



Università  
Ca' Foscari  
Venezia

Corso di Laurea Magistrale in  
Economia e Finanza

Tesi di Laurea

## **Le reti neurali a supporto dell'analisi tecnica**

**Relatore**

Claudio Pizzi

**Laureando**

Marco Bernardi

Matricola 860156

**Anno Accademico**

2020 / 2021



# Indice

<b>Introduzione</b>	1
<b>Capitolo I: Analisi tecnica</b>	3
<b>1.1 La teoria di Dow</b>	4
1.1.1 Limiti della teoria di Dow	7
<b>1.2 L'analisi grafica</b>	8
1.2.1 Grafico lineare (line chart)	8
1.2.1 Grafico a barre (bar chart)	9
1.2.3 Grafico punto e croce	10
1.2.4 Grafico equivolume	11
1.2.5 Grafico a candele giapponesi (candlestick)	12
<b>1.3 Concetti fondamentali</b>	13
1.3.1 Pattern grafici	15
<b>1.4 Analisi tecnica algoritmica</b>	19
1.4.1 Indicatori di trend	20
1.4.2 Indicatori di momentum	28
1.4.3 Indicatori di volume	33
1.4.4 Indicatori di volatilità	37
<b>Capitolo II: Le reti neurali</b>	41
<b>2.1 Cenni storici</b>	41
<b>2.2 Che cos'è una rete neurale</b>	42
<b>2.3 Struttura di una rete neurale</b>	43
2.3.1 Tipo di apprendimento	45
2.3.2 Tipo di utilizzo	46
2.3.3 Algoritmo di apprendimento	47
2.3.4 Architettura dei collegamenti	50
<b>2.4 Algoritmo di Backward Propagation Error</b>	51
<b>2.5 Reti neurali: potenzialità e limiti</b>	53
<b>2.6 Reti neurali ambito finanziario</b>	54
2.6.1 Previsione serie temporali	54
2.6.2 Classificazione	55
2.6.3 Approssimazioni di funzioni	56
<b>2.7 Costruzione della rete neurale</b>	56
2.7.1 La definizione dell'ambito di indagine	56

2.7.2 Costruzione dell'archivio dati	56
2.7.3 Apprendimento	59
2.7.4 Indicatori di errore	63
2.7.5 Generalizzazione e previsione della serie storica	65
<b>2.8 Una rete neurale ricorrente: Long-Short Term Memory</b>	<b>67</b>
<b>Capitolo III: Trading System</b>	<b>71</b>
3.1 Data Set	72
3.2 Descrizione Trading System	74
3.3 Risultati del Trading System sulla serie storica originaria	76
<b>Capitolo IV: La previsione con le reti neurali</b>	<b>79</b>
4.1 Architettura del modello	79
<b>Capitolo V: Trading system applicato alla previsione della serie storica</b>	<b>89</b>
<b>Conclusioni</b>	<b>93</b>
<b>Bibliografia</b>	<b>95</b>
<b>Sitografia</b>	<b>98</b>
<b>Appendice A</b>	<b>99</b>
<b>Appendice B</b>	<b>102</b>
<b>Appendice C</b>	<b>112</b>

## Introduzione

Una delle questioni più impegnative di sempre nel mercato borsistico è la previsione dell'andamento del prezzo. L'utilizzo sempre più presente dell'analisi tecnica ha fatto sì che negli ultimi anni la ricerca abbia ampliato i propri orizzonti, portando allo sviluppo di nuove tecniche di ottimizzazione basate sull'utilizzo di nuovi modelli e algoritmi, tra i quali, le più diffuse, sono le tecniche di apprendimento automatico. Lo scopo di questa tesi è capire che strumenti possono venirci in aiuto, in particolare in aiuto all'analisi tecnica per migliorare i segnali di trading. Recentemente, l'uso di tecniche di apprendimento automatico in combinazione con gli indicatori di analisi tecnica sta avendo sempre più successo in quanto migliora e anticipa la capacità di previsione; in questa tesi proveremo a confermare, attraverso l'analisi effettiva dei mercati, questa idea. Inizialmente descriverò le caratteristiche dell'analisi tecnica, com'è nata e come si è sviluppata evidenziando in che ambiti viene applicata e le informazioni che si possono ricavare dal suo utilizzo nei mercati finanziari e in particolare nelle serie storiche finanziarie. Successivamente ci sarà una descrizione delle reti neurali, il loro sviluppo e le varie tipologie di utilizzo che se ne fanno per poi sottolineare in che modo possono aiutare l'analisi tecnica nell'esame di una serie storica. Seguirà la programmazione in R e l'implementazione del trading system utilizzando il software R-studio. La strategia è stata creata utilizzando una combinazione di indici descritti nel primo capitolo e l'obiettivo è creare una strategia che abbia valenza in un paniere di titoli il più possibile non correlati tra loro in modo che possa essere applicata ipoteticamente su qualsiasi titolo azionario. Ci sarà quindi un passaggio intermedio di identificazione dei titoli attraverso lo studio della correlazione. Il periodo scelto è stato di dieci anni, dal 2010 al 2020 per la valutazione del trading system. Infine, verranno introdotte le reti neurali a supporto del trading system per il miglioramento della strategia stessa. In particolare, viene fatta una previsione della serie storica con la quale si cerca di anticipare il movimento del prezzo. L'obiettivo è quello di migliorare o per lo meno replicare l'operatività in quanto i segnali dovranno essere generati in anticipo rispetto a quelli derivanti dalla serie finanziaria originale. La previsione è stata fatta su un periodo più breve, ovvero dal 2010 al 2019, escludendo quindi la pandemia. Questa decisione è stata presa in seguito al ragionamento che la diffusione del Covid-19 e il suo conseguente impatto nell'economia sono stati

eventi decisamente troppo rapidi e imprevedibili da poter essere valutati e inseriti nel modello, e inoltre, provare a prevederli non sarebbe stato interessante ai fini della tesi proprio per le caratteristiche uniche dell'evento.

## Capitolo I: Analisi tecnica

Una definizione sufficientemente precisa di analisi tecnica viene data da John J. Murphy, il quale afferma che “l’analisi tecnica è lo studio del movimento del mercato tramite l’uso sistematico di grafici, allo scopo di prevedere l’andamento futuro dei prezzi.” Dalla definizione si evince che può essere considerata una scienza che utilizza un insieme di indicatori e grafici al fine di prevedere le future tendenze delle dinamiche borsistiche.

L’analista fa uso quindi dell’analisi tecnica grafica, ovvero lo studio dei grafici dei prezzi, al fine di individuare eventuali pattern significativi, e l’analisi tecnica algoritmica che si sostanzia nell’utilizzo di indicatori tecnici per la determinazione delle strategie operative da mettere in atto. Si sottolinea che non ci si trova di fronte ad una scelta fra questi due strumenti, ma devono essere utilizzati congiuntamente al fine di massimizzare l’efficacia della strategia.

Spesso l’analisi tecnica viene contrapposta all’analisi fondamentale, la quale cerca strategie d’investimento attraverso lo studio del valore intrinseco di un’azienda, che può essere identificato attraverso la sua solidità patrimoniale e finanziaria (variabili microeconomiche) o influenze derivanti dal sistema economico nella sua generalità (variabile macroeconomica).

Una differenza tra l’analisi fondamentale e tecnica è il periodo di valutazione. Per l’analisi fondamentale si tende ad avere una visione di lungo periodo in quanto le valutazioni sono fatte attraverso i bilanci e gli indici di un’azienda, documenti rilasciati solitamente una volta all’anno e che quindi portano a valutazioni che devono essere fatte su un periodo di tempo esteso. Diversamente, l’analisi tecnica è molto più dinamica e non ha queste limitazioni in quanto grafici e indici possono avere scale ben più ridotte, e quindi si prestano meglio per le valutazioni di breve periodo. Fondamentale per l’analisi tecnica è il timing di entrata ed uscita dal mercato in quanto la sua natura previsionale ha lo scopo di anticipare i movimenti del mercato.

L’analisi tecnica si basa su tre principi fondamentali:

1. *Il mercato “sconta” tutto*: questa affermazione è fondamentale per l’analisi tecnica, l’analista agisce con la convinzione che il prezzo sconti qualsiasi informazione possibile e che quindi siano incorporati quei fattori di tipo fondamentale, politico, psicologico, ecc., che ne hanno determinato l’andamento. Di conseguenza, ai fini della previsione, tutto quello che viene richiesto è lo studio dei movimenti di prezzo. Ciò che viene asserito è che se la domanda supera

l'offerta le quotazioni dovrebbero salire, viceversa, se l'offerta supera la domanda il prezzo teoricamente scende, questa teoria è alla base di tutte le previsioni economiche. I grafici, quindi, non influiscono in alcun modo nelle dinamiche del mercato ma riflettono il suo pensiero rialzista o ribassista.

2. *Il mercato si muove in trend*: il mercato non si muove in modo del tutto casuale ma segue delle tendenze, dette "trend". Un trend si presuppone intatto finché non viene confermata la sua inversione dagli indicatori, è più facile che abbia un andamento continuo piuttosto che una brusca inversione. L'obiettivo dell'analista tecnico è individuare il trend in essere per assumere posizioni nella direzione del trend stesso o anticiparne l'inversione.
3. *La storia si ripete*: quanto detto indica che a fronte di situazioni passate analoghe si presuppone che il comportamento degli investitori sia lo stesso e che ciò sia poi riflesso sul prezzo, perciò al ripresentarsi di determinati movimenti del trend, dovuti a comportamenti ben definiti degli operatori, si sia in grado di anticiparli.

## 1.1 La teoria di Dow

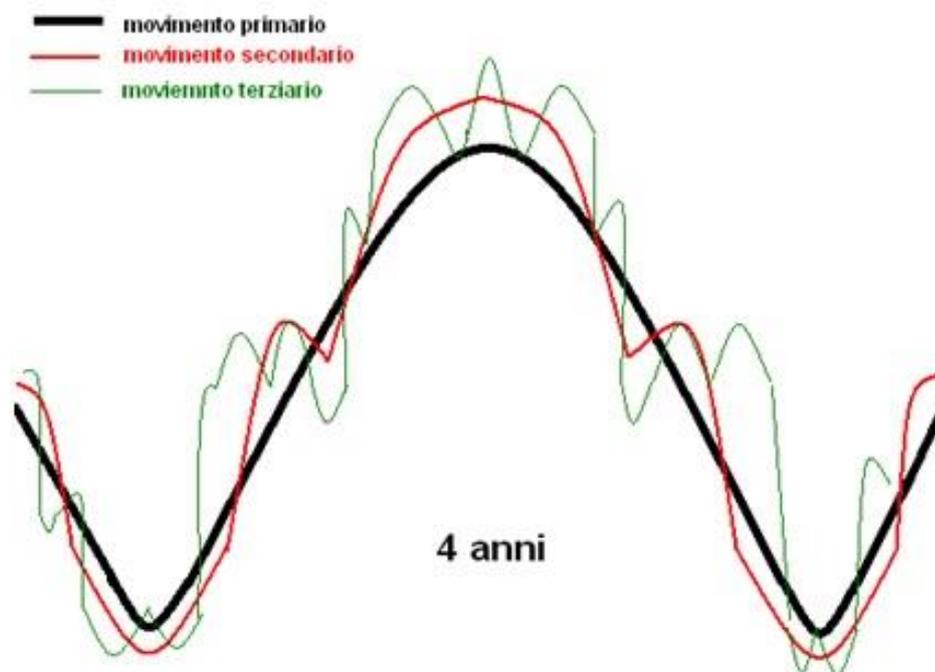
Le origini dell'analisi tecnica si possono ricondurre agli studi effettuati da Charles Dow attorno gli inizi del 1900 e pubblicati in diversi articoli sul Wall Street Journal. Dow, considerando l'andamento dei titoli azionari un ottimo indicatore per la valutazione delle condizioni del mercato, costruì due indici, il Dow Jones Industrial Average (DJIA) e il Dow Jones Transportation Average (DJTA), che sintetizzavano l'andamento medio ponderato dei prezzi delle azioni, rispettivamente, di dodici delle più importanti società industriali (oggi trenta) e di venti società operanti nel trasporto ferroviario.

Si può riassumere la sua teoria in sei principi fondamentali:

1. *Le medie-indice scontano tutto*: i prezzi di chiusura riflettono tutte le informazioni sia correnti sia potenziali, tenendo conto delle decisioni e delle emozioni di tutti gli operatori del mercato e tutto ciò venga scontato e formi il prezzo corrente.
2. *Il mercato ha tre tipi di movimento*: nel mercato azionario ci sono contemporaneamente tre movimenti, il movimento primario o principale, le reazioni secondarie e i movimenti minori. Il movimento primario è il più consistente ed importante, solitamente non dura meno di un anno ed è formato da ampie movimentazioni al rialzo o al ribasso del prezzo, chiamati rispettivamente

bull and bear trend. Non c'è una regola che definisce la durata del movimento, ma si presuppone che tale trend continui fino a quando non viene provato chiaramente il contrario. Ad esempio, un trend primario rialzista è caratterizzato da un lungo movimento a rialzo dei prezzi e questo viene confermato fino a quando non c'è un brusco ed importante ribasso. Le reazioni secondarie sono delle correzioni del trend in corso, possono durare diversi mesi e sono tipicamente identificate come un grande declino (rialzo) in un trend primario rialzista (ribassista). Infine, i movimenti minori, che possono durare da qualche ora fino a qualche settimana possono essere facilmente manipolati e non hanno valore previsionale per investimenti di lungo termine.

Figura 1: I movimenti del trend.

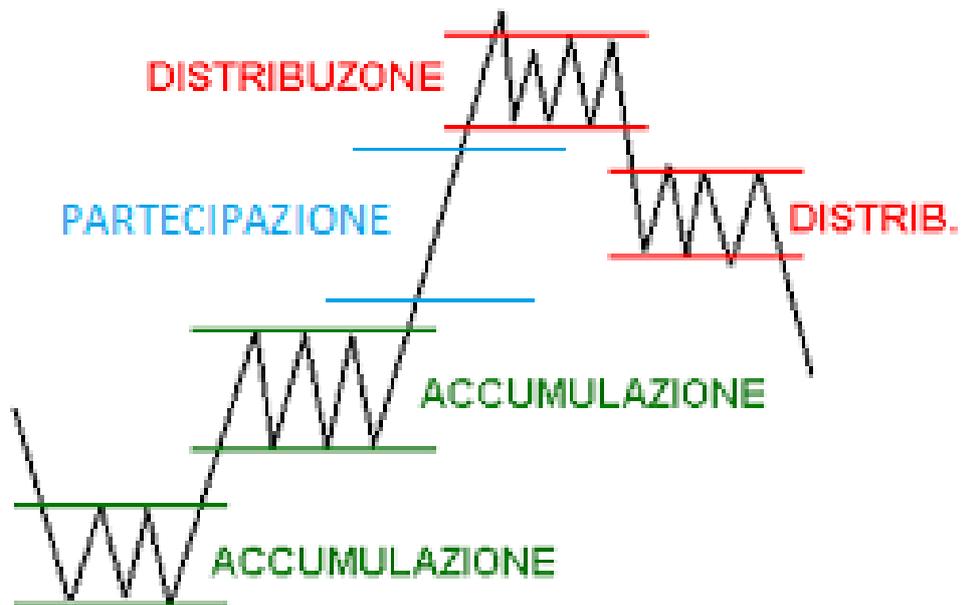


Fonte: <https://www.percorsofinanza.it/>.

3. *Il trend primario ha tre fasi:* ci sono due tipi di operatori che operano sul mercato, le c.d. “mani forti”, caratterizzate da più ampie disponibilità finanziarie, maggiori e migliori informazioni a disposizione e con la capacità di muoversi razionalmente sul mercato, e le “mani deboli” caratterizzate da minori disponibilità economiche, limitate conoscenze e operanti irrazionalmente, principalmente seguendo il mercato. Il comportamento di queste due categorie di persone secondo Dow

risulta essere la causa della ciclicità del mercato. La prima fase di un trend rialzista è la **fase di accumulazione** ed è creata dagli acquisti degli operatori più informati che ritengono il titolo attualmente sottovalutato dal mercato. In questa fase quindi il prezzo scende sempre più lentamente fino ad assestarsi nel punto di equilibrio tra l'acquisto delle mani forti e la vendita delle mani deboli. Successivamente, si ha la **fase di partecipazione pubblica** in cui il prezzo comincia a salire supportato anche dalla domanda delle mani deboli che vedendo il prezzo iniziare a salire diminuiscono l'offerta. In questo momento comincia la terza fase, detta **di distribuzione** in cui c'è euforia, caratterizzata da un costante aumento del prezzo, supportato dalle mani deboli in cui aumenta il volume speculativo. Proprio sul finire di questa fase, quando comincia a rallentare l'aumento del prezzo le mani forti si rendono conto che il prezzo ha superato il vero valore e cominciano a vendere, invertendo il trend in atto.

Figura 2: Fasi del ciclo di Dow.



Fonte: <http://www.pianetagratis.it/borsa/analisi-tecnica-base/dow.htm>.

4. *Le relazioni tra prezzo e volume costituiscono lo sfondo:* ciò implica, in un normale contesto, che i volumi di scambio aumentino quando i prezzi crescono, viceversa, diminuiscano quando il prezzo scende. Reazioni opposte, quindi,

possono essere indicatrici dell'inversione del trend in atto. Tendenzialmente queste divergenze si possono avere sia nelle fasi di accumulazione, in quanto il prezzo scende ma si ha un aumento dei volumi dovuti dall'acquisto delle mani forti, sia nelle fasi di distribuzione in quanto il prezzo crescente non è più supportato dai volumi che iniziano a calare creando i presupposti per la successiva inversione di tendenza.

5. *L'azione del prezzo determina la tendenza:* una serie successiva di incrementi dei punti di massimo determina la presenza di un trend rialzista, mentre i punti di minimo sono meno rilevanti, viceversa, una serie successiva di diminuzione dei punti di minimo indica un trend ribassista, nel quale i punti di massimo sono irrilevanti.
6. *Le medie-indice devono confermarsi a vicenda:* i due indici DJIA e DJTA devono essere considerati simultaneamente e devono sempre confermarsi l'uno con l'altro. Se effettivamente i due indici sono degli indicatori dello stato di salute del mercato, in un'economia in espansione i prezzi dovranno essere crescenti sia per quanto riguarda le società che producono merci sia per quelle che le trasportano.

### 1.1.1 Limiti della teoria di Dow

Uno dei principali limiti della teoria di Dow è che la strategia non risulta essere tempestiva: se l'obiettivo è individuare il trend in atto e anticiparne i movimenti la teoria fornisce segnali troppo ritardati che determinano una perdita di possibili profitti; se poi, si tiene conto degli eventuali costi di commissione presenti nell'operatività il rischio di non conseguire un profitto è ben più concreto. Inoltre, vi è una mancanza di opportune segnalazioni che consentano di sfruttare movimenti intermedi speculativi, la teoria di Dow utilizza esclusivamente i prezzi di chiusura di un titolo, ma nella realtà le quotazioni subiscono forti oscillazioni anche all'interno della giornata stessa.

Un altro limite della teoria è che gli strumenti grafici e gli indicatori dell'analisi tecnica assolvono solo alla funzione del timing, permettendo, almeno in linea teorica, una buona entrata o uscita dal mercato, ma non hanno alcuna capacità di selezione dei titoli con i quali operare e l'investitore non riceve alcuna indicazione dei titoli da inserire in portafoglio.

## 1.2 L'analisi grafica

Come precedentemente descritto l'analisi tecnica si avvale principalmente di strumenti grafici supportati dagli indicatori al fine di interpretare un segnale. Ciò significa che c'è anche una componente soggettiva in quanto è l'analista stesso a decidere e individuare la presenza di trend in atto, possibili inversioni o eventuali pattern grafici significativi che possano descrivere l'andamento futuro del titolo.

Esistono diverse tipologie di grafici che sintetizzano le informazioni rilevanti quali prezzo di apertura, prezzo di chiusura, massimi, minimi e volumi di scambio. Tra le principali rappresentazioni grafiche troviamo:

- Grafico lineare
- Grafico a barre
- Grafico punto e croce
- Grafico equivolume
- Grafico a candele giapponesi

### 1.2.1 Grafico lineare (line chart)

Si tratta del grafico più semplice da costruire, tramite un piano cartesiano avente la scala dei prezzi in ordinata e l'arco temporale in ascissa e il prezzo preso in considerazione è solo quello di chiusura, di riferimento o l'ultimo fatto dal mercato, collegato tramite una semplice linea. Normalmente, date le poche informazioni disponibili non viene utilizzato per il trading speculativo ma per valutazioni di lungo termine in quanto permette di individuare in maniera approssimativa la direzione principale del trend.

Figura 3: Line chart del titolo Ftsemib dal 1 gennaio 2018 al 31 dicembre 2020.



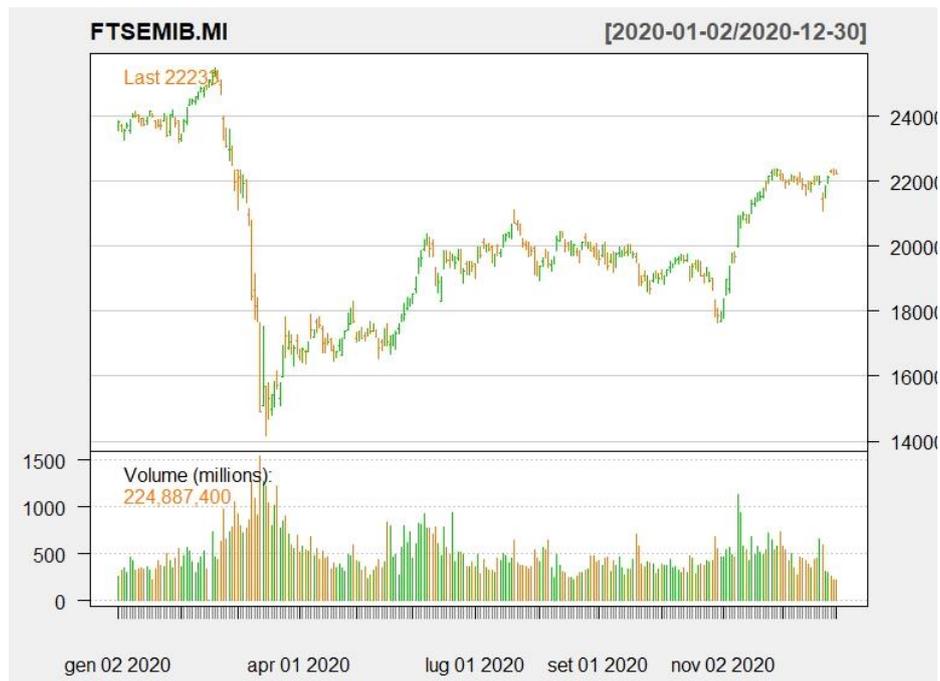
Fonte: grafico ottenuto utilizzando R-studio.

### 1.2.1 Grafico a barre (bar chart)

In questo grafico la rappresentazione del piano cartesiano è la stessa vista precedentemente ma le informazioni presenti sono maggiori. Il grafico rappresenta le escursioni giornaliere dei prezzi tramite la successione di segmenti verticali. In ciascun segmento viene evidenziato il range del prezzo, dal minimo al massimo raggiunto in giornata, inoltre, sono indicati i prezzi di apertura (tratto orizzontale sulla sinistra) e di chiusura (tratto orizzontale sulla destra) del titolo. Quando il prezzo di chiusura sarà maggiore rispetto a quello di apertura la barra sarà di colore verde, viceversa, si colorerà di rosso. Questo tipo di rappresentazione è utile per individuare pattern grafici che con il grafico lineare non sono visibili.

Non di rado nella parte inferiore del grafico sono rappresentati anche i volumi di scambio della giornata.

Figura 4: Bar chart del titolo Ftsemib dal 1 gennaio 2020 al 31 dicembre 2020.



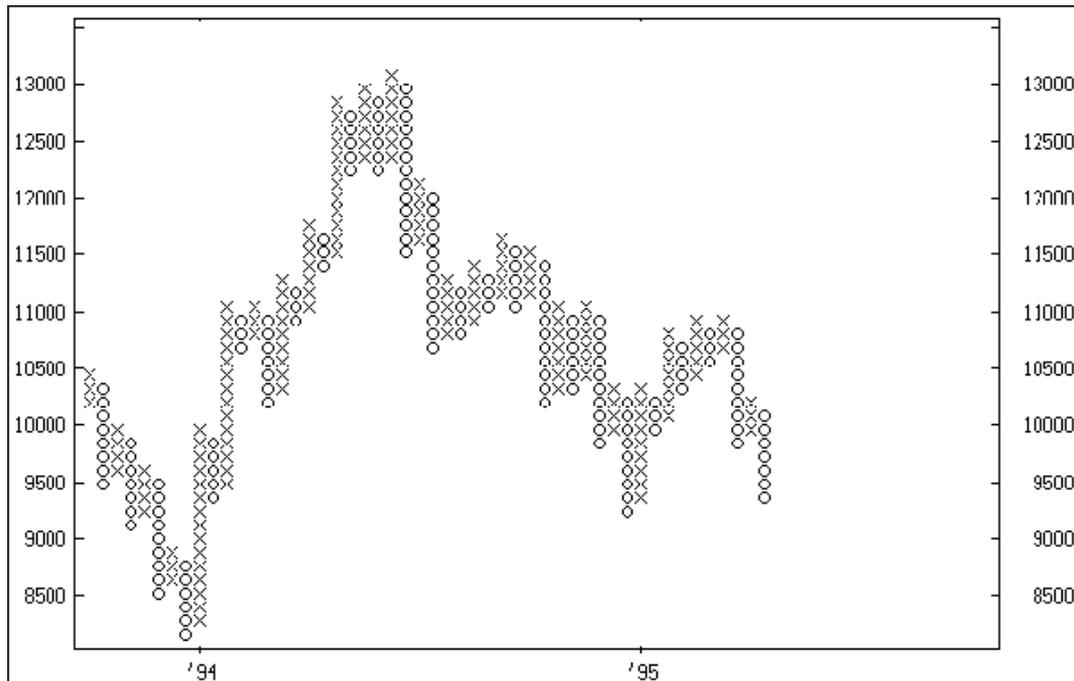
Fonte: grafico ottenuto utilizzando R-studio.

### 1.2.3 Grafico punto e croce

Questo metodo comprende una serie di colonne contraddistinte dai simboli X e O, rispettivamente in caso di risalita discesa del prezzo, ogni simbolo indica un'ampiezza del movimento, chiamata box size, predefinita.

Va ricordato che questa tipologia di grafico venne sviluppata dai traders tra il 1880 e il 1890, ovvero prima della nascita della teoria di Dow. Ad oggi, questa tipologia di grafico risulta scarsamente utilizzata.

Figura 5: Grafico punto e croce

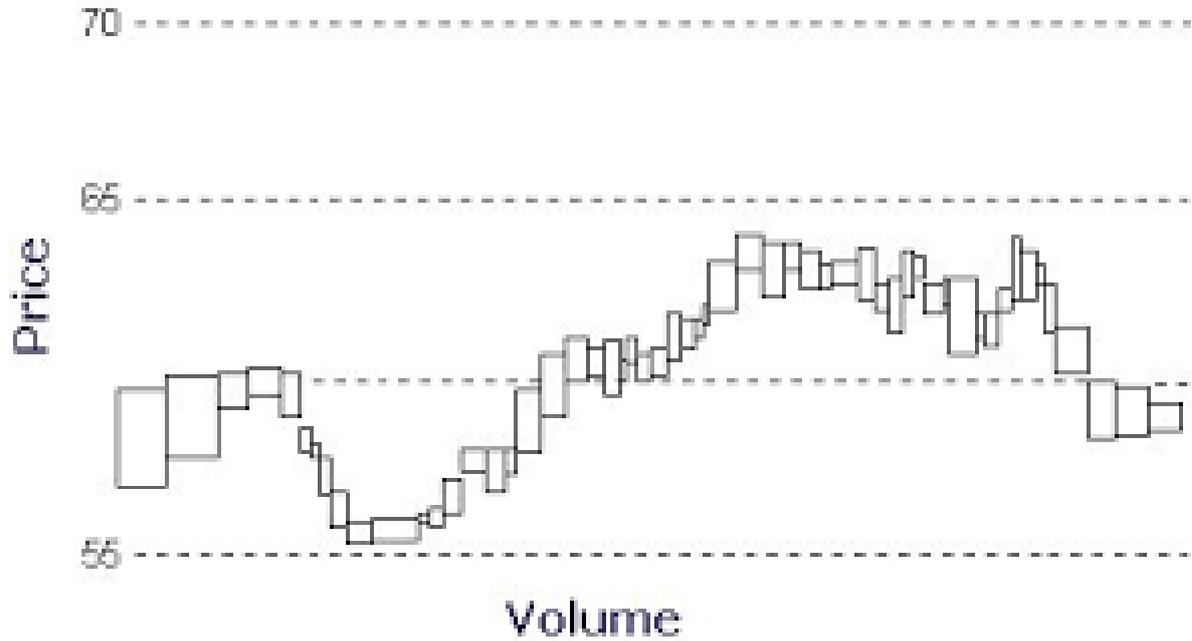


Fonte: [https://www.traderlink.it/didattica/corso\\_analisi\\_tecnica/](https://www.traderlink.it/didattica/corso_analisi_tecnica/).

