LUISS

Dipartimento di Economia e Finanza

Indirizzo Banche ed Intermediari Finanziari

Il Pair Trading come strumento per la diversificazione del portafoglio

Relatore:
Prof. Nicola Borri
Correlatore:
Prof. Giancarlo Mazzoni

Candidato:
Nicola Gherardi
Matricola 766691

Anno Accademico 2023/2024

Indice

Introduzione

L'obiettivo di questo lavoro di ricerca, vorrebbe essere quello di fornire una risposta plausibile alla seguente domanda: "In che misura una strategia di Pair Trading può essere impiegata, per diversificare un portafoglio e ridurne il rischio?". La diversificazione è una pratica fondamentale per ridurre il rischio in un portafoglio, poiché si basa sulla distribuzione degli investimenti in una varietà di asset.

È necessario innanzitutto definire la metodologia utilizzabile, e capirne le possibilità applicative. Il Pair Trading potrebbe essere un approccio efficace, poiché possiede una caratteristica fondamentale che lo rende particolarmente adattabile a questo scopo. In alcuni casi infatti, è verosimile realizzare una situazione equiparabile all'arbitraggio, ossia alla possibilità di conseguire un guadagno, annullando quasi del tutto il rischio di mercato.

L'elemento teorico di partenza , consiste nel selezionare coppie di titoli appartenenti a diverse asset class, come ETF, obbligazioni, azioni, strumenti derivati, i cui rendimenti siano in stretta relazione l'uno con l'altra. La possibilità d'identificare il momento in cui tale relazione possa essere interrotta, consente di scommettere sul loro futuro ri-allineamento. Nell'ottica di una gestione più ampia del portafoglio, il rapporto temporale a cui si fa riferimento, non può che essere quello di lungo periodo; il breve periodo infatti, con molta probabilità, non permetterebbe ogni volta il riallineamento dei rendimenti degli asset presi in considerazione.

Per applicare efficacemente tale strategia, è utile richiamare alcune nozioni di natura statistico-econometrica, all'interno dell'analisi, che ben si prestano a spiegare il comportamento di due asset. È necessaria un'analisi empirica che coinvolga l'utilizzo del concetto di *cointegrazione*, tema centrale quest'ultimo per la fattibilità del metodo proposto, come pure un'analisi di sensitività, consistente nella verifica del metodo proposto, in termini di studio degli effetti impattanti all'interno di un portafoglio differenziato.

In tale contesto diviene importantissimo avere a riferimento la letteratura prodotta sul tema, tale da offrire notevoli spunti di partenza, adattabili alle tecnologie innovative (su tutte il machine learning), che permettano di realizzare in modo migliore e più efficace, gli obiettivi degli stessi metodi tradizionali.

La ricerca può essere suddivisa nel modo seguente:

Il primo capitolo è dedicato ad un'analisi approfondita delle strategie di investimento e in particolare, alla tecnica del Pair Trading. Cominceremo con una panoramica generale delle difficoltà e delle sfide incontrate dagli investitori nel contesto attuale, caratterizzato da elevati tassi di inflazione e instabilità dei tassi di interesse. Esploreremo poi i concetti fondamentali del rischio specifico e sistemico, delineando come le strategie *market neutral* possono essere utilizzate per mitigare questi rischi. Inoltre, discuteremo l'evoluzione storica del Pair Trading, partendo dalle prime applicazioni della *Principal Component Analysis (PCA)* fino agli sviluppi più recenti nel campo del trading algoritmico ad alta frequenza fornendo una visione d'insieme delle teorie e delle pratiche che hanno plasmato questa strategia attraverso la citazione di studi e contributi significativi come quelli di Avellanada et al., Ian T. Jolliffe, e il lavoro pionieristico di Gatev.

L'obiettivo del secondo capitolo è quello di spiegare nel dettaglio il funzionamento della strategia del Pair Trading. I riferimenti e le metodologie utilizzate, consistono in tecniche di natura statistico-econometrica dell'ampia letteratura prodotta sull'argomento, e nell'adozione di tecniche innovative, che tengano primariamente conto degli elementi di sviluppo al trading algoritmico proposti negli ultimi anni, come le tecniche avanzate del machine learning e delle reti neurali, strumentalmente alla previsione dei ritorni degli asset, oltre che all'analisi delle componenti principali (PCA), per decomporre i rendimenti delle azioni in componenti sistematiche e idiosincratiche. Questi metodi variano in efficacia a seconda dell'accuratezza nella gestione del rischio e dell'identificazione delle coppie di asset.

Nel terzo capitolo vengono delineate le modalità operative d'implementazione della strategia di Pair Trading, con l'utilizzo dell'approccio della *cointegrazione*, in grado di offrire una solida base teorica e tecnica che consenta d'individuare e sfruttare le discrepanze temporanee nei prezzi di due securities, e con la possibilità al mantenimento di una relazione di equilibrio nel lungo periodo.

Dopo aver affrontato tutti gli argomenti teorici necessari per costruire la strategia di Pair Trading, il quarto capitolo si concentra sull'analisi empirica, eseguita considerando il periodo storico dal 31 aprile 2019 al 31 aprile 2024 per lo studio del comportamento degli asset selezionati. L'oggetto è una categoria specifica di asset all'interno di un mercato: si tratta di alcune azioni quotate nell'S&P 500, l'indice di borsa americano che comprende le 500 società statunitensi con la maggiore capitalizzazione.

Nel quinto capitolo esamineremo i risultati e la gestione del rischio della strategia implementata. Dopo aver spiegato il funzionamento della strategia, analizzeremo i risultati ottenuti durante il periodo di validazione. Ci concentreremo sulla definizione delle trade, la gestione delle posizioni e il valore finale del portafoglio. Successivamente, esamineremo i rendimenti del portafoglio e discuteremo delle considerazioni sul rischio.

Il sesto ed ultimo capitolo sarà dedicato alla verifica dell'effetto del Pair Trading in un portafoglio. Proporrò una risposta alla domanda iniziale contestualizzandola in una prospettiva operativa della diversificazione del portafoglio.

Capitolo 1

La definizione del contesto

L'idea alla base degli investimenti nei mercati finanziari, è quella insita nella possibilità di impiegare denaro in strumenti finanziari con l'intento di generare un profitto positivo attraverso l'acquisto di asset sottovalutati e conseguentemente la vendita di quelli considerati sopravvalutati. Comprendere anticipatamente le tendenze finanziarie nella loro dinamicità, non è affatto un compito banale, in quanto richiede un'attenta analisi dei portafogli in modo approfondito con la finalità di conoscere la qualità del sottostante, considerando una serie di variabili macroeconomiche dipendenti anche dal periodo storico di riferimento.

Risulta particolarmente difficile trovare delle strategie di investimento, che abbiano dei buoni ritorni e che comportino un basso profilo di rischio. Questa è generalmente la principale sfida affrontata dai vari trader, e investitori sul mercato.

In particolare nell'attuale periodo storico, lo svolgimento di questo tipo di attività è divenuto ancor più arduo, se si considerano fattori come gli alti tassi d'inflazione, la mancanza di stabilità dei tassi di interesse presenti sul mercato che hanno raggiunto livelli senza precedenti, facendo conseguentemente salire i rendimenti dei titoli di Stato, generalmente considerati risk free in condizioni normali di mercato, e quindi bassi rispetto a investimenti in asset class quali obbligazioni o azioni. È dunque evidente, per gli investitori ma anche per le famiglie, la necessità di mettere i propri risparmi al riparo dall'inesorabile perdita di valore, che questi subiscono nel tempo a causa dell'andamento dell'economia globale. Allo stesso modo, è essenziale neutralizzare una parte sostanziale del rischio sostenuto dagli investitori stessi, ossia quello causato da dinamiche del tutto imprevedibili (ne è un esempio la pandemia da Covid 19) determinanti un crollo sostanziale del valore degli asset di ciascuno.

Tuttavia, quand'anche effettuata con finalità di copertura, ogni strategia di investimento espone l'investitore ad una molteplicità di rischi, che coinvolgono sia l'andamento generale del mercato, sia gli asset specificamente negoziati. Infatti, quando si parla di un qualsiasi tipo di asset o portafoglio di strumenti finanziari su cui è possibile investire nel mercato, due sono le componenti di rischio da considerare:

- rischio **specifico**, anche detto *idiosincratico*, relativo allo strumento o agli strumenti sui quali si sta basando l'intera strategia. Tale rischio non è eliminabile, ed è specifico per ciascun titolo a cui si fa riferimento;

- rischio **sistematico**, quasi indipendente dalla natura dei titoli coinvolti, in quanto comune ad ognuno di essi ma, in quanto tale, eliminabile adottando delle strategie che neutralizzano il rischio di mercato nel portafoglio.

Questo espone senza dubbio il soggetto che opera sui mercati, ad una molteplicità di rischi che devono essere posti sotto un'attenta analisi e mitigati.

In generale, l'obiettivo di una strategia *market neutral* è quello di rimuovere la componente di rischio sistemico derivante dal mercato. È noto che, in accordo con il *Capital Asset Pricing Model o CAPM* proposta da Shapre (1964) [25], il rendimento di qualunque asset può essere modellato in funzione del premio per il rischio di mercato; più nello specifico, definendo con r il rendimento di un titolo, e secondo la riformulazione del modello fornita da Jensen (1968) [15] tale rendimento può essere scomposto nel modo seguente:

$$(r_{i,t} - r_{rf,t}) = \alpha_i + \beta_i E[r_{m,t} - r_{rf,t})] + \varepsilon_{i,t} \tag{1.1}$$

dove:

- α quantifica il rendimento in eccesso della security o del portafoglio rispetto all'attività priva di rischio

- β indica la sensibilità dell'attività oggetto di analisi rispetto al mercato definita come la reattività di questa a movimenti di mercato. Ad esempio, se il β è pari a 1, il titolo è perfettamente sensibile e coordinato al mercato: vorrà dire quindi per un movimento a rialzo o al ribasso del mercato di riferimento che per semplicità è possibile riassumere in un indice come l'S&P 500 corrisponderà un aumento o una diminuzione della componente di rendimento sistematica dello stesso ammontare. Questa correlazione può essere anche negativa, e quindi inversamente proporzionale al mercato, e può assumere diversi valori a seconda del livello di sensibilità accennati in precedenza;

- r_m rappresenta il rendimento del mercato rappresentato come proxy ad esempio da un indice di borsa (come appunto S&P 500;

- r_f descrive il rendimento specifico di un asset risk free che è possibile approssimare ad un titolo di Stato con una maturity di un anno o inferiore.

- ε mostra il rischio idiosincratico che invece è che si raggiunge attraverso la diversificazione.

All'interno di questo modello, i portafogli *market neutral* sono definiti come quelli per i quali la sensibilità al mercato e quindi il β è pari a 0, rendendo il rendimento dell'intero portafoglio dipendente dalle sole componenti idiosincratiche.

1.1 Definizione e caratteristiche del Pair Trading

Come anticipato nell'introduzione, il Pair Trading è una strategia che mira a trarre profitto dalla differenza di prezzo tra due titoli che sono correlati. Se dovessimo descriverla sommariamente, potremmo definirla come una metodologia che prevede l'acquisto di un asset e la vendita simultanea dell'altro sfruttando le oscillazioni di prezzo tra i due ritenute momentanee, seguendo delle precise indicazioni fornite dai segnali generati dai vari algoritmi.

Innanzitutto, si parte dall'identificazione dei titoli. In questa fase, tra i vari indici statistici che rendono possibile l'analisi, uno è ritenuto fondamentale, la *cointegrazione*. Tale indicatore viene utilizzato con la finalità di verificare una delle condizioni imprescindibili, affinché il processo di selezione possa andare avanti e riscontrare la presenza di una relazione tra le due securities. Questa relazione va ricercata nel lungo periodo con l'utilizzo delle serie storiche.

Una delle caratteristiche che rende particolare tale metodo è quella di essere *market neutral* in quanto coinvolge sia l'acquisto che la vendita simultanea dei titoli, e permette di generare profitti indipendentemente dalle condizioni di mercato, annullando la componente sistematica dell'investimento. Questa neutralità tuttavia, può essere raggiunta solo attraverso una rigorosa gestione del rischio in cui le posizioni aperte sono attentamente monitorate (e nel caso specifico automatizzate), implementando le misure protettive da mettere in atto in caso di movimenti imprevedibili e sfavorevoli del mercato.

Allo stesso tempo però, è bene non dimenticare dei *transaction cost*, che sebbene oggi siano ridotti grazie alla diffusione delle numerose piattaforme dove è possibile effettuare trading, sono una componente non trascurabile di costo.

La considerazione di questi fattori, determina la selezione di coppie di titoli; l'operatore acquista il titolo sottovalutato e vende quello sopravvalutato, ritenendo che la divergenza dei prezzi sia solo temporanea poiché causata da motivi di diversa natura, tra i quali possono esserci l'avere appreso notizie finanziarie riguardanti uno dei due titoli o cambiamenti di domanda/ offerta rispetto a uno dei sottostanti.

In generale, possiamo riassumere le caratteristiche del Pair Trading come segue:

- i segnali di acquisto o di vendita si basano su una regola ben definita e non su scenari macroeconomici e aspetti finanziari della società su cui si fonda il titolo. Tuttavia, sebbene questo aspetto possa rappresentare un vantaggio dal punto di vista computazionale e interpretativo, dall'altro potrebbe trasformarsi in un importante limite dato che tali eventi non sono affatto trascurabili in quanto il più delle volte possono essere alla base di eventuali scostamenti dall'equilibrio di lungo periodo, con la conseguente impossibilità di attuare la strategia in maniera efficace;

- i ritorni del portafoglio sono *market neutral*, cioè non correlati con il mercato ma oggetto di rischi di natura *idiosincratica*: se infatti si manifestasse sul mercato uno shock macroeconomico i titoli subirebbero perdite e guadagni che si andrebbero di conseguenza a compensare;

- la strategia è *autofinanziata* nel senso che le posizioni (lunghe e corte) vengono aperte per lo stesso controvalore sicché il portafoglio viene di fatto costruito a costo zero.

1.2 La letteratura del Pair trading

La disponibilità di tecnologie di calcolo sempre più precise ed efficaci, il miglioramento sotto il profilo dell'accessibilità alle piattaforme di trading, e soprattutto l'aumento della competitività dei broker con il conseguente abbassamento dei costi di transazione, hanno consentito nel tempo lo sviluppo di una letteratura sempre più ricca e dettagliata in grado di sviluppare metodi di lettura sempre più efficaci della tecnica che si sta descrivendo. Non è ben chiaro chi sia stato il pioniere di questa tecnica, quel che è certo è che una strategia precorritrice del Pair Trading, è stata sicuramente quella proposta dal noto gruppo di trader matematici e quant guidato da Nunzio Tartaglia presso Morgan Stanley. Nel 1987 il team guidato da Tartaglia riuscì nella grande impresa di contabilizzare un profitto societario pari a 50 milioni di dollari. Tuttavia, non si trattava di una strategia esattamente pari a quella oggetto dello studio, piuttosto di una *Principal Component Analysis (PCA)* effettuata su due ETFs. È necessario partire proprio dalla PCA, per poter determinare gli sviluppi di questo approccio di investimento.

1.2.1 Principal Component analysis

La *Principal Component Analysis (PCA)* è un metodo statistico che viene utilizzato con la finalità di ridurre la dimensionalità dei dati utilizzati all'interno di un'analisi. Come è noto infatti, la maggiore quantità di dati si traduce molto spesso in una maggiore complessità di calcolo, e conseguentemente in un incremento sostanziale della probabilità di ottenere outliers nelle osservazioni.

L'applicazione del metodo determina la trasformazione di un insieme di dati costituiti da variabili correlate, in un paniere di variabili linearmente indipendenti chiamate componenti principali. Queste componenti vengono costruite e ordinate in maniera tale che le prime conservino la maggior parte della variazione, e quindi delle informazioni presenti nei dati originali. Nello specifico, se volessimo affrontare il problema seguendo un approccio vettoriale, sarebbe necessario dapprima individuare le direzioni o assi, lungo i quali i dati variano maggiormente. Questi ultimi vengono dunque proiettati sugli assi, riducendo la dimensionalità dell'insieme degli stessi pur preservando il maggior numero delle variazioni originali. Questo approccio utilizza i prezzi storici dei titoli, su un database di N asset. Come spiegato da Avellanada et al. (2010) [1] è possibile rappresentare i dati contenenti i ritorni delle singole securities in una data t_0, tornando indietro di M +1 giorni come una matrice:

$$R_{ik} = \frac{S_i(t_0 - k - 1)\Delta t - S_i(t_0 - k\Delta t)}{S_i(t_0 - k\Delta t)}, \quad k = 1, \ldots, M \quad i = 1, \ldots, N \tag{1.2}$$

dove S_{it} è il prezzo dell'azione al tempo t aggiustato per i dividendi e $\Delta t = \frac{1}{252}$[1]. A questo punto, sapendo che la volatilità non è mai la stessa tra titoli diversi, è necessario standardizzare i rendimenti:

$$Y_{ik} = \frac{R_{ik} - \bar{R}_i}{\bar{\sigma}_i} \tag{1.3}$$

dove:

$$\bar{R}_i = \frac{1}{M} \sum_{k=1}^{M} (R_{ik}) \tag{1.4}$$

e:

$$\bar{\sigma}_i{}^2 = \frac{1}{M-1} \sum_{k=1}^{M} (R_{ik} - \bar{R}_i)^2 \tag{1.5}$$

La matrice che esprime le correlazioni tra i dati è così definita:

$$\rho_{ij} = \frac{1}{M-1} \sum_{k=1}^{M} Y_{ik} Y_{jk} \tag{1.6}$$

che è simmetrica e definita non negativamente.

La dimensione di ρ è tipicamente 500 x 500 o 1000 x 1000, un numero comunque piccolo se confrontato con il numero di parametri che devono essere stimati. Infatti, se da un lato si considera il fatto che dati economici distanti nel tempo non hanno alcun senso nell'analisi in quanto irrilevanti, dall'altro è pur vero che se venissero considerati solo i rendimenti dell'anno passato non si avrebbero a disposizione dati a sufficienza. È qui che interviene la *PCA*: considerando gli autovettori della matrice delle correlazioni, e ordinando gli autovalori della stessa in ordine decrescente tale per cui:

$$N \geq \lambda_1 > \lambda_2 \geq \lambda_3 \geq \ldots \geq \lambda_N \geq 0.$$

[1] Vengono presi come riferimento 252 giorni perché si fa riferimento ai trading days in un anno

In tal modo è possibile ordinare l'autovettore corrispondente:

$$v^{(j)} = (v_1^{(j)}, \ldots, v_N^{(j)}), j = 1, \ldots, N$$

E quindi la trasformazione dei dati originari nella nuova base di componenti principali può essere intesa come:

$$Y = \rho_{ij} * v^{(j)} \tag{1.7}$$

Dove Y è la matrice dei vettori trasformati poiché il prodotto $\rho_{ij} * v^{(j)}$ rappresenta l'applicazione di una trasformazione lineare definita dagli autovettori delle osservazioni originali. Un autore che discute questo principio è Ian T. Jolliffe nel suo libro "Principal Component Analysis" (2002) [17]. Jolliffe è uno dei principali esperti nel campo della PCA, e fornisce una trattazione dettagliata delle proprietà matematiche e delle applicazioni pratiche di questa tecnica. Nella sua pubblicazione, spiega il concetto di massimizzazione della varianza sulle prime componenti principali, come fondamento teorico della PCA.

Il risultato è una nuova rappresentazione dei dati in termini di componenti principali, dove la varianza è massimizzata sulla prima componente, poi sulla seconda in maniera via via decrescente. Obiettivo infatti della *PCA* è quello di scegliere le componenti principali, in modo da trovare quelle che catturano le direzioni di massima varianza nei dati, mentre le successive componenti spiegano varianze sempre minori.

1.2.2 L'evoluzione del Pair trading

A seguito dei progressi nel campo dell'informatica e non solo, si è assistito ad un incremento del numero di transazioni sempre più efficienti nel mercato. In particolare, questo ha sviluppato un interesse sempre maggiore nella realizzazione di strategie di trading algoritmico ad alta frequenza.

Tra i personaggi più influenti in materia, riconosciuto come uno degli esponenti più importanti della letteratura, vi è certamente Gatev [10] che, insieme ad altri ricercatori, esplorò nei suoi studi differenti dimensioni del Pair Trading, inclusi i criteri per la selezione delle coppie di azioni, le metodologie per la costruzione del portafoglio e strategie di gestione del rischio, considerano variabili anche i costi di transazione e le commissioni, che, come è ovvio, possono influenzare significativamente la redditività delle strategie di Pair Trading nell'ambiente reale del mercato.

La prassi della valutazione degli asset si articola su due fronti: quello assoluto e quello relativo. La valutazione assoluta dei titoli si fonda su parametri fondamentali quali il flusso di cassa futuro scontato, un processo notoriamente ostico caratterizzato da un'ampia tolleranza degli errori, principalmente per la quantità di assunzioni che vengono effettuate.

La valutazione relativa, pur essendo più accessibile, presenta delle complessità proprie, suggerendo che due titoli considerati sostanzialmente equivalenti, dovrebbero concretizzarsi nel medesimo prezzo di scambio. Di conseguenza, la valutazione relativa ammette la possibilità di bolle speculative nell'economia, sebbene non garantisca per forza l'opportunità di arbitraggi o speculazioni redditizie. Il principio del Prezzo Unico [LOP] - e la sua variante ”quasi-LOP” applicabile alla valutazione relativa - sottolinea questa dinamica, ammettendo che il prezzo stabilito potrebbe rivelarsi errato.

Tale concetto, espresso da Chen e Knez (1995) [3] e ripreso da Gatev et al. (2006) [10] nel suo paper, si estende anche a mercati integrati, ossia strettamente interconnessi e collegati tra loro, i quali dovrebbero assegnare ai payoffs dei prezzi simili.

Le conclusioni principali dello studio di Gatev et al., permettono di osservare che il Pair Trading può essere una strategia redditizia, quando il mercato si presenta particolarmente volatile e non efficiente, sottolineando che la redditività della strategia può essere influenzata da variabili come i costi di transazione, e la presenza di trend di lungo periodo nel mercato.

La loro ricerca [10] ha infatti contribuito in modo significativo alla comprensione pratica del Pair Trading come strategia di investimento. La loro analisi empirica fornisce agli investitori e agli operatori di mercato, un quadro utile per valutare l'efficacia e i rischi associati a questa strategia nell'ambiente finanziario reale.

Capitolo 2

Implementazione del Pair Trading: approcci e tecniche

L'intento di questo capitolo è quello di spiegare nel dettaglio il funzionamento della strategia del Pair Trading. Alcuni di questi argomenti sono stati parzialmente presentati nelle fasi iniziali della ricerca, mentre altri sono l'espressione di nuovi approcci e tecniche emergenti della letteratura recente. L'obiettivo è proporre un'alternativa d'investimento innovativa, che tenga conto delle problematiche attuali, e che sia in grado di offrire nuovi elementi di sviluppo al campo del trading algoritmico, in particolare attraverso l'adozione integrativa di modelli costituiti da algoritmi di trading automatico per rendere possibile la previsione degli andamenti di mercato e, soprattutto, ottenere un'adattabilità reattiva in tempo reale.

2.1 Idee alla base dell'arbitraggio

Come già in parte accennato, il Pair Trading è una strategia che rientra nella categoria degli arbitraggi statistici, basata sull'identificazione di discrepanze di prezzo di diversi strumenti finanziari. Tali discrepanze vengono individuate attraverso dei modelli statistico-econometrici come il modello della cointegrazione, ossia la relazione tra variazione di prezzo di un'attività e di un'altra correlata, rispetto alle quali gli operatori si aspettano di poter trarre profitto dal probabile futuro riallineamento, con l'azzeramento nel lungo periodo di queste differenze. In modo simile a qualsiasi forma di arbitraggio, comporta inevitabilmente l'assunzione di rischi in particolare dei profitti non garantiti, che saranno oggetto di analisi nei capitoli successivi. Alla base di tale metodo è bene considerare ed adottare dei piani di *risk management*, che permettano di bilanciare il portafoglio.

Diversi sono i metodi che permettono di raggiungere questo risultato; ognuno di essi possiede inevitabilmente punti di forza e criticità. Krauss (2016) [20] evidenzia cinque approcci rispetto ai quali è possibile trarre spunto per l'applicazione della strategia.

2.1.1 Distance approach

Il *Distance approach*, proposto da Do and Faff (2010) [6], rappresenta uno dei metodi più seguiti in letteratura. Si procede innanzitutto dividendo i periodi in cui vengono studiati gli asset in *formation period*, dove le varie metriche che misurano la distanza tra i prezzi degli asset vengono utilizzate al fine di identificare i titoli che 'co-muovono', e *trading period* in cui vengono stabilite ed utilizzate delle regole che prendono a riferimento delle *threshold* non parametriche, che permettono di generare dei segnali di trading.

Una metrica molto utilizzata al fine di verificare le cosiddette distanze, è la somma dei quadrati delle differenze di prezzo tra i prezzi normalizzati di due asset nel formation period effettuata nel modo seguente:

$$P_t^A = \frac{P_t^A - P_0^A}{P_0^A} \tag{2.1}$$

e

$$P_t^B = \frac{P_t^B - P_0^B}{P_0^B} \tag{2.2}$$

È possibile dunque calcolare la somma delle differenze al quadrato come:

$$SSD_{A,B} = \sum_t^N (p_t^A - p_t^B)^2 \tag{2.3}$$

dove N è il numero totale dei *trading days* nel *formation period*. Si definisce una buona coppia quella che minimizza questa differenza. Se N è il numero di azioni in considerazione è necessario anche calcolare $\frac{Nx(N-1)}{2}$ differenze di prezzo normalizzate. A questo punto, non resta che calcolarne la *deviazione standard* σ, la quale determina la regola di entrata e di uscita dalla trade. Infatti, Gatev et al. (2006), identificano una soglia pari a due volte la deviazione standard come quella oltre la quale le posizioni long/short vengono aperte e entro la quale vengono invece chiuse. La deviazione standard del quadrato delle differenze di prezzo si calcola come:

$$\sigma_{SSD_{A,B}} = \frac{1}{N-1} \sum_{t=1}^{N} [(P_t^A - P_t^B)^2 - \overline{(P_t^A - P_t^B)}^2] \tag{2.4}$$

È quindi possibile definire i segnali di entrata e uscita della trade in ossequio con quanto dimostrato dagli autori precedentemente menzionati:

- *Open Trade signal* se: $SSD_{A,B} > 2\sigma SSD_{A,B}$
- *Close Trade signal* se: $SSD_{A,B} \leq 2\sigma SSD_{A,B}$

2.1.2 Metodo di cointegrazione

L'applicazione di questo metodo è stata proposta precedentemente da numerosi autori, tra cui Vydyamutri (2004) [27]. Nel suo paper si basa sull'approccio proposto da Engle e Granger (1987) [9] parametrizzando la strategia di Pair Tradig e facendo riferimento alla *cointegrazione*. Prima di declinare il concetto appena esposto, è bene considerare un argomento affine: l'*integrazione*.

Considerando una serie temporale y_t, con $t = 1, \ldots, T$ è possibile definire un operatore di retrocessione o Lag L che, in questa applicazione, non fa altro che riportare l'argomento al periodo precedente:

$$Ly_t = y_{t-1} \tag{2.5}$$

con y_t che viene detto integrato di ordine n se $(1 - L^n)y_t$ è una matrice di covarianza stazionaria. In questo caso si parla di una serie integrata di ordine n $I(n)$; sebbene a volte troviamo alcune serie temporali integrate di ordine n, la loro combinazione lineare può avere un ordine inferiore ad n. In quel caso le due serie storiche potrebbero essere cointegrate. La *cointegrazione* è una relazione statistica, che può essere osservata tra serie storiche, che hanno lo stesso ordine di integrazione *n* (*I(n)*), che se linearmente combinate, producono un'unica serie storica integrata di ordine *n-b*, dove $b > 0$, dove il parametro b rappresenta il numero di differenze-differenziate necessarie per rendere la combinazione lineare delle serie storiche stazionaria, in altre parole indica quante volte è necessario differenziare la combinazione lineare per raggiungere l'ordine di integrazione I(n-b), che è inferiore all'ordine di integrazione I(n) delle serie storiche originarie.

Al fine di applicarla al Pair Trading è necessario avere due serie storiche di prezzi, che siano integrati di ordine 1 (*I(1)*), se combinati linearmente producono una serie temporale stazionaria anche detta *I(0)*. Un processo stocastico y_t è stazionario *in senso debole I(0)*, se è caratterizzato da una media ed una varianza finite e costanti nel tempo tali per cui:

$$E[y_t] = \mu \tag{2.6}$$

$$Var(y_t) = \sigma^2 \tag{2.7}$$

con autocovarianza o autocorrelazione *indipendenti* dal tempo t.

$$\gamma_k = E[(y_t - \mu)(y_{t-k} - \mu)], \forall t \tag{2.8}$$

$$\rho_k = \frac{\gamma_k}{\gamma_0} = \frac{\gamma_k}{\sigma^2}, \forall t \tag{2.9}$$

Da qui la definizione del concetto di *mean reverting process*, ovvero quando i rendimenti di un processo, come nel caso in esame, si discostano momentaneamente dal valore medio stimato, si attende sempre in futuro una loro convergenza rispetto al valore medio.

Quindi, date le serie storiche di prezzi di due asset di tipo *I(1)* per definizione non stazionarie in quanto come noto seguono un processo di random walk, l'obiettivo è quello di cercare dei trend comuni che leghino l'andamento delle stesse nel tempo; si cerca un vettore v che renda l'intero sistema stazionario. Considerando due strumenti finanziari A e B e volendo investire equamente nei due titoli tale per cui $P_t^A = \alpha P_t^B$, con α fattore di scala, è possibile riscrivere l'equazione in termini logaritmici:

$$log(P_t^A) - log(P_t^B) = log(\alpha) \tag{2.10}$$

Di conseguenza, se l'investimento ha durata di un periodo, il relativo rendimento sarà:

$$log(\frac{P_t^A}{P_{t-1}^A}) - log(\frac{P_t^B}{P_{t-1}^B}) \tag{2.11}$$

Considerando esclusivamente la log-differenza dei prezzi al tempo t è possibile scrivere la differenza come segue:

$$[log(P_t^A) - log(P_t^B] \tag{2.12}$$

con $\varepsilon_t \sim N(0, \sigma^2)$.

La relazione di equilibrio di lungo periodo tra i prezzi dei due asset,viene analizzata attraverso l'Augmented Dickey Fuller Test (d'ora in poi ADF). L'applicazione dello stessa sui residui, permette di verificare la stazionarietà nel lungo periodo.

È possibile dunque scrivere lo spread come:

$$spread_t = log(P_t^A - \gamma log(P_t^B) = \alpha + \varepsilon_t \tag{2.13}$$

Dire che due serie sono cointegrate, equivale ad evidenziare un rapporto di dipendenza. Se ciò è vero, allora è possibile scrivere il logaritmo del prezzo di un titolo in funzione dell'altro, attraverso una combinazione lineare dei due:

$$log(P_t^A) = \alpha + \gamma log(P_t^B) + \varepsilon_t \tag{2.14}$$

In questa trade si evidenzia una posizione long sull'asset A di un'unità, ed una corta sull'asset b di γ unità, che danno luogo ad un valore atteso dell'investimento pari ad $\alpha +$ un termine d'errore ε_t che dipende dall'aleatorietà del mercato.

2.1.3 Serie temporali

Tale approccio, proposto per la prima volta da Elliott et al. (2005) [8], è maggiormente incentrato sullo spread attraverso un processo di tipo *mean reverting* ignorando il *formation period.* Si assume quindi, che la variabile latente x_k possa essere descritta come segue:

$$x_{k+1} - x_k = (a - bx_k)\tau + \sigma\sqrt{\tau}\varepsilon_{k+1} \tag{2.15}$$

con $\varepsilon_t \sim N(0,1)$ e τ che rappresenta l'intervallo tra le osservazioni al tempo k e al tempo $k+1$. Questo processo è mean reverting, nel senso che nel lungo periodo tende alla sua media $\mu = \frac{a}{b}$. Può essere riscritto come:

$$x_{k+1} = A + Bx_k + C\varepsilon_{k+1} \tag{2.16}$$

dove $A = \alpha\tau, B = 1 - b\tau, e\ \ C = \sigma\sqrt{\tau}$

Se si considera il tempo continuo è possibile descrivere il processo come di tipo *Ornstein-Uhlenbeck o (OU process)*:

$$dX_t = \rho(\mu - X_t)dt + \sigma dW_t \tag{2.17}$$

dove dW_t è il moto geometrico Browniano in uno spazio di probabilità. Il parametro $\mu = \frac{a}{b}$ rappresenta la media e $\rho = b$ descrive la velocità con cui il processo tende alla sua media. La seconda componente di un modello di questo tipo, è l'equazione di misura; lo spread osservato è definito come somma della variabile di stato x_k e un rumore Gaussiano tale per cui $w_k \sim N(0,1)$:

$$y_k = x_k + Dw_k,\ \ con\ \ D > 0 \tag{2.18}$$

Ad ogni modo, non c'è un criterio fisso in questo caso che permetta la costruzione della strategia. L'aspetto fondamentale a cui far riferimento è lo *spread* tra gli asset, scelti sulla base di un'analisi a priori, a seguito di un processo stazionario, le cui dinamiche sono modellate e stimate in modo da dare la possibilità di prevedere l'andamento futuro dello spread. La regola identificata da Krauss (2017 è la seguente: viene aperta la trade quando $y_k \geq \mu + c(\frac{\sigma}{\sqrt{2\rho}})$ o quando $y_k \leq \mu - c(\frac{\sigma}{\sqrt{2\rho}})$ dove c rappresenta un parametro fisso per cui Elliott et al. (2005) [8] non forniscono alcuna indicazione sulla sua determinazione.

