

UNIVERSITÀ DEGLI STUDI DI PADOVA

FACOLTÀ DI SCIENZE STATISTICHE

TESI DI LAUREA SPECIALISTICA IN STATISTICA E INFORMATICA

**STUDIO DELLA STABILITÀ DELLE STIME
DEL MODELLO GARCH PER SERIE
FINANZIARIE**

STABILITY EVALUATION OF GARCH FITTING FOR
FINANCIAL TIME SERIES

RELATORE: Chiar.mo Prof. ADELCHI AZZALINI

CANDIDATA: ROBERTA CURTO
603651 - STF

ANNO ACCADEMICO 2010 - 2011

Indice

Ringraziamenti.....	1
Introduzione.....	3
Capitolo 1.	
I modelli GARCH e le distribuzioni utilizzate come componente d'errore...7	
1.1 Regolarità empiriche dei rendimenti.....	8
1.2 Il modello GARCH.....	10
1.3 Le distribuzioni analizzate come componente d'errore.....	12
1.3.1 La distribuzione Skew-Normal (SN).....	15
1.3.2 La distribuzione Skew-t (ST).....	18
1.3.3 La distribuzione di Subbotin.....	19
1.4 I titoli azionari: oggetto di studio.....	20
1.5 Analisi preliminare.....	23
Capitolo 2.	
Le funzioni per il calcolo del modello GARCH.....	31
2.1 Ambienti statistici a confronto e rispettive librerie che implementano modelli GARCH.....	32
2.1.1 La libreria fGarch e le sue funzioni.....	33
2.1.2 La libreria rgarch e le sue funzioni.....	34
2.1.3 Il modulo S+FinMetrics.....	34
2.2 Calcolo di funzione di log-verosimiglianza per modelli GARCH con distribuzioni asimmetriche per componente d'errore.....	35
2.2.1 Particolarità nel calcolo del stima della log-verosimiglianza....	39
2.3 Utilizzo di optim per il calcolo della stima di massima verosimiglianza.....	40

2.4 Confronto tra le varie routine per la stima del vettore θ	42
Capitolo 3.	
Analisi della stabilità delle stime.....	47
3.1 La ricerca della stabilità delle stime dei parametri	48
3.2 Creazioni routine che simulano serie dei modelli GARCH.....	54
3.2.1 Ripetizione dei test sulle serie simulate.....	56
3.3 Confronto con le routine preesistenti.....	58
3.4 Analisi su valori di n_0 prossimi a n	59
Conclusioni.....	63
Appendice A.....	67
A.1 Codice che calcola la log-verosimiglianza con l'opzione V1 per i vincoli.....	67
A.2 Codice che calcola la log-verosimiglianza con l'opzione V2 per i vincoli.....	69
A.3 Codice che calcola la log-verosimiglianza con l'opzione V3 per i vincoli.....	72
A.4 Codice che genera serie con l'opzione V3 per i vincoli.....	74
Appendice B.....	77
Bibliografia.....	99

Ringraziamenti

Non sono mai stata brava a causa del mio carattere nel tradurre in parole l'affetto che provo per le persone che mi circondano, ma mi rendo conto che qualche volta è importante e doveroso farlo e questa mi pare un'ottima occasione vista l'enorme gioia che accompagna il taglio di questo importante traguardo come la laurea. Rimane comunque difficile scrivere dei ringraziamenti perché molte persone hanno contribuito nel bene e nel male a farmi crescere dal punto di vista intellettuale e umano e a farmi essere la persona che oggi sono ed è impossibile elencarli tutti in una paginetta o poco più.

Il lavoro condotto per questa tesi non è completo per il fatto che ci sono molte cose che avrei potuto affinare e sviluppare e per questo mi scuso con il mio relatore prof. Azzalini, ma allo stesso tempo lo ringrazio per avermi dedicato il suo tempo per raggiungere questo mio traguardo. La mia stima nei suoi confronti è dovuta oltre alla sua profonda conoscenza ed esperienza nel campo della Statistica anche alla sua simpatia nell'affrontare i problemi. Spero comunque di essere stata all'altezza di quanto mi è stato chiesto di fare, di certo l'impegno e la ricerca costante al miglioramento da parte mia non sono mancati. Mi sento di dover ringraziare anche il prof. Melucci per avermi convinto a fare il passaggio da Ingegneria a Statistica per la laurea specialistica. Sono molto soddisfatta della scelta e sarei pronta a rifarla.

Passando ora alle persone che mi hanno permesso di essere qui ora... Un enorme e primo ringraziamento va alla Mamma e Papà per l'affetto, il sostegno, la fiducia e ... la pazienza che hanno dovuto avere in questi tre anni. A Leonardo, il mio fratellino, che riesce sempre a strapparmi un sorriso anche nei momenti più difficili con la sua ingenuità. Alla nonna Lena e a tutte le zie e zii, cugini e cugine e parenti stretti che mi sono stati vicini soprattutto nell'ultimo periodo a causa della "Gina".

Un ringraziamento a tutti i compagni ingegneri e statistici incontrati in questi

Ringraziamenti

sei anni di università tra cui Enrico, Fede, Panfi e Julia e a tutti gli amici conosciuti attraverso la pallavolo che amo, pratico e ora anche insegno. Troppi da elencare, ma a tutti voglio dire grazie. Un ringraziamento particolare va alle mie “Fratte senza fastidi” e a tutta la dirigenza del Fratte perché l'ultimo anno, quello di tesi, l'ho trascorso con loro al mio fianco e grazie alla loro simpatia e amicizia mi hanno regalato un anno dal punto di vista emotivo fantastico.

Un ringraziamento doveroso va alla mia “mamma adottiva” Grazia che insieme alla mia famiglia ha cercato sempre di rendere ogni mio giorno migliore e di farmi crescere come una persona con i giusti principi. Alla mia amica Annamaria perché è colei che rende ancora più straordinaria la Tenuta Civrana dove sono nata, grazie alle nostre chiacchierate immerse nella natura.

Ed infine altre due persone meritano un mio enorme grazie. Grazie alla mia allenatrice preferita Federica che oltre ad essere stata importante insegnante nello sport, lo è stata anche nella vita di tutti i giorni dal punto di vista motivazionale. A lei devo molto della persona che sono oggi perché è sempre stata presente con i suoi consigli o meglio ancora inviti a riflettere.

Per ultimo ma non per ordine di importanza, vorrei ringraziare la persona che più mi è stata vicina nell'ultimo periodo di difficoltà perché da quando è entrato nella mia vita ha dato tantissimo alla mia persona sia dal punto di vista affettivo che dal punto di vista pratico. È riuscito a dare una svolta per il raggiungimento di questo traguardo, grazie al suo continuo sostegno ed incitamento...e amore. Grazie Pier.

Introduzione

Negli ultimi anni lo studio dell'andamento dei rendimenti finanziari del portafoglio si è andato diffondendo, in quanto si cerca di fornire una sempre migliore misurazione del rischio a chi si avvicina allo studio dei titoli azionari.

Il nostro lavoro si è concentrato nella valutazione della stabilità delle stime dei parametri del modello GARCH. Abbiamo proceduto alla stima dei parametri considerando man mano la serie dello stesso titolo con un minor numero di osservazioni rispetto alla lunghezza effettiva con l'intento di ricercare le caratteristiche del titolo stesso solamente su un limitato numero di rendimenti consecutivi che compongono la serie.

Per lo studio utilizziamo le serie di rendimenti di titoli finanziari prese direttamente dal mercato europeo, più precisamente quello tedesco, e i modelli GARCH, i più ampiamente diffusi per analisi finanziarie di questo tipo, con diverse distribuzioni per la componente d'errore, quali la distribuzione normale e la sua versione asimmetrica, la t di Student e la sua versione asimmetrica e la distribuzione di Subbotin. Se ne presenta una trattazione teorica nel capitolo 1.

Poiché è noto come la distribuzione marginale dei rendimenti finanziari abbia regolarità empiriche particolari, quali un comportamento asimmetrico e delle code più spesse se confrontate con la distribuzione normale, tale fatto può avere conseguenza sulla valutazione della distribuzione che prendiamo come componente d'errore per il modello GARCH.

In un'analisi preliminare abbiamo calcolato i coefficienti di curtosi, non campionari ma derivanti dalle stime di serie reali di rendimenti ottenute dai parametri di forma, per i modelli che utilizzavano la t di Student e della distribuzione generalizzata degli errori come componente d'errore. Successivamente abbiamo sviluppato delle considerazioni sul parametro che modella l'asimmetria, per le distribuzioni normale e t di Student asimmetriche, in modo da farsi un'idea completa delle caratteristiche dei rendimenti. Per questa analisi, nell'implementazione dei modelli GARCH sono state utilizzate varie

librerie già pronte a riguardo per l'ambiente R e S-PLUS, rispettivamente *fGarch* e *rgarch* per R e *FinMetrics* per S-PLUS.

Man mano che l'analisi procedeva ci siamo accorti della necessità di creare della apposite routine per i modelli GARCH (si veda capitolo 2), che utilizzassero per la componente d'errore la versione della normale asimmetrica e della t di Student del prof. Azzalini, in quanto più facilmente leggibile e di facile computazione per l'inferenza, rispetto a quelle implementate nelle librerie per R che usano la versione di Fernández e Steel. Per le nostre implementazioni consideriamo un modello molto semplice, GARCH(1,1) che considera unicamente ritardi unitari per i parametri che caratterizzano la volatilità: è il più usato per tali analisi.

Una volta ultimate, l'attenzione si è spostata nella ricerca di una finestra, intesa come la possibilità di prendere solamente un determinato numero di osservazioni consecutive, in sostituzione della serie completa, per avere indicazioni su come il titolo azionario si comporta. Le analisi che verranno effettuate prevedono sia una variazione nella lunghezza sia uno slittamento del punto di partenza di questa finestra. La ricerca di questa finestra ha suggerito molte considerazioni importanti insieme ad altre analisi sui parametri dei vari modelli analizzati.

Successivamente, abbiamo creato delle routine per la generazione di serie pseudocasuali in modo da mettere in confronto le serie reali dei titoli azionari con serie simulate, con partenza dalle stime ottenute con le serie reali. Questo confronto è stato necessario in quanto, sapendo delle caratteristiche particolari che possono avere le serie reali, volevamo vedere se i quesiti che ci siamo posti come punto di partenza nell'analisi della stabilità delle stime, ottenevano le stesse risposte di quelle reali. Inoltre sono state fatte delle valutazioni sulle routine per la generazione di serie: proponendo un confronto di risultati tra quelle simulate con queste routine e quelle simulate con la funzione *garchSim* del pacchetto *fGarch*.

I risultati precedenti ci hanno portato a sviluppare ulteriori analisi per verificare come le stime dei parametri di "parti di serie" si avvicinano alle stime di riferimento del titolo man mano che la lunghezza della finestra si avvicina alla lunghezza della serie completa. Questo riscontro è stato necessario in quanto le

analisi precedenti non ci hanno permesso di ottenere quello che noi auspicavamo come si potrà vedere nel capitolo 3.

Nelle appendici vengono riportati sia il codice R necessario al calcolo delle funzioni di log-verosimiglianza e delle simulazioni, sia dei grafici esplicativi relativi alla ricerca della finestra temporale per i titoli finanziari analizzati.

Capitolo 1.

I modelli GARCH e le distribuzioni utilizzate come componente d'errore

In questo capitolo definiamo alcune peculiarità relativamente ai rendimenti, oggetti del nostro studio. L'idea è di rappresentare questi oggetti sia come realizzazioni di un processo stocastico, sia come serie storiche, vale a dire come sequenze di dati relativi ad osservazioni misurate a diversi istanti temporali e nel seguito, presentare gli strumenti statistico-econometrici che possono essere applicati a tali serie storiche per cercare di recuperare il processo generatore dei dati. L'interesse è legato anche allo studio della distribuzione di probabilità condizionatamente al passato che meglio modella il processo stocastico rendimento.

Nella sezione 1.1 presentiamo i rendimenti e la descrizione della loro distribuzione empirica insieme alle evidenze empiriche che li caratterizzano, emerse negli ultimi anni. Nella sezione 1.2 è possibile trovare una descrizione dello strumento che noi utilizzeremo come generatore dei rendimenti e del suo funzionamento. A riguardo noi utilizzeremo il modello GARCH per studiare i rendimenti finanziari.

A seguire nella sezione 1.3 presentiamo le varie versioni delle distribuzioni di probabilità che sono ipotizzabili come processo generatore dei dati, utilizzate dai modelli GARCH come componente d'errore, con particolare riferimento alla normale e alla t di Student asimmetrica, oltre alla distribuzione di Subbotin.

Poiché quello che è interessante è applicare ai dati reali quello che stiamo analizzando nella sezione 1.4 presentiamo i titoli finanziari delle società quotate in borsa che sono oggetto delle nostre analisi e considerazioni. Concludiamo quindi nella sezione 1.5 con una prima analisi preliminare su questi dati a nostra disposizione, in modo da toccare con mano fin da subito i problemi che si

riscontrano. Si intende far questo perché spesso, quando si fanno considerazioni nel generale e nel teorico, nel pratico è facile trovare casi in cui si può essere smentiti.

1.1 Regolarità empiriche dei rendimenti

Nell'analisi finanziaria vi sono due fondamentali oggetti di analisi: i prezzi e i rendimenti. Il prezzo di riferimento riportato nelle pagine finanziarie è di per sé non informativo se non viene accompagnato dalle caratteristiche del titolo stesso. Quello che noi andiamo a studiare non è l'andamento dei prezzi (quotazioni), ma quello dei rendimenti, principalmente per due motivi:

- i rendimenti rappresentano un sommario completo e adimensionale dell'opportunità d'investimento in quanto misurano la redditività dell'attività finanziaria;
- le serie dei rendimenti sono di più facile trattabilità dal punto di vista statistico.

I prezzi possono essere indicizzati dall'istante temporale al quale sono osservati (P_t) e sono interpretabili come capitalizzazione rispetto al prezzo osservato al tempo precedente (P_{t-1}). È evidente comunque che i prezzi sono strettamente connessi ai rendimenti, infatti i rendimenti sono definiti dai prezzi giornalieri dei titoli come segue:

$$r_t \equiv 100 [\log(P_t) - \log(P_{t-1})] \quad (1.1)$$

A tutti gli effetti, stiamo studiando un processo stocastico in quanto ciascuna variabile casuale r_t è riferita ad un diverso istante temporale.

Le serie storiche finanziarie sono da diversi anni al centro dell'analisi teorica ed empirica: presenteremo ora i fatti stilizzati che risultano caratterizzare nella maggior parte dei casi le serie finanziarie. Per un approfondimento più dettagliato si rimanda a Gallo e Pacini (2002).

I rendimenti risultano essere incorrelati: tale assenza di correlazione temporale lascia comunque spazio a forme di dipendenza tra le osservazioni. Dal punto di

vista applicativo, questo comporta la presenza di volatilità variabile nel tempo. La volatilità è una misura delle fluttuazioni future dei prezzi.

Sebbene non sia possibile individuare una qualche forma di autocorrelazione nella serie dei rendimenti, vi è una qualche regolarità nell'ampiezza delle fluttuazioni: si alternano periodi persistenti di fluttuazioni più basse a periodi di fluttuazioni più elevate. Questo fenomeno viene identificato con il nome di *volatility clustering* e corrisponde in un qualche modo all'alternanza di periodi di volatilità alta e bassa.

In queste condizioni, si è in presenza di una varianza dei rendimenti che cambia nel tempo, condizionatamente all'informazione passata: in termini statistici rientriamo in un'ipotesi di eteroschedasticità condizionata.

Un'altra caratteristica non meno importante è che la distribuzione marginale dei rendimenti è caratterizzata da un eccesso di curtosi e da una lieve asimmetria: le code risultano essere più spesse se le mettiamo a confronto con la distribuzione normale di riferimento (*leptocurtosi*).

La nostra analisi parte dall'insieme informativo a nostra disposizione I_{t-1} , che contiene i valori ritardati di r_t e di altre variabili d'interesse. Dato che l'evoluzione dei prezzi è guidata dai rendimenti, è utile distinguere la media del rendimento, calcolata sulla base dell'informazione a disposizione nel tempo immediatamente precedente, da quello che costituisce un'innovazione rispetto a quello che ci si aspettava. In questa logica, la media è anch'essa indicizzata dal tempo (μ_t) come pure l'innovazione ε_t :

$$r_t = \mu_t + \varepsilon_t \quad (1.2)$$

L'attenzione non è tanto focalizzata sulla media del rendimento in quanto viene fatta l'assunzione che $\mu_t = \mu$ costante, per semplicità e per evidenza empirica; ma sulla componente che rappresenta la volatilità condizionata in quanto essa può essere interpretabile come misura del rischio di un'attività.

Lo scopo quindi è modellare al meglio la varianza dei rendimenti sulla base dell'informazione a disposizione, vale a dire

$$\sigma_t^2 \equiv \text{Var}[\varepsilon_t | I_{t-1}] \quad (1.3)$$

dato che una delle regolarità empiriche che abbiamo appena osservato sulle serie è il fenomeno del *volatility clustering*.

1.2 Il modello GARCH

Un primo importante tentativo per catturare le evidenze empiriche della distribuzione dei rendimenti, citate nella sezione precedente, è il modello presentato da Engle (1982), *l'AutoRegressive Conditional Heteroskedasticity* (ARCH), che prevede un andamento della varianza del processo generatore dei dati di tipo autoregressivo.

Il modello ARCH è un processo stocastico $\{\varepsilon_t\}$ che si presenta così nel caso più semplice (sotto l'ipotesi di distribuzione normale):

$$\varepsilon_t = \sigma_t v_t \quad (1.4)$$

$$v_t \sim i.i.d N(0,1) \quad (1.5)$$

Per definizione ε_t è serialmente non correlato con media zero; la varianza condizionata di ε_t è positiva e varia nel tempo. Al variare di σ_t^2 si otterranno distribuzioni di probabilità di tipo normale più o meno disperse attorno al centro di simmetria.

Una possibile specificazione della varianza condizionata è:

$$\sigma_t^2 = \omega + \sum_{i=1}^q \alpha_i \varepsilon_{t-i}^2, \quad (1.6)$$

cioè come una funzione lineare dei valori passati del processo al quadrato dove $\omega > 0$ e $\alpha_i \geq 0$. Questo modello, noto come ARCH(q) ha il pregio di catturare gli effetti di *volatility clustering*, ma presenta il grosso limite che le procedure di stima dei corrispondenti parametri si possono complicare notevolmente se nelle serie storiche si manifestano effetti autoregressivi sui quadrati dei rendimenti per un numero elevato di ritardi.

La prima estensione del modello ARCH è stata il modello GARCH (*Generalised ARCH*) introdotto da Bollerslev (1986), la cui idea è stata quella di riprodurre la parsimonia del modello ARMA¹ in termini del numero di parametri utilizzati, inserendo nel modello ARCH, i valori ritardati della varianza condizionata.

Il modello GARCH permette di approssimare un ARCH di ordine molto più elevato usando un numero limitato di parametri e, di conseguenza, un numero ridotto di vincoli di non negatività:

$$\sigma_t^2 = \omega + \sum_{i=1}^q \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^p \beta_j \sigma_{t-j}^2 . \quad (1.7)$$

Per garantire la positività della varianza condizionata si pongono i vincoli $\omega > 0$, $\alpha_i \geq 0$ e $\beta_j \geq 0$. Il processo risulta stazionario in senso debole se e solo se

$$\sum_{i=1}^q \alpha_i + \sum_{j=1}^p \beta_j < 1 . \quad (1.8)$$

Nel modello GARCH(p,q), il valore passato della varianza condizionata (σ_{t-j}^2) riassume in sé le informazioni passate, e quindi spetta al valore dell'innovazione osservata (ε_{t-i}^2) nel periodo precedente il compito di alterare la previsione corrente della varianza condizionata.

In altri termini, le informazioni passate sono sintetizzate dai valori ritardati della varianza, mentre le “novità” e la capacità di variazione del tempo delle stime della varianza sono racchiuse nei termini ε_{t-i}^2 .

L'andamento della distribuzione dei rendimenti saranno da noi modellate con la versione più semplice per questa classe di modelli, un GARCH(1,1) con le seguenti caratteristiche:

¹ Il modello ARMA (AutoRegressive Moving Average) è un tipo di modello matematico lineare che fornisce istante per istante un valore di uscita basandosi sui precedenti valori in entrata e in uscita.

$$r_t = \mu + \varepsilon_t$$

$$\varepsilon_t = \sigma_t v_t \quad v_t \sim i.i.d N(0,1) \quad (1.9)$$

$$\sigma_t^2 = \omega + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2$$

Quello che cambieremo per rispondere alle necessità di un miglior adattamento del modello ai dati saranno le ipotesi distributive alla base del modello stesso; in particolar modo considerando una legge differente alla gaussiana per la distribuzione marginale delle innovazioni che dia la possibilità di catturare le evidenze empiriche di asimmetria ed eccesso di curtosi.

1.3 Le distribuzioni analizzate come componente d'errore

Vogliamo qui presentare le distribuzioni che saranno ipotizzate per la distribuzione marginale delle innovazioni.

L'assunzione di normalità per questa distribuzione risulta in diverse situazioni inadeguata a causa delle evidenze empiriche sopra enunciate quali l'asimmetria e l'eccesso di curtosi che non riescono ad essere colte, come si può vedere nella Figura 1.1.

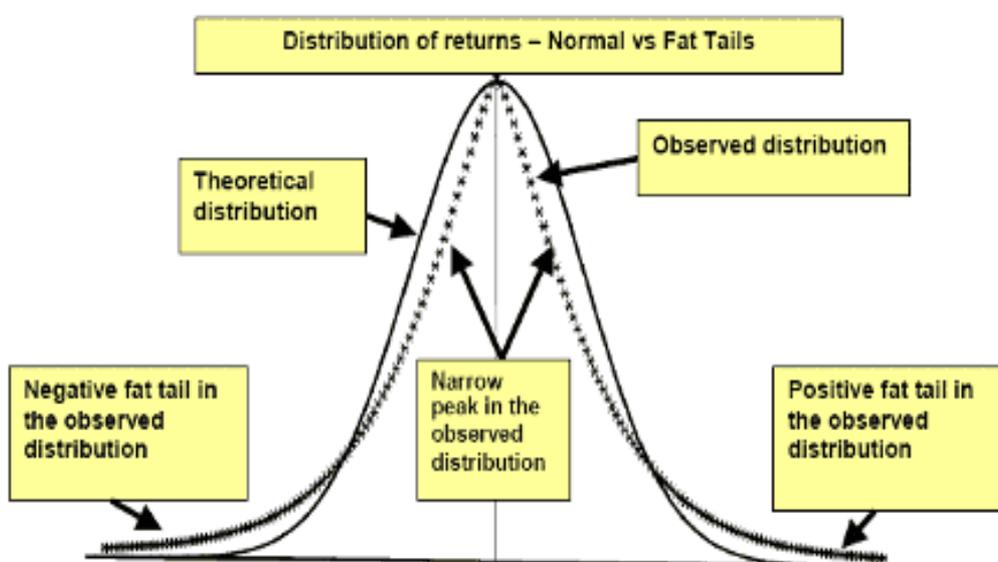


Figura 1.1: La distribuzione dei rendimenti con evidenza alle code pesanti.

1.3 Le distribuzioni analizzate come componente d'errore

Per risolvere il problema dell'eccesso di curtosi abbiamo scelto la distribuzione della t di Student e, secondariamente, la *Generalized Error Distribution* (GED, di cui esiste una formulazione alternativa sotto il nome di distribuzione di Subbotin) in quanto esse presentano delle code più pesanti rispetto alla comune gaussiana come si può vedere in Figura 1.2 e 1.3ⁱⁱ.

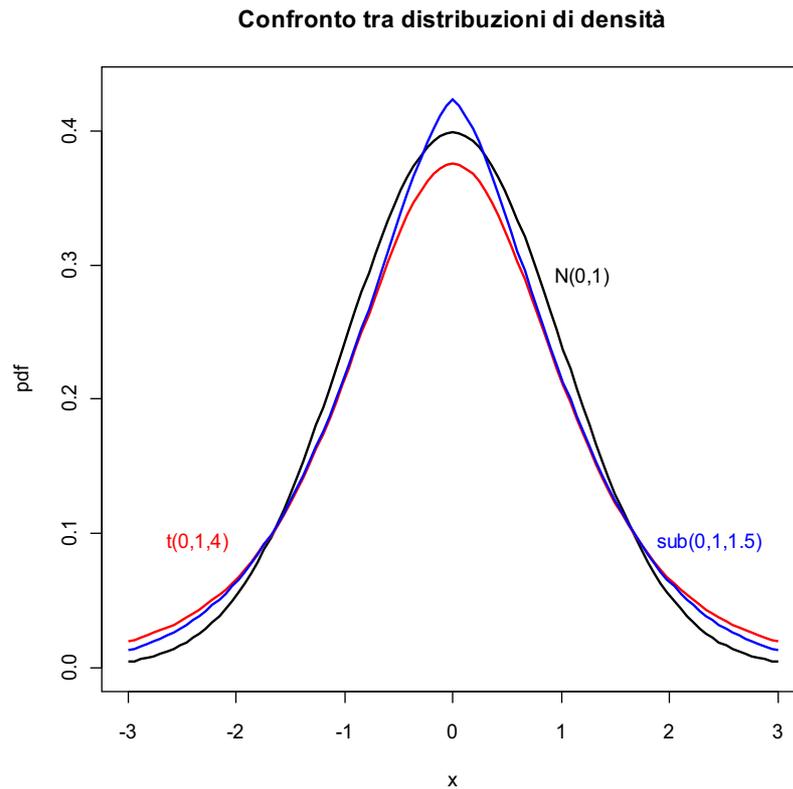


Figura 1.2: Confronto tra la funzione di densità di una normale standardizzata, una t di Student standardizzata con 4 gradi di libertà e una distribuzione di Subbotin con parametro di forma 1.5.

ⁱⁱ I grafici sono stati realizzati con l'ambiente R.

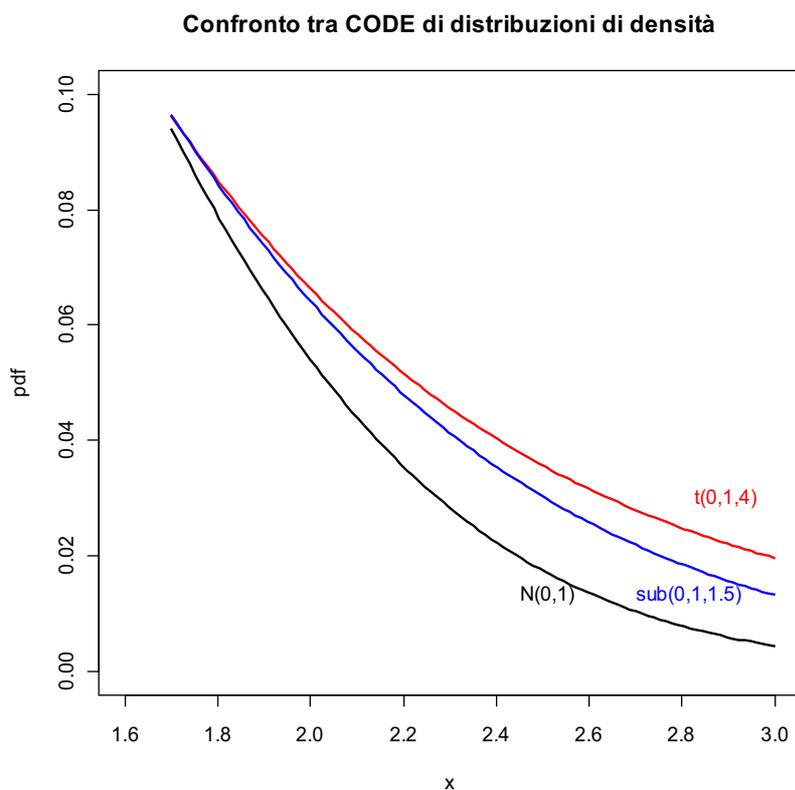


Figura 1.3: Evidenziamo la pesantezza delle code delle funzioni di densità.