#### 1.2.4 Grafico equivolume

Questa tipologia di grafico, rispetto agli altri, ci permette di enfatizzare in maniera particolare la relazione tra volume di scambio e relativi prezzi combinando le due informazioni graficamente. Il grafico equivolume è costruito creando un rettangolo per ogni unità temporale la cui altezza indica la relativa escursione del prezzo mentre la larghezza il relativo volume di scambio. Fondamentali risultano quindi essere la forma e le dimensioni dei rettangoli.

Figura 6: Grafico equivolume.



Fonte: <https://www.mql5.com/en/articles/1504>.

### 1.2.5 Grafico a candele giapponesi (candlestick)

È la rappresentazione grafica più utilizzata e conosciuta, molto simile al grafico a barre, mostra prezzo di apertura, chiusura, massimi e minimi del periodo considerato. La “candela”, anch’essa a colori, è formata da un “real body” (rettangolo ampio) formato in base al prezzo di apertura e chiusura, in particolare sarà di colore verde se il prezzo di chiusura è più alto di quello di apertura e di colore rosso in caso contrario; e da due “shadows” o code, segmenti verticali che indicano il prezzo massimo ed il prezzo minimo raggiunto nella seduta. Questa metodologia, rispetto alle altre, permette di individuare facilmente specifiche forme delle candele e specifici pattern che possono dare indicazioni riguardo il consolidamento o l’inversione del trend.

Figura 7: Candlestick chart del titolo Ftsemib dal 1 ottobre 2020 al 31 dicembre 2020.



Fonte: grafico ottenuto utilizzando R-studio.

### 1.3 Concetti fondamentali

Prima di vedere le più importanti formazioni grafiche è utile avere bene in mente alcuni concetti base:

- **Trend-line:** in generale il trend rappresenta la direzione del mercato, ma esso, spesso, non segue un andamento lineare, ma è formato da un movimento a “onde” in cui si formano massimi e minimi relativi consecutivi. In particolare, chiameremo un trend ascendente se è formato da almeno due minimi consecutivi crescenti in cui è possibile tracciare una linea retta che li collega, viceversa, discendente quando è possibile tracciare una linea retta tra due punti di massimo consecutivi decrescenti.
- **Supporti e resistenze:** il supporto in analisi tecnica è quel livello dove solitamente si ha un arresto dell’andamento a ribasso del prezzo e si ha quindi una concentrazione di acquisti che ne rallentano la discesa, mentre la resistenza è dove si blocca la crescita del prezzo perché l’elevato volume di vendite ne impedisce il rialzo; infatti, si può dire che le trend-line rappresentano la resistenza ed il supporto nelle rispettive fasi di crescita e decrescita del mercato. In caso i

supporti o le resistenze siano rappresentati da linee parallele all'asse delle ascisse si dirà che sono statici, invece, se sono rappresentati da linee di tendenza oblique saranno dinamici. Se sono presenti sia dei supporti che delle resistenze nello stesso periodo analizzato allora si può parlare di canali, all'interno dei quali il prezzo tende a “rimbalzare” tra i massimi ed i minimi relativi.

Figura 8: Trend-line, supporti e resistenze.



Fonte: <https://www.forzaforex.it/le-trend-line>.

- **Segnali di inversione:** il segnale di inversione, indipendentemente che sia crescente o decrescente, si ha quando il prezzo “perfora” la trend-line e si ha quindi un’inversione di tendenza dell’andamento del mercato. Quando succede si deve però verificare che l’inversione sia effettiva, analizzando il periodo temporale nel quale persiste la rottura della trend-line, che è stabilito in base alla volatilità del titolo, e la percentuale di variazione che deve essere significativa altrimenti si incorre nella possibilità di falsi segnali che possono portare a un’operatività sbagliata con conseguenti perdite.
- **Ritracciamenti:** con questo termine si definisce un momentaneo cambio di direzione dei prezzi rispetto alla precedente tendenza. Quindi un trend rialzista vedrà la presenza di un ritracciamento nel caso in cui si avrà una temporanea discesa dei prezzi; avviene quando chi è riuscito ad entrare nel mercato ad un prezzo basso decide di vendere e incassare il proprio guadagno aumentando la pressione delle vendite, quando il prezzo sarà sceso chi era rimasto fuori dal mercato tornerà ad acquistare facendo salire il prezzo fino a quando non si registrerà una nuova rottura a rialzo, ovvero riprendendo la tendenza iniziale. I

ritracciamenti sono difficili da individuare e possono portare ad un'operatività sbagliata se vengono interpretati come segnali di inversione.

### 1.3.1 Pattern grafici

Tra i pattern grafici si distinguono le figure di consolidamento da quelle di inversione: le prime indicano che il trend in corso non ha ancora finito la sua forza e si formano durante le fasi di espansione e flessione dei prezzi, le seconde, invece, si formano nelle fasi di accumulo e distribuzione e indicano che la tendenza è vicina ad una possibile inversione.

Le principali figure di consolidamento sono il “flag” e il “gap di fuga”:

- **Flag:** si ha quando all'interno del trend si forma un canale di dimensioni decisamente inferiori rispetto al trend stesso, ma con direzione opposta ad esso. Si chiamerà “flag rialzista” il pattern che si forma all'interno di un trend rialzista creando, quindi, una momentanea discesa del prezzo, mentre “flag ribassista” sarà quello in cui ci sarà una momentanea risalita del prezzo durante un movimento discendente. La creazione grafica sarà concreta nel momento in cui il trend fuoriesce dal canale riprendendo la direzione originale.
- **Gap di fuga:** si ha un gap quando il minimo raggiunto in un determinato periodo è comunque più alto rispetto al massimo del periodo precedente, o viceversa, quando il massimo raggiunto è più basso del minimo del precedente periodo. Questo tipo di indicazioni possono essere evidenziate solo nei grafici a barre o nei grafici a candela, visivamente formando un “buco” tra le barre o candele. Si forma generalmente quando il mercato è dominato da un'elevata emotività degli operatori.

Figura 9: Gap di fuga.

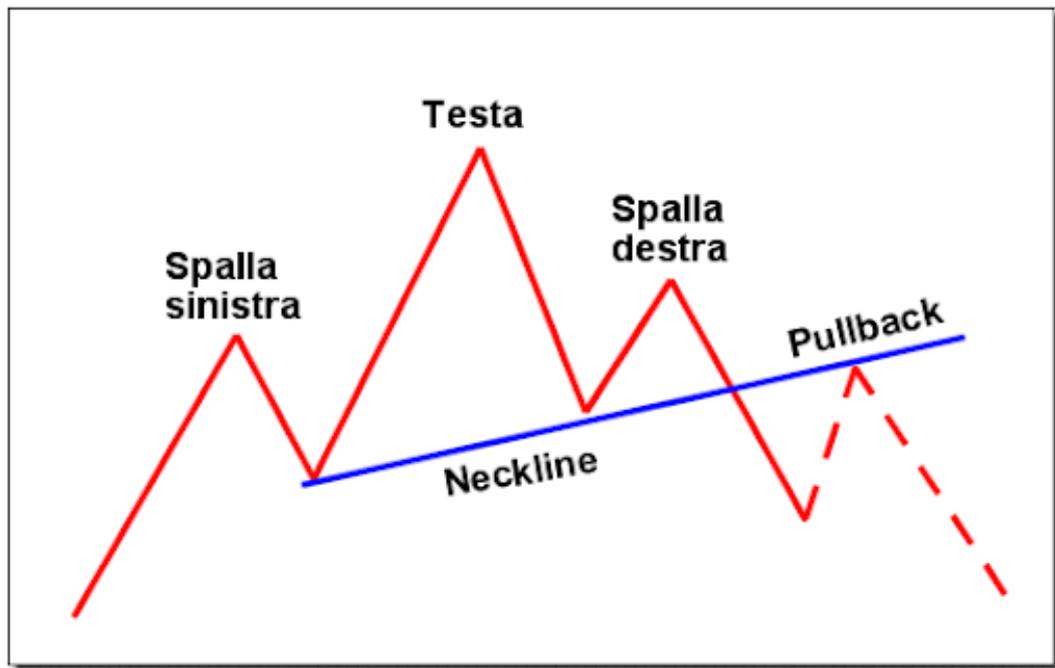


Fonte: <https://investire.biz/academy/analisi-tecnica/gap>.

Per quanto riguarda invece le inversioni del trend le figure principali sono:

- **Testa e spalle:** questa configurazione grafica si forma nel momento in cui un trend rialzista ha iniziato la sua inversione e visivamente è data da un punto di massimo finale, chiamato “testa” che separa altri due punti di massimo inferiori chiamati “spalle”, che non sono necessariamente identici. La conferma avviene quando il trend perfora a ribasso la linea di supporto che si era creata con i punti di minimo relativo tra le due spalle chiamata “neck line” (linea di collo), che ora diventa una possibile indicazione come resistenza dei prezzi e viene definita “pullback”. Nel caso di un trend ribassista si può riscontrare la formazione del modello **testa e spalle rovesciato** nel quale la neck line sarà la linea di resistenza formata tra i due punti di massimo relativi delle spalle. Questo pattern grafico è decisamente attendibile e indica chiari e concreti segnali di inversione della tendenza, dove solitamente più è lungo il suo periodo di formazione maggiore è la durata del trend indicato.

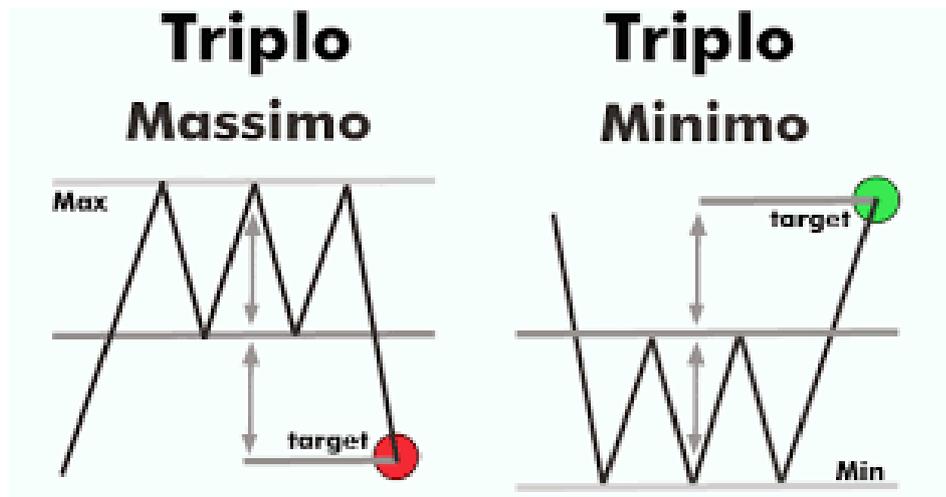
Figura 10: Grafico testa e spalle.



Fonte: <http://www.tradingprofessionale.it>.

- **Doppio o triplo massimo:** è una configurazione di inversione che si ha nel caso in cui il trend formi dei punti di massimo (minimo) che per due o tre volte non superano un determinato livello, formando quindi un'ideale linea di resistenza (supporto) in prossimità di tale prezzo. Anche in questo caso la configurazione è completa solo quando il trend rompe la linea di resistenza (supporto) che si era formata nel trend rialzista (ribassista). In questa configurazione come in quella precedente il verificarsi di queste condizioni deve essere accompagnato da un trend dei volumi crescente che supporta il segnale ed indica la fase di accumulazione del mercato. Graficamente può essere paragonata al pattern che si crea con la figura precedente ma in questo caso i punti raggiunti dalla testa e dalle spalle sono i medesimi.

Figura 11: Grafico triplo massimo e triplo minimo.

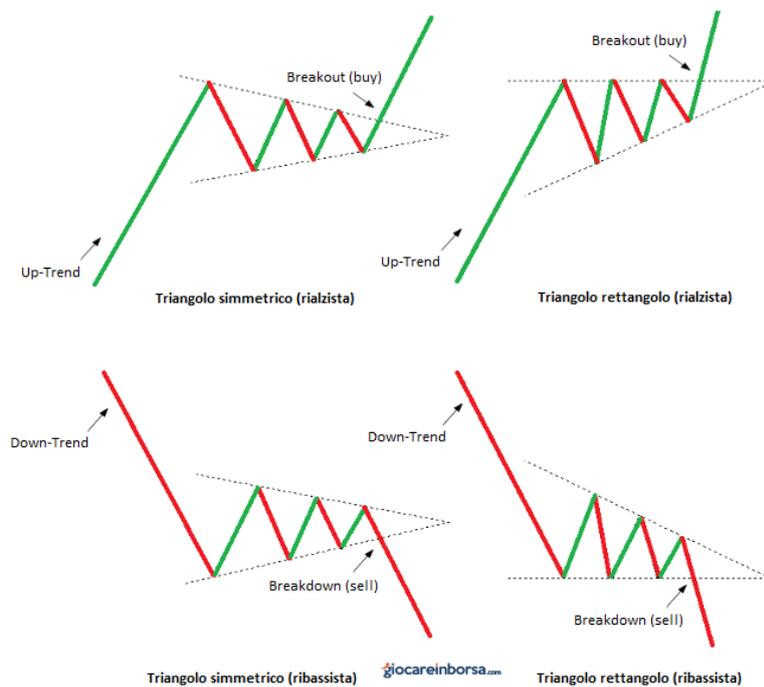


Fonte: <http://tradinghd.blogspot.com>.

Infine, vi è una configurazione particolare che può avere significato ambiguo, chiamata **triangolo**, e può essere di due tipi:

- **Triangolo simmetrico:** si forma quando sul trend è possibile osservare almeno due punti di minimo successivi crescenti ed almeno due punti di massimo successivi decrescenti, in questo modo il supporto e la resistenza che si formano convergono formando un triangolo. Se durante un trend rialzista si forma il triangolo e successivamente rompe la linea di resistenza allora darà un segnale di consolidamento, altrimenti rompendo la linea di supporto darà un segnale di inversione, viceversa, durante un trend ribassista i segnali saranno opposti.
- **Triangolo asimmetrico:** si configura come una forma particolare del precedente, infatti, mantiene le stesse caratteristiche con la peculiarità di avere una delle due linee parallela all'asse delle ascisse. In questo caso è più probabile che la rottura avvenga nel lato orizzontale del triangolo.

Figura 12: Triangoli simmetrici e asimmetrici.



Fonte: <https://www.giocareinborsa.com>.

#### 1.4 Analisi tecnica algoritmica

Questo metodo di analisi si sostanzia nell'utilizzo di indicatori, tramite i quali viene fatta un'analisi statistica al fine di individuare un segnale operativo. Un indicatore tecnico è una costruzione matematica basata sul prezzo, tempo o volume di un titolo ed offre un supporto all'analisi grafica per prevedere la direzione futura delle quotazioni. Va detto che non esiste un indicatore "magico" che permette di ottenere sempre un profitto, ma piuttosto la loro combinazione può fornire un'idea della psicologia del mercato. Si possono distinguere tra gli indicatori *leading*, detti anche oscillatori, e gli indicatori *lagging*. I primi sono costruiti per confermare un nuovo trend o prevederne un'inversione tramite lo storico dei prezzi, i secondi seguono l'andamento del prezzo e sono utili per confermare il trend in corso, infatti vengono anche chiamati indicatori trend-following, risultando poco utili nel caso di una fase laterale del mercato.

Solitamente, la maggior parte degli indicatori può essere suddivisa in quattro categorie principali: gli *indicatori di trend*, gli *indicatori di momentum*, gli *indicatori di volume* e gli *indicatori di volatilità*.

### 1.4.1 Indicatori di trend

Gli indicatori di trend (o di tendenza) interpretano la direzione del mercato e la forza con cui si muove utilizzando le medie mobili e le costruzioni basate su esse, tra cui i più famosi sono il MACD e l'ADX. Quando il prezzo si muove al di sopra della media, potrebbe verificarsi la presenza di un trend rialzista, quando invece i prezzi scendono al di sotto della media, potremmo essere di fronte ad un trend ribassista.

#### *Le medie mobili*

Data la presenza di volatilità all'interno delle serie storiche finanziarie le medie mobili risultano essere un ottimo indicatore per smorzare le eccessive fluttuazioni di mercato restituendo un grafico più lineare e facilmente interpretabile, privato dal cosiddetto "smoothing effect".

La loro costruzione si basa sulla semplice media aritmetica, aggiornata costantemente con i nuovi dati disponibili, eliminando il dato più vecchio utilizzato, tendenzialmente il prezzo di chiusura del titolo, ed inserendo quello più recente. Evidentemente la cosa più importante da scegliere quando si utilizzano le medie mobili è il numero di osservazioni inserite nel calcolo, infatti, la sua reattività è inversamente proporzionale alla quantità di dati inseriti; un numero particolarmente alto smorza la linea di trend in maniera consistente e risulta appiattita molto di più rispetto a quella effettiva allontanando anche i segnali operativi in quanto poco tempestiva a favore di una bassa probabilità di generare falsi segnali. Viceversa, un numero eccessivamente basso rende la media troppo reattiva e non è in grado di togliere le fluttuazioni irrilevanti, creando una linea che segue troppo quella originale, il che porta ad un'operatività eccessiva con conseguenti falsi segnali.

Questi strumenti sono molto utilizzati per la loro semplicità e perché costituiscono un vero e proprio trading system: indipendentemente dal periodo campionario scelto semplicemente si assume che il trend è impostato a rialzo quando i prezzi si muovono al di sopra della propria media mobile, al contrario, a ribasso. Il segnale operativo si crea quando i prezzi incrociano la media mobile, in particolare se i prezzi incrociano a rialzo è un segnale di acquisto, se incrociano a ribasso è un segnale di vendita.

Esistono diversi modelli di costruzione delle medie mobili, i principali sono:

- *Media mobile semplice (SMA)*: è quella di più immediato calcolo, ma risulta essere anche la meno affidabile, basta sommare i prezzi di un numero "n" di giorni e dividere il risultato per il numero di giorni stessi. Il coefficiente di ponderazione

risulta pari a uno e tiene quindi in uguale considerazione le quotazioni più remote e quelle più recenti:

$$SMA_{(t,n)} = \frac{\sum_{i=0}^{n-1} P_{t-i}}{n}$$

dove P corrisponde al prezzo e  $n$  al numero di osservazioni.

- *Media mobile ponderata (WMA)*: è la più semplice evoluzione della precedente e attribuisce un coefficiente di ponderazione diverso in base al posizionamento nel tempo del valore osservato. Generalmente viene dato un peso maggiore alle osservazioni più recenti rispetto a quelle più remote:

$$WMA_{(t,n)} = \frac{\sum_{i=0}^{n-1} w_i P_{t-i}}{\sum_{i=0}^{n-1} w_i}$$

dove  $w_i$  è il peso attribuito allo  $i$ -esimo dato della media.

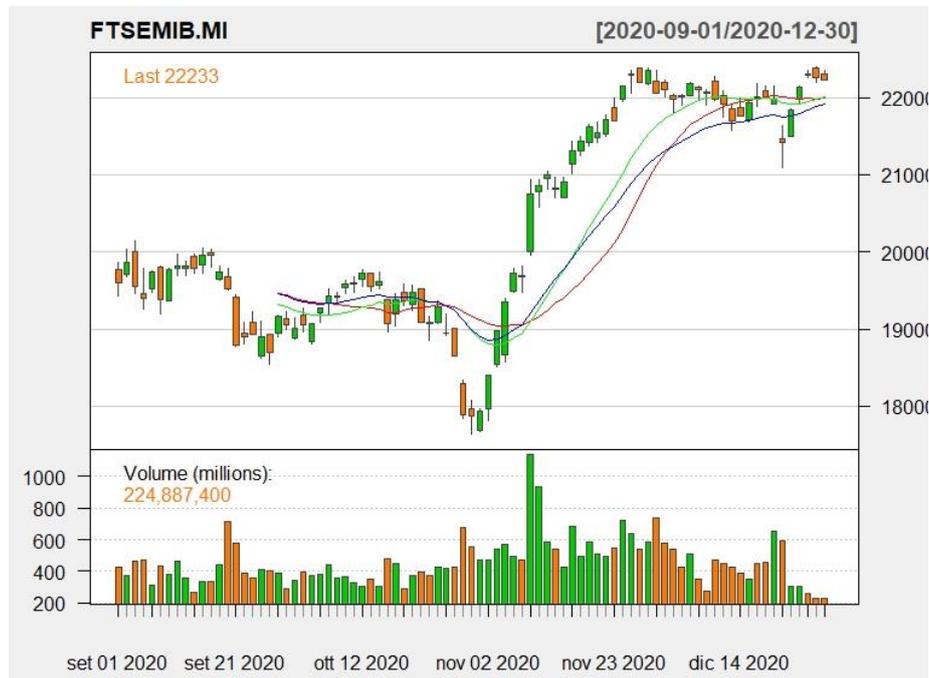
- *Media mobile esponenziale (EMA)*: si tratta di un caso particolare di media mobile ponderata nella quale i pesi vengono dati in funzione dell'ampiezza del periodo preso in considerazione per la media mobile. La formulazione generale di questo tipo di media mobile, una volta individuato il primo valore medio degli elementi in analisi procedendo come per la media semplice, è la seguente:

$$EMA_{(t,n)} = P_t + \frac{1}{a}(EMA_{t-1} - P_t)$$

dove  $a$  corrisponde al rapporto dato tra il peso del dato attuale e quello del dato precedentemente:  $a = w_t/w_{t-1}$  con  $a \geq 1$ .

I vantaggi sono dati dal fatto che il valore è una sintesi di tutti i precedenti valori dando più peso a quelli recenti.

Figura 13: Grafico FTSEMIB con media mobile semplice (rosso), ponderata (verde) ed esponenziale (blu).



Fonte: grafico ottenuto utilizzando R-studio.

Oltre alla operatività descritta precedentemente, ovvero l'incrocio tra il prezzo e la media, è possibile definire un'altra metodologia con la quale cercare segnali nel mercato, ovvero, l'utilizzo combinato di due medie mobili (crossover a due medie), a prescindere che siano semplici, ponderate o esponenziali. Partendo dalla consapevolezza che una media mobile più breve è più reattiva al cambiamento del trend rispetto ad una più lunga, quando tende a seguirlo incrocia, verso l'alto o verso il basso, la media mobile più lunga che per definizione è più lenta e meno reattiva alla variazione di prezzo. Si ha un segnale di acquisto quando la media mobile breve incrocia a rialzo quella più lunga, perché il trend ha avuto la sua inversione e la media mobile più dinamica ha colto questo cambiamento prima rispetto all'altra, viceversa, si ha un segnale di vendita quando la media mobile più reattiva incrocia verso il basso la media mobile più statica. La scelta del dominio delle due medie, come detto precedentemente, risulta quindi fondamentale.

Figura 14: Grafico FTSEMIB con due medie mobili esponenziali rispettivamente a 10 giorni (rossa) e 40 giorni (blu).



Fonte: grafico ottenuto utilizzando R-studio.

### Moving Average Convergence-Divergence (MACD)

Il MACD è un indicatore molto popolare nell'analisi tecnica, fu sviluppato da Gerald Appel attorno al 1970. La sua costruzione è basata su tre diversi strumenti:

- La “*MACD line*”: formata dalla differenza tra una media mobile dei prezzi calcolata su  $n_1$  periodi (solitamente 12 giorni), ed è la più “veloce”, e una media mobile di  $n_2$  periodi (solitamente 26 giorni), la più “lenta”. La formula risulta quindi  $MACD\ Line = (EMA_{12} - EMA_{26})$
- La “*Signal line*”: una media mobile esponenziale calcolata su  $m_1$  periodi (solitamente nove giorni) dei dati che costituiscono la precedente *MACD Line*.
- Il “*MACD histogram*”: rappresenta per ciascun periodo la differenza tra la *MACD Line* e la *Signal Line*.

Questo indicatore per come è costruito si basa interamente sulla convergenza o la divergenza delle due medie esponenziali. Se il mercato presenta un trend crescente la *MACD Line* assume valori positivi crescenti quanto più la media mobile veloce si allontana positivamente da quella lenta, ugualmente, se il mercato presenta un trend ribassista la *MACD Line* assumerà valori negativi quanto più la media mobile veloce si allontana, in questo caso negativamente, da quella lenta. I valori saranno vicini allo zero

quando le due medie tenderanno ad incrociarsi. Inoltre, quando la media mobile veloce si allontana fortemente da quella più lenta (vale a dire quando la MACD Line aumenta), indica che il prezzo del titolo può essere eccessivamente elevato e presto tornerà a livelli più bassi; infatti, rappresenta un ottimo indicatore di ipercomprato/ipervenduto. Sfortunatamente non esistono “soglie standard” di ipercomprato o ipervenduto.

L’indicatore MACD fornisce segnali quando si osserva:

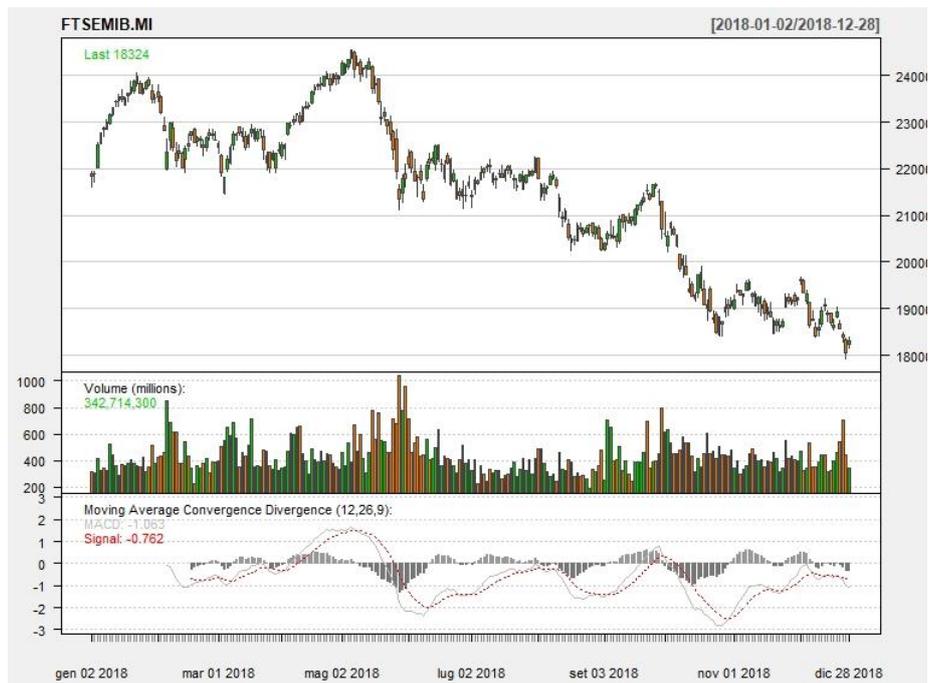
- Crossover della Signal Line
- Crossover della linea zero
- Divergenze

Il primo tipo di segnale è il metodo più utilizzato e si basa sull’incrocio tra la MACD Line e la Signal Line. Si genera un segnale di acquisto (bullish crossover) nel momento in cui la MACD Line incrocia dal basso verso l’alto la Signal Line, mentre è un segnale di vendita (bearish crossover) nel caso l’incrocio sia dall’alto verso il basso. Questa metodologia è caratterizzata da una buona tempestività del segnale a discapito però di un’alta possibilità di creare falsi segnali, per questo l’analista può avvalersi di alcuni filtri che permettano di limitare la possibilità di cadere in errore. Alcuni esempi di cautele possono essere l’applicazione di filtri temporali e/o di significatività o ancora l’utilizzo di medie mobili con più ampi domini ma con conseguente ritardo nella tempestività del segnale. Inoltre, è possibile individuare segnali anche tramite l’osservazione del MACD Histogram, in quanto costruito sulla differenza tra le due linee, in particolare, genera un segnale di acquisto qualora passi da valori negativi a valori positivi e un segnale di vendita in caso contrario.

Il secondo tipo di segnale prevede solo l’utilizzo della MACD Line e non della Signal Line ma il segnale di crossover ha una premessa molto simile a quella precedente. Operativamente si hanno segnali di acquisto nel momento in cui la MACD Line incrocia dal basso verso l’alto la “linea dello zero”, viceversa indica un mercato a ribasso. L’unica differenza rilevante dal metodo precedente è che indica anche la forza del trend: valori prossimi allo zero segnalano un trend con poca forza mentre, quanto più si discostano dallo zero i valori, tanto più il trend sarà rilevante.

Infine, come accade in quasi tutti gli oscillatori, una divergenza tra l’andamento della MACD Line e l’andamento del prezzo potrebbe suggerire una perdita di forza del trend e che quindi il mercato sia vicino ad un’inversione di tendenza.

Figura 15: Grafico FTSEMIB con indicatore MACD.



Fonte: grafico ottenuto con R-studio.

### *Average Directional Movement Index*

L'Average Directional Movement Index (ADX), il Minus Directional Indicator e il Plus Directional Indicator rappresentano un gruppo di indicatori di tendenza sviluppati nel 1978 da J. Welles Wilder. L'ADX può essere applicato a qualsiasi strumento finanziario e permette di individuare la forza del trend tramite la serie storica senza indicarne la direzione futura, per questo motivo il Plus Directional Indicator ( $DI^+$ ) e il Minus Directional Indicator ( $DI^-$ ) vengono aggiunti per completarne l'analisi, infatti, la loro costruzione permette di individuare la direzione del trend, fornendo così, possibili segnali. Congiuntamente, quindi, i tre indicatori permettono di determinare sia la direzione che la forza del trend.

