POLITECNICO DI TORINO

Corso di Laurea
in Matematica per l'Ingegneria

Tesi di Laurea

Stima delle Copule in Finanza: Analisi sul DAX

Relatori
Franco Pellerey
firma dei relatori

. . . . . .

Candidato
Andrea Rostagno
firma del candidato

. . . . . .

Anno Accademico 2024-2025

Sommario

Questa tesi esplora il ruolo delle copule nella modellazione della dipendenza tra variabili casuali, con particolare attenzione alle applicazioni in finanza. Dopo un'introduzione ai concetti fondamentali delle copule e alla loro teoria matematica, vengono analizzate diverse famiglie di copule, le loro proprietà e il loro utilizzo nella gestione del rischio finanziario. Successivamente, si discutono i metodi di stima dei parametri delle copule, con un focus su metodi classici e bayesiani, e si presenta un'implementazione pratica sui dati finanziari del DAX. Infine, vengono confrontati i risultati ottenuti con diverse copule e si discutono le implicazioni per la gestione del rischio e l'ottimizzazione di portafoglio.

Ringraziamenti

Da scrivere

Indice

Parte I

Fondamenti delle Copule

Capitolo 1

Introduzione

1.1 Definizione e motivazione dello studio

Le copule sono strumenti matematici che permettono di modellare e stimare la dipendenza tra diverse variabili casuali. Sono particolarmente utili in finanza, dove la dipendenza tra i rendimenti degli asset, i tassi di interesse e i tempi di default sono fattori cruciali per la valutazione del rischio e la determinazione del prezzo di strumenti finanziari complessi. L'importanza delle copule risiede nella loro capacità di separare la modellazione delle distribuzioni marginali delle singole variabili dalla modellazione della loro struttura di dipendenza.

In altre parole, invece di dover specificare una funzione di distribuzione congiunta per tutte le variabili, è possibile utilizzare una copula per combinare le distribuzioni marginali di ciascuna variabile in una distribuzione congiunta che rifletta la dipendenza desiderata. Questo approccio offre grande flessibilità nella modellazione, poiché consente di scegliere le distribuzioni marginali e la copula in modo indipendente, a seconda delle caratteristiche specifiche dei dati e del problema in esame.

Ad esempio, si potrebbe utilizzare una distribuzione t di Student per modellare i rendimenti degli indici azionari, che spesso presentano code più spesse rispetto alla distribuzione normale, e quindi utilizzare una copula di Gumbel per rappresentare la dipendenza asimmetrica tra i mercati, con una maggiore probabilità di movimenti congiunti al rialzo rispetto a quelli al ribasso.

La teoria delle copule si basa sul teorema di Sklar, che afferma che ogni funzione di distribuzione congiunta può essere espressa in termini di una copula e delle distribuzioni marginali delle variabili. Il teorema di Sklar garantisce l'esistenza e l'unicità della copula nel caso di variabili casuali continue.

Esistono diverse famiglie di copule, ciascuna con proprietà specifiche in termini di dipendenza di coda, simmetria e altre caratteristiche. Alcune delle famiglie di copule più utilizzate in finanza includono la copula gaussiana, la copula t di Student, le copule Archimedee (come la copula di Gumbel, la copula di Clayton e la copula di Frank) e la copula di Marshall-Olkin. La scelta della copula più adatta dipende dalla natura del problema e dalle caratteristiche della dipendenza che si desidera modellare.

Ad esempio, la copula t di Student é spesso preferita alla copula gaussiana quando si vogliono modellare dipendenze di coda più elevate, mentre le copule Archimedee consentono di modellare diversi tipi di dipendenza asimmetrica.

Le copule trovano applicazione in diversi ambiti della finanza, tra cui:

- **Pricing di opzioni multivariate e altri derivati**: le copule possono essere utilizzate per modellare la dipendenza tra i sottostanti di un'opzione basket, un'opzione rainbow o altri derivati multi-asset, consentendo una valutazione più accurata del prezzo di questi strumenti.
- **Gestione del rischio**: le copule sono ampiamente utilizzate nella modellazione del rischio di credito, dove consentono di stimare la probabilità di default congiunta di diverse attività o controparti. Le copule sono anche utilizzate nella stima del Value at Risk (VaR) di portafogli contenenti attività con distribuzioni non normali e dipendenze complesse.
- **Calibrazione e simulazione**: la flessibilità delle copule consente di calibrare i modelli ai dati di mercato in modo efficiente e di simulare scenari di mercato realistici che tengano conto della dipendenza tra le variabili.

In sintesi, le copule rappresentano uno strumento matematico versatile e potente per la modellazione della dipendenza in finanza, con un ampio spettro di applicazioni pratiche nella valutazione del rischio, nel pricing di derivati e nella gestione del portafogli.

Capitolo 2

Fondamenti Matematici delle Copule

2.1 Definizione di copula e Teorema di Sklar

Le copule sono strumenti matematici che permettono di modellare e rappresentare la dipendenza tra variabili casuali. A differenza di misure di dipendenza tradizionali come la correlazione lineare, le copule catturano la dipendenza in modo più completo, includendo la dipendenza nelle code delle distribuzioni e non limitandosi a relazioni lineari.

Ecco una spiegazione delle formule e delle proprietà chiave:

2.1.1 Definizione di Copula

Una d-copula è una funzione $C : [0,1]^d \to [0,1]$, dove $d \geq 2$ (numero di variabili; nelle proprietà seguenti consideriamo le copule bivariate), che soddisfa le seguenti proprietà:

1. **Groundedness:**
$$C(u,0) = C(0,v) = 0, \quad \forall u, v \in [0,1]^2$$
Ciò significa che la copula è zero se una delle variabili è zero.

2. **Marginalità:**
$$C(u,1) = u, \quad C(1,v) = v, \quad \forall u, v \in [0,1]^2$$
Questa proprietà assicura che la copula sia coerente con le distribuzioni marginali, ovvero che quando una delle variabili assume il suo valore massimo, la copula coincida con la funzione di ripartizione dell'altra variabile.

3. **2-crescita (o 2-increasing):**
$$C(u_2, v_2) - C(u_2, v_1) - C(u_1, v_2) + C(u_1, v_1) \geq 0, \quad \forall u_1 \leq u_2, v_1 \leq v_2 \in [0,1]^2$$
Questa proprietà assicura che la copula sia non decrescente in entrambe le variabili, il che è necessario affinché la copula rappresenti una dipendenza positiva o negativa tra le variabili.

2.1.2 Teorema di Sklar

Questo teorema, centrale nella teoria delle copule, stabilisce un legame tra le copule e le funzioni di distribuzione congiunta.

In breve, il teorema afferma che: Data una funzione di distribuzione congiunta $F(x, y)$ con marginali $F_1(x)$ e $F_2(y)$, esiste una copula C tale che:

$$F(x, y) = C(F_1(x), F_2(y))$$

Inoltre, se $F_1(x)$ e $F_2(y)$ sono continue, allora la copula C è unica.

Conseguenze del Teorema di Sklar

- **Costruzione di modelli di dipendenza:** Permette di costruire una funzione di distribuzione congiunta a partire da distribuzioni marginali arbitrarie e da una copula che ne modella la dipendenza. Questa proprietà è particolarmente utile per modellare dati reali, dove spesso si conoscono le distribuzioni marginali ma non la struttura di dipendenza.

- **Separazione tra marginali e dipendenza:** Mette in luce come la struttura di dipendenza tra le variabili sia completamente catturata dalla copula, indipendentemente dalle distribuzioni marginali.

2.2 Famiglie principali di copule (Gaussiane, t-Student, Archimedee)

2.2.1 Famiglie di copule

Esistono diverse famiglie di copule, classificate in base alla loro struttura o ai metodi utilizzati per la loro costruzione. Di seguito, vengono elencate alcune delle principali famiglie:

- **Fréchet-Hoeffding:** Questa famiglia include le copule che rappresentano i limiti inferiore (W) e superiore (M) della dipendenza tra due variabili casuali. La copula W rappresenta la perfetta dipendenza negativa, mentre la copula M rappresenta la perfetta dipendenza positiva.

- **Cuadras-Augé:** Questa famiglia di copule è costruita come una media geometrica ponderata delle copule M e P, dove P rappresenta l'indipendenza tra le variabili.

- **Marshall-Olkin:** Questa famiglia di copule è spesso utilizzata per modellare la dipendenza tra variabili casuali che rappresentano tempi di vita.

- **Archimedee:** Queste copule sono generate da una funzione detta "generatore". Le copule Archimedee sono popolari per la loro flessibilità e la relativa facilità di utilizzo.

2.2.2 Proprietà delle copule

Le copule possiedono diverse proprietà che le rendono utili per la modellazione della dipendenza. Alcune di queste proprietà sono:

- **Invarianza rispetto a trasformazioni monotone crescenti:** Le copule sono invarianti rispetto a trasformazioni strettamente crescenti delle variabili marginali.
- **Misure di concordanza:** Diverse misure di concordanza come la rho di Spearman e la tau di Kendall possono essere espresse in termini di copule.
- **Dipendenza di coda:** Le copule possono catturare la dipendenza tra le code delle distribuzioni marginali, ovvero la tendenza delle variabili ad assumere valori estremi congiuntamente.

Possiamo quindi affermare che le copule offrono un approccio potente e flessibile per la modellazione della dipendenza tra variabili casuali. La loro capacità di separare la struttura di dipendenza dalle distribuzioni marginali, la loro invarianza rispetto a trasformazioni monotone crescenti e la loro capacità di catturare la dipendenza di coda le rendono strumenti preziosi in molte applicazioni pratiche.

Parte II

Applicazioni Finanziarie delle Copule

Capitolo 3

Copule e Gestione del Rischio Finanziario

3.1 Superamento della correlazione lineare nelle distribuzioni non normali

L'assunzione di normalità dei rendimenti, tipico di modelli come quello di Black-Scholes, è spesso disatteso nei mercati finanziari. Le serie storiche di strumenti come azioni ed obbligazioni dimostrano la presenza di code più pesanti rispetto a quanto previsto dalla distribuzione normale e la diffusione di prodotti derivati con payoff non lineari amplifica ulteriormente questo fenomeno.

In questo contesto, la **correlazione lineare**, misurata ad esempio con il coefficiente di Pearson, si dimostra uno strumento limitato. Essa cattura solo le dipendenze lineari tra le variabili, mentre le relazioni tra gli asset finanziari possono assumere forme ben più complesse. La correlazione lineare è efficace solo quando le variabili sono legate da relazioni lineari. Tuttavia, in presenza di legami non lineari, la correlazione lineare potrebbe essere fuorviante. Ad esempio, una variabile con distribuzione chi-quadrato è perfettamente correlata al suo quadrato, che ha distribuzione normale, ma la correlazione lineare non sarebbe in grado di rappresentare correttamente questa relazione.