2.1.4 Stochastic spread method

Il contributo sostanziale per tale approccio è quello fornito da Jurek e Yang (2007) [18] esposto nel paper di Krauss (2017) [20], in cui viene esaminata la situazione di un investitore che deve allocare denaro,investendo in una coppia di titoli che generano un processo di tipo *mean reverting*, oppure investendo in titoli privi di rischio. Nel fare questo, l'investitore fronteggia due rischi principali:

- rischio di orizzonte, legato all'incertezza rispetto a quando la relazione di mispricing evidenziata possa tornare alla normalità;
- rischio di divergenza, ovvero che la distribuzione diverga dalla situazione di equilibrio.

Il *mispricing* può aumentare in maniera significativa prima di convergere, ed il rischio legato ad entrambi è assunto dal processo definito nel paragrafo precedente: *OU process*. Il primo può essere misurato mediante l'incertezza nel periodo considerato, rispetto al tempo necessario prima che lo *spread* ritorni al livello di equilibrio. Il secondo può essere misurato dalla varianza della distribuzione, ovvero del suo valore massimo (o minimo), rispetto al valore medio di lungo periodo.

A partire da questo ragionamento, è possibile definire l'evoluzione del prezzo dell'indice di mercato:

$$\frac{dP_{m,t}}{P_{m,t}} = (r_f + \mu_m)dt + \sigma_m dB_t \tag{2.19}$$

dove, μ_m è il premio al rischio, σ_m è la volatilità del mercato, r_f è il tasso privo di rischio e B_t indica il moto geometrico Browniano che ne regola parte dell'evoluzione. Quindi, definendo con $P_{1,t}$ e $P_{2,t}$ i prezzi di due asset rischiosi che seguono le seguenti dinamiche:

$$\frac{dP_{1,t}}{P_{1,t}} = (r + B\mu_m)dt + \beta\sigma_m dB_t + \sigma dZ_t + bdZ_{1,t} - \lambda_1 X_t dt \tag{2.20}$$

$$\frac{dP_{2,t}}{P_{2,t}} = (r + B\mu_m)dt + \beta\sigma_m dB_t + \sigma dZ_t + bdZ_{2,t} - \lambda_1 X_t dt \tag{2.21}$$

$$X_t = ln(P_{1,t}) - ln(P_{2,t}) \tag{2.22}$$

Dove, λ_i, β, b e σ sono costanti e $Z_t, Z_{i,t}$ sono mutualmente indipendenti e seguono il modo geometrico Browniano per $i = 1,2$ e X_t è il termine di errore.

Dato che la somma di λ_1 e λ_2 è maggiore di 0, X_t è stazionaria e i prezzi logaritmici sono cointegrati.

Gli stessi autori Jurek e Yang (2007) [18] derivano l'equazione di Hamilton-Jacobi-Bellman (o HJB) sotto il vincolo di funzione di utilità del tipo power utility[1], in un orizzonte temporale finito. Ne segue che la strategia ottimale, è quella di avere in portafoglio, allo stesso tempo, entrambi gli asset rischiosi (in acquisto o in vendita), anche se i prezzi dovessero convergere. In secondo luogo, talvolta è ottimale per gli investitori, detenere anche solo un titolo, però in netto contrasto con la neutralità tipica del Pair Trading.

[1]La funzione di utilità power utility, anche detta di Epstein-Zin è una funzione la cui elasticità di sostituzione intertemporale dei consumatori è il reciproco del coefficiente di avversione al rischio

2.1.5 Altre strategie

Tra quelle identificate da Krauss (2017) [20], possono essere menzionate altre metodologie meno diffuse, ma che potrebbero essere utilizzate per sviluppare una situazione di Pair Trading.

Machine Learning e reti neurali

Si tratta dell'approccio identificato da Huck (2010) [12], il quale parte dalla previsione dei ritorni, proiettandoli una settimana in avanti, e basandosi sui ritorni passati:

$$\hat{x}_{ij,T+1} = \hat{x}_{i,T+1}|\iota_{ij,T} - \hat{x}_{j,T+1}|\iota_{ij,T} \tag{2.23}$$

dove $\hat{x}_{i,T+1}|\iota_{ij,T}$ è il ritorno per ogni security condizionato all'informazione passata $\iota_{ij,T}$. Nello specifico viene effettuato uno step di ordinamento nel quale, seguendo uno specifico criterio, il *multicriteria decision method*, Huck (2009) [11] ordina le securities in base al loro valore stimato; gli strumenti finanziari sottovalutati sono considerati tra i primi, mentre quelli sopravvalutati per ultimi. Nell'ultimo step dove si svolgono effettivamente le trades, gli strumenti primi nel rank vengono comprati perchè sottovalutati, viceversa per gli ultimi. Dopo una settimana le posizioni vengono chiuse e si ripete il processo a partire dal ranking.

Approccio PCA

Come menzionato nella sezione 1.2.1, Avellanada e Lee (2010) [1] utilizzarono questo approccio con la finalità di applicare l'arbitraggio statistico alle azioni nel mercato americano per il superamento di un mld USD di capitalizzazione. I due operarono scomponendo il ritorno di ciascuna azione, in componente sistematica e idiosincratica.

$$R_i = \beta_i F + \varepsilon_i \tag{2.24}$$

dove $\beta_i F$ è la componente sistematica legata al mercato e ε_i è quella idiosincratica. Andando a scomporre la prima componente in m fattori e sviluppando un modello multifattoriale si ottiene:

$$R_i = \sum_{j=1}^{m} \beta_{ij} F_j + \varepsilon_i \tag{2.25}$$

Il passo successivo è quello di sviluppare un modello di valutazione relativo, basandosi sul modello multifattoriale. I ritorni delle azioni seguono un processo di questo tipo:

$$\frac{dP_{i,t}}{P_{i,t}} = \mu_i dt + \sum_{j=1}^{m} \beta_{ij} \frac{dI_{j,t}}{I_{j,t}} + dX_{i,t} \tag{2.26}$$

dove il tasso di crescita istantaneo del prezzo dell'azione i al tempo t ($\frac{dP_{i,t}}{P_{i,t}}$), è pari alla somma tra:

- $\mu_i dt$ che rappresenta la componente drift o di trend del sottostante per ogni aumento infinitesimale del tempo t;
- $\sum_{j=1}^{m} \beta_{ij} \frac{dI_{j,t}}{I_{j,t}}$ dove m è il numero di fattori sistematici considerati nel modello, β_{ij} è la sensibilità del prezzo dell'azione i rispetto al fattore j, $\frac{dI_{j,t}}{I_{j,t}}$ rappresenta il tasso di crescita istantaneo del fattore j al tempo t;
- $dX_{i,t}$ è la componente idiosincratica e rappresenta i movimenti del prezzo dell'azione che non possono essere spiegati dai fattori sistematici considerati nel modello.

Capitolo 3

La stretegia del Pair Trading

L'obiettivo di questo capitolo, è quello di delineare il framework di azione rispetto alla strategia di Pair Trading. Nello specifico, l'intera strategia verrà costruita sulla base di uno dei metodi precedentemente menzionati, ovvero mediante l'utilizzo dell'approccio della *cointegrazione*.

Come ripetuto ormai più volte, lo scopo ultimo è quello di trarre profitto attraverso l'identificazione di anomalie temporanee nelle relazioni tra i prezzi di due titoli, che nel lungo periodo invece tornano alla relazione di equilibrio. Talvolta, questo equilibrio viene meno; la situazione diventa tale per cui il prezzo di una security è sopravvalutato e quello dell'altra è sottovalutato rispetto all'andamento dei due asset mostrato nel lungo periodo. La differenza di prezzo viene identificata dallo *spread*, che in termini logaritmici può essere scritta riprendendo l'equazione 2.12 come:

$$spread_t = log(p_t^A) - \gamma log(p_t^B) \tag{3.1}$$

dove γ rappresenta un fattore di scala, ovvero l'ammontare di asset acquistati del titolo B per ogni unità del titolo A.

Vendendo la posizione lunga ed acquistando quella corta, nel momento in cui la relazione che ha portato allo squilibrio momentaneo e quindi alla formazione di uno *spread* considerevole, torna alla normalità.

Naturalmente, si parla di un processo piuttosto lungo del quale di seguito si riporta il tentativo di descriverlo sommariamente:

1. il primo passo consiste nell'identificazione delle *securities* identificabili come potenzialmente cointegrate;

2. successivamente si passa al calcolo statistico, e all'analisi dei dati, quindi alla verifica delle ipotesi precedentemente elaborate;

3. In seguito, si passa all'impostazione delle cosiddette *trading rules*, ovvero all'identificazione delle *threshold* in base alle quali delineare il funzionamento della strategia;

4. Infine, il modello viene ”allenato” attraverso una fase detta appunto di *training*, passando poi per l’effettivo sviluppo del modello ed alla sua implementazione nella fase di *validation*.

3.1 Selezione degli asset

La prima fase in assoluto consiste nell’identificazione di una serie di titoli, che per molteplici caratteristiche, siano esse di natura economica perché riferite allo stesso mercato (ad esempio quello americano) o di natura statistica, possono essere considerati *cointegrati*. L’obiettivo è quindi quello di avere una lista di securities, da considerare come possibili candidate allo sviluppo della strategia. Tale processo inizia con una semplice *pre-selezione* volta all’individuazione sommaria di tecniche semplici e dirette (come quelle relative alla matrice di varianza-covarianza, oppure ai coefficienti di correlazione), tali da non gravare il procedimento, e determinanti l’applicazione della cointegrazione a ciascuna coppia potenzialmente identificata.

In linea con quanto proposto da Huck e Afawubo (2014) [13], diversi sono i metodi che è possibile adottare per la selezione di coppie a cui applicare la strategia in esame. Una delle tecniche più semplici ed utilizzate a partire è il metodo già illustrato per sommi capi al paragrafo 2.1.1 della *distanza minima*. A titolo di semplificazione, l’approccio di Vidyamurthy (2004) [27] si pone come obiettivo quello di produrre una lista ordinata di coppie di titoli basata sul grado di co-movimento. Nello specifico, più alto è lo score restituito dal test, maggiore è il grado di co-movimento, e quindi maggiore è la probabilità che si stia parlando di una coppia utilizzabile per la strategia. In particolare, si parte dal modello di trend comune introdotto per la prima volta da Stock e Watson (1988) [26], e l’idea alla base del ragionamento, è proprio quella per cui una serie temporale può essere espressa come combinazione lineare di una componente *stazionaria* e una *non stazionaria*:

$$x_t = \eta_{x_t} + \varepsilon_{x_t} \tag{3.2}$$

$$y_t = \eta_{y_t} + \varepsilon_{y_t} \tag{3.3}$$

dove η_{x_t} e η_{y_t} rappresentano le componenti non stazionarie e ε_{x_t} e ε_{y_t} quelle stazionarie. Le due serie di prezzi possono essere definite come cointegrate se e solo se esiste un parametro γ tale per cui il loro trend stocastico risulta generato dallo stesso processo non stazionario e differiscono di un solo fattore lineare di scala γ, che possiamo assumere $\in R$, il quale rappresenta anche il coefficiente di cointegrazone:

$$\eta_{x_t} = \gamma \eta_{y_t} \tag{3.4}$$

Pertanto, tale ragionamento può essere spostato sui rendimenti che possiamo scrivere come:

$$r_{x_{t+1}} = \eta_{x_{t+1}} - \eta_{x_t} \tag{3.5}$$

$$r_{y_{t+1}} = \eta_{y_{t+1}} - \eta_{y_t} \tag{3.6}$$

Anche qui è possibile delineare un coefficiente γ allo stesso modo rispetto quanto fatto in precedenza, ossia come:

$$r_{x_{t+1}} = \gamma r_{y_{t+1}} \tag{3.7}$$

Dunque, data una relazione lineare tra rendimenti, il coefficiente di cointegrazione può essere stimato attraverso una regressione lineare semplice:

$$\hat{\gamma} = \frac{cov(r_{x_{t+1}}, r_{x_{t+1}})}{var(r_{y_{t+1}})} \tag{3.8}$$

3.1.1 La minimizzazione dei quadrati delle differenze

È possibile identificare una coppia sulla base del fatto che la stessa minimizzi la somma delle differenze al quadrato dei prezzi dei titoli normalizzati a un euro. Seguendo proprio l'approccio di Gatev et al. (2006) [10], vengono considerati come potenziali coppie i 20 asset la cui *SSD o sum of squared differences* ha la più bassa misura di *SSD*.

$$SSD_{A,B} = \sum_{t=1}^{T} (P_t^A - P_t^B)^2 \tag{3.9}$$

dove P_t^A e P_t^B sono i prezzi normalizzati per le securities A e B nel giorno t e T è il numero di giorni considerati nel *training period* solitamente di una lunghezza fissa tra 1 e 3 anni.

Come proposto da Vidyamurthy (2004) [27], seguendo l'*arbitrage pricing theory o APT*, supponiamo che le azioni A e B abbiano vettori di fattori comuni identici fino a un fattore di scala (γ), il che significa che il vettore di rendimenti per l'azione B è semplicemente una versione scalata del vettore di rendimenti per l'azione A.

Azione A: $w_t = (w_{1,t}, w_{2,t}, \ldots, w_{n,t})$
Azione B: $\gamma w_t = (\gamma w_{1,t}, \gamma w_{2,t}, \ldots, \gamma w_{n,t})$

Si definisce quindi il vettore $b = (b_1, b_2, \ldots, b_n)$ che rappresenta i carichi fattoriali delle azioni rispetto a n fattori di rischio comuni indicando quanto le azioni siano esposte rispetto a ciascun fattore di rischio comune e $r_{A,t}^{id}$ e $r_{B,t}^{id}$, i rispettivi ritorni dovuti a fattori idiosincratici per le due azioni che sono stazionari.

Questi ritorni possono essere espressi come:

$$r_{A,t} = (b_1 w_{1,t}, b_2 w_{2,t}, \ldots, b_n w_{n,t}) + r_{A,t}^{id} \tag{3.10}$$

$$r_{B,t} = \gamma(b_1 w_{1,t}, b_2 w_{2,t}, \ldots, b_n w_{n,t}) + r^{id}_{B,t} \tag{3.11}$$

Assumendo quindi che le serie siano cointegrate ossia che la relazione tra i rendimenti relativi ai fattori comuni siano: $r^{fc}_{B,t} = \gamma r^{fc}_{A,t}$ dove:

$$r^{fc}_{A,t} = (b_1 w_{1,t}, b_2 w_{2,t}, \ldots, b_n w_{n,t}) \tag{3.12}$$

$$r^{fc}_{B,t} = \gamma(b_1 w_{1,t}, b_2 w_{2,t}, \ldots, b_n w_{n,t}) \tag{3.13}$$

Si arriva pertanto alla definizione della seguente identità:

$$r_{B,t} - \gamma r_{A,t} = r^{id}_{B,t} - \gamma r^{id}_{A,t} \tag{3.14}$$

Questa equazione, nella sua semplicità, rappresenta a pieno il senso dell'intera analisi che c'è dietro, ovvero che il ritorno del portafoglio composto dai due asset dipende solo dalla componente di rischio *idiosincratica*.

Una delle misure proposte come indice sintetico e che permette un'analisi sommaria del grado di cointegrazione tra le variabili è il valore della *correlazione* in valore assoluto:

$$|\rho| = \left| \frac{cov(r^{fc}_{A}, r^{fc}_{B})}{\sqrt{var(r^{fc}_{A}) var(r^{fc}_{B})}} \right| \tag{3.15}$$

Chiaramente un valore più vicino all'unità comporta un grado maggiore di correlazione, e quindi una maggiore probabilità di cointegrazione.

Tuttavia, come evidenziato da Do et al. (2006) [7], il ragionamento è fallace rispetto ad un'importante teoria precedentemente menzionata, l'APT. Infatti, secondo tale teoria, il rendimento privo di rischio si aggiunge a quello dei fattori comuni:

$$r_{A,t} = r_{f,t} + (b_1 w_{1,t}, b_2 w_{2,t}, \ldots, b_n w_{n,t}) + r^{id}_{A,t} \tag{3.16}$$

$$r_{B,t} = r_{f,t} + \gamma(b_1 w_{1,t}, b_2 w_{2,t}, \ldots, b_n w_{n,t}) + r^{id}_{B,t} \tag{3.17}$$

Questo suggerisce che rispetto alla formulazione presentata, quando i profili di esposizione al rischio degli asset A e B sono identici fino a un fattore scalare (cioè quando $\gamma = 1$), ciò non implica necessariamente che il rendimento di una unità del titolo A sia identico al rendimento di γ unità del titolo B più qualche rumore gaussiano.

A questo problema è possibile ovviare, considerando i ritorni in eccesso rispetto al tasso *risk free* definiti come $(r_{a,t} - r_{f,t})$ e $(r_{B,t} - r_{f,t})$.

3.1.2 Ulteriori criteri di screening

Una volta individuate le potenziali coppie di asset, è possibile applicare delle tecniche di screening aggiuntive che permettono una disamina più approfondita, ancor prima di effettuare un'attenta analisi dal punto di vista statistico-econometrico. È il caso dell'analisi grafica, una componente essenziale, che consente ai trader di identificare pattern e tendenze nei grafici dei prezzi, che possano indicare opportunità di trading.

Le medie mobili

Le medie mobili sono uno strumento fondamentale nell'analisi tecnica così come nel Pair Trading. Ampiamente utilizzate per la facilità di costruzione, forniscono dei segnali attraverso una rappresentazione *smoothed* dei dati relativi ai prezzi, che aiuta ad identificare la direzione generale di un trend, senza considerare gli effetti del mercato. Lo scopo principale di una media mobile, in accordo con J.J. Murphy (1999) [24], è quello di identificare o segnalare che un nuovo trend sta iniziando (o è iniziato) o sta terminando (o è terminato), oppure ancora che il trend sta per essere oggetto di un'inversione. Si rivolge al passato, e come sostenuto dall'autore è un *follower*, non un *leader*, pertanto non è idonea a prevedere dei trend futuri sul lungo termine, quanto piuttosto utile invece ad analizzare un aspetto della reazione ed in generale dei pattern, rispetto a qualcosa che si è già manifestato. Una delle principali tecniche utilizzate con le medie mobili, è quella detta *double crossover method*. Nello specifico, vengono prese due medie mobili differenti, una breve e una a lungo periodo, ad esempio una media mobile a cinque giorni e una a venti, oppure una a dieci ed una a cinquanta. Il segnale di acquisto del titolo, si manifesta nel momento in cui la media mobile a meno giorni supera quella più lunga. Viceversa, un segnale di vendita viene riscontrato quando la media mobile più corta scende al di sotto di quella più lunga.

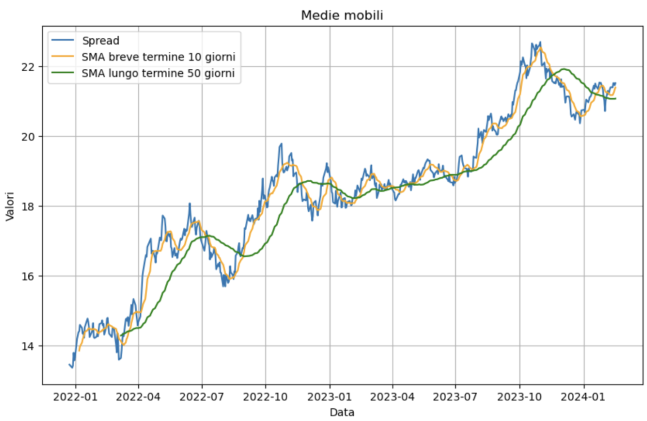

Figura 3.1: Esempio di applicazione delle medie mobili su Spread calcolato tra iShares 7-10 Year Treasury Bond ETF e iShares 3-7 Year Treasury Bond ETF

Nei capitoli successivi, in cui si parlerà dell'implementazione della strategia, sarà possibile mettere in pratica l'utilizzo delle medie mobili con la finalità d'identificare i segnali di acquisto e vendita in maniera dinamica.

Bande di Bollinger

La tecnica delle bande di Bollinger, adottata da John Bollinger, consiste nel tracciare le due bande sopra e sotto ad una media mobile, solitamente a venti giorni, a distanza di due deviazioni standard dal pattern del prezzo. Essendo quest'ultima una misura di dispersione intorno ad un valore medio, ci si assicura che i dati dei prezzi cadano all'interno delle bande con un 95% di probabilità. Come regola, si considerano 'ipercomprati' i titoli che toccano la banda superiore, mentre s'intendono 'ipervenduti' i titoli che toccano la banda inferiore. I trader possono osservare e modificare la larghezza, in modo tale da valutare la volatilità del mercato. È infatti normale, che queste bande si allarghino nel momento in cui si osserva un'alta volatilità, viceversa per il contrario. Al pari delle medie mobili, possono essere utilizzate come un ottimo indicatore, volto a segnalare i potenziali punti di ingresso e di uscita delle rispettive operazioni.

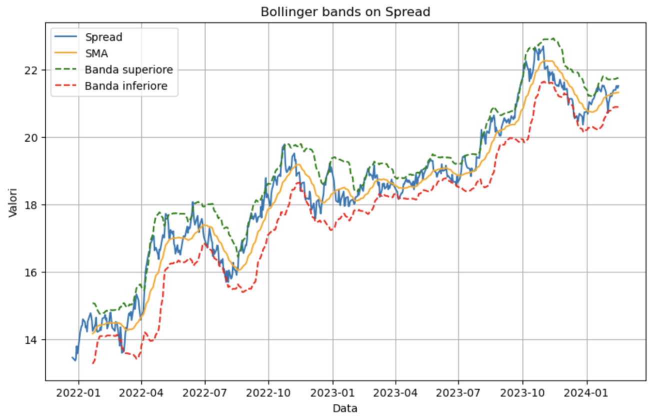

Figura 3.2: Esempio di applicazione delle Bande di Bollinger su Spread calcolato tra iShares 7-10 Year Treasury Bond ETF e iShares 3-7 Year Treasury Bond ETF

3.2 Stazionarietà e Augmented Dickey-Fuller Test

Come sotttolineato più volte, le serie storiche dei prezzi non possiedono la caratteristica della stazionarietà. Pertanto, essendo necessaria ai fini della nostra analisi la caratteristica manzionata, è possibile considerare una misura alternativa a quella dei prezzi che è data dal *price ratio*. Affinché sia possibile generare profitti nel Pair Trading, è necessario che il *price ratio* abbia media e volatilità costanti nel corso del tempo. È proprio questa caratteristica, che permette di considerare la temporaneità degli scostamenti dalle stesse, e quindi di assumere in un certo senso la possibilità di un ritorno ad una relazione di stabilità nel lungo periodo, che si avvicini ai valori 'normali' di media e volatilità.

3.2.1 ADF test

Un metodo popolare in ambito econometrico nonché abbastanza semplice ed immediato è quello proposto da Dickey e Fuller (1979) [5].

L'ipotesi nulla dell'ADF test senza trend, verifica l'ipotesi nulla dell'esistenza di una radice unitaria nella seguente regressione:

$$y_t = c + \phi y_{t-1} + \beta_1 \Delta y_{t-1} + \ldots + \beta_p \Delta y_{t-p} + \varepsilon_t \tag{3.18}$$

dove Δ è l'operatore differenza tale per cui $\Delta y_t = y_t - y_{t-1}$, mentre il numero di lags è determinato empiricamente in modo tale che l'errore ε_t sia serialmente non correlato ed abbia media pari a zero. Si parla di radice unitaria in quanto si verifica l'ipotesi nulla dell'esistenza di una radice unitaria (appunto detto *unit root test*) rispetto al primo termine y_{t-1} dunque $\phi = 1$, rispetto all'alternativa che indica $\phi < 1$. L'esistenza di una radice unitaria, indicherebbe infatti una condizione che rende impossibile l'adozione di questa strategia poiché non vi sarebbe stazionarietà.

A questo punto, attraverso i vari test econometrici, è possibile eseguire il test ottenendo il *p-value* ed il relativo livello di significatività (solitamente lo 0,05).

Questo test, presenta vantaggi e svantaggi relativi alla sua implementazione. Partendo dai primi, sicuramente la semplicità nell'utilizzo, trattandosi di una verifica di ipotesi, è indubbiamente un punto a favore, così come la relativa interpretazione e la conseguente ampia diffusione. Tuttavia, analizzando l'altra faccia della medaglia, si riscontrano alcuni limiti celati nelle assunzioni statistiche, che potrebbero non essere soddisfatte nella pratica, come la lunghezza delle serie storiche disponibili, non sempre sufficientemente ampie e tali da permettere la corretta esecuzione del test basato sul passato, e quindi tali da non consentire di prevedere gli andamenti futuri dello *spread*.

3.3 La verifica per la cointegrazione

3.3.1 Test di Engle e Granger

Nel corso della verifica di due serie multivariate, al fine di determinare l'esistenza di una relazione tra le stesse, Engle e Granger osservarono, che anche prendendo in esame due serie non stazionarie, talvolta è possibile che sia stazionaria la combinazione lineare delle due.

La spiegazione della cointegrazione è fornita dalla correzione rispetto all'errore. Nello specifico, i sistemi cointegrati hanno un equilibrio di lungo periodo corrispondente all'equilibrio della combinazione delle due serie temporali. Se c'è una deviazione dalla media di lungo periodo, allora o una o entrambe le serie temporali si muoveranno per tornare all'equilibrio.

Engle e Granger: two steps Test

Il primo passo per l'esecuzione del test a due step, è la costruzione di una regressione lineare stimata con l'OLS. Tuttavia, come in precedenza osservato, è importante che le serie considerate siano integrate di ordine 1 $I(1)$ e che quindi abbiano superato l'ADF test. Tale regressione restituisce due coefficienti:

- α che rappresenta l'intercetta della regressione;
- β che rappresenta il numero di unità da comprare o da vendere del titolo su cui si effettua la regressione rispetto ad una unità del titolo che viene scelto come variabile dipendente

Il parametro β può essere ottenuto mediante la seguente relazione:

$$\hat{\beta} = \frac{\sum_{t=1}^{T}(x_t - \bar{x_t})(y_t - \bar{y_t})}{\sum_{t=1}^{T}(x_t - \bar{x_t})^2} \tag{3.19}$$

ottenendo quindi $al\hat{p}ha = \bar{y_t} - \hat{\beta}\bar{x_t}$ con $\bar{x_t}$ e $\bar{y_t}$ sono le medie di x_t e y_t rispettivamente.

Il secondo step è costituito dall'esecuzione di un ADF test dei residui ottenuti dalla stima dell'OLS. Tali residui si ottengono dalla stima precedente come:

$$\hat{\varepsilon}_t = y_t - \hat{y}_t \tag{3.20}$$

Se tali residui sono stazionari, allora è verificato che la coppia di titoli sono cointegrati. La verifica d'ipotesi viene effettuata come segue:

$$H_0 : \hat{\varepsilon}_t \sim I(1) \quad non\ stazionaria$$

$$H_1 : \hat{\varepsilon}_t \sim I(0) \quad stazionaria$$

Sotto l'ipotesi nulla, essendo l'errore una serie integrata di ordine 1 e quindi di per sé non stazionaria, non è possibile esprimere y_t come combinazione lineare di x_t.

Pertanto, al fine di effettuare la verifica di ipotesi occorre considerare il modello:

$$\Delta\hat{\varepsilon}_t = \psi\hat{\varepsilon}_{t-1} + \sum_{j=1}^{k} \gamma_j \Delta\hat{\varepsilon}_{t-j} + u_t \tag{3.21}$$

dove $\Delta\hat{\varepsilon}_t$ è la differenza tra l'errore al tempo i e quello al tempo i-1, $\hat{\varepsilon}_{t-1}$ sono i residui stimati al tempo t-1 e u_t è un termine d'errore. Quindi il test d'ipotesi verifica quanto segue:

$$H_0: \ \psi = 0 \quad (non\ stazionaria)$$

$$H_1: \ \psi < 0 \quad (stazionaria)$$

Tuttavia, è bene considerare, come affermato da Kirchgassner (2012) [19], che i valori critici della regressione in oggetto utilizzati per performare l'ADF test sono differenti. Per quanto riguarda gli errori si tratta infatti di valori generati e non di valori osservati, i quali dipenderebbero dal numero di k variabili considerate nella serie temporale, ed anche dalla possibile considerazione del time trend. In conclusione di quanto appena detto, i valori critici si modificano a seconda delle casistiche e, sarebbero calcolabili attraverso l'utilizzo della simulazione di Monte Carlo, la quale come si vedrà per altre situazioni, può essere utilizzata al fine di considerare un numero molto grande di casistiche.

Infine, se i vari test precedentemente effettuati hanno prodotto un riscontro positivo, dando luogo a $\hat{\varepsilon}_t \sim I(0)$ è possibile costruire un modello che va sotto il nome di *Error Correction Model o ECM*, che aggiunge una componente di correzione dell'errore, che tiene conto della deviazione dalla relazione di equilibrio di lungo termine tra le variabili:

$$\Delta Y_t = \alpha + \beta_1 \Delta X_t + \gamma(\bar{Y}_t - \bar{\beta X_t}) + \varepsilon_t \tag{3.22}$$

dove:

- ΔY_t *e* ΔX_t rappresentano le variazioni delle serie temporali;
- $\bar{Y}_t$ e $\bar{X}_t$ rappresentano i valori medi delle serie storiche;
- $\bar{\beta}$ rappresenta la stima del coefficiente di regressione stimato nel primo step
- γ rappresenta il coefficiente di correzione dell'errore ossia quanto velocemente le variabili si riequilibrano quando si discostano dalla relazione di equilibrio di lungo termine.

3.3.2 Test di Johansen

La verifica inerente la cointegrazione, segue l'approccio proposto da Engle e Granger (1987) [9]; essendo una metodologia piuttosto semplice, soffre di alcune limitazioni, tra cui l'impossibilità di ottenere più di una relazione di cointegrazione per volta, e di conseguenza la necessità di analizzare una coppia di asset per volta. In aggiunta, l'aspetto negativo principale, è costituito dall'elevato grado di sensibilità nella scelta della security considerata come variabile dipendente.

Pertanto, un metodo diverso è senza dubbio quello proposto da Johansen (1988) [16].

La procedura consiste nel prendere in considerazione un modello di tipo VaR di ordine q. L'ordine del modello è dato da un criterio di informazione, nello specifico il BIC o criterio di informazione Bayesiano. Si ottiene pertanto:

$$Y_t = \mu + A_1 Y_{t-1} + \ldots + A_q Y_{t-q} + \varepsilon_t \tag{3.23}$$

Può essere scritta differenziando la serie e ottenendo:

$$\Delta Y_t = \mu + A Y_{t-1} + \Gamma_1 \Delta Y_{t-1} \ldots + \Gamma_p \Delta Y_{t-p} + \varepsilon_t \tag{3.24}$$

con $\Delta Y_t = Y_t - Y_{t-1}$ rappresenta il differenziale, dove A è la matrice dei coefficienti per il primo lag e Γ_i sono le matrici per ogni lag differenziato.

Anche in questo caso il test si esplica in una verifica d'ipotesi, che accerta l'assenza di cointegrazione che avviene quando la matrice $A = 0$.

Per farlo, si considera il rango della matrice A che è dato da r. Tale test di Johansen verifica se il rango della matrice è uguale a 0 fino ad ottenere il rango pari a $r = n - 1$ con n numero di serie presenti, oggetto di verifica di esistenza della relazione di cointegrazione.

A questo punto è possibile interpretare ciascun risultato come segue:

- un rango della matrice A tale per cui $r = 0$ identifica l'assenza di cointegrazione per tutte le serie considerate;
- un rango della matrice A tale per cui $r > 0$ implica l'esistenza di cointegrazione tra almeno due o più serie temporali tra quelle oggetto di analisi.