Secondo Wilder è da considerare rilevante un trend quando l'indicatore ADX supera il valore di 25, mentre siamo in assenza di trend, quando l'indicatore si trova sotto il valore di 20. Rimane una zona di neutralità, quella compresa fra 20 e 25. C'è da specificare che ogni mercato ha le sue caratteristiche, per esempio sul mercato italiano si può accettare un valore di ADX inferiore (pari a 20) come soglia critica, superata la quale, il titolo è da considerarsi in trend, in quanto i titoli italiani sono in linea di massima e con le dovute eccezioni, meno volatili di quelli americani.

La costruzione dell'ADX è basata sui valori di  $DI^+$  e  $DI^-$ , che vengono ottenuti tramite i prezzi massimi, minimi e di chiusura di ogni time-frame (normalmente giornaliero). Questi due valori vengono a loro volta calcolati tramite il Plus Directional Movement (+DM), il Minus Directional Movement (-DM) e l'Average True Range (TR).

Il +DM è dato da:

$$+DM_t = \begin{cases} (H_t - H_{t-1}) & \text{se } (H_t - H_{t-1}) > (L_{t-1} - L_t) \wedge (H_t - H_{t-1}) > 0 \\ 0 & \text{altrimenti} \end{cases}$$

Il -DM da:

$$-DM_t = \begin{cases} (L_{t-1} - L_t) & \text{se } (L_{t-1} - L_t) > (H_t - H_{t-1}) \wedge (L_{t-1} - L_t) > 0 \\ 0 & \text{altrimenti} \end{cases}$$

E, infine, l'Average True Range è dato da:

$$ATR_t = \frac{(ATR_{t-1})(n-1) + TR_t}{n}$$

Dove TR:

$$TR_t = \max[(H_t - L_t), |H_t - C_{t-1}|, |L_t - C_{t-1}|]$$

Specifichiamo che:

- $H_t$  sta per il prezzo massimo raggiunto al tempo t
- $L_t$  sta per il prezzo minimo raggiunto al tempo t
- $C_t$  sta per il prezzo di chiusura al tempo t

Dopo aver fissato il numero di periodi per il calcolo, solitamente pari a 14 giorni si procede alla misurazione di  $DI^+$  e  $DI^-$  che vengono semplicemente trovati dividendo i rispettivi +DM e -DM per l'ATR:

$$DI_t^+ = \frac{+DM_t}{ATR_t}$$

$$DI_t^- = \frac{-DM_t}{ATR_t}$$

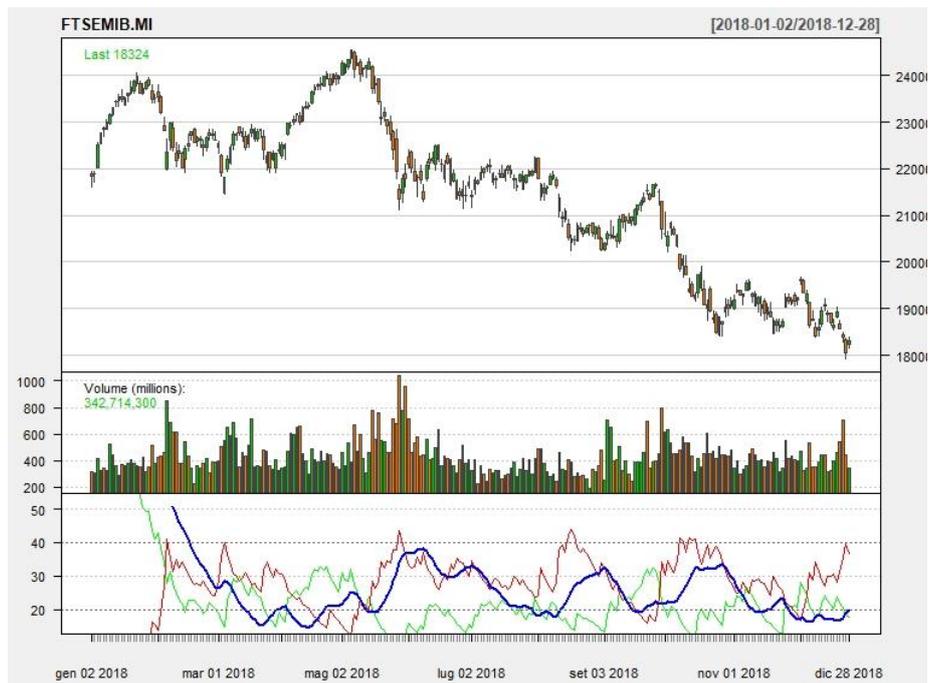
Infine, l'ADX viene calcolato con la seguente espressione:

$$ADX_t = \frac{1}{k} \sum_{i=0}^{k-1} \frac{DI_{t-i}^+ - DI_{t-i}^-}{DI_{t-i}^+ + DI_{t-i}^-} \times 100$$

L'ADX assieme al  $DI^+$  e al  $DI^-$  indicano il momento più adeguato a entrare o uscire dal mercato, in particolare, si genera un segnale operativo con tre specifiche indicazioni:

- *Crossover rule*: si genera un segnale di acquisto quando la linea del  $DI^+$  incrocia dal basso verso l'alto la linea del  $DI^-$ . Viceversa, si genera un segnale di vendita quando  $DI^+$  incrocia dall'alto verso il basso  $DI^-$ .
- *Extreme Point Rule*: indica il momento più adeguato a entrare o uscire dal mercato, dopo che si è registrato un segnale operativo con la regola del Crossover. Tale regola prevede di determinare il massimo o minimo generato nella giornata in cui è avvenuto il segnale e procedere con l'operazione una volta che il prezzo ha superato l'estremo indicato.
- *Turning Point Rule*: individua una possibile inversione di tendenza nel mercato quando la curva ADX scende sotto le curve DI. In questo caso è necessario che la presenza della tendenza sia confermata, ovvero, che il valore dell'ADX sia superiore a 25.

Figura 16: Grafico FTSEMIB con indicatore ADX.



Fonte: grafico ottenuto con R-studio.

#### 1.4.2 Indicatori di momentum

Gli indicatori di momentum analizzano la velocità con la quale si muovono i prezzi per valutare la forza o la debolezza di un trend tramite il suo andamento e le informazioni sugli estremi del prezzo. Questi indicatori sono principalmente oscillatori e possono essere distinti tra due tipologie: quelli che oscillano attorno ad un determinato valore come il *Rate of Change* e quelli che oscillano all'interno di una fascia di valori (Relative Strength Index o lo Stochastic Oscillator e William %R).

##### *Rate of Change (ROC)*

Il ROC è un indicatore di momentum che misura la variazione percentuale di prezzo da un periodo all'altro. Viene calcolato semplicemente confrontando il prezzo attuale con quello del periodo precedente. In questo modo viene generato un indicatore che fluttua attorno allo zero, con valori positivi che indicano una tendenza al rialzo dei prezzi e valori negativi che indicano una tendenza al ribasso dei prezzi. In formule:

$$ROC_t = \frac{P_t - P_{t-n}}{P_{t-n}}$$

dove  $P_t$  indica il prezzo al tempo  $t$ .

Il segnale operativo è dato principalmente quando l'indicatore incrocia la linea dello zero, rispettivamente come segnale d'acquisto se assume valore positivo e segnale di vendita in caso assuma valori negativi.

Figura 17: Grafico FTSEMIB con indicatore ROC.

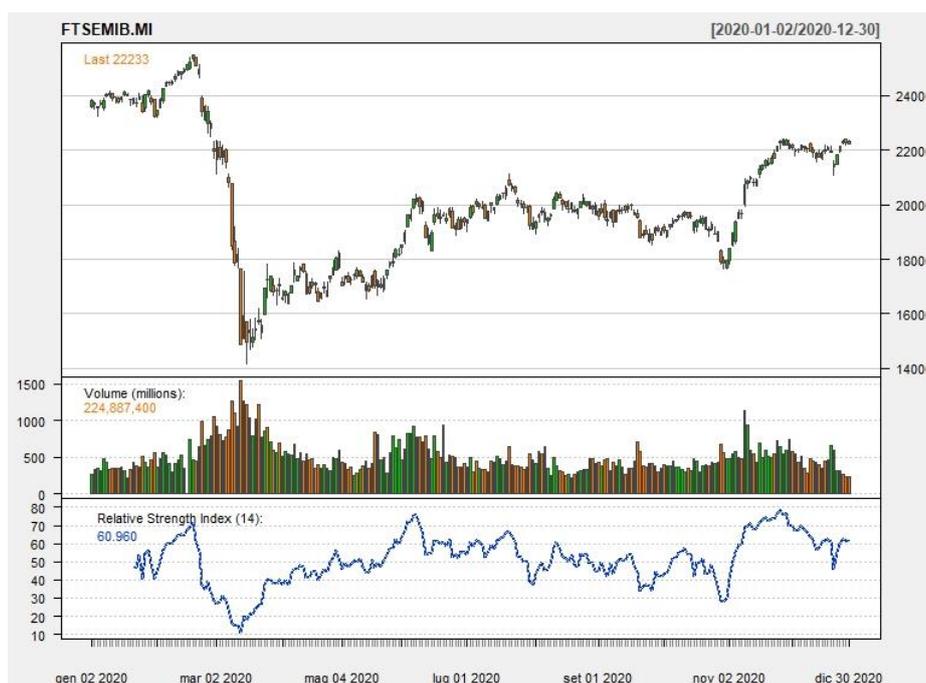


Fonte: grafico ottenuto con R-studio.

### *Relative Strength Index (RSI)*

Il Relative Strength Index è un indicatore di momentum sviluppato anch'esso da J. Welles Wilder e misura la forza e la velocità dei movimenti di un prezzo. Il valore dell'indice è compreso tra 0 e 100 e generalmente indica una situazione di ipercomprato quando supera il valore di 70, mentre se scende sotto i 30 definisce una situazione di ipervenduto.

Figura 18: Grafico FTSEMIB con indicatore RSI.



Fonte: grafico ottenuto con R-studio.

La formula per il calcolo di questo indicatore è la seguente:

$$RSI_t = 100 - \frac{100}{1 + RS_t}$$

dove

$$RS_t = \frac{MI_t}{MD_t}$$

dove a loro volta  $MI_t$  e  $MD_t$  sono rispettivamente una media degli incrementi e decrementi di prezzo al tempo  $t$ . La media viene calcolata solitamente in un arco temporale di 14 giorni. Mentre i primi valori di  $MI_1$  e  $MD_1$  sono calcolati tramite una semplice media aritmetica degli incrementi ( $\Delta^+$ ) e decrementi ( $\Delta^-$ ) del prezzo i valori successivi sono trovati tramite una metodologia diversa, in formule:

$$MI_1 = \frac{\sum_{i=1}^n \Delta_i^+}{n}$$

$$MD_1 = \frac{\sum_{i=1}^n \Delta_i^-}{n}$$

$$MI_t = \frac{MI_{t-1} \cdot (n - 1) + \Delta_t^+}{n}$$

$$MD_t = \frac{MD_{t-1} \cdot (n - 1) + \Delta_t^-}{n}$$

Come detto può assumere valori compresi tra 0, nel caso in cui nel periodo considerato si osservino solo decrementi ( $MI_t = 0$ ), e 100, nel caso contrario in cui si osservino solo incrementi di prezzo ( $MD_t = 0$ ).

In base ai parametri di ipercomprato e ipervenduto si generano i segnali operativi:

- Quando il valore supera la soglia di 70 si presenta uno stato di ipercomprato con conseguente segnale di vendita.
- Movimenti dell'RSI compresi tra 70 e 30 indicano uno stato di regolarità delle oscillazioni dei prezzi, indicando quindi di non procedere con alcuna operazione.
- Se l'RSI scende sotto la soglia di 30 allora si conferma uno stato di ipervenduto con successivo segnale di acquisto.

Gli ultimi due indicatori di momentum citati, l'oscillatore stocastico e il William %R, si caratterizzano per un utilizzo ed una finalità simili; perciò, analizzeremo in maniera completa solo la costruzione del primo in quanto uno è la relativa rappresentazione negativa dell'altro.

#### *Oscillatore stocastico*

L'oscillatore stocastico è uno degli indicatori più usati in analisi tecnica, fu introdotto da George C. Lane nel 1970 il quale notò che i prezzi di chiusura vicini ai massimi e minimi di giornata risultano essere tipici segnali, rispettivamente, di trend al rialzo e trend al ribasso. Una fase di rialzo è quindi caratterizzata da prezzi di chiusura che tendono a collocarsi in prossimità dei massimi registrati nel periodo di riferimento e man mano che si giunge al termine della fase si allontanano, viceversa, una fase di ribasso è caratterizzata da prezzi di chiusura che rimangono in prossimità dei minimi. L'indicatore confronta il prezzo di chiusura, il massimo e il minimo al tempo t e può essere distinto in tre categorie:

- L'oscillatore stocastico veloce (fast stochastic oscillator)

- L'oscillatore stocastico lento (slow stochastic oscillator)
- L'oscillatore stocastico completo (full stochastic oscillator)

Il primo, è costituito da due curve: la “%K Line” (o fast line), che indica la posizione del prezzo di chiusura all'interno del periodo misurato, e la “%D Line” (o slow line) che non è altro che una media mobile a tre giorni della %K Line:

$$\%K_{t,n} = 100 \cdot \frac{(C_t - m_{t,n})}{(M_{t,n} - m_{t,n})}$$

$$\%D = SMA_3(\%K)$$

dove  $C_t$  è il prezzo di chiusura al tempo  $t$ ,  $m_{t,n}$  e  $M_{t,n}$  sono rispettivamente il prezzo minimo e massimo degli ultimi  $n$  giorni rispetto all'istante  $t$ , solitamente  $n$  è posto pari a 14. Il valore dell'indice è espresso in percentuale e indica la posizione del prezzo rispetto al range delle quotazioni degli ultimi  $n$  giorni.

Il secondo, quello lento (indicato con il pedice  $s$ ), è creato ed utilizzato in conseguenza del fatto che quello veloce risulta essere troppo reattivo e di difficile interpretazione in quanto porta spesso a falsi segnali.  $\%K_s$  non è altro che la media mobile a tre periodi della  $\%D$  dell'oscillatore stocastico veloce e la linea  $\%D_s$  è la media mobile sempre a tre periodi della  $\%K_s$ .

$$\%K_s = SMA_3(\%D)$$

$$\%D_s = SMA_3(\%K_s)$$

Infine, la versione “completa” ed anche più utilizzata è una generalizzazione della versione lenta dell'oscillatore, nella quale i parametri non sono prestabiliti ma vengono scelti dall'analista:

$$\%K_{full} = SMA_m(\%K)$$

$$\%D_{full} = SMA_m(\%K_{full})$$

Come nel caso dell'RSI anche l'oscillatore stocastico può assumere valori compresi tra 0 e 100 con la medesima divisione di ipercomprato e ipervenduto.

I segnali operativi anche per questo indicatore possono derivare sia per l'intersezione tra le due linee lenta e veloce sia per una divergenza con l'andamento dei prezzi. In particolare, si genera un segnale quando la linea %K si trova all'interno delle fasce di ipercomprato o ipervenduto e incrocia al rialzo o al ribasso %D indicando una prossima inversione del trend.

#### 1.4.3 Indicatori di volume

Il volume rappresenta l'ammontare di scambi che si verifica durante un dato intervallo di tempo indipendentemente dal prezzo; gli indicatori di volume basandosi su esso misurano il grado di partecipazione nel mercato e il suo andamento in relazione al volume stesso. Un rialzo improvviso dei prezzi non supportato da un aumento dei volumi può significare che il prezzo sia inflazionato e che presto torni ai suoi livelli iniziali, in quanto non significativo, e la sua crescita sia dovuta solo a fini speculativi. Tra i principali indicatori di volume ci sono l'OBV (On Balance Volume) e l'MFI (Money Flow Index).

##### On Balance Volume

L'OBV è stato sviluppato da Joe E. Granville nel 1963 che lo presenta per la prima volta nel suo elaborato "Granville's New Key to Stock Markets Profit". Fu uno dei primi indicatori tecnici che forniva le informazioni per eventuali cambiamenti dell'andamento del mercato grazie ai flussi di volumi positivi e negativi. L'indicatore mette in relazione le variazioni di prezzo con l'attività di scambio, attribuendo un valore positivo o negativo ai volumi registrati per ogni periodo considerato a seconda che il prezzo di chiusura del mercato sia rispettivamente superiore o inferiore a quello fissato nella seduta immediatamente precedente. In pratica distingue i flussi delle giornate che hanno registrato una variazione dei prezzi positivi da quelli negativi e ne aggiunge o toglie i rispettivi volumi all'indice. Quindi l'indicatore OBV si presenta come una sorta di somma cumulata dei volumi il cui segno dipende dal segno della variazione dei prezzi.

Si può sintetizzare nel seguente modo:

$$OBV_t = OBV_{t-1} + \begin{cases} V_t & \text{se } C_t > C_{t-1} \\ 0 & \text{se } C_t = C_{t-1} \\ -V_t & \text{se } C_t < C_{t-1} \end{cases}$$

Dove  $C_t$  = prezzo di chiusura al tempo  $t$ ,  $C_{t-1}$  = prezzo di chiusura del giorno precedente,  $V_t$  = volume al tempo  $t$ .

Analizzando il comportamento dell'indicatore OBV, Granville notò che esso spesso anticipava i movimenti di prezzo e ciò che è importante non è necessariamente il valore ma quanto più le caratteristiche della linea. Si può riassumere con una tabella:

Tabella 1: Interpretazione segnali OBV.

	Prezzi in espansione	Prezzi in congestione	Prezzi in flessione
OBV crescente	Uptrend in accelerazione	Fase di accumulazione	Downtrend in indebolimento
OBV stabile	Uptrend in rallentamento	Fase stagnante	Downtrend in rallentamento
OBV decrescente	Uptrend in indebolimento	Fase di distribuzione	Downtrend in accelerazione

Questi segnali sono basati sulla teoria che l'andamento dei volumi anticipi il movimento dei prezzi.

Sono state mosse alcune critiche riguardo questo indicatore per quanto riguarda la sua modalità di costruzione, la principale è che somma e sottrae l'intero ammontare di volumi anche a fronte di piccole variazioni del prezzo di chiusura, ma questo problema può essere risolto ponendo un filtro di significatività, in modo tale che oscillazioni di prezzo minime non vengano rilevate e quindi calcolate.

Figura 19: Grafico FTSEMIB con indicatore OBV.



Fonte: grafico ottenuto con R-studio.

### Money Flow Index

Questo indicatore come il precedente utilizza sia i prezzi che i volumi per l'individuazione di zone di ipercomprato e ipervenduto e misura la forza del flusso di denaro, ovvero la quantità di denaro che affluisce o defluisce in uno strumento. È simile all'RSI visto precedentemente con l'unica differenza che questo incorpora al suo interno anche i volumi, infatti è anche chiamato "Volume-weighted RSI", ovvero RSI pesato per i volumi.

La costruzione dell'indice si articola in diversi passaggi, il primo prevede di calcolare il "Typical price" ( $Tp_t$ ) che corrisponde alla media aritmetica semplice del prezzo massimo, minimo e di chiusura del giorno:

$$Tp_t = \frac{H_t + L_t + C_t}{3}$$

Successivamente si calcola il flusso di denaro detto "Money Flow" ( $MF_t$ ) moltiplicando il "Typical price" per il volume di scambio ( $V_t$ ) del giorno:

$$MF_t = (Tp_t) \cdot V_t$$

A questo punto si confrontano i due “Typical price” relativi alle due sedute consecutive e si distingue:

- Se  $Tp_t > Tp_{t-1}$  allora  $MF_t$  è  $MF_t^+$  (Positive Money Flow)
- Se  $Tp_t < Tp_{t-1}$  allora  $MF_t$  è  $MF_t^-$  (Negative Money Flow)
- Se  $Tp_t = Tp_{t-1}$  allora  $MF_t = 0$

Ora, si calcola il rapporto tra le somme dei Money Flow positivi ( $MF^+$ ) e le somme dei Money Flow negativi ( $MF^-$ ) del periodo considerato, solitamente 14 giorni. Tale rapporto è chiamato “Money Flow Ratio”:

$$MFR = \frac{\sum_{i=0}^{k-1} MF_{t-i}^+}{\sum_{i=0}^{k-1} MF_{t-i}^-}$$

Infine, si calcola il valore dell’indice con la seguente formula:

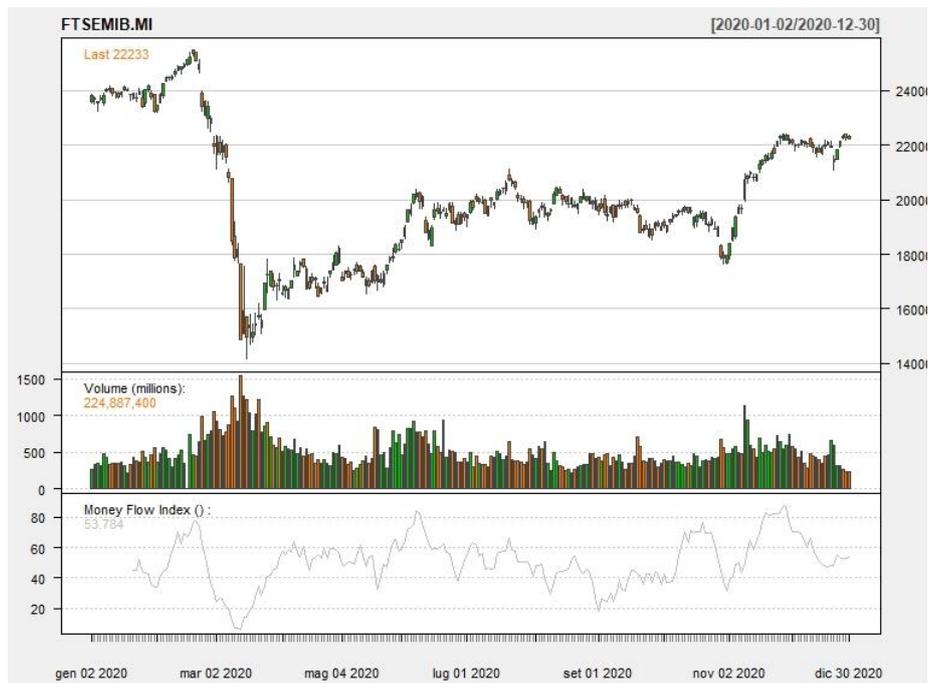
$$MFI_t = 100 - \frac{100}{1 + MFR_t}$$

Questo indicatore può assumere valori nell’intervallo (0, 100) e si prestabilisce l’area di ipercomprato superiore ad 80 e di ipervenduto inferiore a 20. Secondo Gene Quong e Avrum Soudack, i due creatori, ha diversi tipi di segnale operativo: il più evidente lo si ha nel momento in cui arriva in prossimità dei limiti, superiore e inferiore, generando, rispettivamente, opportunità di vendita quando si trova nella zona di ipercomprato, e di acquisto quando si trova nella zona di ipervenduto.

Il secondo segnale si ottiene dall’osservazione di divergenze tra l’andamento della curva dei prezzi e di quella dell’indicatore: avremo un segnale di imminente inversione rialzista in presenza di trend dei prezzi decrescente e trend dell’indicatore crescente e un segnale di inversione ribassista in presenza di trend dei prezzi crescente e trend dell’indicatore calante poiché la forza del segnale si sta indebolendo.

In conclusione, il MFI è un buon indicatore per segnali di entrata e di uscita dal mercato, ma non rappresenta il trend di mercato e quindi risulta opportuno abbinarlo ad altri indicatori.

Figura 20: Grafico FTSEMIB con indicatore MFI.



Fonte: grafico ottenuto con R-studio.

#### 1.4.4 Indicatori di volatilità

La volatilità è la misura della variazione percentuale del prezzo di un titolo in relazione al tempo. Gli indicatori di volatilità misurano, quindi, la velocità con cui si muove il prezzo a prescindere dalla direzione presa dal movimento stesso. Il più famoso indicatore di volatilità sono le bande di Bollinger.

##### *Le Bande di Bollinger*

Le bande di Bollinger sono state pensate da John Bollinger negli anni 90 che le ha poi formalizzate nel 2002 nel suo libro “Bollinger on Bollinger Bands”. L’indicatore serve a valutare la relazione che intercorre tra i livelli di un prezzo del titolo e la sua volatilità su un periodo di tempo predeterminato. Per formare l’indicatore vengono tracciate tre linee: una linea centrale (*CL*), una banda inferiore (*LB*) ed una superiore (*UB*).

Tutte e tre le linee sono calcolate a partire dal “Typical Price” (*TP*):

$$TP_t = \frac{H_t + L_t + C_t}{3}$$

Dove  $H_t, L_t$  e  $C_t$  sono rispettivamente il prezzo massimo, minimo e di chiusura al tempo  $t$ . Spesso vengono calcolate usando solo i prezzi di chiusura e non il “Typical price”.

La linea centrale è calcolata come una media mobile semplice del  $TP_t$  su  $n$  periodi, solitamente  $n = 20$ . Si preferisce la media semplice a quella esponenziale per garantire una coerenza tra il calcolo della media mobile e della deviazione standard mobile utilizzata per le bande:

$$CL_t = \frac{1}{n} \sum_{i=0}^{n-1} TP_{t-i}$$

La banda superiore è data dalla linea centrale alla quale va sommata la deviazione standard del prezzo calcolata su  $n$  periodi e moltiplicata per un moltiplicatore  $k$ , tipicamente posto pari a 2:

$$UB_t = CL_t + k\sigma_{TP,t}$$

La banda inferiore, allo stesso modo, viene calcolata prendendo la linea centrale e sottraendo la deviazione standard moltiplicata per il moltiplicatore:

$$LB_t = CL_t - k\sigma_{TP,t}$$

Dove la deviazione standard  $\sigma_{TP,t}$  è:

$$\sigma_{TP,t} = \frac{1}{n} \sqrt{\sum_{i=0}^{n-1} (TP_{t-i} - CL_t)^2}$$

Bollinger suggerisce di utilizzare un moltiplicatore della deviazione standard  $k > 2$  nel caso in cui vengano aumentati gli  $n$  periodi utilizzati per il calcolo della media.

L’ampiezza della banda ( $BW_t$ ) è data da:

$$BW_t = UB_t - LB_t = 2k\sigma_{TP,t}$$

Ossia la differenza tra la banda superiore e la banda inferiore.

La banda superiore e la banda inferiore funzionano come aree di supporto e di resistenza dinamica, in altre parole il loro superamento indica fasi di ipercomprato e ipervenduto. Inoltre, quando la distanza tra le due bande è ampia significa che il periodo è caratterizzato da elevata volatilità che potrebbe portare all'esaurimento della tendenza in atto. Viceversa, nel caso in cui le bande si contraggono, il mercato si trova in un periodo caratterizzato da bassa volatilità che potrebbe implicare l'avvio di un nuovo trend.

In particolare, normalmente, i prezzi del titolo sono compresi tra le due bande, un buon segnale operativo potrebbe essere quando i prezzi fuoriescono: quando il prezzo raggiunge la banda superiore l'indicatore fornisce un segnale di vendita, viceversa, quando il prezzo raggiunge la banda inferiore indica un segnale di acquisto.

Ulteriori contributi all'individuazione dei segnali operativi legati alle Bande di Bollinger ci derivano dagli studi di Arthur Merrill, che identifica un segnale di acquisto nel caso in cui si osservi la formazione di due punti di minimo relativo successivi e decrescenti ma che non violano la banda inferiore, opposto segnale nel caso in cui vi sia la formazione di due punti di massimo relativo successivi e crescenti che non escono dalla banda superiore.

Figura 21: Grafico FTSEMIB con Bande di Bollinger.



Fonte: grafico ottenuto con R-studio.



## Capitolo II: Le reti neurali

### 2.1 Cenni storici

Alcuni problemi, come per esempio il riconoscimento dei volti, sono molto semplici per un essere umano, ma diventano estremamente complessi quando devono essere risolti da un elaboratore. Da qui nasce l'idea di approcciare i problemi cercando di imitare i meccanismi degli esseri viventi più evoluti.

I primi studi inerenti a questo ambito, anche se ancora ben distanti dagli attuali sviluppi, appaiono verso il 1940, con la nascita della cibernetica, che viene definita come “scienza che studia i processi intelligenti”, fondata poi da Robert Wiener nel 1947.

I primi a parlare di neuroni artificiali sono stati il neurofisiologo Warren Sturgis McCulloch e il matematico Walter Pitts con la pubblicazione nel 1943 di "*A logical calculus of the ideas immanent in nervous activity*". In questa pubblicazione venivano presentate delle unità, i neuroni artificiali, come dei modelli di neuroni biologici facenti parte di reti caratterizzate da un neurone di entrata e un neurone di uscita. Attraverso questo modello, identificato come “combinatore lineare a soglia”, in cui i dati dovevano essere binari, si era in grado di calcolare delle funzioni AND e OR.

Nel 1947, Donald Hebb ebbe l'idea che i neuroni nel cervello apprendono attraverso la modifica della forza delle connessioni tra di essi. Questo processo si chiama iper-apprendimento, cioè dove due neuroni che si attivano insieme avranno la loro connessione rafforzata, mentre due neuroni che non si attivano insieme l'avranno indebolita.

