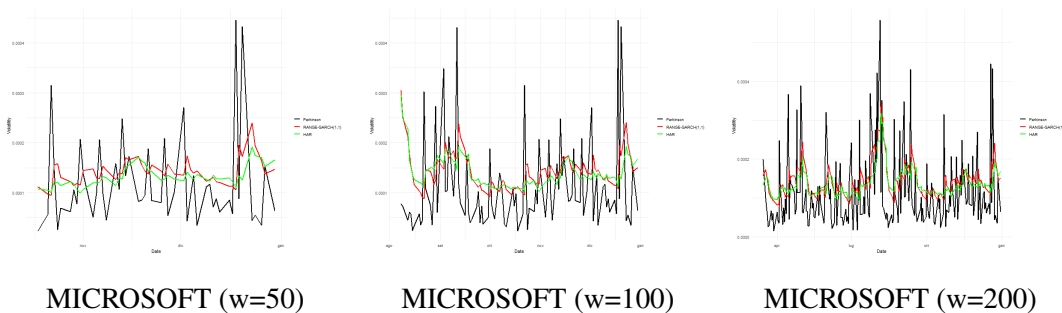
Ticker	GARCH(1,1)			RANGE-GARCH(1,1)		
	$w = 50$	$w = 100$	$w = 200$	$w = 50$	$w = 100$	$w = 200$
AAPL	0.30	0.29	0.36	0.26***	0.21***	0.18***
MSFT	0.37	0.34	0.37	0.32***	0.28***	0.22***
NVDA	0.36	0.34	0.34	0.31	0.23***	0.16***

Tabella 4.8: **RMSE per EGARCH(1,1) e HAR. \*\*\* indica quando la differenza è significativa. (p-value < 0.001).  $w$  indica l'orizzonte temporale.**

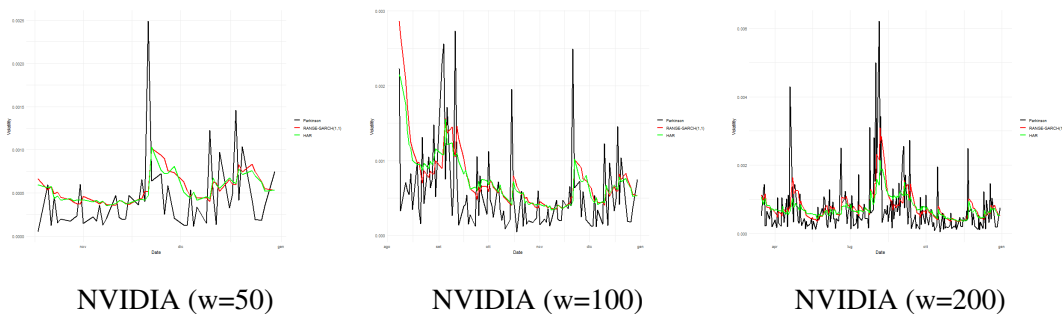
Ticker	EGARCH(1,1)			HAR		
	$w = 50$	$w = 100$	$w = 200$	$w = 50$	$w = 100$	$w = 200$
AAPL	0.40	0.32	0.45	0.29***	0.22***	0.16***
MSFT	0.44	0.50	0.43	0.36***	0.27***	0.21***
NVDA	0.30	0.25	0.26	0.27	0.23***	0.19***



**Figura 4.0.5: Confronto delle stime di volatilità per APPLE con diversi orizzonti temporali (w=50, 100, 200).**



**Figura 4.0.6: Confronto delle stime di volatilità per MICROSOFT con diversi orizzonti temporali (w=50, 100, 200).**



**Figura 4.0.7: Confronto delle stime di volatilità per NVIDIA con diversi orizzonti temporali (w=50, 100, 200).**

Ora, in questa seconda parte del caso studio, come già accennato in precedenza includiamo la possibilità di una relazione non lineare tra la volatilità effettiva del *range* e i suoi valori ritardati. Come primo passo dell'analisi, è opportuno verificare la presenza di caratteristiche non lineari e di eteroschedasticità condizionata, ovvero i cosiddetti effetti ARCH, nelle serie storiche di volatilità per ogni stock considerato, costruite considerando lo stimatore di Garman-Klass. A tal fine, sono stati utilizzati il test di Teraesvirta e il test di White per verificare l'ipotesi di linearità, mentre per la rilevazione degli effetti ARCH è stato impiegato il test LM.



I risultati riportati nella tabella 4.9 mostrano un chiaro rigetto dell'ipotesi di linearità e la presenza di effetti ARCH in tutte e tre le serie analizzate.