Efficacia delle tecniche di cointegrazione

Abbiamo quindi riscontrato, come la cointegrazione possa essere un buon metodo di analisi per la scelta e la selezione delle coppie di titoli utilizzabili nel Pair Trading. Il lavoro di Huck e Afawubo (2015) a riguardo, di particolare rilevanza, [13], considera 3 diverse tipologie di approccio, mettendo a confronto i risultati ottenuti, e avendo modo di verificare quale dei tre metodi risulta superiore in termini di ritorni. Nella ricerca, i due autori si concentrano sulle azioni dell'S&P 500, che per via della loro liquidità sono ideali per una strategia di questo tipo, e per quel che riguarda l'apertura e la chiusura delle posizioni viene utilizzata la regola generale proposta da Gatev et al. (2006) [10], delle soglie impostate ad un livello pari a 2 standard deviation e un'altra soglia pari a 3 standard deviation.

3.4 Indicatori di performance

Al fine di eseguire una selezione di asset che sia la migliore possibile, è bene considerare anche indicatori di natura finanziaria, ossia di performance, attraverso i quali sia possibile determinare l'efficacia, e la redditività della strategia messa in atto.

3.4.1 Lo sharpe ratio

Lo *Sharpe Ratio (o SR)*, sviluppato da William F. Sharpe, è una misura utilizzata per valutare il rendimento di un'attività finanziaria rispetto al rischio associato. È calcolabile come:

$$\text{Sharpe Ratio} = \frac{(R_p - R_f)}{\sigma_p} \tag{3.25}$$

dove:

- R_p è il rendimento del portafoglio che nel caso in esame sarà composto dai due titoli presi in considerazione per la strategia di Pair Trading;
- R_f è il tasso di rendimento di un titolo privo di rischio. Generalmente viene utilizzato come proxy il rendimento di un titolo di stato a breve termine (ad esempio un US Treasury bond);
- σ_p è la deviazione standard dei rendimenti dei titoli in portafoglio.

Il massimo profitto viene realizzato attraverso la massimizzazione dello Sharpe ratio, permettendo di confrontare diversi investimenti o portafogli in termini di rendimento aggiustato per il rischio. Anche se una strategia ha uno Sharpe Ratio elevato, potrebbe comportare un livello di rischio superiore rispetto a quello che gli investitori siano disposti a sopportare e quindi non essere ottimale.

Un'altra caratteristica da valutare, è la stabilità nel tempo. Infatti, una variabilità elevata dell'indicatore suggerirebbe una strategia poco robusta e resistente a variazioni di mercato e condizioni di trading.

3.4.2 Sortino Ratio

Il Sortino Ratio misura il rendimento aggiustato per il rischio, considerando solo la deviazione standard dei rendimenti negativi.

$$\text{Sortino Ratio} = \frac{R_p - R_f}{\sigma_N} \tag{3.26}$$

dove:

- R_p è il rendimento del portafoglio;
- R_f è il tasso di rendimento privo di rischio;
- σ_N è la deviazione standard dei rendimenti negativi.

Il Sortino Ratio viene utilizzato per valutare la performance di un portafoglio, tenendo conto solo dei rischi negativi. Un valore più alto del Sortino Ratio, indica che il portafoglio ha generato un rendimento maggiore per unità di rischio negativo assunto. Ecco alcune interpretazioni generali:

- Sortino Ratio > 1: indica che il portafoglio ha avuto buone performance aggiustate per il rischio negativo. I rendimenti generati superano significativamente i rischi negativi.
- Sortino Ratio = 1: indica che il portafoglio ha generato rendimenti esattamente proporzionati ai rischi negativi assunti.
- Sortino Ratio < 1: indica che il portafoglio non ha generato rendimenti sufficienti rispetto ai rischi negativi assunti. Più basso è il Sortino Ratio, peggiore è la performance aggiustata per il rischio negativo.

Tra i benefici dell'indice c'è la capacità di concentrarsi sulla volatilità negativa, offrendo una visione più approfondita del rischio quando gli investitori desiderano monitorare esclusivamente le perdite, evitando penalizzare i portafogli con una alta volatilità se questa viene considerata positiva dagli stessi investitori. Tuttavia, può essere influenzato dalla scelta del tasso di rendimento risk free, o dalla selezione della soglia di rendimento target, e richiedere una quantità troppo grande di dati per calcolare una deviazione standard significativa dei rendimenti negativi.

3.5 Strutturazione della strategia

Dopo aver selezionato la coppia di asset che soddisfa i criteri sopra menzionati sia dal punto di vista statistico che economico, non rimane che elaborare un insieme di regole ottimali che permettano la massimizzazione del profitto, e al tempo stesso la minimizzazione del rischio nell'apertura delle trades. Questo si traduce nell'impostazione di thresholds, che determinino l'apertura e la chiusura di posizioni lunghe e corte nel portafoglio composto dalle securities. L'idea alla base del metodo del Pair Trading è quella di considerare lo spread, che nel momento in cui raggiunge una determinata soglia, segnala la necessità di aprire una posizione lunga su un asset e corta sull'altro. In altre parole, è possibile richiamare il concetto di spread ed indicarlo come segue:

$$log(p_t^A) - \gamma log(p_t^B) = \mu + \varepsilon_t \tag{3.27}$$

Quando lo spread diverge dal suo valore di equilibrio, perché ad esempio è al di sotto dello stesso, la strategia è quella di comprare un'azione del titolo A e di vendere γ azioni del titolo B. Allo stesso modo quando lo spread è al di sopra del suo valore di equilibrio, l'approccio è inverso: si vende un'azione del titolo A e si acquistano γ azioni del titolo B. Il profitto si ottiene non appena avviene il ritorno al valore di equilibrio di lungo periodo.

3.5.1 Impostazione delle threshold

Diversi sono gli approcci e le tecniche che possono essere utilizzate nella scelta di specifici livelli soglia, per i quali sia prevista l'apertura o la chiusura di una posizione. Un metodo per costruire queste soglie, è quello utilizzato da Vidyamurthy (2004) [27] che modella lo spread come una serie *white noise*.

Lo spread come white noise

Questo approccio consiste nell'osservare la serie del rumore bianco gaussiano, che altro non è se non una serie rappresentativa delle deviazioni rispetto alla distribuzione gaussiana dello spread stesso, come differenza dei prezzi logaritmici delle securities. Come anticipato, viene comprata un'unità di spread, quando questa ha un valore uguale o minore a $-\Delta$, e similmente si vende un'unità di spread quando si osserva un valore maggiore o uguale a Δ.

La probabilità che il processo white noise devii di un ammontare pari o superiore a Δ ,è determinata dall'integrale calcolabile come $1 - N(\Delta)$. Prendendo come riferimento un tempo T, ci si può aspettare di avere $T(1 - N(\Delta))$ casi in cui lo spread sia più grande di Δ. Allo stesso modo, seguendo la proprietà di simmetria degli integrali, è possibile scrivere la probabilità di un valore uguale o minore di $-\Delta$ come $N(-\Delta) = 1 - N(\Delta)$ con un numero di casi in cui lo spread sia minore o uguale di Δ ancora una volta pari a $T(1-N(\Delta))$. Pertanto, ci si aspetta in media un numero di volte in cui si supera la soglia, e la conseguente apertura delle trades pari a $T(1-N(\Delta))$. Il profitto per ogni volta che si compra e si vende sarà pari a 2Δ e quindi la misura del profitto nel *trading period* sarà pari a $2T\Delta(1-N(\Delta))$.

Sebbene la modellizzazione sia teoricamente abbastanza semplice, non è allo stesso modo semplice la messa in pratica dei concetti appena espressi. Infatti, come evidenziato da Vidyamurthy [27], nell'esame di tecniche sempre più complesse, modellizzare lo spread in termini parametrici non è poi così scontato. In primis, l'assunzione stessa di normalità dell'errore è probabilmente un azzardo. Basterebbe pensare al trading di titoli, non sempre costante, soprattutto per volumi, poiché ad esempio i momenti di apertura e di chiusura dei mercati in una singola giornata risultano i momenti più volatili. Lo stesso discorso, in termini più ampi, può essere fatto rispetto ai periodi precedenti le festività o estivi, dove i volumi sono in genere molto più bassi rispetto al resto dell'anno. In ogni caso, sembrerebbe più realistico modellare gli spread di rumore bianco come valori tratti da distribuzioni normali, ma con deviazioni standard che dipendono dal tempo. La distribuzione complessiva dei valori dello spread in questo caso, può essere definita come una distribuzione mista gaussiana. La soluzione potrebbe essere quella di stimare in maniera dinamica la volatilità, e far variare le thresholds sulla base della volatilità registrata.

Lo z-score

In virtù della complessità legata all'impostazione di una soglia specifica a causa della variabilità del mercato e delle molteplici situazioni imprevedibili che possono presentarsi, uno degli approcci più efficienti in termini di semplicità e di risultato, è quello proposto da Gatev et al. (2006) [10].

Partendo dalla definizione di Price Ratio, per cui:

$$Ratio_{t;A,B} = \frac{P_{A,t}}{P_{B,t}}$$

Dopo aver verificato che le serie sono cointegrate, e che il rapporto tra i prezzi è una serie stazionaria integrata di ordine 0, si può procedere definendo la misura di $z-score$ come:

$$z_t = \frac{\text{Price ratio}_t - \mu}{\sigma} \tag{3.28}$$

dove μ è la media del *Price ratio* e σ la sua deviazione standard.

In base a questa misura così calcolata, è possibile fissare le soglie di riferimento. Diverse sono le correnti di pensiero in materia. Tuttavia è bene avere chiaro che, fissare una soglia alta in valore assoluto comporta un numero inferiore di trades perché semplicemente è più difficile da raggiungere. Per converso, se si fissasse una soglia bassa in valore assoluto, questo comporterebbe necessariamente un aumento delle trades, incrementando potenzialmente i profitti ma anche i relativi rischi.

Tra le molteplici possibilità, prendiamo a riferimento la modalità attraverso cui fissare le soglie. Ipotizziamo che si realizzi la possibilità di adottare la strategia di z score empirici differenti per segnali di acquisto e vendita come proposto da Avellanada e Lee (2010) [1], in questo lavoro si prende a riferimento il metodo proposto da Caldeira e Moura (2013) [2]: l'acquisto del portafoglio avviene nel momento in cui lo z score è minore o uguale di -2, il portafoglio è sottovalutato e quindi occorre andare long dell'asset A e short per γ unità dell'asset B; analogamente, la vendita del portafogio avviene quando lo z score è maggiore o uguale a 2 e quindi occorre andare short nell'asset e long per γ unità dell'asset B.

Distanza di Mahalanobis

La distanza di Mahalanobis è una misura di 'dissimilarità' tra due punti dati in uno spazio multidimensionale, considerando la struttura di covarianza dei dati. Nello specifico, una distanza di Mahalanobis bassa indica che i due punti dati sono simili rispetto alla struttura di covarianza dei dati, mentre una distanza alta indica che i punti sono dissimili.

La scelta di una soglia dipende dall'obiettivo dell'analisi. In alcune applicazioni, potrebbe essere considerata una distanza di Mahalanobis al di sotto di un certo valore come "bassa" (ad esempio, inferiore a 2), altre volte, si potrebbero utilizzare percentili o valori specifici basati sulla distribuzione delle distanze nel dataset.

Nell'ambito del Pair Trading, una distanza di Mahalanobis bassa potrebbe essere considerata come un segnale per selezionare coppie di asset simili. Ad esempio, se la distanza tra i rendimenti di due azioni è inferiore a una soglia specifica, ciò porterebbe probabilmente a definire una buona coppia per il pair trading.

Supponendo di avere due asset finanziari A e B, e di aver raccolto i loro rendimenti storici, calcolando la matrice di covarianza tra i rendimenti di A e B. Questa matrice tiene conto delle correlazioni tra i rendimenti dei due titoli.

Ora è possibile calcolare la distanza di Mahalanobis tra i rendimenti di A e B utilizzando la seguente formula:

$$D_{A,B} = \sqrt{R_A - R_B)^T \Sigma^{-1} (R_A - R_B)} \tag{3.29}$$

Dove:

- $D_{A,B}$ è la distanza di Mahalanobis tra gli asset A e B
- R_A e R_B sono i vettori dei rendimenti storici di A e B
- Σ^{-1} è l'inverso della matrice di covarianza tra i rendimenti di A e B.

Dalle molteplici applicazioni a cui tale indice si presta, l'individuazione della correlazione, e conseguentemente la scelta di asset molto simili tra loro, può essere una buona approssimazione di questo indicatore sintetico che permette di individuare le migliori coppie di asset possibili.

3.5.2 Considerazione dei rischi

Trattandosi di una strategia che comporta l'acquisto e la vendita di titoli nel mercato, ciò rende inevitabile l'esistenza di una serie di rischi che devono essere fronteggiati. Infatti, sebbene si parli di una strategia neutrale rispetto al mercato, non si tratterebbe di un arbitraggio puro. Gli eventi di mercato comportano delle inefficienze di prezzo o dei cambiamenti strutturali, che possono invalidare i modelli creati. Può succedere ad esempio che il *Price Ratio* non converga immediatamente, ma che prima di avviare il processo di tipo *mean reverting* si muova in direzione opposta per molto tempo. È dunque possibile che il valore di equilibrio dove avviene la convergenza, possa modificarsi nel tempo. Inoltre, tale metodo è indubbiamente esposto a delle limitazioni di natura tecnica, in quanto l'utilizzo di assunzioni statistiche per individuare i profitti, possono rivelarsi errate nel tempo.

3.5.3 Indicatori di rischio

Quando siamo di fronte ad un titolo o ad un portafoglio di titoli, è bene prima di tutto considerare il rischio ad esso associato, che può essere definito in due modi differenti:

- rischio simmetrico, ossia quello per cui si indica la probabilità che il rendimento di un titolo devii dal suo rendimento atteso;
- rischio asimmetrico, ossia quello per cui i rendimenti del titolo si attestino per valori inferiori rispetto al rendimento atteso.

Soffermandoci sulla prima tipologia di rischio, è doveroso prendere in considerazione i vari strumenti che quantificano e misurano effettivamente le componenti dello stesso rischio.

Varianza (storica o campionaria)

$$\hat{\sigma}^2 = \frac{1}{n-1}\sum_{t=1}^{n}(r_t - \bar{r})^2 \tag{3.30}$$

dove:

- $\bar{r}$ è la media dei rendimenti del titolo o del portafoglio se si considera quest'ultimo

- r_t i rendimenti al tempo t
- n numero di periodi considerati

Una misura alternativa è costituita dalla radice della varianza, che indica la dispersione intorno al valore medio.

Tuttavia, questo indice soffre di alcune problematiche. Nello specifico, la varianza non è mai costante durante il periodo d' investimento. Infatti, è ormai comunemente condiviso che i prezzi dei titoli non abbiano una distribuzione normale, ancorché presentino delle similitudini. Possiedono infatti delle code più spesse, definite leptocurtica. La leptocurtosi è il fenomeno per cui la probabilità di accadimenti di fenomeni estremi, ovvero di outliers, è più alta rispetto a quella modellata dalla distribuzione di probabilità normale. Questo fenomeno è conseguente alla volatilità. È un fenomeno persistente, ossia il valore assunto al tempo t è strettamente correlato con quello assunto nel tempo precedente; si parla di *volatility clustering* per cui "grandi cambiamenti tendono ad essere seguiti da grandi cambiamenti, mentre i piccoli cambiamenti tendono ad essere seguiti da piccoli cambiamenti" Mandelbrot (1963) [22].

Per modellare questo fenomeno sono stati introdotti i cosiddetti modelli ad eteroschedasticità condizionata come ad esempio modello ARCH e l'evoluzione di questo ovvero il GARCH.

Proprio quest'ultimo definisce la varianza di un modello GARCH(p,q) come segue:

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{p} \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^{q} \beta_j \sigma - t - j^2 \tag{3.31}$$

dove i termini ε riguardano il rumore bianco della serie e i termini σ fanno riferimento alla varianza storica registrata nei periodi precedenti. In dottrina il modello più utilizzato è il GARCH(1, 1) cioè:

$$\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1} + \beta_1 \sigma_{t-1}^2 \tag{3.32}$$

Tale specificazione può anche essere rappresentata come un ARCH(∞):

$$\sigma_t^2 = \alpha_0(1-\beta_1) + \alpha_1 \sum_{i=1}^{\infty} \beta_1^{i-1} \varepsilon_{t-i}^2 \tag{3.33}$$

Tuttavia è bene osservare che i GARCH sono simmetrici; ciò significa che non è importante il segno dell'errore ε_t Questo, nelle serie storiche dei prezzi nei mercati finanziari, non è propriamente vero. Infatti, nei mercati si osserva un' ulteriore dinamica legata al cosiddetto effetto leva o *leverage effect* introdotto da Black nel 1972 che verifica empiricamente l'esistenza di una correlazione negativa tra volatilità e rendimenti; quando i prezzi delle azioni subiscono uno shock negativo, la società rispetto cui si riferiscono le azioni diviene più rischiosa, e gli strumenti finanziari ad essa connessi risulteranno a loro volta più rischiosi. Per poter inglobare il fenomeno nello studio, sono stati introdotti dei modelli generalizzati asimmetrici (AGARCH) che rispondono in maniera asimmetrica a variazioni in aumento della volatilità,distinguendo se tale aumento sia dovuto a notizie positive o negative.

Value at Risk (VaR)

Nell'ambito della misurazione del rischio non può non essere considerato il **Value at Risk (VaR)**, che fornisce una stima delle potenziali perdite di valore di un portafoglio, o di un investimento, in un determinato orizzonte temporale ed a un livello di confidenza specifico.

Questo indicatore è una misura di rischio *asimmetrica*, che consente di comprendere il potenziale rischio derivante dal probabile ribasso degli investimenti. In altre parole, quantifica l'entità delle possibili perdite finanziarie all'interno di un portafoglio, in un arco temporale prestabilito. Grazie al VaR è possibile stimare l'entità di una determinata perdita, rispetto ad una certa probabilità che la stessa si verifichi. Diversi sono i metodi con cui il VaR può essere calcolato. Si può esprimere calcolando il VaR assoluto e il VaR relativo. Il primo è espresso in valori monetari, e fornisce una stima diretta della perdita massima prevista; il secondo invece, è come suggerisce lo stesso termine, una misura relativa, e quindi esprime in percentuale il valore del portafoglio.

Non esiste un unico metodo per il calcolo del rischio; possono essere diversi gli approcci utilizzati, registrando delle differenze piuttosto rilevanti nei risultati.

- Metodo storico: è uno dei metodi più semplici che prevede l'analisi dei dati passati e misura la perdita massima sperimentata fino al momento della misurazione. Il limite di questo metodo, sta nel fatto che non è detto che il passato si ripeta, e che quindi la perdita sperimentata in passato sia una buona proxy per approssimare la perdita nel futuro.

- Varianza-covarianza (approccio parametrico), che assume i rendimenti del portafoglio distribuiti normalmente, ed utilizza la media e la varianza dei rendimenti per calcolare il VaR:
$$VaR = W_0 - W^*_{\alpha,t} \tag{3.34}$$
  dove W_0 è l'investimento iniziale e $W^*_{\alpha,t}$ è il valore dell'investimento nel caso di perdita massima ad uno specifico livello di confidenza α e in un orizzonte temporale t. Il limite di questo approccio è insito proprio nell'assunzione di normalità dei rendimenti dei quali sia stata già discussa la non veridicità, dovuta all'evidenza per cui le code della distribuzione sono in realtà più spesse.

- Simulazione di Montecarlo: è un metodo che utilizza simulazioni casuali di variazioni e andamenti dei prezzi per valutare il rischio. Tali scenari diversi, sono in realtà basati su distribuzioni di probabilità differenti, e si calcolano i rendimenti del portafoglio per ciascuno di essi ordinandoli in modo crescente. Selezionando poi un livello di confidenza desiderato, solitamente pari al 5% o all'1 % il VaR è calcolato come in precedenza (equazione 3.38) come differenza tra il valore dell'investimento iniziale e il valore del portafoglio al livello di confidenza scelto ma calcolato con la simulazione di Monte Carlo.

In tale contesto, si rendono necessarie delle regole che permettano di fare *risk management* portando ad una limitazione delle perdite e ad una coerente gestione del rischio.

Expected Shortfall (cVaR)

Il CVaR, noto anche come Expected Shortfall, è una misura di rischio che considera non solo la soglia di perdita definita dal VaR, ma anche l'entità delle perdite che eccedono tale soglia. Introdotto da Rockafellar e Uryasev (2000), il CVaR al livello di confidenza α è definito come il valore atteso delle perdite nei peggiori $\alpha\%$ degli esiti. Formalmente:

$$\max_{\theta} \mathbb{E}_{\theta}[R] \quad \text{s.t.} \quad \Phi(\theta) \geq b \tag{3.35}$$

dove $\mathbb{E}_{\theta}[R]$ denota il valore atteso di R rispetto a θ, e $\Phi(\theta) \geq b$ è il vincolo.

Il CVaR è particolarmente utile perché fornisce una visione più completa del rischio di coda rispetto al VaR, catturando le perdite estreme che possono avere un impatto significativo sui portafogli.

Il Downside risk

Il downside risk, o rischio al ribasso, è una misura della volatilità dei rendimenti negativi di un investimento. In sostanza, si concentra sulle perdite potenziali anziché sull'intera distribuzione dei rendimenti.

Nella finanza, il downside risk è considerato un indicatore importante della sicurezza di un investimento, poiché fornisce informazioni specifiche sulla probabilità e sull'entità delle perdite, e perché il suo calcolo consente agli investitori di valutare meglio il rischio potenziale associato a un determinato investimento, nonché di prendere decisioni più consapevoli sulla gestione del portafoglio.

La formula per calcolare il downside risk è la seguente:

$$\text{Downside Risk} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(R_i - MAR)^2} \tag{3.36}$$

dove:

- R_i rappresenta i rendimenti negativi selezionati;
- MAR è il tasso minimo accettabile di rendimento;

- n è il numero di rendimenti negativi selezionati.

Tale indicatore misura la deviazione standard dei rendimenti negativi, rispetto al tasso minimo accettabile di rendimento; più alto è il downside risk, maggiore è la volatilità dei rendimenti negativi, e maggiore il rischio al ribasso associato all'investimento.

Ottimizzazione del CVaR

L'ottimizzazione del CVaR è un problema complesso che può essere formulato come un programma stocastico. Quando il payoff ha la struttura $R = f_\theta(X)$, dove f_θ è una funzione deterministica e X è una variabile casuale che non dipende da θ, l'ottimizzazione del CVaR può essere risolta utilizzando vari approcci.

Nei sistemi di code, nell'allocazione delle risorse e nell'apprendimento per rinforzo, i parametri di ottimizzazione controllano anche la distribuzione degli esiti casuali. In questi casi, i metodi tradizionali di ottimizzazione del CVaR non sono sufficienti. Prashanth e Ghavamzadeh (2013) [21] hanno esplorato l'ottimizzazione sensibile al rischio in questi domini, evidenziando la necessità di sviluppare algoritmi di ottimizzazione del CVaR più generali.

Un ulteriore vantaggio dell'estimatore del gradiente del CVaR, è la sua capacità di incorporare il campionamento per importanza, cruciale quando α è piccolo, e il CVaR cattura eventi rari. Questo permette di migliorare l'accuratezza delle stime del rischio e di ottimizzare le strategie di gestione del portafoglio in modo più efficace.

L'uso del VaR e del CVaR rappresenta un passo fondamentale nella gestione del rischio finanziario. Mentre il VaR fornisce una misura semplice e intuitiva delle perdite potenziali, il CVaR offre una visione più completa, considerando anche le perdite estreme. L'ottimizzazione del CVaR, sebbene complessa, offre strumenti potenti per migliorare le decisioni di investimento e la gestione del rischio, rendendola una pratica essenziale per le istituzioni finanziarie moderne.

Analisi delle perdite dal massimo storico: il Drawdown

Il drawdown è una misura fondamentale nell'analisi finanziaria per valutare il rischio associato agli investimenti, poiché rappresenta la perdita massima registrata da un investimento rispetto al suo massimo storico.

Dal punto di vista dell'interpretazione, un drawdown negativo indica che l'investimento è attualmente al di sopra del suo massimo storico, il che può essere considerato positivo dagli investitori. D'altra parte, un drawdown positivo indica che l'investimento ha subito una perdita rispetto al suo massimo storico,e che potrebbe essere motivo di preoccupazione per gli investitori. Il contributo dell'indicatore non si limita ad essere meramente analitico, ma aiuta a prendere decisioni informate sulla gestione del portafoglio. Ad esempio, un investitore potrebbe utilizzare il drawdown per impostare livelli di stop loss, o per valutare la performance relativa di diversi fondi o strategie di investimento nel tempo.

È importante fare attenzione alla difficoltà di calcolo del drawdown di portafoglio, derivante principalmente dalla diversità dei time frame e degli strumenti sottostanti su cui ogni singola strategia è stata applicata. Se tutte le strategie operassero sullo stesso dataset, sarebbe possibile assumere che i massimi e minimi si verificano simultaneamente, come avviene nel backtest di una singola strategia. Non solo, quando le strategie operano su sottostanti diversi, anche a parità di time frame, diventa impossibile garantire questa simultaneità poiché ad esempio il minimo di un sottostante potrebbe verificarsi mentre un altro sottostante raggiunge il massimo, o in qualsiasi altra fase del range di prezzo.

Nonostante siano molteplici gli approcci che possono essere utilizzati per il Drawdown, in questa sede verrà calcolato su una frequenza di prezzi giornalieri, secondo la definizione fornita da Hull, John C. (2012)[14]:

$$\text{Drawdown}\ (\%) = \left(\frac{P_{\max} - P_{\min}}{P_{\max}}\right) \cdot 100 \tag{3.37}$$

Dove:

- *Pmax* è il picco massimo nell'intervallo di tempo considerato;
- *Pmin* è il picco minimo nell'intervallo di tempo considerato.

Stop loss

Nel momento in cui un portafoglio subisce delle grandi perdite, si può essere costretti a dover liquidare alcune posizioni anche se le stesse non fossero problematiche; ciò comporta inevitabilmente delle reazioni a catena che compromettono l'intera situazione.

Al fine di limitare il rischio di perdite catastrofiche, i trader possono utilizzare gli *stop loss*, ovvero degli ordini che vengono utilizzati in maniera automatica nel momento in cui il prezzo di un titolo raggiunge un livello specifico. Scegliere tale livello è però tutt'altro che semplice, e nel Pair Trading questo si traduce in una complessità maggiore considerando i due titoli correlati. Un errore comune, è pensare che questa strategia limiti in ogni caso le perdite. Nell'eventualità di shock sui mercati, i prezzi dei titoli si muovono in maniera discontinua e le posizioni vengono chiuse a un livello inferiore rispetto al prezzo di soglia prefissato dallo stop loss. Caldeira e Moura (2013) [2] utilizzano uno stop loss del 7 % chiudendo quindi la posizione quando questa sperimenta una perdita pari alla soglia prestabilita.

Holding period

Un'altra considerazione doverosa, riguarda il periodo di tempo durante il quale le posizioni vengono mantenute in portafoglio. L'equilibrio di lungo periodo infatti, potrebbe modificarsi col tempo, diventando del tutto diverso rispetto a quello su cui è stata costruita la strategia. Proprio Caldeira e Moura (2013) [2] fissano un *holding period* massimo, che è rappresentato da 50 giorni, in quanto dai campioni che gli stessi hanno analizzato è stato evidenziato un calo della profittabilità dopo i 50 giorni.

Per definizione tuttavia, è bene ricordare che il trading algoritmico comporta in maniera fisiologica un calo dell'hoding period, accorciando l'orizzonte temporale del mercato, con una concentrazione minore sui fondamentali delle società sottostanti, e maggiormente sui movimenti di prezzo a breve termine.

Capitolo 4

Analisi empirica

Dopo aver affrontato gli argomenti che sono alla base della teorizzazione della strategia in esame, concentriamo l'attenzione sull'analisi empirica, eseguita attraverso la considerazione di un determinato periodo di tempo entro il quale analizzare il comportamento degli asset. In particolare, il riferimento principale è rivolto ai dati delle serie storiche dei prezzi comprese tra il 31 Aprile 2019 e il 31 Aprile 2024. Oggetto di analisi è una categoria specifica di asset, le azioni, e più nello specifico quelle quotate nell'S&P 500, principale indice di borsa americano che ricomprende al suo interno le prime 500 società americane per capitalizzazione.

4.1 Selezione dei titoli

4.1.1 I dati

Come anticipato, per lo studio che si va delineando, sono stati scelti una serie di titoli, che nei capitoli successivi verranno presi a riferimento nell'effettuare tutte le analisi volte a comprenderne la fattibilità e la possibilità di realizzare la strategia con successo. Nel caso specifico, come anticipato, sono state analizzate le componenti più importanti dell'S&P 500 (circa 80 azioni), di conseguenza la valuta considerata è espressa in dollari americani (USD).

Trattasi di una classe specifica di strumenti finanziari, ovvero quella del *listed equity*. Considerato che lo screening viene effettuato tra le prime 100 in ordine di capitalizzazione, il riferimento non può che riguardare le cosiddette '*Blue Chips*', società stabilizzate e, di norma, affidabili poiché si sono affermate e consolidate nel corso del tempo. Proprio questa caratteristica le rende solitamente più stabili rispetto alle loro controparti, ossia le *Penny stocks* che sono una classe di azione tendenzialmente con prezzo più basso ma che offre rendimenti nel tempo molto più instabili con conseguente aumento della volatilità.

Hanno una serie di caratteristiche specifiche che, in linea teorica, le rendono perfette per la strategia di Pair Trading che si vuole mettere in atto. Infatti, tali asset sono contraddistinti da:

- Stabilità e affidabilità: le azioni dell'S&P 500 sono emesse da società consolidate e affermate nel tempo, con capitalizzazioni di mercato significative che può ridurre il rischio di movimenti improvvisi dei prezzi, il che è importante per tale metodo che si basa su modelli di relativa stabilità nel rapporto tra i prezzi dei due asset o per lo meno sulla tendenza di questi all'equilibrio;

- Liquidità: esse tendono ad avere un'elevata liquidità sul mercato, rendendo di conseguenza più facile acquistare e vendere grandi quantità di queste azioni senza influenzare significativamente il prezzo di mercato, ma soprattutto concedono la possibilità di ri-bilanciare nell'immediato il portafoglio in caso di eventuale necessità;

- Basso spread e costi di transazione: diretta conseguenza del punto precedente, dato il loro alto volume di trading, tali azioni tendono ad avere costi di transazione più bassi rispetto ad altre azioni meno liquide. Inoltre, essendo scambiati su volumi molto alti, solitamente lo spread è baso. Questo è importante perché riduce gli oneri associati all'esecuzione di strategie di trading.

Il panel di dati

Come anticipato, per l'analisi in oggetto, sono stati considerati 70 titoli quotati nell'indice americano, scelti tra i primi 100 in ordine di capitalizzazione [1]. L'intervallo periodale considerato è di 5 anni dal 31/04/2019 al 31/04/2024 e l'intervallo temporale scelto per ottenere la serie dei prezzi è giornaliero, per un totale di circa 88130 osservazioni. I dati sono scaricati secondo i prezzi di chiusura *"Closing price"* direttamente mediante l'utilizzo del tool Yahoo Finance[2] per Python.

Per ciò che concerne la possibile esistenza di dati non disponibili "*NaN*" o "*inf*", questi non vengono riportati ma sostituiti dal valore risalente all'ultima osservazione prima del valore distorto. Sebbene si tratti di un'assunzione non proprio trascurabile, questo permette di ottenere un Dataset completo, che a seguito dei vari calcoli non porti a risultati distorti e plausibili.

4.1.2 Primo screening dei dati

È verosimile ritenere che avere a che fare con un numero così ampio di dati, comporti una certa difficoltà nell'elaborarli simultaneamente. Pertanto, è necessario attivare dei

[1] Si riporta in appendice la lista completa dei titoli considerati

[2] https://it.finance.yahoo.com/

meccanismi iniziali che permettano di effettuare almeno una prima scrematura del Dataset, eliminando quelle securities che in assoluto non avrebbe senso utilizzare per la nostra analisi, perché ritenute inadeguate rispetto alle caratteristiche dei titoli utilizzati nel Pair Trading. Il riferimento sostanziale è incarnato da una caratteristica fondamentale qual'è la correlazione. I Titoli che in assoluto non sono correlabili gli uni con gli altri, non possono essere utilizzati nell'analisi, poiché verrebbe meno l'assunzione principale dietro ai modelli basati sulla *cointegrazione*.

A tale scopo, viene calcolata la correlazione per ciascuna delle combinazioni che portano alla creazione di ipotetiche coppie di asset. Per un totale di 70 asset, si può applicare il calcolo combinatorio attraverso la seguente formula:

$$C(n,k) = \frac{n!}{k! \cdot (n-k)!} \tag{4.1}$$

dove:

- $C(n,k)$ indica appunto il numero di coppie;
- $n = 70$ ovvero il numero totale di asset;
- $k = 2$ ovvero il numero di asset da selezionare per creare una coppia.

Di conseguenza il risultato è di 2415 coppie, un numero davvero troppo ampio se si considera che la selezione finale verterà sulla scelta di una singola coppia sul totale di quelle trovate.