Nel 1949, Norbert Wiener propose i sistemi cibernetici, cioè sistemi che, avendo sensori e attuatori, creano un ciclo di feedback (retroazione) e un sistema auto-regolatore.

Successivamente nel 1957, Frank Rosenblatt del Cornell Aeronautical Laboratory sviluppa l'idea del cosiddetto apprendimento guidato: il perceptrone (perceptron), un'entità caratterizzata da un layer d'ingresso ed uno di uscita, la cui regola di apprendimento si basa sulla minimizzazione dell'errore. Il sistema confronta l'uscita effettiva prodotta dalla rete con quella prevista e, di conseguenza, aggiorna i pesi delle varie connessioni. Il Perceptron è una rete neurale costituita da dispositivi logici in grado di risolvere semplici problemi di riconoscimento di forme. Si iniziano quindi a vedere le vere potenzialità del sistema in grado di adattarsi per apprendimento al contesto.

Negli anni '60 la ricerca si spense gradualmente per 2 ragioni principali:

- Il fatto che i ricercatori avevano usato neuroni che erano binari; tuttavia, per funzionare, la retropropagazione ha bisogno di funzioni di attivazione che sono continue. In quel momento, i ricercatori non ebbero l'idea di usare neuroni continui, e non pensarono che si potesse addestrare i neuroni con i gradienti, poiché i neuroni binari non sono differenziabili.
- Usando neuroni continui, si sarebbe dovuto moltiplicare l'attivazione di un neurone con un peso per contribuire alla somma pesata. Però, prima degli anni '80, la moltiplicazione di due numeri, specialmente se numeri in virgola mobile, era particolarmente lenta. Questo portò un altro punto a favore all'esclusione dei neuroni continui.

Nel 1986, David E. Rumelhart, G. Hinton e R. J. Williams creano l'algoritmo di addestramento di Backpropagation, il quale prevede la correzione dei pesi tra i nodi in base alla correttezza o meno del risultato. In questo contesto vengono introdotti i livelli intermedi tra i neuroni di ingresso e di uscita (hidden layer).

È dalla fine degli anni '80 che avviene un'altra grande accelerazione con l'arrivo sul mercato di nuovi potenti processori e avanzati chip in grado di supportare applicazioni intensive come quelle delle analisi e delle simulazioni; da qui in poi il percorso di avanzamento tecnologico dell'hardware non si è più arrestato.

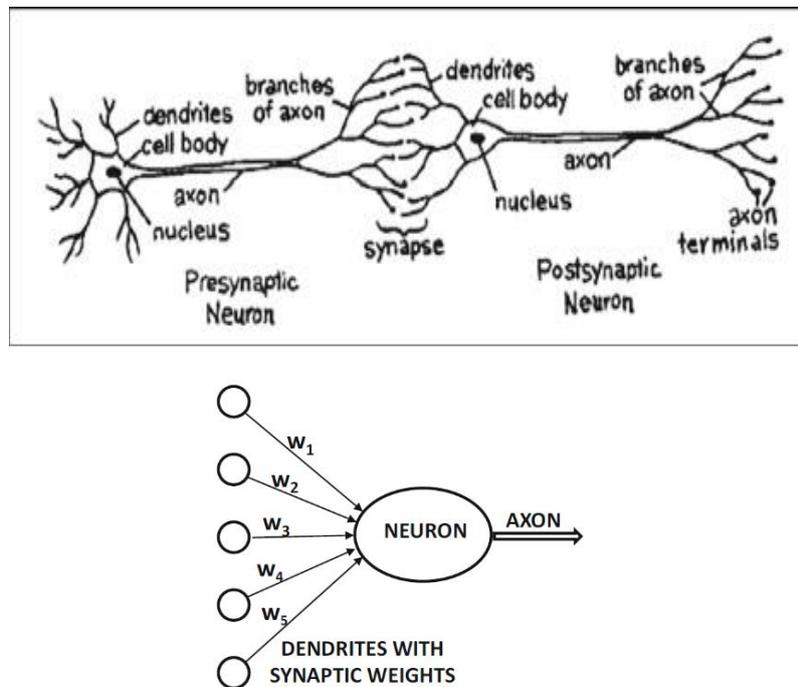
## 2.2 Che cos'è una rete neurale

Il “machine learning” è la branca dell'intelligenza artificiale che sviluppa la capacità delle “macchine” (i sistemi informatici) di imparare qualcosa senza che siano date loro esplicitamente le indicazioni per farlo. In particolare, ci si riferisce ad algoritmi che sviluppano autonomamente le loro conoscenze grazie ai pattern di dati ricevuti.

Le reti neurali artificiali sono tecniche di “machine learning”, o in italiano anche noto come apprendimento automatico, che simulano il meccanismo di apprendimento dei sistemi nervosi biologici. Il sistema nervoso umano contiene cellule, chiamate *neuroni*. I neuroni sono connessi tra loro tramite l'uso dell'*assone*, che è la fibra principale dalla quale escono gli output, e i *dendriti*, composti da fibre minori che raccolgono i segnali di input, e le regioni di collegamento tra gli assoni e i dendriti sono chiamate *sinapsi*. In relazione agli stimoli esterni l'intensità del segnale delle connessioni sinaptiche cambia,

questo cambiamento è il modo in cui l'apprendimento avviene negli organismi viventi. Questo meccanismo biologico è simulato nelle reti neurali artificiali che contengono unità di calcolo che svolgono il ruolo dei neuroni. Le unità di calcolo sono collegate tra loro tramite pesi che svolgono lo stesso ruolo della forza del segnale nelle connessioni sinaptiche biologiche. Ogni input in un neurone viene ridimensionato con un peso, che influisce sulla funzione calcolata in quell'unità. Una rete neurale calcola una funzione degli input propagando i valori calcolati dai neuroni di input ai neuroni di output e utilizzando i pesi come parametri intermedi. L'apprendimento avviene modificando i pesi che connettono i neuroni. Da qui in avanti utilizzeremo il termine "reti neurali" per riferirci a quelle artificiali piuttosto che a quelle biologiche.

Figura 22: Rete neurale biologica e artificiale.



Fonte: Neural network and deep learning. A textbook, Springer, 2018.

### 2.3 Struttura di una rete neurale

Come detto, queste unità, note anche come Perceptron, possono ricevere informazioni dall'ambiente, emettere risposte o comunicare solo tra loro all'interno della rete, in uno o più strati (layers); rispettivamente possono essere definite unità di ingresso (input), unità di uscita (output) e unità nascoste (hidden). Possono trovarsi in due stati: attivi o a riposo.

Ciascuna unità svolge un'operazione semplice, si attiva se la somma pesata dei segnali in ingresso supera un certo valore di attivazione e a questo punto emette, a sua volta, un segnale che viene trasmesso alle altre unità a cui è connessa. Quando, invece, il risultato non supera la soglia il neurone rimane in uno stato di riposo. Ciascun punto di connessione agisce come un filtro che trasforma il messaggio ricevuto in un segnale inibitorio o eccitatorio aumentando o diminuendo l'intensità relativa a seconda delle proprie caratteristiche. In base al modo nel quale il segnale in uscita agisce, cambia anche il modello matematico adottato dal neurone.

Benché le unità di calcolo più elementari nella rete neurale siano ispirate da algoritmi di apprendimento automatico tradizionali come la regressione dei minimi quadrati, la caratteristica distintiva delle reti neurali artificiali consiste proprio nella capacità di apprendere dall'esperienza, caratteristica che non risulta invece presente se si prendono in considerazione altri modelli statistici; la rete neurale artificiale, infatti, una volta fornita di una serie di input e target, procede all'identificazione di quelle relazioni e quei legami che intercorrono tra i primi e che concorrono a formare i secondi. Una volta individuate queste relazioni la rete continua con un processo di perfezionamento tramite addestramento.

Oggi le reti neurali hanno un ampio utilizzo in molteplici settori, possono essere utilizzate per il controllo del traffico, per il riconoscimento e l'elaborazione delle immagini, per la diagnosi medica, il controllo qualità a livello industriale, ecc.

In ambito finanziario la loro applicazione verte a risolvere problemi di previsione, classificazione, analisi del rischio, controllo e ottimizzazione dei processi, gestione del portafoglio, stima delle curve di rendimento, valutazione di titoli e come analizzato in questa tesi, strategie di trading.

Dato l'enorme utilizzo delle reti neurali, non esiste un vero e proprio sistema di classificazione, i criteri maggiormente utilizzati sono:

- Tipo di apprendimento
- Tipo di utilizzo
- Algoritmo di apprendimento
- Architettura dei collegamenti

### 2.3.1 Tipo di apprendimento

Il criterio più significativo è quello relativo al tipo di apprendimento che a sua volta è diviso in:

- **Supervisionato** (supervised learning): nel caso in cui si dispongono un insieme di dati per l'addestramento (input) con i relativi target (output) corrispondenti; in tal modo la rete, addestrata mediante un opportuno algoritmo (solitamente il backpropagation che è appunto un algoritmo d'apprendimento supervisionato), può esplorare l'insieme dei valori possibili che i pesi possono assumere, in modo tale da minimizzare la funzione di perdita (loss function) al fine di ottimizzare l'obiettivo. Infatti, se l'addestramento ha successo la rete impara a riconoscere la relazione incognita che lega gli input con gli output ed è quindi in grado di fare previsioni laddove l'output non è nota a priori. Quando gli output sono corretti, non si fa niente, quando sono sbagliati, si va a sistemare i parametri della macchina, correggendo l'output verso quello desiderato. Il trucco qua è come capire di quanto correggere il valore e quale direzione usare per modificare il parametro. In altre parole, l'obiettivo finale è la previsione del valore dell'output per ogni valore di input, basandosi soltanto su un numero limitato di esempi di corrispondenza.
- **Per rinforzo** (reinforcement learning): è utilizzato nei casi in cui non è possibile specificare pattern di input-output come per i sistemi di apprendimento supervisionato. Viene fornito un rinforzo al sistema, il quale lo interpreta come un segnale positivo/negativo sul suo comportamento e aggiusta i parametri di conseguenza. In sostanza, un determinato algoritmo si prefigge lo scopo di individuare un procedimento a partire dall'osservazione dell'ambiente esterno; ogni azione ha un impatto sull'ambiente esterno che produce una retroazione che guida l'algoritmo nel processo di apprendimento. In questo contesto l'operatore intraprende una serie di azioni alle quali l'ambiente fornisce in risposta un segnale positivo o negativo, a seconda del caso. In questo modo si tenta di massimizzare gli incentivi positivi.
- **Non supervisionato** (unsupervised learning): è basato su algoritmi di addestramento che modificano i pesi facendo esclusivamente riferimento ad un insieme di dati che considera solo le variabili d'ingresso senza fornire nessun

target corrispondente con cui fare un confronto. Tali algoritmi raggruppano i dati d'ingresso con il fine di individuare delle configurazioni all'interno dei dati stessi che danno vita a risposte non prefissate ma che possiedono un significato, utilizzando metodi statistici.

- **Semi-supervisionato:** è un modello “ibrido” dove al computer viene fornito un set di dati incompleti per l'apprendimento; alcuni di questi input sono “dotati” dei rispettivi esempi di output (come nell'apprendimento supervisionato), altri invece ne sono privi (come nell'apprendimento non supervisionato). L'obiettivo, di fondo, è sempre lo stesso: identificare regole e funzioni per la risoluzione dei problemi, nonché modelli e strutture di dati utili a raggiungere determinati obiettivi.

### 2.3.2 Tipo di utilizzo

Una seconda classificazione, la meno rilevante, può essere fatta in base al tipo di utilizzo, nel quale si possono distinguere tre categorie:

- **Memorie associative:** possono apprendere associazioni tra patterns in modo tale che la presentazione di un pattern A dia come risultato un pattern B anche se il primo è parziale o incompleto.
- **Simulatori di funzioni automatiche:** sono in grado di comprendere la funzione che lega output con input in base a degli esempi forniti in fase di apprendimento. Successivamente, gli output possono essere forniti anche per input non usati in fasi di addestramento. Ne consegue una capacità della rete di interpolazione ed estrapolazione sui dati del training set. Tale capacità risulta utile per la previsione di fenomeni dei quali non sia chiaro matematicamente il legame tra input e output. La rete si comporta come una “black box”, poiché non svela in termini “leggibili” la funzione di trasferimento che è contenuta al suo interno. Questa categoria comprende anche la rete più utilizzata ovvero la rete backpropagation.
- **Classificatori:** in questo caso è possibile classificare i dati in base a caratteristiche simili. Questa categoria comprende l'apprendimento non supervisionato.

### 2.3.3 Algoritmo di apprendimento

Le diverse possibilità di configurazione sono molteplici per questo la scelta corretta e più ottimale del tipo di struttura è essenziale per affrontare al meglio il problema, tenendo sempre a mente che non c'è un modello migliore rispetto ad un altro ma piuttosto migliore nel fornire risposte corrette allo specifico problema. L'algoritmo di apprendimento può dunque assumere strutture completamente diverse l'una dall'altra in funzione dell'obiettivo dell'applicazione.

#### *Il Perceptron*

La rete più semplice è formata da un solo neurone, il Perceptron, con  $n$  input e un solo output. Ogni neurone riceve in input  $n$  segnali, descritti dalla funzione  $\bar{X} = [x_1, \dots, x_n]$  con intensità definite dai pesi  $\bar{W} = [w_1, \dots, w_n]$  fornendo un solo segnale di uscita  $y$ . La funzione di previsione  $\hat{y}$  è data quindi da:

$$\hat{y} = \sum_{i=1}^n w_i x_i$$

Il valore  $\hat{y}$  sta ad indicare che è una previsione piuttosto che un valore osservato. L'errore di previsione sarà quindi dato da  $E(\bar{X}) = \hat{y}_t - y_t$ .

Il dato di output, invece, nasce da un'opportuna *funzione di attivazione*:

$$y = f(y - \theta)$$

Dove  $\theta$  corrisponde ad una soglia che risulta essere specifica per ogni singolo neurone. Solitamente è una funzione non lineare che assume valori appartenenti al campo reale  $[-1,1]$ .

Le più comuni funzioni di attivazione utilizzate all'inizio degli sviluppi delle reti neurali erano la funzione segno, sigmoidea e tangente iperbolica, rispettivamente rappresentate:

$$\Phi(v) = \begin{cases} 1 & : t \geq 0 \\ -1 & : t < 0 \end{cases}$$

$$\Phi(v) = \frac{1}{1 + e^{-v}}$$

$$\Phi(v) = \frac{e^{2v} - 1}{e^{2v} + 1}$$

Solitamente si utilizza il numero di Eulero (e) ma può assumere altri valori che conferiscono più o meno ripidità alla funzione.

Di recente sono state sviluppate nuove tipologie di funzioni di attivazione che meglio si adattano ai problemi, come la funzione “rettificatore” e la funzione “hard tanh”:

$$\Phi(v) = v^+ = \max(0, v)$$

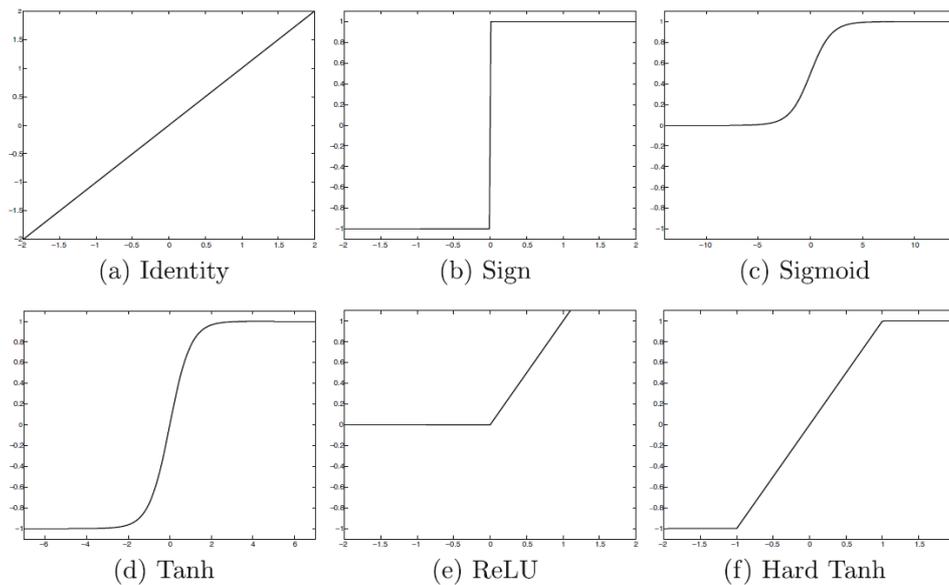
$$\Phi(v) = \begin{cases} -1 & : z < -1 \\ z & : -1 \leq z \leq 1 \\ 1 & : z > 1 \end{cases}$$

Nel 2018, la funzione rettificatore, anche chiamata unità lineare rettificata (ReLU) è stata riconosciuta come la funzione di attivazione più largamente utilizzata nelle reti neurali profonde.

La funzione “hard tanh” è preferita alla funzione “tanh” poiché è computazionalmente più economica.

È da notare che il Perceptron contiene due livelli effettivi, ma in ragione al fatto che il livello di input non esegue alcun calcolo e trasmette solo i valori non è incluso nel conteggio del numero di livelli. Nella maggior parte dei casi, il primo strato è determinato in maniera casuale, motivo per cui viene chiamato strato associativo. Poiché il Perceptron contiene un singolo strato computazionale è detto single-layer network.

Figura 23: Funzioni di attivazione.



### *Le reti Multi-layer Perceptron (MLP)*

Sono definite reti multi-layer Perceptron quelle che contengono uno strato di input, uno strato di output e uno o più strati di neuroni intermedi, ovvero non sono altro che unità di calcolo connesse tra loro, le quali formano nodi, che si distribuiscono in una delle tre posizioni appena elencate.

Come avviene per il Perceptron anche il funzionamento del MLP si fonda sul concetto di funzione di ottimizzazione dei pesi, in questo caso in ciascun nodo presente all'interno del modello, ma la maggiore complessità di questa struttura richiede necessariamente l'utilizzo di metodologie di ottimizzazione più complesse; nella maggior parte dei casi l'algoritmo di apprendimento utilizzato è il “*Backpropagation*”. In questo modello, proposto da David E. Rumelhart nel 1986, ad ogni modifica dei pesi effettua il ricalcolo dell'errore partendo dal livello di output e risalendo attraverso i nodi intermedi, con piccole variazioni dei pesi, graduali e progressive, determinate dalla stima dell'errore tra il risultato prodotto e quello desiderato. Le reti apprendono tramite una serie, talvolta anche prolungata, di tentativi che consentono di modellare in modo (quasi) ottimale i pesi che collegano gli input con l'output passando per i neuroni degli strati nascosti. Nella pratica i pesi assumono valori compresi tra 0 e 1 e vengono continuamente aggiustati con un numero di confronti che deve essere sufficientemente grande. Tale procedimento verrà effettuato finché l'errore non risulta basso, in riferimento ad un valore prefissato.

### 2.3.4 Architettura dei collegamenti

Un'altra classificazione importante può essere fatta in base al tipo di architettura, ovvero in base al modo nel quale sono collegati tra loro i vari neuroni. Le strutture principali sono Feedforward e ricorrente.

#### *Struttura Feedforward*

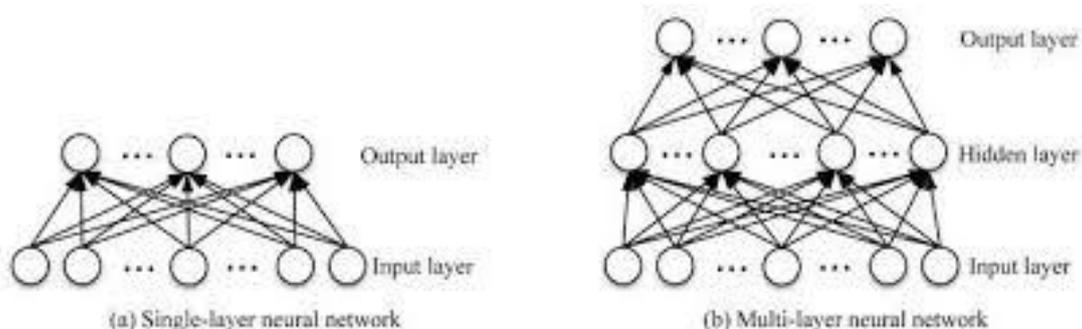
Le reti feedforward possono essere presenti sia nelle strutture single-layer network sia in quelle multi-layer network. Questo tipo di reti neurali sono state le prime ad essere ipotizzate e applicate data la loro semplice struttura. In una rete di questo tipo le connessioni tra le unità non formano cicli e le informazioni si muovono solamente “in avanti”, partendo dalle unità iniziali, attraverso i nodi nascosti, fino ad arrivare ai nodi d'uscita. In una rete di tipo feedforward i segnali si propagano dall'input all'output solo attraverso i neuroni intermedi, non avvalendosi di connessioni trasversali.

Nel single-layer network la struttura sarà formata esclusivamente da un nodo di ingresso ed uno di uscita e i dati si propagano in un solo verso, con la caratteristica che ci sarà solo un livello che effettua l'elaborazione dati in quanto non presenta nodi nascosti.

Nel multi-layer network sono presenti anche livelli nascosti ma come nel caso precedente la propagazione del segnale può essere fatta in un solo verso.

Una rete feedforward particolare è rappresentata dalla CNN (Convolutional Neural Network) usata spesso nel riconoscimento di immagini e la loro classificazione, o ancora, il riconoscimento dei volti.

Figura 24: Rete feedforward a singolo e multi strato.

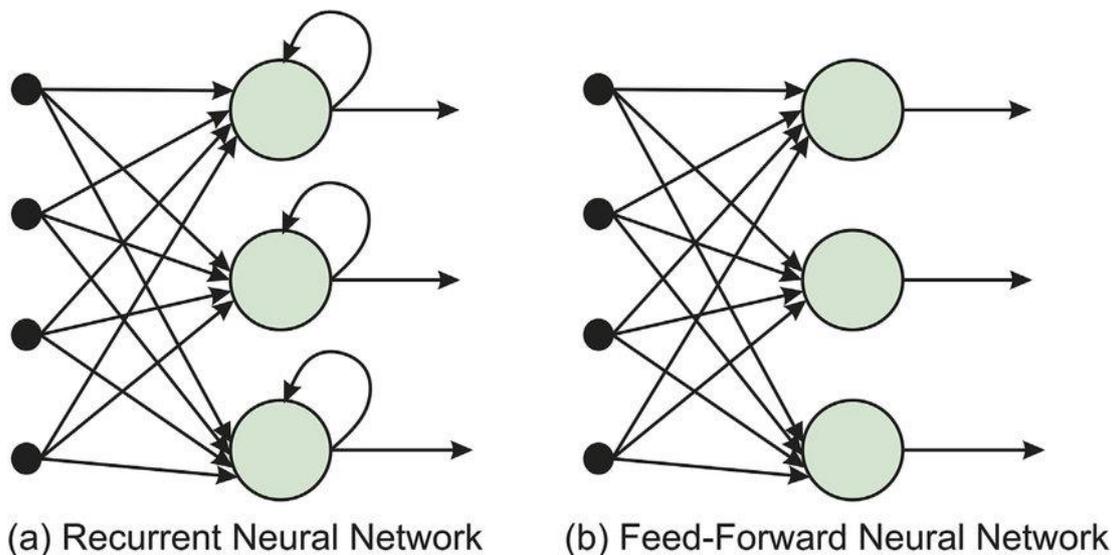


Fonte: Neural network and deep learning. A textbook, Springer, 2018.

### Struttura ricorrente

Una rete ricorrente si distingue da una feedforward in quanto presenta carattere ciclico. Significa che i valori di output di uno strato di un livello superiore vengono a loro volta utilizzati come valori d'ingresso anche in nodi precedenti o nodi del livello stesso e non solo come dati di input in strati superiori. Tale caratteristica è fondamentale per permettere l'utilizzo di uno strato come memoria di stato, consentendo di fornire dei valori temporali in ingresso e di modellarne un comportamento dinamico rispetto alle informazioni ricevute in precedenza. Questa caratteristica rende questo tipo di architettura adatta a compiti di analisi predittiva su sequenze di dati, come lo sono le serie storiche. Tra le reti neurali ricorrenti più utilizzate troviamo le LSTM (Long-Short Term Memory) che introducono una cella di memoria che permette di ricordare un gran numero di dati rilevanti nel caso in cui siano a molte distanze temporali.

Figura 25: Confronto tra una rete ricorrente e una feedforward.



Fonte: Rete neurali artificiali: Teoria ed Applicazioni, Crescenzo Gallo, 2007.

## 2.4 Algoritmo di Backward Propagation Error

Il termine Backward Propagation, più comunemente detto Backpropagation, è spesso frainteso come l'algoritmo di apprendimento delle reti neurali, in realtà, si riferisce solo al metodo per calcolare il gradiente mentre un altro algoritmo, come la Discesa del Gradiente Stocastico, viene utilizzato per eseguire l'apprendimento. Consiste in una generalizzazione della "regola Delta", che a sua volta è un modo per aggiornare i pesi

sinaptici basato sul metodo della Discesa del Gradiente (Gradient Descent), una procedura iterativa la cui finalità è quella di individuare il punto di minimo globale di una funzione sulla base di informazioni disponibili localmente. In sostanza il metodo della discesa del gradiente, data una funzione, permette di ottimizzarla ed è, quindi, il metodo con il quale si trova il punto di minimo, mentre il Backpropagation è un algoritmo che permette di calcolare in modo efficiente il gradiente della funzione rispetto ai vari layer. L'algoritmo della discesa del gradiente ottimizza (minimizza) la funzione di perdita, in modo da ridurre l'errore di previsione per tutti gli elementi presenti nel training set. Per farlo, i pesi vengono aggiornati andando a sottrarre il gradiente della funzione di perdita moltiplicato per il learning rate, che regola la velocità di apprendimento dell'algoritmo. Si conclude quando si raggiunge il livello desiderato di errore. Una corretta regolazione dei pesi garantisce un modello che può essere generalizzato.

Invece, data una rete neurale e una funzione di perdita, chiamata anche "loss function", l'algoritmo di Backpropagation calcola il gradiente della loss function rispetto ai pesi della rete. La backpropagation si basa sulla cosiddetta "chain rule" ovvero una regola di derivazione che permette di calcolare la derivata della funzione composta di due funzioni derivabili: questo algoritmo è l'applicazione ricorsiva di questa regola. I parametri dei neuroni sono aggiornati in modo proporzionale alla derivata parziale della loss function rispetto al parametro stesso.

Una delle principali difficoltà dell'uso della backpropagation è il problema noto come *scomparsa del gradiente (vanishing gradient)*, dovuto all'uso di funzioni di attivazione non lineari, comprese nell'intervallo (0;1), che causano una diminuzione esponenziale del valore del gradiente all'aumentare della profondità della rete neurale. Poiché nella retropropagazione i gradienti ai vari livelli sono moltiplicati tramite la chain rule, il prodotto di  $n$  numeri (0;1) decresce esponenzialmente rispetto alla profondità  $n$  della rete. Quando, invece, le funzioni di attivazione, sempre non lineari, non sono comprese nell'intervallo (0;1) e possono assumere valori anche superiori si può manifestare il problema analogo dell'*esplosione del gradiente (exploding gradient)*. Alcune soluzioni furono trovate nel 1997, da Sepp Hochreiter e Jurgen Schmidhuber che introdussero l'architettura LSTM in cui lo stato del modello si propaga nel tempo senza attraversare funzioni non lineari prevenendo così questi problemi.

## 2.5 Reti neurali: potenzialità e limiti

Gli esseri umani e i computer sono adatti a diversi tipi di compiti. Per esempio, calcolare la radice cubica di un grande numero è molto semplice per una macchina, mentre risulta estremamente difficile per un essere umano. D'altra parte, un compito come riconoscere un'immagine o un oggetto è relativamente semplice per un essere umano e fino a poco tempo fa risultava complesso per una macchina. È solo negli ultimi anni che questo tipo di algoritmi di apprendimento automatizzato hanno sviluppato una maggiore precisione fino a superare recentemente le prestazioni umane grazie allo sviluppo del deep learning. Un vantaggio consistente delle reti neurali rispetto alle tradizionali tecniche di apprendimento automatico è che il primo rispetto al secondo fornisce un livello di astrazione superiore delle funzioni in cui sono inseriti i dati mediante la progettazione del modello.

Il secondo vantaggio è che le reti neurali forniscono un modo semplice per regolare la complessità di un modello aggiungendo o rimuovendo neuroni dall'architettura secondo la disponibilità di dati di addestramento o la potenza di calcolo. Le reti neurali per come sono costruite lavorano in parallelo e sono quindi in grado di trattare molti dati mentre nei calcolatori tradizionali ciascun dato viene elaborato individualmente e in successione. Le performance del machine learning rimangono migliori per set di dati limitati grazie alla maggior possibilità di scelta e maggior facilità nell'interpretazione del modello.

In generale, la capacità delle reti neurali di estrapolare pattern di dati che non si erano presentati in fase di addestramento le rende particolarmente adatte alla classificazione di input caratterizzati da forti rumori. Per contro, non c'è una garanzia che i criteri utilizzati nella rete attraverso l'esplorazione di un campione siano replicabili a tutta la popolazione della quale si vuole descrivere il comportamento. Perciò se da un lato l'errore minimo della rete in fase di addestramento è indice della sua accuratezza, dall'altro è importante evitare una classificazione troppo dettagliata per non individuare eccezioni piuttosto che regole con un sufficiente grado di generalità.