3.1.1 Coefficiente di Pearson

Il coefficiente di correlazione lineare, noto anche come correlazione di Pearson, è una misura della dipendenza lineare tra due variabili casuali che assumono valori reali e che hanno varianza finita. È definito come la covarianza delle due variabili divisa per il prodotto delle loro deviazioni standard. Formalmente, per due variabili casuali non degeneri X e Y appartenenti a L^2, il coefficiente di correlazione lineare ρ_{XY} è:

$$\rho_{XY} = \frac{\operatorname{cov}(X, Y)}{\sqrt{\operatorname{var}(X)\operatorname{var}(Y)}}$$

Il coefficiente di correlazione di Pearson assume valori compresi tra -1 e +1, dove:

- +1 indica una perfetta correlazione lineare positiva: all'aumentare di una variabile, l'altra aumenta in modo perfettamente lineare.
- -1 indica una perfetta correlazione lineare negativa: all'aumentare di una variabile, l'altra diminuisce in modo perfettamente lineare.
- 0 indica l'assenza di correlazione lineare: non c'è una relazione lineare tra le due variabili.

È importante sottolineare che il coefficiente di correlazione di Pearson misura solo la dipendenza lineare. Due variabili possono essere fortemente dipendenti in modo non lineare e avere comunque un coefficiente di correlazione di Pearson pari a zero.

3.1.2 Vantaggi delle Copule rispetto alla Correlazione Lineare

Le **copule**, invece, offrono un approccio più flessibile per modellare la dipendenza tra variabili casuali, anche in presenza di distribuzioni non normali. Il vantaggio principale delle copule risiede nella loro capacità di separare la struttura di dipendenza dalle distribuzioni marginali. Questo permette di combinare diverse distribuzioni marginali, capaci di cogliere la non-normalità dei rendimenti (ad esempio la distribuzione t di Student o distribuzioni asimmetriche), con una vasta gamma di copule che descrivono la struttura di dipendenza.

Il **Teorema di Sklar**, fondamento della teoria delle copule, afferma che ogni funzione di distribuzione congiunta può essere espressa in termini di distribuzioni marginali e di una copula. Questa proprietà permette di costruire modelli di dipendenza altamente flessibili, adatti a rappresentare le complesse relazioni tra gli asset finanziari. Ad esempio, è possibile utilizzare una copula gaussiana per modellare la struttura di dipendenza, pur mantenendo distribuzioni marginali non gaussiane per i singoli asset. In questo modo, si ottiene un modello in grado di catturare sia la non-normalità dei rendimenti sia le strutture di dipendenza tra gli stessi.

In definitiva, le copule rappresentano uno strumento più completo e affidabile rispetto alla correlazione lineare per modellare le dipendenze tra variabili casuali, soprattutto in presenza di distribuzioni non normali. La loro flessibilità e capacità di rappresentare accuratamente le complesse relazioni tra gli asset le rendono essenziali per una corretta valutazione del rischio, un pricing accurato e una migliore comprensione delle dinamiche dei mercati finanziari.

3.2 Dipendenza di coda e impatti su Value-at-Risk (VaR) e Expected Shortfall

La dipendenza di coda si riferisce alla tendenza di due o più variabili casuali a muoversi insieme in modo più estremo nelle code delle loro distribuzioni, rispetto a quanto previsto da una distribuzione normale con la stessa correlazione lineare. In altre parole, la dipendenza di coda misura la probabilità che si verifichino eventi estremi congiuntamente.

Sappiamo che la non normalità a livello univariato è associata al cosiddetto problema della *fat-tail*. In un contesto multivariato, il problema della *fat-tail* può essere riferito sia alle distribuzioni marginali univariate che alle distribuzioni congiunte di probabilità di grandi movimenti di mercato. Questo concetto è chiamato **tail dependence**. L'uso di funzioni copula ci permette di modellare separatamente queste due caratteristiche. Per rappresentare la dipendenza dalla coda consideriamo la probabilità che un evento con probabilità inferiore a v si verifichi nella prima variabile, dato che un evento con probabilità inferiore a v si verifica nella seconda. In concreto, ci chiediamo quale sia la probabilità di osservare, ad esempio, un crollo con probabilità inferiore di $v = 1\%$ nell'indice Nikkei 225, dato che nell'indice S&P 500 si è verificato un crollo con probabilità inferiore all'1%. Si ha:

$$\lambda(v) \equiv \Pr(Q_{NKY} \leq v \mid Q_{SP} \leq v)$$

$$= \frac{\Pr(Q_{NKY} \leq v, Q_{SP} \leq v)}{\Pr(Q_{SP} \leq v)}$$

$$= \frac{C(v, v)}{v}$$

3.2.1 Tipi di dipendenza di coda

Dopo che abbiamo calcolato il nostro $\lambda(v)$, possiamo dividere la dipendenza di coda in due tipi principali:

- **Dipendenza di coda inferiore**: misura la probabilità che entrambe le variabili assumano valori estremamente bassi contemporaneamente.

$$\lambda_L \equiv \lim_{v \to 0^+} \frac{C(v, v)}{v}$$

- **Dipendenza di coda superiore**: misura la probabilità che entrambe le variabili assumano valori estremamente alti contemporaneamente.

$$\lambda_U = \lim_{v \to 1^-} \lambda_v \equiv \lim_{v \to 1^-} \frac{\Pr(Q_{NKY} > v, Q_{SP} > v)}{\Pr(Q_{SP} > v)}$$

$$= \lim_{v \to 1^-} \frac{1 - 2v + C(v, v)}{1 - v}$$

3.2.2 Importanza nel VaR e nell'Expected Shortfall

La dipendenza di coda ha un impatto significativo sul calcolo del **VaR** e dell'**Expected Shortfall**, due misure di rischio ampiamente utilizzate nella gestione del rischio finanziario.

- **VaR (Value at Risk)**: rappresenta la perdita massima stimata che un portafoglio potrebbe subire in un determinato orizzonte temporale e con un dato livello di confidenza.
- **Expected Shortfall**: rappresenta la perdita media attesa in caso di superamento del VaR.

In presenza di dipendenza di coda, il VaR e l'Expected Shortfall calcolati assumendo una distribuzione normale tendono a sottostimare il rischio effettivo del portafoglio. Questo perché la distribuzione normale non riesce a catturare adeguatamente la probabilità di eventi estremi congiunti. Mentre utilizzando le copule per modellare la dipendenza tra gli asset di un portafoglio, è possibile ottenere una stima più accurata del VaR e dell'Expected Shortfall, tenendo conto della probabilità di eventi estremi congiunti. Quindi grazie alle copule è possibile una misura più accurata del rischio di portafoglio e si possono prendere decisioni più consapevoli.

3.3 Pricing di derivati multivariati con copule e gestione del rischio

3.3.1 Tariffare le opzioni

La valutazione di opzioni multivariate, come le opzioni basket o rainbow, che dipendono da più attività sottostanti, rappresenta una sfida significativa in finanza. Le copule forniscono un potente strumento per affrontare questo problema.

- In sostanza, una copula viene utilizzata per costruire la distribuzione congiunta dei prezzi delle attività sottostanti alla scadenza.
- Questa distribuzione viene quindi utilizzata per calcolare il valore atteso del payoff dell'opzione in base a tutti i possibili risultati dei prezzi delle attività sottostanti.
- Attualizzando questo valore atteso al tasso privo di rischio, si ottiene il prezzo dell'opzione.

Nei mercati incompleti, dove non esiste una misura di probabilità unica priva di arbitraggio, le copule sono fondamentali per derivare strategie di super-replicazione. Queste strategie mirano a creare un portafoglio di attività negoziabili che replichi il payoff dell'opzione in tutte le possibili situazioni future, garantendo così l'assenza di opportunità di arbitraggio. Le copule consentono di determinare i limiti superiori e inferiori del prezzo dell'opzione in base alle diverse ipotesi sulla struttura di dipendenza tra le attività sottostanti. Mostriamo ora degli esempi:

- **Opzioni arcobaleno**: queste opzioni, che dipendono dal minimo o dal massimo di un paniere di attività, possono essere valutate utilizzando le copule per catturare la dipendenza tra i rendimenti delle attività. Le fonti dimostrano come le copule possano essere utilizzate per derivare strategie di super-replicazione per le opzioni arcobaleno, fornendo limiti superiori e inferiori al prezzo.
- **Opzioni barriera**: per le opzioni in cui l'esercizio è condizionato al fatto che il prezzo dell'attività sottostante raggiunga o meno una determinata barriera, le copule possono essere utilizzate per modellare la dipendenza tra il processo del prezzo dell'attività e l'evento di attivazione della barriera.

3.3.2 Gestione dei rischi

Le copule possono essere utilizzate per modellare la dipendenza tra diversi tipi di rischio, come rischio di mercato, rischio di credito e rischio operativo. Ciò è particolarmente utile per le istituzioni finanziarie che sono esposte a più tipi di rischio, in quanto consente loro di valutare il rischio complessivo a cui sono esposte. Ad esempio, una banca può utilizzare le copule per modellare la dipendenza tra le insolvenze sui prestiti e i movimenti dei tassi di interesse, consentendo loro di valutare il rischio del proprio portafoglio prestiti in diversi scenari economici.

In particolare, nella gestione del rischio di credito, le copule vengono utilizzate nella valutazione di strumenti di debito strutturati come le obbligazioni garantite da crediti (CDO). Le CDO sono obbligazioni garantite da un pool di attività sottostanti, come mutui o prestiti alle imprese. Il rischio di credito di una CDO dipende dalla dipendenza tra le insolvenze delle attività sottostanti. Le copule forniscono un modo flessibile per modellare questa dipendenza, consentendo agli investitori di valutare il rischio e il rendimento delle CDO in modo più accurato.

3.4 Modelli di copula

3.4.1 Tipi di modelli di copula

- **Copula gaussiana**: descrive la dipendenza tra variabili casuali utilizzando la distribuzione normale multivariata. Non è in grado di catturare la dipendenza dalla coda. È definita come la funzione di distribuzione congiunta di un vettore normale multivariato standard, dove ogni variabile marginale è stata trasformata nella sua forma univariate standard utilizzando la funzione di distribuzione normale standard inversa.

$$C_R^{Ga}(u,v) = \Phi_R(\Phi^{-1}(u), \Phi^{-1}(v))$$

$$= \int_{-\infty}^{\Phi^{-1}(u)} \int_{-\infty}^{\Phi^{-1}(v)} \frac{1}{2\pi\sqrt{1-\rho_{XY}^2}} \exp\left(\frac{2\rho_{XY}st - s^2 - t^2}{2(1-\rho_{XY}^2)}\right) ds\, dt$$

- **Copula t di Student**: Questa copula può catturare la dipendenza dalla coda e viene spesso utilizzata per modellare i rendimenti degli asset che mostrano code pesanti. La copula t di Student bivariata è data dalla seguente formula:

$$C_t(u,v) = \int_{-\infty}^{t_\nu^{-1}(u)} \int_{-\infty}^{t_\nu^{-1}(v)} \frac{\Gamma\left(\frac{\nu+2}{2}\right)}{\Gamma\left(\frac{\nu}{2}\right)\pi\nu\sqrt{1-\rho^2}} \left(1 + \frac{x^2 - 2\rho xy + y^2}{\nu(1-\rho^2)}\right)^{-\frac{\nu+2}{2}} dx\, dy$$

  dove t_ν^{-1} è l'inversa della funzione di distribuzione t di Student univariata con ν gradi di libertà e ρ è il coefficiente di correlazione.