Sebbene praticamente il calcolo della correlazione venga performato per la totalità del Dataset, un esempio del correlogramma è quello proposto in Figura 4.1, nella quale vengono considerati per semplicità visiva i soli titoli *tech*.

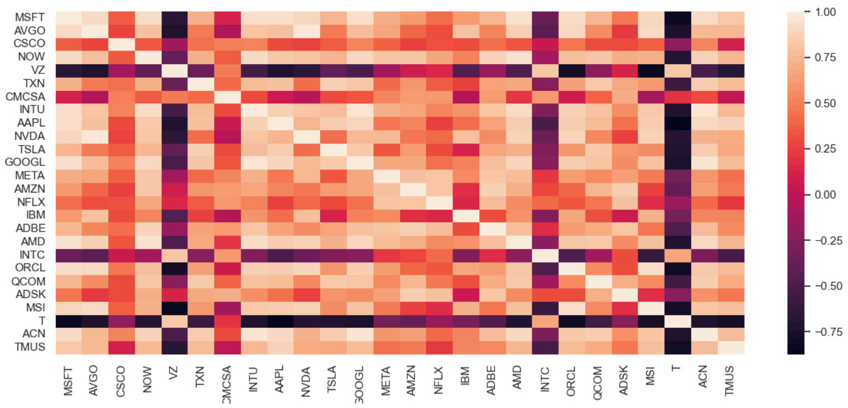

Figura 4.1: Correlogramma top società tech del S&P 500

Da una prima analisi del grafico, è possibile notare come la correlazione in valore assoluto tra i vari asset sia piuttosto alta. Questo risultato non è affatto sorprendente,in quanto sono state prese in considerazione tutte le società che fanno parte di un settore, quello tecnologico; è bene tenere presente che la maggior parte dei fattori che influenzano l'andamento di un titolo sono comuni a tutti gli altri, e che quindi i titoli possano rispondere in modo più o meno simile alle fluttuazioni di mercato.

Dopo quanto esposto, è possibile e necessario restringere il campo di osservazione e di analisi del numero dei titoli giungendo così alle conclusioni dello studio.

Si procede alla stima delle correlazioni tra tutte le coppie di asset. Le prime 9 coppie di asset in ordine di correlazione, costituiranno il panel effettivo dei dati rispetto al quale saranno concentrate le successive considerazioni.

Azione 1	Azione 2	Correlazione
LLY	NVO	0.986364
AVGO	NVDA	0.973661
MS	GS	0.968282
ELV	UNH	0.967951
NVO	AVGO	0.962762
LLY	AVGO	0.962494
INTU	ACN	0.958655
MSFT	CMG	0.957655
ACN	GOOGL	0.957410

Tabella 4.1: Top 9 azioni per correlazione

In linea generale, come è possibile notare nella tabella 4.1, almeno due sono gli aspetti principali che devono essere presi in considerazione:

- la correlazione per le prime 9 coppie di asset si attesta su valori piuttosto alti e prossimi alla correlazione perfetta in valore assoluto (Correlazione $\sim |1|$);
- la maggior parte dei titoli che fanno parte di ciascuna coppia sono titoli tecnologici.

Tale ultima evidenza relativa all'argomento della correlazione, è riscontrabile in figura 4.2, dove per valori molto alti di correlazione, i punti dello scatterplot si allineano quasi a formare una linea retta a 45°.

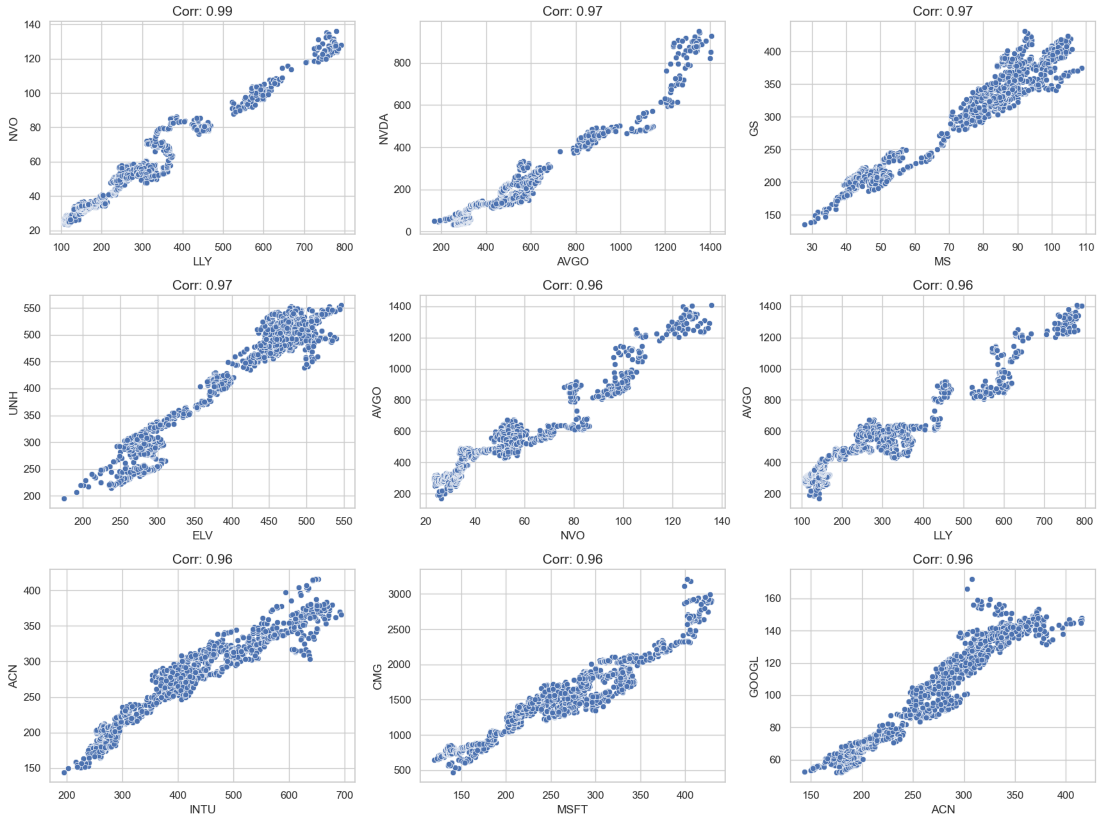

Figura 4.2: Correlazione per coppie (top 9 coppie di titoli da S&P 500)

Infine nella tabella 4.2 viene mostrata la serie degli ultimi 5 prezzi per ciascun titolo

Date	ACN	AVGO	CMG	ELV	GOOGL	GS	INTU	LLY	MS	MSFT	NVDA	NVO	UNH
2024-04-23	316.83	1249.19	2915.00	532.92	158.26	424.00	630.88	745.69	93.76	407.57	824.23	128.64	486.18
2024-04-24	313.54	1256.82	2926.76	533.73	159.13	423.04	635.49	732.20	93.85	409.06	796.77	126.16	487.30
2024-04-25	309.00	1294.42	3111.97	539.68	156.00	420.05	626.39	724.87	92.56	399.04	826.32	125.79	493.86
2024-04-26	308.01	1344.07	3186.97	537.26	171.95	427.57	636.55	733.51	92.83	406.32	877.35	126.85	495.35
2024-04-29	303.16	1338.62	3209.47	533.98	166.15	430.81	638.39	737.20	92.11	402.25	877.57	126.88	489.03

Tabella 4.2: Serie dei prezzi per asset presenti nelle prime 9 coppie per correlazione

Lo scaling dei dati

Quando si confrontano serie di prezzi di titoli diversi, è possibile e normale che vi siano delle differenze di scala tra gli stessi. Questo di fatto, si traduce nell'impossibilità di confrontare serie di prezzi che si riferiscono a scale diverse, ed anche ad una distorsione generale per quanto concerne la considerazione dei dati.

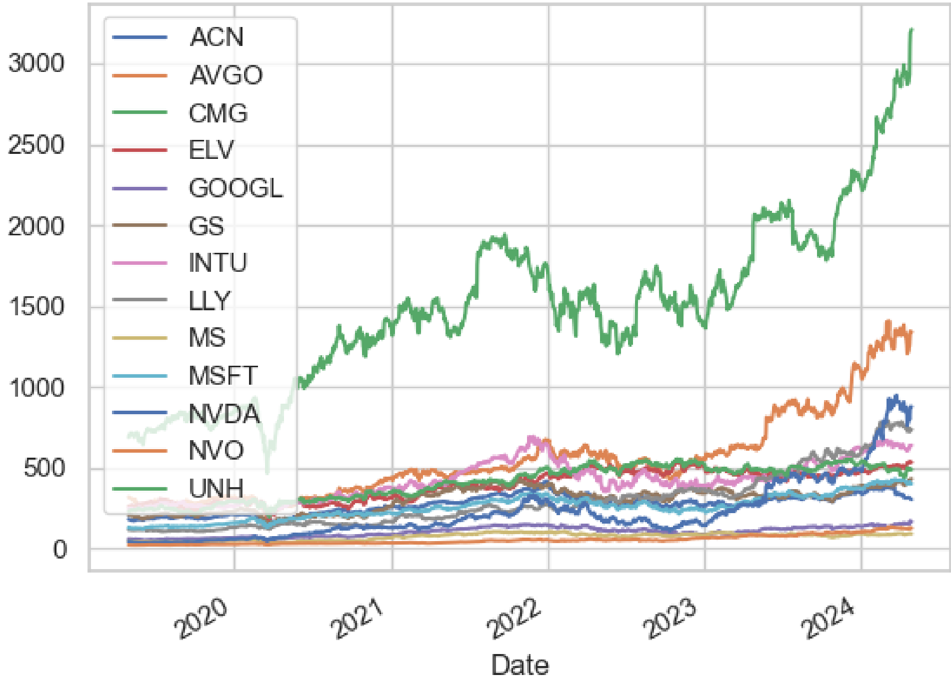

Figura 4.3: Serie storica dei prezzi non normalizzata

In specifico, questo fenomeno, oltre che dalla serie dei prezzi 4.3, è evidenziato da un particolare tipo di grafico, il boxplot 4.4, che evidenzia la distribuzione dei dati nell'intervallo temporale considerato.

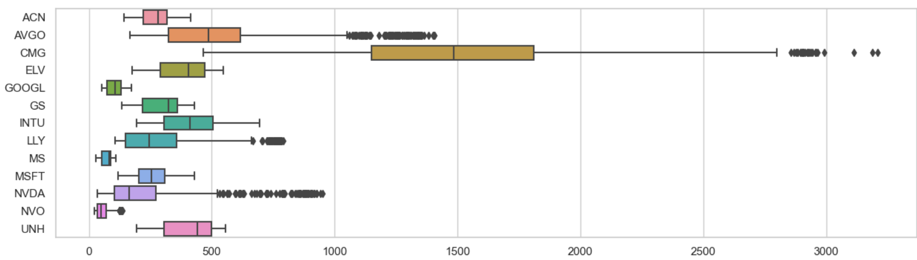

Figura 4.4: Box plot per serie di prezzi non scalate

Proprio per questa ragione, si rende necessaria un'operazione di scaling dei dati, che consiste nella normalizzazione degli stessi. Tale operazione viene effettuata con la seguente modalità:

$$P_n = \frac{P_t - \overline{P_t}}{\sigma_{P_t}} \tag{4.2}$$

Il risultato è un Dataset scalato, che come si può notare dal secondo boxplot (figura 4.5), presenta valori uniformemente distribuiti, nonostante non debba essere trascurata la presenza di outliers.

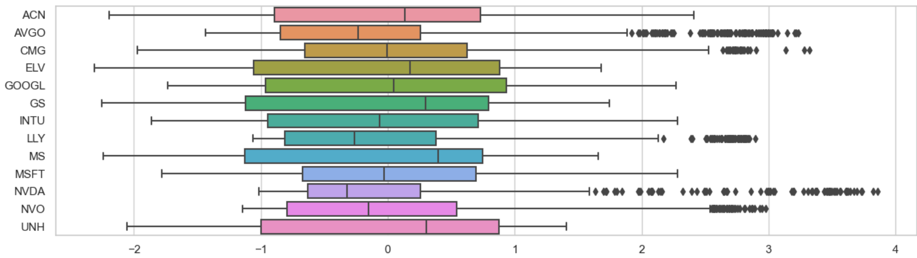

Figura 4.5: Box plot per serie di prezzi normalizzata

4.2 Ricerca della cointegrazione

In primo luogo, la condizione imprescindibile per quanto riguarda il funzionamento della strategia è la presenza di cointegrazione. Come anticipato nell'analisi teorica, la prima fase che permette di effettuare la verifica per la cointegrazione consiste nell'analisi della stazionarietà della serie storica considerata. Pertanto, è necessario verificare il cosiddetto ordine d'integrazione di ciascuna delle serie storiche in esame. Saranno considerati di conseguenza idonei rispetto alla verifica della cointegrazione, quelle serie storiche per le quali la loro combinazione risulterà integrata di ordine 0.

4.2.1 Analisi di stazionarietà

Come evidenziato a più riprese, nessuna serie storica di prezzi è stazionaria, ed affinché sia possibile mettere in atto una strategia di questo tipo, è necessario che lo siano gli stessi rendimenti. Ai fini di una verifica, come già riscontrato nella sezione 3.2.1, si mette in pratica un test d'ipotesi, l'Augmented Dickey Fuller Test sulla serie dei log-rendimenti, una delle proxy più utilizzate per il calcolo dei ritorni in percentuale.

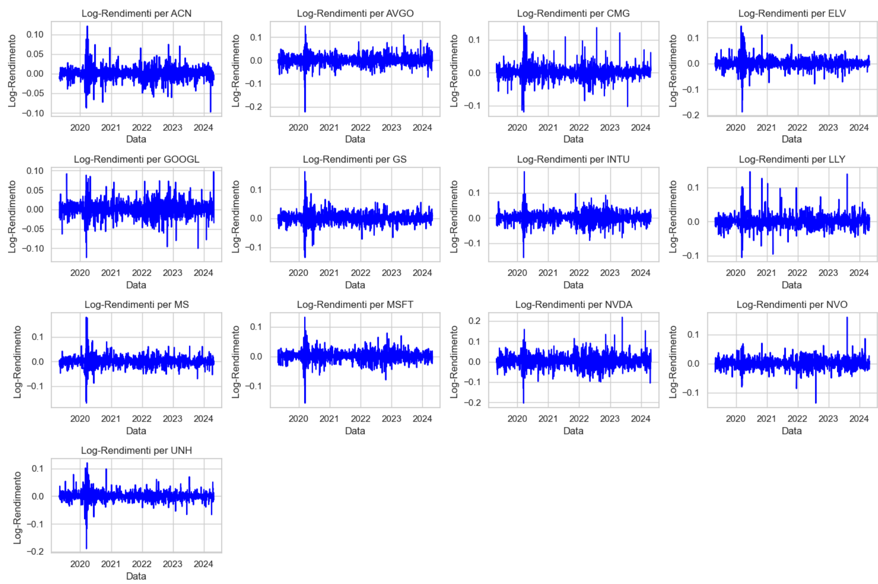

Figura 4.6: Log rendimenti per i 9 asset con correlazion più alta

Come riportato in tabella 4.3, l'ADF test per verificare la stazionarietà delle differenze prime dei prezzi ovvero i rendimenti, restituisce proprio il risultato aspettato dimostrando la stazionarietà per ciascuno dei titoli considerati, ed è proprio questo il punto di partenza che permette la verifica nel pratico dell'esistenza di una relazione tra asset.

Asset	Test Statistic	No. Lags	Critical value (1%)	Critical value (5%)	Critical value (10%)	P-Value
ACN	-10.57	9	-3.44	-2.86	-2.57	0.0 (Stazionaria)
GOOGL	-11.55	8	-3.44	-2.86	-2.57	0.0 (Stazionaria)
AVGO	-13.55	7	-3.44	-2.86	-2.57	0.0 (Stazionaria)
LLY	-11.79	8	-3.44	-2.86	-2.57	0.0 (Stazionaria)
NVO	-14.72	7	-3.44	-2.86	-2.57	0.0 (Stazionaria)
INTU	-11.72	9	-3.44	-2.86	-2.57	0.0 (Stazionaria)

Tabella 4.3: Risultati del test di Dickey-Fuller aumentato per ciascun asset

Il prossimo passo consiste nell'effettuare un' ulteriore scrematura sulla base di uno specifico test di cointegrazione mediante l'utilizzo di una funzione presente in una libreria specifica *statsmodel*: 'coint', che come suggerisce il nome stesso, permette di testare se due o più serie temporali non stazionarie siano cointegrate, ossia se esiste una combinazione lineare delle serie che è stazionaria. Tale test restituisce, ancora una volta come output il valore del pvalue e della statistica test. Fissando infatti un valore limite per il pvalue pari a 0.05 al di sotto del quale i titoli non sono considerati l'uno la combinazione lineare dell'altro, è possibile ottenere una *heatmap* che mostri il livello di significatività di tale rapporto di cointegrazione, e che restituisca ancora una volta quelle che possono essere considerate coppie di asset ,o appunto *pairs*.

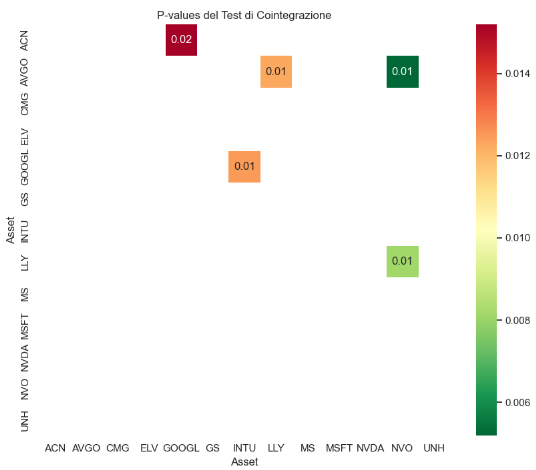

Figura 4.7: Heatmap di cointegrazione (livello di significatività 5%

Pertanto, a seguito della verifica del test per gli asset, il risultato è rappresentato dalle seguenti coppie: [('ACN', 'GOOGL'), ('AVGO', 'LLY'), ('AVGO', 'NVO'), ('GOOGL', 'INTU'), ('LLY', 'NVO')][3]

[3] Per i nomi degli asset vengono considerati i ticker riportati su Yahoo Finance

Engle e Granger test

A questo punto, un ulteriore step da mettere in atto rispetto ai titoli rimanenti, è quello introdotto da Engle & Granger [9] descritto nella sezione 3.3.1; si tratta di effettuare un test di causalità, ovvero di verificare nel pratico, se una variabile causa l'altra o viceversa. Per farlo, in Python è stata costruita una matrice, detta matrice di causalità, che verifica la significatività del rapporto causale della variabile x rispetto a y, e viceversa attraverso la statistica del p-value; se tale valore è più basso di una soglia standard fissata allo 0.05, allora può essere considerata l'ipotesi nulla come vera. Ciò implica di conseguenza, che l'evidenza per cui x non causa y può essere rifiutata: il rapporto di causalità è dunque verificato.

	ACN_x	GOOGL_x	AVGO_x	LLY_x	NVO_x	INTU_x
ACN_y	1.00	0.00	0.21	0.38	0.34	0.00
GOOGL_y	0.11	1.00	0.07	0.00	0.02	0.17
AVGO_y	0.00	0.00	1.00	0.00	0.00	0.02
LLY_y	0.37	0.21	0.01	1.00	0.00	0.03
NVO_y	0.50	0.03	0.00	0.02	1.00	0.12
INTU_y	0.04	0.06	0.16	0.08	0.10	1.00

Tabella 4.4: P-values della matrice di causalità

Come si può notare, la stima del coefficiente di causalità produce per diverse coppie risultati che dimostrano l'esistenza di causalità. Per semplicità espositiva, d'ora in avanti l'attenzione verrà spostata su due titoli:

- GOOGLE: ticker Yahoo Finance 'GOOGL';
- ACCENTURE: ticker Yahoo Finance 'ACN'.

4.3 La strategia nella pratica

Una volta definiti i 'pairs', occorre entrare nel vivo dell'analisi con l'obiettivo di attuare la strategia vera e propria, e conseguentemente definirne le caratteristiche principali.

Relativamente alla sua costruzione, diverse sono le variabili considerate:

- lo Z-score, attorno al quale ruota l'intera strategia;
- le thresholds rispetto cui aprire o chiudere le posizioni e quindi generare i segnali;
- i costi di transazione e il loro impatto rispetto al denaro generato;

Calcolo dello Z-score

Lo Z-score già menzionato nel paragrafo 3.5.1, è la misura sulla base della quale viene definita la strategia di Pair Trading in questa sede.

Si tratta di una variabile, che così com'è calcolata è già di per sé standardizzata. Tuttavia, per la specificità di questo studio, l'attenzione è "dirottata" verso un'altra variante dello z-score semplice: lo z-score classico calcolato sul semplice valore medio infatti non è in grado di considerare diverse prospettive. Per questa ragione vengono utilizzate due medie mobili con diverse finestre temporali: una a breve termine fissata a 5 giorni, e l'altra a lungo termine fissata a 60 giorni.

Prima di tutto deve essere definita una misura, sulla base del quale effettuare il calcolo statistico. Questa misura è definita come il rapporto tra i prezzi dei due asset (ratio). Di conseguenza, la serie storica che verrà utilizzata con la finalità di calcolare lo z-score è proprio quella che si basa sul rapporto dei prezzi definito come:

$$Ratio_t = \frac{P_t^{ACN}}{P_t^{GOOGL}} \tag{4.3}$$

A questo punto la statistica d'interesse può essere facilmente calcolata come segue:

$$Rolling_{Z-score_t} = \frac{MA_{rolling_{5d}} - MA_{rolling_{60d}}}{\sigma_{MA(rolling_{60d})}} \tag{4.4}$$

con:

- $MA_{rolling_{5d}}$ definita come la media mobile sul rapporto dei prezzi calcolata utilizzando una finestra temporale a 5 giorni, dunque nel breve periodo;
- $MA_{rolling_{60d}}$ definita come la media mobile sul rapporto dei prezzi calcolata utilizzando una finestra temporale a 60 giorni e quindi a medio-lungo periodo;
- $\sigma_{MA(rolling_{60d})}$ definita come la deviazione standard calcolata sulla media mobile sul rapporto dei prezzi utilizzando una finestra temporale a 60 giorni.

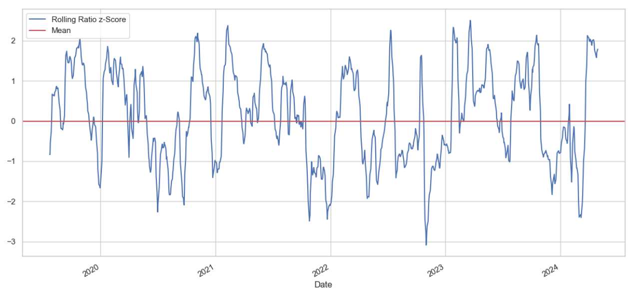

Figura 4.8: Rolling Z-score calcolato per sul rapporto dei prezzi di ACN e GOOGL

Definizione degli orizzonti temporali

Il processo di sviluppo, ottimizzazione e test di una strategia di trading, è essenziale per garantire che la strategia sia robusta, efficace, e pronta per l'implementazione in un ambiente di trading reale. A prescindere dal tipo di modello che si vuole utilizzare, è necessario dapprima suddividere il DataFrame in due parti, chiamate train e validation . Il Training del modello è definito come quel sottoinsieme del Dataset che viene utilizzato per l'addestramento del modello, solitamente circa il 70/80% dei dati. Tuttavia, talvolta è possibile che si incorra in un fenomeno chiamato *overfitting*, per il quale nonostante il modello si adatti bene ai dati storici passati non sia in grado di generalizzare sui dati nuovi. La seconda porzione di dati viene utilizzata per la fase di validation con la finalità ultima appunto di validare il modello stimato in precedenza. La percentuale totale rispetto all'intero Dataset, si attesta ad un livello compreso tra il 20% e il 30%, sebbene non vi sia una regola fissa le percentuali consentendo quindi di mantenere un certo livello di discrezionalità.

Nel caso specifico verrà utilizzata:

- per la fase di training il 70% del Dataset;
- per la fase di validation il restante 30%

Impostazione delle threshold

Nel momento in cui si effettua una strategia di arbitraggio statistico come in questo caso, la definizione dei segnali in entrata e in uscita è di fatto fondamentale. Tali segnali vengono realizzati sulla base dell'andamento di una variabile che si è scelta essere lo z-score. In questo ambito, diversi sono gli approcci che possono far funzionare la strategia. Com'è noto, è bene considerare le azioni di qualsiasi titolo, soggette a fluttuazioni nel corso del tempo. Tali fluttuazioni dipendono sia dalla componente idiosincratica, che da quella sistemica. Si rende pertanto necessaria una considerazione ulteriore, per cui tale mutevolezza cui sono sottoposti questi asset, non rende la strategia ugualmente efficace nel tempo.

Si potrebbe pensare a due macro approcci rispetto ai quali definire le condizioni di entrata e di uscita da una posizione:

- approccio statico;
- approccio dinamico.

Con il primo si vuole intendere una strategia mediante la quale vengono fissate delle soglie numeriche fisse, seguendo approcci rigorosi proposti in passato da alcuni economisti come Caldeira[2], Gatev [10]. Sebbene tale approccio abbia per costituzione un vantaggio importante, rappresentato dalla sua semplicità di utilizzo, non risulta efficace nel momento in cui si dovessero verificare cambiamenti delle condizioni di mercato o delle azioni sottostanti. Infatti, sebbene nell'orizzonte temporale considerato la strategia funzioni, non è detto che essa possa performare allo stesso modo in momenti in cui ad esempio, si manifesta un aumento della volatilità, oppure paradossalmente una stabilità eccessiva dei mercati.

Pertanto, in questa ricerca si tenterà di riprodurre questa strategia, utilizzando un approccio parametrico, ovvero volto ad identificare le soglie per i segnali in entrata e in uscita, che siano dipendenti dalle condizioni di mercato, che verranno approssimate alla serie della media mobile calcolata, sulla serie dello z-score, e alle relative statistiche di media e deviazione standard.

In modo particolare, si definiscono 3 soglie:

- *Upper Threshold*, di ingresso nelle posizioni per i rispettivi titoli e definita come:

$$\text{Upper Threshold} = MA(Z - score_{10d}) + 2 \cdot \sigma(MA(Z - score_{10d})) \tag{4.5}$$

- *Lower Threshold*, di ingresso nelle posizioni per i rispettivi titoli e definita come:

$$Lower\ Threshold = MA(Z - score_{10d}) - 2 \cdot \sigma(MA(Z - score_{10d})) \tag{4.6}$$

- *Exit Threshold*, di uscita dalle posizioni per i rispettivi titoli e definita come:

$$\text{Exit Threshold} = MA(Z - score_{10d}) \tag{4.7}$$

Per il motivo considerato parlando degli orizzonti temporali, la media mobile viene calcolata rispetto ad un periodo medio-breve, in modo da essere reattiva alle fluttuazioni di mercato, allo stesso tempo rimanendo abbastanza conservativi rispetto ad esse.

4.3.1 Implementazione della strategia

Dopo aver definito tutte quelle che sono le caratteristiche principali della strategia, non rimane che l'effettiva implementazione del modello sui titoli scelti.

Dotazione iniziale

Trattandosi di una strategia di arbitraggio *dollar neutral*, in maniera del tutto esemplificativa, è possibile in realtà arrivare ad investire a costo 0, ovvero senza denaro e posizioni in portafoglio.

In realtà, tale metodo richiede l'utilizzo dello short selling, costoso per svariate ragioni. In primo luogo, molto spesso la stessa piattaforma non permette di fare vendita allo scoperto in maniera gratuita, applicando delle commissioni all'investitore che così facendo specula sulla possibile discesa del titolo; inoltre, elemento ancora più importante, lo short selling richiede la presenza di un margine, ossia di un ammontare abbastanza consistente che funge da assicurazione o cuscinetto nel caso in cui, a seguito della vendita del titolo, il prezzo dello stesso salisse creando una situazione d'instabilità nel capitale dell'investitore, che a quel punto non sarebbe più in grado di restituire il titolo per via del prezzo troppo alto.

Per semplicità, in questa sede non verranno considerate questi due elementi.

Si ipotizza quindi una dotazione iniziale di capitale pari a 1000 $ e posizioni iniziali pari a 0 per ciascun titolo.

Acquisto e vendita degli asset

Per comprendere come vengono utilizzate le threshold definite nel paragrafo 4.3, è opportuno definire le condizioni di apertura e chiusura per ciascuna trade, tradotte nell'algoritmo che ne effettua la relativa applicazione.

Pertanto si rende necessario comprendere in che modo le soglie sono utilizzate. Come già in parte illustrato, i segnali di apertura si manifestano al superamento della soglia superiore (upper threshold) o di quella inferiore (lower threshold). In particolare, il segnale di vendita del *'ratio'* si attiva quando:

$$z - score[i] > UpperThreshold[i] \tag{4.8}$$

Quello che si sta affermando è che, nel momento in cui lo z-score supera la soglia superiore, il rapporto tra il prezzo dell'asset A e quello dell'asset B in un preciso istante è più alto rispetto a quello che può essere considerato come il range di equilibrio. In altre parole, il prezzo dello strumento finanziario A è molto più alto rispetto a quello di B. Pertanto, ci si aspetta che, affinché il rapporto torni alla normalità, il prezzo di A diminuisca e il prezzo di B aumenti e, di conseguenza, la strategia da attuare consiste nell'acquistare l'asset B e nel vendere l'asset A.

Il ragionamento analogo ma inverso, è quello che avviene nel caso opposto ossia nell'ipotesi di un rapporto tra i prezzi al di sotto del range che ne identifica l'equilibrio di lungo periodo e quindi di un prezzo di A inferiore a B; ciò comporterebbe la vendita di B e l'acquisto contestuale di A.

Ciò che non è stata definita è la quantità di ciascun asset, che per ogni trade deve essere acquistata e venduta.

Infatti, come ovvio che sia, il rapporto non può essere di 1:1, poichè dato che le azioni hanno prezzi diversi non sarebbe possibile né parlare di strategia dollar neutral, né tanto meno di arbitraggio. In questo modo si andrebbe infatti a creare un'esposizione maggiore su uno dei due titoli, e quindi non si potrebbe mettere in atto il Pair Trading.

Pertanto, si definisce la quantità che deve essere acquistata e quindi il peso di un titolo in funzione all'altro.

Per ogni unità acquistata (o venduta) dell'asset A (nel nostro caso GOOGLE), verrà venduta (o acquistata) una quantità pari a rapporto tra i prezzi al tempo t. Pertanto, se $Zscore > UpperThreshold$:

$$\begin{aligned} Q_{A_t} &= 1 \\ Q_{B_t} &= -Ratio_t \end{aligned} \tag{4.9}$$

Allo stesso modo, se $Zscore < LowerThreshold$:

$$\begin{aligned} Q_{A_t} &= -1 \\ Q_{B_t} &= Ratio_t \end{aligned} \tag{4.10}$$

Allo stesso modo si può definire la exit strategy, per la quale le posizioni in portafoglio vengono azzerate come:

$$\begin{aligned} |Zscore_t - ExitThreshold_t| &< 0.5 \\ dove: ExitThreshold_t &= MA(Zscore_t)_{10d} \end{aligned} \tag{4.11}$$

Il risultato, a seguito della costruzione dell'algoritmo, sarà dato dalla quantità di denaro risultante dall'iterazione del processo e dalle posizioni rimanenti, se presenti nel portafoglio, al termine del periodo di validation.

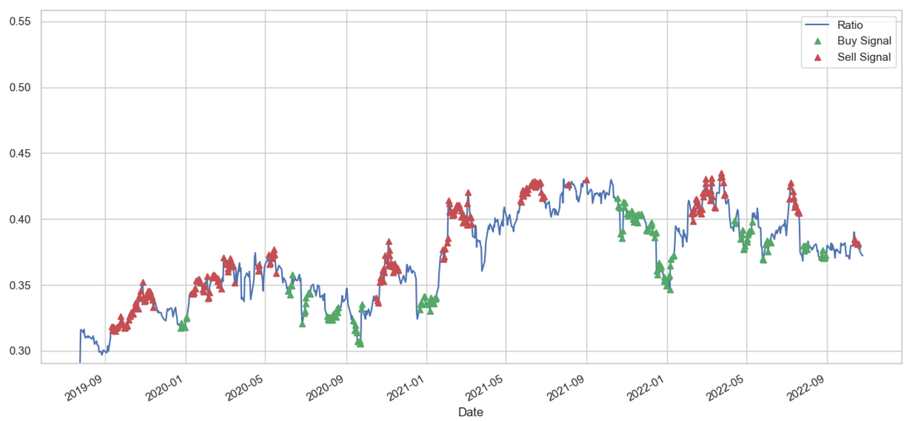

Figura 4.9: Segnali di compravendita Z-score e threshold statiche per GOOGL e ACN nel Training period

4.3.2 I costi di transazione

Quando si considera una qualsiasi strategia che comporta un investimento nei mercati finanziari, è fondamentale tener conto dei costi associati a tale attività. Questi costi possono influenzare significativamente la redditività della strategia. Anche se un metodo può sembrare profittevole,in teoria l'introduzione dei costi di transazione può erodere i guadagni attesi, rendendo la strategia meno vantaggiosa o addirittura non profittevole.