Un problema delle reti neurali può essere definito come quello dei "minimi locali". La funzione di errore di un MLP può essere considerata come una superficie frastagliata con molteplici avvallamenti più o meno profondi ed alcuni di questi, anche se non sono i più profondi in assoluto, rischiano di intrappolare il corso dell'ottimizzazione della rete. L'incapacità di aggirare questi minimi locali nel momento in cui vengono rilevati è uno dei maggiori punti di criticità degli algoritmi di apprendimento iterativo: l'algoritmo Back

Propagation, per esempio, ricercando il minimo sempre e comunque nella direzione di massima pendenza negativa della superficie di errore, vincola indissolubilmente le proprie opportunità di convergenza nello stato iniziale dei pesi che, come sappiamo, è determinato casualmente.

Un altro dei più grandi limiti delle reti neurali, anche se molto efficienti, è che non sono completamente spiegabili in un linguaggio simbolico umano, i risultati non sono deducibili e derivano da un processo che non può essere spiegato.

Come per qualsiasi algoritmo, anche le reti neurali sono efficienti solo se le variabili predittive scelte hanno un carattere esplicativo all'interno del modello. Non sono in grado di trattare in modo efficiente le variabili di tipo categorico, ovvero valori testuali e non numerici (per esempio, il nome delle città), con molti valori diversi.

Inoltre, hanno bisogno di una fase di apprendimento e addestramento del sistema che può risultare estremamente lunga per riuscire a stabilire i pesi dei singoli neuroni.

Infine, non esistono teoremi o modelli che definiscono la rete neurale migliore, ma la buona riuscita della costruzione del modello stesso dipende dal creatore.

## 2.6 Reti neurali ambito finanziario

Come descritto precedentemente gli ambiti di sviluppo delle reti neurali in campo economico-finanziario sono esplosi negli ultimi anni tanto da rendere difficile una loro classificazione, tuttavia, è possibile suddividerle, in base ai campi applicativi, in tre macro aree:

- Previsione serie temporali
- Classificazione
- Approssimazioni di funzioni

### 2.6.1 Previsione serie temporali

In questo ambito troviamo la maggior parte degli sviluppi delle reti neurali in campo finanziario il cui obiettivo è quello di realizzare profitti speculativi tramite trading di attività finanziarie. Il classico dato con il quale avviene l'elaborazione è la serie storica che deve essere suddivisa in due parti: la prima composta dalle osservazioni "*in-sample*", funge da base per l'addestramento e l'apprendimento dell'algoritmo, mentre la seconda,

composta dalle osservazioni “*out-of-sample*”, ha lo scopo di verificare la validità dell’apprendimento stesso. La rete può essere addestrata per la previsione di ampi orizzonti temporali utilizzando le sue stesse previsioni a breve termine come input per previsioni a lungo termine, risulta, quindi, essere estremamente vantaggioso includere tra gli input anche l’output ottenuto dall’applicazione della medesima rete ad un pattern precedente:  $Y_t = f(X, Y_{t-1})$ . Un modello ricorsivo come questo consente di migliorare l’accuratezza della stima introducendo un fattore correttivo ricavato dagli esiti della funzione stessa.

Le reti neurali consentono di affrontare problemi finanziari che solitamente sono analizzati con metodi lineari, ma nella pratica i fenomeni, non solo finanziari, raramente si manifestano in forma lineare, a maggior ragione le decisioni umane, per questo si cerca di sviluppare un modello che sia il più possibile uguale al processo decisionale umano.

Per l’efficienza e l’efficacia di questo tipo di applicazioni è molto importante la cura di determinati aspetti tecnici:

- *Scelta delle variabili di input*: deve avvenire considerando che la rete non fornisce alcuna funzione esplicativa e quindi potrebbe utilizzare variabili non significative, questo è dovuto anche al fatto che nel tempo le relazioni tra le variabili cambiano e di conseguenza input significativi oggi potrebbero non esserlo più in futuro.
- *Livello ottimo di apprendimento*: è necessario tener conto che un processo di training troppo corto non consente alla rete di cogliere le relazioni tra le variabili, mentre un processo troppo lungo potrebbe rendere la rete incapace di generalizzare (over training).
- *Scelta dell’orizzonte temporale di riferimento per la previsione*: orizzonti temporali più brevi accrescono il numero di previsioni corrette, portando un profitto medio più elevato, al contrario di quelle lunghe che risultano mediamente meno corrette.

## 2.6.2 Classificazione

Applicazioni tipiche di queste finalità riguardano il rischio di credito o le decisioni di affidamento. Nei casi di classificazione la rete ha il compito di assegnare gli input a un certo numero di categorie predefinite cui corrispondono altrettanti output.

### 2.6.3 Approssimazioni di funzioni

In questo caso le reti vengono applicate in tutte le funzioni avanzate di pricing e di risk management in cui manca una forma funzionale precisa per la valutazione degli strumenti finanziari, come possono esserlo i derivati, si pensi alle opzioni americane o esotiche. Sono tematiche complesse che hanno numerosi progetti di ricerca come, per esempio, modelli per l'investimento azionario, previsioni per applicazioni di option pricing, modelli per la valutazione e negoziazione di futures, ecc.

## 2.7 Costruzione della rete neurale

Per elaborare una rete neurale in ambito finanziario è utile seguire un percorso che si divide in cinque fasi principali:

- L'individuazione dell'*obiettivo* della previsione;
- La costruzione dell'*archivio* dei dati su cui attivare l'apprendimento della rete neurale;
- L'*apprendimento* e la scelta dell'*architettura* e dei *parametri* necessari;
- L'individuazione degli *indicatori di errore* dell'output;
- La generalizzazione dell'output per la *previsione* dei mercati finanziari.

### 2.7.1 La definizione dell'ambito di indagine

La prima fase per la costruzione di una rete neurale consiste nell'individuare l'obiettivo dell'analisi. Per esempio, è importante tenere a mente la frequenza dei dati utilizzati su cui calcolare l'output, che dipende dall'obiettivo della previsione. L'orizzonte temporale può essere decisamente diverso tra un trader che opera nel breve-medio periodo rispetto ad un investitore che opera sul lungo periodo.

### 2.7.2 Costruzione dell'archivio dati

Un momento importante è la costruzione dell'archivio dati ovvero il set di informazioni che verranno utilizzate dalla rete neurale per riconoscere l'evoluzione della variabile di output. Le componenti rilevanti sono:

- La raccolta
- L'analisi e la trasformazione
- La selezione delle variabili di input e di output

La raccolta deve rispondere ad alcuni principi fondamentali. In primo luogo, è necessario che i dati siano raccolti dai mercati con regolare frequenza di rilevazione per garantire la continuità della serie storica. In secondo luogo, la rete dovrebbe essere messa nelle medesime condizioni conoscitive dell'analista di mercato, ovvero, i fattori che determinano le scelte devono poter essere riconosciuti dalla rete stessa. Inoltre, è fondamentale prevedere il rinnovamento periodico dell'archivio dei dati, al fine di aggiornare i parametri della serie. Non è inoltre possibile fare apprendere una rete utilizzando informazioni la cui rilevazione è relativa a periodi caratterizzati da trasformazioni strutturali del mercato.

La successiva fase di analisi e trasformazione ha come scopo principale quello di scegliere la miglior configurazione possibile dei dati. In generale la sua finalità è quella di ridurre il "rumore" dei dati della serie storica mediante l'utilizzo di trasformazioni che non alterino la dinamica del fenomeno. La soluzione più frequente risulta essere la normalizzazione della serie attraverso trasformazioni logaritmiche o regressioni statistiche.

La selezione dei dati, sia di input che di output, diventa delicata dal momento in cui è necessario mantenere tutti i dati rilevanti ed eliminare quelli che risultano non significativi in relazione al contenuto della rete ed hanno, quindi, un'incidenza negativa sui risultati.

La procedura che l'analista dovrebbe seguire può essere riassunta nelle seguenti fasi:

1. Definizione del primo database molto ampio;
2. Prima applicazione del meccanismo di apprendimento della rete;
3. Valutazione del contenuto informativo delle singole variabili;
4. Analisi della matrice di correlazione fra le variabili in input;
5. Progressiva eliminazione delle variabili risultate meno indicative e contestualmente più autocorrelate fra loro;
6. Nuova applicazione dell'algoritmo di apprendimento sul nuovo database ridotto.

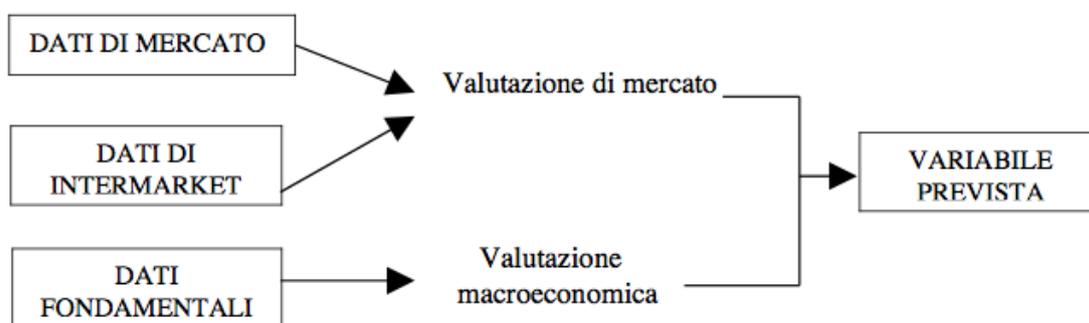
Questo procedimento dovrebbe portare all'individuazione dell'archivio dati ottimale che garantisce la miglior capacità di apprendimento della rete con il minimo sforzo

informativo. L'osservazione dei coefficienti di correlazione e dei relativi grafici consente di valutare la natura dell'eventuale correlazione all'interno dell'insieme delle variabili in input e anche la loro relazione con la variabile in output.

La rete neurale per la previsione di serie storiche finanziarie può essere impostata utilizzando differenti tipologie di informazioni:

- *Market data*, reperibili direttamente dal mercato e sono direttamente legate alla variabile di output;
- *Intermarket data*, risultano collegate solo operativamente alla variabile di output;
- *Fundamental data*, dipendenti dalle caratteristiche fondamentali del titolo.

Figura 26: Input dei dati nelle previsioni finanziarie.



Fonte: Rete neurali artificiali: Teoria ed Applicazioni, Crescenzo Gallo, 2007.

Quanto appena detto è comunque una definizione generale e semplificata in quanto ogni problema richiede una valutazione propria dovuta alle caratteristiche specifiche del problema stesso. Le variabili intermarket e fondamentali non sempre risultano significative per la spiegazione del fenomeno ma possono dipendere, per esempio, dalla frequenza delle osservazioni o dall'esistenza di ostacoli strutturali all'interno del mercato di valutazione.

Come detto nel precedente capitolo, l'analisi tecnica si fonda sull'assunzione che i prezzi espressi dal mercato riflettano le spinte che concorrono alla formazione futura dei prezzi stessi e, dal momento che il mercato è soggetto a movimenti ciclici dovuti al comportamento degli operatori presenti, osservando le evoluzioni passate si può prevedere le reazioni al presentarsi di configurazioni già avvenute. Quindi l'utilizzo degli indicatori tecnici come input nella fase di addestramento della rete neurale può portare numerosi vantaggi informativi ma si deve tenere in considerazione la necessità di non

risultare ridondanti nelle informazioni fornite, selezionando soltanto quegli indicatori che risultino il più possibile differenti e complementari tra loro.

A questi elementi di mercato si aggiungono almeno due elementi fondamentali: il ciclo economico e la dinamica degli indici dei prezzi. Un problema associato all'inserimento dei dati fondamentali è quello della frequenza di rilevazione: produzione industriale e inflazione hanno una cadenza mensile e risultano inadatti per previsioni al di sotto di questo periodo.

### 2.7.3 Apprendimento

La costruzione della rete neurale passa per alcune fasi che devono consentire di fissare i parametri utili per l'apprendimento ideale alla soluzione del problema. La prima fase è quella della scelta dell'architettura più adatta per l'apprendimento. Il meccanismo e la struttura delle connessioni tra input e output, tramite gli strati nascosti, risultano decisivi per la risoluzione del problema. Esistono alcuni algoritmi proposti in letteratura per ottimizzare la scelta dell'architettura (*pruning algorithm*, *polynomial time algorithm*, *canonical decomposition technique*, *network information criterion*) ma si ritiene che la fase di test sia un criterio più affidabile quando non si hanno riferimenti diretti al problema che si vuole risolvere.

In generale si può sostenere che esistano alcuni parametri da determinare nella definizione dell'architettura:

1. *La suddivisione temporale del database*: una volta definito il contenuto del database da cui selezionare le variabili di input, l'analista deve suddividere la serie storica in sottoperiodi, i quali indicano il periodo di apprendimento (*training set*) e l'ambito di valutazione, che viene a sua volta diviso in *test set* e *generalisation set*. La rete impara cercando di riconoscere la dinamica del training set, poi ne verifica il suo adattamento nel test set e infine lo applica su un insieme di dati (*generalisation set*) che non ha mai osservato. I criteri di determinazione ed estrazione dei sottoperiodi possono essere estremamente vari, ma nel caso di reti neurali applicate alla previsione dei mercati finanziari il criterio che viene esclusivamente utilizzato prevede che la serie storica venga rigidamente divisa nei tre periodi di training, test e generalisation secondo questo ordine cronologico. Anche per la suddivisione non c'è una regola universale ma si suggerisce di utilizzare da un minimo di 2/3 del totale della serie per il training fino a 4/5 dei

dati. Le soluzioni più utilizzate sono, rispettivamente (60%, 20%, 20%) e (60%, 30%, 10%).

2. *Il numero degli strati nascosti e dei neuroni da inserire in ciascuno strato:* in relazione al numero di strati, diversi contributi indicano che uno strato nascosto è sufficiente per approssimare le funzioni non lineari più ricorrenti con elevato grado di accuratezza, tuttavia, tale metodo ha il limite di dover utilizzare un elevato numero di neuroni, il che limiterebbe il processo di apprendimento alla rete stessa. L'utilizzo di reti con due strati nascosti risulta più efficace per problemi di previsione su dati ad alta frequenza. In riferimento al numero di neuroni da inserire in ciascuno strato il criterio da seguire nella scelta è quello della minimizzazione del rischio di overlearning che avviene quando si decide di inserire un numero eccessivamente elevato di neuroni i quali spiegano perfettamente il pattern della serie storica ma non sono in grado di generalizzare una previsione affidabile perché riducono in maniera consistente il contributo degli input. Se, viceversa, viene inserito un basso numero di neuroni in ogni strato si riduce la capacità di apprendimento della previsione. Le formule proposte in letteratura sono varie:

$$n_{hl} = 2 \cdot n_{input} + 1$$

$$n_{hl} = 2 \cdot n_{input}$$

$$n_{hl} = n_{input}$$

$$n_{hl} = \frac{n_{input} + n_{output}}{2} + \sqrt{n_{training}}$$

Dove:

$n_{hl}$ : numero di neuroni nascosti;

$n_{input}$ : numero di input;

$n_{output}$ : numero di output;

$n_{training}$ : numero di osservazioni contenute nel training set

Purtroppo, i contributi empirici dimostrano che nessuna di queste regole risulta generalizzabile ad ogni problema previsionale.

3. *Le connessioni dell'architettura:* in relazione alle connessioni dell'architettura viene considerata, seppur non l'unica esistente ma la maggior utilizzata, l'architettura Back Propagation. Come già descritto è un modello supervisionato il cui apprendimento risulta condizionato sia da variabili di input che di output. In questo caso è possibile individuare diverse possibilità di connessione tra gli strati della rete: le *connessioni standard*, le *connessioni a salti* e le *connessioni ripetute*. Le connessioni standard prevedono connessioni tra input e output passanti per gli strati nascosti ai quali sono assegnati dei neuroni che facilitano l'apprendimento della rete. Le connessioni sono dirette e non prevedono né salti né ritorni su sé stessi (cicli). Le connessioni a salti prevedono, invece, che la rete assegni dei pesi connettivi anche fra neuroni presenti in strati non adiacenti: nel caso più semplice lo strato di input presenta una serie di connessioni non solo con lo strato nascosto ma anche con quello di output. Questa ramificazione connettiva si amplifica ulteriormente se il numero di strati nascosti aumenta. Infine, le reti neurali a connessioni ripetute prevedono la possibilità che i neuroni assegnati agli strati nascosti possano ritornare sulle variabili di input con processi iterativi così da quantificare in modo preciso il peso connettivo. Quest'ultima tipologia di connessioni risulta particolarmente utile per l'analisi delle serie storiche finanziarie in quanto riconosce le sequenze che si creano nell'ambito dei mercati.

Figura 27: Architettura backpropagation a uno strato nascosto con connessioni standard.

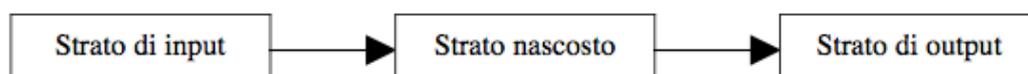


Figura 28: Architettura backpropagation a uno strato nascosto con connessioni a salto.

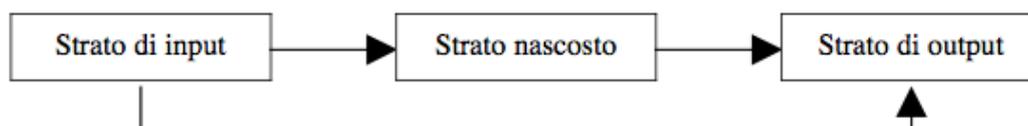
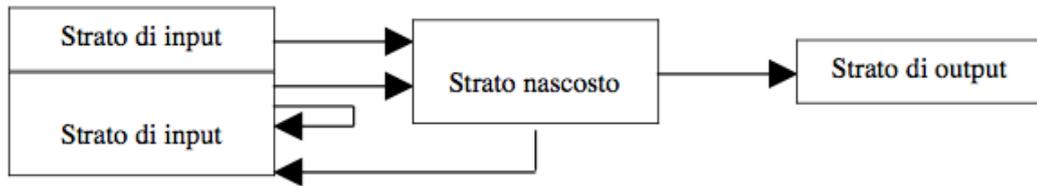


Figura 29: Architettura backpropagation a uno strato nascosto con connessioni ripetute.

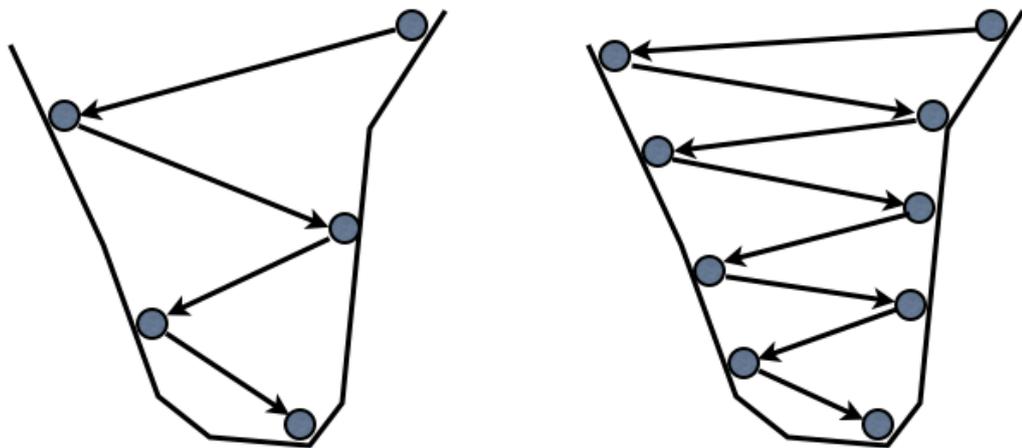


Fonte: Rete neurali artificiali: Teoria ed Applicazioni, Crescenzo Gallo, 2007.

4. *La funzione di attivazione*: come precedentemente detto ad inizio capitolo non esiste una regola universale per definire la miglior funzione di attivazione dei vari strati che compongono la rete neurale ma ne esistono diverse che possono risultare migliori o peggiori in base alla tipologia di problema che si cerca di affrontare. Inoltre, sebbene molti studi presentino funzioni diverse per ogni strato è comune adottare lo stesso modello per input, hidden e output. Per esempio, la funzione di attivazione lineare viene utilizzata per lo strato di output in quanto evita che il risultato tenda verso il minimo o il massimo, mentre per quelli nascosti è sconsigliata perché i neuroni risulterebbero connessi proprio su una base funzionale che si vuole evitare. Il limite di questa funzione è quello di non consentire un fitting adeguato nelle serie storiche caratterizzate da un trend. Ancora, la funzione sigmoidea che varia in un intervallo  $(0;1)$  risulta particolarmente utile negli strati nascosti delle reti applicate alle serie finanziarie perché coglie in maniera particolarmente precisa le dinamiche che avvengono all'interno del mercato. Allo stesso modo la funzione tangente iperbolica adatta la rete in maniera affidabile negli strati nascosti (in particolare nelle reti a tre strati), soprattutto nel caso in cui lo strato di output utilizzi una funzione lineare o sigmoidea.
5. *Le regole di apprendimento e il momentum*: è necessario definire i criteri di arresto dell'apprendimento a seconda dell'obiettivo che ci si è posti. Se la rete neurale è costruita con finalità previsionali è consigliato valutare l'apprendimento sul test set, altrimenti, se lo scopo è lo studio di un fenomeno, è più indicato l'apprendimento tramite training set. I criteri utilizzati sono legati agli indicatori di errore.
6. *L'aggiornamento dei pesi di connessione dei neuroni*: in questa fase è necessario definire con quale tasso di apprendimento (*learning rate*) la rete deve modificare

i propri pesi dei vari neuroni rispetto alla significatività dell'errore connesso. Il rischio consiste di condizionare la rete ad avere variazioni troppo ampie che non riuscirebbero a seguire l'evoluzione corretta della serie storica, senza arrivare, quindi, ad una convergenza dei risultati. In tal caso è necessario modificare il *learning rate*. Evidentemente, dover trovare il valore più adatto dei parametri comporta la necessità di effettuare un'ampia serie di tentativi che portano verosimilmente a risultati differenti. L'analista deve decidere il valore iniziale del peso da dare alle connessioni o può essere casuale per poi valutare se le osservazioni sono caratterizzate da un alto livello di rumore. Un metodo generale suggerito è quello di evitare ampie variazioni dei parametri. Un tasso di apprendimento molto alto aumenta anche il rischio delle variazioni erratiche (oscillazioni del risultato) è quindi l'instabilità del sistema. Per ridurre le oscillazioni e favorire la convergenza dei risultati nella formula di aggiustamento dei pesi si utilizza un secondo parametro ( $\alpha$ ), detto momento. Sia il parametro dei pesi che il momento variano da zero a uno (0;1).

Figura 30: Sviluppo della rete con tassi di apprendimento differenti.



Fonte: Rete neurali artificiali: Teoria ed Applicazioni, Crescenzo Gallo, 2007.

#### 2.7.4 Indicatori di errore

I parametri di apprendimento sono legati agli indicatori degli errori commessi dalla rete e sono generalmente legati a: errore medio, errore massimo e numero di epoche senza miglioramento dell'errore. Una cosa importante è che il valore di errore si stabilizzi, ovvero converga. Se anche gli errori risultano modesti, ma l'oscillazione ha portato ad

un'elevata divergenza, è opportuno verificare l'adeguatezza dei parametri. In questo senso esistono vari indicatori che possono fungere da funzione di perdita, tanto più qualitativi quanto più è basso l'indice di errore; tra i vari indicatori di errore sviluppati in ambito statistico spiccano:

- L'indice di determinazione ( $R^2$ )
- Mean Absolute Error (MAE)
- Mean Absolute Percentage Error (MAPE)
- Mean Square Error (MSE)
- Root Mean Square Error (RMSE)

Si tratta di indicatori che misurano in vario modo la differenza tra l'output originario e quello previsionale stimato dalla rete, ma solo nel caso in cui tutti i parametri fossero in grado di individuare il fenomeno originale la differenza tra output reale e stimato sarebbe nulla ottimizzando questi indicatori. Il limite di questi indicatori è quello di basarsi su un concetto di scostamento simmetrico rispetto al valore reale, mentre in finanza l'errore è solitamente misurato in termini di perdita, per questo sarebbe opportuno stimare i pesi della rete basandosi sui profitti ottenuti e nel caso adottare filtri per rimuovere il problema ex-post.

Per indice di errore si intende la funzione di perdita derivante da precisi parametri scelti del modello. Si può quindi riformulare l'obiettivo di trovare i parametri che minimizzano la funzione di perdita. La funzione di perdita per una rete si ottiene sommando tutti i valori di perdita per ogni output.

Tabella 2: Indicatori statistici di errore di stima, dove:  $\hat{y}_i$  è il valore stimato,  $\bar{y}$  è la media dei valori osservati,  $y_i$  sono i valori osservati e  $n$  è il numero di osservazioni.

Indice	Formula
Indice di determinazione	$R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$
Mean Absolute Error	$MAE = \frac{\sum_{i=1}^n  \hat{y}_i - y_i }{n}$
Il Mean Absolute Percentage Error	$MAPE = \frac{1}{n} \sum_{i=1}^n \left  \frac{\hat{y}_i - y_i}{\hat{y}_i} \right $

Mean Square Error	$MSE = \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}$
Root Mean Square Error	$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}}$

Un indicatore spesso utilizzato per ottimizzare la rete è il coefficiente di Theil, che varia all'interno di un intervallo standardizzato [0;1], dove 0 rappresenta un modello previsionale perfetto.

### 2.7.5 Generalizzazione e previsione della serie storica

L'ultima fase della costruzione di una rete neurale consiste nella verifica della sua bontà previsionale. Infatti, è possibile che il modello riesca a descrivere ottimamente il training e il test set ma poi i risultati siano inadeguati in fase di generalizzazione, ovvero nel nostro caso, la previsione. L'analista dovrà pertanto testare sul generalisation set la rete neurale, analizzando, su una serie storica mai osservata, gli indicatori di errore precedentemente descritti. Nel caso risultassero significativamente peggiori o, comunque, non accettabili la rete dovrà ricalibrare i pesi ed essere nuovamente testata. La previsione finanziaria viene poi spesso condizionata dalla capacità del modello di individuare tempestivamente le inversioni cicliche del fenomeno. Inoltre, è possibile utilizzare l'output originale e quello stimato dalla rete per misurare il ritardo con cui la rete stessa apprende l'inversione. Numerosi sono gli strumenti utili per analizzare la capacità previsionale della rete neurale: in particolare, ad ogni inversione di tendenza del ciclo la serie dei valori del fenomeno deve registrare una variazione dei segni. Se dunque si passa da un trend crescente a uno decrescente, i segni passeranno da "+" a "-".

Per verificare con quale capacità la rete segnala questa inversione, un sistema efficace è quello basato sul calcolo dell'intervallo di confidenza dell'output della rete, seguendo il seguente schema:

1. Calcolo dello scarto quadratico medio della serie di output della rete;
2. Definizione del limite di confidenza;
3. Misurazione dei valori estremi dell'intervallo: tali valori di massimo e minimo corrispondono all'output della rete a cui rispettivamente si aggiunge o si toglie un valore che dipende dall'intervallo di confidenza scelto;

4. Verifica dei casi in cui il segno dei valori del precedente punto coincidono: si può affermare che la probabilità di prevedere correttamente il trend della serie storica coincide con l'intervallo di confidenza;
5. Controllo delle volte in cui viene commesso realmente un errore di segnalazione, valutando così l'affidabilità dello strumento previsionale;
6. Decisione della posizione di mercato da prendere nei casi in cui i valori di massimo e di minimo dell'intervallo hanno il medesimo segno;
7. Se si presentano più occasioni consecutive aventi lo stesso segno è un'indicazione di mantenere la posizione assunta.

In conclusione, le reti neurali artificiali rappresentano un tentativo di individuare dinamiche di mercato esistenti ma non ancora facilmente esplicitabili in modelli in quanto la componente sociale presente nei sistemi finanziari li rende sostanzialmente diversi da quelli matematici. In ambito operativo offrono un evidente contributo nella previsione, in particolare nei mercati caratterizzati da incertezza e caoticità (ma comunque deterministici), ma benché oramai siano largamente utilizzate e studiate non è facile classificare e individuare i risultati delle ricerche per diversi motivi. In primo luogo, i risultati si traducono in architetture e modelli posseduti dalle grandi istituzioni finanziarie: ciò è coerente con l'idea di trovare un modello superiore che possa rimanere tale e quindi non condiviso con la concorrenza. Inoltre, è difficile trovare più ricerche indipendenti l'una dall'altra con il medesimo scopo e le medesime caratteristiche iniziali (basti pensare al set di dati input). Infine, anche la grande vastità di modelli presenti e utilizzabili rende difficile trovare una linearità nelle ricerche.

Si può quindi dire che non è tanto la difficoltà a reperire le informazioni l'ostacolo più grande quanto piuttosto l'abbondanza di esse, ciò succede sempre quando il campo di ricerca è nuovo, vi è un numero di contributi irrilevanti almeno pari, se non superiore, a quelli importanti.