**Tabella 4.9: Statistica test e p-value per i test di linearità di Teraesvirta e White e per il test LM sugli effetti ARCH, applicati alla volatilità basata sul range delle tre azioni selezionate.**

Stock	Teraesvirta Test		White Test		LM Test	
	Statistic	p-value	Statistic	p-value	Statistic	p-value
AAPL	69	0.00	70	0.00	30	0.00
MSFT	31	0.00	27	0.00	27	0.00
NVDA	33	0.00	34	0.00	55	0.00

Consideriamo una possibile estensione del modello *HAR* di base per la volatilità basata sul *range*, ovvero il modello NN-HAR.

La rete neurale implementata utilizza connessioni *skip* e generalizza il modello HAR classico, che è contenuto nel modello NN-HAR. Pertanto, il nuovo modello aggiunge componenti non lineari alla relazione lineare di base. Una volta fissato il numero di neuroni appartenenti all'unico strato nascosto, in questo caso pari a 2, la rete neurale con input  $V_{t-1}, V_t^{(5)}$  e  $V_t^{(22)}$ , è stata stimata utilizzando l'algoritmo di apprendimento BFGS. L'addestramento della rete è stato ripetuto 50 volte a partire da stati iniziali casuali per evitare che il modello rimanesse intrappolato in minimi locali. Le ultime 50 osservazioni sono state utilizzate come test set.

Come *benchmark*, è stato utilizzato il modello HAR di base, anch'esso con input  $V_{t-1}, V_t^{(5)}$  e  $V_t^{(22)}$ . Il modello NN-HAR(02) e il *benchmark* sono stati confrontati per le previsioni *one step ahead* e per le distribuzioni delle previsioni nel *test set*. La procedura consiste nell'aggiungere progressivamente un valore dal *test set* al *training set*, uno alla volta, e utilizzare il modello per prevedere il valore successivo. Questo processo è stato ripetuto fino a considerare l'intero *test set*, ottenendo i valori della previsione *one step ahead*.

Le distribuzioni delle previsioni sono state ottenute utilizzando la procedura di *bootstrap* (algoritmo 6 capitolo 3), con  $B = 1699$ . Il numero dispari è stato scelto per evitare interpolazioni quando si calcolano i percentili della distribuzione di previsione *bootstrap*. Tutti gli intervalli di previsione sono stati calcolati con un livello di confidenza nominale del 95%. L'accuratezza della previsione puntuale è stata valutata utilizzando il *Root Mean Square Error* (RMSE) e il *Mean Percentage Absolute Error* (MAPE). L'accuratezza delle distribuzioni di previsione è stata valutata tramite il *Winkler Score* (WS), il *Quantile Score* (QS) e il *Continuous Ranked Probability Score* (CRPS).

La tabella 4.10 riporta le cinque misure di accuratezza delle previsioni. È evidente che, nella maggior parte dei casi, il modello NN-HAR(02) ha prestazioni migliori, sia in termini di previsioni puntuali che di distribuzioni delle previsioni. Un'eccezione è rappresentata dal caso dell'azione NVIDIA, in cui tutte e cinque le metriche sono molto simili tra loro.

Tabella 4.10: **Confronto tra i modelli HAR e NN-HAR(02).** Per ogni misura, il valore minimo è riportato in grassetto.

Ticker	HAR					NN-HAR(02)				
	RMSE	MAPE	WS	QS	CRPS	RMSE	MAPE	WS	QS	CRPS
AAPL	0.29	130.48	6.30	0.07	0.10	<b>0.25</b>	<b>126.94</b>	<b>4.3</b>	<b>0.03</b>	<b>0.09</b>
MSFT	0.36	93.51	13.92	0.15	0.16	<b>0.33</b>	<b>87.52</b>	<b>4.33</b>	<b>0.02</b>	<b>0.13</b>
NVDA	0.27	129.54	6.95	0.05	0.09	0.27	127.70	6.90	0.06	0.09