- **Copula di Clayton**: questa copula mostra una forte dipendenza dalla coda inferiore, il che significa che le variabili hanno maggiori probabilità di assumere insieme valori estremi bassi. È esaustiva e fornisce la copula del prodotto se $\alpha = 0$, il limite inferiore di Fréchet $\max(v+z-1, 0)$ quando $\alpha = -1$ e quello superiore per $\alpha \to +\infty$. È definita dalla seguente formula:

$$C(v,z) = \max[(v^{-\alpha} + z^{-\alpha} - 1)^{-\frac{1}{\alpha}}, 0]$$

- **Copula di Gumbel**: questa copula mostra una forte dipendenza dalla coda superiore, indicando che le variabili hanno maggiori probabilità di assumere insieme valori estremi elevati. Fornisce la copula del prodotto se $\alpha = 1$ e il limite superiore di Fréchet $\min(v, z)$ per $\alpha \to +\infty$. È definita dalla seguente formula:

$$C(v, z) = \exp\left(-[(-\ln v)^{\alpha} + (-\ln z)^{\alpha}]^{\frac{1}{\alpha}}\right)$$

- **Copula di Frank**: questa copula è una copula simmetrica che può catturare sia la dipendenza dalla coda superiore che quella dalla coda inferiore. Si riduce alla copula del prodotto se $\alpha = 0$ e raggiunge i limiti inferiori e superiori di Fréchet rispettivamente per $\alpha \to -\infty$ e $\alpha \to +\infty$. È definita dalla seguente formula:

$$C(v, z) = -\frac{1}{\alpha} \ln\left(1 + \frac{(e^{-\alpha v} - 1)(e^{-\alpha z} - 1)}{e^{-\alpha} - 1}\right)$$

Parte III

Preparazione dei Dati e Assunzioni

Capitolo 4

Preparazione dei Dati per la Modellazione con Copule

4.1 Struttura e caratteristiche dei dati

Il seguente dataset contiene dati storici sul DAX, che rappresentano prezzi o indici di mercato rilevanti. La raccolta dati fornisce i valori di Open, High, Low, Close partendo dal giorno 02/01/2020 ore 01:15:00 e terminando con il giorno 03/03/2022 ore 09:14:00. La raccolta e la gestione adeguata di tali dati sono fondamentali per analizzare le dipendenze tra strumenti finanziari tramite modelli di copula. Tramite questi dati calcoleremo i rendimenti giornalieri, un passaggio necessario per la modellazione delle dipendenze.

DateTime	**Open**	**High**	**Low**	**Close**
02/01/2020 01:15:00 +01:00	13174	13194.5	13171.5	13177.5
02/01/2020 01:16:00 +01:00	13177	13185	13177	13180.5
02/01/2020 01:17:00 +01:00	13180.5	13183	13179	13181.5
02/01/2020 01:18:00 +01:00	13181.5	13182	13180.5	13182
02/01/2020 01:19:00 +01:00	13182	13183	13180.5	13181.5
...	...	...	...	...
03/03/2022 09:10:00 +01:00	14019	14031	14010	14013
03/03/2022 09:11:00 +01:00	14013	14019	14010	14000
03/03/2022 09:12:00 +01:00	13999	14013	13996	14006
03/03/2022 09:13:00 +01:00	14007	14015	13995	14009
03/03/2022 09:14:00 +01:00	14009	14020	14009	14020

Tabella 4.1. 616397 osservazioni raccolte di open, high, low and close

4.2 Pulizia e preprocessing

I dati finanziari spesso includono anomalie come valori mancanti o outlier che devono essere gestiti prima dell'analisi. Saranno implementate le seguenti tecniche di pulizia dei dati:

- **Gestione dei valori mancanti**: rimuovere eventuali righe con valori mancanti per evitare distorsioni.
- **Gestione degli outlier**: utilizzare tecniche di filtraggio per identificare ed eliminare gli outlier, assicurando che l'analisi si concentri sui valori centrali più rappresentativi.

Outlier: nel contesto dei dati azionari, un outlier (o valore anomalo) è un'osservazione che si discosta significativamente dalla norma o dalla tendenza generale del dataset. In altre parole, si tratta di un dato che è molto diverso rispetto agli altri valori presenti nel campione.

Il motivo per cui vanno eliminati è che possono distorcere le analisi statistiche, poiché influenzano la media, la deviazione standard e altre misure di dispersione dei dati.

Ecco alcune righe di codice utilizzate per "pulire" i dati:

```
import pandas as pd

# Load the data
data = pd.read_csv('DAX_3Y-1M.csv',index_col='DateTime',
parse_dates=True)

# Drop rows with missing values
data=data.dropna()

# Verifica e gestione degli outlier tramite interquartile
range (IQR)
Q1 = data.quantile(0.25)
Q3 = data.quantile(0.75)
IQR = Q3 - Q1
filtered_data = data[~((data < (Q1 - 1.5 * IQR)) |
(data > (Q3 + 1.5 * IQR))).any(axis=1)]
```

Circa 6000 righe di dati sono state eliminate da questo processo.

4.3 Trasformazione dei dati

Dopo la pulizia, è necessario trasformare i dati per ottenere una scala uniforme. Poiché i modelli di copula richiedono margini uniformi, trasformeremo i dati in rendimenti logaritmici per ottenere stazionarietà e calcoleremo i punteggi standardizzati:

Ecco alcune righe di codice utilizzate per "trasformare" i dati:

```
import numpy as np

# Calcolo dei rendimenti logaritmici
log_returns = np.log(filtered_data / filtered_data.shift(1)).dropna()

# Standardizzazione dei dati
# (sottraggo la media e divido per la deviazione standard)
standardized_data = (log_returns - log_returns.mean()) / log_returns.std()
```

DateTime	**Open**	**High**	**Low**	**Close**
02/01/2020 01:16:00 +01:00	0.513283	-1.507703	0.853297	0.498567
02/01/2020 01:17:00 +01:00	0.598720	-0.555875	0.310070	0.166015
02/01/2020 01:18:00 +01:00	0.170871	0.079182	0.154914	0.028292
02/01/2020 01:19:00 +01:00	0.085317	0.158563	0.077349	-0.083337
02/01/2020 01:20:00 +01:00	-0.085772	0.237926	0.154896	0.498415
...	...	...	...	...
03/03/2022 09:10:00 +01:00	-1.447601	-0.000208	-0.437935	-1.250808
03/03/2022 09:11:00 +01:00	-0.965660	-1.791008	-1.459979	-2.033947
03/03/2022 09:12:00 +01:00	-2.254511	-0.149511	-0.146421	0.938266
03/03/2022 09:13:00 +01:00	1.288211	-0.448180	-0.584449	0.469065
03/03/2022 09:14:00 +01:00	0.321767	0.746358	2.043913	1.719639

Tabella 4.2. Rendimenti logaritmici standardizzati

4.4 Normalizzazione

Per applicare correttamente i modelli di copula, i dati devono essere trasformati in una distribuzione uniforme sull'intervallo [0,1]. Questo passaggio permette ai dati di adattarsi meglio alla funzione di copula che verrà utilizzata per modellare le dipendenze:

```
from scipy.stats import norm

# Normalizzazione tramite la funzione di distribuzione cumulativa (CDF)
uniform_data = norm.cdf(standardized_data)
```

Open	**High**	**Low**	**Close**
0.69612324	0.06581532	0.80325258	0.69095767
0.7253202	0.28914825	0.6217461	0.56592753
0.56783755	0.53155597	0.56155544	0.53303114
...	...	...	...
0.01288201	0.44057533	0.44186549	0.82604578
0.90116376	0.32701177	0.279459	0.68048837
0.62618555	0.7722743	0.97951894	0.95725093

Tabella 4.3. Dati uniformati all'intervallo [0,1]

4.5 Assunzioni nei modelli di copula

Per l'uso corretto dei modelli di copula, è essenziale discutere alcune assunzioni chiave:

- **Uniformità marginale**: l'assunzione primaria nella modellazione delle copule è che le variabili marginali abbiano distribuzioni uniformi. Abbiamo utilizzato la funzione di distribuzione cumulativa per garantire questa uniformità.
- **Struttura di dipendenza**: i modelli di copula modellano la struttura di dipendenza tra le variabili senza fare ipotesi sui margini. Ciò significa che, dopo la trasformazione, possiamo utilizzare diversi tipi di copule per analizzare come i vari strumenti finanziari si muovono insieme.
- **Stazionarietà**: è importante che i dati siano stazionari, ovvero che le loro proprietà statistiche (come media e varianza) siano costanti nel tempo. Abbiamo utilizzato la differenziazione logaritmica per rendere i dati stazionari.
- **Normalità**: per l'utilizzo di una copula Gaussiana, i margini devono approssimare la normalità. Sebbene non sia strettamente necessario per altre copule come la t-Copula, una trasformazione per avvicinarsi alla normalità può essere utile per semplificare l'analisi.

4.6 Distribuzione dei rendimenti

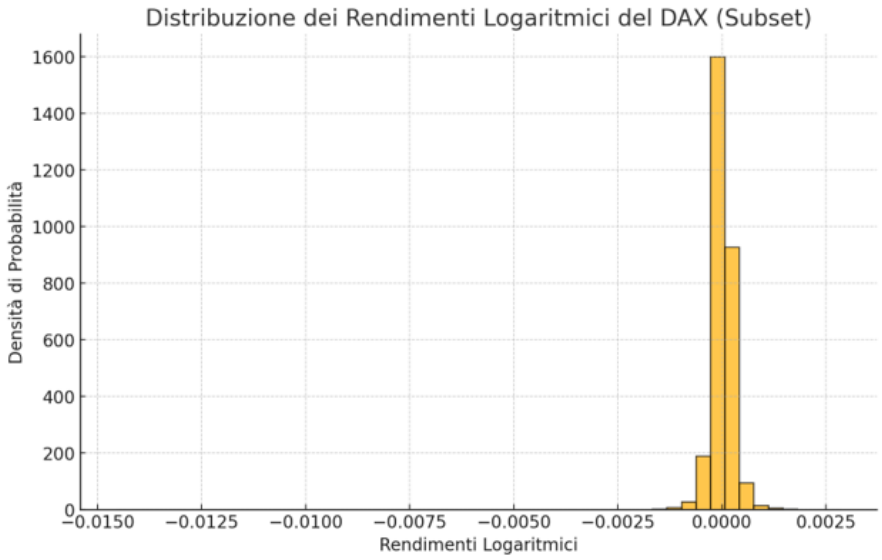

Figura 4.1. Distribuzione dei Rendimenti Logaritmici del DAX (Subset)

Ecco la distribuzione dei rendimenti logaritmici del DAX, calcolata su un subset dei dati disponibili. La distribuzione mostra la tipica forma a campana, con alcune code più pesanti, suggerendo la presenza di eventi estremi più frequenti rispetto a una normale distribuzione Gaussiana. Questa caratteristica supporta l'idea di utilizzare copule come la Student-t, che meglio cattura queste dipendenze nelle code.