Successivamente, per ricercare una miglior adattabilità dei dati al modello abbiamo ipotizzato come distribuzione delle innovazioni la distribuzione normale asimmetrica e la t di Student asimmetrica. Quest'ultima, grazie alle sue caratteristiche, dovrebbe riuscire ad adempiere ad entrambe le evidenze empiriche.

Allo stato dell'arte sono presenti dei pacchetti software (si veda per chiarimento sezione 2.2) che implementano i modelli GARCH utilizzando diverse "versioni" di queste distribuzioni.

Se per riferirci alle comuni distribuzioni normale e t di Student non vi sono ambiguità, il problema subentra già per la GED, in alternativa alla sopra citata distribuzione di Subbotin. Esse vengono spesso confuse, ma hanno una parametrizzazione leggermente diversa che potrebbe portare a conclusioni affrettate errate.

Per la formulazione della GED rimandiamo all'articolo di Nelson (1991) dove

il parametro ν , proprio della distribuzione, rappresenta la “tail-thickness”; se $\nu = 2$ la distribuzione rappresenta una gaussiana standardizzata.

Per la formulazione della distribuzione di Subbotin rimandiamo invece alla sezione 1.3.3.

Il problema si amplia quando andiamo a considerare le distribuzioni asimmetriche: in letteratura esistono almeno due modi per generare famiglie asimmetriche, il primo proposto da Fernández e Steel (1999) (per inciso utilizzato dai pacchetti software attualmente disponibili), alternativo a quello proposto da Azzalini (1985).

Il secondo metodo risulta preferibile in quanto consente di ritrovare in una famiglia di distribuzioni alcune caratteristiche importanti, come:

- un’adeguata trattabilità matematica ed alcune proprietà formali proprie delle classi parametriche standard;
- un meccanismo semplice di generazione di variabili stocastiche;
- l’opportunità, tramite pochi parametri, di regolare le distribuzioni con elevata flessibilità per la distorsione e per le relative caratteristiche principali, tra cui asimmetria e curtosi.

In base a queste considerazioni nel capitolo successivo saranno presentate delle funzioni per il calcolo di questi modelli che assumano come distribuzione marginale dell’innovazione la versione di Azzalini per la distribuzione normale asimmetrica e t di Student asimmetrica.

1.3.1 La distribuzione *Skew-Normal* (SN)

Negli ultimi anni sono stati molti gli studi condotti nello sviluppo di famiglie parametriche che avessero migliori proprietà in termini inferenziali e di trattabilità. Presentiamo la versione di Azzalini della distribuzione normale asimmetrica.

Dall’articolo di Azzalini (1985) è importante evidenziare il Lemma 1 che ci permette di capire come egli intende affrontare il problema delle distribuzioni asimmetriche.

Lemma 1. Se f_0 è una uni-dimensionale funzione di densità di probabilità simmetrica rispetto a 0 e G è una funzione di distribuzione uni-dimensionale tale che G' esiste ed è una densità simmetrica rispetto a 0, si ha che

$$f(z) = 2 f_0(z) G[w(z)] \quad (-\infty < z < \infty) \quad (1.10)$$

è una funzione di densità per ogni funzione dispari $w(\cdot)$.

Utilizzando questa espressione (1.10) rispettivamente con f_0 funzione di densità della normale standard e con G funzione di ripartizione della normale standard, e $w(z) = \alpha z$ con $\alpha \in R$, si ottiene la distribuzione normale asimmetrica con parametro di forma α , indicata con la notazione $Z \sim SN(\alpha)$.

$$f(z; \alpha) = 2 \phi(z) \Phi(\alpha z) \quad \text{con } z \text{ e } \alpha \in R \quad (1.11)$$

Partendo dalla seguente formula si ottiene $Y = \zeta + \omega Z$, con $\zeta \in R$ e $\omega \in R^+$, che può essere indicata con $Y \sim SN(\zeta, \omega^2, \alpha)$ in cui ζ è il parametro di posizione, ω di scala e α di forma. Per maggiori chiarimenti e dimostrazioni, invitiamo a consultare Azzalini e Capitanò (2003) e Azzalini (2005).

Relative alla distribuzione normale asimmetrica vi sono alcune proprietà che meritano di essere menzionate. Sia per semplicità $Z \sim SN(0, 1, \alpha)$; si ha che:

- se $\alpha = 0$ si ha la densità della normale standard;
- vale che $-Z \sim SN(-\alpha)$;
- per $\alpha \rightarrow \infty$, la densità converge alla mezza normale per $2\phi(z)$ per $z \geq 0$;
- per qualsiasi α , $Z^2 \sim \chi_1^2$.

La distribuzione normale risulta quindi un elemento interno a questa famiglia di distribuzioni.

Per dare un'idea della forma che la curva dell'equazione (1.11) assume si veda la Figura 1.4. Nel grafico sono state inserite le curve per alcune scelte del parametro α positivo. Per α negativi, come conseguenza della seconda proprietà, avremo che il grafico della densità risulta speculare rispetto all'asse verticale.

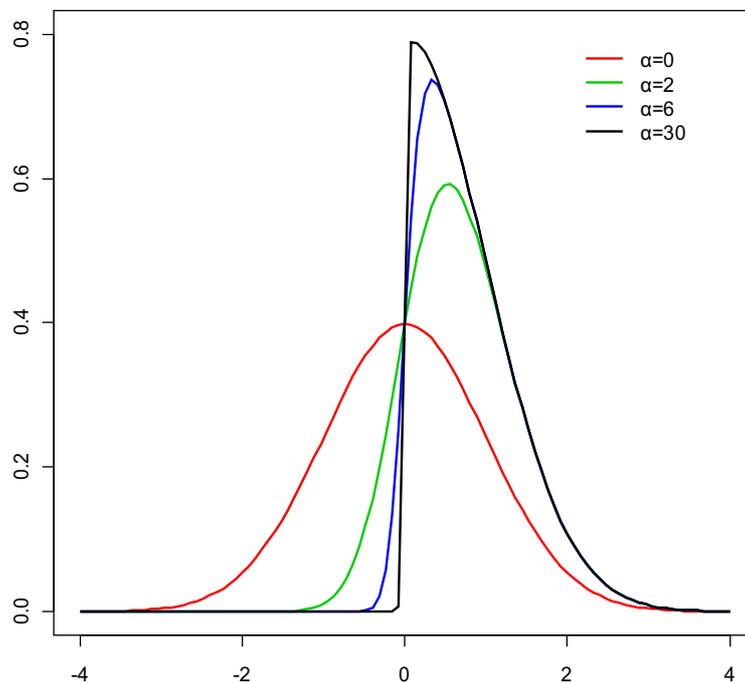


Figura 1.4: Funzione di densità per una $SN(\alpha)$ per alcune scelte di α .

Ve ne sono altre proprietà di livello approfondito; per maggiori dettagli, invitiamo a consultare Azzalini (2005).

Tutti i momenti della distribuzione SN esistono finiti ed hanno una forma analitica semplice. Tuttavia, differiscono sotto diversi aspetti da quelli della distribuzione normale:

- il parametro di posizione ζ e il quadrato del parametro di scala ω corrispondono a media e varianza, come nel caso normale, solo se il parametro di forma α è nullo;
- le code della distribuzione SN sono sempre più leggere rispetto a quelle della normale quando $\alpha \neq 0$;
- i momenti sono esprimibili attraverso il parametro $\delta = \frac{\lambda}{\sqrt{1+\lambda^2}}$ si può infatti scrivere l'espressione per la media di Z come

$$\mu_z = \delta \sqrt{\frac{2}{\pi}}.$$

È possibile quindi esprimere media e varianza di Y in funzione di μ_z , e conseguentemente in funzione di λ , nonché dei parametri di posizione ξ e di scala ω :

$$\mu = E[Y] = \xi + \omega \mu_z$$

$$V[Y] = \omega^2 (1 - \mu_z^2).$$

1.3.2 La distribuzione *Skew-t* (ST)

Un'altra opzione, che permette a livello applicativo di modellare sia l'asimmetria che la pesantezza delle code, è considerare la distribuzione della t di Student asimmetrica nella versione di Azzalini, ottenibile utilizzando nell'espressione (1.10) per f_0 la funzione di densità della t di Student.

Si può definire la distribuzione ST tramite la trasformazione

$$Y = \xi + \omega \frac{Z}{\sqrt{V}}$$

dove $Z \sim SN(\alpha)$ e $V \sim \chi_v^2/v$, variabili casuali tra loro indipendenti.

Lemma 2. Se $V \sim \text{Gamma}(\psi, \eta)$, allora per ogni $a, b \in \mathbb{R}$,

$$E[\Phi(a\sqrt{V} + b)] = P[T \leq a\sqrt{\psi/\eta}]$$

dove T è una variabile t non-centrale con 2ψ gradi di libertà e parametro di non-centralità $-b$.

Applicando il Lemma 2 ad una Gamma $\left(\frac{\psi}{2}, \frac{\psi}{2}\right)$, tramite alcuni semplici passaggi algebrici, è possibile ricavare la funzione di densità per Y .

Si dice che una variabile casuale continua Y ha distribuzione *Skew-t* se la sua densità è del tipo

$$f_Y(y) = \frac{2}{\omega} t_1(z; \nu) T_1\left(\alpha z \sqrt{\frac{\nu+1}{z^2+\nu}}; \nu+1\right), \quad (1.12)$$

dove (ξ, ω, α) sono gli stessi visti nella sezione precedente, $z = \frac{y-\xi}{\omega}$,

$$t_1(z; \nu) = \frac{\Gamma\left(\frac{1}{2}(\nu+1)\right)}{(\pi \nu)^{\frac{1}{2}} \Gamma\left(\frac{1}{2}\nu\right) \left(1 + \frac{z^2}{\nu}\right)^{\frac{\nu+1}{2}}}, \quad (1.13)$$

è la funzione di densità di una t di Student con ν gradi di libertà, e $T_l(x; \nu+1)$ indica la funzione di ripartizione di una t di Student con $\nu+1$ gradi di libertà. In questo caso si scrive $Y \sim St(\xi, \omega, \alpha, \nu)$.

La famiglia appena presentata gode di alcune proprietà interessanti, alcune delle quali ne indicano il legame con SN :

- se $\nu \rightarrow \infty$, la famiglia converge alla densità di una SN ;
- la classe delle densità è chiusa per trasformazioni affini con α e ν invarianti;
- la famiglia permette la scelta tra una gamma illimitata per gli indici di asimmetria e gli indici di curtosi (purché non negativi) per le componenti individuali.

1.3.3 La distribuzione di Subbotin

Un piccolo spazio è giusto sia dedicato anche a questa distribuzione perché è quella che successivamente andremo ad implementare nelle routine. E in quanto tale, è opportuno riferirci ad essa con la dovuta notazione.

La distribuzione di densità della distribuzione di Subbotin, che noi utilizziamo è la seguente

$$f(x; \nu) = \frac{e^{-\frac{|x|^\nu}{\nu}}}{2\nu^{\nu-1} \Gamma\left(\frac{1}{\nu}\right)} \quad \text{con } x \in \mathbb{R} \text{ e } \nu \in \mathbb{R}^+ \quad (1.14)$$

dove ν rappresenta il parametro di forma.

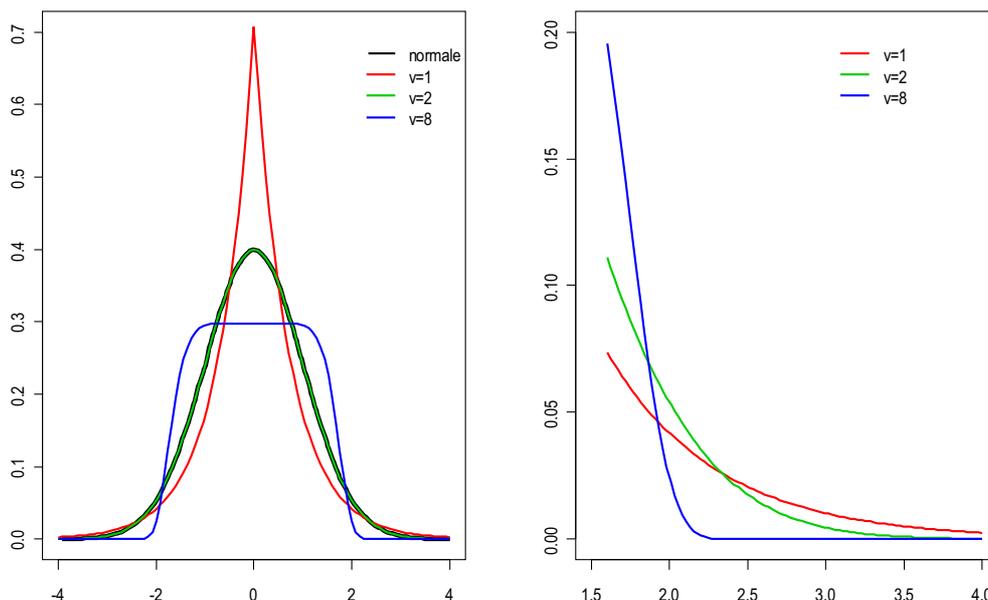


Figura 1.5: Funzione di densità della Subbotin per alcune scelte di ν con particolare attenzione alle code.

Un esempio della forma che assume la distribuzione al variare del parametro ν , si può vedere in Figura 1.5.

1.4 I titoli azionari: oggetto di studio

I prezzi dei titoli che sono oggetto del nostro studio sono stati estratti dal sito www.finance.yahoo.comⁱⁱⁱ. In tale sito è possibile trovare i dati relativi alle quotazioni dei titoli azionari di molte società; noi ci concentriamo su quelli a frequenza giornaliera.

La struttura dei dati disponibili usualmente riporta un insieme di campi standard:

ⁱⁱⁱ Sito dove è possibile trovare gli indici dei principali mercati del mondo.

- la data,
- il prezzo rilevati all'apertura del mercato (*Apertura*),
- il prezzo più alto della giornata (*Massimo*),
- il prezzo più basso della giornata (*Minimo*),
- il prezzo rilevato alla chiusura (*Chiusura*),
- il volume di scambio (espresso in numero di azioni) rilevato durante il giorno (*Volume*),
- il prezzo di chiusura aggiustato per dividendi e split (*Chiusura aggiustata*).

Un esempio di questa struttura è visibile nella Figura 1.6 relativamente alle rilevazioni giornaliere del titolo BASF N dal 12 aprile al 5 maggio 2011.

BASF N (BAS.F)

Scheda BAS.F

QUOTAZIONI
 Riepilogo
 Analisi del book
 Opzioni

Dati storici

DERIVATI
 Warrants
 Prodotti finanziari

GRAFICI
 Interattivo
 Grafico base

INFORMAZIONI
 Notizie
 Eventi aziendali
 Forum

AZIENDA
 Profilo
 Statistiche chiave
 Rapporti alla SEC
 Concorrenti
 Attività
 Componenti

ELENCO ANALISTI
 Opinione degli analisti
 Stime degli analisti
 Analisti Star

PROPRIETÀ
 Maggiori azionisti
 Transazioni insiders
 Elenco insider

DATI FINANZIARI
 Conto economico

Dati storici Trova Dati storici per

IMPOSTA PERIODO

Data iniziale: 4 Ago 2010 Es. Gen 1, 2010
 Scadenza: 5 Mag 2011

Giornaliero
 Settimanale
 Mensile
 Solo dividendi

Ottieni i dati

Primo | Indietro | Avanti | Ultimo

VALORI						
Data	Apertura	Massimo	Minimo	Chiusura	Volume	Chiusura aggiustata*
5 mag 2011	67,00	67,51	65,27	66,18	40.100	64,04
4 mag 2011	67,95	69,21	66,40	66,99	51.000	64,83
3 mag 2011	68,35	68,85	67,84	68,60	22.500	66,38
2 mag 2011	69,80	70,19	68,21	68,50	71.900	66,29
29 apr 2011	68,65	69,50	68,28	69,21	33.900	66,97
28 apr 2011	68,60	69,20	67,95	68,60	35.900	66,38
27 apr 2011	66,80	67,95	66,58	67,80	45.100	65,61
26 apr 2011	66,00	66,70	65,45	66,55	35.000	64,40
21 apr 2011	65,95	66,47	65,62	65,88	53.500	63,75
20 apr 2011	63,70	65,26	63,70	65,09	45.900	62,99
19 apr 2011	62,55	62,86	62,30	62,53	20.000	60,51
18 apr 2011	64,10	64,20	61,98	62,43	30.900	60,41
15 apr 2011	63,25	64,18	63,13	64,18	30.900	62,11
14 apr 2011	62,80	63,31	62,44	63,20	23.700	61,16
13 apr 2011	62,70	63,31	62,62	62,96	25.000	60,93
12 apr 2011	63,40	63,57	62,41	62,73	29.400	60,70

Figura 1.6: Esempio dei dati estraibili dal sito www.yahoo.finance.it.

Capitolo 1. I modelli GARCH e le distribuzioni utilizzate come componente d'errore

Qui di seguito presentiamo un elenco dei titoli azionari del mercato europeo, particolarmente di quello tedesco, analizzati successivamente con relativa descrizione della società a cui il titolo si riferisce e l'abbreviazione utilizzata:

- compagnia chimica BASF N (BAS);
- azienda farmaceutica BAYER AG (BAYN);
- azienda tedesca BMW GROUP (BMW), produttrice di autoveicoli e motoveicoli;
- azienda tedesca DAIMLER N (DAI), produttrice di automobili e di mezzi di trasporto;
- banca tedesca DEUTSCHE BANK (DBK) con sede a Francoforte sul Meno;
- compagnia aerea tedesca DEUTSCHE LUFTHANSA (LHA);
- azienda Linde AG (LIN), fornitrice di un'ampia gamma di gas industriali, gas medicali, gas speciali, refrigeranti e altri prodotti chimici;
- società tedesca Man SE (MAN), produttrice di automezzi pesanti;
- azienda di distribuzione e *cash and carry*^{IV} tedesca METRO AG (MEO);
- azienda multinazionale europea Sap AG (SAP), per la produzione di software;
- azienda K+S Aktiengesellschaft (SDF), fornitrice a livello mondiale di fertilizzanti standard e specializzati;
- azienda SIEMENS N (SIE), produttrice di soluzioni per l'industria, l'energia e la salute;
- azienda siderurgica tedesca Thyssenkrupp AG (TKA);
- azienda automobilistica tedesca VOLKSWAGEN (VOW);
- azienda multinazionale di telefonia cellulare e fissa Vodafone Group plc (VOD).

Le serie storiche analizzate non presentano la stessa lunghezza in quanto non

^{IV} Il cash and carry, letteralmente "paga e porta via", rappresenta una tipologia di operatore della grande distribuzione organizzata (GDO).

si voleva incorrere in questa eventuale particolarità.

In particolare abbiamo a che fare con lunghezze che spaziano da circa 1800 osservazioni per arrivare ad un massimo di 5000: si osservi che circa 250-260 osservazioni coprono l'orizzonte temporale di un anno.

1.5 Analisi preliminare

Prima di procedere allo studio diretto dei modelli GARCH e della stima dei parametri per questi modelli, la nostra attenzione si è focalizzata in alcuni test sui parametri che indicano curtosi e asimmetria delle distribuzioni. Questo passaggio è necessario per verificare con esempi concreti sulle serie a nostra disposizione le evidenze empiriche sopra menzionate e trarre alcune considerazioni. Tale analisi è stata condotta con gli strumenti software attualmente disponibili: si rimanda al capitolo successivo per una descrizione più dettagliata.

Quello che è importante chiarire fin da subito è la notazione utilizzata per le tabelle che qui sotto verranno presentate. Per identificare le distribuzioni utilizzate per la componente d'errore, la "n" indica la distribuzione di densità della normale, la "t" della t di Student, la "ged" della distribuzione generalizzata degli errori, "st" della t di Student asimmetrica e "sged" della distribuzione degli errori asimmetrica. Mentre per le serie, sono stati utilizzati le abbreviazioni presentate nella sezione precedente. Inoltre per identificare i parametri: con "skew", indichiamo il parametro che modella l'asimmetria della distribuzione e con "shape" il parametro che modifica la coda della distribuzione e quindi la curtosi; "s.e." sta per *standard error* riferito alla stima del parametro.

Un nota particolare va fatta per la sigla "c.r." che sta ad indicare che nel calcolo per ottenere i parametri è stato riscontrato un errore: nel calcolo della matrice hessiana in quanto il sistema in considerazione nel caso specifico è risultato singolare.

Capitolo 1. I modelli GARCH e le distribuzioni utilizzate come componente d'errore

R - package fGarch -													
	n	t		ged		st				sged			
DATI		shape	s.e.	shape	s.e.	skew	s.e.	shape	s.e.	skew	s.e.	shape	s.e.
BAS	rif.	6.70	1.045	1.37	0.062	0.99	0.030	6.71	1.046	1.00	0.032	1.37	0.063
BAYN	rif.	4.23	0.353	c.r		1.02	0.025	4.22	0.352	c.r			
BMW	rif.	7.74	1.320	1.46	0.066	1.01	0.032	7.75	1.327	1.02	0.024	1.46	0.066
DAI	rif.	5.89	0.795	1.27	0.057	0.99	0.030	5.88	0.794	1.01	0.026	1.27	0.057
DBK	rif.	6.61	0.940	1.35	0.059	0.99	0.032	6.61	0.940	1.00	0.025	1.35	0.059
DPW	rif.	4.94	0.500	c.r		1.02	0.026	4.94	0.500	c.r.			
LHA	rif.	6.43	0.909	1.33	0.059	1.05	0.032	6.37	0.903	1.04	0.024	1.33	0.059
LIN	rif.	5.84	0.817	1.29	0.058	1.05	0.032	5.97	0.856	1.05	0.024	1.30	0.059
MAN	rif.	6.08	0.786	c.r		1.02	0.033	6.08	0.786	c.r.			
MEO	rif.	4.99	0.577	1.26	0.054	1.02	0.032	4.99	0.576	1.01	0.019	1.26	0.054
SAP	rif.	4.28	0.417	1.09	0.046	1.03	0.031	4.28	0.418	1.02	0.017	1.09	0.044
SDF	rif.	4.09	0.341	1.00	0.043	1.04	0.025	4.13	0.347	1.04	0.016	1.00	0.040
SIE	rif.	5.40	0.639	1.27	0.053	0.99	0.032	5.39	0.637	1.00	0.025	1.27	0.054
TKA	rif.	5.20	0.604	c.r		0.98	0.030	5.21	0.607	c.r.			
VOW	rif.	4.49	0.462	c.r		1.12	0.036	4.61	0.486	c.r.			

Tabella 1.1: Parametri calcolati con libreria *fGarch*.

R - package rgarch -													
	n	t		ged		st				sged			
DATI		shape	s.e.	shape	s.e.	skew	s.e.	shape	s.e.	skew	s.e.	shape	s.e.
BAS	rif.	6.77	1.067	1.38	0.063	0.98	0.030	6.78	1.069	1.00	0.033	1.38	0.063
BAYN	rif.	4.21	0.353	1.03	0.036	1.01	0.025	4.20	0.350	1.01	0.002	1.04	0.041
BMW	rif.	7.66	1.302	1.45	0.066	1.01	0.032	7.67	1.305	1.02	0.024	1.45	0.066
DAI	rif.	5.93	0.808	1.28	0.057	0.99	0.030	5.92	0.805	1.00	0.026	1.28	0.057
DBK	rif.	6.60	0.940	1.35	0.059	0.99	0.032	6.60	0.940	1.00	0.027	1.35	0.059
DPW	rif.	4.96	0.505	1.35	0.059	1.02	0.027	4.96	0.505	1.00	0.027	1.35	0.059
LHA	rif.	6.70	0.993	1.34	0.061	1.04	0.035	6.64	1.161	1.00	0.027	1.35	0.074
LIN	rif.	5.78	0.806	1.28	0.058	1.05	0.031	5.90	0.843	1.05	0.023	1.29	0.059
MAN	rif.	6.11	0.794	1.28	0.058	1.01	0.033	6.10	0.792	1.05	0.033	1.20	0.141
MEO	rif.	5.05	0.569	1.27	0.065	1.02	0.030	5.05	0.568	1.01	0.021	1.27	0.064
SAP	rif.	4.28	0.419	1.08	0.067	1.03	0.031	4.28	0.420	1.02	0.002	1.08	0.044
SDF	rif.	4.08	0.341	1.00	0.043	1.04	0.025	4.12	0.347	1.03	0.001	0.98	0.017
SIE	rif.	5.42	0.645	1.27	0.054	0.99	0.032	5.41	0.642	1.00	0.024	1.27	0.054
TKA	rif.	5.16	0.600	1.09	0.038	0.98	0.030	5.18	0.603	0.97	0.002	1.10	0.038
VOW	rif.	4.47	0.461	1.13	0.048	1.12	0.036	4.58	0.485	1.09	0.024	1.16	0.073

Tabella 1.2: Parametri calcolati con libreria *rgarch*.

S-PLUS					
	n	t		ged	
DATI		shape	s.e.	shape	s.e.
BAS	rif.	6.70	1.134	1.37	0.061
BAYN	rif.	4.23	0.314	1.06	0.020
BMW	rif.	7.69	1.421	1.45	0.063
DAI	rif.	5.94	0.768	1.29	0.048
DBK	rif.	6.64	0.876	1.36	0.047
DPW	rif.	5.00	0.398	1.12	0.016
LHA	rif.	6.39	0.842	1.34	0.051
LIN	rif.	5.82	0.873	1.30	0.056
MAN	rif.	6.18	0.552	1.23	0.020
MEO	rif.	4.97	0.680	1.27	0.053
SAP	rif.	4.27	0.389	1.09	0.036
SDF	rif.	4.12	0.312	1.04	0.024
SIE	rif.	5.41	0.630	1.29	0.046
TKA	rif.	5.18	0.433	1.13	0.017
VOW	rif.	4.46	0.443	1.15	0.039

Tabella 1.3: Parametri calcolati con libreria *FinMetrics*.

Capitolo 1. I modelli GARCH e le distribuzioni utilizzate come componente d'errore

Le tabelle mettono in risalto quanto detto nella sezione 1.1 riguardante la non adattabilità ai dati del modello con componente d'errore gaussiana.

Possiamo osservare che nella colonna della “n” è stato posta una serie di “rif.” che sta per “rifiutata”. Questo “rifiutata” sta ad indicare che il test di Jacque-Bera^v che viene fatto sui residui del modello per verificare l'ipotesi di normalità, rifiuta l'ipotesi di normalità. Tale rifiuto può implicare due alternative:

- il rifiuto dell'ipotesi di simmetria della distribuzione e, quindi, il rifiuto dell'ipotesi di normalità;
- il rifiuto della normalità, ma non necessariamente il rifiuto dell'ipotesi di simmetria nella distribuzione dei rendimenti delle serie.

Si può inoltre notare che le stime per i parametri considerati, ottenute con i tre diversi pacchetti, sono abbastanza omogenee e a tal riguardo sembrano indicare tutte, essendo il parametro *skew* circa uguale 1, che le distribuzioni dei rendimenti sono pressoché simmetriche. A tal ragione non facciamo altre analisi, ma consideriamo tale caratteristica qualcosa caratterizzante queste serie scelte.

^v Il test di Jacque-Bera è un test statistico per la verifica dell'ipotesi di normalità ed è impiegato molto spesso in campo econometrico. Esso si basa sulla misura dell'asimmetria e della curtosi di una distribuzione.

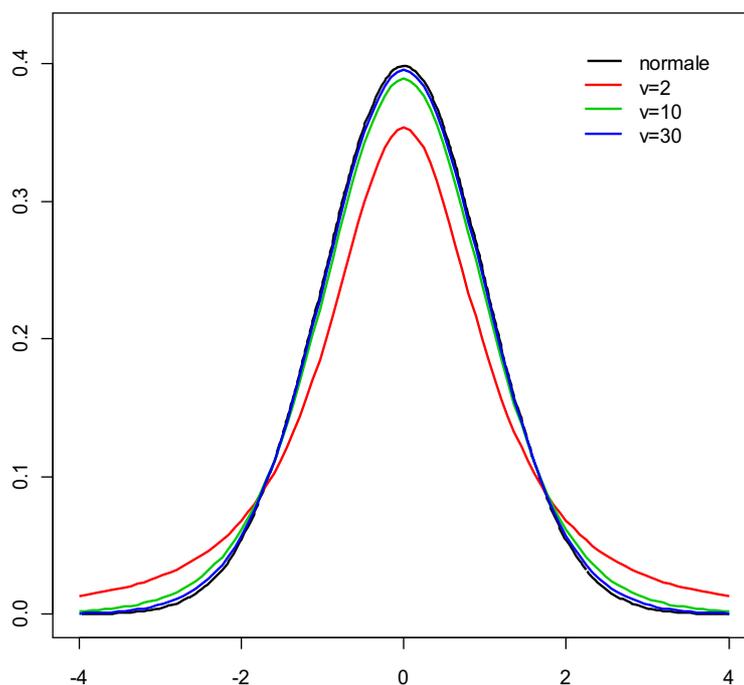


Figura 1.7: Funzione di densità per una $t(v)$ per alcune scelte di v .