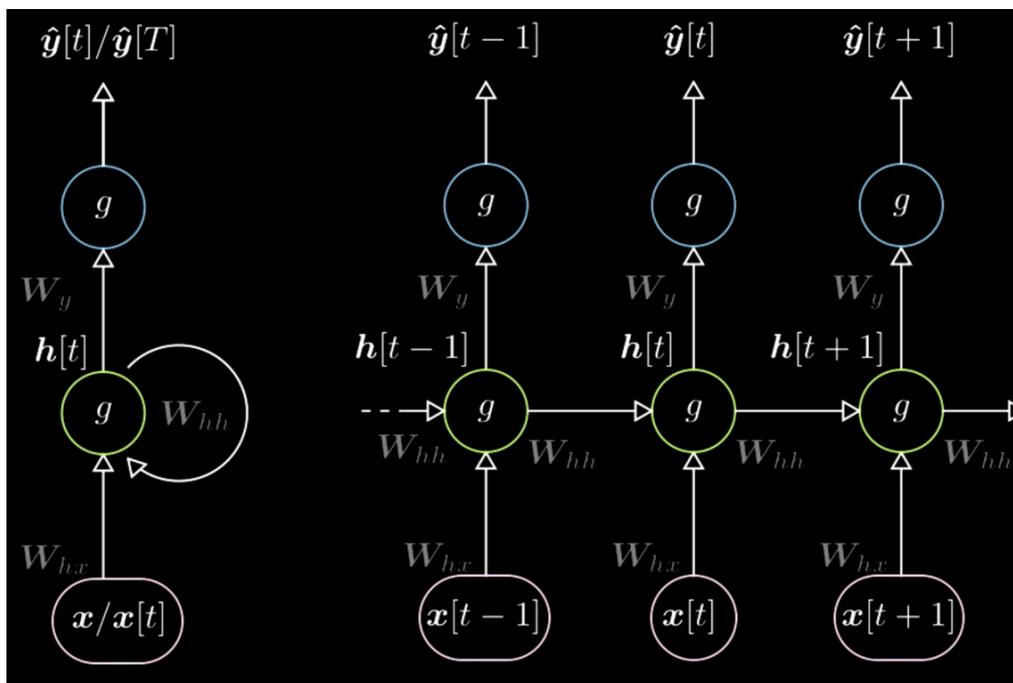
Un altro aspetto interessante è la valutazione comparata delle reti neurali e i modelli statistici di previsione. Ciò che è giusto chiedersi è dove le prime battano i secondi? Dove le reti neurali mostrano vantaggi significativi è nel caso di modelli funzionali fortemente complessi e altamente non lineari. È dimostrato che le reti, sotto precise ipotesi, arrivano a fungere da "approssimatori universali di funzione".

## 2.8 Una rete neurale ricorrente: Long-Short Term Memory

Come già detto precedentemente, in ambito finanziario, le reti neurali che meglio si adattano alle esigenze e gli obiettivi di previsione sono le reti neurali ricorrenti in quanto permettono di gestire una sequenza di dati, come le serie storiche; la differenza rispetto ad una rete neurale è che l'output non dipende solo dall'input corrente ma anche dallo stato del sistema.

L'architettura di una RNN è la seguente:

Figura 31: Architettura Rete Neurale Ricorrente.



Fonte: <https://atcold.github.io/pytorch-Deep-Learning/it/week02/02-1/>.

Nel disegno di sinistra in figura 31 appare la rappresentazione del ciclo, in quello di destra “srotola” il ciclo in un’unica riga nel tempo.

Le rappresentazioni nascoste si possono indicare come:

$$\begin{cases} h(t) = g(W_h \begin{bmatrix} x(t) \\ h(t-1) \end{bmatrix} + b_h) \\ h(0) \doteq 0, \quad W_h \doteq [W_{hx} W_{hh}] \\ \hat{y}(t) = g(W_y h(t) + b_y) \end{cases}$$

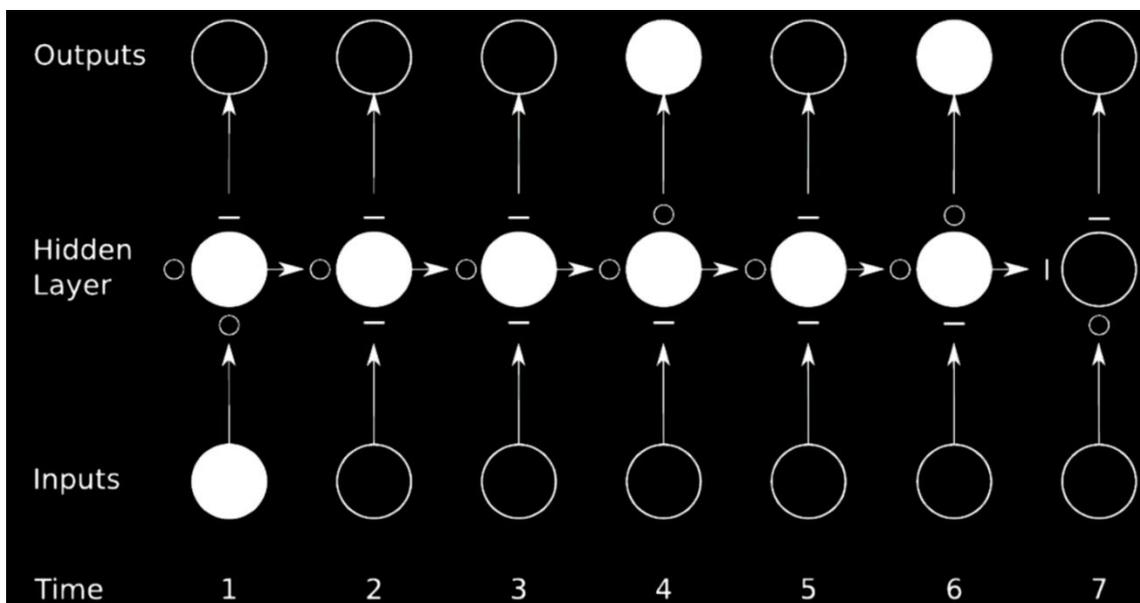
La prima equazione indica una funzione non lineare dove  $W_h$  può essere scritta come due matrici separate  $[W_{hx}W_{hh}]$ , e quindi, sotto determinate condizioni, può essere scritta come:

$$W_h = W_{hx} \cdot x(t) + W_{hh} \cdot h(t - 1)$$

Questa formula evidenzia la presenza di retroazione in quanto l'output all'istante  $t$ , è funzione dell'ingresso  $x(t)$  moltiplicato per il peso  $W_{hx}$  sommato al valore di  $h(t - 1)$  moltiplicato per il peso  $W_{hh}$ , che corrisponde output al tempo precedente.

Il problema principale deriva dal fatto che le matrici possono modificare le dimensioni degli output, in quanto se il determinante è maggiore di 1 il gradiente si gonfia, e viceversa, se è prossimo allo zero, lo porterà a scomparire. Nelle tipiche RNN i gradienti sono propagati attraverso tutte le possibili frecce e questo comporta l'insorgenza, quasi certa, di questa problematica. Un metodo ideale per evitare l'esplosione o la scomparsa del gradiente è quello di saltare le connessioni.

Figura 32: Rete ricorrente con porte logiche.

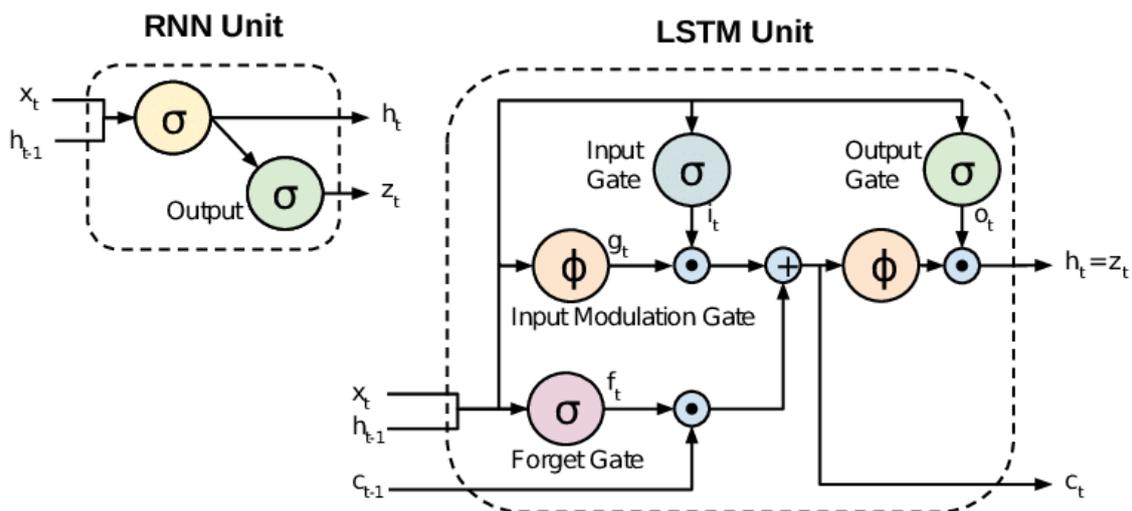


Fonte: <https://atcold.github.io/pytorch-Deep-Learning/it/week02/02-1/>.

In questo esempio nel layer nascosto la porta aperta, rappresentata da un piccolo cerchio, consente il passaggio del gradiente mentre la porta chiusa, indicata con una barra, blocca

la propagazione. Tale tecnica è chiamata “Gated Recurrent Network” (rete ricorrente con porte logiche). La LSTM è la principale RNN gated.

Figura 33: Rappresentazione di una cella Long-Short Term Memory dove  $\sigma$  rappresenta una funzione di attivazione sigmoide e  $\phi$  una funzione di attivazione tangente iperbolica.



Fonte: <https://www.developersmaggioli.it/blog/le-reti-neurali-ricorrenti/>.

Dall'immagine è chiaro che il neurone LSTM è molto più complesso di un semplice neurone ricorrente. Mentre quest'ultimo gestisce semplicemente un'uscita supplementare  $h(t)$  per lo stato attuale e un ingresso  $h(t - 1)$  per lo stato precedente, oltre che due funzioni di attivazione differenti per lo stato  $h$  e l'uscita  $z$ , il neurone LSTM ha al suo interno diverse porte (gate) che gli consentono di decidere in autonomia (durante la fase di addestramento): cosa vale la pena memorizzare o dimenticare, se e come combinare l'ingresso con lo stato interno e se e come restituire l'uscita. Alle porte logiche è applicata una funzione di attivazione sigmoide che restituirà un valore compreso tra 0 e 1.

- La *forget gate* decide se l'informazione in ingresso deve essere mantenuta o meno. Qui arrivano le informazioni dell'input corrente  $x(t)$  e dell'uscita precedente in retroazione  $h(t - 1)$ . Inoltre, questa porta sta moltiplicando il valore precedente dello stato  $c(t - 1)$ . Quindi se la funzione di attivazione restituisce un valore prossimo a 0, lo stato precedente tenderà ad azzerarsi (venire dimenticato), mentre se restituisce un valore vicino a 1, lo stato precedente tenderà a venire memorizzato.

- L'*input gate*, allo stesso modo, decide se i valori in ingresso  $x(t)$  e  $h(t - 1)$  possono essere elaborati insieme allo stato precedente  $c(t - 1)$  o quello che risulta dopo il passaggio attraverso la forget gate. Questo verrà fatto dopo che gli input passano anche attraverso la funzione di attivazione tangente iperbolica  $g(t)$  i cui dati saranno moltiplicati per  $i(t)$ .
- L'*output gate* sfrutta sempre i valori in ingresso  $x(t)$  e  $h(t - 1)$ .
- Lo stato attuale del neurone  $c(t)$  è pari alla somma di valori della forget gate e input gate.
- La rappresentazione finale  $z(t)$ , che corrisponde anche al prossimo ingresso in retroazione  $h(t)$  non è altro che la moltiplicazione dell'*output gate*  $o(t)$  e la versione in tangente iperbolica della cella  $c(t)$ .

Una rete ricorrente fatta di neuroni LSTM è evidentemente complessa in quanto presenta numerosi parametri.

### Capitolo III: Trading System

L'obiettivo di questo elaborato è dimostrare che la previsione del prezzo costruita tramite la rete neurale porti ad una operatività simile a quella che si ha in un trading system costruito sulla serie storica originale del titolo. La previsione sarà fatta utilizzando le reti neurali, mentre la costruzione del trading system si basa su alcuni indicatori tecnici descritti nel primo capitolo, in particolare, data l'enorme quantità di indicatori presenti, sono stati scelti l'incrocio di due medie mobili semplici con periodi differenti e l'RSI. Le prime sono, quindi, delle indicatrici di trend, mentre il secondo un indicatore di momentum.

Il periodo scelto per il trading system applicato ai dati storici va dal 01 gennaio 2010 al 31 dicembre 2020, mentre quello applicato alla previsione si ferma al 2019. È stato escluso il periodo della pandemia in quanto evento singolare e non utile per l'identificazione di un processo automatizzato che provi a prevedere l'andamento di un prezzo, data la sua imprevedibilità e unicità.

Nello specifico, avviene un segnale di acquisto quando la media mobile veloce incrocia a rialzo la media mobile lenta e questo segnale è confermato dalla soglia di ipervenduto dell'RSI. Viceversa, si ha un segnale di vendita quando la media mobile veloce incrocia verso il basso quella lenta, e il segnale è confermato dall'indicatore RSI che si trova nella zona di ipercomprato. La data nella quale viene eseguito l'ordine, sia di acquisto che di vendita, corrisponde esattamente alla data valida immediatamente successiva a quella nella quale avviene il segnale operativo.

I valori utilizzati per questi indicatori sono 20 periodi per la media mobile veloce ( $SMA_{fast}$ ), 55 periodi per la media mobile lenta ( $SMA_{slow}$ ) e 14 periodi per l'RSI con le bande di ipercomprato e ipervenduto poste rispettivamente a 70 e 30. Il periodo preso in considerazione è di dieci anni.

Riassumendo:

$$Signal_t = \begin{cases} -1 & \text{se } SMA_{fast} < SMA_{slow} \wedge RSI > 70 \\ 1 & \text{se } SMA_{fast} > SMA_{slow} \wedge RSI < 30 \end{cases}$$

In questo trading system non è ammessa la vendita allo scoperto e i costi operativi vengono posti pari all' 1% dell'importo scambiato. Il capitale iniziale è di 1.000 \$. Inoltre, per com'è costruito l'algoritmo, il segnale di vendita porta alla chiusura totale della

posizione aperta, mentre in un segnale di acquisto non viene utilizzato tutto il capitale ma il 20%, ovvero 200 \$.

### 3.1 Data Set

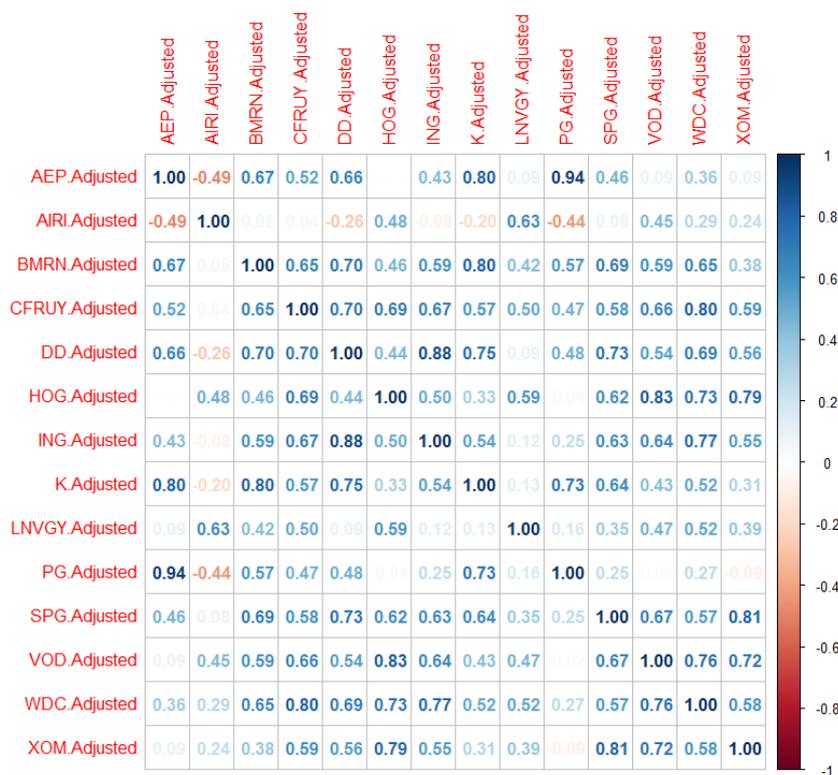
Per testare la validità sono stati scelti quattordici titoli diversificati in base alla suddivisione dei settori del mercato azionario fatta con il sistema GICS (*Global Industries Classification Standard*), il più utilizzato. I settori sono: energetico, materiali, industriale, servizi, sanitario, finanziario, beni voluttuari, beni di consumo, Information Technology (IT), comunicazione e immobiliare.

I titoli scelti sono, rispettivamente ai settori: Exxon Mobil (XOM), Dupont (DD), Air Industries (AIRI), American Electric Power (AEP), BioMarin Pharmaceutical (BMRN), ING Group (ING), Harley-Davidson (HOG) e Richemont (CFRUY), Kellogg (K) e The Procter & Gamble (PG), Lenovo (LNVGY) e Western Digital Corporation (WDC), Vodafone (VOD), Simon Property Group (SPG).

I titoli sono stati scelti appositamente in maniera diversificata per poter evidenziare l'efficacia generale del trading system.

La diversificazione è confermata dalla matrice di correlazione in figura 34.

Figura 34: Matrice di correlazione



Fonte: Grafico generato con R-studio.

Dal grafico è possibile notare che, generalmente, i titoli tra di loro non sono caratterizzati da una correlazione consistente.

Prima di procedere all'applicazione del trading system va fatta una precisazione sugli avvenimenti storico-finanziari avvenuti nel periodo di valutazione.

Nel 2006 scoppiò la crisi dei mutui subprime negli Stati Uniti d'America, innescata da una bolla immobiliare, che si riflesse poi in una crisi economica mondiale durata dal 2007 al 2013 arrivando in Europa nel 2010 con la crisi del debito sovrano.

Tra il 2019 e il 2020 scoppia la pandemia di COVID-19 che avrà un forte impatto in tutto il sistema economico mondiale. Come vedremo in seguito potrà essere rilevante nel trading system.

Per lo sviluppo del Trading System è stato utilizzato il *quantstrat Package* disponibile in R.

### 3.2 Descrizione Trading System

Descriverò brevemente i passaggi più importanti che sono stati fatti nello script in R per arrivare all'equity finale. Lo script completo è in appendice A alla fine della tesi.

Per il TS, come detto, sono stati utilizzati i parametri standard per l'RSI, ovvero 70 e 30 per le soglie di ipercomprato e ipervenduto e 14 giorni per la media, mentre le due medie mobili sono costruite, rispettivamente, a 20 giorni quella veloce ( $SMA_{fast}$ ) e 55 quella lenta ( $SMA_{slow}$ ).

Inizialmente è stato caricato il pacchetto *Quantstrat* (Quantitative Strategy Model Framework) necessario per costruire un trading system. Poi sono stati impostati i parametri iniziali, come la data di inizio e di fine del periodo preso in considerazione, la quantità di patrimonio iniziale e la valuta utilizzata (USD).

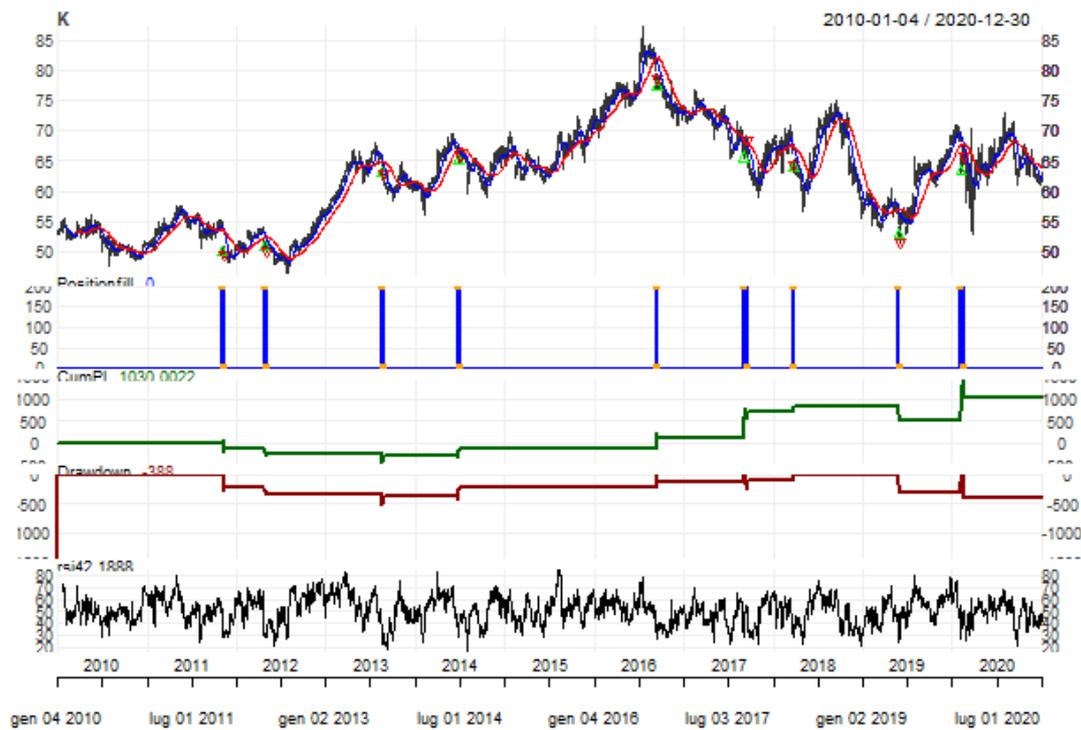
Con il comando *getSymbols* sono state scaricate le serie finanziarie dei titoli scelti direttamente da Yahoo Finance ed è stata costruita la matrice di correlazione per confermare che i titoli fossero diversificati.

Per creare il TS sono stati tre i comandi fondamentali:

1. *add.indicator*, con il quale sono stati inseriti i tre indicatori (le due medie e l'RSI) e i relativi parametri;
2. *add.signal*, con il quale sono stati definiti i segnali da tenere in considerazione, ovvero l'incrocio a rialzo della media mobile veloce rispetto quella lenta nella zona di ipervenduto e, viceversa, l'incrocio a ribasso della media mobile veloce rispetto quella lenta nella zona di ipercomprato;
3. *add.rule*, con il quale sono stati collegati gli ordini di vendita e di acquisto in base ai due segnali stabiliti nel punto precedente.

Infine, con il comando *chart.Posn* è possibile vedere riassunto graficamente il TS per un singolo titolo evidenziando gli indicatori, i punti di entrata ed uscita, il guadagno o la perdita relativa al titolo e il massimo drawdown.

Figura 35: TS applicato al titolo Kellogg (K).



Fonte: Grafico generato con R-studio.

In figura 36 viene mostrato anche il percorso dell'equity globale durante tutta la strategia.

Figura 36: Equity totale.

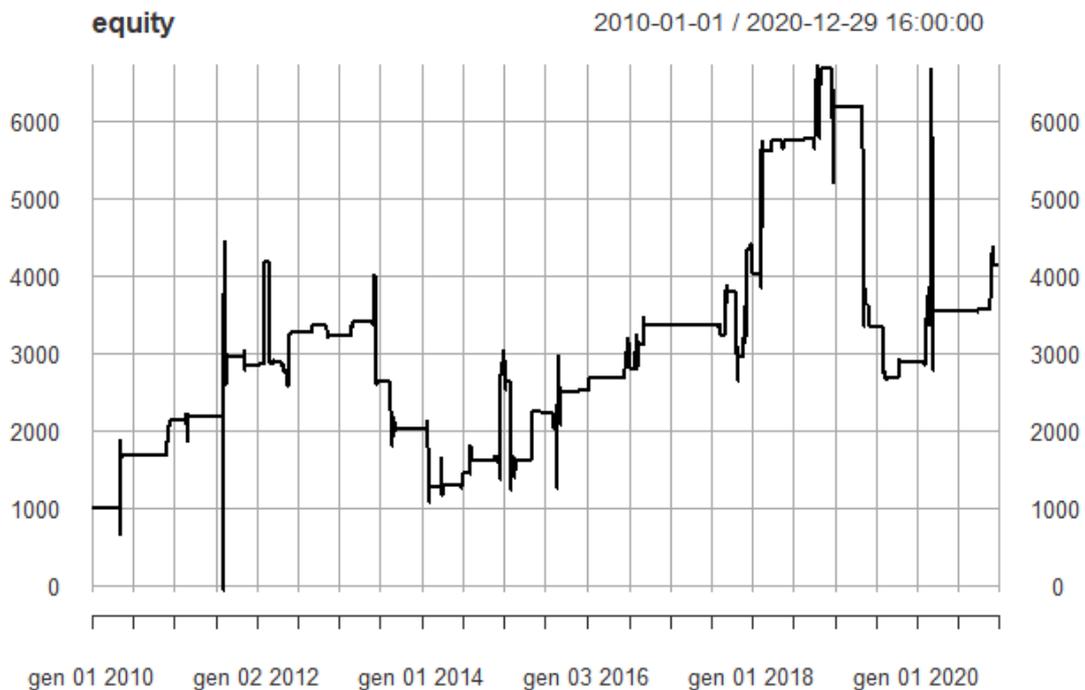
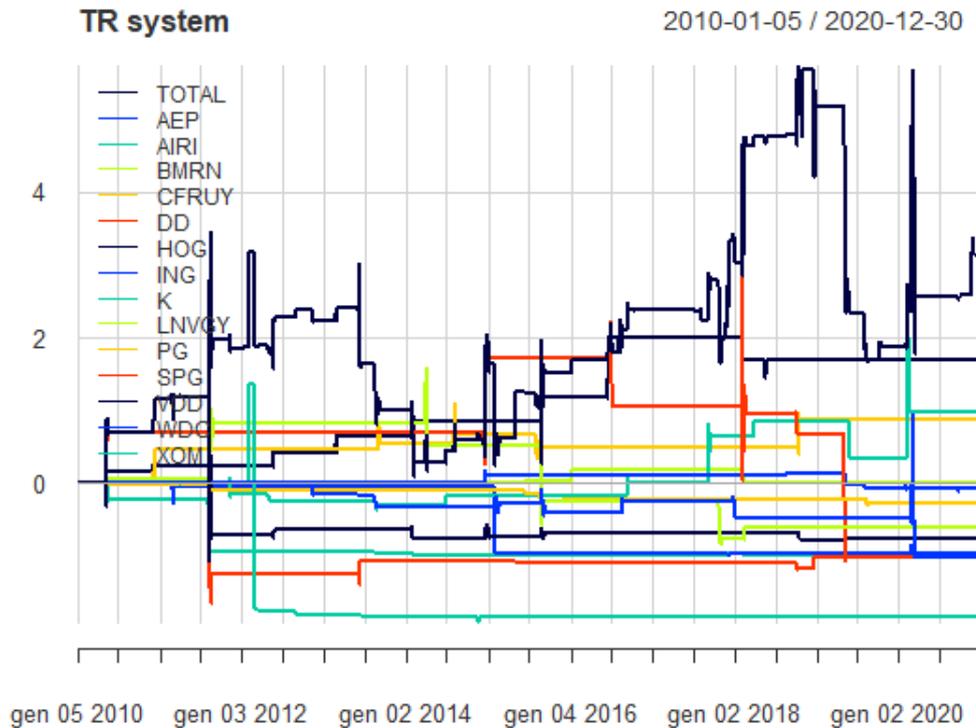


Figura 37: Equity totale comparato con l'equity di ogni operazione



Fonte: Grafici generato con R-studio.

### 3.3 Risultati del Trading System sulla serie storica originaria

Dopo aver descritto la costruzione del trading system vengono evidenziati i risultati ottenuti. Le Figure 38 e 39 riassumono tutte le statistiche relative agli scambi.

Le informazioni più importanti sono sicuramente l'equity finale e il massimo drawdown.

Figura 38: Tabella riassuntiva dei risultati ottenuti con il TS.

	AEP	AIRI	BMRN	CFRUY	DD	HOG	ING
Num.Txns	18	17	12	12	15	14	13
Num.Trades	9	8	6	6	7	7	6
Avg.Trade.PL	-51,55	22,00	140,67	-46,00	-43,19	-23,43	-3,67
Med.Trade.PL	-50,00	7,00	73,00	-48,00	200,00	148,00	2,00
Largest.Winner	558,00	92,00	838,00	18,00	1.384,26	368,00	112,00
Largest.Loser	-722,00	-4,00	-824,00	-116,00	-2.755,32	-536,00	-114,00
Gross.Profits	972,00	180,00	1.668,00	18,00	2.824,20	1.098,00	160,00
Gross.Losses	-1.436,00	-4,00	-824,00	-294,00	-3.126,52	-1.262,00	-182,00
Std.Dev.Trade.PL	362,59	33,56	588,17	50,09	1.315,28	382,14	76,57
Std.Err.Trade.PL	120,86	11,87	240,12	20,45	497,13	144,43	31,26
Percent.Positive	33,33	87,50	83,33	33,33	57,14	57,14	50,00
Percent.Negative	66,67	12,50	16,67	66,67	42,86	42,86	50,00
Profit.Factor	0,68	45,00	2,02	0,06	0,90	0,87	0,88
Ann.Sharpe	-2,26	10,40	3,80	-14,58	-0,52	-0,97	-0,76
Max.Drawdown	-3.456,00	-1.310,00	-1.486,00	-328,00	-3.035,74	-1.218,00	-274,00
Max.Equity	1.642,00	1.342,00	1.164,00	14,00	2.519,42	506,00	176,00
Min.Equity	-1.814,00	-2,00	-322,00	-314,00	-516,32	-712,00	-98,00
End.Equity	-482,00	158,00	832,00	-288,00	-318,32	-178,00	-36,00

Fonte: Grafico generato con R-studio.