Di seguito la matrice di correlazione tra i rendimenti logaritmici delle diverse colonne di prezzo (Open, High, Low, Close) sempre del subset.

	Open	**High**	**Low**	**Close**
Open	1.0000	0.4716	0.4890	0.3141
High	0.4716	1.0000	0.3353	0.5610
Low	0.4890	0.3353	1.0000	0.5142
Close	0.3141	0.5610	0.5142	1.0000

Tabella 4.4. Matrice di correlazione tra Open, High, Low e Close

4.7 Assunzioni generali

Considerando i dati analizzati e i risultati ottenuti dalle elaborazioni, possiamo formulare le seguenti assunzioni per l'applicazione dei modelli di copula:

1. **Uniformità marginale**
   Una delle assunzioni principali per applicare i modelli di copula è che le variabili marginali siano uniformi. Nel nostro caso, i rendimenti logaritmici calcolati per le diverse variabili (Open, High, Low, Close) sono stati trasformati in modo tale da poter essere considerati approssimativamente stazionari, ma non sono ancora stati resi uniformi. Per l'applicazione delle copule, sarà necessario trasformare i rendimenti normalizzati in una distribuzione uniforme sull'intervallo [0,1], ad esempio usando la funzione di distribuzione cumulativa empirica. Questo passaggio garantisce che le dipendenze tra variabili siano modellate correttamente senza distorsioni derivanti dalle distribuzioni marginali.

2. **Stazionarietà dei dati**
   Per applicare correttamente i modelli di copula, è necessario che le serie temporali siano stazionarie, ovvero che le proprietà statistiche come media e varianza siano costanti nel tempo. Abbiamo trasformato i prezzi in rendimenti logaritmici per ottenere una serie più stazionaria rispetto ai dati originali di prezzo. Tuttavia, è possibile che ci siano ancora componenti non stazionarie, come trend residui o stagionalità, che potrebbero influire sui risultati. La verifica e il trattamento di eventuali residui non stazionari sono fondamentali per garantire la validità dei modelli di copula.

3. **Struttura di dipendenza**
   L'analisi dei rendimenti logaritmici ha mostrato una correlazione positiva tra le variabili, sebbene con valori differenti per ciascuna coppia (ad esempio, correlazione relativamente più alta tra High e Close, e più bassa tra Open e Close). Questa osservazione implica che esiste una struttura di dipendenza tra le variabili di prezzo, che va oltre la correlazione lineare. Le copule ci permetteranno di catturare meglio questa dipendenza, soprattutto nei casi in cui le correlazioni sono condizionate da situazioni estreme (code pesanti).

4. **Dipendenze di coda**
   Osservando la distribuzione dei rendimenti logaritmici, è evidente che la distribuzione presenta code più pesanti rispetto a una normale distribuzione Gaussiana. Questo suggerisce una maggiore probabilità di eventi estremi (sia positivi che negativi), specialmente durante periodi di volatilità del mercato. Pertanto, è ragionevole assumere che le variabili presentino dipendenza di coda, rendendo modelli come la copula di Student-t o la copula di Clayton adatti per catturare le correlazioni nelle code inferiori, particolarmente durante i ribassi di mercato.

5. **Non-normalità delle distribuzioni marginali**
   I rendimenti logaritmici non seguono una distribuzione normale; piuttosto, mostrano asimmetria e code più pesanti. Sebbene la copula Gaussiana possa essere utilizzata

per una prima analisi, è preferibile utilizzare copule come la Student-t per gestire deviazioni dalla normalità, particolarmente utili per modellare le code e le correlazioni durante gli eventi estremi.

6. **Asimmetria nelle dipendenze**
   La matrice di correlazione calcolata tra i rendimenti (Open, High, Low, Close) mostra differenze nei livelli di correlazione tra le diverse variabili. Questa variabilità nelle correlazioni suggerisce che alcuni modelli di copula, come la Clayton (per le code inferiori) o la Gumbel (per le code superiori), possano offrire una descrizione più accurata delle dipendenze rispetto a una semplice copula Gaussiana.

7. **Condizioni di diversificazione**
   La copula di Frank potrebbe essere adatta per modellare le dipendenze tra le variabili che non mostrano comportamenti particolarmente forti nelle code (ovvero, dipendenze moderate e stabili). Tuttavia, i dati indicano la presenza di code pesanti, quindi questa copula potrebbe essere utilizzata solo come confronto con modelli che catturano meglio le dipendenze estreme.

4.7.1 Conclusione sulle assunzioni

Uniformità e stazionarietà sono requisiti fondamentali per l'applicazione dei modelli di copula. I dati sono stati trasformati per soddisfare parzialmente queste assunzioni.

Dipendenze di coda e non-normalità suggeriscono l'uso di copule robuste come la Student-t o modelli asimmetrici come la Clayton o la Gumbel per catturare meglio le relazioni tra variabili durante condizioni di stress di mercato.

L'asimmetria delle correlazioni evidenziata dalla matrice di correlazione indica la necessità di copule che possano gestire differenti tipi di dipendenze nelle code.

Queste assunzioni ci permettono di scegliere il modello di copula più appropriato per analizzare le dipendenze tra i rendimenti del DAX e comprendere meglio il comportamento del mercato in diverse condizioni economiche.

Parte IV

Stima dei Parametri delle Copule e Implementazione Pratica

Capitolo 5

Stime dei parametri

5.1 Introduzione alla Stima dei Parametri delle Copule

La stima dei parametri delle copule è cruciale per descrivere con precisione il tipo e il grado di dipendenza tra variabili. In particolare, in finanza:

- **Gestione del rischio**: La corretta modellazione della dipendenza è essenziale per calcolare indicatori come il Value-at-Risk (VaR) e il Conditional VaR (CVaR), nonché per valutare il rischio di portafoglio in condizioni estreme.
- **Ottimizzazione del portafoglio**: La conoscenza della dipendenza consente di costruire portafogli ottimizzati, tenendo conto delle correlazioni non lineari tra asset.
- **Eventi estremi**: La modellazione della *tail dependence* (dipendenza nelle code) permette di catturare correttamente la probabilità di eventi congiunti estremi, come crisi finanziarie o fallimenti simultanei di istituzioni.

L'importanza della stima risiede nella capacità di rappresentare accuratamente la struttura di dipendenza osservata nei dati, migliorando l'affidabilità e la precisione dei modelli finanziari.

Per stimare i parametri delle copule, esistono diversi metodi, ciascuno con vantaggi e limitazioni:

1. **Metodo della Massima Verosimiglianza (MLE - Maximum Likelihood Estimation)**: Massimizzando la funzione di verosimiglianza costruita sui dati, si ottengono degli stimatori per i vari parametri.
2. **Metodo dei Momenti**: Confronta i momenti osservati con quelli teorici per stimare i parametri della copula.
3. **Metodi Bayesiani**: Utilizzano distribuzioni a priori per stimare i parametri delle copule e aggiornano le stime con i dati osservati.

La stima dei parametri delle copule rappresenta un passaggio fondamentale nella modellazione della dipendenza tra variabili finanziarie. La scelta del metodo più appropriato dipende dalla complessità del modello, dalla disponibilità di dati e dai vincoli computazionali. L'utilizzo corretto delle copule migliora significativamente l'accuratezza dei modelli finanziari, con importanti applicazioni nella gestione del rischio, nella costruzione di portafogli e nella previsione di eventi estremi.

Nelle prossime pagine mostreremo un'analisi dettagliata dei vari metodi di stima (anche un implementazione Python) e un applicazione pratica utilizzando i dati del DAX.

5.2 Metodi di stima dei parametri

5.2.1 Stima di Massima Verosimiglianza (MLE)

Il Metodo della Massima Verosimiglianza (MLE) é una delle tecniche più utilizzate per la stima dei parametri delle copule grazie alla sua efficienza e alla solidità teorica. Questo metodo si basa sul principio di determinare i parametri che massimizzano la probabilità (o verosimiglianza) dei dati osservati sotto il modello scelto.

Procedura:

1. Supponiamo di avere n osservazioni $x_i = (x_{i1}, x_{i2}, ..., x_{id})$, con $i = 1, ..., n$, da una distribuzione congiunta $F(x; \theta)$, dove θ è il vettore dei parametri della copula.
2. La funzione di verosimiglianza è costruita come:
$$L(\theta; x_1, ..., x_n) = \prod_{i=1}^{n} f(x_i; \theta)$$
dove $f(x_i; \theta)$ è la densità congiunta.
3. Usando il Teorema di Sklar, la densità congiunta può essere scritta come:
$$f(x_i; \theta) = c(F_1(x_{i1}), ..., F_d(x_{id}); \theta) \prod_{j=1}^{d} f_j(x_{ij})$$
dove $f_j(x_{ij})$ sono le densità marginali e $c(\cdot; \theta)$ è la densità della copula.
4. Se le distribuzioni marginali sono note, la funzione di verosimiglianza dipende solo dalla copula:
$$L(\theta) = \prod_{i=1}^{n} c(F_1(x_{i1}), ..., F_d(x_{id}); \theta)$$
5. La stima $\hat{\theta}$ dei parametri si ottiene massimizzando il logaritmo della funzione di verosimiglianza:
$$\hat{\theta} = \arg\max_{\theta} \sum_{i=1}^{n} \log c(F_1(x_{i1}), ..., F_d(x_{id}); \theta)$$

Dopo aver spiegato come funziona il metodo matematicamente, forniamo dei motivi per cui l'MLE è comunemente utilizzato per le copule.

Vantaggi del MLE:

- **Efficienza statistica:**
  L'MLE produce stime consistenti, non distorte e asintoticamente efficienti sotto ipotesi regolari, garantendo la massima accuratezza possibile per grandi campioni.
- **Flessibilità:**
  L'MLE è adatta sia a copule con marginali note che a copule con marginali sconosciute (in quest'ultimo caso, si utilizza la variante del *Pseudo-MLE*).
- **Compatibilità con il teorema di Sklar:**
  Grazie alla separazione tra marginali e dipendenza, l'MLE consente di stimare parametri che riflettono esclusivamente la struttura di dipendenza, riducendo l'influenza delle marginali.
- **Applicazioni pratiche:**
  In ambito finanziario, l'MLE è particolarmente efficace per stimare copule complesse (es. copule t o Archimedee) che modellano fenomeni come la *tail dependence* (dipendenza nelle code).
- **Supporto computazionale:**
  L'MLE è ben supportata da software statistici e librerie numeriche (ad esempio, Python, R, MATLAB), rendendola una scelta praticabile anche per problemi reali con dataset di grandi dimensioni.