I costi di transazione includono diverse componenti, tra cui:

- Commissioni di brokeraggio: Sono le commissioni applicate dal broker o dalla piattaforma di trading utilizzata. Queste commissioni possono essere fisse (un importo fisso per transazione) o variabili (una percentuale del valore della transazione).
- Bid-Ask Spread: Il bid-ask spread rappresenta la differenza tra il prezzo di acquisto (ask) e il prezzo di vendita (bid) di un asset. Questo spread è un costo implicito per il trader, in quanto l'acquisto al prezzo ask e la vendita al prezzo bid comportano una perdita immediata pari allo spread.
- Commissioni per Short-Selling: Questi costi sono associati all'operazione di vendere allo scoperto (short selling). Solitamente, includono le commissioni per prendere in prestito i titoli e possono variare in base alla domanda e offerta del titolo stesso.
- Slippage: Lo slippage si verifica quando c'è una differenza tra il prezzo desiderato di esecuzione di un ordine e il prezzo effettivo di esecuzione. Questo fenomeno è comune nei mercati ad alta volatilità o con bassa liquidità, dove i prezzi possono cambiare rapidamente durante l'esecuzione dell'ordine.

Nel contesto dell'analisi, consideriamo coppie di azioni quotate nel mercato statunitense. Queste azioni, come noto, sono caratterizzate da un'elevata liquidità, riducendo così l'impatto potenziale del bid-ask spread e dello slippage, che per semplicità espositiva verranno trascurati nel corso dell'esecuzione della strategia. Tuttavia, è essenziale considerare i costi di transazione per valutare correttamente la redditività dell'applicazione di questo metodo.

Calcolo dei Costi di Transazione Medi

Per calcolare i costi di transazione medi, prendiamo in considerazione sia una componente fissa che una variabile, basate sullo spread medio durante i periodi di allenamento (training periods). Come riferimento per le commissioni di brokeraggio, utilizziamo i dati di una piattaforma di trading ben conosciuta, DEGIRO.

Secondo DEGIRO[4], le commissioni per gli strumenti finanziari categorizzati come azioni sono strutturate come segue: le azioni che rientrano nella categoria "USA" hanno una commissione di transazione pari a € 1, ma una commissione di gestione di € 1.

Per semplicità, applicheremo una commissione media di €2 per transazione. Questo valore medio rappresenta una stima ragionevole, che tiene conto delle diverse commissioni applicabili.

4.3.3 Lo spread

Lo *spread* è un concetto fondamentale nei mercati finanziari, e rappresenta una delle principali voci di costo per gli investitori e i trader. Esso indica la differenza tra il prezzo denaro (bid) e il prezzo lettera (ask) di uno strumento finanziario, noto anche come bid-ask spread. Questo differenziale riflette vari aspetti del mercato, come la liquidità, il rischio e l'efficienza, ed è un indicatore critico per comprendere i costi di transazione e la dinamica della domanda e dell'offerta.

Le determinanti dello spread sono:

- Prezzo Denaro (Bid): È il prezzo massimo che un acquirente è disposto a pagare per un'azione, un'obbligazione o un altro strumento finanziario.
- Prezzo Lettera (Ask): È il prezzo minimo al quale un venditore è disposto a vendere lo stesso strumento.

Lo spread ha diverse implicazioni. In primo luogo rispetto al costo di transazione, rappresenta un costo implicito per l'investitore, poiché chi acquista pagherà il prezzo ask mentre chi vende riceverà il prezzo bid. La differenza tra i due prezzi, rappresenta un costo diretto per l'esecuzione della transazione. Funziona inoltre come indicatore di liquidità, poiché valori più stretti (e quindi differenze bid-ask minori) indicano solitamente un mercato più liquido e competitivo, dove è facile comprare e vendere rapidamente senza grandi variazioni di prezzo. Al contrario, spread più ampi possono indicare minore liquidità, con maggiori difficoltà nel trovare controparti per le transazioni. Questo si riversa inevitabilmente sui mercati, quando spread ampi possono riflettere un maggiore livello di rischio percepito, come l'alta volatilità o l'incertezza riguardo il valore reale dello strumento finanziario. Diversi fattori possono influenzare lo spread. Innanzitutto, mercati con elevati volumi di scambio, tendono ad avere spread più ridotti grazie alla maggiore disponibilità di acquirenti e venditori. In secondo luogo, in periodi di alta volatilità, gli spread possono aumentare poiché i market maker aumentano i loro margini, per compensare il rischio di rapidi movimenti di prezzo. Nello specifico, per i titoli considerati, sempre considerando DEGIRO, lo spread medio è pari a:

- Per Accenture: spread medio pari a 0.1 €;
- Per Google: spread medio pari a 0.02 €.

Capitolo 5

Risultati e gestione del rischio

Dopo aver spiegato il funzionamento della strategia, è necessario analizzare quelli che sono i risultati che la stessa permette di raggiungere.

Innanzitutto, a differenza di quanto fatto nel capitolo precedente, il periodo considerato è diverso, perché i risultati a cui si perviene si basano su un'altra porzione del dataset, ovvero il restante 20%, che nel pratico significa avere dati a partire dal 28.04.2022 fino al 29.04.2024.

Date	GOOGL	ACN
2022-10-27	92.22	278.84
2022-10-28	96.29	287.78
2022-10-31	94.51	283.90
2022-11-01	90.47	281.47
2022-11-02	86.97	272.45
...	...	...
2024-04-23	158.26	316.83
2024-04-24	159.13	313.54
2024-04-25	156.00	309.00
2024-04-26	171.95	308.01
2024-04-29	166.15	303.16

Tabella 5.1: Valori delle azioni GOOGL e ACN durante il validation period

In generale, quello che si può osservare, per entrambe le serie storiche nel periodo considerato, è un aumento generale del prezzo nel corso del tempo, sebbene quello di Google sia più che proporzionale rispetto ad Accenture che nell'ultimo periodo mostra una leggera flessione.

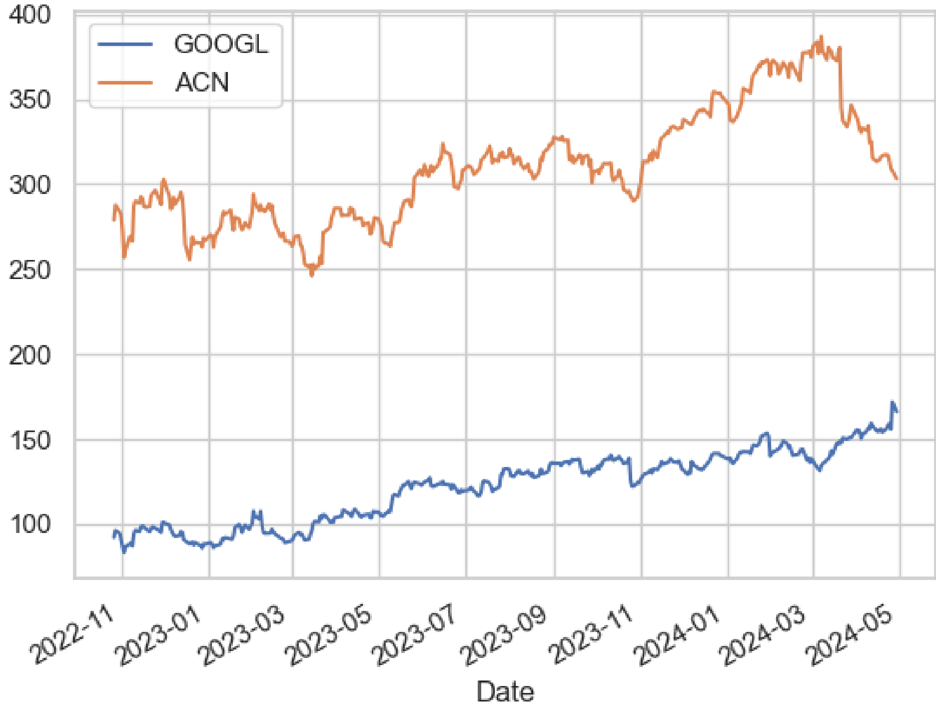

Figura 5.1: Serie storica prezzi ACN e GOOGL durante il validation period

5.1 La definizione delle trade

La funzione trade, definita sulla base delle caratteristiche menzionate nel capitolo 4, ha al suo interno diversi input:

- Le 2 serie storiche di prezzi S1 e S2, separatamente considerate;
- 2 finestre temporali sulla base della quale definire l'ampiezza delle medie mobili:
  - Window 1, che definisce la finestra di media mobile di breve periodo;
  - Window 2, che definisce la finestra di media mobile di medio-lungo periodo.
- I costi di transazione che vengono calcolati nello stesso modo in cui sono stati presentati nella sezione 4.3.2 e che di default sono pari a 2;
- Z-score window, per tale intendendosi la finestra temporale della media mobile utilizzata per calcolare la threshold dinamica e che è stata selezionata pari a 10 per le ragioni spiegate in precedenza.

In aggiunta, possono essere impostati ulteriori elementi quali il denaro iniziale (money), posto = 100$ in quanto non rilevante ma necessario solo per studiarne i ritorni e l'ammontare di posizioni detenute inizialmente, impostate a 0 per entrambe le azioni.

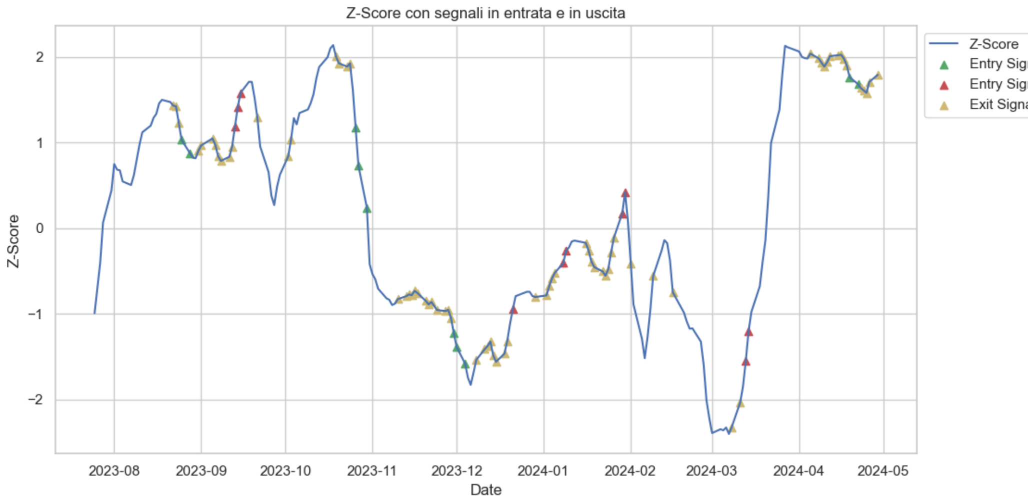

Figura 5.2: Segnali di trading sullo Z-score basato sul rapporto dei prezzi

Successivamente alla definizione della funzione, non resta che la sua applicazione pratica. Nella figura 5.2 vengono mostrati i segnali di compravendita nonché quelli di uscita dalle posizioni investite sulla base del livello dello z-score, in virtù delle soglie dinamiche definite in precedenza.

5.2 Gestione delle posizioni e valore finale del portafoglio

In generale, e come evidenziato dalla figura 5.3, è opportuno considerare che il valore dell'investimento nel corso dell'implementazione dell'algoritmo possa variare e anche in modo significativo al variare delle posizioni in portafoglio. È proprio questa volatilità la principale componente di rischio, che si riflette nell'investitore che adotta tale strategia.

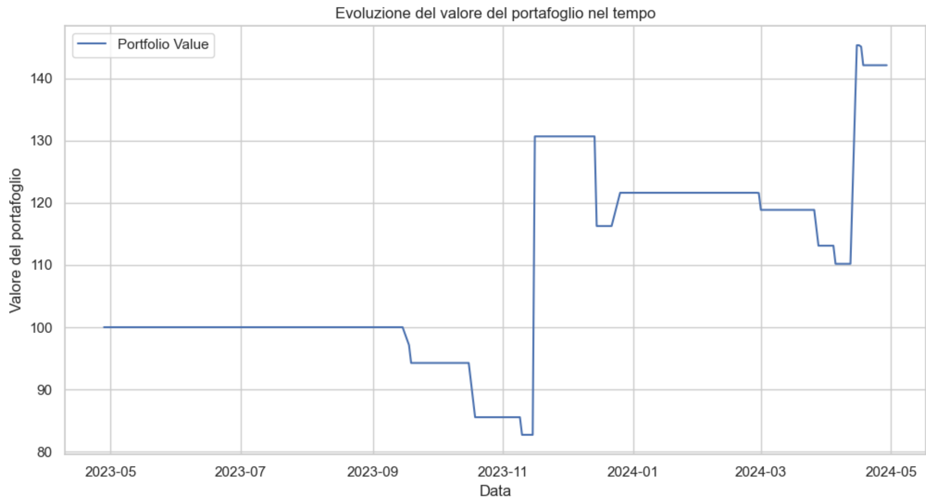

Figura 5.3: Valore del portafoglio nel tempo

Il valore del portafoglio varia nel corso del tempo in maniera piuttosto significativa. In un orizzonte temporale di un anno (dal 28/04/2023 al 29/04/2024) la variazione registrata è di circa il 20% ,in aumento e in diminuzione; ciò significa una variazione piuttosto levata di circa il 40%.

Al 29/04/2024 si registra una posizione short su Google esattamente pari a -2 azioni e una long su Accenture pari a circa 1.101. Ovviamente questi non sono dei pesi ma delle quantità. Per ottenere i primi, è necessario un ulteriore step che può essere facilmente eseguito utilizzando la formula che segue:

$$\begin{aligned} Weight_{GOOGL} &= \frac{Posizioni_{GOOGL}}{Posizioni_{GOOGL} + Posizioni_{ACN}} = 1.97 \\ Weight_{ACN} &= \frac{Posizioni_{ACN}}{Posizioni_{GOOGL} + Posizioni_{ACN}} = -0.97 \end{aligned} \tag{5.1}$$

Come è possibile notare i pesi del portafoglio sommano sempre a 1.

5.2.1 Analisi dei rendimenti

La funzione trade restituisce inoltre il valore del portafoglio al termine dell'investimento, che risulta pari a 142.12 \$ con una performance YTD pari a +14% circa e ritorno rispetto all'investimento iniziale pari a 42%, mostrando una deviazione standard complessiva pari a 14.95%.

Uno degli indicatori più rilevanti, per comprendere quanto la strategia messa in atto possa essere efficace, è lo Sharpe Ratio già definito nella sezione 3.4.1. Prendendo come proxy per il tasso risk free al 30/04/2024 il valore di riferimento di un Treasury 1Y pari a 5.25%[1] e avendo a riferimento il valore della voloatilità della strategia pari a 15.5% è possibile calcolare i principali indicatori i redditività per l'applicazione del metodo. Considerando il rendimento annuale menzionato in precedenza pari a circa 42%, lo sharpe ratio è uguale a 2.40.

Dunque, in questo caso siamo di fronte ad un ottimo rendimento poiché superiore a 2; per ogni unità di deviazione standard assunta,il portafoglio genera 2.40 unità di rendimento in eccesso.

In termini pratici, anche qui un valore positivo e superiore a 1 è generalmente considerato buono, indicando che il portafoglio ha generato un rendimento sufficiente rispetto al rischio di downside. Un valore di 0.60 suggerisce, che il portafoglio ha un rendimento relativamente moderato rispetto alla sua volatilità downside. Non è eccezionale, ma neanche negativo. È importante considerare questo valore nel contesto di altri fattori, come gli obiettivi di investimento, il profilo di rischio dell'investitore e le condizioni di mercato.

5.3 Considerazioni sul rischio

Tra i modelli utilizzati per le considerazioni sul rischio c'è sicuramente il criterio del VaR, menzionato già nella sezione 3.5.3 e conseguentemente del cVaR, calcolati secondo diversi criteri: quello storico e quello parametrico.

5.3.1 VaR e cVaR storico

Per calcolare il VaR e il cVaR storico sono state utilizzate delle funzioni che considerano la serie dei ritorni storici con il 5° percentile della distribuzione. L'assunzione fondamentale in questo caso è la normalità dei rendimenti.

La definizione della funzione per calcolare il VaR e il cVaR richiede anche la definizione di un intervallo temporale all'interno del quale calcolare entrambi i valori. Occorre considerare, che data la frequenza scelta ovvero quella giornaliera, non avrebbe senso calcolare entrambi i valori su un intervallo temporale di un solo giorno. Per questo il calcolo è stato effettuato su un orizzonte comunque breve di 5 giorni, ma che rende significative le stime in termini di interpretazione dei risultati, in un'ottica d'investimento continuativa.

Value at Risk (VaR) 95th CI Il Value at Risk (VaR) al 95% di intervallo di confidenza è 17.21 $ quantificando la perdita massima attesa su un portafoglio in un determinato

[1] Valori di riferimento disponibili su https://www.cnbc.com/quotes/US1Y

Metrica	Valore
Value at Risk 95th CI	$ 17.21
Conditional VaR 95th CI	$ 23.78

Tabella 5.2: Riepilogo delle metriche del VaR e cVaR per intervallo temporale di 5 giorni

periodo di tempo con un dato livello di probabilità in questo caso appunto il 95% che le perdite del portafoglio non superino 17.21 $ calcolato su un intervallo di 5 giorni. Questo significa che, in condizioni normali di mercato, le perdite settimanali del portafoglio (5 sono i trading day di una settimana) non dovrebbero superare questo valore nel 95% dei casi.

Il Conditional Value at Risk (CVaR) al 95% di intervallo di confidenza è 23.78 indicando la perdita media del portafoglio nelle peggiori condizioni, cioè nel 5% dei casi. Questo valore fornisce una stima della gravità delle perdite in condizioni estreme di mercato e complementa il VaR e in questo caso indica che, nelle situazioni più sfavorevoli, le perdite medie potrebbero raggiungere 23.78 $.

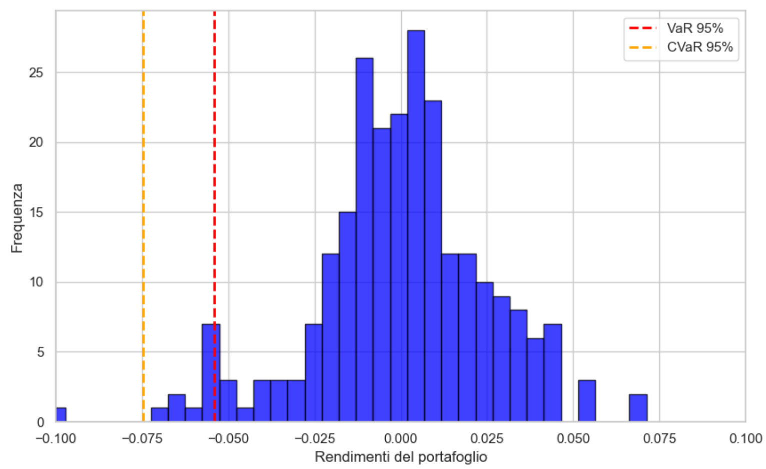

Figura 5.4: Distribuzione dei rendimenti del portafoglio con VaR e CVaR (weekly) storico

5.3.2 VaR e cVaR parametrico

Un approccio diverso consiste nell'utilizzo del metodo parametrico per la stima dei due indicatori di rischio utilizzando due distribuzioni diverse per ciascuno (normale o t-Student).

In particolare, le funzioni implementate in Python utilizzano le funzioni di quantile inverse (norm.ppf) della distribuzione normale per calcolarli:

$$VaR = norm.ppf(\frac{1-\alpha}{100}) \cdot \sigma_{Portfolio} - E(returns_{Portfolio}) \tag{5.2}$$

Analogamente, se considerata la t-Student come distribuzione a quel punto sarà necessario definire un'altra funzione di quantile inversa (t.ppf):

$$VaR = np.sqrt(\frac{(\nu-2)}{\nu}) \cdot t.ppf(\frac{1-\alpha}{100}, \nu) \cdot \sigma_{Portfolio} - E(returns_{Portfolio}) \tag{5.3}$$

dove ν sono i gradi di libertà.

Per quanto riguarda invece il Conditional VaR, il discorso è un po' più complicato, anche se allo stesso tempo fornisce una valutazione della gravità delle perdite in quei casi estremi, offrendo quindi una visione più completa del rischio.

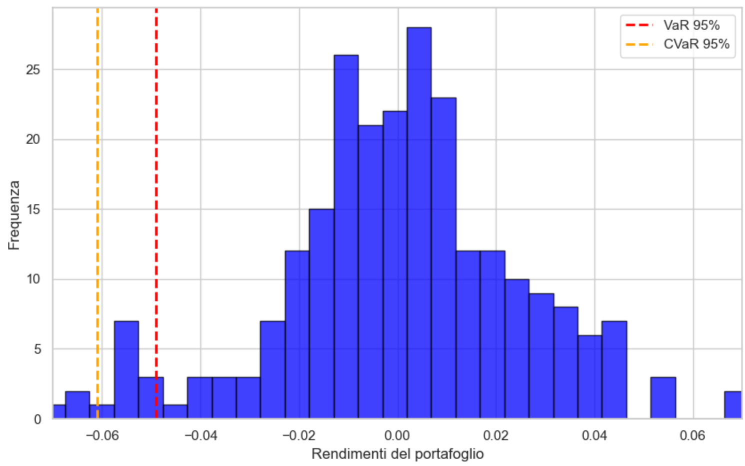

Figura 5.5: Enter Caption

Figura 5.6: Distribuzione dei rendimenti del portafoglio con VaR e CVaR (weekly) parametrici

CVaR con distribuzione normale

Quando si utilizza una distribuzione normale, la formula per calcolare il CVaR è data da:

$$\text{CVaR} = \frac{1}{\alpha} \int_{-\infty}^{VaR_\alpha} x \cdot f(x)\, dx \tag{5.4}$$

dove:

- α è il livello di confidenza desiderato;
- VaR_α è il VaR al livello di confidenza considerato;
- $f(x)$ è la funzione di densità di probabilità della distribuzione normale (o t-Student nel caso di distribuzione t).

L'implementazione pratica, per quanto riguarda l'utilizzo di una funzione di distribuzione normale, è la seguente:

$$CVaR = \frac{1}{\alpha} \cdot \phi(\Phi^{-1}(\alpha)) \cdot (\sigma - \mu) \tag{5.5}$$

dove α rappresenta il livello di confidenza, norm.pdf è la funzione di densità di probabilità della distribuzione normale, norm.ppf è il quantile della distribuzione normale, portfolioStd è la deviazione standard del portafoglio e portfolioReturns è il rendimento atteso del portafoglio.

Nel caso della distribuzione t-Student, la formula per il CVaR diventa:

$$CVaR = -\frac{1}{\alpha} \cdot \frac{1}{1-\nu} \cdot (\nu - 2 + x_{\alpha,\nu}^2) \cdot t(x_{\alpha,\nu}, \nu) \cdot (\sigma - \mu) \tag{5.6}$$

dove:

- α è il livello di confidenza (= 0.05 nel caso specifico);
- ν rappresenta i gradi di libertà della distribuzione t-Student;
- $x_{\alpha,\nu}$ è il quantile della distribuzione t-Student;
- $t(x_{\alpha,\nu}, \nu)$ è la funzione di densità di probabilità della distribuzione t-Student.

Metrica	Valore
Normal VaR 95th CI	15.60
Normal CVaR 95th CI	19.37
t-dist VaR 95th CI	15.07
t-dist CVaR 95th CI	20.01

Tabella 5.3: Riepilogo delle metriche del VaR e cVaR parametrico per intervallo temporale di 5 giorni

La tabella presenta una panoramica delle metriche di rischio, calcolate per il portafoglio. Il Value at Risk (VaR) rappresenta la massima perdita attesa, con una probabilità specifica durante un intervallo temporale. Utilizzando una distribuzione normale, il VaR al 95% di confidenza è stimato a 15.60 $, indicando che c'è una probabilità del 5% che la perdita superi questo valore.

Il Conditional Value at Risk (CVaR), invece, va oltre il VaR e fornisce una stima della media delle perdite che eccedono il VaR, pesate dalla probabilità di coda. Il CVaR al 95% di confidenza, calcolato con una distribuzione normale, è pari a 19.37 $, suggerendo che in caso di perdite che superano il VaR, ci si aspetta che esse siano in media di questo valore.

Esaminando i risultati ottenuti, ed utilizzando una distribuzione t-Student anziché una normale, notiamo leggere differenze. Il VaR al 95% di confidenza con una distribuzione t-Student è leggermente inferiore a 15.07 $, suggerendo una maggiore incertezza rispetto al caso normale. Similmente, il CVaR con distribuzione t-Student è stimato a 20.01 $, riflettendo una distribuzione dei rendimenti più pesante nelle code rispetto alla distribuzione normale. La distribuzione dei rendimenti, e il risultato delle nuove statistiche possono essere analogamente mostrate in figura 5.7.

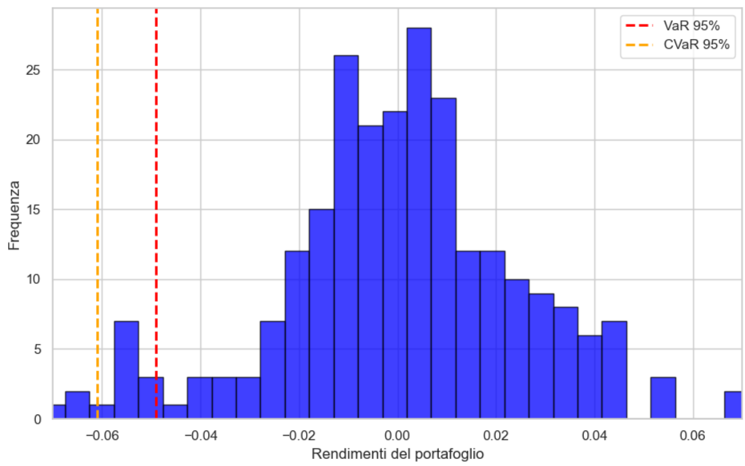

Figura 5.7: Distribuzione dei rendimenti con VaR e CVaR parametrici (weekly)

Downside risk

Un altro indicatore trattato nelle sezioni precedenti, è quello del rischio di Downside risk la cui logica è in realtà abbastanza simile rispetto a quanto visto con il Sortino ratio. Come è stato possibile osservare, viene calcolato utilizzando la deviazione standard dei rendimenti negativi del portafoglio, poiché si concentra solo sulle perdite piuttosto, che sull'intero intervallo dei rendimenti. A differenza delle frequenze di analisi dei dati per quanto riguarda l'interpretazione delle misure economiche (quali Sharpe Ratio e Sortino

Ratio) e in ossequio rispetto al modus operandi osservato nel calcolo di VaR e CVaR nei due metodi sebbene questi ultimi utilizzino delle finestre temporali più estese di 5 giorni, il focus è in questo caso sui rendimenti giornalieri negativi.

Nel caso specifico, approssimando il MAR al rendimento giornaliero del Treasury 1Y, con una serie di rendimenti negativi di lunghezza pari a n = 69, è possibile calcolare il Downside risk giornaliero come segue:

$$DownsideRisk = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(R_i^- - MAR)^2} = \sqrt{\frac{1}{69}\sum_{i=1}^{69}(R_i^- - \frac{0.0525}{252})^2} = 0.040 \quad (5.7)$$

Un Downside Risk di circa 0.040 indica che, in media, i rendimenti negativi del portafoglio rispetto al tasso minimo accettabile di rendimento (MAR) si discostano di circa 0.040 dalle loro medie. Questo suggerisce che vi è una certa volatilità nei rendimenti negativi rispetto al MAR, con rendimenti che possono variare di circa 0.040 rispetto al tasso minimo accettabile, interpretabile come una volatilità dei rendimenti negativi del portafoglio rispetto al tasso minimo accettabile relativamente bassa.

Il calcolo del Drawdown

In questa sede verrà utilizzata Max DD (Daily): calcolato considerando solo i risultati giornalieri. Riprendendo la definizione fornita da Hull, John C. (2012)[14], può essere definito come:

$$Drawdown\% = \frac{Pmax - Pmin}{Pmax} \cdot 100 \quad (5.8)$$

dove con *Pmax* e *Pmin* s'intendono i valori di picco e di minimo registrati di volta in volta nell'intervallo temporale considerato. Il max Drawdown, definito come la percentuale massima di Drawdown registrata nell'intervallo di tempo risultante è pari a 19%, considerabile dunque come sostanziale e potenzialmente preoccupante per gli investitori e i trader. Infatti, una riduzione del 19% può comportare una notevole perdita di valore nel portafoglio, richiedendo un recupero significativo per tornare ai livelli di profitto precedenti.

Tuttavia, è importante valutarlo nel contesto della strategia di trading complessiva e del piano di gestione del rischio in quanto un livello di questo tipo potrebbe essere accettabile, se è stato pianificato e gestito adeguatamente all'interno di un piano di trading prudente e ben strutturato.

Inoltre, è essenziale considerare il periodo di tempo durante il quale si è verificato il Drawdown e la capacità della strategia di recuperare da questa fase di perdita, poiché potrebbe essere temporaneo e seguito da un periodo di ripresa, o potrebbe indicare problemi più profondi nella strategia, richiedendo un'attenta valutazione e possibili modifiche.

5.3.3 Metodo Monte Carlo e forecasting del valore del portafoglio

Nell'ambito dell'applicazione di una qualsivoglia strategia di investimento, è naturale che l'investitore fronteggi un certo livello di incertezza nell'effettuare delle previsioni, o nello stimare in un certo modo il valore futuro di una determinata variabile che spesso viene approssimata al valore medio. Ovviamente, una stima di questo tipo, porta con sé evidenti limiti che non permettono di arrivare a livelli di accuratezza elevati, ritenuti fondamentali in una strategia come quella del Pair Trading. Proprio per questa ragione, uno dei metodi utilizzabili è quello relativo alla simulazione di Monte Carlo. Inventato negli anni '40 da John von Neumann e Stanislaw Ulam, tale metodo utilizza valori multipli e simulazioni in una vastissima gamma di ambiti, che sono contraddistinti dall'esistenza di incertezza e in generale di variabili casuali.

Il presupposto è questa casualità, che non permette di prevedere con relativa sicurezza l'andamento futuro delle variabili (nel caso specifico i prezzi delle azioni). Pertanto, il metodo Monte Carlo assegna alle variabili del modello dei valori casuali basati su una distribuzione (la normale), ripetendo più volte il processo, fino ad ottenere un numero n di simulazioni. Una volta arrivati al termine delle simulazioni, i risultati vengono calcolati attraverso la media degli stessi per giungere ad una stima complessiva.

Di seguito, viene fatta una breve descrizione del modello riprodotto in Python per eseguire le simulazioni.

Il primo passo è quello di definire i rendimenti, nel nostro caso giornalieri. Successivamente si definiscono una serie di variabili tra cui la varianza e la media dei rendimenti che sono necessari ai fini del calcolo di un'ulteriore componente: il Drift, ovvero quella componente che nelle serie storiche per i prezzi delle azioni crea il trend di lungo periodo. Esso può essere definito come:

$$Drift = AverageRet - \sigma^2/2 \tag{5.9}$$

Quest'ultima componente può essere talvolta anche trascurata, soprattutto per periodi brevi, dove non si notano enormi differenze tra le serie con e senza trend.

In seguito, si determina la componente casuale. Attraverso la distribuzione normale, si genera un input di tipo randomico che modifica la volatilità dei rendimenti come:

$$\sigma_{rand} = \sigma \cdot RandomValue_{Normal} \tag{5.10}$$

Infine, il prezzo del titolo o il valore del portafoglio è definibile come segue:

$$P_{t+1} = P_t \cdot e^(Drift + RandomValue_{Normal}) \tag{5.11}$$

Ovviamente questo processo può essere reiterato, e può portare alla stima del valore futuro del portafoglio per periodi ben più lunghi.

Nella figura 5.8 viene mostrata la simulazione Monte Carlo per il valore del portafoglio; nel nostro caso, sono state scelte un numero n = 300 simulazioni per un intervallo temporale complessivo pari a 200 giorni. È importante pensare che aumentando il numero di giorni considerati per effettuare le simulazioni, il modello perde di precisione; allo stesso tempo, all'aumentare del numero di simulazioni le stime possono ottenere un grado di correttezza maggiore.

Inoltre, il valore iniziale del portafoglio è pari al valore registrato al 29/04/2024, pari a $ 142.12 con i rispettivi pesi nei due titoli pari a 1.97 per GOOGLE, -0.97 per ACCENTURE.