Figura 39: Tabella riassuntiva dei risultati ottenuti con il TS.

	K	LNVGY	PG	SPG	VOD	WDC	XOM
Num.Txns	18	8	10	10	18	14	19
Num.Trades	9	4	5	5	9	6	9
Avg.Trade.PL	116,44	7,00	162,80	333,28	131,52	103,00	-200,67
Med.Trade.PL	128,00	11,00	158,00	628,00	154,98	122,00	-378,00
Largest.Winner	566,00	168,00	406,00	1.580,31	338,00	924,00	634,00
Largest.Loser	-294,00	-162,00	-96,00	-733,89	-70,00	-1.156,00	-856,00
Gross.Profits	1.618,00	200,00	910,00	2.908,31	1.253,67	1.906,00	1.100,00
Gross.Losses	-570,00	-172,00	-96,00	-1.241,89	-70,00	-1.288,00	-2.906,00
Std.Dev.Trade.PL	291,18	135,89	192,08	951,71	118,65	738,98	485,83
Std.Err.Trade.PL	97,06	67,94	85,90	425,62	39,55	301,69	161,94
Percent.Positive	55,56	50,00	80,00	60,00	88,89	66,67	33,33
Percent.Negative	44,44	50,00	20,00	40,00	11,11	33,33	66,67
Profit.Factor	2,84	1,16	9,48	2,34	17,91	1,48	0,38
Ann.Sharpe	6,35	0,82	13,45	5,56	17,60	2,21	-6,56
Max.Drawdown	-520,00	-176,00	-418,00	-1.702,00	-334,00	-1.774,00	-2.328,00
Max.Equity	1.418,00	196,00	856,00	2.388,42	1.387,68	1.236,00	144,00
Min.Equity	-426,00	0,00	0,00	-1.138,41	-10,16	-1.686,00	-2.184,00
End.Equity	1.030,00	20,00	804,00	1.656,42	1.165,67	602,00	-1.826,00

Fonte: Grafico generato con R-studio.

Ad esempio, prendendo in considerazione il titolo Kellogg possiamo notare che il TS ha portato alla generazione di 9 scambi completi con un totale di 18 transazioni (ogni scambio include una o più transazioni di acquisto e una transazione di vendita). Poi vengono indicate la media e la mediana dei profitti o delle perdite delle operazioni. In questo caso sono positive, pari rispettivamente a 116,44\$ e 128,00\$. Successivamente si elencano il maggior profitto e la maggiore perdita di una transazione, che per Kellogg sono state 566,00\$ e -294,00\$. Altro dato rilevante sono i profitti lordi, ovvero prima delle tasse, e le perdite che includono anche i costi operativi, pari a 1618,00\$ e -570,00\$. Vengono poi indicate la deviazione standard e l'errore standard, pari a 291,18 e 97,06. La percentuale positiva (55,56) e negativa (44,44) descrivono quante operazioni sono state positive o negative, informazione che presa singolarmente può indicare ben poco in quanto non tiene conto del valore delle operazioni. Il Profit Factor, in questo caso pari a 2,84, è dato dal valore assoluto del rapporto tra i profitti e le perdite lorde e mette in relazione il profitto ottenuto con un'unità di rischio. È dunque un indicatore della qualità del trading. Lo Sharpe ratio, invece, descrive quanto rendimento aggiuntivo ricevi rispetto al rendimento di un'attività priva di rischio per la volatilità maggiore sopportata del titolo rischioso. Nella tabella è mostrato lo Sharpe ratio annualizzato, pari a 6,35. Il drawdown rappresenta la perdita massima realizzata in tutto il periodo, in questo caso -520,00\$. Questo dato è rilevante perché evidenzia la perdita che un investitore deve riuscire a sopportare, sia in termini finanziari che, soprattutto, in termini psicologici. Spesso perdite consistenti del proprio patrimonio risultano psicologicamente insostenibili anche se l'equity finale risulta a posteriori positivo. Per questo è molto importante che in un trading automatizzato una volta stabilita la strategia la si segua assiduamente intervenendo il meno possibile. L'equity massimo, minimo e finale, rispettivamente di 1418,00\$, -426,00\$ e 1030,00\$ indicano il percorso che ha fatto il patrimonio evidenziando il suo punto di massimo, di minimo e quello al termine del periodo di valutazione. Per gli altri 13 titoli le considerazioni sono le stesse.

Il trading system porta ad ottenere un profitto al netto dell'investimento iniziale di *3139,77 dollari*, con un rendimento in termini percentuali del 313,98% in 10 anni. Una semplice strategia buy & hold applicata all'indice Standard & Poor's nello stesso periodo avrebbe portato ad un profitto del 237,25%.

## Capitolo IV: La previsione con le reti neurali

Per quanto riguarda la rete neurale con la quale viene effettuata la previsione è stato utilizzato il modello di rete neurale ricorrente specificato nei capitoli precedenti, ovvero quello che utilizza come nodo la LSTM unit. La scelta del numero di nodi e dei layer nascosti si è basata essenzialmente su due fattori rilevanti: l'overfitting e la velocità di calcolo.

L' *overfitting* o sovradattamento avviene quando un modello, in questo caso la rete neurale, si adatta troppo bene ai dati osservati ma non riesce a generalizzare quando poi viene testato, questo spesso accade perché c'è un numero eccessivo di parametri rispetto alle osservazioni utilizzate. Viceversa, con un numero troppo basso di parametri utilizzati si può incorrere nell'*underfitting* e il modello soffre di una eccessiva discrepanza (high bias) in quanto troppo semplice. In entrambi i casi il modello raggiunge un buon risultato sui dati di addestramento ma ha scarse performance sui dati di test. Per capire se c'è la presenza di overfitting o underfitting si deve verificare l'accuratezza dei dati sia su quelli di test sia su quelli di addestramento. Un errore alto nei dati del training test indica la presenza di underfitting, invece, se l'errore sui dati di training è accettabile ma l'errore sui dati di test è elevato c'è un problema di overfitting.

L'altro fattore rilevante è la *velocità di calcolo*, ovvero il tempo impiegato dalla rete per l'addestramento. Questa dipende dalla complessità della rete e dalla potenza di calcolo del processore utilizzato. Le reti neurali con molti nodi hanno chiaramente più capacità espressiva a discapito del tempo di calcolo impiegato e il rischio maggiore di overfitting. La previsione della serie storica è stata fatta utilizzando i dati dal 2010 al 2019 escludendo quindi il periodo del Covid come detto precedentemente.

### 4.1 Architettura del modello

Ogni modello ha diversi iper-parametri specifici, pensati in modo che si adattino alle caratteristiche del titolo valutato.

I parametri totali generati che saranno utilizzati per la previsione variano in base a quanti layer ci sono e a quanti nodi sono inseriti all'interno del layer stesso.

In seguito, un elenco di iper-parametri che vengono utilizzati nel periodo di validazione con i quali i modelli sono stati creati:

- L'input e l'output layer prevedono un solo nodo anche in seguito al fatto che i dati inseriti provengono da una sola variabile. Inoltre, l'input contiene anche i valori ritardati di cinque periodi e consiste nei prezzi di chiusura aggiustati (*Adj Close*) dei titoli.
- Numero di epoche. È il numero di iterazioni che il modello fa durante l'addestramento. Il numero di epoche è stato scelto in base al fatto che superata una certa soglia la funzione di perdita si stabilizzava o scendeva molto lentamente e ciò significa che ulteriori ripetizioni non miglioravano il modello. La scelta delle epoche è una variabile fondamentale nella costruzione di una rete, ma nel nostro caso, essendo un modello semplificato, ulteriori indagini non sarebbero state inerenti allo scopo di questa tesi.
- Dropout rate. Il dropout layer è inserito per migliorare la generalizzazione della rete, in fase di addestramento "spegne" in maniera casuale un numero di connessioni pari alla percentuale indicata.
- Attivazione. Funzione di attivazione dei nodi.
- Funzione di perdita. La funzione di perdita utilizzata.
- Funzione di ottimizzazione (solitamente quella utilizzata è la "adam"). Essa determina come verrà aggiornata la rete in base alla funzione di perdita. Adam (Adaptive moment estimation) è una variante della funzione di discesa stocastica del gradiente e in particolare un'estensione della variante RMSProp (Root Mean Square Propagation) che si basa su un metodo con tasso di apprendimento adattivo.
- Metrics. È la funzione utilizzata per giudicare la performance del modello. Le funzioni di metrica sono simili alle funzioni di perdita, tranne per il fatto che i risultati della valutazione non vengono utilizzati durante l'addestramento del modello.

A questo punto posso inserire le immagini di overfitting e loss function dei titoli.

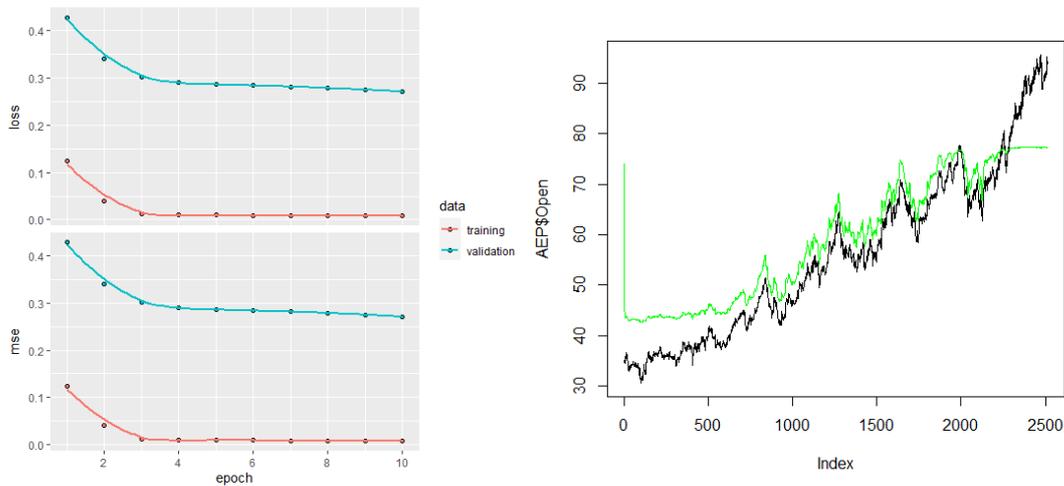
La prima immagine di ogni titolo è composta dalla funzione di perdita e dalla funzione metrica per la valutazione del modello; più le due funzioni tendono ad allontanarsi più significa che c'è presenza di overfitting, viceversa, indicano una buona stima del modello.

La seconda immagine, invece, sovrappone la serie storica del titolo (in nero) con quella

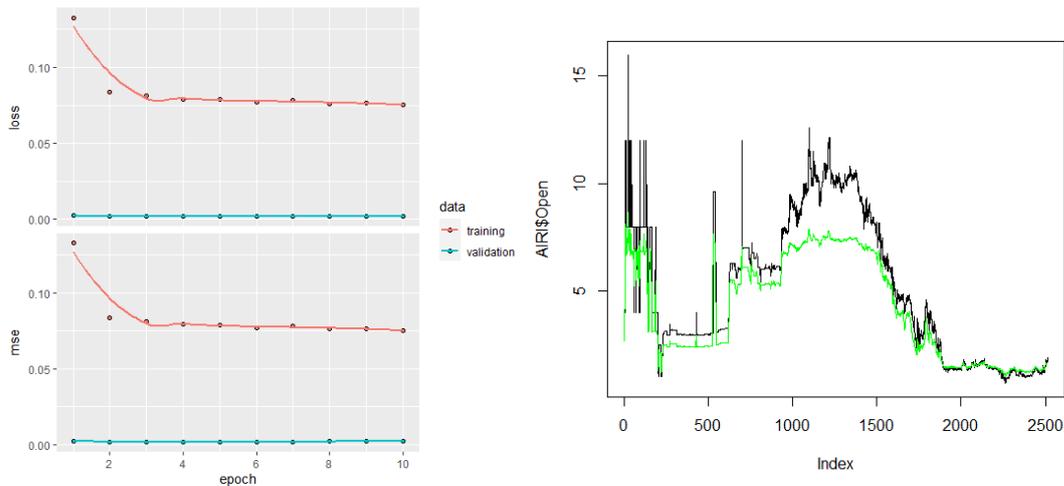
stimata dalla rete neurale (in verde), alla quale verrà poi applicato il trading system. Questo poi permetterà di valutare la performance migliore o peggiore di quella reale. Chiaramente le immagini sono indicative in quanto la rete se venisse ripetuta altre volte sugli stessi dati genererebbe dei risultati diversi ogni volta, le immagini servono solo ad evidenziare la capacità della rete di generalizzare il modello.

I comandi relativi alla preparazione dei data set, la costruzione dei modelli e le previsioni si trovano nell'Appendice B a fine tesi. Sono stati utilizzati i pacchetti Keras e Tensorflow, relativi all'ambiente Python.

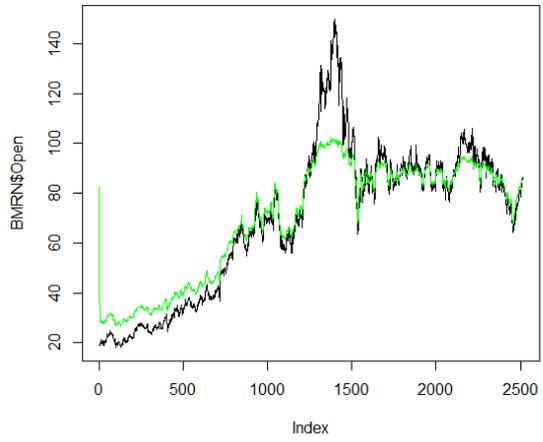
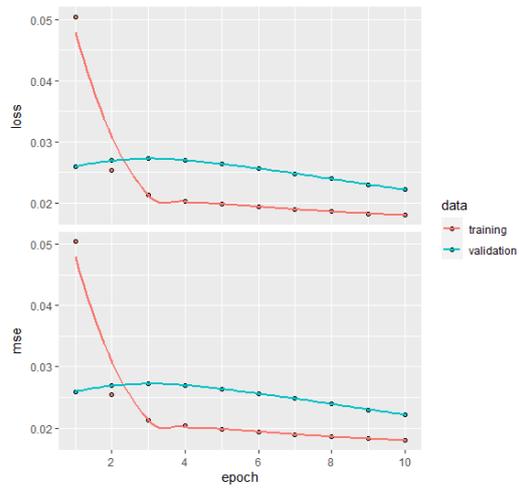
### American Electric Power (AEP)



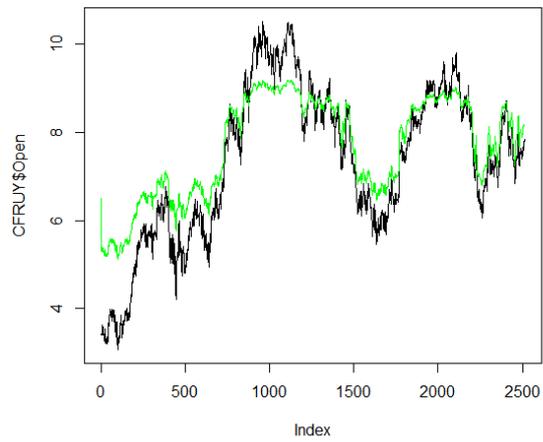
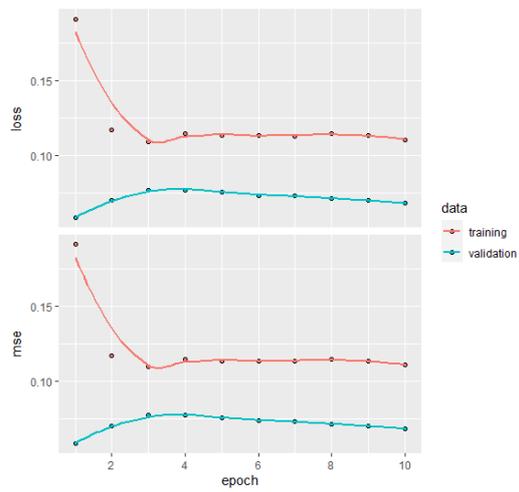
### Air Industries (AIRI)



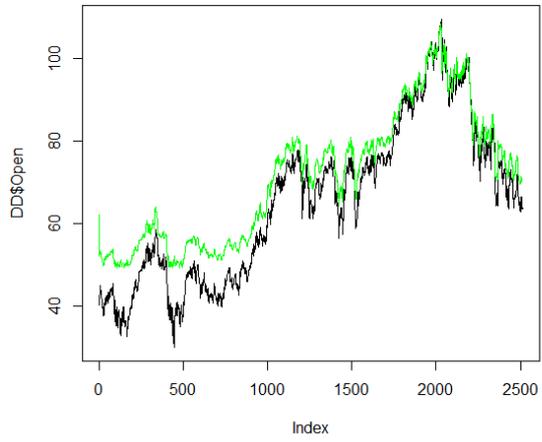
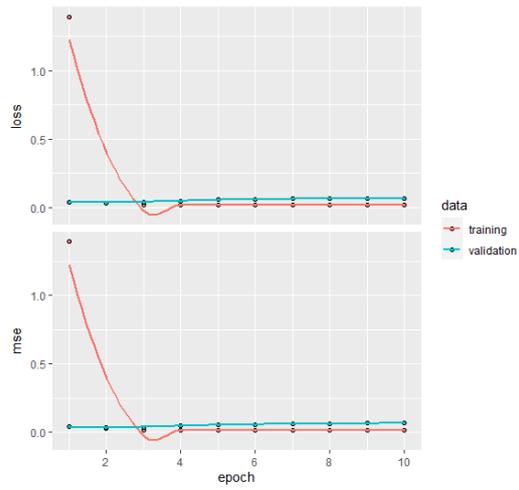
## BioMarin Pharmaceutical (BMRN)



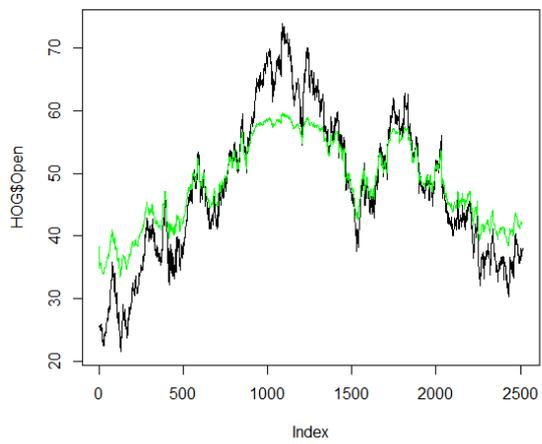
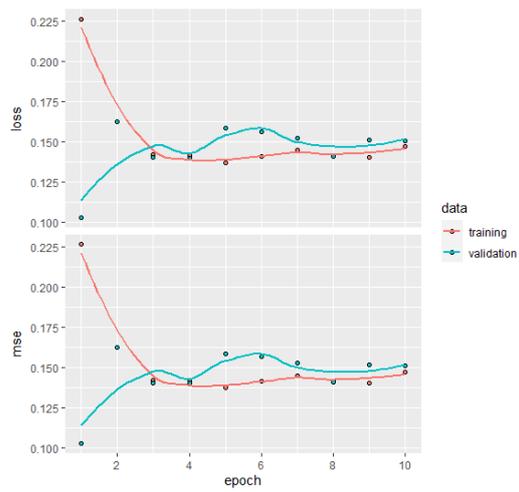
## Richemont (CFRUY)



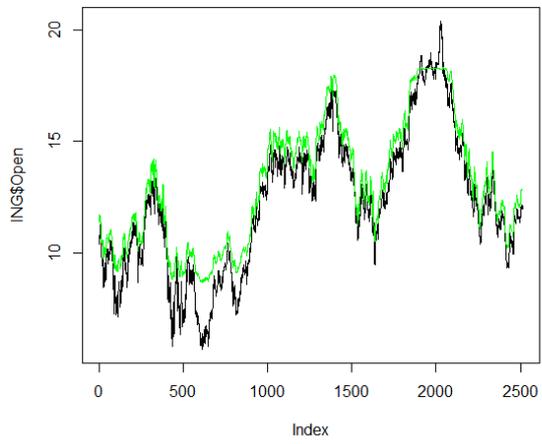
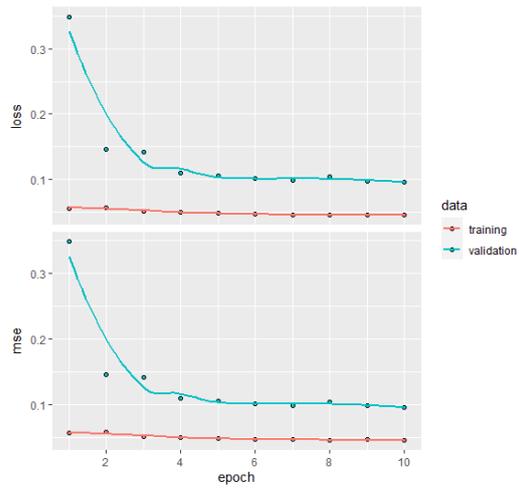
## Dupont (DD)



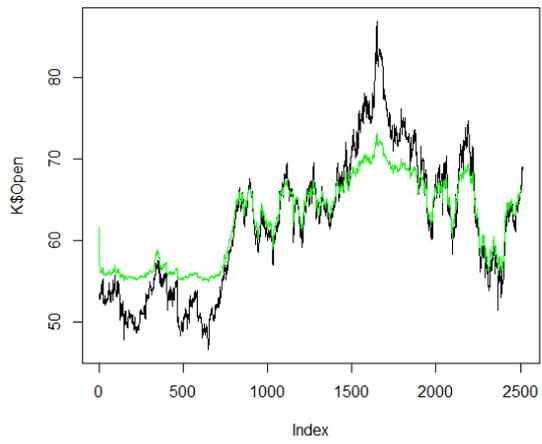
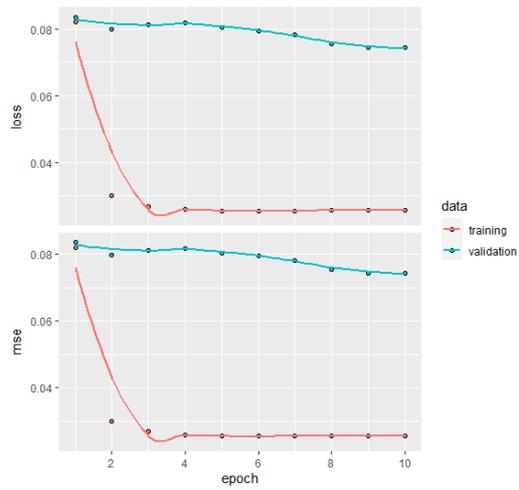
## Harley-Davidson (HOG)



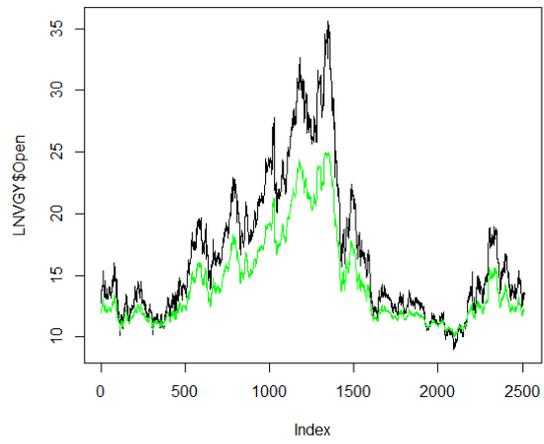
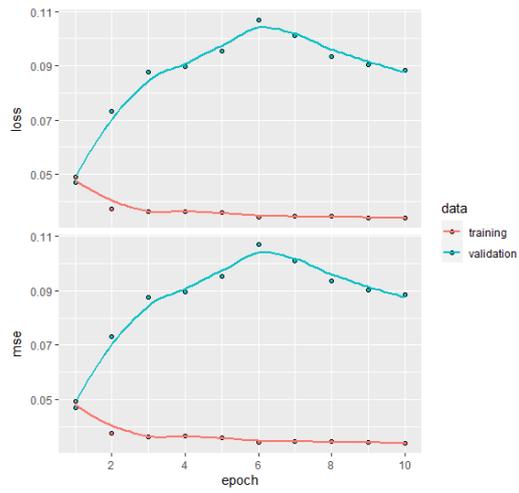
## ING Group (ING)



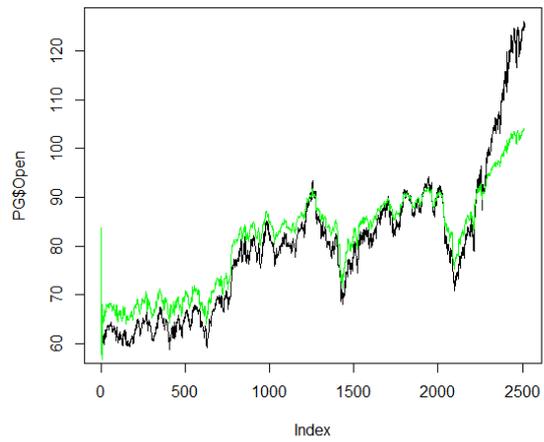
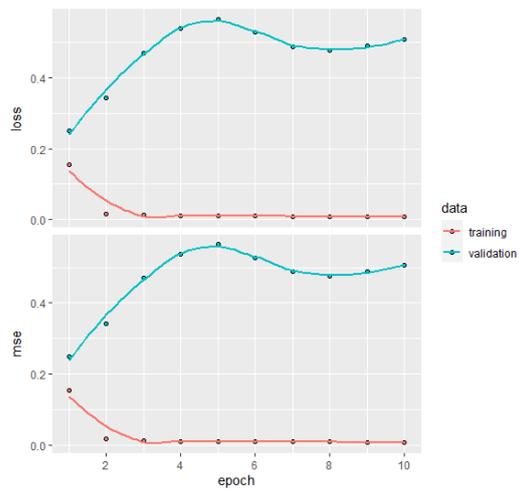
## Kellogg (K)



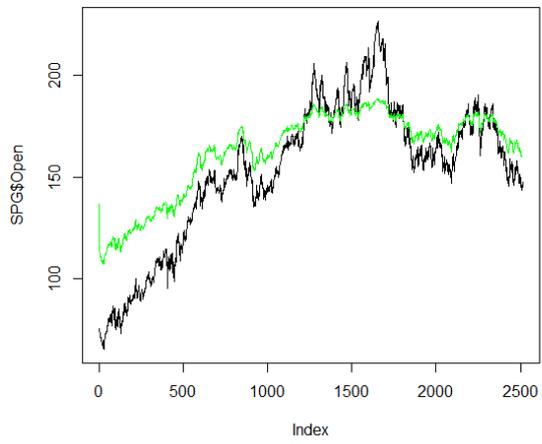
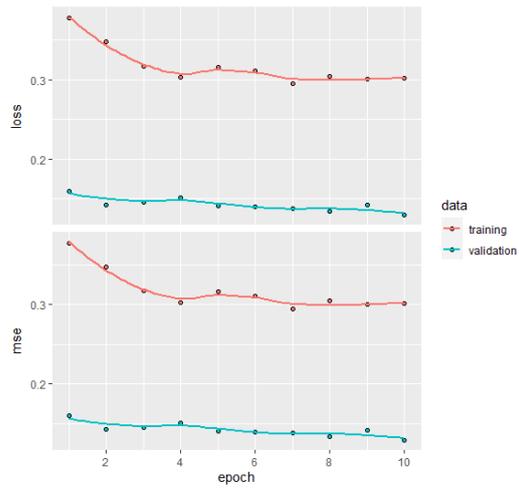
## Lenovo (LNVGY)



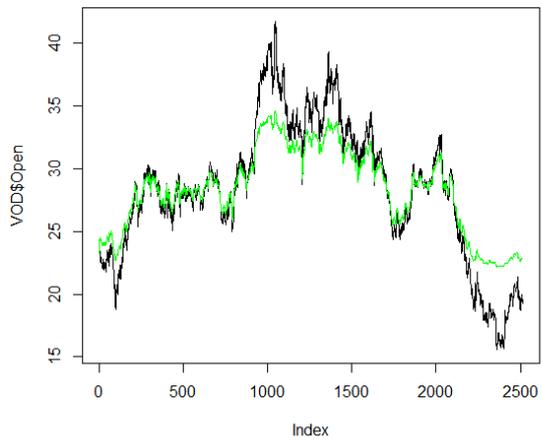
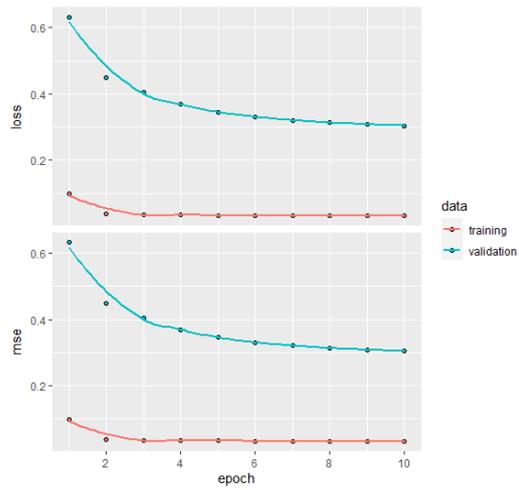
## The Procter & Gamble (PG)



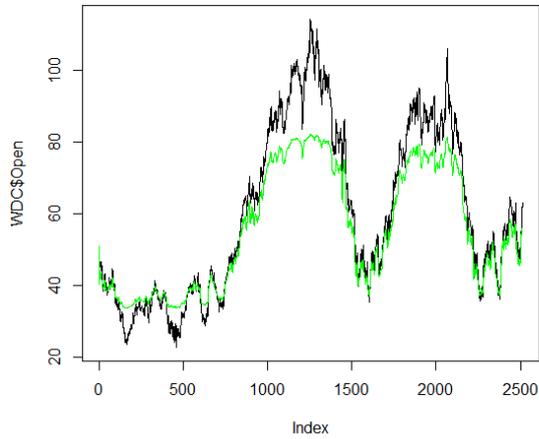
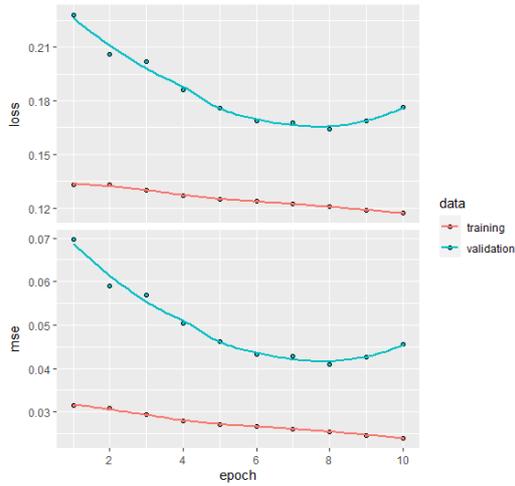
## Simon Property Group (SPG)



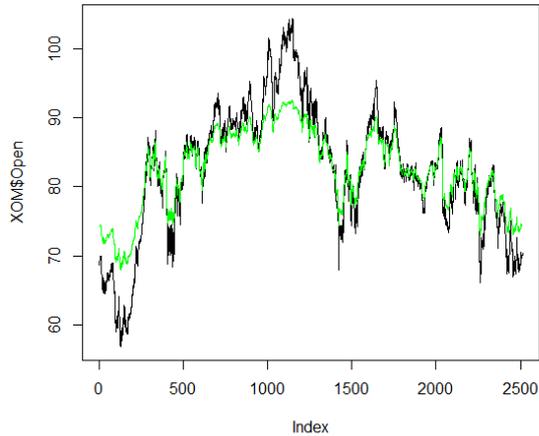
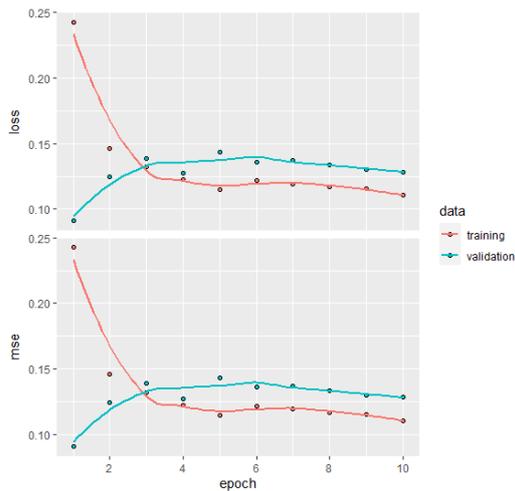
## Vodafone (VOD)



## Western Digital Corporation (WDC)



## Exxon Mobil (XOM)



Per esempio, come si può vedere nel grafico relativo al titolo Dupont (DD), i valori delle curve del training set e del validation set non discostano molto e si avvicinano attorno alla quarta epoca, questa è un'indicazione che non c'è overfitting nel modello. Inoltre, la funzione di perdita si stabilizza già dalla quarta epoca, segno che sta a evidenziare che dieci epoche di iterazione sono più che sufficienti.