Il metodo della Massima Verosimiglianza 'e una tecnica robusta e versatile per la stima dei parametri delle copule, garantendo precisione ed efficienza nella modellazione della dipendenza. La sua applicazione, unita alla separazione tra marginali e struttura di dipendenza, ne fa uno strumento indispensabile nella modellazione finanziaria e in molte altre discipline quantitative.

Vediamo anche qualche limitazione di questo metodo.

Limitazioni del MLE:

- **Complessità Computazionale:**
  - La funzione di verosimiglianza può diventare complessa da calcolare, specialmente per modelli con molte variabili o copule con parametri complessi.
  - Richiede spesso ottimizzazione numerica iterativa (es. Newton-Raphson, metodi stocastici), che può essere lenta o soggetta a problemi di convergenza.
- **Sensibilità ai Dati:**
  - L'MLE assume che il modello scelto sia corretto. In ambito finanziario, però, i dati reali possono violare le ipotesi standard (es. non normalità delle marginali, dati mancanti, errori di misurazione).

  - Le stime possono risultare inefficaci se il modello non rappresenta bene i dati.

- **Dipendenza dalle Condizioni Asintotiche:**
  - L'efficienza dell'MLE è garantita solo per campioni sufficientemente grandi. Nei dataset finanziari con pochi dati (es. eventi estremi rari), l'MLE può essere instabile.

- **Problemi di Robustezza:**
  - L'MLE è sensibile agli *outlier*, che sono comuni nei dati finanziari (ad esempio, picchi improvvisi di volatilità).
  - Per mitigare questo problema, possono essere necessari metodi alternativi.

- **Vincoli Numerici e Positività:**
  - Nel caso di copule multivariate, come la copula Gaussiana o t-Student, la matrice di correlazione deve essere positiva definita.
  - L'imposizione di vincoli aggiunge complessità computazionale.

Considerazioni finali

L'MLE è uno strumento potente e teoricamente solido per la stima dei parametri in contesti finanziari. Tuttavia, la sua applicazione pratica deve tenere conto delle caratteristiche specifiche dei dati finanziari e della complessità computazionale del modello scelto. In alternativa, quando i limiti dell'MLE diventano problematici, possono essere considerati altri approcci, come il metodo dei momenti, il metodo bayesiano o il metodo pseudo-MLE.

5.2.2 Implementazione Python per una copula Student t:

Mostriamo ora un'implementazione in Python di questo metodo.
Iniziamo mostrando le librerie necessarie per l'esecuzione dello script.

```
import numpy as np
import pandas as pd
import scipy.stats as stats
import scipy.optimize as optimize
import matplotlib.pyplot as plt
from scipy.special import gamma
import os
```

Successivamente sará necessario un dataset giá normalizzato su cui applicare questo metodo. In questo esempio viene utilizzato il dataset $uniform\_data$ mostrato nella terza parte.
In questo esempio utilizzeremo una copula Student t bivariata, le cui variabili saranno i valori *Open* e i valori *Close* che andremo ad indicare rispettivamente con u e v.

```
# Selezione di due colonne per la copula
u, v = uniform_data['Open'].values, uniform_data['Close'].values
```

Implementiamo direttamente la Copula Student-t ricordando che la sua densitá é:

$$c(u,v;\rho,\nu) = \frac{\Gamma\left(\frac{\nu+2}{2}\right)\Gamma\left(\frac{\nu}{2}\right)^{-1}\left(1+\frac{x^2+y^2-2\rho xy}{\nu(1-\rho^2)}\right)^{-\frac{\nu+2}{2}}}{\sqrt{1-\rho^2}\cdot\Gamma\left(\frac{\nu+1}{2}\right)^2\Gamma\left(\frac{\nu}{2}\right)^{-2}\left(1+\frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}}\left(1+\frac{y^2}{\nu}\right)^{-\frac{\nu+1}{2}}}$$

dove $x = t_\nu^{-1}(u)$ e $y = t_\nu^{-1}(v)$.

```
def t_copula_pdf(u, v, rho, df):
x = stats.t.ppf(u, df)
y = stats.t.ppf(v, df)

# Protezione contro overflow/underflow
try:

numerator = gamma((df + 2) / 2) * gamma(df / 2) * (
1 + (x ** 2 + y ** 2 - 2 * rho * x * y) /
 (df * (1 - rho ** 2))) ** (-(df + 2) / 2)

denominator = gamma((df + 1) / 2) ** 2 * df * np.pi *
 np.sqrt(1 - rho ** 2) * (1 + x ** 2 / df) ** (
-(df + 1) / 2) * (1 + y ** 2 / df) ** (-(df + 1) / 2)

result = numerator / denominator

# Sostituire infiniti o NaN con un valore molto piccolo ma positivo
result = np.where(np.isfinite(result) & (result > 0), result, 1e-10)
return result
except:
# In caso di errore, ritorna un valore di default
return np.ones_like(u) * 1e-10
```

Implementiamo la funzione likelihood

```
def t_copula_likelihood(params, u, v):
rho = np.tanh(params[0])
df = np.exp(params[1]) + 2
likelihoods = t_copula_pdf(u, v, rho, df)

# Gestione di valori non validi
valid_idx = ~np.isnan(likelihoods) & (likelihoods > 0)
if not np.any(valid_idx):
return 1e10
return -np.sum(np.log(likelihoods[valid_idx]))
```

Implementiamo la stima MLE

```
def estimate_t_copula_MLE(u, v):
initial_guess = [0.5, np.log(8)]
bounds = [(-10, 10), (-10, 10)]  # Limiti per rho e log(df)
result = optimize.minimize(t_copula_likelihood, initial_guess, args=(u, v),
method='L-BFGS-B', bounds=bounds)
rho_estimated = np.tanh(result.x[0])
df_estimated = np.exp(result.x[1]) + 2
return rho_estimated, df_estimated
```

Infine possiamo osservare i risultati ottenuti in questo modo

```
rho_t_mle, df_t_mle = estimate_t_copula_MLE(u, v)
print(f" Parametro stimato (rho) per la Copula t-Student: {rho_t_mle:.4f}")
print(f" Gradi di libertà stimati per la Copula t-Student: {df_t_mle:.2f}")
```

I risultati numerici e i valori di stima del nostro dataset saranno calcolati e mostrati più avanti

5.2.3 Metodo dei Momenti

Il metodo dei momenti è una tecnica di stima che si basa sull'equiparazione dei momenti teorici di una distribuzione con i momenti campionari calcolati dai dati osservati. Nel contesto delle copule, il metodo può essere utilizzato per stimare i parametri che descrivono la dipendenza tra le variabili aleatorie, garantendo coerenza tra la struttura di dipendenza modellata e quella osservata.

Applicazione alle copule

Le copule sono funzioni che legano le distribuzioni marginali di variabili aleatorie alla loro distribuzione congiunta, separando la dipendenza dalla struttura marginale. Per stimare i parametri di una copula con il metodo dei momenti:

1. **Scelta dei momenti di interesse**: Si identificano statistiche che catturano la dipendenza (ad esempio, Kendall's τ, Spearman's ρ o altre misure di concordanza).
2. **Calcolo dei momenti campionari**: Si calcolano i momenti empirici dai dati osservati, ad esempio, calcolando τ o ρ sui campioni.
3. **Imposizione di uguaglianza**: I momenti teorici della copula, funzione dei parametri da stimare, vengono eguagliati ai momenti campionari.
4. **Risoluzione del sistema**: Si risolvono le equazioni risultanti per determinare i parametri della copula.

Ambiti di Applicazione

Il metodo dei momenti applicato alle copule trova largo impiego in settori dove è cruciale modellare la dipendenza tra variabili, come:

- **Finanza**: Per analizzare la dipendenza tra rendimenti di asset.
- **Assicurazioni e gestione del rischio**: Per modellare eventi estremi correlati, come sinistri catastrofici.

Vantaggi e Svantaggi

Elenchiamo alcuni pro e contro di utilizzare questo metodo:

Vantaggi

- **Semplicità computazionale**: Rispetto ad altri metodi (ad esempio, la massima verosimiglianza), il metodo dei momenti è spesso più semplice da implementare e richiede meno assunzioni sul modello.
- **Intuitività**: L'approccio è facilmente interpretabile grazie al legame diretto con misure di dipendenza osservabili come τ e ρ.
- **Robustezza**: È meno sensibile a errori nelle specifiche delle distribuzioni marginali.

Svantaggi

- **Perdita di efficienza**: Gli stimatori ottenuti non sono sempre efficienti, il che significa che potrebbero avere una varianza maggiore rispetto a quelli della massima verosimiglianza.
- **Dipendenza dalla scelta dei momenti**: La qualità della stima dipende fortemente dalla scelta dei momenti, che potrebbero non catturare adeguatamente la complessità della dipendenza.
- **Limitazioni con dati scarsi**: Con campioni piccoli, i momenti campionari potrebbero essere poco rappresentativi.

Limiti in contesti multivariati

Un altro fattore da tenere in considerazione è il numero di variabili a cui è applicato, in quanto potrebbe avere delle limitazioni come:

1. **Difficoltà di Generalizzazione in Dimensioni Elevate**:
   - Le misure di concordanza (ad esempio, Kendall's τ) diventano difficili da calcolare per coppie di variabili.
   - Il numero di parametri della copula aumenta esponenzialmente con la dimensione, rendendo il sistema di equazioni derivato dai momenti difficile da risolvere.
2. **Sensibilità ai Dati Empirici**: I momenti campionari utilizzati per la stima sono sensibili alla presenza di outlier o dati scarsi, introducendo bias significativi nella stima dei parametri.

3. **Rappresentazione Limitata della Dipendenza**:
   - Il metodo dei momenti si basa su misure sintetiche della dipendenza (come τ o ρ), che potrebbero non cogliere adeguatamente relazioni complesse, come:
     - Dipendenze non lineari.
     - Code pesanti o asimmetrie estreme.
4. **Convergenza Non Ottimale**: Gli stimatori ottenuti con il metodo dei momenti non sono necessariamente efficienti in termini di varianza, soprattutto quando i dati presentano una struttura di dipendenza complessa.

5.2.4 Il Tau di Kendall

Abbiamo parlato più volte della misura τ di Kendall, spieghiamo meglio in cosa consiste.

Il tau di Kendall è una misura di concordanza per due variabili casuali continue. È definito come la differenza tra la probabilità di concordanza e la probabilità di discordanza. Può essere interpretato come la probabilità che due coppie di osservazioni scelte a caso dal campione siano concordanti meno la probabilità che siano discordanti.