Per quanto riguarda invece l'altro parametro, *shape*, rimanendo un valore reale positivo, esso ha variabilità e quindi significato differente al cambio della distribuzione considerata. Infatti nel caso della t di Student, esso viene anche chiamato “gradi di libertà” e più questo valore è piccolo (v circa 4) più le code sono spesse e più invece, questo valore è grande, sopra il 10, più le code diventano sottili. Si ricorda che per un *shape* circa uguale a 30, la distribuzione della t di Student tende ad assomigliare alla gaussiana come si può vedere in Figura 1.7.

Per quanto riguarda invece la distribuzione generalizzata degli errori, il campo di variabilità del parametro è diverso, seppur rimanendo un numero reale positivo:

- se è minore di 2, le code sono sottili con riferimento alla gaussiana;
- se è maggiore di 2, le code sono più spesse della stessa;
- se è uguale a 2, la distribuzione di distribuzione generalizzata degli errori coincide con la gaussiana.

Capitolo 1. I modelli GARCH e le distribuzioni utilizzate come componente d'errore

Per meglio capire come tale parametro influisce nella “forma” della distribuzione e quindi nella pesantezza delle code, calcoliamo l'indice di curtosi per queste distribuzioni, dato dal momento quarto centrato standardizzato.

Si consideri un processo Y_t con media μ e varianza σ^2 . Essendo:

$$\mu_r = E[(y - \mu)^r]$$

l' r -esimo momento centrato di Y_t con $\mu_2 = \sigma^2$, allora l'indice di curtosi è definito come:

$$\beta_2 = \frac{\mu_3}{\mu_2^{3/2}} = \frac{E[(y - \mu)^3]}{E[(y - \mu)^2]^{3/2}}$$

Nel caso specifico della t di Student, sostituendo i momenti opportuni e considerando ν , si ottiene:

$$\beta_2 = 3 + \frac{6}{\nu - 4} \quad \text{se } \nu > 4$$

mentre per la GED, considerando $\text{shape} = p$, si ottiene:

$$\beta_2 = \frac{\Gamma\left(\frac{1}{p}\right)\Gamma\left(\frac{5}{p}\right)}{\left[\Gamma\left(\frac{3}{p}\right)\right]^2} \quad \text{se } p > 1$$

Nella tabelle che seguono si è pensato di esaminare come varia l'indice di curtosi al variare del parametro *shape*. A tal riguardo abbiamo preso le stime di tal parametro, ottenute con l'ambiente S-PLUS e il corrispondente pacchetto che implementa i modelli GARCH, e ne abbiamo visto la sua variabilità, calcolando un intervallo di confidenza del 95%. Abbiamo applicato la formula dell'indice di curtosi anche agli estremi dell'intervallo, in modo da esaminare la sua variabilità. Come si può notare nella Tabella 1.4 nel caso della t di Student la variabilità per l'indice di curtosi è elevatissima, si raggiunge anche l'infinito per più di qualche

titolo.

t							
DATI	shape	s.e.	Coef. curtosi	shape - 2*s.e	Coef. curtosi	shape + 2*s.e	Coef. curtosi
BAS	6.70	1.134	5.22	4.43	16.94	8.97	4.21
BAYN	4.23	0.314	29.50	3.60	inf	4.85	10.03
BMW	7.69	1.421	4.63	4.85	10.09	10.53	3.92
DAI	5.94	0.768	6.09	4.40	17.90	7.48	4.73
DBK	6.64	0.876	5.28	4.88	9.79	8.39	4.37
DPW	5.00	0.398	8.99	4.21	32.25	5.80	6.34
LHA	6.39	0.842	5.51	4.71	11.50	8.07	4.47
LIN	5.82	0.873	6.29	4.08	82.22	7.57	4.68
MAN	6.18	0.552	5.75	5.08	8.56	7.29	4.83
MEO	4.97	0.680	9.17	3.61	inf	6.33	5.57
SAP	4.27	0.389	25.28	3.49	inf	5.05	8.73
SDF	4.12	0.312	52.13	3.50	inf	4.75	11.03
SIE	5.41	0.630	7.25	4.15	42.49	6.67	5.25
TKA	5.18	0.433	8.08	4.32	22.02	6.05	5.93
VOW	4.46	0.443	15.98	3.58	inf	5.35	7.45

Tabella 1.4: Indici di curtosi calcolati per la distribuzione t di Student.

ged							
DATI	shape	s.e.	Coef. curtosi	shape - 2s.e.	Coef. curtosi	shape + 2s.e.	Coef. curtosi
BAS	1.37	0.061	4.11	1.25	4.53	1.49	3.79
BAYN	1.06	0.020	5.54	1.02	5.84	1.10	5.28
BMW	1.45	0.063	3.88	1.33	4.23	1.58	3.59
DAI	1.29	0.048	4.37	1.19	4.79	1.38	4.08
DBK	1.36	0.047	4.14	1.26	4.49	1.45	3.88
DPW	1.12	0.016	5.16	1.09	5.34	1.16	4.94
LHA	1.34	0.051	4.20	1.23	4.61	1.44	3.91
LIN	1.30	0.056	4.34	1.19	4.79	1.41	3.99
MAN	1.23	0.020	4.61	1.19	4.79	1.27	4.45
MEO	1.27	0.053	4.45	1.17	4.89	1.38	4.08
SAP	1.09	0.036	5.34	1.02	5.84	1.16	4.94
SDF	1.04	0.024	5.68	0.99	6.09	1.09	5.34
SIE	1.29	0.046	4.37	1.20	4.74	1.38	4.08
TKA	1.13	0.017	5.10	1.10	5.28	1.16	4.94
VOW	1.15	0.039	4.99	1.07	5.47	1.23	4.61

Tabella 1.5: Indici di curtosi calcolati per la distribuzione di Subbotin.

Capitolo 2.

Le funzioni per il calcolo del modello

GARCH

In questo capitolo presentiamo come siamo arrivati alle stime dei parametri dei modelli GARCH con una particolare attenzione ai problemi riscontrati. Sono presentate le librerie e i metodi che implementano questo tipo di modelli per serie storiche finanziarie, ma anche le routine che è stato necessario creare per la stima dei parametri con le distribuzioni asimmetriche nella versione di Azzalini.

Nella sezione 2.1 facciamo una breve introduzione degli ambienti statistici e delle relative librerie utilizzate che implementano i modelli di nostro interesse. Con particolare attenzione alle funzioni che svolgono la stima dei parametri del modello.

Nella sezione 2.2 presentiamo le routine create per il calcolo della log-verosimiglianza del modello GARCH(1,1) con componente d'errore normale, soffermandoci su come abbiamo rispettato i vincoli del modello e, per maggior chiarezza, sulla notazione all'interno delle routine.

Nella sezione 2.3 parliamo delle chiamate alla funzione *optim*, sviluppata in R nella libreria *base*, che implementa gli algoritmi di ottimizzazione per la massimizzazione delle funzioni di log-verosimiglianza. Infine nella sezione 2.4 presentiamo dei test sugli algoritmi di ottimizzazione e sul modo di introdurre i vincoli, oltre che ad un confronto di stabilità tra le stime dei parametri per il modello GARCH che si ottengono con le nostre routine e con quelle ottenute con le funzioni delle librerie.

2.1 Ambienti statistici a confronto e rispettive librerie che implementano modelli GARCH

S-PLUS^I e R^{II} sono ambienti statistici per l'analisi di dati, nati come estensioni del linguaggio di programmazione S sviluppato nei Laboratori Bell americani dell'AT&T, con contributi principali da parte di Rick Becker, John Chambers ed Allan Wilks. I due ambienti sono molto simili, tanto che per l'utente è possibile passare da un ambiente all'altro con grande facilità.

S-PLUS è stato creato da Douglas Martin e successivamente ceduto alla società Statistical Sciences (StatSci), attualmente di proprietà della Mathsoft Inc., che ne cura l'aggiornamento e la commercializzazione.

R è invece un ambiente statistico di pubblico dominio (open source) che, nel rispetto dei diritti d'autore, può essere copiato, modificato da chiunque e distribuito con licenza *GNU GPL*. Inizialmente scritto da Ross Ihaka e Robert Gentleman del Dipartimento di Statistica dell'Università di Auckland, Nuova Zelanda, dal 1997 lo sviluppo di R è affidato ad un gruppo internazionale, *l'R Core Team*. La sua popolarità è dovuta all'ampia disponibilità di moduli distribuiti con la licenza GPL e organizzati in un apposito sito chiamato CRAN (*Comprehensive R Archive Network*)^{III}.

R ha una struttura modulare che prevede l'utilizzo di librerie opzionali, chiamate "*packages*" che possono essere sviluppate anche da soggetti esterni e vengono comunque raccolte dal CRAN. In particolare per il nostro lavoro di tesi, abbiamo utilizzato *fGarch* e *rgarch*.

Il primo fa parte del progetto Rmetrics^{IV}, un'associazione no-profit che opera nell'innovazione del calcolo finanziario. Rmetrics sviluppa software open source che coprono i settori dell'analisi esplorativa dei dati, della modellazione statistica e della creazione di modelli. Il progetto è stato proposto nel 2001 da Diethelm Wuertz, all'interno del Swiss Federal Institute of Technology di Zurigo. La libreria *fGarch* è orientata alla modellazione di processi GARCH e copre sia la

^I Versione 8.0 (maggio del 2007)

^{II} Versione 2.11.0 (22 aprile 2010)

^{III} Per ulteriori informazioni, consultare il sito <http://cran.r-project.org>

^{IV} Per ulteriori informazioni, consultare il sito <http://rmetrics.org>

simulazione che la stima delle serie finanziarie.

Al contrario, *rgarch* nasce all'interno di R-Forge, una piattaforma per lo sviluppo collaborativi del software basato sul linguaggio R: la piattaforma nasce nel 2007 e propone ad oggi oltre 400 progetti che coprono vari settori della ricerca tra cui bioinformatica, data mining ed analisi delle serie finanziarie. La libreria *rgarch* è anch'essa orientata alla modellazione GARCH, e copre l'inferenza, la simulazione e la previsione delle serie.

S+FinMetrics™ è un modulo commerciale di S-PLUS sviluppato da Andrew Bruce, Doug Martin, Jiahui Wang, Eric Zivot all'interno della Insightful Corporation: prevede funzionalità avanzate per la modellazione, analisi e la visualizzazione di dati dei mercati finanziari.

2.1.1 La libreria *fGarch* e le sue funzioni

Vista l'importanza sempre più grande avuta negli ultimi anni dai modelli GARCH nell'analisi delle serie storiche in particolare nelle applicazioni finanziarie con lo scopo di analizzare e prevedere la volatilità, sono state realizzate varie librerie con l'intento di implementare questi modelli. Una fra queste, è la libreria *fGarch*, sviluppata da Diethelm Wuertz nel novembre del 2009. Noi utilizziamo la versione 2110.80.

Questa libreria mette a disposizione una famiglia di funzioni per simulare, stimare e prevedere nel tempo vari tipi di modelli univariati GARCH nella varianza condizionata.

Principalmente, noi utilizziamo la funzione *garchFit*, che è una implementazione numerica della stima di massima verosimiglianza sotto diverse assunzioni distributive, normale, *t* di Student, GED e le loro versioni asimmetriche. Le stime dei parametri sono verificate con diversi strumenti di analisi diagnostica che includono funzioni grafiche e test di ipotesi. Sono disponibili anche funzioni per calcolare previsioni future sotto entrambe le ipotesi di media e varianza condizionata.

La funzione *garchSpec* descrive le specifiche di un processo GARCH che possiamo usare per la simulazione di modelli artificiali. Questo risulta molto utile

per verificare i risultati delle stime dei parametri in quanto tali valori sono specificati a priori.

La funzione *garchSim* simula un processo temporale GARCH univariato descritto dal modello definito come argomento.

2.1.2 La libreria *rgarch* e le sue funzioni

Un'alternativa è la libreria *rgarch*, sviluppata da Alexios Ghalanos nel marzo del 2010; noi utilizziamo la versione 1.70.

Questa libreria ha lo scopo di fornire un ambiente GARCH di modellazione e di test flessibile e completo: la modellazione comprende la definizione delle specifiche del processo che sono seguite dalla stima dei parametri. È possibile fare inferenza attraverso il *summary*, test e grafici, che sono accompagnati da strumenti di previsione, filtraggio e simulazione.

La funzione *ugarchspec* consente di creare le specifiche di un modello GARCH univariato necessarie per fare inferenza, comprendendo una vasta scelta di modelli univariati, distribuzioni ed equazioni per modellare la media.

La funzione *ugarchfit* permette la stima di modelli GARCH univariati. La routine di ottimizzazione inizia con il calcolo di un insieme di punti di partenza ammissibili che inizializzano la ricorsione, mentre la componente principale per il calcolo della verosimiglianza è costruita in codice C per motivi di performance.

La funzione *ugarchsim* affianca le precedenti introducendo funzionalità per la simulazione.

2.1.3 Il modulo S+FinMetrics

S+FinMetrics è un modulo di S-PLUS per la modellazione econometrica e la previsione di serie temporali economiche e finanziarie. Tale modulo è stato proposto da Douglas Martin, e sviluppato principalmente da Jiahui Wang, Quan Wen e Hui Huang. Noi utilizziamo la versione 2.0.4 del settembre del 2006.

La funzione *garch*, da noi utilizzata, consente la stima di un modello GARCH attraverso il passaggio di due parametri che specificano l'equazione della media condizionata e quella della varianza condizionata. È possibile specificare un

argomento (*cond.dist*) opzionale per modificare la distribuzione per la componente d'errore del modello: di default è utilizzata la distribuzione normale, ma è possibile richiedere la t di Student, la *double exponential* e la GED. Facciamo notare la mancanza di distribuzioni asimmetriche.

2.2 Calcolo di funzione di log-verosimiglianza per modelli GARCH con distribuzioni asimmetriche per componente d'errore

I moduli presentati nella sezione precedente per l'implementazione dei modelli GARCH utilizzano la versione di Fernández e Steel per le distribuzioni asimmetriche. Avendo analizzato le migliori caratteristiche della versione di Azzalini nel meccanismo di creazione di distribuzioni asimmetriche, abbiamo creato delle routine che calcolano la funzione di log-verosimiglianza per tutte le distribuzioni in considerazione nella sezione 1.5.

Proponiamo qui di seguito una presentazione del codice che calcola la log-verosimiglianza, per semplicità e chiarezza, sotto le ipotesi di un GARCH(1,1) con media costante e con distribuzione normale per la componente d'errore e rimandiamo all'appendice A per la copertura delle altre casistiche.

Ci siamo serviti come punto di partenza degli appunti del corso di Serie Storiche Finanziarie corso progredito (anno 2009/2010), tenuto nella facoltà di Scienze Statistiche dell'Università di Padova.

Nel capitolo precedente abbiamo parlato dei processi stocastici teoricamente in grado di generare le osservazioni campionarie a disposizione; infatti, abbiamo definito dei processi che dipendono da parametri. Per poter utilizzare questi processi generatori come modelli adatti a descrivere la serie storica osservata dobbiamo poter attribuire dei valori numeri ai parametri incogniti, attraverso la procedura di stima.

Si noti che le variabili casuali che definiscono il processo stocastico hanno una funzione di densità di probabilità congiunta

$$f(y_1, y_2, \dots, y_n | \theta) \quad (2.1)$$

che si suppone nota nella forma a meno della conoscenza dei parametri.

Noi sappiamo che la funzione di verosimiglianza è alla base del metodo di stima dei parametri più diffuso, noto come metodo di stima di massima verosimiglianza, che consiste nella ricerca dei valori dei parametri incogniti che rendono massimo il valore della funzione di verosimiglianza. La funzione di verosimiglianza di un campione qualsiasi y_1, y_2, \dots, y_n coincide con la funzione di densità di probabilità congiunta:

$$L(\theta) = f(y_1, y_2, \dots, y_n | \theta) \quad (2.2)$$

Avendo sottolineato nella sezione 1.1 che i rendimenti non sono indipendenti, la densità congiunta dell'intero campione a disposizione può essere fattorizzata sequenzialmente come segue

$$L(\theta) = f(y_p, y_{p-1}, \dots, y_1 | \theta) \prod_{t=p+1}^n f(y_t | I_{t-1}; \theta) \quad (2.3)$$

ottenendo un prodotto di funzioni di densità di probabilità di una variabile casuale (nel nostro caso il processo stocastico r_t) in un certo istante di tempo, condizionata ai valori assunti dalle variabili casuali precedenti e dalla funzione di probabilità congiunta di un numero p di variabili riferite ai periodi iniziali.

La procedura di massimizzazione viene convenientemente implementata sul logaritmo naturale della funzione di verosimiglianza, dato che, in quanto monotona crescente, la trasformazione logaritmica non altera i punti di massimo e minimo di una funzione. Esprimiamo la funzione di verosimiglianza per le procedure di inferenza tramite la funzione di log-verosimiglianza. Questa trasformazione monotona crescente logaritmica offre alcuni vantaggi per l'esecuzione pratica dei calcoli e rende più semplice la descrizione dei risultati teorici:

2.2 Calcolo di funzione di log-verosimiglianza per modelli GARCH con distribuzioni asimmetriche per componente d'errore

$$l(\theta) = \log(f(y_p, y_{p-1}, \dots, y_1 | \theta)) + \sum_{t=p+1}^n \log(f(y_t | I_{t-1}; \theta)) \quad (2.4)$$

dove, ai fini della procedura di stima, il primo addendo può essere trascurato per n sufficientemente grande.

Facendo riferimento alla formulazione del modello nell'espressione (1.9) e sotto le ipotesi in cui ci siamo messi la funzione di log-verosimiglianza approssimata per un GARCH(1,1) gaussiano da calcolare è:

$$l(\theta) \simeq -\frac{1}{2} \sum_{t=2}^n \log(\omega + \alpha_1 (r_{t-1} - \mu)^2 + \beta_1 \sigma_{t-1}^2) + \frac{1}{2} \sum_{t=2}^n \frac{(r_t - \mu)^2}{\omega + \alpha_1 (r_{t-1} - \mu)^2 + \beta_1 \sigma_{t-1}^2} \quad (2.5)$$

dove i parametri da stimare sono $\theta = (\mu, \omega, \alpha_1, \beta_1)$.

Particolare attenzione va prestata ai vincoli di non negatività e di stazionarietà del modello presentati nella sezione 1.2.

Abbiamo adottato tre strade per assicurare che i vincoli venissero rispettati nelle funzioni. La prima: si è pensato di inserirli nel codice di queste funzioni con delle semplici condizioni “IF...ELSE”. Se il vincolo non è soddisfatto, si fa ritornare alla routine un valore di log-verosimiglianza tendente a meno infinito che sicuramente non rappresenterà un valore valido nell’ottimizzazione.

La seconda, un po’ più complessa dal punto di vista organizzativo, ma come vedremo in seguito con migliori prestazioni, è di trasformare questi parametri, attraverso la mappatura di funzioni matematiche: ad esempio la funzione esponenziale, che avendo come dominio solo i numeri reali positivi, non permette che il parametro risulti negativo (caso $\omega, \alpha_1, \beta_1$).

La terza, combina le caratteristiche delle precedenti in quanto, come vedremo, i vincoli sui singoli parametri sono gestiti dal metodo di ottimizzazione e quello invece della stazionarietà che comprende due parametri contemporaneamente, è gestito direttamente all’interno della funzione di log-verosimiglianza con “IF...ELSE”.

Per far più chiarezza sul codice che verrà qui di seguito presentato, si identifica:

- $theta = \theta$ = vettore dei parametri da stimare,
- $r = r_t$ = rendimenti logaritmici sopra definiti,
- n = lunghezza serie,
- $mu = \mu$,
- $w = \omega$,
- $a1 = \alpha_1$,
- $b1 = \beta_1$,
- $h = \sigma^2$,
- lmv = il valore della funzione di log-verosimiglianza.

Per indicare la versione della funzione utilizzata, ci baseremo su una struttura del tipo “mvdXVY”, dove X rappresenta la distribuzione utilizzata e Y la specifica dei vincoli.

In particolare X può assumere i seguenti valori:

- n per la distribuzione normale;
- sn per la distribuzione normale asimmetrica;
- t per la t di Student;
- st per la t di Student asimmetrica;
- sub per la distribuzione di Subbotin.

Allo stesso tempo Y varia tra le tre opzioni:

- $V1$ per i vincoli con le condizioni “IF...ELSE”;
- $V2$ per i vincoli con la “rimappatura” dei parametri;
- $V3$ per i vincoli con metodo di ottimizzazione “L-BFGS-B”.

Ad esempio, nel caso della distribuzione normale, le routine sono chiamate: “mvdnV1”, “mvdnV2” e “mvdnV3”. Per la visione del codice, si rimanda

all'appendice A.

È stato presentato il caso normale, ma per le altre distribuzioni, il codice non cambia di molto. Si aggiungono ai parametri del modello, nuovi parametri da stimare che sono quelli propri della distribuzione, quali il parametro di forma (ν) e il parametro di asimmetria (α). Oltre a questi vincoli quello che cambierà è la formula per il calcolo della variabile “*lmv*”.

A tal riguardo, noi facciamo utilizzo di un'ulteriore libreria scaricabile nel sito CRAN per le funzioni di densità della distribuzione normale asimmetrica (“*dsn*”) e *t* di Student asimmetrica (“*dst*”). Questa libreria si chiama “*sn*”^v ed è stata realizzato da Azzalini il 2 aprile 2010. Noi utilizziamo la versione 0.4-15.

2.2.1 Particolarità nel calcolo del stima della log-verosimiglianza

Come è stato accennato sopra per scrivere queste funzioni ci siamo basati sugli appunti del corso di Serie Storiche Finanziarie, tenuto in questa facoltà. Quello che abbiamo notato essere di particolare rilievo nel calcolo della stima della funzione di log-verosimiglianza quando viene applicato ai dati è che nel calcolo si utilizza effettivamente l'espressione (2.3), trascurando il primo addendo. Infatti come avevamo anticipato esso è ininfluenza nel calcolo.

Nella nostra routine, conforme a quanto detto non lo abbiamo incluso nel calcolo, in quanto riteniamo di lavorare con n sufficientemente elevati. Per specificare questa accortezza è sono state inserite due semplici istruzioni, che qui di seguito, evidenziamo in rosso:

```
...  
for (i in 2:n) h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]  
r <- r[2:n]  
h <- h[2:n]  
lmv <- sum(dnorm(r, mean=mu, sd=sqrt(h), log=TRUE))  
...
```

^v Per ulteriori informazioni, consultare <http://azzalini.stat.unipd.it/SN>.

2.3 Utilizzo di `optim` per il calcolo della stima di massima verosimiglianza

Le routine che sopra abbiamo presentato calcolano solamente il valore della funzione di log-verosimiglianza che viene restituito con il comando `return(lmv)`. Quello che a noi interessa però è la stima di massima verosimiglianza, cioè quel valore del vettore θ che massimizza la log-verosimiglianza. Per far questo abbiamo bisogno di un algoritmo di ottimizzazione numerica che individui il massimo della funzione.

Nell'ambiente R sono implementati i maggiori algoritmi di ottimizzazione numerica nella funzione `optim` che prende come argomenti principali il valore di partenza e la funzione da ottimizzare.

Abbiamo provato a sviluppare i calcoli con il metodo di default *Nelder-Mead*^{vi} (“NM”), con quello di *quasi-Newton*^{vii} (“BFGS”) e con quello di *quasi-Newton con vincoli*^{viii} (“L-BFGS-B”). Quello con cui abbiamo deciso di sviluppare le nostre ottimizzazioni è il secondo per le migliori prestazioni ed l'affidabilità delle stime ottenute se confrontate con i moduli già esistenti.

Di default, la funzione `optim` minimizza le funzioni che le vengono passate come argomento, è stato quindi necessario inserire un altro argomento, quale `fnscale = -1` per fare in modo che la funzione venisse massimizzata.

Oltre agli argomenti già citati, è necessario aggiungere anche gli argomenti richiesti dalla funzione che vogliamo massimizzare, oltre ai parametri da stimare, ossia la serie dei rendimenti (`r_serie`) e la lunghezza della serie (`n_serie`).

Ponendo `hessian = TRUE` viene ritornata anche la matrice hessiana ottenuta numericamente, necessaria per un calcolo successivo degli *standar error* delle stime dei parametri ottenuti.

Presentiamo qui una chiamata di tale funzione:

^{vi} Questo metodo è un'implementazione di quello di Nelder and Mead (1965) che usa solo i valori puntuali della funzione.

^{vii} Metodo pubblicato nel 1970 da Broyden, Fletcher, Goldfarb and Shanno, basato sia sui valori puntuali che sul calcolo del gradiente.

^{viii} Metodo proposto da Byrd et al. nel 1995. è una variante di BFGS che introduce dei limiti inferiori e superiori per ciascuna variabile.

```
optim(  
  par = init_norm1,  
  fn = mvdnV31,  
  method = "BFGS",  
  r = r_serie,  
  n = n_serie,  
  control = list(fnscale=-1),  
  hessian = TRUE)
```

Come si può notare passiamo a `par` come argomento `init_norm1`, un vettore di quattro parametri che servono come valori iniziali. Il fatto che il termine finisca con “1” non è casuale; infatti per non incorrere nel problema che venga restituito un massimo locale, parallelamente facciamo partire un'altra chiamata ad `optim` con un altro vettore iniziale, in modo che da un confronto dei valori di log-verosimiglianza, si vada a prendere la più grande e si arrivi sperabilmente al massimo globale.

Il fatto di fare queste due chiamate ad `optim` non ci assicura al 100%, che si arriverà al massimo globale, ma con vari tentativi fatti, provando anche a far partire `optim` da un terzo vettore iniziale, si può ritenere sufficiente questa verifica. Un terzo vettore, avrebbe notevolmente alzato il tempo necessario ad ottenere le stime, senza portare a nessun risultato di rilievo.

La chiamata ad `optim` restituisce un vettore θ contenente la stima di massima verosimiglianza dei parametri del modello e il corrispondente valore della funzione di log-verosimiglianza calcolata nel punto di massimo.

Utilizzando la seconda strada per i vincoli, che abbiamo presentato nella sezione precedente, è stato necessario creare un'ulteriore routine che “rimappasse” i parametri che vengono restituiti dall'`optim`, attraverso le stesse funzioni per cui sono stati “mappati”. Rimandiamo all'appendice A per la visione della funzione “`rimapnV2`”, che sfrutta la stessa convenzione vista sopra per l'assegnazione del nome.

La chiamata ad `optim` utilizzando il metodo “`L-BFGS-B`” è leggermente diversa da quelle viste sopra in quanto i vincoli del modello sono espressi direttamente nella chiamata con gli argomenti `upper`, per i limiti superiori del

vettore θ e *lower* per quelli inferiori. Presentiamo qui di seguito la chiamata:

```
optim(  
  par = init_norm1,  
  fn = mvdnV1,  
  method = "L-BFGS-B",  
  lower = c(-Inf, 0, 0, 0),  
  upper = c(Inf, Inf, 1, 1),  
  r = r_serie,  
  n = n_serie,  
  control = list(fnscale=-1))
```

2.4 Confronto tra le varie routine per la stima del vettore θ

In questa sezione presentiamo alcuni test effettuati sulle routine create. Questi test sono stati necessari per avere un'affidabilità nell'utilizzo successivo di queste routine. Qui si seguito, vi sono dei confronti fatti in termini di stabilità di stime, prima di tutto, e poi in termini di tempo per le routine che sopra abbiamo introdotto. Per queste analisi abbiamo utilizzato la soluzione più semplice per il modello: serie BAS con il modello GARCH(1,1) con assunzione di normalità per la componente d'errore. Oltre ad essere il modello con minor numero di parametri da stimare, in quanto non vi sono aggiunti parametri per la distribuzione, siamo sicuri che non vi possono essere diverse versioni per la distribuzione normale.