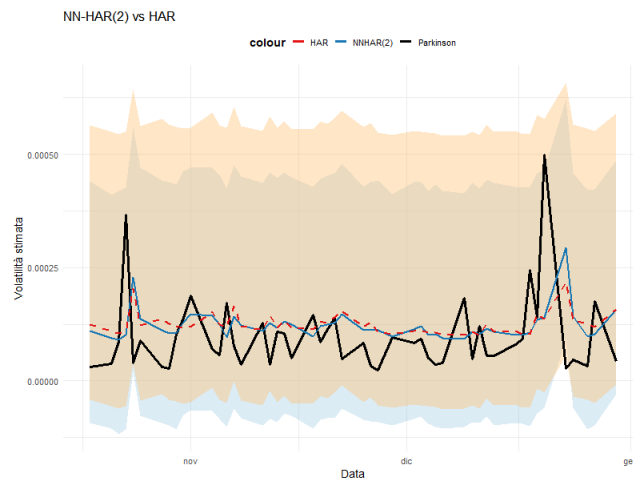


Figura 4.0.8: **APPLE**

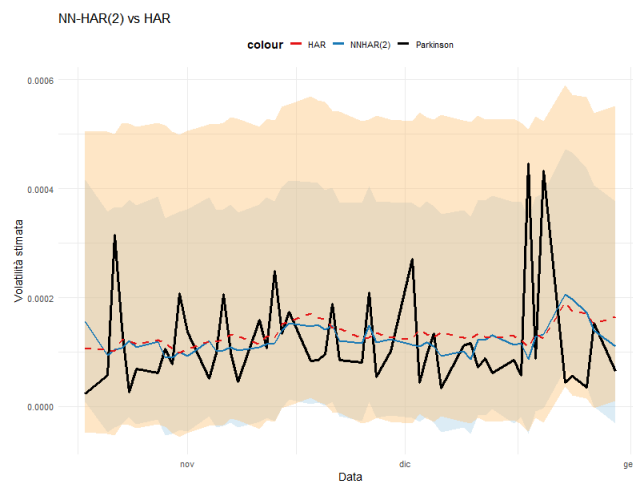


Figura 4.0.9: **MICROSOFT**

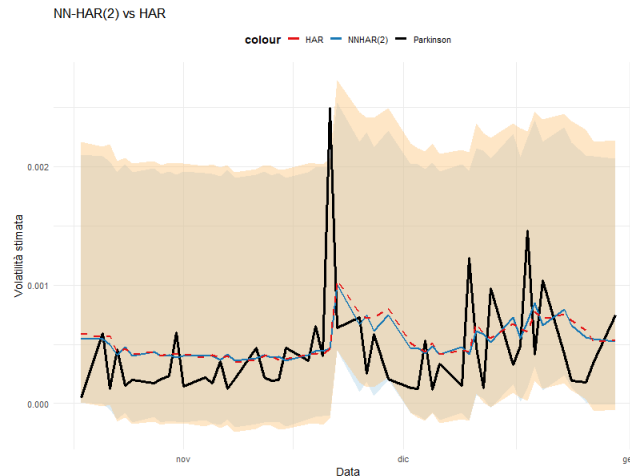


Figura 4.0.10: NVIDIA

Per valutare se il modello NN-HAR considerato riesca a catturare le caratteristiche non lineari e l'alta persistenza osservata nella volatilità del *range*, sono stati eseguiti per ogni azione, il test di Teraesvirta, il test di White e il test LM sui residui del modello.

Dalla tabella 4.11 è chiaro che i residui dei modelli NN-HAR(02) per APPLE e MICRO-SOFT, non evidenziano non linearità trascurate e non presentano effetti ARCH.

Tabella 4.11: **Statistica test e p-value per i test di linearità di Teraesvirta e White e per il test LM sugli effetti ARCH, calcolati sui residui del modello NN-HAR(02).**

Stock	Teraesvirta Test		White Test		LM Test	
	Statistic	p-value	Statistic	p-value	Statistic	p-value
AAPL	2.50	0.27	3.70	0.15	7	0.85
MSFT	0.11	0.94	0.17	0.91	0.50	1
NVDA	7.7	0.02	3.50	0.16	4.90	0.95

Per migliorare le prestazioni del modello NN-HAR(02) per NVIDIA, una delle prime strategie potrebbe essere quella di aumentare la dimensione dello strato nascosto. Passando da 2 neuroni a 4 o 6, il modello potrebbe acquisire una maggiore capacità di catturare le relazioni non lineari presenti nei dati. Un'altra possibilità interessante sarebbe l'inclusione di variabili informative aggiuntive. L'arricchimento del modello con nuove caratteristiche potrebbe consentirgli di comprendere in modo più approfondito i fattori che influenzano la volatilità.

Consideriamo ora un numero di neuroni appartenenti allo strato nascosto pari a 6 e analizziamo le metriche di *performance* ottenute, al fine di valutare eventuali miglioramenti nella capacità predittiva del modello.

Dalla Tabella 4.12 si osservano metriche di *performance* migliori, sia in termini di previsione puntuale che di distribuzione previsionale. Inoltre, come riportato nella Tabella 4.13, i residui del modello NN-HAR(06) per NVIDIA non mostrano non linearità trascurate né evidenziano la presenza di effetti ARCH.

Tabella 4.12: **Metriche di performance per il modello NN-HAR(06)**

Ticker	NN-HAR(06)				
	RMSE	MAPE	WS	QS	CRPS
NVDA	0.23	124.79	5.40	0.03	0.08

Tabella 4.13: **Statistica test e p-value per i test di linearità di Teraesvirta e White e per il test LM sugli effetti ARCH, calcolati sui residui del modello NN-HAR(06).**

Stock	Teraesvirta Test		White Test		LM Test	
	Statistic	p-value	Statistic	p-value	Statistic	p-value
NVDA	4.30	0.11	1.09	0.57	1.70	0.99