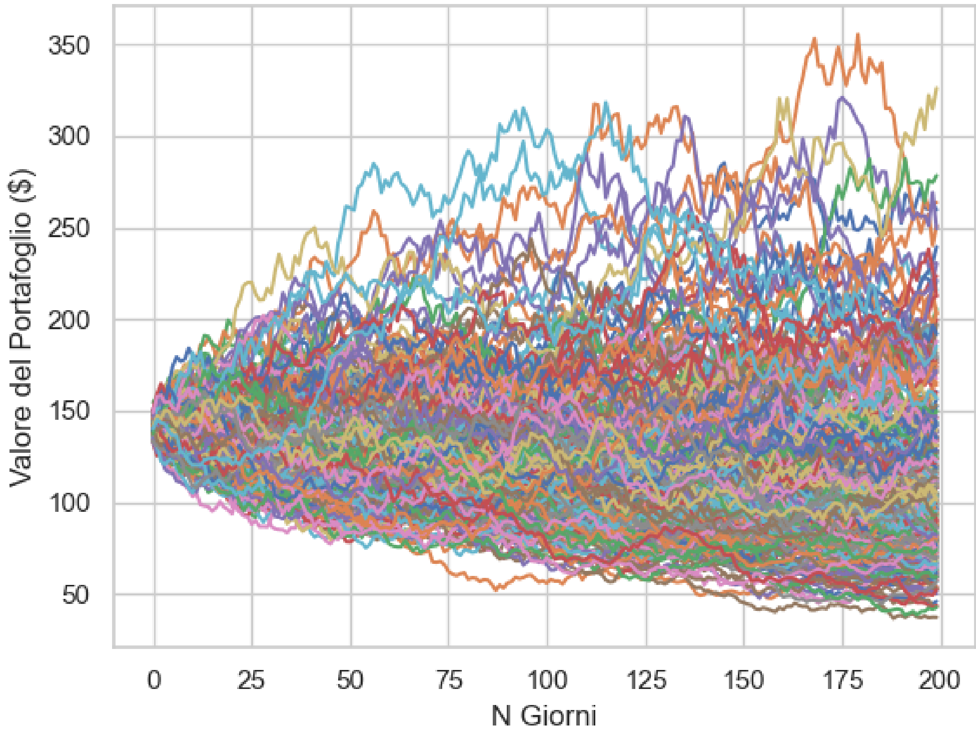

Figura 5.8: Simulazione di Monte Carlo sul portafoglio

VaR Monte Carlo

A partire da questo è possibile ancora una volta stimare il Value at Risk Monte Carlo, un'ulteriore metrica, ancora più precisa e utilizzata per monitorare i principali investimenti nell'ambito del risk management.

Nello specifico, la logica è sempre la stessa: attraverso delle simulazioni sulle serie storiche dei rendimenti è possibile trovare il valore del VaR con un certo livello di confidenza.

In genere gli intervalli di confidenza presi in considerazione sono il 5% e l'1% e sono pari a:

VaR Level	Simulated VaR
5%	-6.27
1%	-8.18

Tabella 5.4: VaR Monte Carlo per diversi CI

È bene considerare tuttavia che a differenza delle stime precedenti, qui i valori calcolati fanno riferimento ad un orizzonte temporale di 1 giorno.

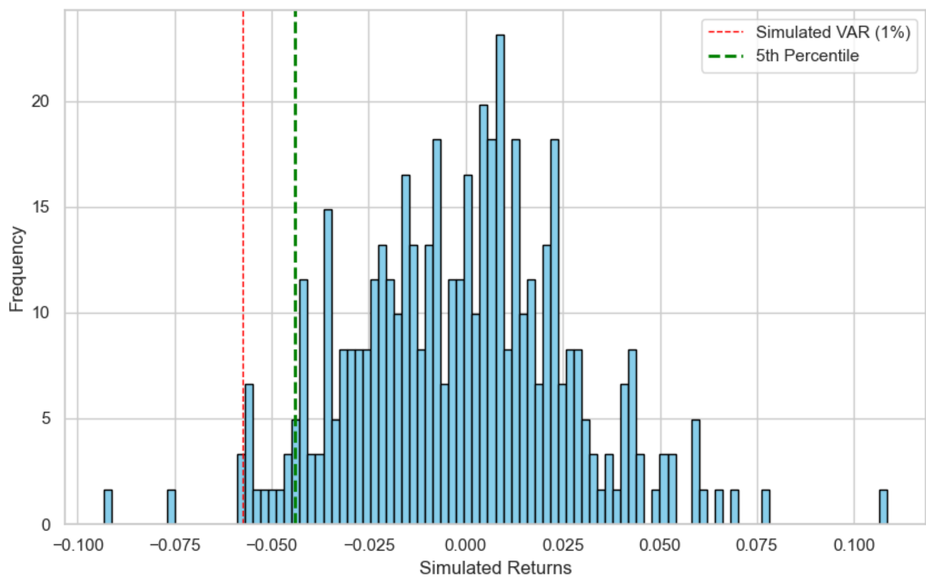

Figura 5.9: Distribuzione dei rendimenti e calcolo del VaR con simulazione di Monte Carlo

Capitolo 6

Il Pair Trading e la diversificazione del rischio

In questo sesto e ultimo capitolo della ricerca, l'obiettivo è quello di dimostrare come, attraverso l'utilizzo del Pair Trading, sia possibile dare vita ad una strategia di diversificazione di un portafoglio preesistente, misurandone i relativi impatti in termini di rendimenti realizzati ma soprattutto di rischiosità. Per farlo, si rende necessario il richiamo di alcuni concetti di teoria di ottimizzazione del portafoglio tra i quali è impossibile non menzionare quello di Markowitz.

6.1 Analisi del portafoglio

Al fine di effettuare tutte le analisi utili a dimostrare la premessa, occorre partire dalla creazione di un portafoglio, avvalendoci della teoria di ottimizzazione del portafoglio proposta da Markowitz (1952)[23]. Tale modello permette di costruire, a partire da un numero n di titoli, un portafoglio in un istante iniziale t,la cui composizione massimizzi i rendimenti, e allo stesso tempo minimizzi la rischiosità.

Pertanto, si dirà che un portafoglio $A \succ B$, ossia che A è dominante rispetto ad un portafoglio B, se a parità di rischio possiede un rendimento più elevato, o viceversa quando a parità di rendimento si ha un rischio inferiore. L'insieme di tutti i portafogli dominanti è costruito proprio a partire da tutti quelli che per ciascun livello di rendimento minimizzano la componente di rischiosità, e costituisce la cosiddetta "frontiera efficiente". È bene precisare però, che la relazione di dominanza al di sopra della frontiera non è assoluta, vale a dire non sempre è possibile confrontare due portafogli e stabilirne un ordine.

6.1.1 Scelta dei titoli

Per la realizzazione di questo studio si continuerà sulla falsa riga di quanto finora esposto, selezionando asset a partire da quelli già utilizzati per lo studio relativo al Pair Trading del capitolo precedente, presenti all'interno dello stesso indice: l'S&P 500.

A tal riguardo, si ritiene che una minore correlazione tra gli asset sia un vantaggio, in quanto ciò sia favorevole ad una maggiore diversificazione del portafoglio. Tuttavia, lo studio si concentra sull'analisi degli effetti derivanti dall'introduzione di una strategia di Pair Trading in un portafoglio già esistente, e non sulla costruzione del portafoglio perfetto, sebbene si faccia comunque riferimento ad una strategia di ottimizzazione come quella proposta da Markowitz.

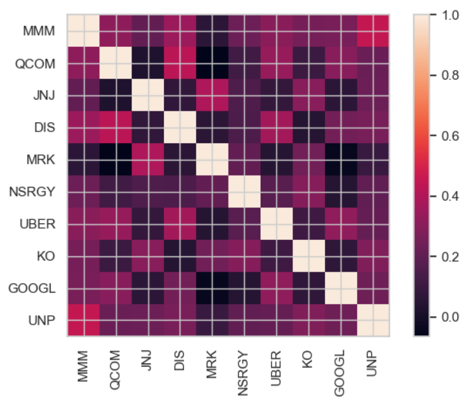

Figura 6.1: Matrice di correlazione per la costruzione del portafoglio

Dalla ricerca degli asset con minore correlazione, nel periodo compreso tra il 30/12/2022 e il 29/12/2023, è stata identificata la lista di titoli e la rispettiva quotazione, che è possibile in parte osservare attraverso la tabella 6.1, e la serie storica proposta con la figura 6.2.

Date	MMM	QCOM	JNJ	DIS	MRK	NSRGY	UBER	KO	GOOGL	UNP
2023-12-22	88.90	143.49	155.46	91.02	107.70	113.44	61.71	58.32	141.49	243.58
2023-12-26	90.39	145.46	156.14	90.95	107.63	113.89	61.98	58.56	141.52	245.29
2023-12-27	90.92	145.72	156.35	90.38	107.98	114.70	63.28	58.71	140.37	245.81
2023-12-28	91.71	145.86	156.58	90.40	108.77	114.73	63.14	58.75	140.23	246.02
2023-12-29	91.40	144.63	156.74	90.29	109.02	115.63	61.57	58.93	139.69	245.62

Tabella 6.1: Coda della serie storica dei prezzi delle azioni dal 22 dicembre 2023 al 29 dicembre 2023

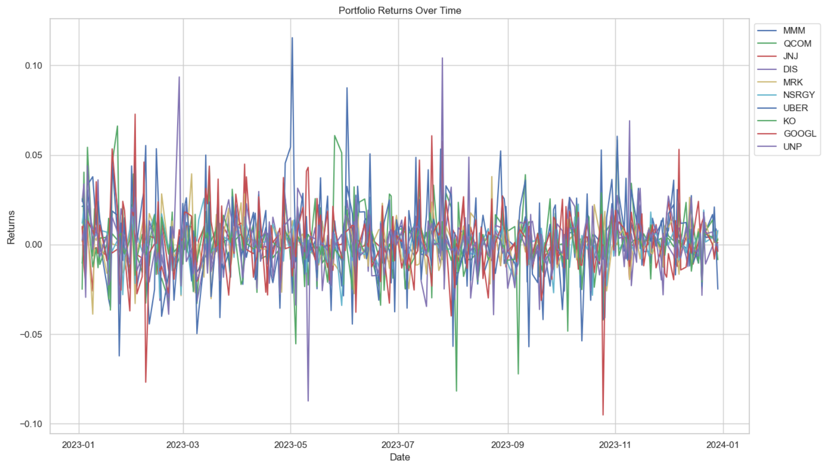

Figura 6.2: Serie storica dei rendimenti degli asset in portafoglio

6.1.2 Composizione e ottimizzazione del portafoglio

A questo punto, al fine di ottimizzare il portafoglio, è possibile procedere determinando e tracciando la frontiera efficiente, composta dagli ottimi paretiani, basati sulla media dei rendimenti e la volatilità degli stessi (varianza).

Si procede calcolando i rendimenti per ciascun asset, presente nella tabella 6.1 delle statistiche di rendimento medio a livello di singolo portafoglio $\mu_i = E(r_i)$, e delle relative volatilità σ_i. Inoltre, devono essere definiti anche i pesi associati al portafoglio. Conseguentemente, il tasso di rendimento del portafoglio sarà dato da:

$$r = \sum_{i=1}^{n} r_i w_i \tag{6.1}$$

e la matrice di varianza covarianza sarà definita come:

$$\Sigma = \omega^T \cdot \sum_{i}^{n} \cdot \omega \tag{6.2}$$

Il processo di ottimizzazione, si pone l'obiettivo di minimizzare la volatilità di ciascun portafoglio per ogni livello di rischio. Analiticamente si traduce nella seguente funzione quadratica:

$$\min \frac{1}{2}\omega^T \Sigma \omega \tag{6.3}$$

$$\sum_{i=1}^{n} \omega_i^T \cdot r_i = \mu \tag{6.4}$$

$$sub \sum_{i=1}^{n} \omega_i = 1 \tag{6.5}$$

Tale problema di ottimizzazione può essere risolto attraverso l'utilizzo della funzione Lagrangiana:

$$L = \frac{1}{2}\omega^T \cdot \Sigma \cdot \omega - \Lambda_1 \cdot (\omega^T r_i - \mu) - \Lambda_2 \cdot (\omega^T \cdot \mu - 1) \tag{6.6}$$

Da cui è possibile ricavare le condizioni del primo ordine:

$$\begin{cases} \frac{\delta L}{\delta \omega} = \Sigma \cdot \omega - \Lambda_1 \cdot r_i - \Lambda_2 \cdot \mu = 0 \\ \frac{\delta L}{\delta \Lambda} = \mu - \omega^T \cdot r_i = 0 \\ \frac{\delta L}{\delta \mu} = 1 - \omega^T \cdot \mu = 0 \end{cases}$$

Tali condizioni, se risolte attraverso diversi metodi matematici, tra cui il metodo di Cramer, portano alla costruzione della frontiera, ovvero a quell'insieme di portafogli, che per ciascun livello di rendimento minimizzano la volatilità del portafoglio.

6.1.3 Analisi dei risultati

Questo processo di ottimizzazione, per quanto sconti una serie di assunzioni importanti, porta alla costruzione della frontiera efficiente, stabilendo la composizione ottimale del portafoglio, dato un numero n di strumenti finanziari.

In generale, è bene considerare che tutti gli asset al di sopra della stessa possono essere considerati ottimali. Conseguentemente, la scelta di dove collocarsi deriva dalle preferenze dell'investitore, che sono per definizione soggettive poiché dipendono dalla funzione di utilità individuale. In questa ricerca, ci collocheremo in modo tale da generare la composizione di portafoglio per cui lo Sharpe ratio possa essere massimizzato, come mostrato in figura 6.3.

In questo caso il portafoglio che massimizza lo Sharpe ratio (evidenziato in rosso in figura 6.3 ha i seguenti pesi per ciascun titolo:

Ticker	MMM	QCOM	JNJ	DIS	MRK	NSRGY	UBER	KO	GOOGL	UNP
Value	0.08	0.09	0.11	0.10	0.10	0.11	0.09	0.13	0.10	0.09

Tabella 6.2: Pesi del portafoglio ottimizzato

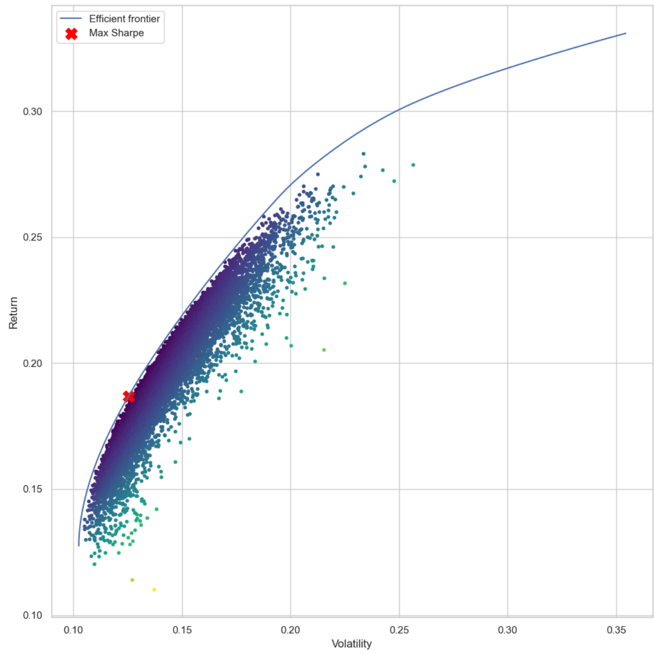

Figura 6.3: Frontiera efficiente per i 10 asset con minore correlazione

Il portafoglio così composto ha uno Sharpe Ratio, calcolato avendo a riferimento il tasso di rendimento di un Treasury 1Y annualizzato e pari al 5.25%, uguale a 1.49, collocandosi come buon investimento dal punto di vista rischio-rendimento, con un rendimento annuo di circa il 18.7% e una volatilità annualizzata pari al 12.5%.

Per quanto riguarda il VaR, ad un livello di confidenza stimato su un intervallo di confidenza del 95%, il VaR è pari a 1.4 %.

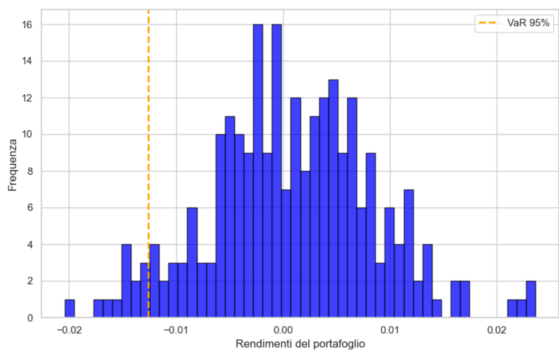

Figura 6.4: Rendimenti del portafoglio ottimizzati con il modello di Markowitz

6.1.4 Verifica dell'impatto del Pair Trading

A questo punto, non ci resta che rispondere alla domanda iniziale della nostra ricerca: verificare l'efficacia della strategia di Pair Trading nell'ambito di un portafoglio di investimenti.

Per fornire una risposta, è stata aggiunta al Dataset della tabella 6.1 la serie storica relativa al valore del portafoglio stimato nel capitolo precedente, da cui è possibile eseguire la medesima procedura al fine di ottenere ancora una volta un portafoglio ottimizzato, con un vettore di pesi più lungo per l'aggiunta di un asset.

Di conseguenza il vettore dei pesi sarà quello rappresentato dalla tabella che segue 6.3:

Ticker	MMM	QCOM	JNJ	DIS	MRK	NSRGY	UBER	KO	GOOGL	UNP	PAIRS
Value	0.11	0.10	0.07	0.10	0.07	0.07	0.11	0.07	0.09	0.10	0.11

Tabella 6.3: Pesi del portafoglio ottimizzato con strategia di Pair Trading

Tali pesi, e in particolare quello relativo ai Pairs, sono in linea con le aspettative, considerando le contribuzioni ai rendimenti visibili nel grafico che segue 6.5

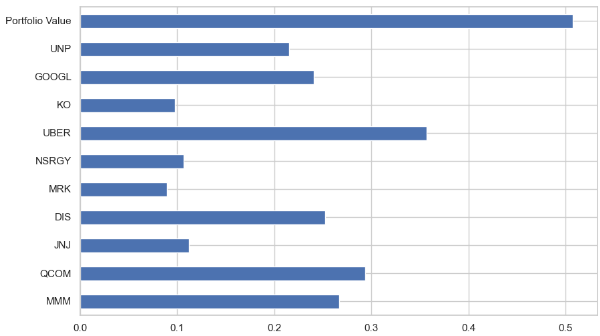

Figura 6.5: Contribuzioni medie al rendimento per singolo asset in portafoglio

Al fine di comparare i risultati con quelli ottenuti rispetto al precedente portafoglio ottimizzato, anche in questa sede è stato preso in considerazione il portafoglio che massimizza lo Sharpe ratio, la cui posizione rispetto alla frontiera efficiente è rappresentata nella figura 6.6. Ciò che cattura immediatamente l'attenzione, è il rendimento annuo del portafoglio pari al 25 % con una volatilità annualizzata registrata pari a 13.1%. Inoltre è possibile notare, che il peso del portafoglio creato attraverso la strategia di Pair Trading, ed inserito nel nuovo processo di ottimizzazione, assume il valore più alto rispetto agli altri titoli già precedentemente presenti.

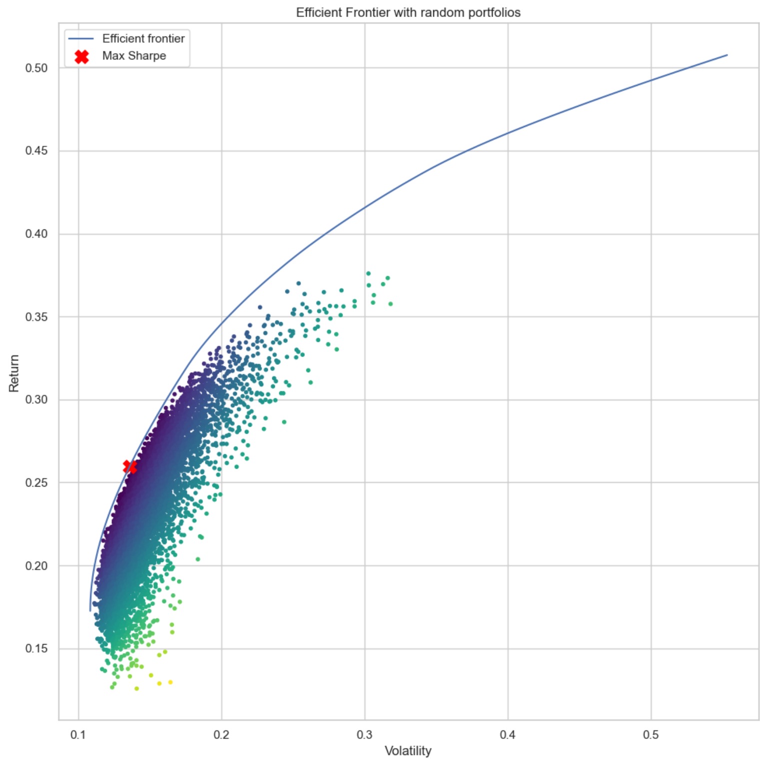

Figura 6.6: Frontiera efficiente per il portafoglio con integrazione della strategia di Pair Trading

Il questo caso,il livello di Sharpe ratio registrato è pari a 1.91, indicando che il rendimento aggiustato per il rischio di un investimento è significativamente superiore al rendimento di un investimento privo di rischio, in rapporto al rischio assunto.

Infine, come si può facilmente notare nella figura 6.7, i rendimenti di questo portafoglio sono molto più concentrati attorno alla media, e il Value at Risk, calcolato seguendo il metodo storico e su un intervallo di confidenza del 5% ,è pari a 1.2%, dimostrando un calo della rischiosità generale dell'investimento.

6.2 Test del portafoglio

L'ultimo step dell'analisi compiuto, al fine di verificare l'esistenza dell'effetto del Pair Trading rispetto alla diversificazione di portafoglio, è il backtesting.

Nello specifico, vengono presi in considerazione i pesi del portafoglio calcolati nella precedente ottimizzazione (visibili in tabella 6.3), che si riferiscono al periodo compreso tra 30.12.2022 e 30.12.2023; successivamente questi vengono applicati ai rendimenti di

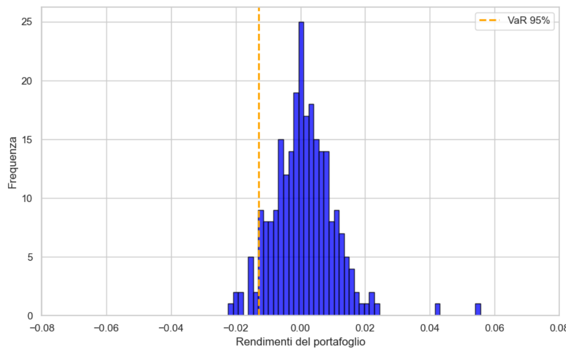

Figura 6.7: Distribuzione dei rendimenti del portafoglio ottimizzato con strategia di Pair Trading integrata

ciascun asset del nuovo Dataset, compresi quelli derivanti dalla strategia di Pair Trading su un intervallo temporale successivo ossia quello che va dal 01.01.2024 al 30.04.2024.

Mantenendo per semplicità un ammontare di denaro iniziale pari a 100 $, il risultato è un investimento che è in grado di produrre:

- Rendimento annuo del portafoglio: 27.9%
- Volatilità annualizzata del portafoglio: 11.1%
- Sharpe Ratio: 2.04.

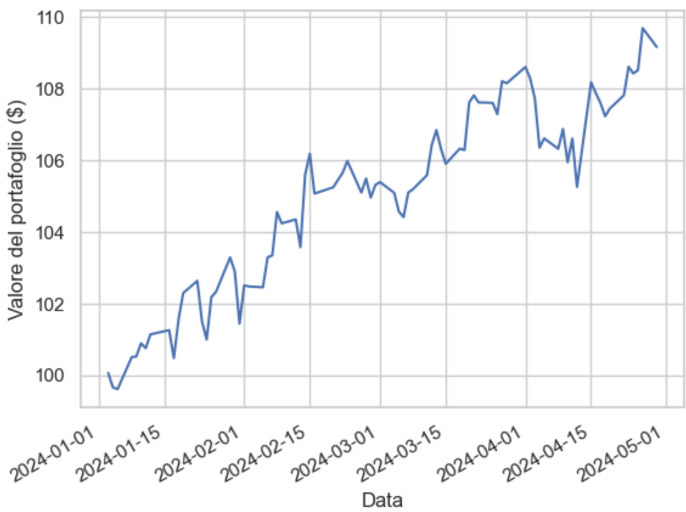

Figura 6.8: Valore del portafoglio ottimizzato in backtesting

Le considerazioni rispetto al confronto tra il portafoglio che integra la strategia dei Pairs, e quello che invece non la utilizza al suo interno, sono varie. In primo luogo, la frontiera efficiente per il primo è spostata più in alto. Ciò si traduce, in un grafico rischio rendimento, in un aumento della redditività media del portafoglio. Di conseguenza, il portafoglio che massimizza lo Sharpe Ratio, ovvero quello scelto come portafoglio ottimo sulla base di un'approssimazione delle preferenze di un investitore medio, si attesta ad un valore più per quanto riguarda i rendimenti complessivi, e ad un livello di volatilità inferiore. Questo si traduce in un migliore rapporto rischio-rendimento rispetto al portafoglio senza la coppia di Pairs, elemento che emerge dall'interpretazione dello Sharpe Ratio, che supera il portafoglio senza la strategia di arbitraggio collocandosi a 2.04.

L'effetto congiunto di tutti questi elementi permettono di raggiungere un rendimento annuo pari al 27.9% nell'intervallo temporale considerato.

La volatilità di portafoglio 11.09% suggerisce comunque un livello complessivamente abbastanza basso di variabilità che si tradurrebbe in un rischio associato al portafoglio, ed allo stesso tempo, il Sortino ratio di 2.02 indica che il portafoglio è stato in grado di generare rendimenti positivi rispetto al rischio specifico, concentrando l'analisi sulle perdite al di sotto di un certo livello di rendimento desiderato, e confermando quanto evidenziato con l'indicazione dello Sharpe ratio. Infine il Value at Risk (VaR) calcolato con un intervallo di confidenza del 5% è pari al 1.05% del valore complessivo del portafoglio, mentre prendendo a riferimento un intervallo di confidenza dell'1% il VaR è pari all'1,3%, confermando ancora una volta che la strategia adottata è efficace nel limitare la rischiosità complessiva del portafoglio.

Statistiche	**Valore**
Sharpe Ratio	2.04
Rendimento annuo	27.9%
Sortino Ratio	2.02
VaR (5%)	1.05%
VaR (1%)	1.3%

Tabella 6.4: Statistiche del Portafoglio con strategia Pairs

Conclusioni

L'elemento centrale di questa ricerca è coinciso con l'individuazione di una strategia di trading che fosse redditizia, mantenendo allo stesso tempo un basso profilo di rischio. Nello specifico, lo studio verte sull'applicazione di una strategia specifica long-short costituita dal Pair Trading, metodo che possiede la capacità di sfruttare i disallineamenti di prezzo tra titoli altamente correlati.

I risultati hanno mostrato diverse evidenze. Attraverso l'analisi statistica delle serie storiche dei prezzi, sono state identificate le coppie di titoli, le cui relazioni di prezzo mostrano una tendenza nel lungo periodo a muoversi insieme, nonostante le fluttuazioni a breve termine. Ciò ha permesso di selezionare coppie con una relazione stabile, avvalendoci dell'analisi della cointegrazione, con una riduzione significativa del rischio di divergenza, e la possibilità che i prezzi delle coppie si discostino permanentemente l'uno dall'altro.

Questo ha permesso di selezionare una coppia di azioni identificate in Accenture e Google, società altamente capitalizzate, caratterizzate da volumi molto alti per quanto riguarda le quantità scambiate sui mercati (e nello specifico nell'S&P 500). Ciò consentirebbe di sviluppare una strategia ad una frequenza maggiore rispetto a quella selezionata per la ricerca. Si è optato per una frequenza giornaliera, in modo tale da poter ricreare un investimento più stabile nel tempo in termini di volatilità.

La strategia del Pair Trading ha mostrato un rendimento medio annualizzato superiore al benchmark di mercato (US Treasury 1Y) pari a 5.25% annuo, suggerendo che l'approccio possa offrire un vantaggio competitivo rispetto a strategie di investimento più tradizionali, con la contestuale possibilità di mantenere un basso profilo di rischio. I dati rilevati, hanno evidenziato che i rendimenti ottenuti attraverso la strategia sono consistenti, e che la stessa è in grado di generare profitti anche in condizioni di mercato molto variabili.

Infine, l'analisi si è concentrata sull'impatto dell'introduzione della strategia di Pair Trading nella diversificazione del portafoglio, con un aumento del rendimento annuo, ed un ulteriore miglioramento dello Sharpe ratio, contribuendo ad una maggiore stabilità dei rendimenti, come evidenziato dalla concentrazione dei rendimenti intorno alla media e dal ridotto livello di Value at Risk (VaR) oltre che dalla volatiltià complessiva.

Bibliografia

[1] Marco Avellaneda and Jeong-Hyun Lee. Statistical arbitrage in the us equities market. *Quantitative Finance*, 10(7):761–782, 2010.

[2] João Caldeira and Guilherme V Moura. Selection of a portfolio of pairs based on cointegration: A statistical arbitrage strategy. *Available at SSRN 2196391*, 2013.

[3] Zhiwu Chen and Peter J Knez. Measurement of market integration and arbitrage. *The Review of Financial Studies*, 8(2):287–325, 1995.

[4] DEGIRO. Degiro tariffario. *Disponibile online: `https://www.degiro.it/data/pdf/it/NIB_Tariffario.pdf` (ultimo accesso: inserire la data di accesso)*, 2024.

[5] David A Dickey and Wayne A Fuller. Distribution of the estimators for autoregressive time series with a unit root. *Journal of the American statistical association*, 74(366a):427–431, 1979.

[6] Binh Do and Robert Faff. Does simple pairs trading still work? *Financial Analysts Journal*, 66(4):83–95, 2010.

[7] Binh Do, Robert Faff, and Kais Hamza. A new approach to modeling and estimation for pairs trading. In *Proceedings of 2006 financial management association European conference*, volume 1, pages 87–99. Citeseer, 2006.

[8] Robert J Elliott, John Van Der Hoek*, and William P Malcolm. Pairs trading. *Quantitative Finance*, 5(3):271–276, 2005.

[9] Robert F Engle and Clive WJ Granger. Co-integration and error correction: representation, estimation, and testing. *Econometrica: journal of the Econometric Society*, pages 251–276, 1987.

[10] Evan Gatev, William N Goetzmann, and K Geert Rouwenhorst. Pairs trading: Performance of a relative-value arbitrage rule. *The Review of Financial Studies*, 19(3):797–827, 2006.

[11] Nicolas Huck. Pairs selection and outranking: An application to the s&p 100 index. *European Journal of Operational Research*, 196(2):819–825, 2009.

[12] Nicolas Huck. Pairs trading and outranking: The multi-step-ahead forecasting case. *European Journal of Operational Research*, 207(3):1702–1716, 2010.

[13] Nicolas Huck and Komivi Afawubo. Pairs trading and selection methods: is cointegration superior? *Applied Economics*, 47(6):599–613, 2015.

[14] John Hull. *Risk management and financial institutions,+ Web Site*, volume 733. John Wiley & Sons, 2012.

[15] Michael C. Jensen. The performance of mutual funds in the period 1945–1964. *The Journal of Finance*, 23(2):389–416, 1968.

[16] Søren Johansen. Statistical analysis of cointegration vectors. *Journal of economic dynamics and control*, 12(2-3):231–254, 1988.

[17] Ian T Jolliffe. *Principal component analysis for special types of data*. Springer, 2002.

[18] Jakub W Jurek and Halla Yang. Dynamic portfolio selection in arbitrage. In *EFA 2006 Meetings Paper*, 2007.

[19] Gebhard Kirchgässner, Jürgen Wolters, and Uwe Hassler. *Introduction to modern time series analysis*. Springer Science & Business Media, 2012.

[20] Christopher Krauss. Statistical arbitrage pairs trading strategies: Review and outlook. *Journal of Economic Surveys*, 31(2):513–545, 2017.

[21] Prashanth La and Mohammad Ghavamzadeh. Actor-critic algorithms for risk-sensitive mdps. *Advances in neural information processing systems*, 26, 2013.

[22] Benoit Mandelbrot. New methods in statistical economics. *Journal of political economy*, 71(5):421–440, 1963.

[23] Harry M Markowits. Portfolio selection. *Journal of finance*, 7(1):71–91, 1952.

[24] John J Murphy. *Technical analysis of the financial markets: A comprehensive guide to trading methods and applications*. Penguin, 1999.

[25] William F Sharpe. Capital asset prices: A theory of market equilibrium under conditions of risk. *The journal of finance*, 19(3):425–442, 1964.