Nella Tabella 2.1 presentiamo le stime ottenute con i vari metodi di ottimizzazione, variando il modo in cui abbiamo inserito i vincoli. La simbologia utilizzata è quella sopra presentata. Per il metodo di ottimizzazione “*L-BFGS-B*” utilizziamo solo la possibilità di inserimento di vincoli con gli argomenti *upper* e *lower*. A fianco alle stime, abbiamo presentato un conteggio di tempo in secondi, ottenuto grazie alla funzione *system.time* presente nell'ambiente R e nella libreria *base*: questo numero serve per farsi un'idea sui tempi di attesa una volta che viene fatta la chiamata.

2.4 Confronto tra le varie routine per la stima del vettore θ

	V1_L_BFGS_B	V2_NM	V2_BFGS	V3_BFGS	V3_NM
mu	0.1053	0.1053	0.1053	0.1053	0.1053
omega	0.0471	0.0471	0.0471	0.0471	0.0471
alpha1	0.0951	0.0950	0.0949	0.0951	0.0949
beta1	0.8878	0.8879	0.8880	0.8879	0.8880
logVer	-3407.10	-3407.10	-3407.10	-3407.10	-3407.10
elapsed	5.89	4.10	4.28	2.92	5.50

Tabella 2.1: Stima dei parametri e tempi di calcolo per diversi metodi di ottimizzazione.

Come si può notare dalle stime ottenute, non vi sono particolari differenze tra i vari metodi in termini di stabilità: i valori sono pressoché uguali. Quello che invece differiscono in modo rilevante, a nostro giudizio, sono i tempi di calcolo. Si passa dai circa 3 secondi del metodo “BFGS” con i vincoli “IF...ELSE” ai 4, poco più del metodo “NM” con vincoli di rimappatura e ancora del “BFGS” con i vincoli “IF...ELSE”. Quello che balza all'occhio fin da subito è che conveniente scegliere il metodo di ottimizzazione BFGS.

Un ulteriore confronto vista la poca differenza tra i tempi ottenuti con i vincoli di rimappatura si può vedere nelle Tabelle 2.2 e 2.3, dove allarghiamo il numero delle stime dei parametri nel senso che richiamiamo *optim* sempre con un numero minore di osservazioni: n sta ad indicare il numero di osservazioni. Anche qui si può avere conferma della stabilità delle stime.

Capitolo 2. Le funzioni per il calcolo del modello GARCH

n	1872	1700	1500	1300	1200	1100	1050	1000
mu	0.107	0.104	0.107	0.100	0.094	0.093	0.105	0.096
omega	0.042	0.045	0.045	0.050	0.048	0.052	0.056	0.049
alpha1	0.102	0.089	0.089	0.094	0.098	0.096	0.099	0.098
beta1	0.888	0.893	0.892	0.886	0.886	0.885	0.882	0.888
logVer	-3413.234	-3032.330	-2691.158	-2405.385	-2249.272	-2092.442	-2021.371	-1934.244

Tabella 2.2: Stime dei parametri ottenute con *Nelder-Mead* e vincoli di rimappatura.

n	1872	1700	1500	1300	1200	1100	1050	1000
mu	0.106	0.104	0.107	0.100	0.094	0.093	0.105	0.096
omega	0.041	0.045	0.045	0.050	0.049	0.052	0.056	0.049
alpha1	0.102	0.089	0.089	0.094	0.098	0.096	0.099	0.098
beta1	0.888	0.893	0.892	0.886	0.886	0.885	0.882	0.888
logVer	-3413.234	-3032.330	-2691.158	-2405.385	-2249.272	-2092.442	-2021.371	-1934.244

Tabella 2.3: Stime dei parametri ottenute con *quasi-Newton* e vincoli di rimappatura.

Nella Tabella 2.4, presentiamo un confronto tra le librerie sopra citate, in modo da poter fare un confronto di stima non solo tra di esse, ma anche con quelle ottenute con le routine da noi costruite. Non abbiamo presentato una variabile che conteggia il tempo, come in precedenza, in quanto qui la velocità di calcolo è veramente irrisoria. Nel momento in cui si fa la chiamata, il risultato è già stampato nello schermo: stiamo quindi parlando di millesimi di secondo. Infatti l'implementazione di queste funzioni è stata realizzata non all'interno dell'ambiente R, ma utilizzando altri linguaggi di programmazione più specifici.

2.4 Confronto tra le varie routine per la stima del vettore θ

	fGarch		rgarch		S-PLUS	
	Stima	Str. Err.	Stima	Str. Err.	Stima	Str. Err.
mu	0.1068	0.0298	0.1052	0.0298	0.1082	0.0299
omega	0.0468	0.0120	0.0478	0.0123	0.0449	0.0089
alpha1	0.0952	0.0132	0.0974	0.0135	0.0994	0.0097
beta1	0.8881	0.0145	0.8860	0.0148	0.8859	0.0105
logVer	-3413.06		-3408.53		-3413.16	

Tabella 2.4: Confronto nella stima dei parametri tra le librerie preesistenti.

Come si può verificare che le stime ottenute con le routine che abbiamo creato, visibili nella Tabella 2.1, sono stabili se confrontate con queste. Questo è molto importante per noi, in quanto sta ad indicare che le routine sono per lo più affidabili.

Sono state fatte anche delle analisi su degli argomenti presenti su *optim* quali *maxit* e *abstol*. Il parametro *maxit* limita il numero di iterazioni che la routine di ottimizzazione può compiere per arrivare al punto di massimo, mentre *abstol* controlla la tolleranza numerica durante la convergenza al punto di massimo. Quello che è emerso è che si possono tranquillamente conservare i valori di default per questi parametri perché per i nostri calcoli non vi sono differenze.

Da queste verifiche abbiamo pensato di utilizzare la funzione *optim* con il metodo di ottimizzazione “*BFGS*” e con l’adempimento dei vincoli con la versione V3 perché più veloce in termini di performance e come si vedrà in seguito preferibile nel momento che sarà richiesta anche lo *standard error* della stima.

Capitolo 3.

Analisi della stabilità delle stime

In questo capitolo presentiamo la ricerca di qualche regolarità dalle stime ottenute sui parametri del modello GARCH. Attraverso l'utilizzo delle routine appositamente create nella sezione 2.2 e delle serie dei titoli finanziari, stimiamo i parametri del modello ottenuti considerando man mano la serie dello stesso titolo con un minor numero di osservazioni rispetto alla lunghezza effettiva. Ad esempio presa una serie di un titolo con lunghezza effettiva di 2000 osservazioni, noi vediamo di confrontare le stime dei parametri ottenute con 1900 osservazioni, con 1800, con 1700, con 1600, con 1500 e così via fino a circa 500 osservazioni e andiamo a confrontare queste stime.

Definiamo “parte di serie” del titolo come il fatto di prendere un numero minore di osservazioni rispetto alla lunghezza effettiva. Tali parti sono definite prendendo osservazioni contigue ed eliminando man mano le osservazioni più vecchie del titolo. Se, dal confronto di queste stime delle parti di serie, ci si accorge che, per una porzione di serie, le stime possono essere considerate stabili, noi definiamo “finestra di lunghezza n_0 ” (ovviamente $n_0 < n$) la porzione di serie contenente n_0 osservazioni. In questa prima analisi per “stime stabili” intendiamo, stime che ad una prima visione sembrano non discostarsi molto da quelle di riferimento.

Una volta definita la lunghezza della finestra, all'interno della serie completa del titolo facciamo muovere la nostra finestra, in modo da vedere come le stime cambiano. Ad esempio presa una serie di un titolo con lunghezza effettiva di 2000 osservazioni e con una finestra di n_0 uguale a 1000, noi andiamo a confrontare le stime dei parametri ottenute dalla finestra contenente le osservazioni dalla 2000^a alla 1000^a, poi con la finestra dalla 1900^a alla 900^a, da 1800^a alla 800^a, dalla 1700^a alla 700^a, dalla 1600^a alla 600^a, e così via fino dalla 1000^a all'ultima.

Da questo lavoro sono emerse delle importanti considerazioni sui titoli da noi

presi in esame, ma anche sulla distribuzione che prendiamo come componente d'errore dei modelli che andiamo a considerare.

Nella sezione 3.1 presentiamo alcuni quesiti che ci siamo posti cercando di far cogliere al lettore il nostro punto di partenza nella ricerca della stabilità delle stime. A questi quesiti abbiamo cercato di dare delle risposte con delle successive implementazioni ai termini “parte di serie” e “finestra di lunghezza n_0 ” date sopra, aiutandoci anche con delle tabelle e dei grafici che saranno riportati nell'appendice B. Dall'osservazione di questi grafici sono emerse importanti considerazioni in risposta ai nostri quesiti.

A causa di quanto emerso nella sezione 3.1, è stato necessario creare delle routine per la generazione di serie pseudocasuali, che sono presentate nella sezione 3.2 in modo da poter confrontare i risultati ottenuti con le serie reali dei titoli azionari e quelli con le serie simulate. La sezione 3.2.1 è proprio dedicata al confronto tra le caratteristiche delle serie reali con le serie simulate.

Nella sezione 3.3 presentiamo un ulteriore confronto tra le stime ottenute dalle serie simulate, in un caso con le routine da noi create e nell'altro sfruttando la funzione *garchSim* del pacchetto *fGarch*. Infine nella sezione 3.4 ci poniamo un ulteriore quesito vista l'impossibilità di trovare un n_0 valido nell'intervallo da 1000 a 1500 in cui ci eravamo ristretti. Abbiamo pensato di concentrare la nostra attenzione su n_0 prossimo ad n , in modo da vedere come le stime si avvicinano a quelle di riferimento.

3.1 La ricerca della stabilità delle stime dei parametri

Nella sezione 1.4 abbiamo presentato i titoli azionari oggetto della nostra analisi, che come avevamo anticipato hanno diverse lunghezze.

Come punto di partenza del nostro lavoro, ci siamo posti una serie di quesiti sul comportamento delle stime dei parametri del modello per la valutazione della loro stabilità in relazione anche ad una finestra che si sposta all'interno della serie completa del titolo.

La prima fra tutte è: “È possibile confrontare le stime dei parametri ottenute sulla serie completa con quelle ottenute solamente su una parte di essa?”. L'idea

3.1 La ricerca della stabilità delle stime dei parametri

è quella di ricercare un valore n_0 per la lunghezza della finestra, cioè un numero di osservazioni sufficientemente minore della lunghezza della serie ($n_0 < n$), in modo da poter fare analisi su questa parte di serie ristretta. L'esistenza di un tale valore comporterebbe diversi vantaggi, tra cui un minor tempo necessario alla stima dei parametri, inoltre la possibilità di ricavare informazioni sul titolo azionario a partire da una parte della serie ci indicherebbe la non necessità di avere ampia conoscenza del passato, visto che come abbiamo anticipato togliamo man mano le osservazioni meno recenti.

La seconda, *“È possibile fare una valutazione di stabilità delle stime rispetto ad una “finestra” che si muove all'interno della serie completa del titolo?”*. Data precedentemente la definizione per ciò che noi intendiamo per finestra, una volta stabilito questo valore per la lunghezza della finestra, ci chiediamo come cambiano le stime dei parametri ottenuti da questi frammenti, muovendosi all'interno della stessa serie con la finestra e prendendo man mano parti di serie diverse.

La terza, date le varie stime ottenute con la finestra di lunghezza n_0 che si sposta: *“È possibile individuare degli allontanamenti (“sforamenti”) dalle stime della serie completa presa come riferimento oltre un limite ragionevole di tolleranza?”*.

Il fatto di poter definire una finestra di lunghezza prefissata è ragionevole in quanto consentirebbe di proporre considerazioni generali sull'andamento del titolo azionario. Si potrebbero calcolare dei coefficienti di riepilogo (come il VaR) o fare una previsione su come il titolo possa comportarsi in un futuro vicino o lontano, se possa ad esempio avere delle perdite piuttosto che dei guadagni.

Inoltre, sottolineiamo il fatto che stiamo parlando di titoli finanziari: lo studio dell'andamento dei rendimenti di questi titoli a scopi di lucro è, da molti anni, al centro della ricerca di molti studiosi del mondo della finanza, nell'intento di costruire modelli sufficientemente affidabili che si adattano sempre meglio ai dati.

Come si vedrà, sebbene la ricerca di questo n_0 non porta completamente ai risultati sperati, siamo comunque stati in grado di produrre importanti osservazioni sulle nostre analisi che possono essere argomento di discussione.

È importante sottolineare fin da subito, che qualunque confronto da noi

proposto sulle stime, deve essere inteso con riferimento ai risultati ottenuti con la serie completa: nei grafici che in seguito presenteremo questo riferimento è evidenziato con una linea rossa tratteggiata.

Premettiamo che le osservazioni della serie sono disposte in ordine temporale decrescente, quindi la più vecchia è la prima, mentre l'ultima è la più recente e dal nostro punto di vista la più interessante. Per tale ragione, noi lavoreremo con queste serie con scostamenti decrescenti.

Per rispondere al primo quesito, abbiamo stimato i parametri della serie con parti di serie di lunghezza decrescente che comprendono man mano un numero minore di osservazioni. Ad esempio, con riferimento ad una serie di lunghezza $n = 2000$, abbiamo calcolato i parametri del modello sfruttando le osservazioni dalla 2000^a alla 100^a , dalla 2000^a alla 200^a , dalla 2000^a alla 300^a e così via... e poi abbiamo messo a confronto tutte le stime ottenute.

Un comportamento per noi auspicabile avrebbe previsto un andamento molto vicino al riferimento per lunghezze delle parti di serie vicini ad n , per esibire poi un allontanamento sensibile in relazione alla riduzione dei campioni utilizzati, in modo da poter esibire come n_0 le stime di lunghezza minima da noi considerate stabili o comunque non tanto lontane dal riferimento.

Tale comportamento non si è purtroppo presentato, prima di tutto per il fatto che, analizzando diversi titoli, si è potuto notare che non era possibile definire un singolo n_0 soddisfacente per le diverse casistiche. Inoltre, in determinate situazioni si è potuto assistere ad un peggioramento immediato delle stime a cui è seguito un ritorno a valori molto vicini al riferimento. Nell'appendice B, si può vedere nelle Tabelle dalla B.1 alla B.5 le particolarità appena evidenziate.

Le tabelle prendono in esame tutte le distribuzioni per la componente d'errore, rispettivamente la normale, la normale asimmetrica, la t di Student, la t di Student asimmetria e la distribuzione di Subbotin. Soffermandoci nel caso normale per dare una delucidazione, la prima riga identificata con "n" specifica la lunghezza della parte di serie con la quale si ottengono le stime che, procedendo verso destra, decresce. Dalla seconda riga alla quinta abbiamo i diversi parametri. Ed infine nella sesta riga identificata con "logVer", evidenziata in giallo, i valori della log-verosimiglianza nel corrispondente punto di massimo per il vettore θ . Le

colonne evidenziate in colori diversi rappresentano le stime di riferimento (verde) e i valori degli *standard error* delle corrispondenti stime (giallo).

Anche se non è stato possibile individuare l'auspicato n_0 univoco per tutte le serie, abbiamo comunque avuto la possibilità di restringere il campo dei valori ammissibili.

Una delle considerazioni che sono emerse è la possibilità di utilizzare finestre parziali con n_0 molto vicino alla metà della lunghezza della serie completa, che presenta comunque aspetti positivi per i vantaggi precedentemente esposti.

Quindi abbiamo proceduto nella nostra analisi considerando diversi n_0 nell'intervallo da 1000 a 1500 visto che la maggior parte delle nostre serie hanno una lunghezza attorno alle 2000 osservazioni.

Per rispondere quindi al secondo quesito, abbiamo considerato varie casistiche. La nostra attenzione non è solamente rivolta allo spostamento della finestra ma anche alla variazione dell' n_0 . Affinché fosse possibile un confronto tra porzioni della stessa serie, lo spostamento della finestra all'interno della serie dipende dalla lunghezza della serie completa, in quanto serie più lunghe prevedono spostamenti maggiori. Ad esempio per il titolo BMW che ha lunghezza 1904, lo spostamento della finestra è di 75 osservazioni alla volta, mentre per il titolo SDF che lunghezza 2688, lo spostamento della finestra è di 200 osservazioni.

È possibile vedere nella Figura B.1 come abbiamo raccolto le nostre stime. Mentre per rispondere al primo quesito abbiamo presentato direttamente i parametri numerici in forma tabulare, in questa occasione abbiamo pensato di introdurre anche la componente grafica per avere un'immediata visione della variabilità.

A riguardo è importante fare una breve spiegazione di come sono state realizzate le figure riportate nell'appendice B. Prima di tutto per figura noi intendiamo una serie di grafici in cui ogni riga corrisponde ad un diverso parametro e ogni colonna un diverso n_0 per la finestra. Tutto questo è stato necessario per riassumere contemporaneamente tutte le casistiche in un'unica figura.

Ogni grafico ha poi per asse delle ordinate il campo di variabilità della stima

del parametro e per asse delle ascisse il numero che rappresenta la prima osservazione della finestra. Per cui ad esempio se il parametro μ è stato stimato con 0.45, l'asse delle ordinate coprirà un intervallo tra 0 e 0.8. Per l'asse delle ascisse avremo, come primo valore, la lunghezza effettiva della serie, come secondo la lunghezza effettiva della serie completa a cui è tolto il valore dello spostamento, il terzo sarà pari al secondo valore meno il valore dello spostamento, fino ad arrivare a far slittare la finestra per 5 volte.

Dall'analisi grafica che presentiamo, oltre a non essere possibile trovare regolarità per l'individuazione di n_0 , è difficile anche poter fare qualche considerazione sugli sforamenti delle stime come richiesto dal terzo quesito. Per una definizione più rigorosa di "stime stabili", abbiamo pensato all'introduzione di un intervallo di accettazione delle stime: proponiamo un metodo per determinare un campo ammissibile correlato alle stime di riferimento.

Come avevamo precedentemente accennato nella sezione 2.3 quando facciamo partire la chiamata ad *optim*, poniamo l'argomento *hessian = TRUE*, in modo che ci venga restituita la matrice hessiana calcolata con metodi numerici. Infatti sotto opportune condizioni di regolarità, la stima di massima verosimiglianza è asintoticamente distribuita come una normale,

$$\hat{\theta}_n \sim N(\theta, n^{-1} V_{as}) \text{ se } n \rightarrow \infty, \quad (3.1)$$

dove la varianza asintotica (V_{as}) è uguale all'inversa della matrice hessiana (J):

$$V_{as} = J^{-1} \quad (3.2)$$

Dalla matrice hessiana otteniamo quindi la varianza asintotica della stima di massima verosimiglianza, da cui quindi successivamente gli *standard error* (*str.err.*) delle stime dei parametri per poter calcolare con questi degli intervalli di confidenza al livello di confidenza del 95%. L'intervallo considerato ha questi limiti:

$$[\hat{\theta}_n - 2 \text{str. err.}, \hat{\theta}_n + 2 \text{str. err.}] \quad (3.3)$$

Questi intervalli, uno per ogni parametro, ci permettono di verificare se la variazione della stime è contenuta nell'intervallo ammissibile. Affinché l'intervallo di accettazione delle stime possa essere di immediata visione nei grafici relativi a questa sezione, che presenteremo nell'appendice B, lo abbiamo evidenziato con delle linee oblique di color grigio.

Anche qui, sarebbe stato auspicabile osservare che le stime, dopo un certo n_0 , fossero tutte comprese all'interno dell'area grigia per tutti i titoli presi in considerazione e per tutte le distribuzioni analizzate come componente d'errore, e magari questo n_0 fosse valido per tutti.

Quello che è emerso invece è qualcosa di diverso. Ricordiamo che le nostre analisi prendono in considerazione simultaneamente nella modellazione con componente d'errore le cinque distribuzioni presentate nella sezione 2: per cui noi stampiamo grafici che considerano separatamente i parametri che sono stati ottenuti con modelli con diversa componente d'errore.

Vi sono delle caratteristiche comuni per tutte le distribuzioni. Per primo, vogliamo far notare che vi sono casi, anche se rari, in cui è possibile trovare un n_0 che è valido per tutti i parametri e che quindi permetterebbe di prendere un intervallo parziale per spiegare la serie completa come si può vedere nella Figura B.2 dove sarebbe possibile scegliere una finestra di 1200 osservazioni.

In secondo luogo, si può osservare un andamento simmetrico tra i parametri α_i e β_i : ad una decrescita di uno, vi è corrispondentemente una crescita dell'altro. Questo è possibile notarlo su tutte le figure che di seguito verranno riportate: è ben evidente nella Figura B.3.

Inoltre, capita a volte di osservare un “picco”, ossia una stima anomala di un parametro, nel senso che vi è un gran allontanamento dalla stima subito precedente per poi tornare a un comportamento più smorzato nel passaggio a quella successiva. La cosa strana è che questo comportamento sembra essere subito assorbito se si passa a considerare la colonna successiva e cioè analizzando una finestra con un n_0 diverso. Nella Figura B.4 si può notare subito il picco di cui stiamo parlando nella colonna della “Finestra 1200”, come si può notare il picco scompare immediatamente nella colonna successiva: sembra che nell'arco delle 100 osservazioni successive questo piccolo sconvolgimento sia stato assorbito.

Sottolineano il fatto che 100 osservazioni coprono un arco temporale di circa tre mesi: potrebbe essere abbastanza plausibile che una notizia abbia portato dei cambiamenti immediati, e che poi il titolo sia tornato alla normalità.

Infine, si nota che, se nei nostri auspici avremmo voluto avere delle stime molto vicine ai valori di riferimento, tuttavia questo non accade. Come si può vedere nella Figura B.5 nella riga dedicata al parametro α_I , vi è piuttosto un considerevole allontanamento: addirittura le stime non cadono mai all'interno dell'area ammissibile.

A queste considerazioni generali, si affiancano alcune proprie delle distribuzioni. Per la distribuzione normale, è assai raro che il parametro μ non rientri nel campo di ammissibilità.

Per le distribuzioni asimmetriche, è possibile notare un'ulteriore simmetria nell'andamento dei parametri: questa volta ad essere interessati sono i parametri μ e α , come si può vedere nelle Figure B.5 e B.6. Anche qui ad una decrescita graduale di α si assiste ad una crescita graduale di μ . Sempre con riferimento a queste distribuzioni, si nota che il valore del parametro α , che ricordiamo indicare l'asimmetria, è spesso stimato in un intorno di 0: questo conferma quanto visto nell'analisi preliminare del capitolo 1 e cioè che l'introduzione dell'ipotesi di asimmetria non sembra migliorare l'adattamento del modello ai dati.

Una considerazione a parte va fatta anche per il parametro di forma v , nelle distribuzioni in cui è presente: in alcuni casi è possibile trovare un n_0 come si può vedere in Figura B.3 e in altri assolutamente no (vedi Figura B.7).

Una particolare nota va fatta in quanto vi sono stati alcuni casi in cui non è stato possibile calcolare l'intervallo di confidenza di alcuni parametri in quanto sono stati restituiti dei NA¹ nel calcolo degli *standard error*.

3.2 Creazioni routine che simulano serie dei modelli

GARCH

Dalle osservazioni fatte nel paragrafo precedente sembra emergere che non sia possibile individuare un n_0 nemmeno con l'aiuto dell'intervallo di confidenza.

¹ NA (“Not Available” o “Missing Value”): questa sigla è indicatrice di valore mancante.

Fino a questo momento noi abbiamo applicato le funzioni per la stima di modelli GARCH solamente ai dati reali. Un ulteriore quesito che ci siamo posti è il seguente: “È possibile che queste “irregolarità” riscontrate siano dovute al comportamento dei dati reali, che come sappiamo possono presentare delle particolarità?”. Per rispondere a questo quesito abbiamo quindi pensato di rifare i nostri test con serie simulate, in modo che, essendo le stime dei parametri decise a priori, non vi sia la possibilità di credere che non siano buone.

Abbiamo creato delle routine che, partendo dalle stime di riferimento, generassero delle serie della stessa lunghezza della serie reale. Anche qui come spunto abbiamo utilizzato gli appunti del corso di Serie Storiche Finanziarie.

Presentiamo come esempio la routine per la distribuzione normale. Visto che per i calcoli della funzione di log-verosimiglianza abbiamo utilizzato la specifica di vincoli “IF...ELSE”, abbiamo seguito solo questa strada per la creazione di queste routine. Non ci soffermiamo sulla simbologia utilizzata perché è la stessa che si trova nella sezione 2.2.

A differenza delle routine per il calcolo della log-verosimiglianza qui non siamo in presenza di un vettore dei rendimenti da passare come argomento, ma al contrario esso vien man mano generato con la funzione *rnorm*ⁱⁱ. La chiamata per questa funzione, infatti richiede come argomenti:

- *theta* che sta per il vettore dei parametri da cui simulare la serie;
- *n* che sta ad indicare la lunghezza desiderabile per la serie simulata;
- *n_start*, di default impostato a 100, che sta ad indicare che è necessario di un breve periodo di *burn-in*ⁱⁱⁱ prima di considerare buone le osservazioni simulate.

Prima della chiamata alla funzione è necessario fare una chiamata alla funzione *set.seed* in quanto, passando un valore numerico a questa funzione, si permette all'utente di utilizzare un seme fissato per il generatore di numeri

ⁱⁱ La funzione *rnorm*(*n*, *mean*, *sd*) genera un campione di numerosità *n* da una variabile casuale normale di media *mean* e deviazione standard *sd*.

ⁱⁱⁱ *Burn-in* (qualcosa di simile al rodaggio) è il processo al quale sono sottoposti i componenti di un sistema prima di essere messi in servizio. Nel nostro caso, sono le osservazioni che sono necessarie eliminare prima di considerare buone le osservazioni simulate.

pseudocasuali in modo risalire alla stessa serie simulata. In caso di necessità di ricalcolo della serie simulata dei rendimenti, tale accorgimento ci permette di rigenerarla uguale.

Il nome della funzione è stato scelto con la stessa struttura delle funzioni per la log-verosimiglianza a cui è stato aggiunto all'inizio “*sim_*” che sta ad indicare che ci stiamo riferendo a funzioni che simulano. Per intero il nome della funzione per la distribuzione normale è “*sim_mvdnV3*”. Il codice relativo a questa routine assieme a quello per le altre distribuzioni si trova nell'appendice A.4.

3.2.1 Ripetizione dei test sulle serie simulate

Ci siamo posti gli stessi quesiti della sezione 3.1 per le serie simulate con le routine che precedentemente abbiamo illustrato. Non siamo ripartiti dal primo quesito ma abbiamo cercato di analizzare subito se l'idea dell'intervallo di confidenza potesse essere valida. Presentiamo in appendice B i grafici relativi a questa sezione facendo notare che la linea rossa tratteggiata sta ancora ad indicare le stime di riferimento della serie completa simulata, e che la linea blu tratteggiata e puntata sta ad indicare il valore relativo al parametro con cui noi abbiamo simulato.

Ci saremmo aspettati, innanzitutto, che queste due righe fossero pressoché sovrapposte in quanto se noi generiamo con un certo valore per il parametro e poi ci calcoliamo la stima su dati generati con tale parametro, ci aspetteremmo che esso non cambiasse significativamente. Questo ci lascia un po' perplessi e come vedremo ci porterà ad altre analisi con l'intento di verificare che le routine di generazione di serie siano valide in modo da escludere qualunque dubbio. Vi sono comunque alcuni casi in cui le due linee coincidono come si può verificare in Figura B.10.

Ripercorriamo le caratteristiche viste nella sezione 3.1 per le serie reali, cercando di ritrovarle anche nelle nostre simulazioni.

Per primo, anche nelle serie simulate, è possibile trovare per alcuni casi un n_0 valido per tutti i parametri. Osserviamo però come questo non si è verificato per la stessa serie (BMW) ma per esempio per la DBK (vedi Figura B.9).

In secondo luogo l'andamento simmetrico per i parametri α_1 e β_1 si verifica anche per le serie simulate come è possibile vedere nelle Figure dalla B.8 alla B.10.