Principio di base

Dato un insieme di n coppie di dati $(x_1, y_1), (x_2, y_2), \ldots, (x_n, y_n)$, il Kendall's Tau si basa sul confronto tra tutte le coppie di osservazioni:

1. **Coppia concordante**: Una coppia (x_i, y_i) e (x_j, y_j) è *concordante* se i segni delle differenze $(x_j - x_i)$ e $(y_j - y_i)$ sono uguali (entrambi positivi o entrambi negativi). In parole semplici:
$$(x_j - x_i)(y_j - y_i) > 0$$

2. **Coppia discordante**: Una coppia (x_i, y_i) e (x_j, y_j) è *discordante* se i segni delle differenze $(x_j - x_i)$ e $(y_j - y_i)$ sono opposti (uno positivo e l'altro negativo). In parole semplici:
$$(x_j - x_i)(y_j - y_i) < 0$$

3. **Coppia legata**: Una coppia è *legata* se $x_j = x_i$ o $y_j = y_i$.

Formula del Kendall's Tau

Il Kendall's Tau è calcolato come:

$$\tau = \frac{C - D}{\binom{n}{2}}$$

Dove:

- C è il numero di coppie concordanti.
- D è il numero di coppie discordanti.
- $\binom{n}{2} = \frac{n!}{2(n-2)!} = \frac{n(n-1)}{2}$ è il numero totale di coppie.

Valori di τ:

- $\tau = 1$: perfetta concordanza.
- $\tau = -1$: perfetta discordanza.
- $\tau = 0$: assenza di relazione (casuale).

Differenza con il Coefficiente di Correlazione di Spearman

Sebbene sia il tau di Kendall che il rho di Spearman misurino la concordanza, i loro valori possono essere diversi. La relazione tra i due dipende dalla particolare famiglia di copule che descrive la dipendenza tra le variabili. Ci sono disuguaglianze universali che mettono in relazione i due coefficienti, ma la relazione specifica può variare.

La scelta tra il tau di Kendall e il rho di Spearman dipende spesso dalla specifica applicazione e dalle preferenze personali. Il tau di Kendall è talvolta preferito per la sua interpretazione probabilistica più diretta e per la sua robustezza agli outlier. Il rho di Spearman è talvolta preferito per la sua relazione con la correlazione lineare e la sua maggiore sensibilità alle differenze nei ranghi.

In generale, sia il tau di Kendall che il rho di Spearman sono misure utili della concordanza e possono fornire informazioni preziose sulla dipendenza tra due variabili casuali.

5.2.5 Relazioni tra parametri delle copule e misure di dipendenza

Le formule che legano il tau di Kendall ai parametri delle diverse copule:

1. **Copula Gaussiana**

$$\tau = \frac{2}{\pi} \arcsin(\rho)$$

   Inversione:

$$\rho = \sin\left(\frac{\pi\tau}{2}\right)$$

2. **Copula t-Student**

$$\tau = \frac{2}{\pi} \arcsin(\rho)$$

   Stessa relazione della Gaussiana, indipendente dai gradi di libertà ν.

3. **Copula di Clayton**

$$\tau = \frac{\theta}{\theta + 2}$$

   Inversione:

$$\theta = \frac{2\tau}{1 - \tau}$$

4. **Copula di Gumbel**

$$\tau = 1 - \frac{1}{\theta}$$

   Inversione:

$$\theta = \frac{1}{1 - \tau}$$

5. **Copula di Frank**

$$\tau = 1 - \frac{4}{\theta}(1 - D_1(\theta))$$

dove $D_1(\theta)$ è la funzione di Debye:

$$D_1(\theta) = \frac{1}{\theta}\int_0^\theta \frac{t}{e^t - 1}dt$$

Non esiste una formula di inversione semplice.

5.2.6 Codice Python metodo dei momenti

Applicazione alla copula Gumbel

Per prima cosa definiamo la relazione tra il tau di Kendall e il parametro theta della Gumbel.

```
def gumbel_theta_kendall(tau):
"""
Calcola il parametro theta della copula di Gumbel a partire da tau di Kendall.
Formula: theta = 1 / (1 - tau)
"""
return 1 / (1 - tau)
```

Successivamente impostiamo una funzione che calcoli il tau di Kendall in maniera ottimizzata (non possiamo calcolarlo come nell'esercizio numerico in seguito all'elevato costo computazionale).

```
from scipy.stats import kendalltau

def kendall_tau_multivariate_optimized(dataset):
"""
Calcola il Kendall tau medio per un dataset multivariato usando scipy.
"""
n_vars = dataset.shape[1]
taus = []

# Calcola Kendall tau per ogni coppia di variabili
for i in range(n_vars):
for j in range(i + 1, n_vars):
tau, _ = kendalltau(dataset[:, i], dataset[:, j])
taus.append(tau)

# Restituisce il tau medio e il numero di coppie concordanti/discordanti
tau_mean = np.mean(taus)
return tau_mean
```

Per concludere implementiamo il metodo dei momenti ottimizzato.

```
def method_of_moments_gumbel_optimized(dataset):
"""
Applica il metodo dei momenti per stimare
il parametro theta della copula di Gumbel.
"""
# Calcola il tau di Kendall medio
tau_mean = kendall_tau_multivariate_optimized(dataset)

# Calcola il parametro theta
theta = gumbel_theta_kendall(tau_mean)
return theta, tau_mean
```

Ora non ci resta che applicare questo metodo al nostro dataset $uniform\_data$ e ottenere il risultato.

```
# Stima il parametro theta
theta_gumbel, tau_mean = method_of_moments_gumbel_optimized(uniform_data)
print(f"Parametro theta stimato della Gumbel: {theta_gumbel:.4f}")
```

Applicazione alla copula Clayton

Definiamo la relazione tra il tau di Kendall e il parametro theta della Clayton.

```
def clayton_theta_kendall(tau):
"""
Calcola il parametro theta della copula di Clayton a partire da tau di Kendall.
Formula: theta = 2*tau / (1 - tau)
"""
return 2*tau / (1 - tau)
```

La funzione per il calcolo del tau di Kendal é esattamente uguale alla precedente.

Implementiamo il metodo dei momenti ottimizzato.

```
def method_of_moments_clayton_optimized(dataset):
"""
Applica il metodo dei momenti per stimare il parametro theta della copula di Clayton.
"""
# Calcola il tau di Kendall medio
tau_mean = kendall_tau_multivariate_optimized(dataset)

# Calcola il parametro theta
theta = clayton_theta_kendall(tau_mean)
return theta, tau_mean
```

Applichiamo questo metodo al nostro dataset $uniform\_data$ e otteniamo il risultato.

```
# Stima il parametro theta
theta_clayton, tau_mean = method_of_moments_clayton_optimized(uniform_data)
print(f"Parametro theta stimato della Clayton: {theta_clayton:.4f}")
```

5.2.7 Stima Bayesiana

La stima bayesiana è un approccio alla statistica basato sul teorema di Bayes, che combina informazioni a priori con dati osservati per aggiornare la nostra conoscenza su un fenomeno incerto. È ampiamente utilizzata in applicazioni come l'apprendimento automatico, l'inferenza statistica e il processo decisionale.

Teorema di Bayes

La statistica bayesiana si fonda sul teorema di Bayes, che dice che, data due variabili aleatorie (anche vettoriali) X e Y, allora:

$$f(x \mid y) = \frac{f(y,x)}{f(y)} = \frac{f(y \mid x)f(x)}{f(y)}$$

dove $f(y) = \int f(y,x)dx$.

Il teorema di Bayes permette di passare dalla condizionata di y rispetto a x a quella di x rispetto a y.

Un altro modo di vedere il teorema di Bayes è di tipo "iterativo":

- Ho una distribuzione a priori su x, $f(x)$.
- Osservo una nuova variabile y, che dipende da x, tramite $f(y \mid x)$.
- Allora l'informazione che ho su x, dopo aver osservato y, cambia in $f(x \mid y)$.

Consideriamo ora $x = \theta$ come parametri di una densità di probabilità (ad esempio μ, σ^2 se parliamo di una normale), allora avremo:

- $f(\theta \mid y)$ è chiamata distribuzione a posteriori di θ.
- $f(y \mid \theta)$ è la congiunta delle osservazioni, che è possibile vedere anche come la verosimiglianza.
- $f(\theta)$ è la distribuzione a priori. Questa distribuzione riflette ciò che sappiamo dei parametri prima di osservare il campione y.
- $f(y)$ è la costante di normalizzazione, in genere meno rilevante poiché non dipende da θ.

La stima bayesiana, specialmente in combinazione con le copule, offre numerosi vantaggi in contesti caratterizzati da dati limitati o mercati volatili. Questi vantaggi emergono dal fatto che il paradigma bayesiano consente di integrare informazioni a priori, aggiornare le credenze in modo dinamico e modellare con precisione dipendenze complesse.

Vantaggi Generali della Stima Bayesiana

1. **Integrazione di conoscenze a priori**
   La stima bayesiana consente di utilizzare informazioni preesistenti, come storie storiche o conoscenze di esperti, per costruire una distribuzione a priori. Questo è particolarmente utile in scenari con dati limitati.

2. **Aggiornamento dinamico delle credenze**
   Il paradigma bayesiano permette di aggiornare i parametri di interesse man mano che nuovi dati diventano disponibili, utilizzando il teorema di Bayes.

3. **Distribuzione a posteriori invece di stime puntuali**
   La stima bayesiana produce distribuzioni a posteriori che rappresentano l'incertezza sui parametri, offrendo un quadro probabilistico più robusto rispetto all'uso di stime puntuali.

4. **Gestione di modelli complessi**
   Grazie alla capacità di combinare la struttura di dipendenza (copule) con modelli marginali, la stima bayesiana è particolarmente adatta per problemi multivariati o non lineari.

Vantaggi Specifici nei Contesti con Copule

Le **copule** sono utili per modellare le dipendenze tra variabili aleatorie, anche quando queste non seguono distribuzioni gaussiane o hanno comportamenti estremi (ad esempio, correlazioni nelle code). Quando abbinate alla stima bayesiana, i vantaggi si amplificano nei seguenti modi:

1. **Modellazione della dipendenza in contesti di dati limitati**
   Nei mercati finanziari o assicurativi con dati scarsi, l'approccio bayesiano consente di sfruttare distribuzioni a priori sui parametri della copula, riducendo il rischio di sottostimare o sovrastimare la dipendenza.

2. **Robustezza in mercati volatili**
   Le copule consentono di modellare cambiamenti nella dipendenza tra variabili in mercati turbolenti. L'approccio bayesiano aggiorna dinamicamente questi parametri con nuovi dati, migliorando la risposta ai cambiamenti rapidi.

3. **Modellazione delle code**
   Le copule come la Gumbel o la t-Student sono in grado di catturare la dipendenza nelle code della distribuzione. In combinazione con l'approccio bayesiano, è possibile stimare parametri di coda con maggiore affidabilità anche in presenza di pochi dati osservati.

4. **Riduzione del rischio di overfitting**
   La stima bayesiana utilizza distribuzioni a priori regolarizzanti, che prevengono l'overfitting tipico nei modelli di dipendenza con molti parametri. Questo é particolarmente importante in contesti con dati limitati.