[26] James H Stock and Mark W Watson. Testing for common trends. *Journal of the American statistical Association*, 83(404):1097–1107, 1988.

[27] Ganapathy Vidyamurthy. *Pairs Trading: quantitative methods and analysis*, volume 217. John Wiley & Sons, 2004.

Appendice

```python
# -*- coding: utf-8 -*-
"""TESI _FINALE-Copy1.ipynb

Automatically generated by Colab.

Original file is located at
    https://colab.research.google.com/drive/1XtlfIo8rX6wpY48JjYBt0ipSA4GXY7M1

# Strategia di Pair Trading - Tesi Nicola Gherardi
# Libera Universit  degli Studi Sociali - LUISS Guido Carli
"""

# Commented out IPython magic to ensure Python compatibility.
# Importazione delle librerie

import numpy as np
import pandas as pd
import statsmodels
from scipy import stats
import statsmodels.api as sm
from statsmodels.tsa.stattools import coint, adfuller
import matplotlib.pyplot as plt
import seaborn as sns; sns.set(style="whitegrid")
import matplotlib.pyplot as plt
import plotly.express as px
import seaborn as sns
from scipy.stats import norm, t
from pandas_datareader import data as pdr
import datetime
import yfinance as yf
yf.pdr_override()
from sklearn import datasets # Per importare i dataset
from sklearn.preprocessing import StandardScaler # Per trasformazione dei dataset
from sklearn.cluster import DBSCAN # Per clustering
from sklearn.metrics import pairwise_distances # Per valutazione del modello
from sklearn.neighbors import NearestNeighbors
# %matplotlib inline
import numpy as np
import matplotlib.pyplot as plt
import cvxopt as opt
from cvxopt import blas, solvers
import pandas as pd
from tqdm import tqdm

import plotly
import plotly.graph_objects as go
```

```python
from plotly.subplots import make_subplots
import plotly.express as px
import plotly.figure_factory as ff


# Importa Statsmodels
from statsmodels.tsa.api import VAR
from statsmodels.tsa.stattools import adfuller
from statsmodels.tools.eval_measures import rmse, aic

# Scelta degli Asset iniziali: per modificare gli asset sufficiente aggiungere il rispettivo
# ticker cos  come presente in Yahoo Finance

# Definizione dei tickers per ciascuna categoria
tickers = []
tickers_tech = ['MSFT','AVGO','CSCO','NOW','VZ','TXN','CMCSA','INTU', 'AAPL', 'NVDA', 'TSLA', 'GOOGL', 'META', 'AMZN', 'NFLX', 'IBM','ADBE','AMD','INTC','ORCL','QCOM','ADSK','MSI','T','ACN','TMUS', 'VZ' ]
tickers_banks = ['BRK-B', 'JPM','M','V', 'BAC', 'WFC', 'AXP', 'GS', 'MS', 'C', 'AXP','BX', 'SPGI']
tickers_company = ['MCD','UNP','COST','PM', 'PEP', 'KO','WMT','NSRGY','CAT', 'NKE', 'ADDYY', 'SBUX','MDLZ','CMG','HD','BA','UBER','DIS', 'CAT', ]
tickers_pharma = ['LLY', 'UNH','TMO', 'JNJ','MRK', 'PG', 'ABBV', 'ABT', 'PFE', 'ELV', 'CL', 'MMM','MRNA' ,'NVO']
tickers_petrol = ['XOM','CVX', 'SUN', 'MPC','LIN']

# Scelta intervallo temporale
start = datetime.datetime(2019, 4, 30)
end = datetime.datetime(2024, 4, 30)

# Funzione per ottenere i dati per un ticker specificato
def get_data(ticker):
    data = yf.download(ticker, start, end)
    return data

# Creazione dei dataframe per ciascuna categoria di ticker
dataframes_tech = pd.DataFrame({ticker: get_data(ticker)['Close'] for ticker in tickers_tech})
dataframes_banks = pd.DataFrame({ticker: get_data(ticker)['Close'] for ticker in tickers_banks})
dataframes_company = pd.DataFrame({ticker: get_data(ticker)['Close'] for ticker in tickers_company})
dataframes_pharma = pd.DataFrame({ticker: get_data(ticker)['Close'] for ticker in tickers_pharma})
dataframes_petrol = pd.DataFrame({ticker: get_data(ticker)['Close'] for ticker in tickers_petrol})
```

```

# Creazione di un dataset unico
alltickers = pd.merge(dataframes_banks, dataframes_tech, left_index=True, right_index=True, how='outer')
alltickers = pd.merge(alltickers, dataframes_company, left_index=True, right_index=True, how='outer')
alltickers = pd.merge(alltickers, dataframes_pharma, left_index=True, right_index=True, how='outer')
alltickers.head()
alltickers = pd.DataFrame(alltickers)

# Calcolo della matrie di correlazione per tutti gli asset
correlation_matrix = alltickers.corr()

# Trasforma la matrice di correlazione in una Serie
correlation_series = correlation_matrix.stack()

# Esempio Matrice delle correlazioni (asset tech)
plt.figure(figsize = (15,6))
sns.heatmap(dataframes_tech.corr())
plt.savefig("/Users/nicolagherardi/Desktop/correlogramma_tech")
plt.show()

# Screening parziale per criterio di correlazione

# Prendi il valore assoluto della correlazione
correlation_series_abs = correlation_series.abs()

# Ordina la Serie di correlazione in ordine decrescente
sorted_correlation = correlation_series_abs.sort_values(ascending=False)

# Seleziona le prime 9 coppie con la correlazione pi  alta (non perfetta)
top_9_corr_pairs = sorted_correlation[sorted_correlation < 1].drop_duplicates().head(9)

print("Prime 9 coppie con correlazione pi  alta (non perfetta):")
print(top_9_corr_pairs)

# Lista delle colonne per le coppie con correlazione pi  alta
column_pairs = top_9_corr_pairs.index.tolist()

# Crea il numero corretto di subplot
n_rows = 3
n_cols = 3
fig, axes = plt.subplots(n_rows, n_cols, figsize=(16, 12))

# Itera attraverso le coppie di colonne e disegna i grafici di
```

```
    dispersione
for i, (x_col, y_col) in enumerate(column_pairs):
    row = i // n_cols
    col = i % n_cols
    ax = axes[row, col]

    sns.scatterplot(x=x_col, y=y_col, data=alltickers, ax=ax, color= 'b')
    ax.set_title(f"Corr: {correlation_matrix.loc[x_col, y_col]:.2f}", fontsize=14)

# Rimuovi gli assi vuoti se ce ne sono
for i in range(len(column_pairs), n_rows * n_cols):
    row = i // n_cols
    col = i % n_cols
    fig.delaxes(axes[row, col])

plt.tight_layout()
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/correlazioni_pairs")
plt.show()

# Estrai i ticker come singoli
tickers = sorted(set(top_9_corr_pairs.index.get_level_values(0)).union(top_9_corr_pairs.index.get_level_values(1)))

# Stampa i ticker come singoli
ticker_list = [ticker for ticker in tickers]

print("Lista dei ticker:")
print(ticker_list)

# Ottieni i dati per le prime 9 coppie di asset

def get_data(ticker):
    data = yf.download(ticker, start, end)
    return data

# Creazione dei dataframe per ciascuna categoria di ticker

dataframe = pd.DataFrame({ticker: get_data(ticker)['Close'] for ticker in ticker_list}, index=get_data(tickers[0]).index)
dataframe.tail()

#  Creazione   del correlogramma per le prime 9 coppie

plt.figure(figsize = (15,6))
sns.heatmap(dataframe.corr(), annot=True)
plt.show()
```

```

# Creazione grafico con i prezzi dei 9 asset

dataframe.plot()
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/
    Prices_notnormalized")

# Analisi della Cointegrazione
def find_cointegrated_pairs(data):
    n = data.shape[1]
    score_matrix = np.zeros((n, n))
    pvalue_matrix = np.ones((n, n))
    keys = data.keys()
    pairs = []
    for i in range(n):
        for j in range(i+1, n):
            S1 = data[keys[i]]
            S2 = data[keys[j]]
            result = coint(S1, S2)
            score = result[0]
            pvalue = result[1]
            score_matrix[i, j] = score
            pvalue_matrix[i, j] = pvalue
            if pvalue < 0.05:
                pairs.append((keys[i], keys[j]))
    return score_matrix, pvalue_matrix, pairs

# Esempio di utilizzo
# Supponiamo che 'df' sia il DataFrame contenente i prezzi di
    chiusura degli asset
scores, pvalues, pairs = find_cointegrated_pairs(dataframe)

# Funzione per ottenere i dati storici di un ticker
def get_historical_data(ticker):
    data = yf.download(ticker, start, end)['Close']
    return data

# Funzione per calcolare i prezzi normalizzati
def normalize_prices(pair):
    # Ottieni i dati storici per entrambi i ticker
    ticker1_data = get_historical_data(pair[0])
    ticker2_data = get_historical_data(pair[1])

    # Calcola la media mobile esponenziale a 20 periodi per
    entrambi i ticker
    mean_ticker1 = ticker1_data.mean()
    stdticker1 = ticker1_data.std()
    mean_ticker2 = ticker2_data.mean()
    stdticker2 = ticker2_data.std()
```

```python

    # Normalizza i prezzi utilizzando la differenza tra il prezzo e la EMA
    normalized_ticker1 = (ticker1_data - mean_ticker1)/ticker1_data.std()
    normalized_ticker2 = (ticker2_data - mean_ticker2)/ticker2_data.std()

    return normalized_ticker1, normalized_ticker2

# Plot dei prezzi normalizzati per ciascuna coppia di asset cointegrati
plt.figure(figsize=(12, 8))

for pair in pairs:
    normalized_ticker1, normalized_ticker2 = normalize_prices(pair)
    plt.plot(normalized_ticker1, label=pair[0])
    plt.plot(normalized_ticker2, label=pair[1])

plt.title('Prezzi normalizzati per coppie di asset cointegrati')
plt.xlabel('Data')
plt.ylabel('Prezzo normalizzato')
plt.legend()
plt.grid(True)
plt.show()

# Creazione del Box Plot per il dataframe

plt.figure(figsize = (15,4))
sns.boxplot(data = dataframe, orient = "h")
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/boxplot_notscaled")
plt.show()

# Calcolo dei log-rendimenti
log_ret = np.log(dataframe).diff().dropna()
print(type(log_ret))
log_ret.head()

# Numero di righe e colonne per i grafici
n_rows = 4
n_cols = 4

# Calcola il numero totale di grafici
n_plots = len(log_ret.columns)

# Crea una nuova figura
fig, axs = plt.subplots(n_rows, n_cols, figsize=(15, 10))
```

```

# Itera sui sottogruppi dei titoli e dei relativi log-rendimenti
for i, (colonna, serie) in enumerate(log_ret.iteritems()):
    # Calcola l'indice di riga e colonna per il grafico attuale
    row = i // n_cols
    col = i % n_cols

    # Disegna il grafico per il titolo corrente
    axs[row, col].plot(serie, color='blue')
    axs[row, col].set_title(f'Log-Rendimenti per {colonna}')
    axs[row, col].set_xlabel('Data')
    axs[row, col].set_ylabel('Log-Rendimento ')
    axs[row, col].grid(True)

# Rimuovi sottografici vuoti
for i in range(n_plots, n_rows * n_cols):
    fig.delaxes(axs.flatten()[i])


# Aggiusta lo spaziatura tra i grafici
plt.tight_layout()

plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/
    logreturns")

# Visualizza i grafici
plt.show()

# Creazione di un dataset Scalato: standardizzazione della serie
    di valori

scaler = StandardScaler()
scaled_array = scaler.fit_transform(dataframe)
scaled_dataframe = pd.DataFrame(scaled_array, columns=dataframe.
    columns, index=dataframe.index)


# Gestisci i dati mancanti

scaler = StandardScaler()
scaled_array = scaler.fit_transform(dataframe)
scaled_dataframe = pd.DataFrame(scaled_array, columns=dataframe.
    columns, index=dataframe.index)
scaled_dataframe = scaled_dataframe.replace(0, method='ffill')
scaled_dataframe = scaled_dataframe.replace(0, method='bfill')

# Converti l'array NumPy in DataFrame

scaled_dataframe = pd.DataFrame(scaled_dataframe, columns=
```

```
    scaled_dataframe.columns)
scaled_dataframe.head(80)

# Creazione del Box Plot per il dataframe scalato

plt.figure(figsize = (15,4))
sns.boxplot(data = scaled_dataframe, orient = "h")
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/boxplot_scaled")
plt.show()

#Definizione della funzione per Augmented Dickey Fuller test

def adfuller_test(series, signif=0.05, name='', verbose=False):
    """Perform ADFuller to test for Stationarity of given series and print report"""
    r = adfuller(series, autolag='AIC')
    output = {'test_statistic':round(r[0], 4), 'pvalue':round(r[1], 4), 'n_lags':round(r[2], 4), 'n_obs':r[3]}
    p_value = output['pvalue']
    def adjust(val, length= 6): return str(val).ljust(length)

    # Print Summary
    print(f'    Augmented Dickey-Fuller Test on "{name}"', "\n   ", '-'*47)
    print(f' Null Hypothesis: Data has unit root. Non-Stationary.')
    print(f' Significance Level    = {signif}')
    print(f' Test Statistic        = {output["test_statistic"]}')
    print(f' No. Lags Chosen       = {output["n_lags"]}')

    for key,val in r[4].items():
        print(f' Critical value {adjust(key)} = {round(val, 3)}')

    if p_value <= signif:
        print(f" => P-Value = {p_value}. Rejecting Null Hypothesis.")
        print(f" => Series is Stationary.")
    else:
        print(f" => P-Value = {p_value}. Weak evidence to reject the Null Hypothesis.")
        print(f" => Series is Non-Stationary.")

# ADF Test on each column
for name, column in log_ret.iteritems():
    adfuller_test(column, name=column.name)
    print('\n')

#COINTEGRAZIONE
```

```python
def find_cointegrated_pairs(data):
    n = data.shape[1]
    score_matrix = np.zeros((n, n))
    pvalue_matrix = np.ones((n, n))
    keys = data.keys()
    pairs = []
    for i in range(n):
        for j in range(i+1, n):
            S1 = data[keys[i]]
            S2 = data[keys[j]]
            result = coint(S1, S2)
            score = result[0]
            pvalue = result[1]
            score_matrix[i, j] = score
            pvalue_matrix[i, j] = pvalue
            if pvalue < 0.05:
                pairs.append((keys[i], keys[j]))
    return score_matrix, pvalue_matrix, pairs

# Esempio di utilizzo
# Supponiamo che 'df' sia il DataFrame contenente i prezzi di
    chiusura degli asset
scores, pvalues, pairs = find_cointegrated_pairs(dataframe)

# Creazione della heatmap con annotazioni dei valori
fig, ax = plt.subplots(figsize=(10, 8))
sns.heatmap(pvalues, xticklabels=tickers, yticklabels=tickers,
    cmap='RdYlGn_r', mask=(pvalues >= 0.05), square=True, annot=
    True, fmt=".2f", ax=ax)

# Impostazione del titolo e dei label degli assi
plt.title('P-values del Test di Cointegrazione')
plt.xlabel('Asset')
plt.ylabel('Asset')
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/
    cointegration_heatmap")
plt.show()

# Stampa delle coppie cointegrate
print("Coppie cointegrate:")
print(pairs)

print(type(pairs))

# Definizione della lista dei ticker
ticker_list = []
for pair in pairs:
    for ticker in pair:
        ticker_list.append(f'{ticker}')
```

```
print(ticker_list)

# Ottengo il DataFrame ristretto
def get_data(ticker):
    data = yf.download(ticker, start, end)['Close']
    return data

# Creazione dei dataframe per ciascuna categoria di ticker
dataframe_dict = {}
for pair in pairs:
    for ticker in pair:
        dataframe_dict[ticker] = get_data(ticker)

dataframe = pd.DataFrame(dataframe_dict, index=get_data(pairs
    [0][0]).index)
dataframe.tail()

# ADF Test

adf_results = []

# Eseguire il test ADF per ciascun titolo
for colonna in dataframe.columns:
    result = adfuller(dataframe[colonna])
    adf_results.append({
        'Titolo': colonna,
        'Statistiche ADF': result[0].round(2),
        'Valore p': result[1].round(2),
        'Valori critici (1%)': result[4]['1%'].round(2),
        'Valori critici (5%)': result[4]['5%'].round(2),
        'Valori critici (10%)': result[4]['10%'].round(2)
    })

# Creare un DataFrame per i risultati del test ADF
adf_results_df = pd.DataFrame(adf_results)

# Stampare il DataFrame dei risultati del test ADF
print(adf_results_df.transpose())
adf_results_t = adf_results_df.transpose()

# Definizione della funzione per ADF TEST
def rifiuto_ipotesi(valore_p, alpha=0.05):
    if valore_p < alpha:
        return 'S  (stazionaria)'
    else:
        return 'No (non stazionaria)'

# Calcolo dei log-rendimenti
log_ret = np.log(dataframe).diff().dropna()
```

```
print(type(log_ret))
log_ret.head()

# Grafici dei log rendimenti

# Numero di righe e colonne per i grafici
n_rows = 2
n_cols = 3

# Calcola il numero totale di grafici
n_plots = len(log_ret.columns)

# Crea una nuova figura
fig, axs = plt.subplots(n_rows, n_cols, figsize=(15, 10))

# Itera sui sottogruppi dei titoli e dei relativi log-rendimenti
for i, (colonna, serie) in enumerate(log_ret.iteritems()):
    # Calcola l'indice di riga e colonna per il grafico attuale
    row = i // n_cols
    col = i % n_cols

    # Disegna il grafico per il titolo corrente
    axs[row, col].plot(serie, color='blue')
    axs[row, col].set_title(f'Log-Rendimenti per {colonna}')
    axs[row, col].set_xlabel('Data')
    axs[row, col].set_ylabel('Log-Rendimento ')
    axs[row, col].grid(True)

# Aggiusta lo spaziatura tra i grafici
plt.tight_layout()

# Visualizza i grafici
plt.show()

# Definiamo le coppie di serie temporali

# Definiamo gli intervalli temporali di formazione
Training = log_ret[:(round(len(log_ret)*0.70))] #70% della serie
    temporale


# Funzione per calcolare la regressione OLS e il coefficiente di
    cointegrazione
def calcola_cointegrazione(formazione, coppia):
    # Estraiamo le serie temporali dalla formazione
    Y = formazione[coppia[0]]
    X = formazione[coppia[1]]

    # Aggiungi una costante alla variabile indipendente (X)
```

```
    X = sm.add_constant(X)

    # Stima il modello di regressione lineare
    model = sm.OLS(Y, X).fit()

    # Ottieni il coefficiente di cointegrazione
    beta = model.params[1]  # Il secondo parametro    il coefficiente di cointegrazione

    return beta

# Calcoliamo il coefficiente di cointegrazione per ciascuna coppia e ciascun intervallo di formazione
for coppia in pairs:
    print("Coppia:", coppia)
    print("Formazione al 70% della Serie:")
    beta = calcola_cointegrazione(Training, coppia)
    print("Coefficienti di cointegrazione:", beta)

#Coefficielti Beta di regressione

risultati = []
for coppia in pairs:
    beta = calcola_cointegrazione(Training, coppia)

    risultati.append([coppia[0], coppia[1], beta])

# Creiamo un DataFrame per i risultati
df_risultati = pd.DataFrame(risultati, columns=['Serie 1', 'Serie 2', 'gamma'])

# Stampiamo il DataFrame dei risultati
print(df_risultati)

# Funzione per Engle e Granger test

from statsmodels.tsa.stattools import grangercausalitytests
maxlag=12
df = dataframe
test = 'ssr_chi2test'
def grangers_causation_matrix(data, variables, test='ssr_chi2test', verbose=False):
    """Check Granger Causality of all possible combinations of the Time series.
    The rows are the response variable, columns are predictors. The values in the table
    are the P-Values. P-Values lesser than the significance level (0.05), implies
    the Null Hypothesis that the coefficients of the corresponding
```

```
    past values is
    zero, that is, the X does not cause Y can be rejected.

    data      : pandas dataframe containing the time series variables
    variables : list containing names of the time series variables.
    """
    df = pd.DataFrame(np.zeros((len(variables), len(variables))), columns=variables, index=variables)
    for c in df.columns:
        for r in df.index:
            test_result = grangercausalitytests(data[[r, c]], maxlag=maxlag, verbose=False)
            p_values = [round(test_result[i+1][0][test][1],4) for i in range(maxlag)]
            if verbose: print(f'Y = {r}, X = {c}, P Values = {p_values}')
            min_p_value = np.min(p_values)
            df.loc[r, c] = min_p_value
    df.columns = [var + '_x' for var in variables]
    df.index = [var + '_y' for var in variables]
    return df
grangers_causation_matrix(df, variables = df.columns).round(2)

# Scelgo gli asset che presentano un pi  alto livello di cointegrazione:
# ACN e GOOGL
# Si definisce il rapporto tra i prezzi come ratio:

ratios = dataframe['GOOGL'] / dataframe['ACN']

# Definizione del periodo di Training del modello pari a una percentuale x (70%) del dataset

Training = log_ret[round(len(log_ret)*0.70):]

Training.head()

# Per calcolare lo z-score che normalizza il dataset considero la media mobile a 5 giorni
# e  a 60 giorni come buona proxy per calcolare la variabile standardizzata.

ratios_mavg5 = ratios.rolling(window=5,
                               center=False).mean()

ratios_mavg60 = ratios.rolling(window=60,
                               center=False).mean()
```

```
std_60 = ratios.rolling(window=60,
                        center=False).std()

zscore_60_5 = (ratios_mavg5 - ratios_mavg60)/std_60

# Plot a schermo del grafico
plt.figure(figsize=(15,7))
zscore_60_5.plot()
plt.axhline(0, color='r')
#plt.axhline(1.0, color='red', linestyle='--')
#plt.axhline(-1.0, color='green', linestyle='--')
plt.legend(['Rolling Ratio z-Score', 'Mean', '+1', '-1'])
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/zscore")
plt.show()

# Plot del rapporto dei prezzi con i segnali buy e sell derivanti da z-score statico

plt.figure(figsize=(15,7))

trainlimit = round(len(log_ret)*0.80)

test = ratios[trainlimit+1:]
train = ratios[:trainlimit]
train[60:].plot()
buy = train.copy()
sell = train.copy()
buy[zscore_60_5>-1] = 0
sell[zscore_60_5<1] = 0
buy[60:].plot(color='g', linestyle='None', marker='^')
sell[60:].plot(color='r', linestyle='None', marker='^')
x1,x2,y1,y2 = plt.axis()
plt.axis((x1,x2,ratios.min(),ratios.max()))
plt.legend(['Ratio', 'Buy Signal', 'Sell Signal'])
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/zscorebs")
plt.show()

# Plot dei prezzi degli asset con i segnali buy e sell derivanti da z-score

plt.figure(figsize=(12,7))
S1 = dataframe['GOOGL'].iloc[:trainlimit]
S2 = dataframe['ACN'].iloc[:trainlimit]

S1[60:].plot(color='b')
S2[60:].plot(color='darkorange')
buyR = 0*S1.copy()
```

```
sellR = 0*S1.copy()

# When you buy the ratio, you buy stock S1 and sell S2
buyR[buy!=0] = S1[buy!=0]
sellR[buy!=0] = S2[buy!=0]

# When you sell the ratio, you sell stock S1 and buy S2
buyR[sell!=0] = S2[sell!=0]
sellR[sell!=0] = S1[sell!=0]

buyR[60:].plot(color='g', linestyle='None', marker='^')
sellR[60:].plot(color='r', linestyle='None', marker='^')
x1, x2, y1, y2 = plt.axis()
plt.axis((x1, x2, min(S1.min(), S2.min()), max(S1.max(), S2.max())
    ))



plt.legend(['ACN', 'GOOGL', 'Buy Signal', 'Sell Signal'])
plt.show()

# Nuovo database
new_data = dataframe.iloc[trainlimit+1:][['GOOGL', 'ACN']]
# Seleziona le prime 5 righe
first_5 = new_data.head(5)

# Seleziona le ultime 5 righe
last_5 = new_data.tail(5)

# Concatenare le prime 5 e le ultime 5 righe
new_data = pd.concat([first_5, last_5])
new_data = dataframe.iloc[trainlimit+1:][['GOOGL', 'ACN']]

# Grafico dei prezzi per i due asset
new_data.plot()
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/
    ACNGOOGLprices")

# Definizione della funzione per trading

def trade(S1, S2, window1, window2, transaction_cost,
    zscore_window):

    if (window1 == 0) or (window2 == 0) or (zscore_window == 0):
        return 0

    # Calcola il rapporto e le moving averages
    ratios = S1 / S2
    ma1 = ratios.rolling(window=window1, center=False).mean()
```

```python
    ma2 = ratios.rolling(window=window2, center=False).mean()
    std = ratios.rolling(window=window2, center=False).std()
    zscore = (ma1 - ma2) / std

    # Calcola le soglie dinamiche
    zscore_mean = zscore.rolling(window=zscore_window, center=False).mean()
    zscore_std = zscore.rolling(window=zscore_window, center=False).std()

    upper_threshold = zscore_mean + 2*zscore_std
    lower_threshold = zscore_mean - 2*zscore_std

    exit_threshold = zscore_mean

    # Simulazione della trade
    money = 100
    countS1 = 0
    countS2 = 0
    countS1_prev = 0
    countS2_prev = 0
    for i in range(len(ratios)):
        # Sell short se z-score > upper_threshold
        if zscore[i] > upper_threshold[i]:
            money -= (S1[i] - S2[i] * ratios[i])
            countS1 += 1
            countS2 -= ratios[i]
            money -= transaction_cost * (abs(countS1 - countS1_prev) + abs(countS2 - countS2_prev))
            countS1_prev = countS1
            countS2_prev = countS2
            #print(f'Selling Ratio {money} {ratios[i]} {countS1} {countS2}')
        # Buy long se z-score < lower_threshold
        elif zscore[i] < lower_threshold[i]:
            money += (S1[i] - S2[i] * ratios[i])
            countS1 -= 1
            countS2 += ratios[i]
            money -= transaction_cost * (abs(countS1 - countS1_prev) + abs(countS2 - countS2_prev))
            countS1_prev = countS1
            countS2_prev = countS2
            #print(f'Buying Ratio {money} {ratios[i]} {countS1} {countS2}')
        # Azzera le posizioni se z-score   vicino a exit_threshold
        elif abs(zscore[i] - exit_threshold[i]) < 0.25:
            money += (S1[i] * countS1 + S2[i] * countS2)
            countS1 = 0
```

```
            countS2 = 0
            #print(f'Exit pos {money} {ratios[i]} {countS1} {countS2}')

    return money, S1[-1], S2[-1], countS1, countS2

# Trade
results = trade(dataframe['GOOGL'].iloc[trainlimit+1:], dataframe['ACN'].iloc[trainlimit+1:], 60, 5,2,15)
results

#Calcolo dei pesi

def calculate_weights(quantities):
    total_value = sum(quantities)
    weights = [(quantity / total_value) for quantity in quantities]
    return weights

# Calcolo pesi
asset_quantities = results[3:]
weights = calculate_weights(asset_quantities)
print("Pesi degli asset:", weights)

# Codice per calcolare zscore, soglie e tracciare il grafico

zscore_window = 15
window1 = 5
window2 = 60
S1 = dataframe['GOOGL'].iloc[trainlimit+1:]
S2 = dataframe['ACN'].iloc[trainlimit+1:]

ratios = S1 / S2
ma1 = ratios.rolling(window=window1, center=False).mean()
ma2 = ratios.rolling(window=window2, center=False).mean()
std = ratios.rolling(window=window2, center=False).std()
zscore = (ma1 - ma2) / std

# Calcola le soglie dinamiche
zscore_mean = zscore.rolling(window=zscore_window, center=False).mean()
zscore_std = zscore.rolling(window=zscore_window, center=False).std()

# Creiamo il grafico
plt.figure(figsize=(12, 6))
plt.plot(zscore.index, zscore.values, label='Z-Score')

b_entry_signals = []
```

```python
s_entry_signals = []
exit_signals = []

upper_threshold = zscore_mean + 2*zscore_std
lower_threshold = zscore_mean - 2*zscore_std

exit_threshold = zscore_mean

for i in range(len(zscore)):
    if zscore[i] > upper_threshold[i]:
        s_entry_signals.append((zscore.index[i], zscore.values[i]))
    elif zscore[i] < lower_threshold[i]:
        b_entry_signals.append((zscore.index[i], zscore.values[i]))
    elif abs(zscore[i] - exit_threshold[i]) < 0.25:
        exit_signals.append((zscore.index[i], zscore.values[i]))

# Aggiungiamo punti di ingresso e uscita al grafico
entry_dates_b, entry_values_b = zip(*b_entry_signals)
entry_dates_s, entry_values_s = zip(*s_entry_signals)
exit_dates, exit_values = zip(*exit_signals)

plt.scatter(entry_dates_b, entry_values_b, color='g', marker='^',
    label='Entry Signal')
plt.scatter(entry_dates_s, entry_values_s, color='r', marker='^',
    label='Entry Signal')
plt.scatter(exit_dates, exit_values, color='y', marker='^', label=
    'Exit Signal')

plt.title('Z-Score con segnali in entrata e in uscita')
plt.xlabel('Date')
plt.ylabel('Z-Score')
plt.legend(loc='upper left', bbox_to_anchor=(1, 1))
plt.grid(True)
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/
    zscorebuysell")
plt.show()

# Calcolo del valore del Portafoglio

def trade(S1, S2, window1, window2, transaction_cost,
    zscore_window):
    if (window1 == 0) or (window2 == 0) or (zscore_window == 0):
        return 0

    # Calcola il rapporto e le moving averages
    ratios = S1 / S2
    ma1 = ratios.rolling(window=window1, center=False).mean()
```

```
    ma2 = ratios.rolling(window=window2, center=False).mean()
    std = ratios.rolling(window=window2, center=False).std()
    zscore = (ma1 - ma2) / std

    # Calcola le soglie dinamiche
    zscore_mean = zscore.rolling(window=zscore_window, center=False).mean()
    zscore_std = zscore.rolling(window=zscore_window, center=False).std()

    upper_threshold = zscore_mean + 2 * zscore_std
    lower_threshold = zscore_mean - 2 * zscore_std
    exit_threshold = zscore_mean

    # Simulazione della trade
    money = 100
    countS1 = 0
    countS2 = 0
    countS1_prev = 0
    countS2_prev = 0

    portfolio_values = []
    positions = []

    for i in range(len(ratios)):
        if zscore[i] > upper_threshold[i]:
            money -= (S1[i] - S2[i] * ratios[i])
            countS1 += 1
            countS2 -= ratios[i]
            money -= transaction_cost * (abs(countS1 - countS1_prev) + abs(countS2 - countS2_prev))
            countS1_prev = countS1
            countS2_prev = countS2
            #print(f'Selling Ratio {money} {ratios[i]} {countS1} {countS2}')
        elif zscore[i] < lower_threshold[i]:
            money += (S1[i] - S2[i] * ratios[i])
            countS1 -= 1
            countS2 += ratios[i]
            money -= transaction_cost * (abs(countS1 - countS1_prev) + abs(countS2 - countS2_prev))
            countS1_prev = countS1
            countS2_prev = countS2
            #print(f'Buying Ratio {money} {ratios[i]} {countS1} {countS2}')
        elif abs(zscore[i] - exit_threshold[i]) < 0.25:
            money += (S1[i] * countS1 + S2[i] * countS2)
            countS1 = 0
            countS2 = 0
```

```
            #print(f'Exit pos {money} {ratios[i]} {countS1} {countS2}')

        portfolio_values.append(money)
        positions.append((S1.index[i], money, countS1, countS2))

    return portfolio_values, positions

# Eseguiamo la funzione trade e raccogliamo i dati
S1 = dataframe['GOOGL'].iloc[trainlimit+1:]
S2 = dataframe['ACN'].iloc[trainlimit+1:]
portfolio_values, positions = trade(S1, S2, 60, 5, 2, 15)

# Creiamo una tabella con l'evoluzione delle posizioni
positions_df = pd.DataFrame(positions, columns=['Date', 'Portfolio Value', 'Count S1', 'Count S2'])
positions_df.set_index('Date', inplace=True)
print(positions_df.head())

# Creiamo il grafico dell'evoluzione del valore del portafoglio
plt.figure(figsize=(12, 6))
plt.plot(positions_df.index, positions_df['Portfolio Value'], label='Portfolio Value')
plt.title('Evoluzione del valore del portafoglio nel tempo')
plt.xlabel('Data')
plt.ylabel('Valore del portafoglio')
plt.legend()
plt.grid(True)
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/portfoliovaluesoloPairs")
plt.show()

portfolio_values = pd.DataFrame(positions_df.iloc[:,0])
portfolio_values

portfolio_values.std()

# Importo i dati
def getData(stocks, start, end):
    stockData = pdr.get_data_yahoo(stocks, start=start, end=end)
    stockData = stockData['Close']
    returns = stockData.pct_change()
    meanReturns = returns.mean()
    covMatrix = returns.cov()
    return returns, meanReturns, covMatrix

# Portfolio Performance
def portfolioPerformance(weights, meanReturns, covMatrix, Time):
    returns = np.sum(meanReturns*weights)*Time
```

```python
    std = np.sqrt( np.dot(weights.T, np.dot(covMatrix, weights)) ) * np.sqrt(Time)
    return returns, std

stockList = ['GOOGL', 'ACN']
stocks = [stock for stock in stockList]

# Scelta intervallo temporale
start = datetime.datetime(2023, 4, 29)
end = datetime.datetime(2024, 4, 30)

returns, meanReturns, covMatrix = getData(stocks, start, end)
returns = returns.dropna()
weights

# Definizione funzione statistiche di portafoglio

def portfolioStatistics(returns, weights, risk_free_rate=0):
    if len(weights.shape) == 1:
        weights = weights[:, np.newaxis]

    portfolio_return = np.sum(returns.mean() * weights) * 252
    portfolio_std_dev = np.sqrt(np.dot(weights.T, np.dot(returns.cov() * 252, weights)))
    sharpe_ratio = (portfolio_return - risk_free_rate) / portfolio_std_dev
    sortino_ratio = (portfolio_return - risk_free_rate) / (np.sqrt(np.mean(np.minimum(returns - risk_free_rate, 0)) * 252))
    max_drawdown = calculate_max_drawdown(returns)

    return {
        'Portfolio Return': portfolio_return,
        'Portfolio Standard Deviation': portfolio_std_dev,
        'Sharpe Ratio': sharpe_ratio,
        'Sortino Ratio': sortino_ratio,
        'Max Drawdown': max_drawdown
    }

stockList = ['GOOGL', 'ACN']
stocks = [stock for stock in stockList]


returns, meanReturns, covMatrix = getData(stocks, start, end)
returns = returns.dropna()


weights /= np.sum(weights)

returns['portfolio'] = returns.dot(weights)
```

```python
weights
returns.head(80)

# Calcolo Sharpe Ratio
def sharpe_ratio(returns, risk_free_rate):
    """
    Calcola lo Sharpe Ratio di una serie di rendimenti.

    Args:
    - returns (array-like): Serie di rendimenti.
    - risk_free_rate (float, optional): Tasso di interesse privo
   di rischio, di default    0.

    Returns:
    - float: Sharpe Ratio calcolato.
    """
    excess_returns = returns - risk_free_rate
    mean_excess_return = np.mean(excess_returns*100)
    std_dev = np.std(returns)
    sharpe_ratio = mean_excess_return / std_dev
    return sharpe_ratio
retport = np.log(portfolio_values).diff()*100
risk_free_rate = 0.0387/252
sharpe_ratio(retport, risk_free_rate)

# Calcolo Downside Risk

ret = np.log(portfolio_values).diff()
negative_returns = ret[ret < 0].dropna()
n = len(negative_returns)
MAR = 0.0387/252
DR = np.sqrt(1/n * np.sum((negative_returns - MAR)**2))
DR

negative_returns.std()

# Calcolo del Drawdown

Max = portfolio_values.max()
Min = portfolio_values.min()
Drawdown = (Max-Min)/Max
Drawdown

# Calcolo Sharpe Ratio
def sharpe_ratio(returns, risk_free_rate):
    """
    Calcola lo Sharpe Ratio di una serie di rendimenti.

    Args:
```

```
    - returns (array-like): Serie di rendimenti.
    - risk_free_rate (float, optional): Tasso di interesse privo
   di rischio, di default    0.

    Returns:
    - float: Sharpe Ratio calcolato.
    """
    excess_returns = returns - risk_free_rate
    mean_excess_return = np.mean(excess_returns*100)
    std_dev = np.std(negative_returns)
    sharpe_ratio = mean_excess_return / std_dev
    return sharpe_ratio
retport = np.log(portfolio_values).diff()*100
risk_free_rate = 0.0387/252
sharpe_ratio(retport, risk_free_rate)

# Calcolo del VaR storico e cVaR storico

def historicalVaR(returns, alpha=5):
    if isinstance(returns, pd.Series):
        return np.percentile(returns, alpha)
    elif isinstance(returns, pd.DataFrame):
        return returns.aggregate(historicalVaR, alpha=alpha)
    else:
        raise TypeError("Expected returns to be dataframe or
   series")

def historicalCVaR(returns, alpha=5):
    if isinstance(returns, pd.Series):
        belowVaR = returns <= historicalVaR(returns, alpha=alpha)
        return returns[belowVaR].mean()
    elif isinstance(returns, pd.DataFrame):
        return returns.aggregate(historicalCVaR, alpha=alpha)
    else:
        raise TypeError("Expected returns to be dataframe or
   series")

#5 Giorni
Time = 5

hVaR = -historicalVaR(returns['portfolio'], alpha=5) * np.sqrt(
   Time)
hCVaR = -historicalCVaR(returns['portfolio'], alpha=5) * np.sqrt(
   Time)
pRet, pStd = portfolioPerformance(weights, meanReturns, covMatrix,
    Time)

InitialInvestment = results[0]
print('Expected Portfolio Return:      ', round(InitialInvestment*
```

```
pRet, 2))
print('Value at Risk 95th CI    :      ', round(InitialInvestment*hVaR, 2))
print('Conditional VaR 95th CI  :      ', round(InitialInvestment*hCVaR, 2))

# Grafico per VaR e cVaR storico

InitialInvestment = results[0]
expected_return = round(InitialInvestment * pRet, 2)
value_at_risk = round(InitialInvestment * hVaR, 2)
conditional_var = round(InitialInvestment * hCVaR, 2)

# Creazione dell'istogramma
plt.figure(figsize=(10, 6))
plt.hist(returns['portfolio'], bins=50, alpha=0.75, color='blue', edgecolor='black')

# Linee per VaR e CVaR
plt.axvline(x=-hVaR / np.sqrt(Time), color='red', linestyle='--', linewidth=2, label='VaR 95%')
plt.axvline(x=-hCVaR / np.sqrt(Time), color='orange', linestyle='--', linewidth=2, label='CVaR 95%')

# Aggiungi titolo e etichette
#plt.title('Distribuzione dei rendimenti del portafoglio con VaR e CVaR')
plt.xlabel('Rendimenti del portafoglio')
plt.ylabel('Frequenza')
plt.xlim(-0.07, 0.07)
plt.legend(loc='upper right')
plt.grid(True)
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/histVaRCVaR")
plt.show()

# Calcolo del VaR parametrico e cVaR parametrico

def var_parametric(portofolioReturns, portfolioStd, distribution='normal', alpha=5, dof=6):
    # because the distribution is symmetric
    if distribution == 'normal':
        VaR = norm.ppf(1-alpha/100)*portfolioStd - portofolioReturns
    elif distribution == 't-distribution':
        nu = dof
        VaR = np.sqrt((nu-2)/nu) * t.ppf(1-alpha/100, nu) * portfolioStd - portofolioReturns
    else:
```

```
        raise TypeError("Expected distribution type 'normal'/'t-distribution'")
    return VaR

def cvar_parametric(portofolioReturns, portfolioStd, distribution='normal', alpha=5, dof=6):
    if distribution == 'normal':
        CVaR = (alpha/100)**-1 * norm.pdf(norm.ppf(alpha/100))*portfolioStd - portofolioReturns
    elif distribution == 't-distribution':
        nu = dof
        xanu = t.ppf(alpha/100, nu)
        CVaR = -1/(alpha/100) * (1-nu)**(-1) * (nu-2+xanu**2) * t.pdf(xanu, nu) * portfolioStd - portofolioReturns
    else:
        raise TypeError("Expected distribution type 'normal'/'t-distribution'")
    return CVaR

normVaR = var_parametric(pRet, pStd)
normCVaR = cvar_parametric(pRet, pStd)

tVaR = var_parametric(pRet, pStd, distribution='t-distribution')
tCVaR = cvar_parametric(pRet, pStd, distribution='t-distribution')

print("Normal VaR 95th CI       :      ", round(InitialInvestment*normVaR,2))
print("Normal CVaR 95th CI      :      ", round(InitialInvestment*normCVaR,2))
print("t-dist VaR 95th CI       :      ", round(InitialInvestment*tVaR,2))
print("t-dist CVaR 95th CI      :      ", round(InitialInvestment*tCVaR,2))

# Grafico per VaR e cVaR parametrico

InitialInvestment = results[0]
expected_return = round(InitialInvestment * pRet, 2)
value_at_risk = round(InitialInvestment * hVaR, 2)
conditional_var = round(InitialInvestment * hCVaR, 2)

# Creazione dell'istogramma
plt.figure(figsize=(10, 6))
plt.hist(returns['portfolio'], bins=50, alpha=0.75, color='blue', edgecolor='black')

# Linee per VaR e CVaR
plt.axvline(x=-normVaR / np.sqrt(Time), color='red', linestyle='--', linewidth=2, label='VaR 95%')
```

```
plt.axvline(x=-normCVaR / np.sqrt(Time), color='orange', linestyle
    ='--', linewidth=2, label='CVaR 95%')

# Aggiungi titolo e etichette
#plt.title('Distribuzione dei rendimenti del portafoglio con VaR e
     CVaR')
plt.xlabel('Rendimenti del portafoglio')
plt.xlim(-0.07, 0.07)
plt.ylabel('Frequenza')
plt.legend(loc='upper right')
plt.grid(True)
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/
    paramVaRCVaR")
plt.show()

# Metodo Monte Carlo

mc_sims = 300 # numero di simulazioni
T = 200 # numero giorni

meanM = np.full(shape=(T, len(weights)), fill_value=meanReturns)
meanM = meanM.T

portfolio_sims = np.full(shape=(T, mc_sims), fill_value=0.0)

initialPortfolio = results[0]

for m in range(0, mc_sims):
    # MC loops
    Z = np.random.normal(size=(T, len(weights)))
    L = np.linalg.cholesky(covMatrix)
    dailyReturns = meanM + np.inner(L, Z)
    portfolio_sims[:,m] = np.cumprod(np.inner(weights,
    dailyReturns.T)+1)*initialPortfolio


plt.plot(portfolio_sims)
plt.ylabel('Valore del Portafoglio ($)')
plt.xlabel('N Giorni')
#plt.title('Simulazione di Monte Carlo sul portafoglio')
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/MCsim")
plt.show()

#portfolio_sims.mean()

# Grafico con rendimenti Portafoglio e asset
px.line(returns)

"""# Statistiche varie"""
```

```
meanRet = returns['portfolio'].mean()
meanRet

stdRet = returns['portfolio'].std()
stdRet

var99=np.percentile(returns['portfolio'], 1)
var99

var95=np.percentile(returns['portfolio'], 5)
var95

# Istogramma ritorni
px.histogram(returns)

#TENTATIVO VAR
meanRet

Z_99 = stats.norm.ppf(1-0.99)
price = initialPortfolio
pStd
meanReturns

# VaR simulato MC 5%
np.random.seed(42)
n_sims = 300
sim_returns = np.random.normal(meanRet, stdRet, n_sims)
SimVAR = price*np.percentile(sim_returns, 5)
print('Simulated VAR at 5% is ', SimVAR.round(2))

# VaR simulato MC 1%%
np.random.seed(42)
n_sims = 300
sim_returns = np.random.normal(meanRet, stdRet, n_sims)
SimVAR = price*np.percentile(sim_returns, 1)
print('Simulated VAR at 1% is ', SimVAR.round(2))

# Imposta il seed per la riproducibilit
np.random.seed(42)

# Genera le simulazioni dei rendimenti
n_sims = 300
sim_returns = np.random.normal(meanRet, stdRet, n_sims)

# Calcola il VAR simulato
SimVAR = price * np.percentile(sim_returns, 1)
SimVAR = np.percentile(sim_returns, 1)
# Crea il grafico a istogramma
```

```
plt.figure(figsize=(10, 6))
plt.hist(sim_returns, bins=100, color='skyblue', edgecolor='black', density=True)
#plt.scatter(SimVAR, 0, color='red', marker='x', s= 200, label='Simulated VAR (1%)')
plt.axvline(x=SimVAR, color='red', linestyle='--', linewidth=1, label='Simulated VAR (1%)')
plt.axvline(x=np.percentile(sim_returns, 5), color='green', linestyle='--', linewidth=2, label='5th Percentile')
plt.xlabel('Simulated Returns')
plt.ylabel('Frequency')
#plt.title('Monte Carlo Simulations of Returns')
plt.legend()
plt.grid(True)
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/MCsimreturns")
plt.show()

#print('Simulated VAR (1%) is ', SimVAR)

import plotly.express as px
px.histogram(returns['portfolio'])

portfolio_values = pd.DataFrame(portfolio_values)
df_differenced = portfolio_values.diff().dropna()

"""# Markowitz"""

# Richiamo il Dataset con le stesse date
start = datetime.datetime(2022, 12, 30)
end = datetime.datetime(2023, 12, 30)
alltickers_n = alltickers.loc[start:end]

# Calcolo del valore del Portafoglio

def trade(S1, S2, window1, window2, transaction_cost, zscore_window):
    if (window1 == 0) or (window2 == 0) or (zscore_window == 0):
        return 0

    # Calcola il rapporto e le moving averages
    ratios = S1 / S2
    ma1 = ratios.rolling(window=window1, center=False).mean()
    ma2 = ratios.rolling(window=window2, center=False).mean()
    std = ratios.rolling(window=window2, center=False).std()
    zscore = (ma1 - ma2) / std

    # Calcola le soglie dinamiche
    zscore_mean = zscore.rolling(window=zscore_window, center=
```

```
False).mean()
    zscore_std = zscore.rolling(window=zscore_window, center=False
).std()

    upper_threshold = zscore_mean + 2 * zscore_std
    lower_threshold = zscore_mean - 2 * zscore_std
    exit_threshold = zscore_mean

    # Simulazione della trade
    money = 100
    countS1 = 0
    countS2 = 0
    countS1_prev = 0
    countS2_prev = 0

    portfolio_values = []
    positions = []

    for i in range(len(ratios)):
        if zscore[i] > upper_threshold[i]:
            money -= (S1[i] - S2[i] * ratios[i])
            countS1 += 1
            countS2 -= ratios[i]
            money -= transaction_cost * (abs(countS1 -
countS1_prev) + abs(countS2 - countS2_prev))
            countS1_prev = countS1
            countS2_prev = countS2
            #print(f'Selling Ratio {money} {ratios[i]} {countS1} {
countS2}')
        elif zscore[i] < lower_threshold[i]:
            money += (S1[i] - S2[i] * ratios[i])
            countS1 -= 1
            countS2 += ratios[i]
            money -= transaction_cost * (abs(countS1 -
countS1_prev) + abs(countS2 - countS2_prev))
            countS1_prev = countS1
            countS2_prev = countS2
            #print(f'Buying Ratio {money} {ratios[i]} {countS1} {
countS2}')
        elif abs(zscore[i] - exit_threshold[i]) < 0.25:
            money += (S1[i] * countS1 + S2[i] * countS2)
            countS1 = 0
            countS2 = 0
            #print(f'Exit pos {money} {ratios[i]} {countS1} {
countS2}')

        portfolio_values.append(money)
        positions.append((S1.index[i], money, countS1, countS2))