Inoltre, anche qui capita a volte di osservare un “picco”: tuttavia, mentre prima sembrava essere assorbito alla finestra subito successiva, questo sembra protrarsi almeno per un'altra finestra prima di scomparire. Nella Figura B.8 alla riga corrispondente al parametro β si può notare che il picco relativo alla colonna della “Finestra 1100”, si protrae nella colonna successiva e man mano, fino a svanire con l'aumento del n_0 .

Infine, si nota che nelle serie simulate le stime sono abbastanza lontane dai valori di riferimento. Come si può vedere nella Figura B.8 nella riga dedicata al parametro α_1 , vi è piuttosto un considerevole allontanamento.

In questa figura si può anche notare che non è evidenziato l'intervallo di ammissibilità per il parametro ω : abbiamo potuto analizzare che questo fatto è dovuto al piccolissimo valore dello *standard error* che viene approssimato a 0.

Andiamo ora a ripercorrere le considerazioni proprie delle singole distribuzioni svolte nel caso di serie reali: confermiamo la permanenza del parametro μ all'interno dell'intervallo di ammissibilità, la simmetria relativa ai parametri μ e α , la non rilevanza dell'introduzione dell'ipotesi di asimmetria nel modello.

Ricordiamo che la prassi nelle simulazioni prevede la generazione di un numero significativo di campioni su cui trarre osservazioni: nel nostro caso abbiamo invece generato un unico campione per ogni titolo azionario.

Questa scelta potrebbe limitare l'ampiezza delle nostre considerazioni ma ci consente comunque di concludere come i fenomeni osservati in serie reali si ripresentino nelle serie simulate e sia quindi poco probabile che tali effetti siano dovuti a comportamenti peculiari del titolo stesso.

L'idea dell'impossibilità di determinare un intervallo di ammissibilità dei parametri sembra quindi confermata.

3.3 Confronto con le routine preesistenti

Come anticipato, per verificare che le routine da noi create siano valide, abbiamo confrontato le stime ottenute dalle serie simulate con le routine da noi create, con le stime ottenute dalle serie simulate con la funzione *garchSim* del pacchetto *fGarch*. Tale soluzione presenta delle limitazioni in quanto il confronto effettivo si può fare solamente con la distribuzione normale e con la *t* di Student che siamo sicuri avere la stessa parametrizzazione.

Le funzioni del pacchetto *fGarch* necessarie per queste simulazioni sono *garchSpec* e *garchSim*. Con la prima funzione, specifichiamo i parametri del modello con i quali vogliamo simulare e, come fatto in precedenza, poniamo anche l'argomento "*rseed = 1*", in modo da poter risalire in caso di necessità di ricalcolo della serie simulata dei rendimenti. Allo stesso tempo, con la seconda funzione diamo via alla simulazione, anche qui abbiamo la necessità di specificare l'argomento "*n.start = 100*" per permettere il *burn-in*.

Sarebbe stato auspicabile che venissero generate le stesse serie dei rendimenti, ma abbiamo ragione di credere che qualche altro parametro incluso nelle librerie preesistenti fa in modo che vi sia qualche piccola diversità. Stiamo comunque generando numeri casuali e quindi ciò che, alla fine, a noi interessa è il confronto del comportamento delle stime dei parametri del modello ottenute con le serie simulate, non tanto che siano generate delle stesse serie con valori identici dai due simulatori.

La nostra verifica si sposta quindi all'analisi delle stime ottenute dalle serie simulate con le funzioni della libreria *fGarch*. Nel caso specifico ci interessa verificare se si presentano situazioni diverse da quelle delle serie simulate con le nostre routine.

Rimandiamo alle figure B.11 e B.12 in appendice B per questo confronto. Da queste figure non emergono comportamenti rilevanti, ma non sono state nemmeno smentite le osservazioni relative a peculiarità delle stime che erano emerse nell'analisi delle serie reali. Le quattro osservazioni evidenziate per le serie simulate con le nostre routine sono riproposte e ampiamente visibili anche in questo particolare caso.

Possiamo però affermare che è stato più difficile trovare casi in cui non era

possibile definire un n_0 rispetto ai casi reali. Neanche qui quindi è stato possibile definire un n_0 comune per tutti i titoli azionari tra il campo di ammissibilità in cui ci eravamo ristretti.

Il problema degli *str. err.* prossimi a 0, si verifica anche qui, riteniamo sia legato probabilmente a come noi calcoliamo la matrice hessiana; ricordiamo che la matrice hessiana è calcolata con metodi numerici.

3.4 Analisi su valori di n_0 prossimi a n

Vista l'impossibilità di trovare un n_0 valido nell'intervallo da 1000 a 1500 in cui ci eravamo ristretti, abbiamo pensato di concentrare la nostra attenzione su n_0 prossimo ad n , in modo da vedere come le stime si avvicinano a quelle di riferimento. Questo rappresenta quindi un ulteriore quesito che ci siamo posti a seguito dei risultati ottenuti.

Per rispondere a questa questione ci siamo focalizzati sulla distribuzione normale, senza analizzare anche le altre distribuzioni per la componente d'errore del modello in quanto riteniamo, da quanto evidenziato fin'ora, di non ricadere in casi particolari.

Abbiamo preso in considerazione solamente dei titoli azionari con serie di lunghezza attorno alle 1800/1900 osservazioni e abbiamo fatto le stesse investigazioni fatte precedentemente, con degli n_0 più grandi. L'intervallo per n_0 ora va da 1000 a 1800 osservazioni, in modo da poter fare un parallelo ad ampio spettro. L'analisi grafica fatta a riguardo rispetto alla precedente non presenta cinque slittamenti della finestra ma solamente tre per evitare di sovrappollare inutilmente l'immagine con una mole eccessiva di dati.

Nella Figura B.13, focalizzando la nostra attenzione su un parametro (su una riga) alla volta e sulla prima stima di ogni grafico, procedendo verso destra, ci accorgiamo che questa prima stima man mano si avvicina al valore di riferimento. Questo fatto si verifica per ogni parametro ed è quello che noi auspicabilmente ci aspettavamo: man mano che le parti di serie si avvicinano alla lunghezza della serie complessiva che nel caso in esame è 1872.

Quello che balza però all'occhio sembra contraddire quanto analizzato e notato

fino ad ora. Infatti se noi fino ad ora ci siamo chiesti che valore deve avere la finestra che si sposta all'interno della serie e non siamo riusciti a dare questo valore, *perché qui riusciremmo a darlo anche per valori bassi di n_0* ? Si può notare che sarebbe possibile scegliere un n_0 attorno al 1200 per il titolo BAS nel caso normale. Riteniamo che fare una considerazione di questo tipo sarebbe troppo affrettato, in quanto dal secondo quesito in poi noi ci siamo messi alla ricerca di un n_0 per una finestra che si muove all'interno della serie completa. Infatti il problema non è solo il valore di n_0 per considerare stabili le stime ma anche legato ad una finestra che si muove all'interno della serie completa.

Quello che emerge è che poche osservazioni, e per poche intendiamo 75/100 osservazioni, possono fare la differenza nella caratterizzazione di una serie di rendimenti finanziari. Infatti quando nelle analisi della sezione 3.1 slittavamo di queste quantità per la finestra, notavamo una netta differenza nelle stime. Sembra quindi che l'impossibilità di trovare un valore di n_0 per finestra sia legata all'importanza delle ultime osservazioni.

È possibile verificare questa osservazione anche nella Figura B.14 dove ad essere sotto esame è la serie del titolo DAI, il quale ha una lunghezza di 1870 osservazioni (molto simile alla lunghezza della serie BAS). Mentre per ogni parametro nel titolo precedente era possibile notare un avvicinamento alle stime di riferimento man mano che n_0 cresceva: questo non accade per tale titolo, sintomo che le ultime 70 osservazioni necessarie per raggiungere la lunghezza completa della serie sono fondamentali per avvicinarsi alle stime di riferimento.

Facciamo comunque notare che la scala utilizzata per l'asse delle ordinate è molto piccola: infatti tutti i valori sono all'interno dell'intervallo di confidenza (area a righe oblique grigie) e quindi si potrebbero considerare buoni.

Vogliamo portare d'esempio anche la Figura B.15 per il titolo SDF che ha una serie di lunghezza 2688 per confermare quanto appena osservato sull'importanza dell' n_0 in quanto un valore di 1800, molto lontano da n , non permette un confronto diretto.

Nella Figura B.16, per meglio far capire al lettore la direzione delle nostre considerazioni, abbiamo raccolto le stesse stime della Figura B.13 in maniera diversa: separatamente le prime stime per ogni valore di n_0 della finestra, poi

successivamente le seconde e le terze. Facciamo notare che lo slittamento della finestra è molto ridotto (20 osservazioni alla volta), quindi noi consideriamo come prima stima quella ottenuta con slittamento nullo, come seconda quella ottenuta con slittamento pari a 20, infine come terza quella ottenuta con uno slittamento pari a 40 dalla prima osservazione.

Nell'asse delle ascisse le etichette rappresentano i valori di n_0 , crescenti da 1000 alla lunghezza della serie. La prima colonna mostra quello che noi avremmo auspicato fino da l'inizio, cioè un andamento delle stime che pian piano si avvicina a quelle di riferimento. La seconda e la terza, invece, ci mostra che anche senza uscire dalle bande dell'intervallo, le stime si allontanano da quelle di riferimento. Questo potrebbe essere sintomo che le ultime osservazioni (cioè le più recenti) sono più importanti nelle stime del modello per catturarne le caratteristiche della serie rispetto a quelle meno recenti.

Nella Figura B.17 è verificato a grandi linee quanto detto relativamente alla prima colonna della figura precedente, ma è mostrato anche che come avevamo fatto notare sopra vi sono dei piccoli assestamenti delle stime senza che queste comunque escano dall'intervallo già con $n_0 = 1200$.

Riteniamo importante presentare anche la Figura B.18, la quale mostra che la serie SDF con lunghezza 2688, ha gli sbalzi al di fuori dell'intervallo perché qui gli n_0 proposti sono più lontani da n .

Conclusioni

L'obiettivo di questo elaborato è di studiare la stabilità delle stime dei parametri di un modello GARCH(1,1) con varie distribuzioni per componente d'errore per le serie dei rendimenti finanziari. Prima però di riassumere quanto evidenziato dalle analisi, riteniamo importante fare due considerazioni che un po' si allontanano da questo obiettivo, ma che riteniamo comunque salienti a conferma di studi già esistenti od eventualmente come spunto per successivi.

Nel capitolo 1, e più precisamente nell'analisi preliminare, è stato importante osservare alcune evidenze empiriche dei rendimenti per avere un'idea sui parametri di asimmetria e forma delle distribuzioni che in seguito avremmo analizzato. Per i parametri caratterizzanti solamente il modello, l'idea del loro campo di variabilità era già stata chiarito quando siamo andati ad analizzare i vincoli caratterizzanti il modello stesso. Per esempio il fatto che il valore del parametro α_1 si aggirasse attorno al 0.1 - 0.3 e β_1 attorno al 0.7 - 0.9. Invece per quanto riguarda i parametri propri della distribuzione, le investigazioni fatte non hanno fatto altro che confermare l'evidenza empirica, secondo cui i rendimenti posseggono delle code pesanti.

Prima di introdurci nel fulcro del nostro lavoro, è stato necessario costruire varie routine con l'intento di utilizzare la parametrizzazione della t di Student e della normale asimmetrica del prof. Azzalini. Per crearle, sono state necessarie ricerche per quel che riguarda i metodi di ottimizzazione e come meglio adattarli ai vincoli parametrici dei modelli. Al fine di migliorare la precisione e la velocità del calcolo, abbiamo indagato diverse varianti sia nel metodo di ottimizzazione sia nella specifica della funzione obiettivo, che hanno dimostrato la necessità di scegliere la versione migliore per avere un comportamento ottimo. Nell'ambiente R, vogliamo quindi consigliare l'utilizzo del metodo di ottimizzazione "BFGS" con un adempimento di eventuali vincoli attraverso le condizioni "IF...ELSE"

Passando infine al vero e proprio obiettivo della nostra tesi riteniamo, alla fine delle nostre analisi, che la ricerca della finestra, per come noi l'abbiamo definita,

Conclusioni

non abbia portato ai risultati sperati, ma ci comunque abbia permesso di arrivare ad considerazioni molto importanti.

In primo luogo, la presenza di parametri con un comportamento simmetrico, che definiremmo meglio come inversamente correlati, in quanto abbiamo notato una relazione tra il parametro $\alpha_l - \beta_l$ e $\mu - \alpha$ (α dove presente). Non abbiamo indagato con precisione a che cosa sia dovuto tale comportamento, ma ipotizziamo che il primo sia dovuto al fatto che un rendimento possa essere più influenzato o da quello precedente (α_l da il “peso” al rendimento precedente) o dall'innovazione (β_l da il “peso” al carattere innovativo). Ricordiamo inoltre che tali parametri sono collegati dal vincolo di stazionarietà del modello (si veda l'espressione 1.8). Per la seconda correlazione ipotizziamo una forte relazione tra la media e asimmetria in quanto esse vanno insieme a definire la forma della distribuzione.

In secondo luogo, non riteniamo utile fare un confronto tra le stime dei parametri di riferimento con la stima dei parametri delle parti di serie ottenute con la finestra che si muove. Il confronto con i parametri di riferimento è sensato solamente per individuare l' n_0 che ci permetta di considerare per le indagini solamente una parte della serie completa perché non appena la finestra slitta anche solamente di 75 osservazioni, vanno perse le ultime osservazioni, più precisamente le più recenti, e questo porta allontanamenti importanti dalle stime di riferimento. Riteniamo inoltre che già non considerare 75 osservazioni su una serie di circa 2000 osservazioni può risultare compromettente se si fanno analisi su questa parte di serie per darne un andamento generale del titolo.

L'idea della finestra che si sposta potrebbe essere valida, limitandosi a slittamenti molto piccoli, nell'ordine delle 20/40 osservazioni, in modo da rimanere comunque all'interno dell'intervallo di confidenza, come si è potuto notare nelle ultime figure del capitolo 3.

Nelle analisi abbiamo anche mostrato che andare a considerare n_0 pari a 1000 per serie con lunghezza superiore alle 2500 osservazioni porta a risultati non rilevanti, in quanto la differenza tra i due valori è troppo elevata. Per questo riteniamo sia stato fondamentale analizzare titoli con serie di rendimenti di diversa lunghezza.

Infine, non riterremo di dover cassare in toto l'idea dell'intervallo di confidenza ottenuto con gli *standard error*. È preferibile utilizzarlo solamente per rispondere al primo quesito, cioè per valutare la riduzione della serie completa. La ricerca di un n_0 per ridurre la lunghezza della serie totale per fare considerazioni su di essa sembra essere sensata, soprattutto da quello che emerge nell'ultima sezione del capitolo 3.

Appendice A

A.1 Codice che calcola la log-verosimiglianza con l'opzione V1 per i vincoli.

Si riporta il codice R per il calcolo della funzione di log-verosimiglianza per la distribuzione normale, normale asimmetrica, t di Student, t di Student asimmetrica e per la distribuzione di Subbotin con l'opzione V1 per i vincoli. Ricordiamo che l'opzione V1 fa riferimento all'inserimento con i vincoli con gli argomenti *upper* e *lower* della funzione *optim*.

```

mvdnV1 <- function(theta, r, n)
{
  mu <- theta[1]
  w <- theta[2]
  a1 <- theta[3]
  b1 <- theta[4]
  if (1-a1-b1<=0 | w<=0) return (-10^(9))
  h <- matrix(0, ncol=1, nrow=n)
  h[1] <- w/(1-a1-b1)
  for (i in 2:n) h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
  r <- r[2:n]
  h <- h[2:n]
  lmv <- sum(dnorm(r, mean=mu, sd=sqrt(h), log=TRUE))
  return(lmv)
}

mvdnV1 <- function(theta, r, n)
{
  mu <- theta[1]
  w <- theta[2]
  a1 <- theta[3]
  b1 <- theta[4]
  if (1-a1-b1<=0 | w<=0) return (-10^(9))
  alpha <- theta[5]
  h <- matrix(0, ncol=1, nrow=n)
  h[1] <- w/(1-a1-b1)
  for (i in 2:n) h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
  r <- r[2:n]
  h <- h[2:n]
  lmv <- sum(dsn(x=((r-mu)/sqrt(h)), shape=alpha, log=TRUE) -
log(sqrt(h)))
  return(lmv)
}

```

Appendice A

```
mvdtV1 <- function(theta,r=r_serie,n=n_serie)
{
  mu <- theta[1]
  w <- theta[2]
  a1 <- theta[3]
  b1 <- theta[4]
  v <- theta[5]
  if (1-a1-b1<=0 | w<=0 | v<=2) return(-10^9)
  h <- matrix(0,ncol=1,nrow=n)
  h[1] <- w/(1-a1-b1)
  for (i in 2:n) h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
  r <- r[2:n]
  h <- h[2:n]
  den <- h*(v-2)
  lmv <- sum(dt((r-mu)*sqrt(v/den),df=v,log=TRUE) +
log(sqrt(v/den)))
  return(lmv)
}

mvdstV1 <- function(theta,r,n)
{
  mu <- theta[1]
  w <- theta[2]
  a1 <- theta[3]
  b1 <- theta[4]
  v <- theta[6]
  if (1-a1-b1<=0 | w<=0 | v<=2) return(-10^9)
  alpha <- theta[5]
  h <- matrix(0,ncol=1,nrow=n)
  h[1] <- w/(1-a1-b1)
  for (i in 2:n) h[i] <- w + a1*((r[i-1]-mu)^2) + b1*h[i-1]
  r <- r[2:n]
  h <- h[2:n]
  den <- h*(v-2)
  lmv <- sum(dst(x=(r-mu)*sqrt(v/den), shape=alpha,
df=v,log=TRUE) + log(sqrt(v/den)))
  return(lmv)
}

#####funzioni per la distribuzione di Subbotin #####
subbotin.pdf <- function(x, nu) exp(-abs(x)^nu/nu)/(2*nu^(1/nu-1)*
gamma(1/nu))
subbotin.pdf.log <- function(x, nu) (-abs(x)^nu/nu)- log(2) -
(1/nu-1)*log(nu) - log(gamma(1/nu))
subbotin.cdf <- function(x, nu)
0.5*(1+sign(x)*pgamma(abs(x)^nu/nu, 1/nu))
#####
```

A.1 Codice che calcola la log-verosimiglianza con l'opzione V1 per i vincoli.

```
mvdsuppdfV1 <- function(theta,r=r_serie,n=n_serie)
{
  mu <- theta[1]
  w <- theta[2]
  a1 <- theta[3]
  b1 <- theta[4]
  v <- theta[5] # par. forma
  if (1-a1-b1<=0 | w<=0 | v <=0) return (-10^(9))
  h <- matrix(0,ncol=1,nrow=n)
  h[1] <- w/(1-a1-b1)
  for (i in 2:n) h[i] <- w + a1*((r[i-1]-mu)^2) + b1*h[i-1]
  s <- sqrt(v^(2/v)*gamma(3/v)/gamma(1/v))
  r <- r[2:n]
  h <- h[2:n]
  lmv <- sum((subbotin.pdf.log(x=((r-mu)/sqrt(h))*s,nu=v)) +
log(s/sqrt(h)))
  if(lmv==--Inf) return (-10^(9))
  return(lmv)
}
```

A.2 Codice che calcola la log-verosimiglianza con l'opzione V2 per i vincoli.

Si riporta il codice R per il calcolo della funzione di log-verosimiglianza per la distribuzione normale, normale asimmetrica, t di Student, t di Student asimmetrica e per la distribuzione di Subbotin con l'opzione V2 per i vincoli e le funzioni di rimappatura relative. Ricordiamo che l'opzione V2 fa riferimento all'inserimento dei vincoli con le funzioni di mappatura.

```
mvdnV2 <- function(theta,r,n)
{
  mu <- theta[1]
  w <- exp(theta[2])
  e3 <- exp(theta[3])
  e4 <- exp(theta[4])
  if(w==0 | e3 == Inf | e4 == Inf) return(-10^(9))
  a1 <- e3/(1+e3+e4)
  b1 <- e4/(1+e3+e4)
  h <- matrix(0,ncol=1,nrow=n)
  h[1] <- w*(1+e3+e4)
  for (i in 2:n) h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
  r <- r[2:n]
  h <- h[2:n]
  lmv <- sum(dnorm(r,mean=mu,sd=sqrt(h),log=TRUE))
  return(lmv)
}
```

Appendice A

```
# funzione per la RIMAPPATURA dei parametri
rimapnV2 <- function(theta)
{
  mu <- theta[1]
  w <- exp(theta[2])
  e3 <- exp(theta[3])
  e4 <- exp(theta[4])
  a1 <- e3/(1+e3+e4)
  b1 <- e4/(1+e3+e4)
  theta <- c(mu,w,a1,b1)
  return(theta)
}

mvdsnV2 <- function(theta,r,n)
{
  mu <- theta[1]
  w <- exp(theta[2])
  e3 <- exp(theta[3])
  e4 <- exp(theta[4])
  if (w<=0 | e3 == Inf | e4 == Inf) return(-10^(9))
  a1 <- e3/(1+e3+e4)
  b1 <- e4/(1+e3+e4)
  alpha <- theta[5]
  h <- matrix(0,ncol=1,nrow=n)
  h[1] <- w*(1+e3+e4) # /(1-a1-b1)
  for (i in 2:n) h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
  r <- r[2:n]
  h <- h[2:n]
  lmv <- sum(dsn(x=((r-mu)/sqrt(h)),shape=alpha,log=TRUE) -
log(sqrt(h)))
  return(lmv)
}

rimapsnV2 <- function(theta)
{
  mu <- theta[1]
  w <- exp(theta[2])
  e3 <- exp(theta[3])
  e4 <- exp(theta[4])
  a1 <- e3/(1+e3+e4)
  b1 <- e4/(1+e3+e4)
  alpha <- theta[5]
  theta <- c(mu,w,a1,b1,alpha)
  return(theta)
}
```

A.2 Codice che calcola la log-verosimiglianza con l'opzione V2 per i vincoli.

```
mvdtV2 <- function(theta,r=r_serie,n=n_serie)
{
  mu <- theta[1]
  w <- exp(theta[2])
  e3 <- exp(theta[3])
  e4 <- exp(theta[4])
  v <- exp(theta[5])
  if(w<=0 | e3 == Inf | e4 == Inf | v<=2) return(-10^(9))
  a1 <- e3/(1+e3+e4)
  b1 <- e4/(1+e3+e4)
  h <- matrix(0,ncol=1,nrow=n)
  h[1] <- w*(1+e3+e4)
  for (i in 2:n) h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
  r <- r[2:n]
  h <- h[2:n]
  den <- h*(v-2)
  lmv <- sum(dt((r-mu)*sqrt(v/den),df=v,log=TRUE) +
log(sqrt(v/den)))
  return(lmv)
}

rimaptV2 <- function(theta)
{
  mu <- theta[1]
  w <- exp(theta[2])
  e3 <- exp(theta[3])
  e4 <- exp(theta[4])
  a1 <- e3/(1+e3+e4)
  b1 <- e4/(1+e3+e4)
  v <- exp(theta[5])
  theta <- c(mu,w,a1,b1,v)
  return(theta)
}

mvdstV2 <- function(theta,r,n)
{
  mu <- theta[1]
  w <- exp(theta[2])
  e3 <- exp(theta[3])
  e4 <- exp(theta[4])
  v <- exp(theta[6])
  if(w<=0 | e3 == Inf | e4 == Inf | v<=2) return(-10^(9))
  a1 <- e3/(1+e3+e4)
  b1 <- e4/(1+e3+e4)
  alpha <- theta[5]
  h <- matrix(0,ncol=1,nrow=n)
  h[1] <- w*(1+e3+e4)
  for (i in 2:n) h[i] <- w + a1*((r[i-1]-mu)^2) + b1*h[i-1]
  r <- r[2:n]
  h <- h[2:n]
  den <- h*(v-2)
  lmv <- sum(dst(x=(r-mu)*sqrt(v/den), shape=alpha,
df=v,log=TRUE) + log(sqrt(v/den)))
  return(lmv)
}
```

Appendice A

```
rimapstV2 <- function(theta)
{
  mu <- theta[1]
  w <- exp(theta[2])
  e3 <- exp(theta[3])
  e4 <- exp(theta[4])
  a1 <- e3/(1+e3+e4)
  b1 <- e4/(1+e3+e4)
  alpha <- theta[5]
  v <- exp(theta[6])
  theta <- c(mu,w,a1,b1,alpha,v)
  return(theta)
}

mvdsbpdfV2 <- function(theta,r=r_serie,n=n_serie)
{
  mu <- theta[1]
  w <- exp(theta[2])
  e3 <- exp(theta[3])
  e4 <- exp(theta[4])
  v <- exp(theta[5])
  if (w<=0 | e3 == Inf | e4 == Inf | v == 0 ) return(-10^(9))
  a1 <- e3/(1+e3+e4)
  b1 <- e4/(1+e3+e4)
  h <- matrix(0,ncol=1,nrow=n)
  h[1] <- w*(1+e3+e4)
  for (i in 2:n) h[i] <- w + a1*((r[i-1]-mu)^2) + b1*h[i-1]
  s <- sqrt(v^(2/v)*gamma(3/v)/gamma(1/v))
  r <- r[2:n]
  h <- h[2:n]
  lmv <- sum((subbotin.pdf.log(x=((r-mu)/sqrt(h))*s,nu=v)) +
log(s/sqrt(h)))
  return(lmv)
}
```

A.3 Codice che calcola la log-verosimiglianza con l'opzione V3 per i vincoli.

Si riporta il codice R per il calcolo della funzione di log-verosimiglianza per la distribuzione normale, normale asimmetrica, t di Student, t di Student asimmetrica e per la distribuzione di Subbotin con l'opzione V3 per i vincoli. Ricordiamo che l'opzione V3 fa riferimento all'inserimento dei vincoli con “*IF...ELSE*”.

A.3 Codice che calcola la log-verosimiglianza con l'opzione V3 per i vincoli.

```
mvdnV3 <- function(theta, r, n)
{
  mu <- theta[1]
  if (theta[2] > 0) w <- theta[2]
  else return(-10^(9))
  if (theta[3]+theta[4] < 1 & theta[3] > 0 & theta[4] > 0)
  {
    a1 <- theta[3]
    b1 <- theta[4]
  }
  else return(-10^(9))
  h <- matrix(0, ncol=1, nrow=n)
  h[1] <- w/(1-a1-b1)
  for (i in 2:n) h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
  r <- r[2:n]
  h <- h[2:n]
  lmv <- sum(dnorm(r, mean=mu, sd=sqrt(h), log=TRUE))
  return(lmv)
}

mvdsnV3 <- function(theta, r, n)
{
  mu <- theta[1]
  if (theta[2] > 0) w <- theta[2]
  else return(-10^(9))
  alpha <- theta[5]
  if ((theta[3]+theta[4]) < 1 & theta[3] > 0 & theta[4] > 0)
  {
    a1 <- theta[3]
    b1 <- theta[4]
  }
  else return(-10^(9))
  h <- matrix(0, ncol=1, nrow=n)
  h[1] <- w/(1-a1-b1)
  for (i in 2:n) h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
  r <- r[2:n]
  h <- h[2:n]
  lmv <- sum(dsn(x=((r-mu)/sqrt(h)), shape=alpha, log=TRUE) -
log(sqrt(h)))
  return(lmv)
}

mvdtV3 <- function(theta, r, n)
{
  mu <- theta[1]
  if (theta[2] > 0) w <- theta[2]
  else return(-10^9)
  if (theta[5] > 2) v <- theta[5]
  else return(-10^9)
  if (theta[3]+theta[4] < 1 & theta[3] > 0 & theta[4] > 0)
  {
    a1 <- theta[3]
    b1 <- theta[4]
  }
  else return(-10^(9))
  h <- matrix(0, ncol=1, nrow=n)
  h[1] <- w/(1-a1-b1)
  for (i in 2:n) h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
  r <- r[2:n]
  h <- h[2:n]
  lmv <- sum(dt((r-mu)*sqrt(v/(h*(v-2))), df=v, log=TRUE) +
log(sqrt(v/(h*(v-2))))))
  return(lmv)
}
```

Appendice A

```
mvdstV3 <- function(theta, r, n)
{
  mu <- theta[1]
  alpha <- theta[5]
  v <- theta[6]
  if (v < 2) return(-10^9)
  if (theta[2] > 0) w <- theta[2]
  else return(-10^9)
  if (theta[3]+theta[4] < 1 & theta[3] > 0 & theta[4] > 0)
  {
    a1 <- theta[3]
    b1 <- theta[4]
  }
  else return(-10^9)
  h <- matrix(0, ncol=1, nrow=n)
  h[1] <- w/(1-a1-b1)
  for (i in 2:n) h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
  r <- r[2:n]
  h <- h[2:n]
  den <- h*(v-2)
  lmv <- sum(dst(x=(r-mu)*sqrt(v/den), shape=alpha,
df=v, log=TRUE) + log(sqrt(v/den)))
  return(lmv)
}

mvdsuppdfV3 <- function(theta, r, n)
{
  mu <- theta[1]
  if (theta[2] > 0) w <- theta[2]
  else return(-10^9)
  if (theta[5] > 0) v <- theta[5]
  else return(-10^9)
  if (theta[3]+theta[4] < 1 & theta[3] > 0 & theta[4] > 0)
  {
    a1 <- theta[3]
    b1 <- theta[4]
  }
  else return(-10^9)
  h <- matrix(0, ncol=1, nrow=n)
  h[1] <- w/(1-a1-b1)
  for (i in 2:n) h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
  s <- sqrt(v^(2/v)*gamma(3/v)/gamma(1/v))
  r <- r[2:n]
  h <- h[2:n]
  lmv <- sum(log((subbotin.pdf(x=((r-mu)/sqrt(h))*s, nu=v))) +
log(s/sqrt(h)))
  if(lmv== -Inf) return(-10^9) # è dovuto log(0) = -Inf
  return(lmv)
}
```

A.4 Codice che genera serie con l'opzione V3 per i vincoli.