Applicazioni Specifiche

1. **Finanza (Portafogli e Rischio)**
   - La stima bayesiana con copule permette di modellare la dipendenza tra rendimenti degli asset in portafoglio, considerando eventi estremi (correlazioni nelle code).
   - È utile per stimare la *Value at Risk* (VaR) e il *Conditional VaR* (CVaR) in mercati volatili.
2. **Assicurazioni (Rischi Multivariati)**
   - Modellare la dipendenza tra sinistri assicurativi (ad esempio, danni da eventi naturali correlati).
   - L'approccio bayesiano consente di incorporare conoscenze a priori su rischi rari ma severi.
3. **Analisi Multivariata e Dati Mancanti**
   - Quando alcune variabili o dati sono mancanti, le copule e la stima bayesiana consentono di stimare la dipendenza tra variabili osservate e non osservate.

5.2.8 Metodo MCMC in Python

Il metodo di Monte Carlo a catena di Markov (MCMC) è una tecnica di campionamento utilizzata per approssimare distribuzioni di probabilità complesse. Questo metodo si basa sulla costruzione di una catena di Markov il cui stato stazionario coincide con la distribuzione target desiderata. Tra gli algoritmi più comuni per l'MCMC troviamo Metropolis-Hastings e Gibbs Sampling.

L'MCMC è ampiamente usato in statistica bayesiana, econometria, machine learning e fisica computazionale per la stima di parametri in modelli complessi dove l'integrazione diretta non è possibile.

Implementazione in Python

Di seguito viene presentata un'implementazione del metodo MCMC in Python utilizzando la copula gaussiana per modellare la dipendenza tra variabili finanziarie.

Librerie utilizzate

```
import numpy as np
import pandas as pd
import scipy.stats as stats
import matplotlib.pyplot as plt
from scipy.optimize import minimize
```

Utilizziamo sempre le due colonne *Open* e *Close* del dataset $uniform\_data$ come campioni chiamndole corrispettivamente u_unif e v_unif.

Definizione della funzione di verosimiglianza per la Copula Gaussiana

```
def gaussian_copula_log_likelihood(rho, u, v):
"""Calcola la log-verosimiglianza della copula gaussiana."""
if abs(rho) >= 0.999:  # Evita problemi numerici ai bordi
return -np.inf

x = stats.norm.ppf(u)
y = stats.norm.ppf(v)

# Log-verosimiglianza della copula gaussiana
term1 = -0.5 * np.log(1 - rho ** 2)
term2 = -(rho ** 2 * (x ** 2 + y ** 2) - 2 * rho * x * y) / (2 * (1 - rho ** 2))

return np.sum(term1 + term2)
```

Ora definiamo l'algoritmo MCMC migliorato con multiple chain e diagnostica robusta

```
def improved_metropolis_hastings(log_likelihood, u, v, n_chains=4,
n_iter=20000, burn_in=5000,
target_acceptance=0.234):
"""
Implementazione migliorata di Metropolis-Hastings con catene multiple
e adattamento del passo basato su Gelman et al. (1996).
"""
n_params = 1  # Dimensione del parametro (solo rho)

# Inizializzazione di catene multiple con punti di partenza diversi
chains = np.zeros((n_chains, n_iter, n_params))

# Punti di partenza distribuiti uniformemente tra -0.9 e 0.9
starting_points = np.linspace(-0.8, 0.8, n_chains)

# Parametri per l'adattamento
gamma1 = 0.75  # Parametro che controlla l'adattamento iniziale
adaptation_steps = int(n_iter * 0.6)  # Numero di passi di adattamento

# Memorizza i tassi di accettazione e le larghezze delle proposte
acceptance_rates = np.zeros(n_chains)
proposal_widths = np.ones(n_chains) * 0.1  # Valori iniziali

print("\n====== AVVIO MCMC CON MULTIPLE CATENE ======")
```

```
for c in range(n_chains):
chain = chains[c]
chain[0, 0] = starting_points[c]  # Punto di partenza
accepted = 0

print(f"- Catena {c + 1}: punto di partenza = {starting_points[c]:.4f}")

for i in range(1, n_iter):
current_rho = chain[i - 1, 0]

# Passo adattivo che diminuisce con le iterazioni
if i <= adaptation_steps:
adapt_factor = (i / adaptation_steps) ** gamma1
current_width = proposal_widths[c] *
(1 - adapt_factor) + 0.1 * adapt_factor
else:
current_width = proposal_widths[c]

# Proposta: U(-$\delta$,$\delta$) centrata sull'attuale valore
delta = current_width
proposal = current_rho + np.random.uniform(-delta, delta)

# Rifletti se fuori dai limiti (-0.999, 0.999)
if proposal <= -0.999:
proposal = -0.999 + ((-0.999) - proposal)
elif proposal >= 0.999:
proposal = 0.999 - (proposal - 0.999)

# Calcola il rapporto di accettazione
log_p_current = log_likelihood(current_rho, u, v)
log_p_proposal = log_likelihood(proposal, u, v)
log_accept_ratio = log_p_proposal - log_p_current

# Accetta o rifiuta
if np.log(np.random.random()) < log_accept_ratio:
chain[i, 0] = proposal
accepted += 1
else:
chain[i, 0] = current_rho

# Adatta la larghezza della proposta
if i % 500 == 0 and i <= adaptation_steps:
batch_acceptance = accepted / i

# Aggiusta la larghezza per avvicinarsi al tasso di accettazione target
```

```
if batch_acceptance > target_acceptance:
proposal_widths[c] *= 1.1  # Aumenta
else:
proposal_widths[c] *= 0.9  # Diminuisci

# Limita la larghezza per sicurezza
proposal_widths[c] = max(0.01, min(1.0, proposal_widths[c]))

if i % 5000 == 0:
print(f"  Iterazione {i}, catena {c + 1}: accettazione = {batch_acceptance:.4f}, "
f"larghezza = {proposal_widths[c]:.4f}")

# Statistiche finali per questa catena
acceptance_rates[c] = accepted / n_iter
print(f"- Catena {c + 1} completata: accettazione = {acceptance_rates[c]:.4f}, "
f"larghezza finale = {proposal_widths[c]:.4f}")

# Rimuovi burn-in e unisci le catene per l'analisi
combined_samples = chains[:, burn_in:, 0].flatten()

return chains, acceptance_rates, combined_samples
```

Ora eseguiamo MCMC migliorato

```
n_chains = 4
n_iter = 20000
burn_in = 5000

chains, acceptance_rates, combined_samples = improved_metropolis_hastings(
gaussian_copula_log_likelihood,
u_unif,
v_unif,
n_chains=n_chains,
n_iter=n_iter,
burn_in=burn_in
)
```

Verifichiamo la convergenza del metodo tramite la diagnostica di Gelman-Rubin.

La diagnostica di Gelman-Rubin, nota anche come $\hat{R}$, è un metodo per valutare la convergenza delle catene MCMC. Confronta la varianza tra le catene con la varianza interna delle singole catene, fornendo un'indicazione di stabilità del campionamento. Se $\hat{R} \approx 1$, le catene sono considerate convergenti; valori superiori a 1.1 indicano potenziali problemi di convergenza e la necessità di iterazioni aggiuntive.

```
def gelman_rubin(chains, burn_in=0):
```

```
"""Calcola la diagnostica R-hat di Gelman-Rubin."""
n_chains, n_iter, n_params = chains.shape

if burn_in > 0:
chains = chains[:, burn_in:, :]

n_iter = chains.shape[1]

# Medie delle catene
chain_means = np.mean(chains, axis=1)  # shape: (n_chains, n_params)

# Varianza tra le catene
B = n_iter * np.var(chain_means, axis=0, ddof=1)  # shape: (n_params,)

# Varianza entro le catene
W = np.mean(np.var(chains, axis=1, ddof=1), axis=0)  # shape: (n_params,)

# Stima complessiva della varianza
var_hat = ((n_iter - 1) / n_iter) * W + (1 / n_iter) * B

# Calcolo R-hat
R_hat = np.sqrt(var_hat / W)

return R_hat
```

Infine visualizziamo i risultati.

```
rho_mean = np.mean(combined_samples)
rho_std = np.std(combined_samples)
rho_median = np.median(combined_samples)
rho_ci = np.percentile(combined_samples, [2.5, 97.5])

print("\n====== RISULTATI FINALI ======")
print(f" Stima Bayesiana di $\rho$ con MCMC: {rho_mean:.4f} ± {rho_std:.4f}")
print(f" Mediana: {rho_median:.4f}")
print(f" Intervallo di credibilità al 95%: [{rho_ci[0]:.4f}, {rho_ci[1]:.4f}]")
print(f" Dimensione del campione efficace: {len(combined_samples)}")
```

5.3 Implementazione Pratica: Applicazione ai Dati del DAX

5.3.1 Preparazione dei dati

La descrizione dettagliata del dataset utilizzato è già fornita nella terza parte della tesi. In questa sezione, ci concentreremo invece sull'implementazione di un processo strutturato per la preparazione e la verifica dei dati, con l'obiettivo di garantirne la qualità e l'affidabilità per le analisi successive.

Il codice sviluppato esegue una serie di operazioni di diagnostica e pulizia sui dati di mercato, consentendo di ottenere un dataset coerente, privo di errori e pronto per le operazioni analitiche.