```

```
    return portfolio_values, positions

# Eseguiamo la funzione trade e raccogliamo i dati
S1 = dataframe['GOOGL'][start:end]
S2 = dataframe['ACN'][start:end]
portfolio_values, positions = trade(S1, S2, 60, 5, 2, 15)

# Creiamo una tabella con l'evoluzione delle posizioni
positions_df = pd.DataFrame(positions, columns=['Date', 'Portfolio Value', 'Count S1', 'Count S2'])
positions_df.set_index('Date', inplace=True)
print(positions_df.head())

# Creiamo il grafico dell'evoluzione del valore del portafoglio
plt.figure(figsize=(12, 6))
plt.plot(positions_df.index, positions_df['Portfolio Value'], label='Portfolio Value')
plt.title('Evoluzione del valore del portafoglio nel tempo')
plt.xlabel('Data')
plt.ylabel('Valore del portafoglio')
plt.legend()
plt.grid(True)
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/portfoliovalueprebacktestingpairs")
plt.show()

positions_df_p_reset = positions_df.reset_index()

# Seleziona solo le colonne 'Date' e 'Portfolio Value'
portfolio_value = positions_df_p_reset[['Date', 'Portfolio Value']]
portfolio_value.set_index('Date', inplace=True)
portfolio_value



mean_returns = alltickers_n.mean()
cov_matrix = alltickers_n.cov()

np.random.seed(123)

solvers.options['show_progress'] = False

# Correlazione pi  bassa

# Calcola la matrice di correlazione tra tutti gli asset
correlation_matrix = alltickers.corr()

# Escludi le correlazioni tra ciascun titolo e se stesso
```

```
np.fill_diagonal(correlation_matrix.values, np.nan)

# Trova i titoli meno correlati considerando il valore assoluto delle correlazioni
min_correlation_pairs = correlation_matrix.abs().min().sort_values()

# Visualizza i titoli meno correlati
print("Titoli meno correlati:")
print(min_correlation_pairs.head(10))
less_correlated_tickers = min_correlation_pairs.head(10).index.tolist()

# Ottengo il DataFrame ristretto
def get_data(ticker):
    data = yf.download(ticker, start, end)['Close']
    return data

# Creazione dei dataframe per ciascuna categoria di ticker
dataframe = {}

for ticker in less_correlated_tickers:
    dataframe[ticker] = get_data(ticker)

dataframe = pd.DataFrame(dataframe, index=get_data(pairs[0][0]).index)
dataframe.tail()
dataframe.plot()
plt.ylabel('Prezzo')

# Stampa i ticker come singoli
ticker = [ticker for ticker in less_correlated_tickers]

print("Lista dei ticker:")
print(ticker)

# Creazione dei dataframe per ciascuna categoria di ticker
dataframe_dict = {}

for ticker in ticker:
        dataframe_dict[ticker] = get_data(ticker)

portfolio = pd.DataFrame(dataframe_dict, index=get_data(pairs[0][0]).index)
dataframe.tail()

# Rendimenti
returns = portfolio.pct_change()
returns
```

```python

# Grafico rendimenti
plt.figure(figsize=(14, 8))

for t in returns.columns:
    plt.plot(returns.index, returns[t], label=t)

# Aggiungere le etichette degli assi e il titolo
plt.xlabel('Date')
plt.ylabel('Returns')
plt.title('Portfolio Returns Over Time')

# Posizionare la legenda all'esterno del grafico
plt.legend(loc='upper left', bbox_to_anchor=(1, 1))

# Visualizzare il grafico
plt.tight_layout()
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/returnsportfolionopairs")
plt.show()

import pypfopt
from pypfopt import risk_models
from pypfopt import plotting

sample_cov = risk_models.sample_cov(portfolio, frequency=252)

# Grafico correlazioni
S = risk_models.CovarianceShrinkage(portfolio).ledoit_wolf()
plotting.plot_covariance(S, plot_correlation=True);
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/covariancee")

# Libreria per calcolo ottimizzazione Markowitz
from pypfopt import expected_returns

mu = expected_returns.capm_return(portfolio)

mu.plot.barh(figsize=(10,6));

# Calcolo pesi per frontiera efficiente portafoglio senza pair trading
from pypfopt.efficient_frontier import EfficientFrontier
risk_free_rate=0.0387/252
ef = EfficientFrontier(mu, S)
weights = ef.max_sharpe(risk_free_rate=risk_free_rate)

cleaned_weights = ef.clean_weights()
print(dict(cleaned_weights))
```

```
risk_free_rate = 0.0387/252
risk_free_rate=risk_free_rate
ef.portfolio_performance(risk_free_rate=risk_free_rate, verbose=
    True)

# Calcolo delle performance portafoglio senza pair trading
expected_annual_return, annual_volatility, sharpe_ratio = ef.
    portfolio_performance(risk_free_rate=risk_free_rate)
print("Expected annual return:", expected_annual_return)
print("Annual volatility:", annual_volatility)
print("Sharpe ratio:", sharpe_ratio)

# Simulazioni portafoglio senza pair trading
n_samples = 10000
w = np.random.dirichlet(np.ones(len(mu)), n_samples)
rets = w.dot(mu)
stds = np.sqrt((w.T * (S @ w.T)).sum(axis=0))
sharpes = (rets-risk_free_rate) / stds

print("Sample portfolio returns:", rets)
print("Sample portfolio volatilities:", stds)

meanReturns = rets.mean()
meanReturns
weights = list(cleaned_weights.values())

# Estrai solo i numeri dalla lista di tuple
covMatrix = S
weights = np.array(weights)

portfoliovalue = (weights.T * returns).sum(axis = 1)
portfoliovalue
money = np.cumsum(portfoliovalue)

# Calcolo VaR
var = np.percentile(portfoliovalue, 5)
var

# Plot Frontiera efficiente con MC
ef = EfficientFrontier(mu, S)

fig, ax = plt.subplots(figsize= (10,10))
plotting.plot_efficient_frontier(ef, ax=ax, show_assets=False)
# Trova il portafoglio che massimizza lo sharpe
ef2 = EfficientFrontier(mu, S)
ef2.max_sharpe(risk_free_rate=risk_free_rate)
ret_tangent, std_tangent, _ = ef2.portfolio_performance()

```

```
# Genera portafogli casuali
ax.scatter(stds, rets, marker=".", c=sharpes, cmap="viridis_r")
ax.scatter(std_tangent, ret_tangent, c='red', marker='X',s=150,
    label= 'Max Sharpe')
# Formatta
#ax.set_title("Efficient Frontier with random portfolios")
ax.legend()
plt.tight_layout()
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/
    Frontieraefficientenopairs")
plt.show()

# Creazione dell'istogramma
plt.figure(figsize=(10, 6))
plt.hist(portfoliovalue, bins=50, alpha=0.75, color='blue',
    edgecolor='black')

# Linee per VaR e CVaR

plt.axvline(x=var, color='orange', linestyle='--', linewidth=2,
    label='VaR 95%')
# Aggiungi titolo e etichette
#plt.title('Distribuzione dei rendimenti del portafoglio con VaR e
     CVaR')
plt.xlabel('Rendimenti del portafoglio')
plt.ylabel('Frequenza')
plt.legend(loc='upper right')
plt.grid(True)
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/Histogram
     with var pre pairs")
plt.show()

portfolio_values = pd.DataFrame(portfolio_values)
portfolio_values

# Rinomina la colonna del dataset del portafoglio
portfolio_values.rename(columns={0: 'Portfolio'}, inplace=True)
portfolio_values

# Aggiungi Pairs al dataframe
portfolio_p = pd.concat([portfolio, portfolio_value], axis=1)
portfolio_p


# Media rendimenti
mu = expected_returns.capm_return(portfolio_p)

portfolio_p

```

```
# Histbar rendimenti
mu.plot.barh(figsize=(10,6));


# Grafico delle correlazioni (per attivare plot togliere #)
S = risk_models.CovarianceShrinkage(portfolio_p).ledoit_wolf()
#plotting.plot_covariance(S, plot_correlation=True);
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/
    Histogramma orizzontale")

# Pesi frontiera efficiente per portafoglio con pairs
risk_free_rate = 0.0387/252
ef = EfficientFrontier(mu, S)
weights = ef.max_sharpe(risk_free_rate=risk_free_rate)

cleaned_weights = ef.clean_weights()
print(dict(cleaned_weights))

# Definizione performance
ef.portfolio_performance(risk_free_rate=risk_free_rate,verbose=
    True)

# Run delle nuove simulazioni
n_samples = 10000
w = np.random.dirichlet(np.ones(len(mu)), n_samples)
rets = w.dot(mu)
stds = np.sqrt((w.T * (S @ w.T)).sum(axis=0))
sharpes = rets / stds

print("Sample portfolio returns:", rets)
print("Sample portfolio volatilities:", stds)

# Plot della frontiera efficiente con MC con Pairs
ef = EfficientFrontier(mu, S)

fig, ax = plt.subplots(figsize= (10,10))
plotting.plot_efficient_frontier(ef, ax=ax, show_assets=False)

# Cerca e plotta il portafoglio con max sharpe
ef2 = EfficientFrontier(mu, S)
ef2.max_sharpe()
ret_tangent, std_tangent, _ = ef2.portfolio_performance()

# Plot portafogli random
ax.scatter(stds, rets, marker=".", c=sharpes, cmap="viridis_r")
ax.scatter(std_tangent, ret_tangent, c='red', marker='X',s=150,
    label= 'Max Sharpe')

# Formatta
```

```
ax.set_title("Efficient Frontier with random portfolios")
ax.legend()
plt.tight_layout()
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/
    Frontieraefficienteconpairs")
plt.show()

#Stats
meanReturns = rets.mean()
meanReturns
weights = list(cleaned_weights.values())
# Estrai solo i numeri dalla lista di tuple
covMatrix = S
weights

meanReturns = rets.mean()
meanReturns
weights = list(cleaned_weights.values())
# Estrai solo i numeri dalla lista di tuple
covMatrix = S
weights = np.array(weights)
# Rendimenti
returns_p = portfolio_p.pct_change()
returns_p

# Calcolo del VAR
portfoliovalue_p = (weights.T * returns_p).sum(axis = 1)

portfoliovalue_p

# Calcolo del VaR per pairs

var = np.percentile(portfoliovalue_p, 5)
var

#Stats
meanReturns = rets.mean()
meanReturns
weights = list(cleaned_weights.values())
# Estrai solo i numeri dalla lista di tuple
covMatrix = S
weights

meanReturns = rets.mean()
meanReturns
weights = list(cleaned_weights.values())
# Estrai solo i numeri dalla lista di tuple
covMatrix = S
weights = np.array(weights)
```

```
portfoliovalue_p = (weights.T * returns_p).sum(axis = 1)

portfolio_p

# Calcolo del VaR per pairs

var = np.percentile(portfoliovalue_p, 5)
var

# Creazione dell'istogramma
plt.figure(figsize=(10, 6))
plt.hist(portfoliovalue_p, bins=50, alpha=0.75, color='blue',
    edgecolor='black')
plt.xlim(-0.08, 0.08)
# Linee per VaR e CVaR

plt.axvline(x=var, color='orange', linestyle='--', linewidth=2,
    label='VaR 95%')
# Aggiungi titolo e etichette
#plt.title('Distribuzione dei rendimenti del portafoglio con VaR e
     CVaR')
plt.xlabel('Rendimenti del portafoglio')
plt.ylabel('Frequenza')
plt.legend(loc='upper right')
plt.grid(True)
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/
    returnswithpairshist")
plt.show()

"""# Backtesting della strategia"""

# Backtesting della strategia

# Definizione nuove date:
start = datetime.datetime(2023, 4, 30)
end = datetime.datetime(2024, 4, 30)
alltickers_backtest = alltickers.loc[start:end]

pair_p = ['GOOGL', 'ACN']

def get_data(ticker):
    data = yf.download(ticker, start, end)['Close']
    return data

dataframe = get_data(pair_p)
dataframe.tail()

ticker = [ticker for ticker in less_correlated_tickers]
```

```python

portfolio_back = get_data(ticker)
portfolio_back.tail()

# Calcolo del valore del Portafoglio

def trade(S1, S2, window1, window2, transaction_cost, zscore_window):
    if (window1 == 0) or (window2 == 0) or (zscore_window == 0):
        return 0

    # Calcola il rapporto e le moving averages
    ratios = S1 / S2
    ma1 = ratios.rolling(window=window1, center=False).mean()
    ma2 = ratios.rolling(window=window2, center=False).mean()
    std = ratios.rolling(window=window2, center=False).std()
    zscore = (ma1 - ma2) / std

    # Calcola le soglie dinamiche
    zscore_mean = zscore.rolling(window=zscore_window, center=False).mean()
    zscore_std = zscore.rolling(window=zscore_window, center=False).std()

    upper_threshold = zscore_mean + 2 * zscore_std
    lower_threshold = zscore_mean - 2 * zscore_std
    exit_threshold = zscore_mean

    # Simulazione della trade
    money = 100
    countS1 = 0
    countS2 = 0
    countS1_prev = 0
    countS2_prev = 0

    portfolio_values = []
    positions = []

    for i in range(len(ratios)):
        if zscore[i] > upper_threshold[i]:
            money -= (S1[i] - S2[i] * ratios[i])
            countS1 += 1
            countS2 -= ratios[i]
            money -= transaction_cost * (abs(countS1 - countS1_prev) + abs(countS2 - countS2_prev))
            countS1_prev = countS1
            countS2_prev = countS2
            #print(f'Selling Ratio {money} {ratios[i]} {countS1} {countS2}')
```

```
        elif zscore[i] < lower_threshold[i]:
            money += (S1[i] - S2[i] * ratios[i])
            countS1 -= 1
            countS2 += ratios[i]
            money -= transaction_cost * (abs(countS1 - countS1_prev) + abs(countS2 - countS2_prev))
            countS1_prev = countS1
            countS2_prev = countS2
            #print(f'Buying Ratio {money} {ratios[i]} {countS1} {countS2}')
        elif abs(zscore[i] - exit_threshold[i]) < 0.25:
            money += (S1[i] * countS1 + S2[i] * countS2)
            countS1 = 0
            countS2 = 0
            #print(f'Exit pos {money} {ratios[i]} {countS1} {countS2}')

        portfolio_values.append(money)
        positions.append((S1.index[i], money, countS1, countS2))

    return portfolio_values, positions

# Eseguiamo la funzione trade e raccogliamo i dati
S1 = dataframe['GOOGL'][start:end]
S2 = dataframe['ACN'][start:end]
portfolio_values, positions = trade(S1, S2, 60, 5, 2, 15)

# Creiamo una tabella con l'evoluzione delle posizioni
positions_df = pd.DataFrame(positions, columns=['Date', 'Portfolio Value', 'Count S1', 'Count S2'])
positions_df.set_index('Date', inplace=True)
print(positions_df.head())

# Creiamo il grafico dell'evoluzione del valore del portafoglio
plt.figure(figsize=(12, 6))
plt.plot(positions_df.index, positions_df['Portfolio Value'], label='Portfolio Value')
plt.title('Evoluzione del valore del portafoglio nel tempo')
plt.xlabel('Data')
plt.ylabel('Valore del portafoglio')
plt.legend()
plt.grid(True)
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/portfoliovalueOttimizzato")
plt.show()

portfolio_values_back = pd.DataFrame(positions_df.iloc[:,0])
portfolio_values_back
```

```python
# Aggiungi Pairs al dataframe
portfolio_p_back = pd.concat([portfolio_back, portfolio_values_back], axis=1)
portfolio_p_back

# Definisci la data di inizio
start_date = pd.Timestamp('2024-01-01')

# Filtra il DataFrame per includere solo le date a partire dal 1 gennaio
filtered_portfolio_p_back = portfolio_p_back[portfolio_p_back.index >= start_date]

# Visualizza il DataFrame filtrato
print(filtered_portfolio_p_back)


rets_port_filt = filtered_portfolio_p_back.pct_change()



portfolio_returns = rets_port_filt.dot(weights)

# Calcolo del valore cumulativo del portafoglio
initial_investment = 100  # Valore iniziale del portafoglio
portfolio_value = initial_investment * (1 + portfolio_returns).cumprod()

# Calcolo delle statistiche del portafoglio
mean_return = portfolio_returns.mean()*252
volatility = portfolio_returns.std()*np.sqrt(252)
risk_free_rate = 0.0525  # Tasso privo di rischio annualizzato
sharpe_ratio = (mean_return - risk_free_rate) / volatility

# Visualizza le statistiche
print("Rendimento annuo del portafoglio:", mean_return)
print("Volatilit  annua del portafoglio:", volatility)
print("Sharpe Ratio del portafoglio:", sharpe_ratio)

# Visualizza il valore del portafoglio nel tempo
portfolio_value.plot()
plt.ylabel('Valore del portafoglio ($)')
plt.xlabel('Data')
plt.savefig("/Users/nicolagherardi/Desktop/Immagini tesi/Valoreportafogliomassimizzato")



# Drawdown del portafoglio
```

```

min_portfolio_value = portfolio_value.min()
max_portfolio_value = portfolio_value.max()
DD = (max_portfolio_value - min_portfolio_value)/max_portfolio_value * 100
DD


portfolio_returns = portfolio_returns.dropna()
portfolio_returns = pd.DataFrame(portfolio_returns)
portfolio_returns



# Rinomina la colonna del dataset del portafoglio
portfolio_returns.rename(columns={0: 'Portfolio'}, inplace=True)
portfolio_returns



# Seleziona i rendimenti negativi

neg_rets = portfolio_returns.loc[portfolio_returns['Portfolio'] < 0, 'Portfolio']
std_neg = neg_rets.std() * np.sqrt(252) # Annualizzata
Rp = mean_return
Rf = 0.0525

Sortino = (Rp-Rf)/std_neg



portfolio_returns = portfolio_returns.dropna()



# VaR
var = np.percentile(portfolio_returns, 5)
var



# VaR 1%
var = np.percentile(portfolio_returns, 1)
var
```