Si riporta il codice R per la generazione di serie simulate per la distribuzione normale, normale asimmetrica, t di Student e t di Student asimmetrica con l'opzione V3 per i vincoli.

A.4 Codice che genera serie con l'opzione V3 per i vincoli.

```
# NORMALE #
sim_mvdnV3 <- function(theta,n,n_start=100)
{
  r <- rep(0,(n+n_start))
  h <- rep(0,(n+n_start))
  mu <- theta[1]
  if (theta[2] > 0) w <- theta[2]
  else return(-10^(9))
  n01 <- rnorm(n+n_start)
  if (theta[3]+theta[4] < 1 & theta[3] > 0 & theta[4] > 0)
  {
    a1 <- theta[3]
    b1 <- theta[4]
  }
  else return(-10^(9))
  h[1] <- w/(1-a1-b1)
  r[1] <- mu + n01[1] * sqrt(h[1])
  for (i in 2:n+n_start)
  {
    h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
    r[i] <- mu + n01[i]*sqrt(h[i])
  }
  return(r[(n_start+1):(n+n_start)])
}

# T #
sim_mvdtV3 <- function(theta,n,n_start=100)
{
  r <- rep(0,(n+n_start))
  h <- rep(0,(n+n_start))
  mu <- theta[1]
  if (theta[2] > 0) w <- theta[2]
  else return(-10^9)
  if (theta[5] > 0) v <- theta[5]
  else return(-10^9)
  t01 <- rt((n+n_start),df=v)*sqrt((v-2)/v)
  if (theta[3]+theta[4] < 1 & theta[3] > 0 & theta[4] > 0)
  {
    a1 <- theta[3]
    b1 <- theta[4]
  }
  else return(-10^(9))
  h[1] <- w/(1-a1-b1)
  r[1] <- mu + t01[1]*sqrt(h[1])
  for (i in 2:n+n_start)
  {
    h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
    r[i] <- mu + t01[i]*sqrt(h[i])
  }
  return(r[(n_start+1):(n+n_start)])
}
```

Appendice A

```
# SN #
sim_mvdsnV3 <- function(theta,n,n_start=100)
{
  r <- rep(0,(n+n_start))
  h <- rep(0,(n+n_start))
  mu <- theta[1]
  if (theta[2] > 0) w <- theta[2]
  else return(-10^(9))
  alpha <- theta[5]
  sn01 <- rsn((n+n_start),shape=alpha)
  if (theta[3]+theta[4] < 1 & theta[3] > 0 & theta[4] > 0)
  {
    a1 <- theta[3]
    b1 <- theta[4]
  }
  else return(-10^(9))
  h[1] <- w/(1-a1-b1)
  r[1] <- mu + sn01[1] * sqrt(h[1])
  for (i in 2:n+n_start)
  {
    h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
    r[i] <- mu + sn01[i]*sqrt(h[i])
  }
  return(r[(n_start+1):(n+n_start)])
}

# ST #
sim_mvdstV3 <- function(theta,n,n_start=100)
{
  r <- rep(0,(n+n_start))
  h <- rep(0,(n+n_start))
  mu <- theta[1]
  alpha <- theta[5]
  if (theta[6] > 0) v <- theta[6]
  else return(-10^9)
  if (theta[2] > 0) w <- theta[2]
  else return(-10^9)
  st01 <- rst((n+n_start),shape=alpha,df=v)*sqrt((v-2)/v)
  if (theta[3]+theta[4] < 1 & theta[3] > 0 & theta[4] > 0)
  {
    a1 <- theta[3]
    b1 <- theta[4]
  }
  else return(-10^(9))
  h[1] <- w/(1-a1-b1)
  r[1] <- mu + st01[1]*sqrt(h[1])
  for (i in 2:n+n_start)
  {
    h[i] <- w + a1*(r[i-1]-mu)^2 + b1*h[i-1]
    r[i] <- mu + st01[i]*sqrt(h[i])
  }
  return(r[(n_start+1):(n+n_start)])
}
```

Appendice B

In questa appendice sono riportate le figure e le tabelle del capitolo 3, in quanto vista la ricerca di una maggiore immediatezza nel cogliere le particolarità evidenziate, gli abbiamo fatto occupare una pagina a ciascuna.

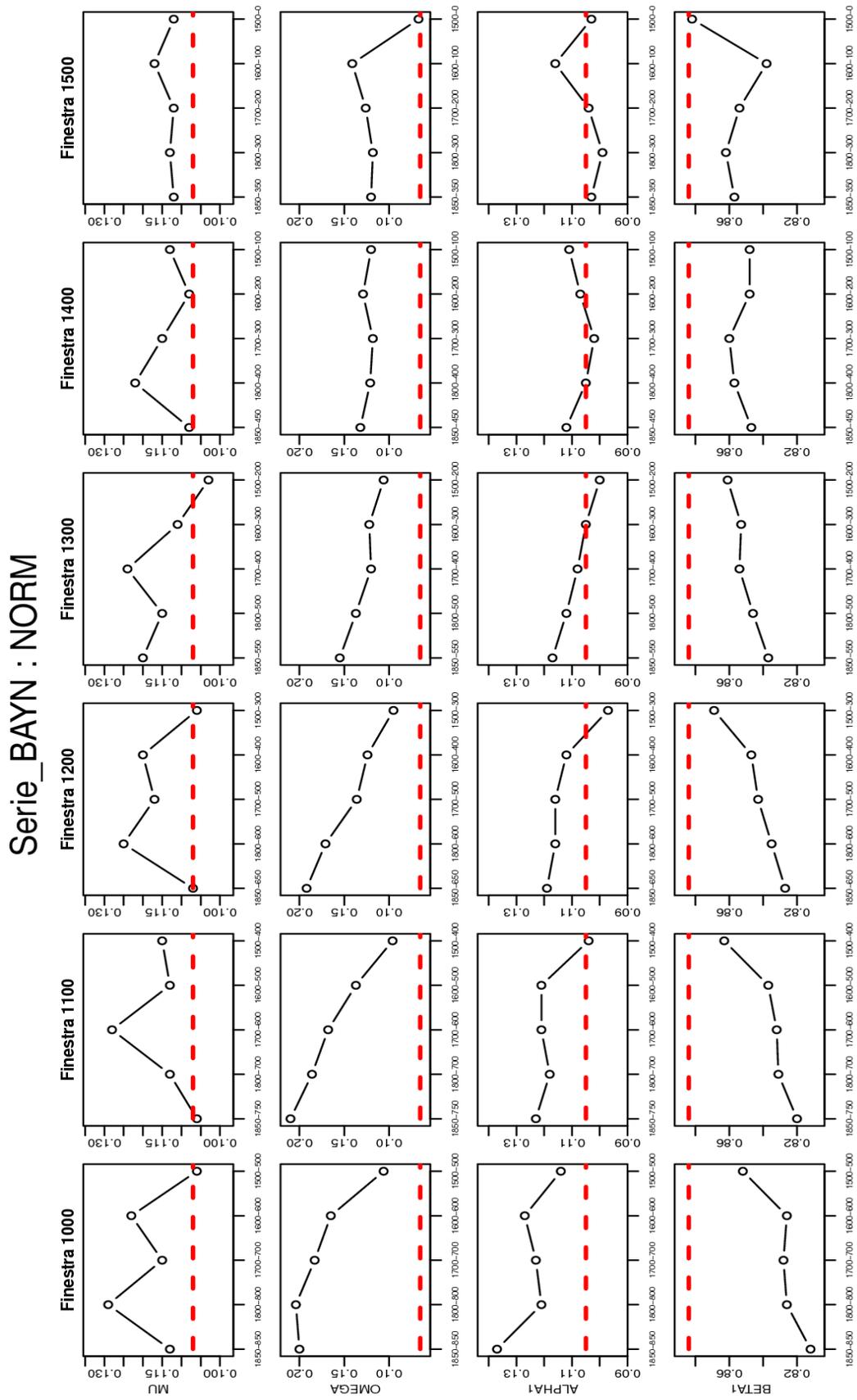


Figura B.1: Serie BAYN nel caso distribuzione normale con slittamenti finestra.

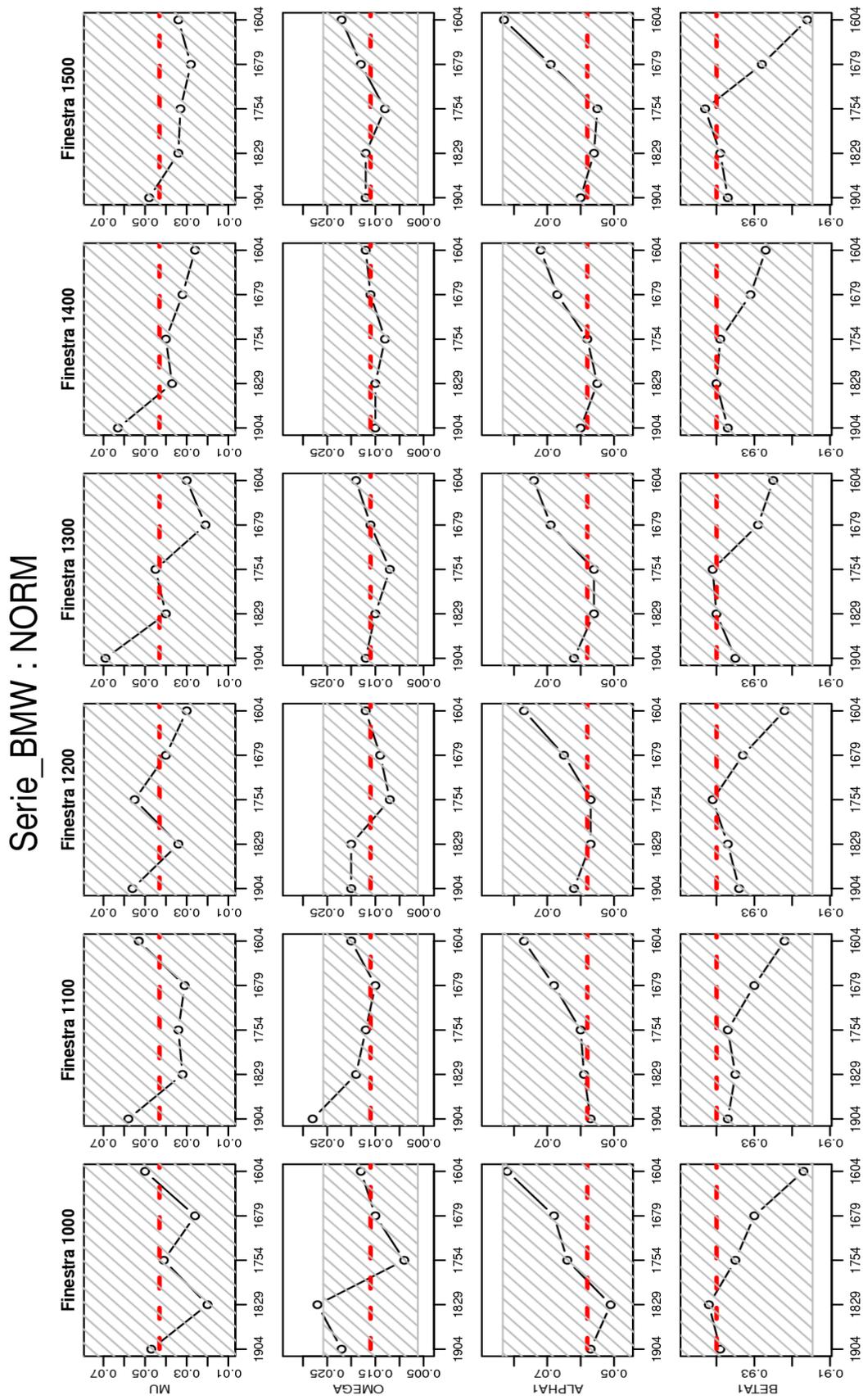


Figura B.2: Serie BMW nel caso della distribuzione normale con intervallo di confidenza al 95%.

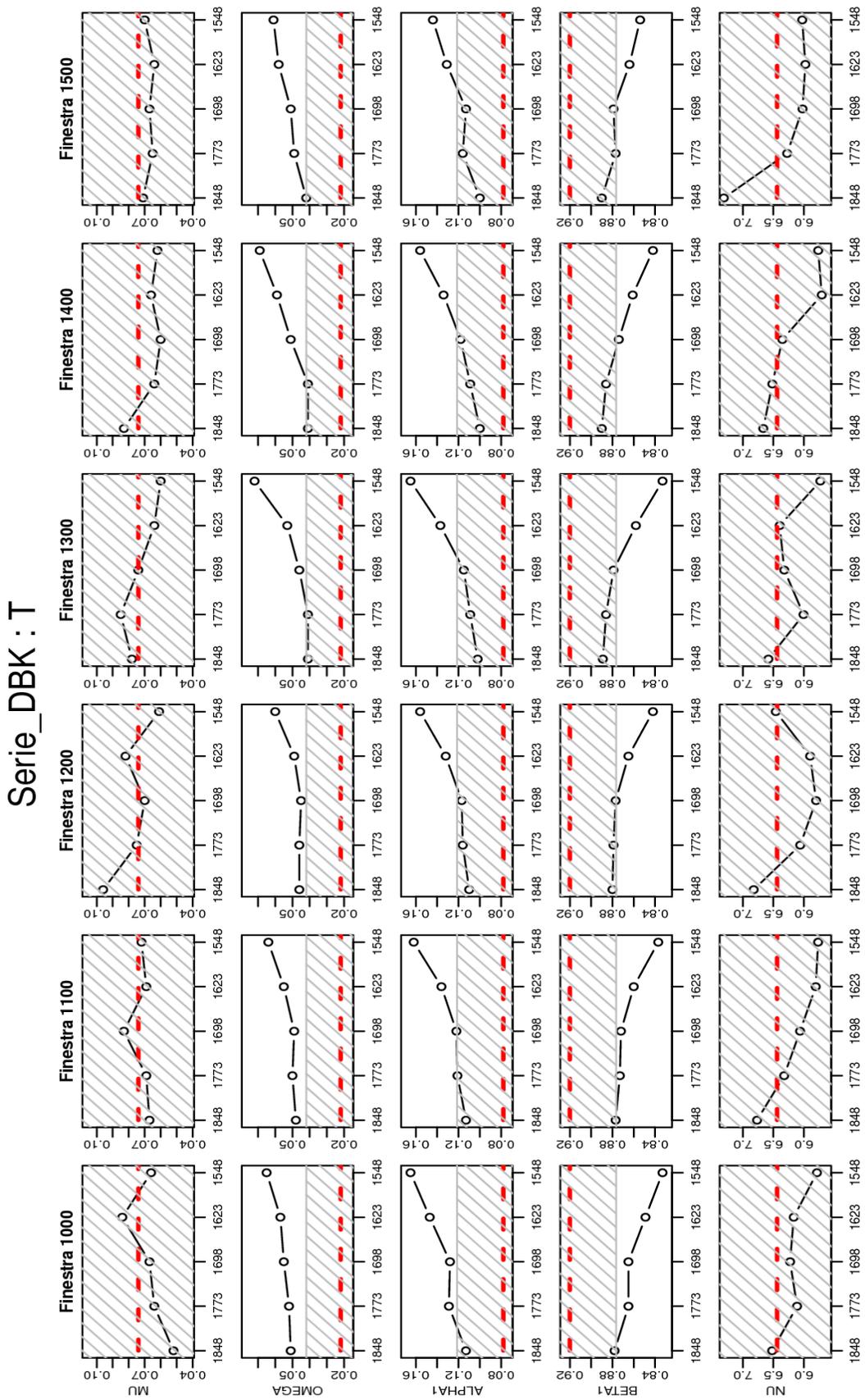


Figura B.3: Serie DBK nel caso della t di Student con intervallo di confidenza al 95%.

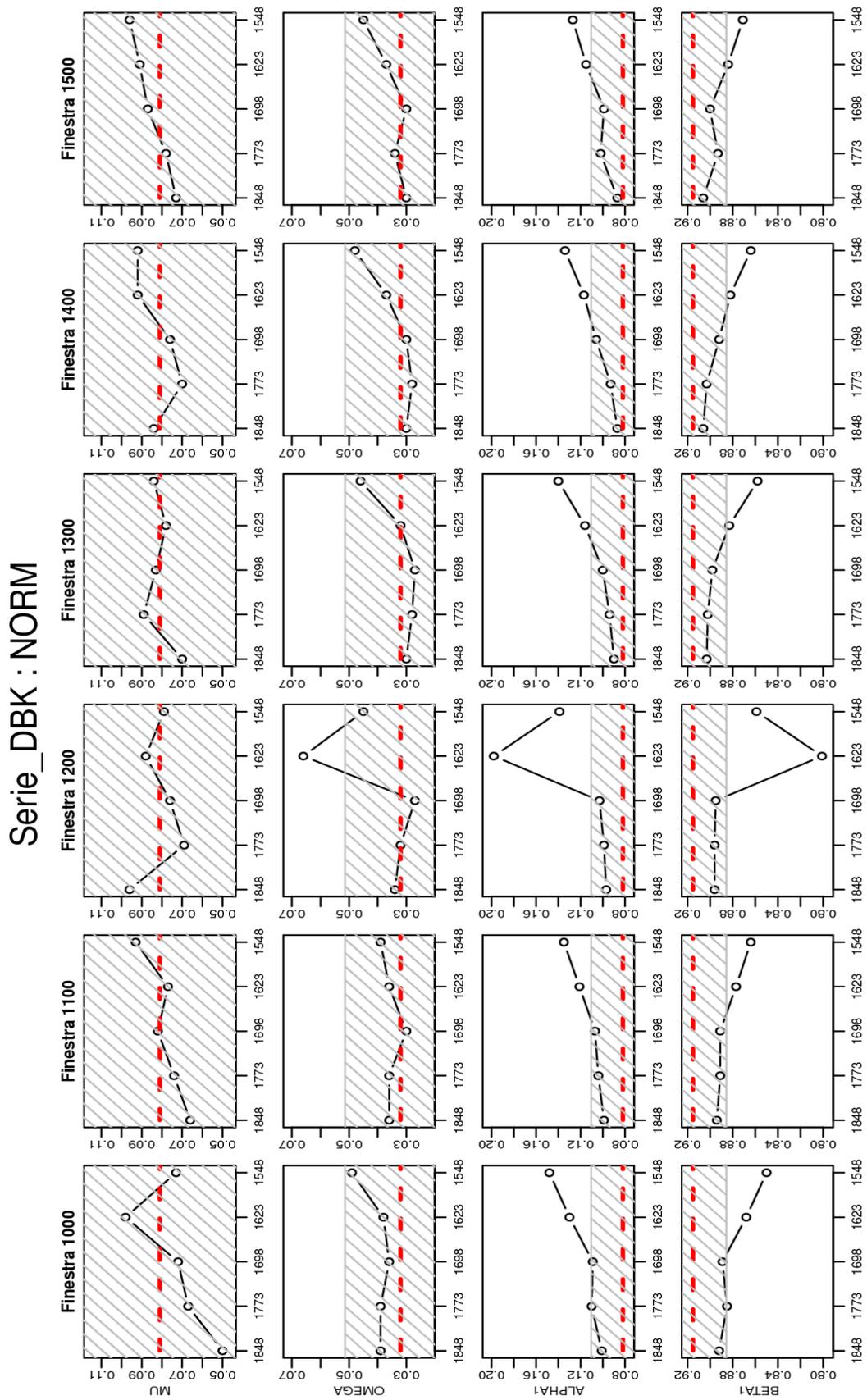


Figura B.4: Serie DBK nel caso della normale con intervallo di confidenza al 95%.

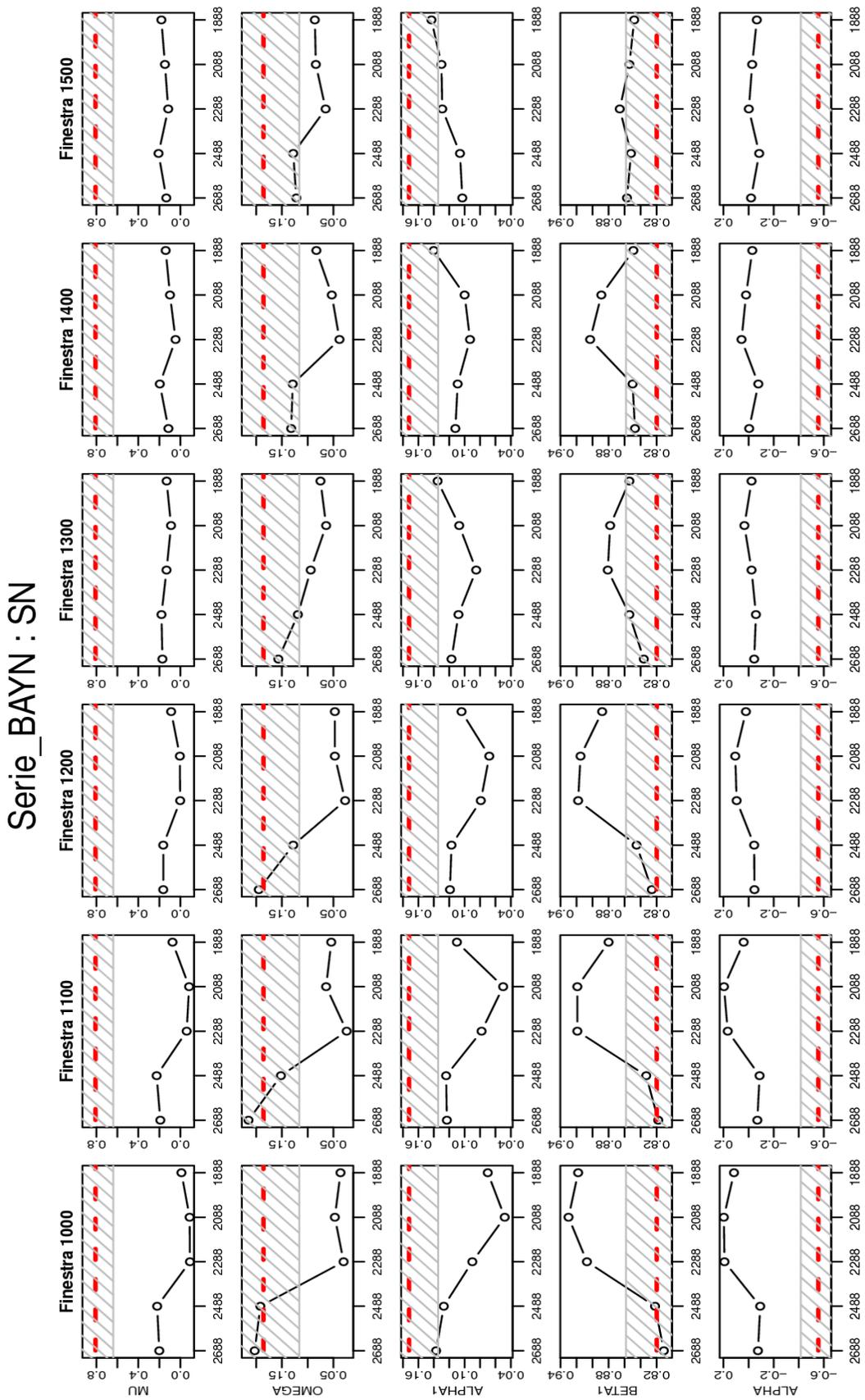


Figura B.5: Serie BAYN nel caso della normale asimmetrica con intervallo di confidenza al 95%.

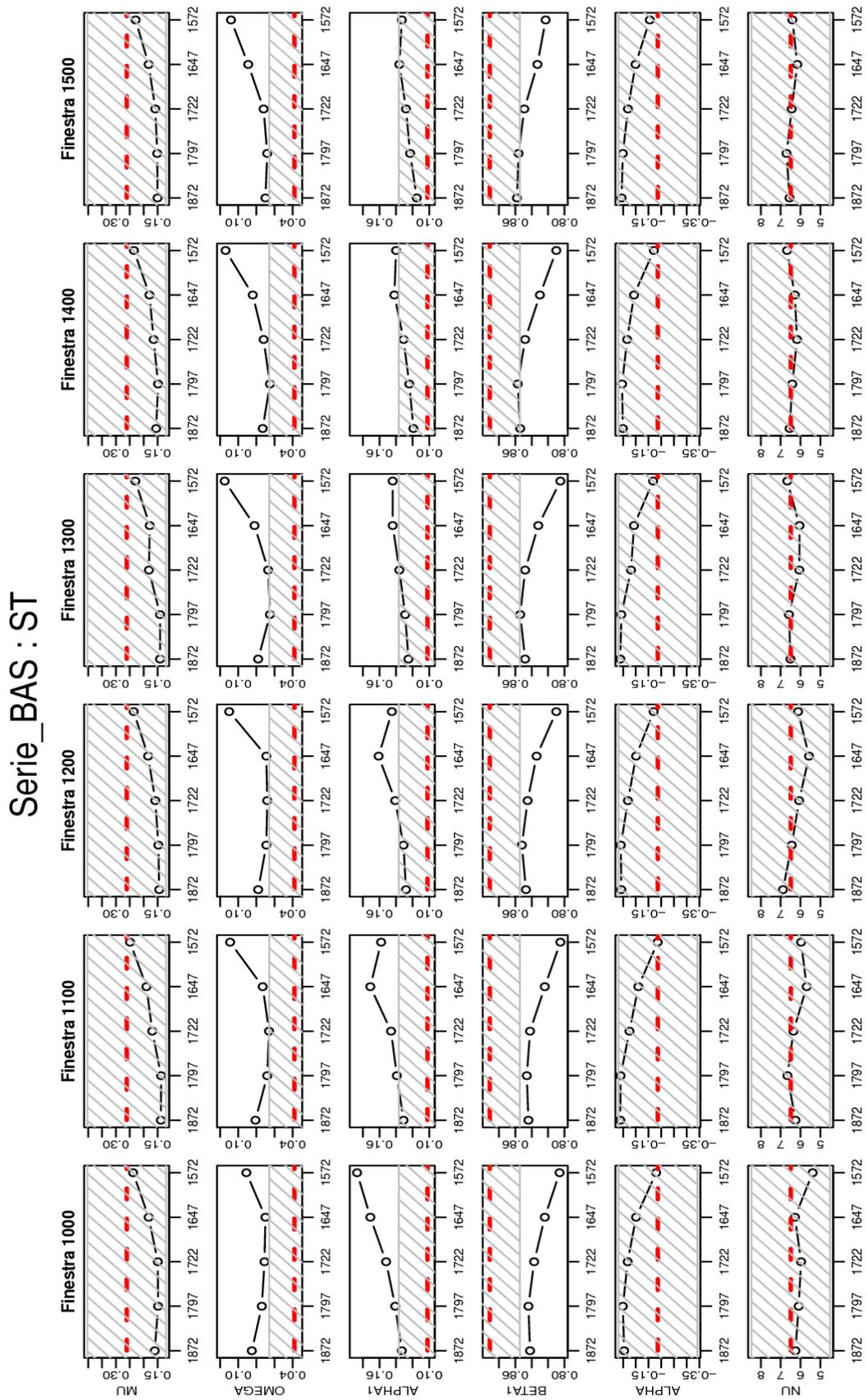


Figura B.6: Serie BAS nel caso della t di Student asimmetrica con intervallo di confidenza al 95%.