Librerie utilizzate

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import os
import csv
```

Parte 1: Diagnostica del file CSV

La prima fase del processo di preparazione dei dati consiste nella **diagnostica del file CSV**. L'obiettivo principale di questa sezione è garantire che il file di dati esista, sia leggibile e abbia una struttura coerente. Per farlo, vengono eseguite le seguenti operazioni:

1. **Verifica dell'esistenza del file**
   - Il codice controlla se il file specificato (`DAX_3Y-1M.csv`) è presente nella directory corrente.
   - Se il file non esiste, viene stampato un messaggio di errore con l'indicazione della directory corrente e dei file disponibili, facilitando il debugging.
2. **Ispezione delle prime righe del file**
   - Se il file è stato trovato, vengono lette e stampate le prime dieci righe del file per ottenere una visione preliminare della sua struttura.
   - Questo aiuta a identificare eventuali problemi evidenti, come caratteri strani, righe mancanti o errori di formattazione.
3. **Identificazione del delimitatore**
   - Poiché i file CSV possono essere separati da virgole (`,`), punti e virgola (`;`), tabulazioni (`\t`) o altri caratteri, viene utilizzata la funzione `csv.Sniffer()` per rilevare automaticamente il delimitatore corretto.

- Questo passaggio è essenziale per evitare errori di lettura quando si carica il file con `pandas`.

4. **Verifica dell'intestazione**
   - Il codice verifica se il file contiene un'intestazione (ovvero nomi di colonne) e la conferma all'utente.
   - Se l'intestazione non è presente, sarà necessario gestire i nomi delle colonne in modo esplicito nelle fasi successive.

Questa prima parte assicura che il file sia disponibile e che la sua struttura sia chiara prima di procedere con la lettura e la pulizia dei dati. Se il file presenta problemi, gli errori vengono segnalati tempestivamente, evitando che il codice fallisca nelle fasi successive. Nel terminale visualizzo:

DIAGNOSTICA DEL FILE CSV

File 'DAX_3Y-1M.csv' trovato.

Ispezione delle prime righe del file:

```
Riga 1: DateTime, Open, High, Low, Close
Riga 2: 02/01/2020 01:15:00 +01:00, 13174, 13194.5, 13171.5, 13177.5
Riga 3: 02/01/2020 01:16:00 +01:00, 13180, 13180.5, 13179, 13181.5
Riga 4: 02/01/2020 01:17:00 +01:00, 13182, 13182, 13180.5, 13181.5
...
```

Delimitatore rilevato: ','
Intestazione rilevata: False

Lettura riuscita con delimitatore ','

	DateTime	Open	High	Low	Close	Unnamed: 5
0	02/01/2020 01:15:00 +01:00	13174	13194.5	13171.5	13177.5	NaN
1	02/01/2020 01:16:00 +01:00	13180	13180.5	13179	13181.5	NaN
2	02/01/2020 01:17:00 +01:00	13182	13182	13180.5	13181.5	NaN

...

Lettura base riuscita
Dimensioni: (616397, 6)
Colonne: ['DateTime', 'Open', 'High', 'Low', 'Close', 'Unnamed: 5']
Prime 5 righe:

```
DateTime                 Open     High     Low     Close  Unnamed: 5
0  02/01/2020 01:15:00 +01:00  13174  13194.5  13171.5  13177.5      NaN
1  02/01/2020 01:16:00 +01:00  13180  13180.5  13179    13181.5      NaN
2  02/01/2020 01:17:00 +01:00  13182  13182    13180.5  13181.5      NaN
...
```

Parte 2: Lettura e Correzione dei Dati

Dopo aver verificato l'esistenza e la struttura del file CSV nella fase di diagnostica, questa seconda parte si concentra sulla **lettura e correzione dei dati**, assicurandosi che il file venga caricato correttamente e che le colonne siano interpretate nel formato corretto.

Le operazioni principali svolte in questa fase sono:

1. **Tentativi di lettura del file con diversi delimitatori**
   - I file CSV possono essere separati da diversi caratteri (virgole `,`, punti e virgola `;`, tabulazioni `\t`, barre verticali `|`).
   - Il codice prova a leggere il file con ciascun delimitatore e controlla quale funziona senza errori.
   - Se la lettura con un certo delimitatore ha successo, il codice interrompe il ciclo e memorizza il delimitatore corretto per le fasi successive.
2. **Gestione di possibili errori di encoding**
   - I file CSV possono essere salvati con differenti codifiche (`UTF-8`, `ISO-8859-1`, `latin1`, `cp1252`, ecc.), e una codifica errata può causare errori nella lettura.
   - Viene tentata la lettura con `UTF-8`, ma se fallisce, vengono provate altre codifiche.
   - Se tutte le codifiche testate falliscono, viene segnalato un errore.
3. **Verifica della struttura del DataFrame**
   - Dopo aver letto il file, vengono stampate le prime righe per confermare che i dati siano stati caricati correttamente.
   - Viene controllato se il dataset contiene colonne indesiderate, come `Unnamed: X`, che potrebbero derivare da errori nel file originale.
   - Se esistono colonne numeriche con dati misti (numeri e stringhe), viene segnalato un avviso.
4. **Conversione dei dati al formato corretto**
   - Le colonne numeriche (`Open`, `High`, `Low`, `Close`) vengono convertite in formato numerico, forzando la conversione (`errors='coerce'`), in modo da trasformare eventuali valori errati in `NaN` (valori mancanti).

- Se una colonna essenziale come `DateTime` esiste, viene convertita in un formato di data/ora (`datetime64`).
- Se alcune righe contengono errori di conversione (es. date errate o valori non numerici), vengono rimosse.

Questa fase assicura che il file venga letto in modo robusto e senza errori, adattando la configurazione al formato effettivo del CSV. Il risultato finale è un dataset pulito e ben strutturato, pronto per ulteriori operazioni di analisi e modellazione.
Nel terminale visualizzeremo:

Correzione e Preprocessing dei Dati

Impossibile convertire 'DateTime': il valore "13/01/2020 01:15:00 +01:00" non corrisponde al formato `"%m/%d/%Y %H:%M:%S %z"` alla posizione 8672. Possibili soluzioni:

- Usare `format` se le stringhe hanno un formato coerente.
- Usare `format="ISO8601"` se le stringhe sono tutte in ISO8601 ma non esattamente nello stesso formato.
- Usare `format="mixed"` per inferire automaticamente il formato per ogni elemento individualmente.

Tentativo di correzione del formato della data: Esempi di date convertite:

```
'02/01/2020 01:15:00 +01:00', '02/01/2020 01:16:00 +01:00',
'02/01/2020 01:17:00 +01:00', '02/01/2020 01:18:00 +01:00'
```

Conversione colonne di prezzo

Colonne convertite: `Open`, `High`, `Low`, `Close` Tipi di dati iniziali:

```
Open      object
High      object
Low       object
Close     object
dtype: object
```

Le seguenti colonne sono state convertite correttamente:

- Colonna `Open`: convertita da formato con virgole
- Colonna `High`: convertita da formato con virgole
- Colonna `Low`: convertita da formato con virgole
- Colonna `Close`: convertita da formato con virgole

Rimosse 0 righe con valori mancanti

Statistiche sui dati numerici

	Open	High	Low	Close
count	616397	616397	616397	616397
mean	13884.68	13887.52	13881.79	13884.66
std	1747.75	1763.92	1733.78	1747.75
min	7970.00	7970.00	7970.00	7970.00
25%	12878.50	12878.50	12878.50	12878.50
50%	13914.00	13914.00	13914.00	13914.00
75%	15487.00	15487.00	15487.00	15487.00
max	16294.00	16294.00	16294.00	16294.00

Tabella 5.1. Statistiche descrittive delle colonne numeriche

Identificazione degli outlier

- Colonna `Open`: 6270 potenziali outlier (1.02%)
- Colonna `High`: 6232 potenziali outlier (1.01%)
- Colonna `Low`: 6331 potenziali outlier (1.03%)
- Colonna `Close`: 6287 potenziali outlier (1.02%)

Impostazione indice e salvataggio

La colonna `DateTime` è stata impostata come indice. I dati puliti sono stati salvati nel file `DAX_cleaned.csv`.

Visualizzazione dei dati puliti

DateTime	Open	High	Low	Close	Unnamed: 5
02/01/2020 01:15:00 +01:00	13174.0	13194.5	13171.5	13177.5	NaN
02/01/2020 01:15:00 +01:00	13174.0	13194.5	13171.5	13177.5	NaN
02/01/2020 01:16:00 +01:00	13177.0	13185.0	13171.0	13180.5	NaN
02/01/2020 01:16:00 +01:00	13177.0	13185.0	13171.0	13180.5	NaN
02/01/2020 01:17:00 +01:00	13180.5	13181.5	13181.5	13182.5	NaN
02/01/2020 01:16:00 +01:00	13177.0	13185.0	13171.0	13180.5	NaN
02/01/2020 01:16:00 +01:00	13177.0	13185.0	13171.0	13180.5	NaN
02/01/2020 01:16:00 +01:00	13177.0	13185.0	13171.0	13180.5	NaN
02/01/2020 01:15:00 +01:00	13174.0	13194.5	13171.5	13177.5	NaN

Tabella 5.2. Dati puliti dopo il preprocessing

Visualizzazione grafica dei dati puliti

Per comodità mostriamo solo il prezzo delle Open.

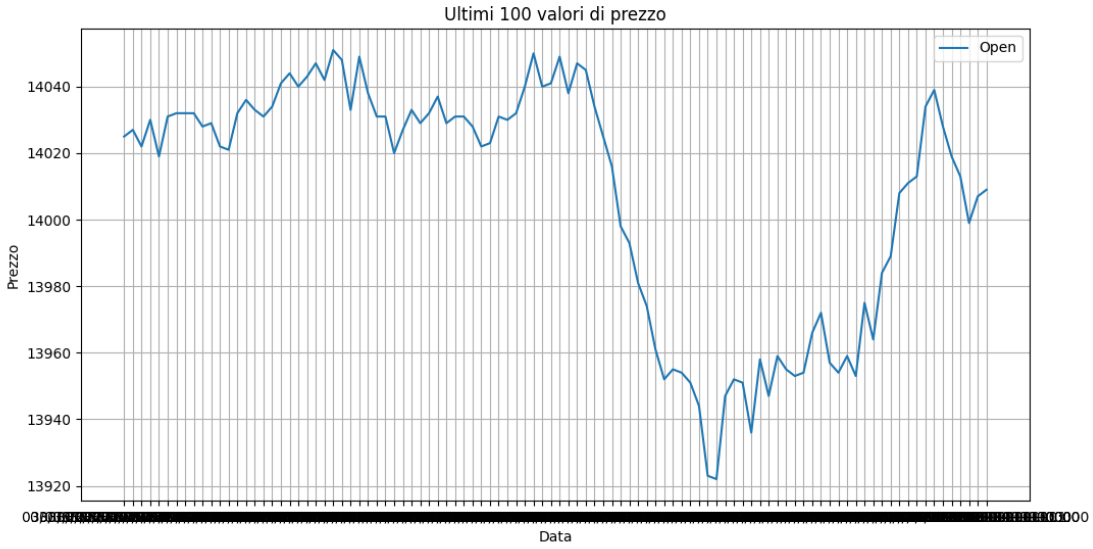

Figura 5.1. Ultimi 100 valori di prezzo

5.3.2 Stima dei Parametri per Diverse Copule

Bibliografia

Umberto Cherubini, Elisa Luciano, and Walter Vecchiato. *Copula Methods in Finance*. Wiley Finance, 2004.

David G. Clayton. A model for association in bivariate life tables and its application in epidemiological studies of familial tendency in chronic disease incidence. *Biometrika*, 65:141–151, 1978.

Simone Demarta and Alexander J. McNeil. The t copula and related copulas. *International Statistical Review*, 73(1):111–129, 2005.

M. J. Frank. On the simultaneous associativity of f(x, y) and x + y - f(x, y). *Aequationes Mathematicae*, 19:194–226, 1979.

Emil J. Gumbel. Bivariate exponential distributions. *Journal of the American Statistical Association*, 55:698–707, 1960.

Harry Joe. *Dependence Modeling with Copulas*. Chapman and Hall/CRC, 2014.

Alexander J. McNeil, Rüdiger Frey, and Paul Embrechts. *Quantitative Risk Management: Concepts, Techniques and Tools*. Princeton University Press, 2015.

Roger B. Nelsen. *An Introduction to Copulas*. Springer, New York, 2nd edition, 2006.