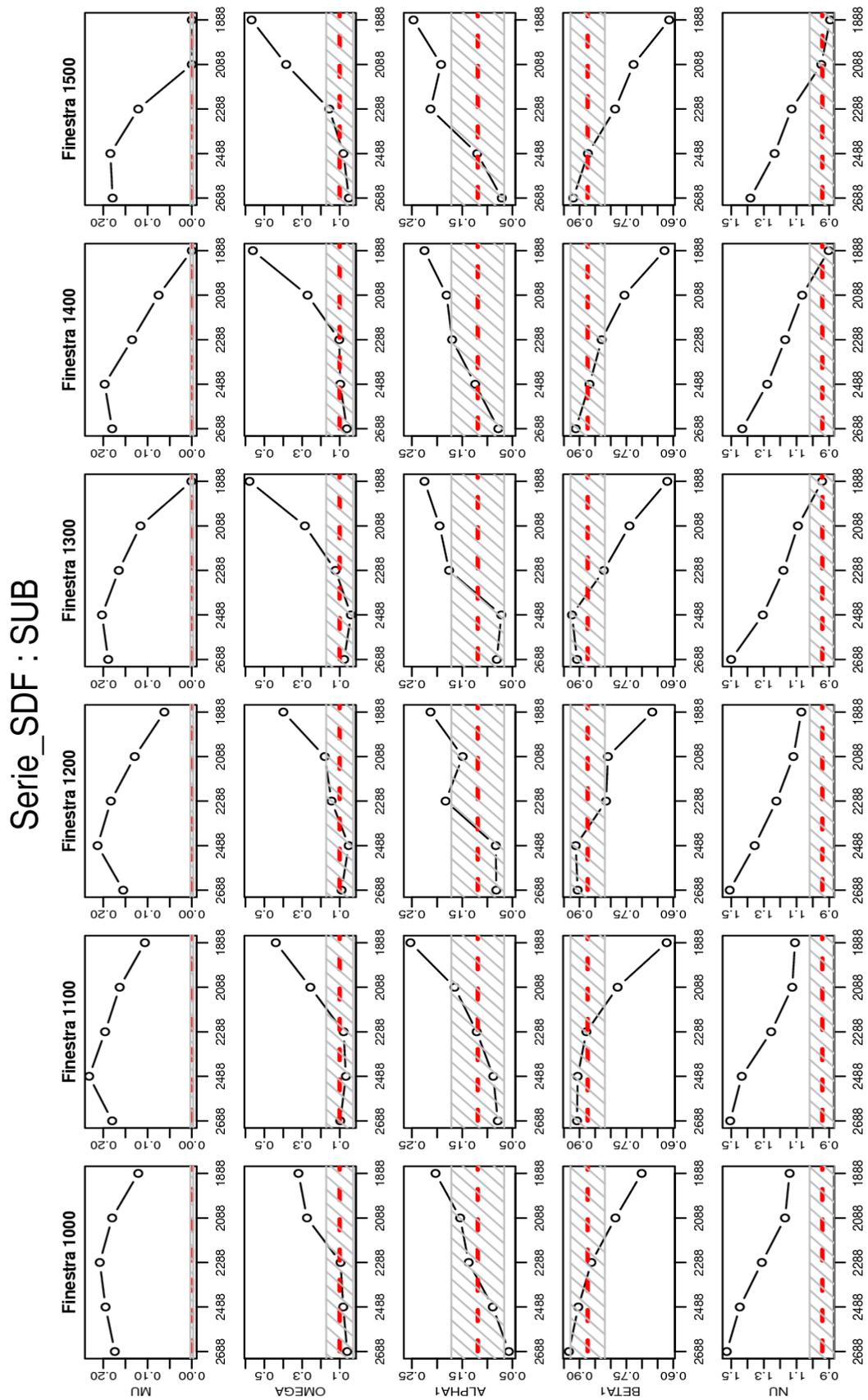


Figura B.7: Serie SDF nel caso della distribuzione di Subbotin con intervallo di confidenza al 95%.

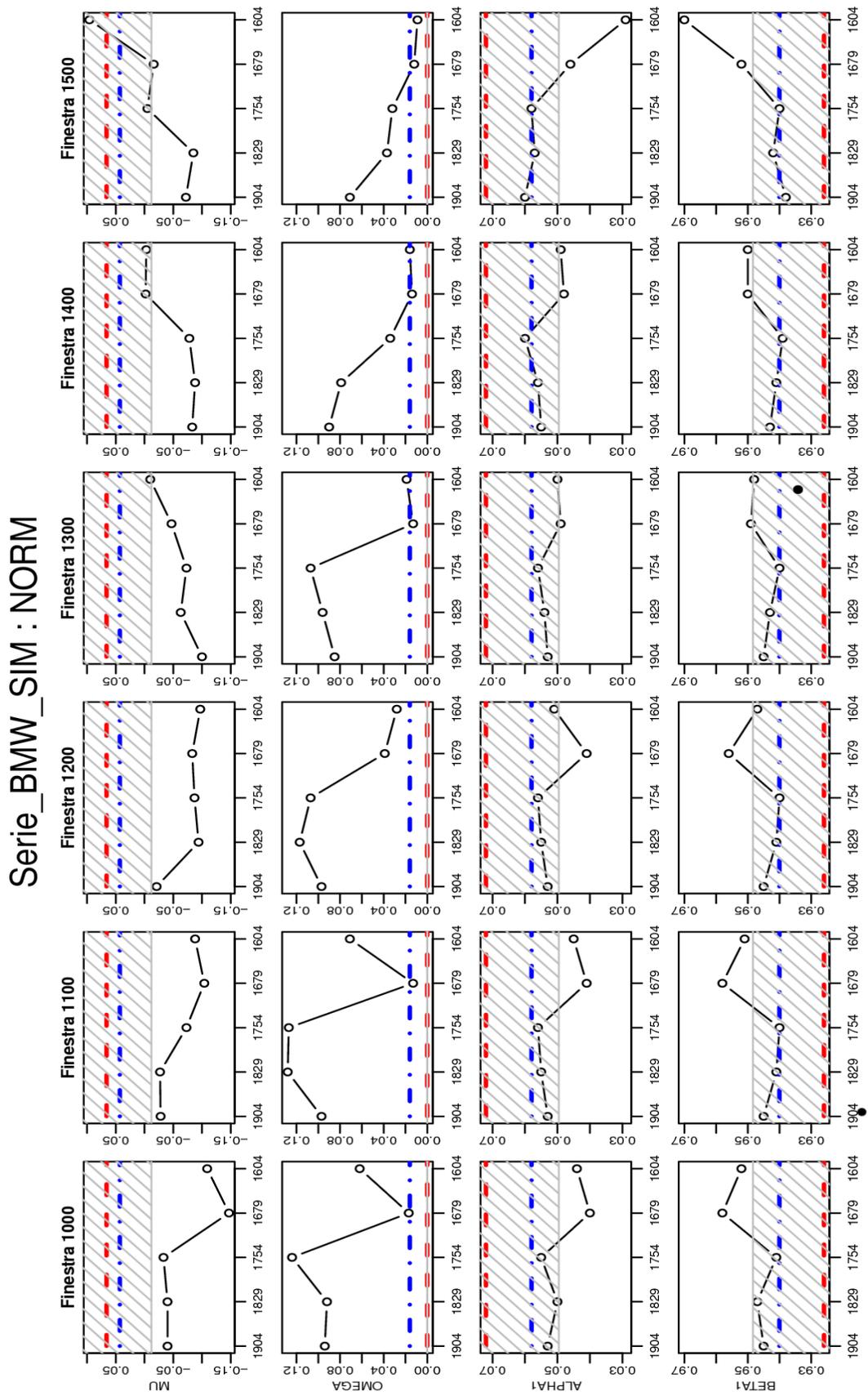


Figura B.8: Serie BMW simulata nel caso della normale con intervallo di confidenza al 95%.

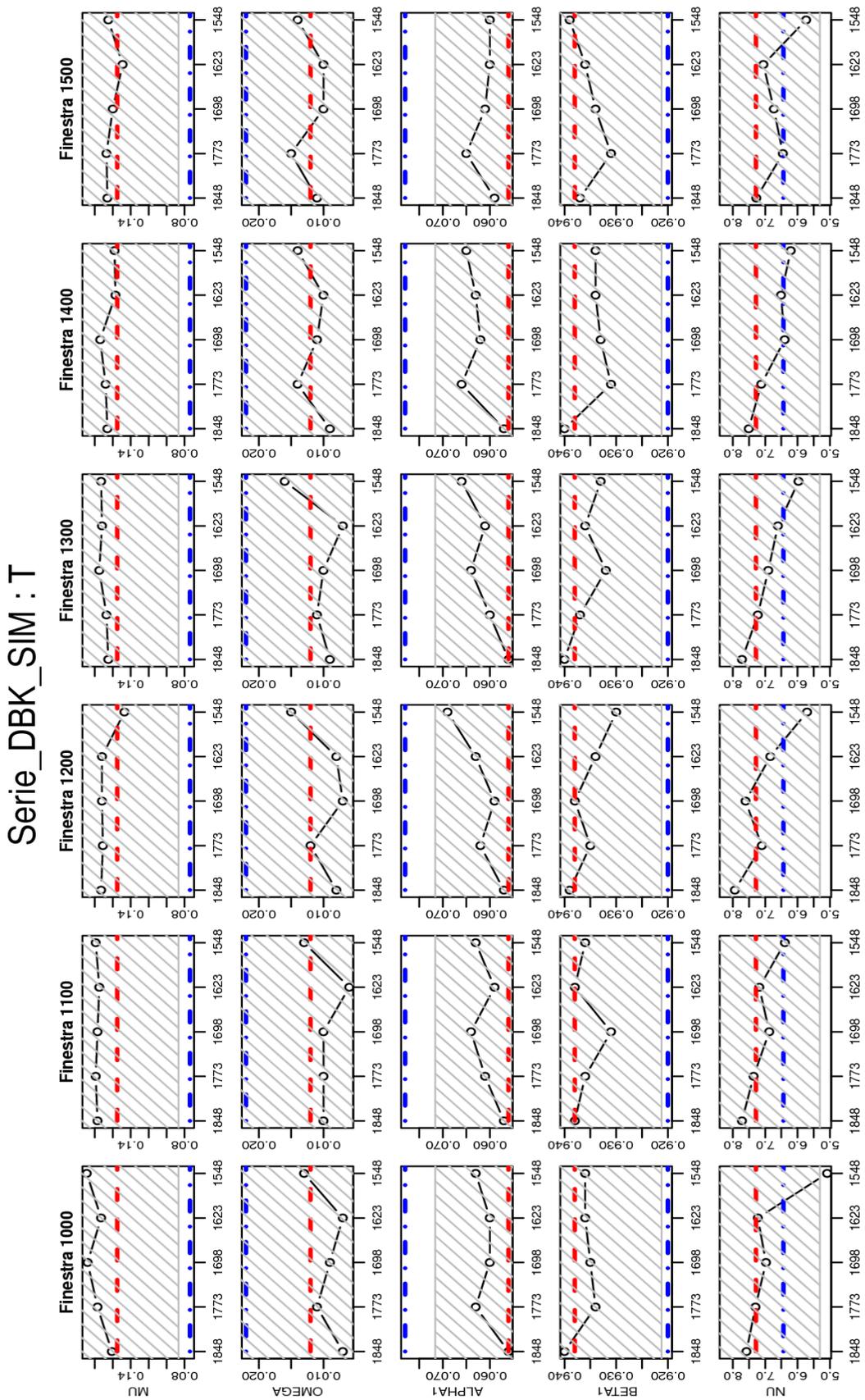


Figura B.9: Serie DBK simulata nel caso della t di Student con intervallo di confidenza al 95%.

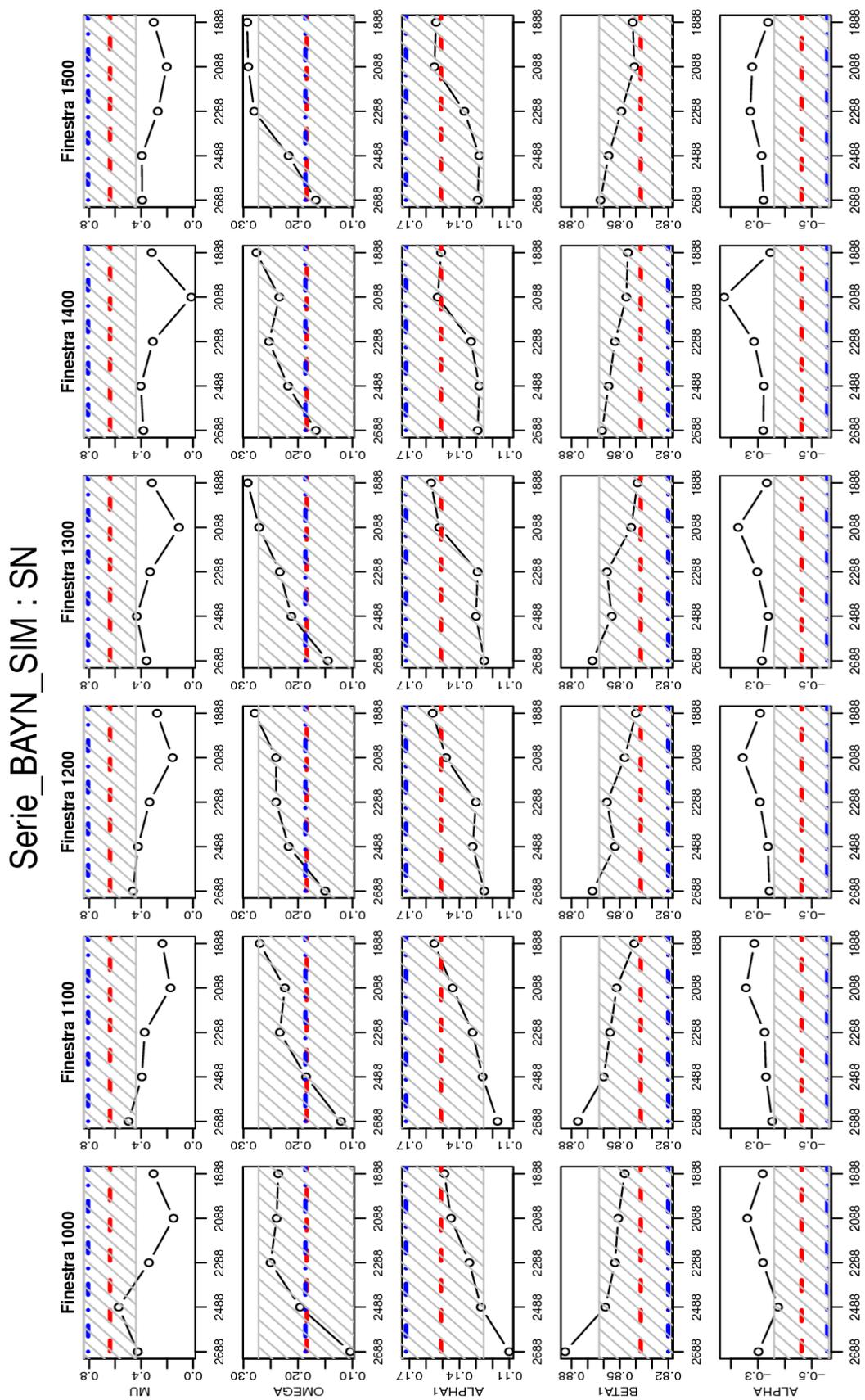


Figura B.10: Serie BAYN simulata nel caso della normale con intervallo di confidenza al 95%.

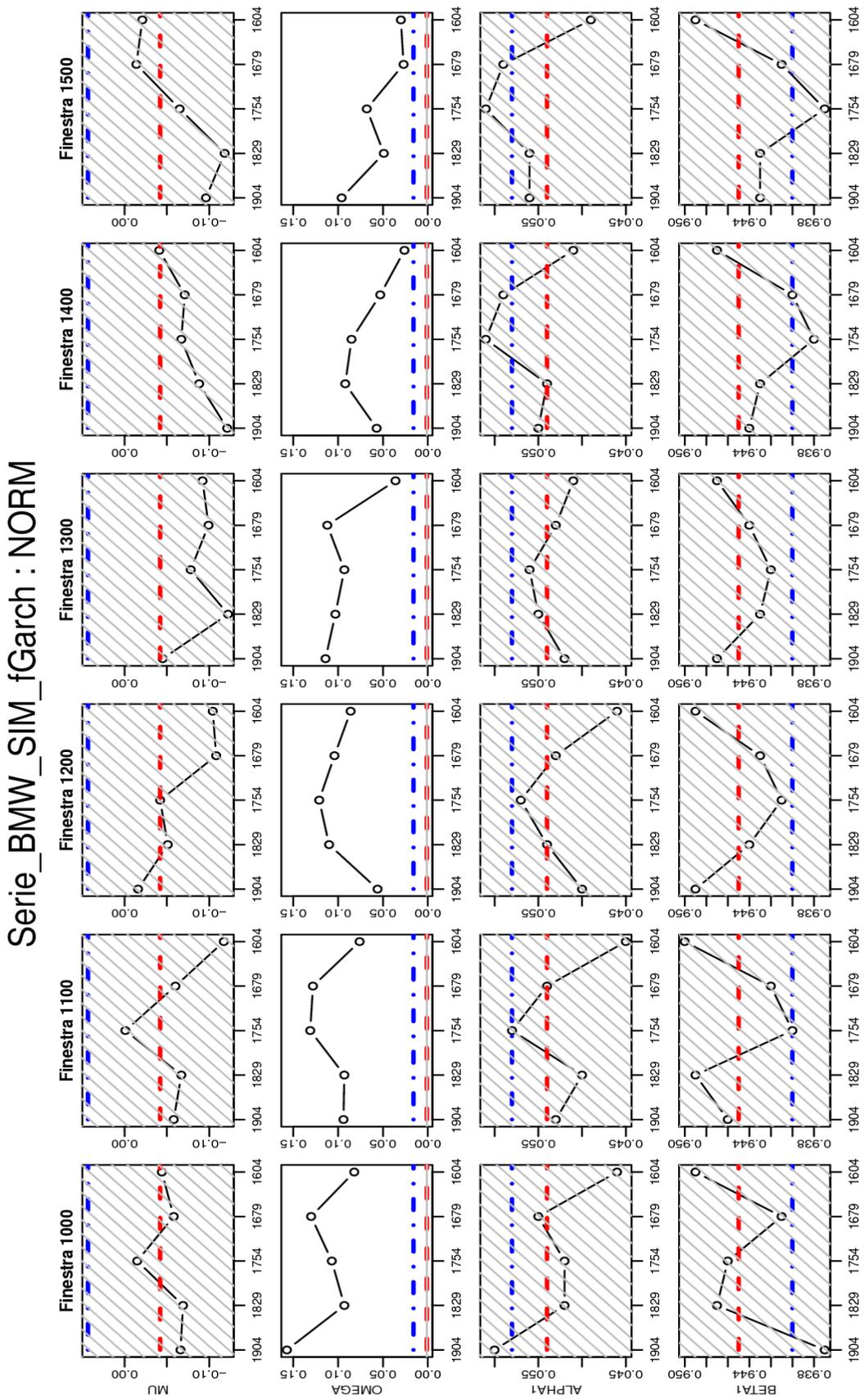


Figura B.11: Serie BMW simulata con *fGarch* nel caso della normale con intervallo di confidenza al 95%.

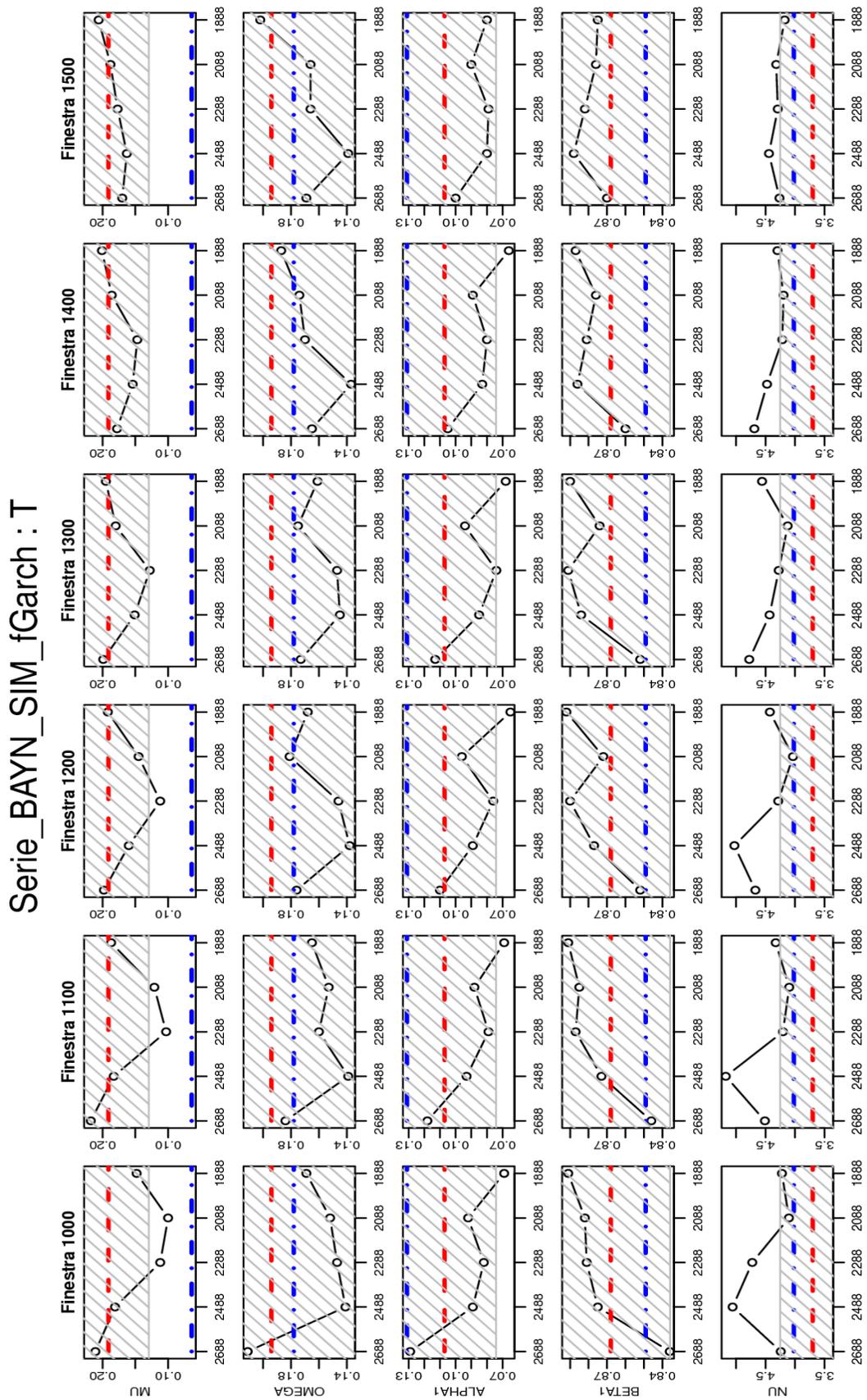


Figura B.12: Serie BMW simulata con $fGarch$ nel caso della t di Student con intervallo di confidenza al 95%.

Serie_BAS : NORM

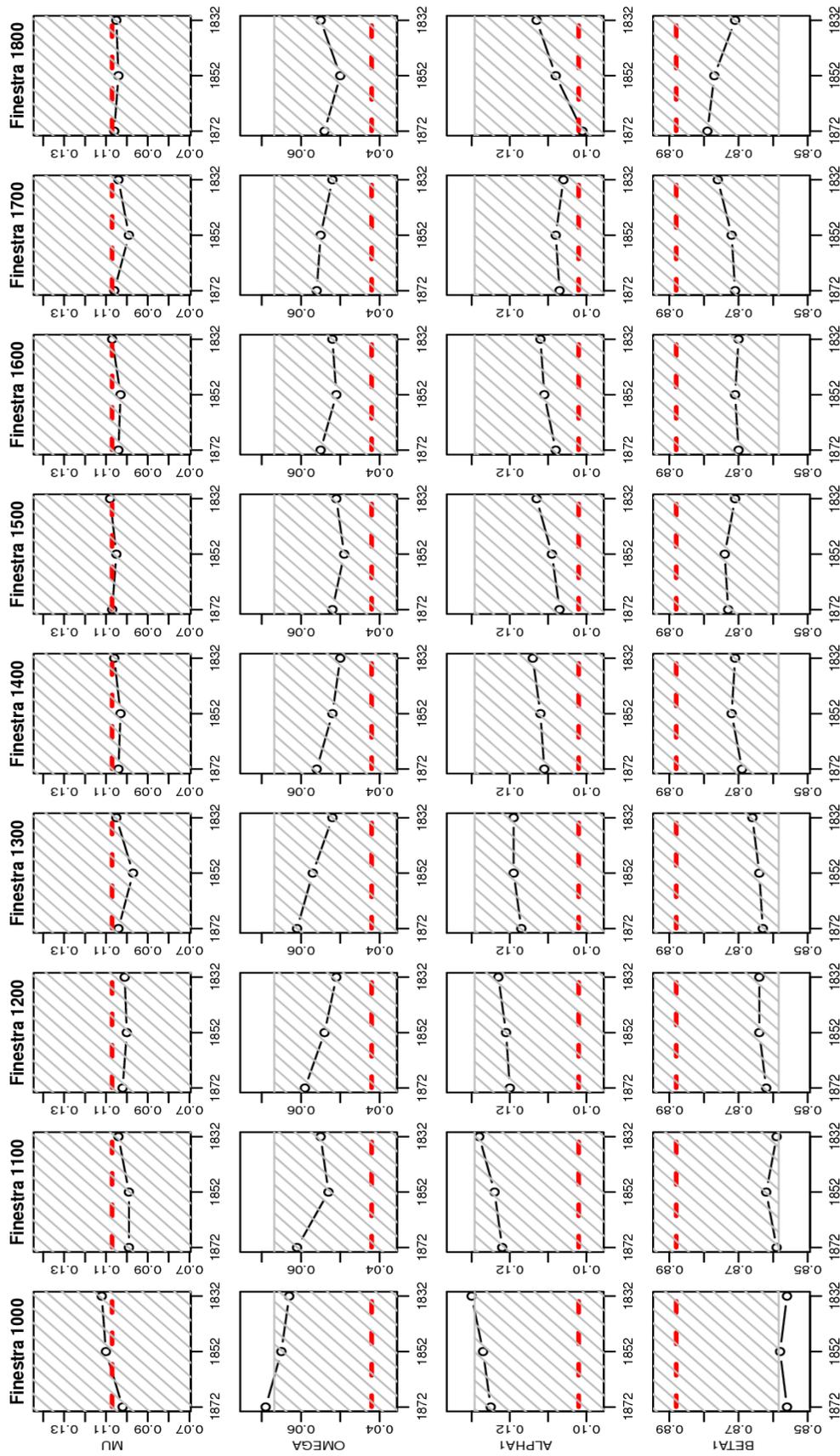


Figura B.13: Serie BAS con intervallo di confidenza al 95% e n_0 che cresce.

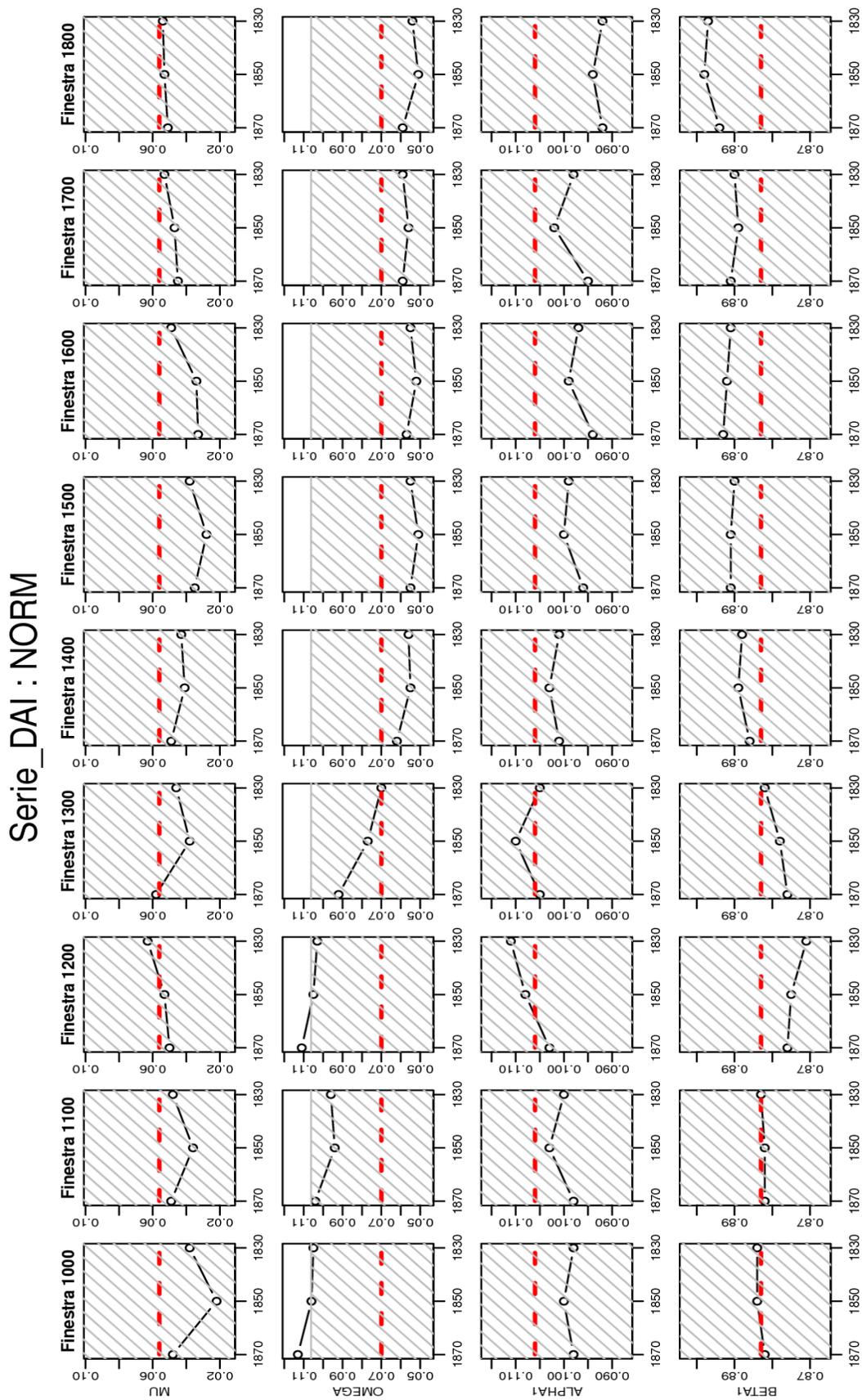


Figura B.14: Serie DAI con intervallo di confidenza al 95% e n_0 che cresce.

Serie_SDF : NORM

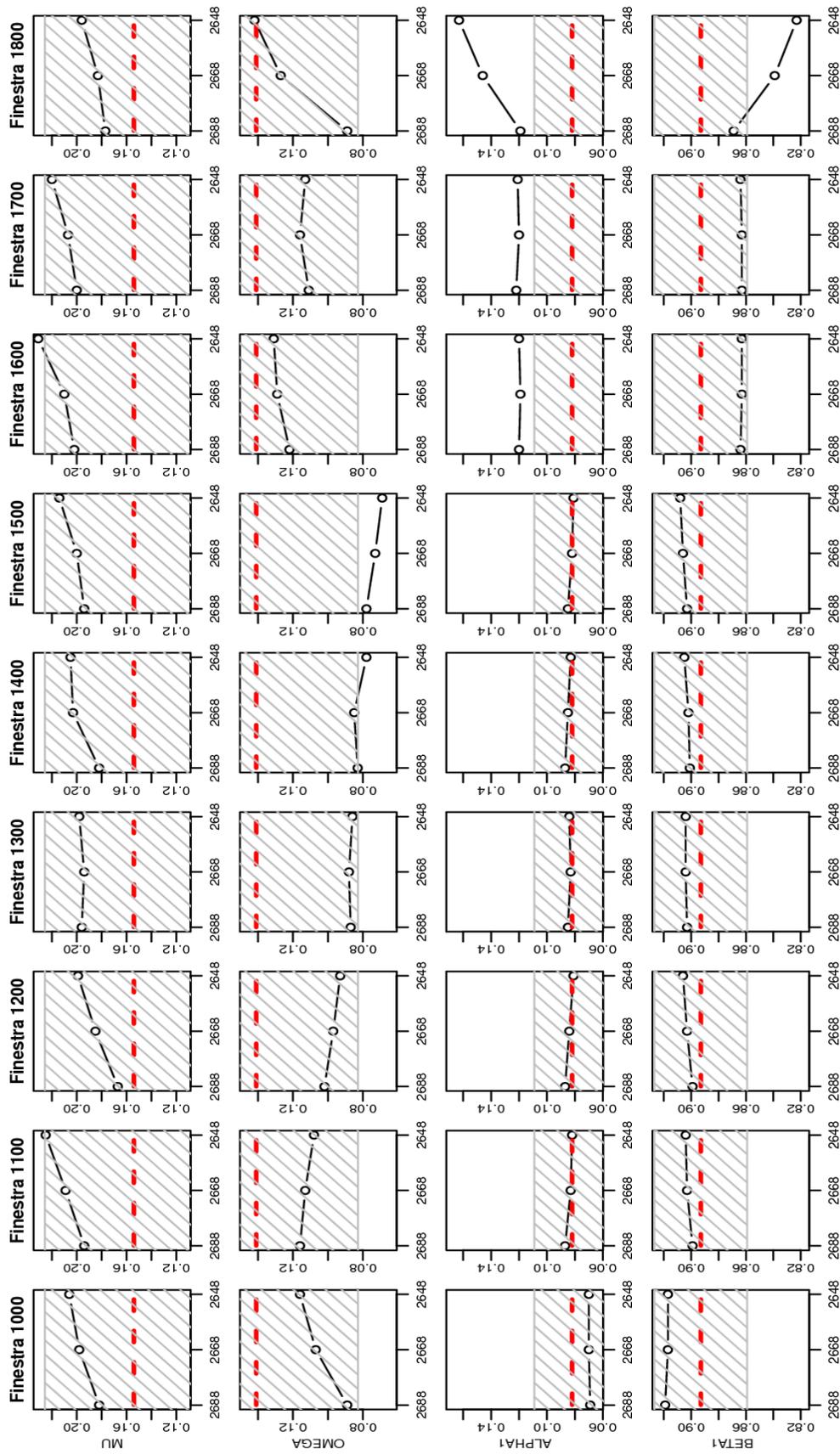


Figura B.15: Serie SDF con intervallo di confidenza al 95% e n_0 che cresce.

Serie_BAS : NORM stime_parti

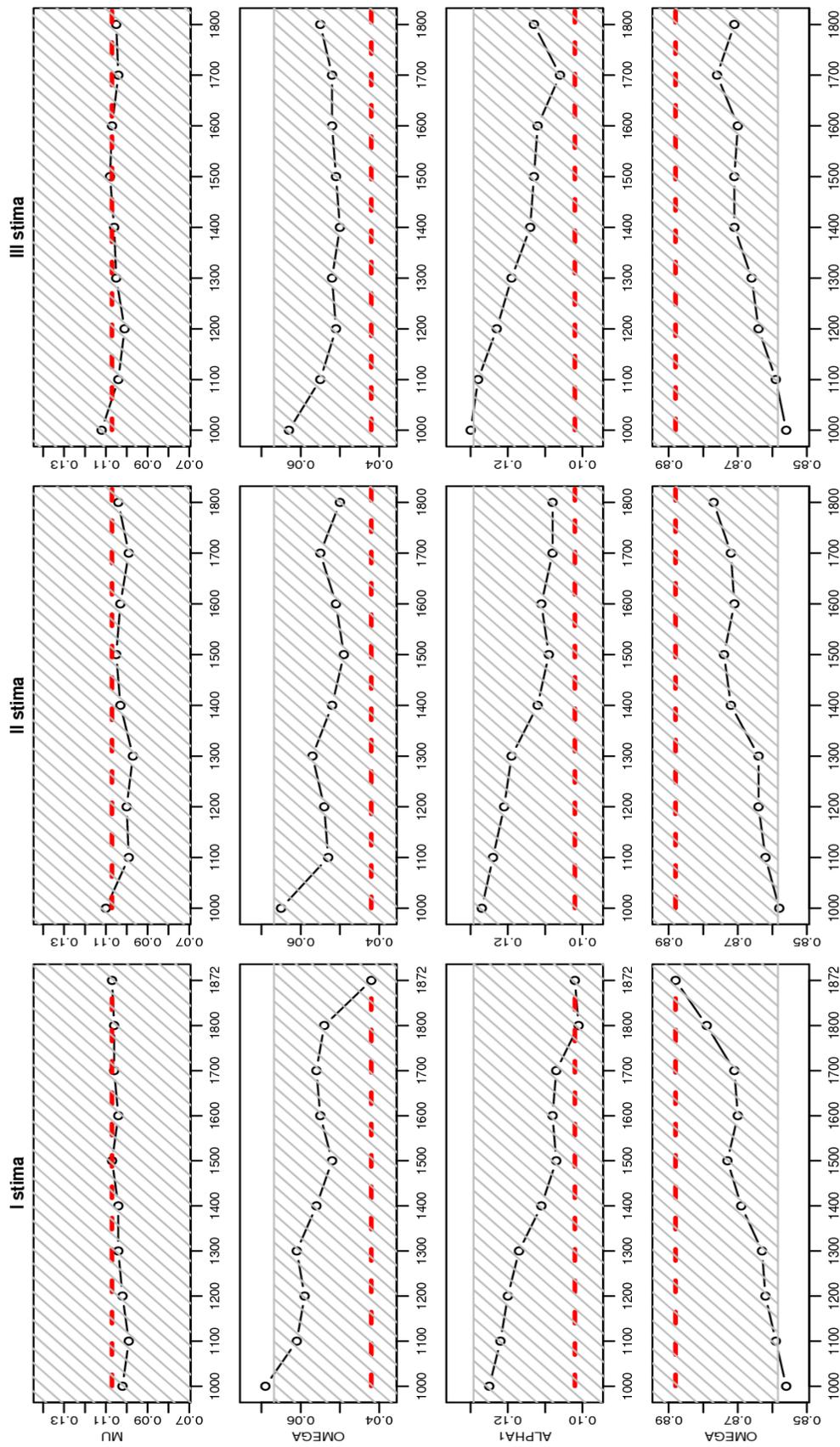


Figura B.16: Serie BAS con particolare attenzione alle tre stime eseguite con la finestra che slitta.

Serie_BMW : NORM stime_parti

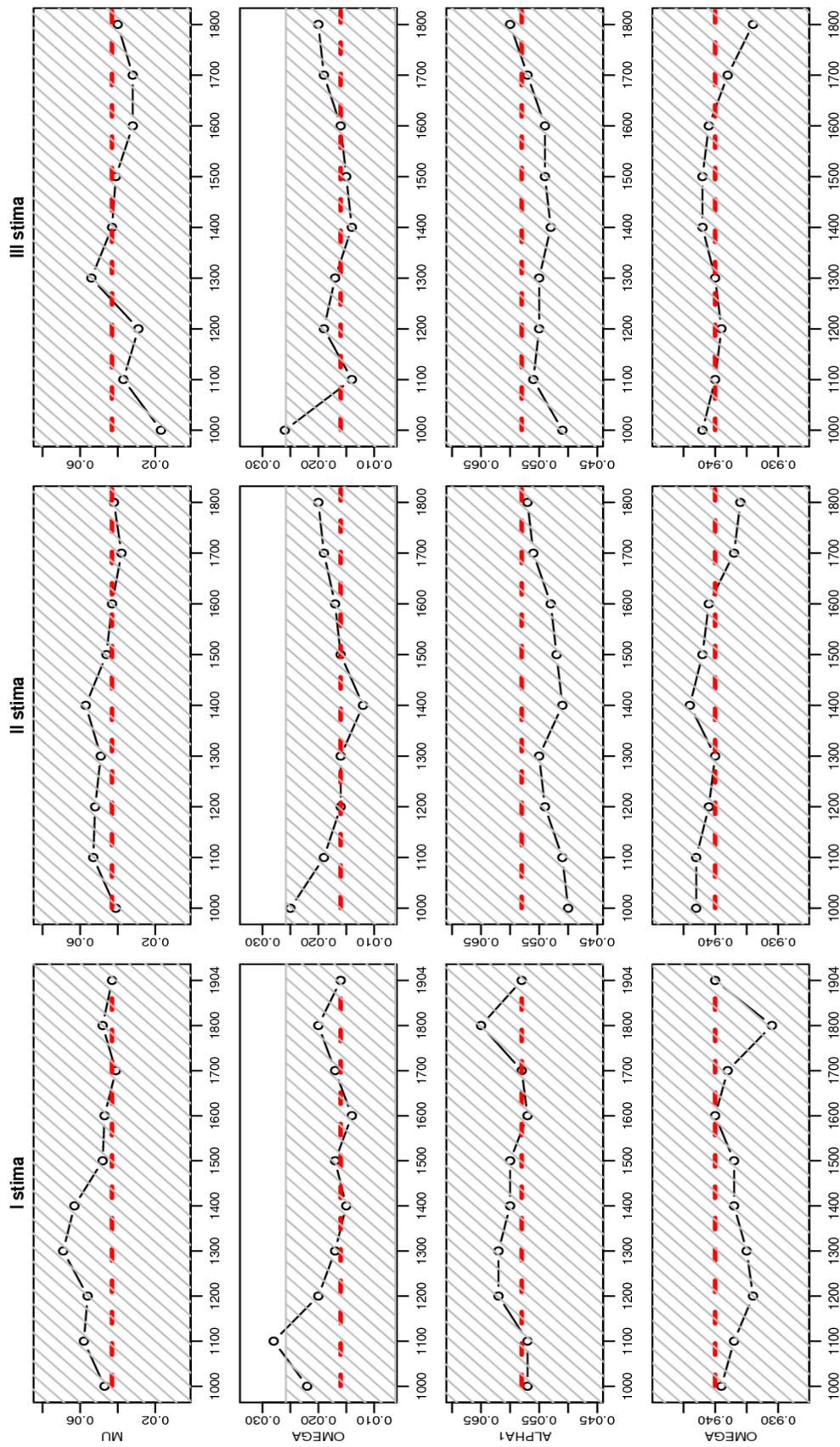


Figura B.17: Serie BMW con particolare attenzione alle tre stime eseguite con la finestra che slitta.

Serie_SDF : NORM stime_parti

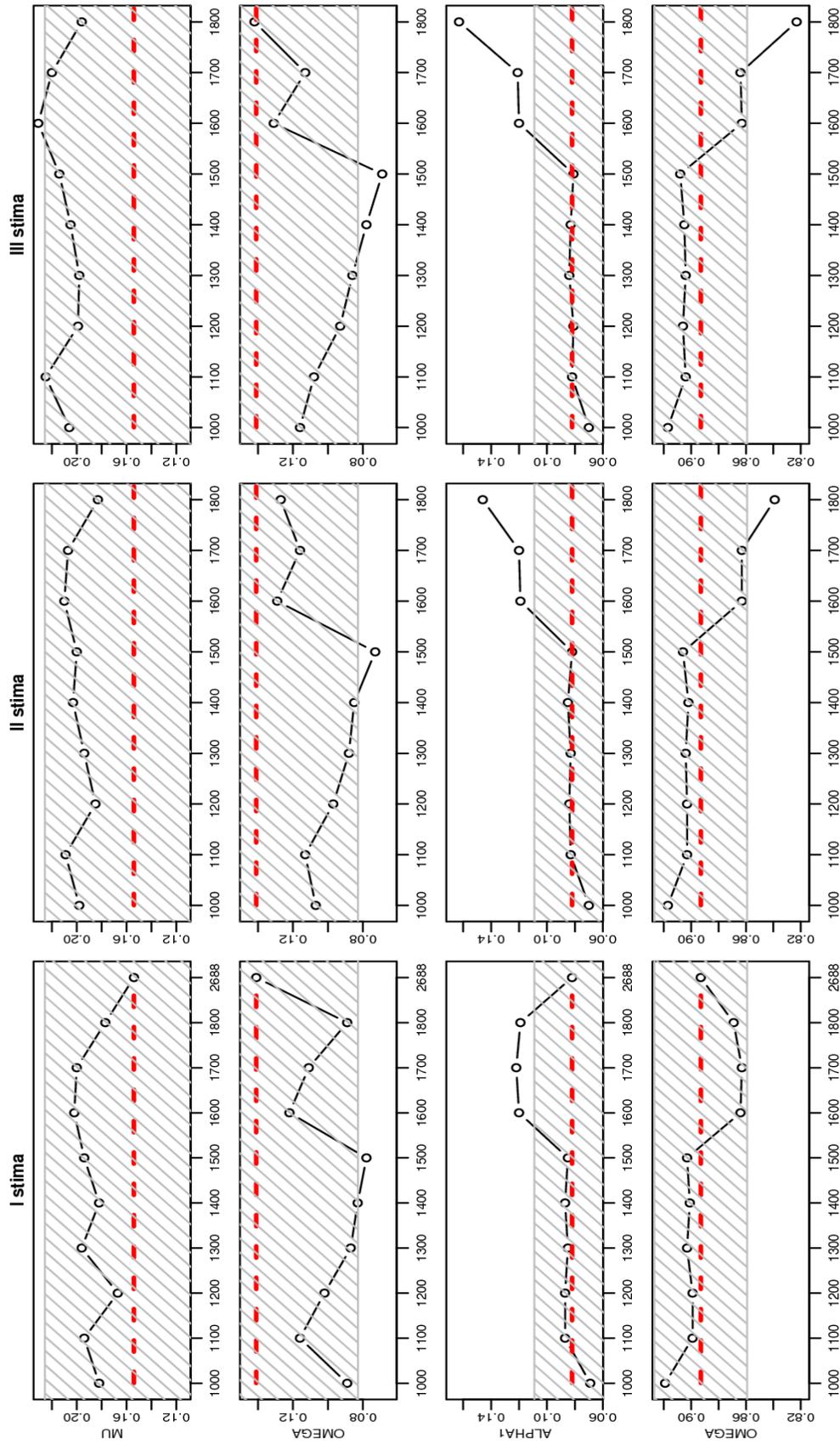


Figura B.18: Serie SDF con particolare attenzione alle tre stime eseguite con la finestra che slitta.

n	1872	Str. Err.	1700	1500	1300	1200	1150	1100	1050	1000	950	900	800
mu	0.107	0.0298	0.104	0.107	0.100	0.094	0.095	0.093	0.105	0.096	0.114	0.121	0.081
omega	0.042	0.0124	0.045	0.045	0.050	0.048	0.049	0.052	0.056	0.049	0.053	0.065	0.072
alpha1	0.102	0.0136	0.089	0.089	0.094	0.098	0.100	0.096	0.099	0.098	0.098	0.101	0.110
beta1	0.888	0.0148	0.893	0.892	0.886	0.886	0.884	0.885	0.882	0.888	0.883	0.877	0.875
logVer	-3413.234		-3032.330	-2691.158	-2405.385	-2249.272	-2164.195	-2092.442	-2021.371	-1934.244	-1847.915	-1783.709	-1649.510

Tabella B.1: Calcolo stime dei parametri modello GARCH con distribuzione NORMALE.

n	1872	Str. Err.	1700	1500	1300	1200	1150	1100	1050	1000	950	900	800
mu	0.323	0.0777	0.316	0.291	0.299	0.268	0.281	0.274	0.314	0.334	0.351	0.400	0.347
omega	0.043	0.0128	0.046	0.046	0.052	0.050	0.050	0.053	0.056	0.050	0.054	0.065	0.077
alpha1	0.107	0.0146	0.093	0.091	0.097	0.100	0.102	0.098	0.102	0.102	0.102	0.104	0.110
beta1	0.884	0.0157	0.890	0.889	0.883	0.884	0.882	0.884	0.880	0.885	0.880	0.875	0.874
alpha	-0.233	0.0792	-0.231	-0.200	-0.206	-0.178	-0.190	-0.179	-0.204	-0.235	-0.231	-0.258	-0.223
logVer	-3408.730		-3028.410	-2688.370	-2402.650	-2247.320	-2161.990	-2090.560	-2019.000	-1931.350	-1845.010	-1780.380	-1647.650

Tabella B.2: Calcolo stime dei parametri modello GARCH con distribuzione NORMALE ASIMMETRICA.

n	1872	Str. Err.	1700	1500	1300	1200	1150	1100	1050	1000	950	900	800
mu	0.102	0.0282	0.101	0.107	0.103	0.102	0.099	0.098	0.108	0.104	0.120	0.126	0.097
omega	0.037	0.0137	0.042	0.040	0.044	0.041	0.041	0.040	0.043	0.038	0.040	0.045	0.071
alpha1	0.099	0.0169	0.088	0.083	0.091	0.095	0.096	0.088	0.093	0.094	0.093	0.095	0.103
beta1	0.895	0.0179	0.897	0.899	0.892	0.892	0.891	0.897	0.893	0.897	0.892	0.887	0.879
nu	6.300	0.9450	6.631	6.631	6.574	6.557	6.250	6.342	6.031	5.881	6.189	6.213	6.110
logVer	-3380.310		-3001.330	-2661.450	-2378.184	-2224.600	-2138.400	-2066.290	-1994.750	-1910.280	-1823.450	-1759.870	-1633.070

Tabella B.3: Calcolo stime dei parametri modello GARCH con distribuzione T di STUDENT.

n	1872	Str. Err.	1700	1500	1300	1200	1150	1100	1050	1000	950	900	800
mu	0.261	0.0282	0.261	0.254	0.266	0.256	0.251	0.246	0.266	0.284	0.292	0.319	0.319
omega	0.038	0.0137	0.043	0.041	0.045	0.042	0.042	0.041	0.044	0.039	0.041	0.047	0.075
alpha1	0.102	0.0169	0.091	0.087	0.095	0.097	0.099	0.090	0.096	0.098	0.097	0.098	0.106
beta1	0.891	0.0179	0.893	0.896	0.889	0.890	0.898	0.895	0.890	0.893	0.889	0.883	0.876
alpha	-0.187	0.0775	-0.191	-0.175	-0.188	-0.177	-0.176	-0.166	-0.177	-0.203	-0.193	-0.206	-0.209
nu	6.491	0.9917	6.800	6.751	6.639	6.583	6.322	6.390	6.115	6.009	6.326	6.415	6.225
logVer	-3377.370		-2998.610	-2659.270	-2375.832	-2222.610	-2136.430	-2064.590	-1992.870	-1907.940	-1821.250	-1757.510	-1631.260

Tabella B.4: Calcolo stime dei parametri modello GARCH con distribuzione T di STUDENT ASIMMETRICA.

n	1872	Str. Err.	1700	1500	1300	1200	1150	1100	1050	1000	950	900	800
mu	0.095	0.0291	0.094	0.105	0.105	0.105	0.098	0.096	0.102	0.097	0.114	0.127	0.093
omega	0.038	0.0147	0.042	0.041	0.046	0.043	0.043	0.045	0.047	0.041	0.044	0.053	0.071
alpha1	0.100	0.0165	0.088	0.086	0.092	0.096	0.097	0.091	0.096	0.096	0.095	0.098	0.104
beta1	0.892	0.0177	0.895	0.896	0.889	0.889	0.889	0.892	0.889	0.893	0.889	0.883	0.878
nu	1.362	0.0623	1.368	1.356	1.350	1.351	1.324	1.315	1.306	1.301	1.307	1.299	1.325
logVer	-3378.090		-2999.370	-2659.450	-2376.733	-2223.460	-2136.680	-2064.220	-1993.290	-1908.310	-1821.920	-1757.890	-1630.830

Tabella B.5: Calcolo stime dei parametri modello GARCH con distribuzione di SUBBOTIN.

Bibliografia

Azzalini, A. (1985). A class of distributions which includes the normal ones. *Scand. J. Statist.*, 12, 171–178.

Azzalini, A. (1986). Further result on a class of distributions which includes the normal ones, si veda errata corrige. *Statistica*, 46, 199–208. Corrigendum: <http://azzalini.stat.unipd.it/SN/errata86.pdf>.

Azzalini, A. (2001). *Inferenza statistica. Una presentazione basata sul concetto di verosimiglianza*. Milano: Springer-Verlag Italia.

Azzalini, A. (2005). The skew-normal distribution and related multivariate families. *Scand. J. Statist.*, 32(2), 159–188.

Azzalini, A. & Capitanio, A. (1999). Statistical applications of the multivariate skew normal distribution. *J. R. Statist Soc. series B*, 61, 579–602.

Azzalini, A. & Capitanio, A. (2003). Distributions generated by perturbation of symmetry with emphasis on a multivariate skew t distribution. *J. R. Statist. Soc. B*, 65, Part 2, 367–389.

Azzalini, A. & Dalla Valle, A. (1996). The multivariate skew-normal distribution. *Biometrika*, 83, 715–726.

Azzalini, A. & Genton, M.G. (2008). Robust Likelihood Methods Based on the Skew- t and Related Distributions. *International Statistical Review*, 76, 1, 106–129.

Bai, J. & Ng, S. (2005). Test for Skewness, Kurtosis, and Normality for Time

Bibliografia

Series Data. *Journal of Business & Economic Statistics*, Vol. 23, N₀ 1, 49-60.

Bollerslev, T. (1986). Generalized Autoregressive Conditional Heteroscedasticity. *Journal of Econometrics*, 31, 307–327.

Bollerslev, T. (1987). Conditionally Heteroskedastic Time Series Model for Speculative Prices and Rates of Return. *Review of Economics and Statistics*, 69, 542-547.

Bortot, P. & Ventura L. & Salvan A. (2000). *Inferenza statistica: applicazioni con S-plus e R*. Padova: CEDAM.

Capobianco, R. (2000). Robustness aspects of the generalized normal distribution. *Quaderni di Statistica*, Vol. 2.

D'Elia, A. & Piccolo, D. (2004). *Statistica per le decisioni: test di autovalutazione*. Bologna: Il mulino.

Fernández, C. & Steel, M.F.J. (1998). On Bayesian Modelling of Fat Tails and Skewness. *J. Am. Stat. Ass.* 90, 1331 – 1340.

Gallo, G.M. & Pacini, B. (2002). *Metodi quantitativi per i mercati finanziari*. Roma: CAROCCI.

Iacus, M. R. & Masarotto, G. (2003). *Laboratorio di statistica con R*. Milano: McGraw-Hill.

Nelson, D.B. (1991). Conditional Heteroskedasticity in asset returns: A New Approach. *Econometria*, 59, 347-370.

Pace, L. & Salvan, A. (2001). *Introduzione alla statistica - II. Inferenza*,

Verosimiglianza, Modelli. Padova: CEDAM.

Tsay, R.S. (2002). Antologis of Finantial Time Series. *Wiley Series in Probability and Statistics*, New York, pp. XII + 448.

Venables, W. N. & Ripley, B. D. (1999). *Modern applied statistics with S-PLUS*. New York: Springer.

Zivot, E. & Wang. J. (2006). *Modelling Finantial Time Series with S-PLUS, Second Edition*. New York: Springer.