Viceversa, per il titolo The Procter & Gamble (PG) si può notare che sarebbero state necessarie molte più epoche per spiegare i dati utilizzati nell'addestramento perché le curve devono ancora stabilizzarsi e allo stesso tempo i valori del validation set alla decima epoca sono ancora ben distanti da quelli del training set indicando la presenza di

overfitting. Come già detto precedentemente ulteriori approfondimenti per migliorare le architetture delle reti neurali possono essere fatti, ma un modello con maggiori capacità computazionali è superfluo per lo scopo di questa tesi.

Quello che sarà rilevante capire nel prossimo capitolo non è tanto se le reti neurali abbiano migliorato o peggiorato il profitto del Trading System quanto piuttosto se abbiano rispettato la variazione avvenuta nei prezzi originali.

## Capitolo V: Trading system applicato alla previsione della serie storica

In questo capitolo verranno mostrati i risultati del Trading System precedentemente descritto applicato alle serie storiche create attraverso l'uso delle reti neurali del capitolo IV. Quello che è stato fatto sostanzialmente è stato prendere l'operatività generata dal trading system sulle reti neurali, e utilizzare i prezzi "reali", ovvero quelli della serie storica originale, per calcolare i profitti e le perdite della strategia.

In Tabella 4 viene mostrata la costruzione del foglio di calcolo che è stata fatta per ogni titolo.

Tabella 3: Trading System sulla Previsione del titolo *LNVGY*

<i>LNVGY</i>	Data	Previsione	Reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	10/06/2015	22,48	29,13	-2	200	LONG	- 5.828,00	30,00	UGUALE
2	12/06/2015	22,29	29,30	-2	-200	SHORT	5.858,00		
3	07/01/2016	14,83	17,72	-2	200	LONG	- 3.546,00	166,00	
4	08/01/2016	15,27	18,57	-2	-200	SHORT	3.712,00		
5	07/07/2017	11,55	12,20	-2	200	LONG	- 2.442,00	- 12,00	
6	10/07/2017	11,53	12,16	-2	-200	SHORT	2.430,00		
7	07/02/2018	10,73	10,57	-2	200	LONG	- 2.116,00	- 164,00	
8	13/02/2018	10,36	9,77	-2	-200	SHORT	1.952,00		
								20,00	20,00
OPERAZIONI		4							

Vengono riportati il numero delle operazioni e delle relative transazioni, le date nelle quali è stata effettuata l'operazione (ovvero le date valide immediatamente successive alla data nella quale è stato generato il segnale), il prezzo di previsione e il prezzo reale applicato. Il costo è sempre pari all'1% della quantità scambiata. Nella colonna Transazioni vengono evidenziati i costi nel caso si assuma una posizione "long", o i ricavi, nel caso si chiuda la posizione ("short"). Sotto la colonna Operazione la cella viene evidenziata in verde se l'operazione ha avuto un profitto e in rosso se alla chiusura si ha subito una perdita. Infine, vengono comparati il totale della strategia applicata ai segnali generati in base ai prezzi della previsione e il totale della strategia applicato al titolo originale. In questo caso si può vedere che il valore è uguale.

In particolare, sui 14 titoli analizzati solo 4 sono risultati con un'operatività diversa rispetto a quella originale, dovuta dall'esclusione del 2020 come anno di calcolo. La rete

neurale non è stata capace di anticipare il movimento del prezzo e dare, quindi, indicazioni diverse.

Quello che però è rilevante evidenziare è se i prezzi generati dalla rete neurale abbiano rispettato l'andamento di quelli originali. In particolare, se nel Trading System originale il prezzo di acquisto con il quale è stata effettuata la transazione è più alto rispetto a quello di chiusura dell'operazione, allora anche i prezzi generati dalla rete neurale devono rispettare la stessa "regola". Come possiamo vedere nella Tabella 3, per il titolo LNVGY, la prima operazione ha avuto come prezzi di acquisto e di chiusura, rispettivamente 29,13 e 29,30, ovvero, il primo più basso rispetto al secondo, ma nella previsione i prezzi sono rispettivamente 22,48 e 22,29, cioè il prezzo di acquisto più alto rispetto a quello di chiusura. In generale, possiamo vedere che su 90 operazioni eseguite 9 non hanno rispettato lo stesso andamento di prezzo, solo il 10%, come evidenziato in Tabella 4.

Si può dire, quindi, che la serie storica generata dalla rete neurale è stata affidabile nel creare una serie di prezzi che replicasse l'andamento di quella originale, e questo è chiaramente visibile sia da quanto appena detto sia dalle immagini presenti nel capitolo III della tesi che mostrano le due serie a confronto, ma non è riuscita a prevedere i movimenti del prezzo e ad anticipare quindi l'operatività, di fatto i segnali sono pressoché identici a quelli originali sotto il punto di vista temporale.

In Appendice C è possibile vedere l'operatività di ogni singolo titolo e un riassunto finale delle strategie.

In Tabella 4, inoltre, vengono comparati il TS originale e quello fatto sulla previsione, evidenziando il numero delle operazioni e delle transazioni totali, i relativi costi e infine il profitto derivante dalle due strategie.

Tabella 4: Riepilogo Trading System

	TS originale	TS previsione
<b>N° operazioni</b>	96	90
<b>N° transazioni</b>	198	186
<b>Costi operativi</b>	-408	-384
<b>PROFITTO</b>	<b>3.139,77</b>	<b>1.885,77</b>

<b>N° di operazioni che non hanno rispettato lo stesso andamento di prezzo</b>	<b>9</b>
--	----------

Transazioni escluse		
Simbolo	Data	Prezzo
AEP	27/02/2020	90,03
AEP	12/03/2020	86,43
AEP	19/11/2020	81,79
AEP	03/12/2020	84,59
ING	26/01/2020	11,00
ING	02/02/2020	10,91
ING	22/09/2020	6,96
ING	30/09/2020	7,09
K	06/02/2020	63,26
K	19/02/2020	65,78
WDC	27/02/2020	52,25
WDC	04/03/2020	56,88

Com'è possibile notare la previsione ha escluso sei operazioni presenti nel trading system originale, elencate per titolo e data di esecuzione, passando da 96 a 90 operazioni totali. Inoltre, la strategia, rispetto a quella applicata alle serie storiche originali, ha conseguito un minor profitto, 1.885,77 \$, a confronto di 3.139,77 \$ con una differenza di 1.254,00 \$ di mancato guadagno dell'ultimo anno.

Questo deriva dal fatto che dei quattro titoli che hanno avuto un'operatività diversa, solo uno è stato più profittevole grazie all'esclusione di alcune sue transazioni, mentre due hanno performato peggio e uno uguale al titolo originale. Delle sei operazioni escluse due hanno portato ad una perdita e quattro ad un profitto. Evidentemente l'operatività avvenuta nell'ultimo anno di valutazione era stata profittevole, anche in presenza di un evento anomalo come il Covid -19.

Questo di fatto è un punto a favore del Trading System, riuscire ad avere ugualmente un profitto conferma gli obiettivi prestabiliti e sottolinea che anche in periodi di mercato negativi e con alta volatilità la strategia costruita riesce a cogliere abbastanza tempestivamente i segnali da generare transazioni profittevoli.

Inoltre, come si può vedere dalla Figura 36 dell'Equity totale l'andamento del portafoglio non ha avuto grandi drawdown, ciò significa che il profitto conseguito è stato accumulato nel tempo e non la fortuna di un'operazione. Questo dato è molto rilevante non tanto se la strategia è automatizzata, ma più nel caso l'operatività sia eseguita e monitorata manualmente dal soggetto incaricato. Quando entra in gioco il fattore psicologico, subire ingenti perdite può spaventare a tal punto da cambiare la strategia scelta, e ugualmente, grandi profitti possono spingere ad investire di più del dovuto e assumere un rischio che inizialmente non era previsto, per questo eliminare il più possibile l'intervento umano nell'operatività è tanto importante, perché esclude la possibilità di errore.

Ad ogni modo la previsione creata ha parzialmente raggiunto gli obiettivi che ci eravamo posti ad inizio tesi, ovvero quelli di replicare fedelmente la serie storica.

Inoltre, c'è da considerare che le reti neurali non sono state costruite attraverso un percorso di ottimizzazione delle stesse, ma semplicemente a "tentativi" ovvero valutando manualmente quale sembrava adattarsi meglio; un'ulteriore analisi più approfondita di questo aspetto potrebbe quindi portare a risultati migliori.

## Conclusioni

L'obiettivo di questa tesi è stato quello di creare un Trading System automatizzato che seguisse delle logiche e dei comandi assegnati, e poi, applicato ad una serie di titoli, portasse ad un profitto finale. Conseguentemente a questo obiettivo ci eravamo posti quello di creare una previsione della serie storica che fosse la più simile possibile a quella originale nel tentativo che anticipasse i movimenti per anticipare a nostra volta l'operatività del TS.

In particolare, si è partiti con una disamina generale dell'analisi tecnica e delle reti neurali andando ad analizzare le diverse tipologie, costruzioni e applicazioni di entrambi gli aspetti.

Dall'analisi dei risultati esposti nel capitolo III possiamo affermare che il modello di Trading System costruito ha portato ad un risultato positivo, sia in termini di profitto, che in termini di applicabilità nel mercato azionario, l'obiettivo è stato ampiamente raggiunto. Tra il 2010 e il 2020, con un investimento iniziale di 1.000,00 dollari ne sono stati guadagnati 3.139,77, analizzando 14 titoli azionari non correlati tra loro. In questo periodo di dieci anni siamo riusciti a stimare in alcuni casi l'andamento del prezzo, questo porta a concludere, che almeno parzialmente il prezzo riflette in maniera efficiente il mercato, uno dei presupposti dell'analisi tecnica che avevamo descritto all'inizio della tesi.

Un approfondimento che potrebbe essere fatto come possibile sviluppo dell'elaborato ipotizza l'applicazione di uno stop loss in caso la perdita raggiunga un determinato valore percentuale. In questo modo il rischio di subire grossi perdite viene evitato perché l'operazione viene immediatamente chiusa. Sarebbe da approfondire qual è il livello ideale di stop loss che permette di ridurre il rischio di un drawdown eccessivo.

Per quanto riguarda la previsione del prezzo tramite l'utilizzo delle reti neurali si può dire che gli obiettivi della ricerca sono stati raggiunti parzialmente. Nel Capitolo IV abbiamo creato le serie storiche tramite le reti neurali ed accostandole a quelle originali abbiamo visto che i prezzi generati hanno rispettato ampiamente l'andamento di quelli originali. Questo, inoltre, è confermato dai movimenti dei prezzi delle due serie, che solo nel 10% delle operazioni non hanno seguito la stessa direzione.

Se quindi la previsione della serie storica ha raggiunto il suo obiettivo in termini di replicabilità non si può dire lo stesso per quanto riguarda l'anticipazione di quella originale. Il Trading System generato sulla base della previsione delle serie storiche non

è riuscito ad anticipare i propri segnali portando quindi ad un'operatività identica a quella di partenza.

In questo caso uno sviluppo successivo potrebbe essere quello di ottimizzare i parametri della rete neurale, magari tramite l'utilizzo di tecniche simili come le Particle Swarm Optimization (PSO) e riallenarla ad intervalli prestabiliti, per esempio ogni settimana o ogni mese in modo che si adatti ad eventuali cambiamenti della serie.

È chiaro, quindi, che l'applicazione delle reti neurali in ambito finanziario e in particolare nell'analisi tecnica è estremamente interessante e può portare a risultati decisamente concreti se paragonati alle strategie standard. Tuttavia, nell'ambito dell'intelligenza artificiale, la sfida più difficile rimane quella di riuscire ad individuare la soluzione che meglio si adatta alle esigenze dell'indagine. In forza delle possibilità di approfondimento precedentemente descritte questo elaborato non deve essere considerato come un punto di arrivo, ma piuttosto, come uno spunto di partenza.

## Bibliografia

- Rajashree Dash, Pradipta Kishore Dash, (2016) “A hybrid stock trading framework integrating technical analysis with machine learning techniques”, *The Journal of Finance and Data Science* 2, 42 – 57;
- Leonardo Conegundes, Adriano C. Machado Pereira, (2020), “Beating the Stock Market with a Deep Reinforcement Learning Day Trading System”;
- Fagner A. de Oliveira, Cristiane N. Nobre, Luis E. Zárate, (2013), “Applying Artificial Neural Networks to prediction of stock price and improvement of the directional prediction index – Case study of PETR4, Petrobras, Brazil”, [www.elsevier.com](http://www.elsevier.com);
- Rohitash Chandraa, Shelvin Chand, (2016), “Evaluation of co-evolutionary neural network architectures for timeseries prediction with mobile application in finance”, *Applied Soft Computing*;
- Omer Berat Sezera, Murat Ozbayoglua, Erdogan Dogdu, (2017), “A Deep Neural-Network Based Stock Trading System Based on Evolutionary Optimized Technical Analysis Parameters”, *Procedia Computer Science* 114, 473–480;
- Chenjie Sang, Massimo Di Pierro, (2018), “Improving trading technical analysis with TensorFlow Long Short-Term Memory (LSTM) Neural Network”, *The Journal of Finance and Data Science* 5, 1-11;
- Dehua Zhang, Sha Lou, (2020), “The application research of neural network and BP algorithm in stock price pattern classification and prediction”, *Future Generation Computer Systems*;
- Jordan Ayala, Miguel García-Torres, José Luis Vázquez Noguera, Francisco Gómez-Vela, Federico Divina, (2021), “Technical analysis strategy optimization

using a machine learning approach in stock market indices”, *Knowledge-Based Systems*;

- Ajit Kumar Rout, P.K. Dash, Rajashree Dash, Ranjeeta Bisoi, (2015), “Forecasting financial time series using a low complexity recurrent neural network and evolutionary learning approach”, *Journal of King Saud University - Computer and Information Sciences*;
- Adil MOGHAR, Mhamed HAMICHE, (2020), “Stock Market Prediction Using LSTM Recurrent Neural Network”, *International Workshop on Statistical Methods and Artificial Intelligence (IWSMAI 2020)*;
- Omer Berat Sezera, Murat Ozbayoglu, Erdogan Dogdu, (2017), “A Deep Neural-Network Based Stock Trading System Based on Evolutionary Optimized Technical Analysis Parameters”, *Complex Adaptive Systems Conference with Theme: Engineering Cyber Physical Systems, CAS October 30 – November 1, 2017, Chicago, Illinois, USA*;
- Cheol-Ho Park, Scott H. Irwin, (2007), “WHAT DO WE KNOW ABOUT THE PROFITABILITY OF TECHNICAL ANALYSIS?”, *Journal of Economic Surveys*;
- Roberto Cervelló-Royo, Francisco Guijarro, Karolina Michniuk, (2015), “Stock market trading rule based on pattern recognition and technical analysis: Forecasting the DJIA index with intraday data”, *Expert Systems with Applications*;
- Alejandro Rodríguez-González, Ángel García-Crespo, Ricardo Colomo-Palacios, Fernando Guldrís Iglesias, Juan Miguel Gómez-Berbís, (2011), “CAST: Using neural networks to improve trading systems based on technical analysis by means of the RSI financial indicator”, *Expert Systems with Applications*;
- Murphy, J.J. (1997), *Analisi tecnica dei mercati finanziari*, Hoepli, Milano;
- Pring, M. (1995), *Analisi tecnica dei mercati finanziari*, McGraw-Hill, Milano;

- Charu C. Aggarwal, (2018), *Neural Networks and Deep Learning*, Springer, NY;
- Gallo C., (2007), *Reti Neurali Artificiali: Teoria ed Applicazioni*, Università degli Studi di Foggia;
- Pizzi C., *Quantitative Trading*, dispense del relatore;
- Bussolon S., (2019), *Analisi dei dati con R*;
- Mohammad Zoynul Abedin, M.Kabir Hassan, Petr Hajek, Mohammed Mohi Uddin, (2021), *The Essentials of Machine Learning in Finance and Accounting*.

## Sitografia

- Keras, <https://keras.io/api/>;
- Investopedia, <https://www.investopedia.com/>;
- Datascience, <https://datascience.eu/it/apprendimento-automatico/comprendione-delle-reti-lstm/>;
- Github, <https://atcold.github.io/pytorch-Deep-Learning/it/week02/02-1/>;
- MSCI, <https://www.msci.com/our-solutions/indexes/gics/>;
- Maggioli-Developers, <https://www.developersmaggioli.it/blog/le-reti-neurali-ricorrenti/>.

## Appendice A

Script relativo al trading system.

```
require(quantstrat)
require(ggpubr)
require(tidyverse)
require(Hmisc)
require(corrplot)

StartDate <- "2010-01-01"
EndDate <- "2020-12-31"
InitDate <- "2010-01-01"
InitEq <- 1000

symbols <- c("AEP", "AIRI", "BMRN", "CFRUY", "DD", "HOG", "ING", "K", "LNVGY", "PG", "S
PG", "VOD", "WDC", "XOM")
getSymbols(symbols, index.class=c("POSIXt", "POSIXct"), from=StartDate, to=EndDate, src='y
ahoo')

x <- data.frame((AEP[,6]),(AIRI[,6]),(BMRN[,6]),(CFRUY[,6]),(DD[,6]),(HOG[,6]),(ING[,6]),
(K[,6]),(LNVGY[,6]),(PG[,6]),(SPG[,6]),(VOD[,6]),(WDC[,6]),(XOM[,6]))
Corr<- cor(x)
corrplot(Corr, method = "number")

Sys.setenv(TZ="America/Los_Angeles")
currency("USD")
stock(symbols,currency = "USD",multiplier = 1)

nFast <- 20
nSlow <- 55
nRsi <- 14
HThreshold <- 70
LThreshold <- 30
orderqty <- 200
fee <- function(TxnQty, TxnPrice, Symbol) { return(-0.01 * abs(TxnQty))}

initPortf(name = "TrSys", symbols, initDate = InitDate)
initAcct(name = "TrSys", portfolios = "TrSys",initDate = InitDate, initEq = InitEq)
initOrders(portfolio = "TrSys", initDate = InitDate)
strategy("system", store = TRUE)

add.indicator("system", name = "SMA",
  arguments = list(x = quote(CI(mktdata)[,1]),n = nFast),
  label="nFast")

add.indicator("system", name="SMA",
  arguments = list(x = quote(CI(mktdata)[,1]),n = nSlow),
  label="nSlow")

add.indicator("system", name = "RSI",
  arguments = list(price = quote(CI(mktdata)), n = nRsi),
  label = "RSI")
```

```

add.signal("system", name = "sigComparison",
           arguments = list(columns = c("nFast", "nSlow"), relationship = "gte"),
           label = "long")

add.signal("system", name = "sigCrossover",
           arguments = list(columns = c("nFast", "nSlow"), relationship = "lt"),
           label = "short")

add.signal("system", name = "sigThreshold",
           arguments = list(column = "RSI", threshold = LThreshold, relationship = "lt", cross = TRUE),
           label = "longthreshold")

add.signal("system", name = "sigThreshold",
           arguments = list(column = "RSI", threshold = HThreshold, relationship = "gte", cross = TRUE),
           label = "shortthreshold")

add.signal("system", name = "sigFormula",
           arguments = list(formula = "long & longthreshold", cross = TRUE),
           label = "longentry")

add.rule("system", name = "ruleSignal",
        arguments = list(sigcol = "short", signal = TRUE,
                        orderqty = "all",
                        ordertype = "market",
                        orderside = "long",
                        replace = TRUE,
                        TxnFees = "fee",
                        prefer = "Open"),
        type = "exit",
        label = "Short")

add.rule("system", name = "ruleSignal",
        arguments = list(sigcol = "longentry", signal = TRUE,
                        orderqty = orderqty,
                        ordertype = "market",
                        orderside = "long",
                        replace = FALSE,
                        TxnFees = "fee",
                        prefer = "Open"),
        type = "enter",
        label = "long")

out <- applyStrategy("system", portfolios = "TrSys")

updatePortf("TrSys")
updateAcct("TrSys")
updateEndEq("TrSys")

chart.Posn("TrSys", "HOG")

```

```
sma20 <- SMA(x = Cl(HOG), n = 20)
sma55 <- SMA(x = Cl(HOG), n = 55)
rsi <- RSI(Cl(HOG), n=14)

add_TA(sma20, on = 1, col = "blue")
add_TA(sma55, on = 1, col = "red")
add_TA(rsi)

tstats<- t(tradeStats("TrSys"))
textplot(tstats)
view(tstats[c(3:4,6:16,25,26,30:32),1:14])

a <- getAccount("TrSys")
equity <- a$summary$End.Eq
plot(equity)

theAccount = getAccount("TrSys")
names(theAccount$summary)
xyplot(theAccount$summary)
```

## Appendice B

Script relativo alla costruzione delle reti neurali di ogni titolo. Nella sezione “GENERALE” possiamo vedere i comandi relativi alla manipolazione dei dati per poter poi essere utilizzati dalla rete neurale. Nelle successive sezioni vengono evidenziati i comandi e i relativi parametri specifici utilizzati per il titolo. La parte dello script che non è presente significa che rimane invariata per ciascun titolo.

```
library(keras)
library(tensorflow)
library(ggplot2)
library(readr)
library(dplyr)

# GENERALE #

XOM <- read_csv("TITOLI/XOM19.csv")

scale_factors <- c(mean(XOM$`Adj Close`), sd(XOM$`Adj Close`))
scaled_train <- XOM %>%
  dplyr::select(`Adj Close`) %>%
  dplyr::mutate(`Adj Close` = (`Adj Close` - scale_factors[1]) / scale_factors[2])

prediction <- 5
lag <- prediction

scaled_train <- as.matrix(scaled_train)

x_train_data <- t(sapply(
  1:(length(scaled_train) - lag - prediction + 1),
  function(x) scaled_train[x:(x + lag - 1), 1]
))
x_train_arr <- array(
  data = as.numeric(unlist(x_train_data)),
  dim = c(
    nrow(x_train_data),
    lag,
    1
  )
)

y_train_data <- t(sapply(
  (1 + lag):(length(scaled_train) - prediction + 1),
  function(x) scaled_train[x:(x + prediction - 1)]
))
y_train_arr <- array(
  data = as.numeric(unlist(y_train_data)),
  dim = c(
    nrow(y_train_data),
    prediction,
    1
  )
)
```

```

)
)
x_test <- XOM$`Adj Close`[(nrow(scaled_train) - prediction + 1):nrow(scaled_train)]
x_test_scaled <- (x_test - scale_factors[1]) / scale_factors[2]
x_pred_arr <- array(
  data = x_test_scaled,
  dim = c(
    1,
    lag,
    1
  )
)

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 8,
             activation = 'relu',
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dropout(rate = 0.2) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
          optimizer = 'adam',
          metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(
  x = x_train_arr,
  y = y_train_arr,
  batch_size = 1,
  epochs = 10,
  verbose = 1,
  shuffle = FALSE,
  validation_split = 0.2)

plot(history)

lstm_forecast <- lstm_model %>%
  predict(x_pred_arr, batch_size = 1) %>%
  .[, , 1]

lstm_forecast <- lstm_forecast * scale_factors[2] + scale_factors[1]
fitted <- predict(lstm_model, x_train_arr, batch_size = 1) %>%
  .[, , 1]

if (dim(fitted)[2] > 1) {
  fit <- c(fitted[, 1], fitted[dim(fitted)[1], 2:dim(fitted)[2]])
} else {
  fit <- fitted[, 1]
}

```

```

fitted <- fit * scale_factors[2] + scale_factors[1]

FIT_XOM <- data.frame(fitted)
plot(XOM$`Adj Close`, type = "l")
lines(FIT_XOM, col="green")

write.csv(FIT_XOM, file = "XOM_FITTED.csv")
XOM_FITTED <- read_csv("TITOLI/IMMAGINI/XOM/XOM_FITTED.CSV")

# AEP #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 4,
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
         optimizer = 'adam',
         metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(
  x = x_train_arr,
  y = y_train_arr,
  batch_size = 1,
  epochs = 10,
  verbose = 1,
  shuffle = FALSE,
  validation_split = 0.2)

# AIRI #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 8,
            activation = 'relu',
            batch_input_shape = c(1, 5, 1),
            return_sequences = TRUE,
            stateful = TRUE) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
         optimizer = 'adam',
         metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(

```

```

x = x_train_arr,
y = y_train_arr,
batch_size = 1,
epochs = 10,
verbose = 1,
shuffle = FALSE,
validation_split = 0.2)

# BMRN #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 4,
             activation = 'relu',
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
          optimizer = 'adam',
          metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(
  x = x_train_arr,
  y = y_train_arr,
  batch_size = 1,
  epochs = 10,
  verbose = 1,
  shuffle = FALSE,
  validation_split = 0.2)

# CFRUY #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 8,
             activation = 'relu',
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dropout(rate = 0.2) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
          optimizer = 'adam',
          metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(

```

```

x = x_train_arr,
y = y_train_arr,
batch_size = 1,
epochs = 10,
verbose = 1,
shuffle = FALSE,
validation_split = 0.2)

# DD #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 6,
             activation = 'relu',
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
          optimizer = 'adam',
          metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(
  x = x_train_arr,
  y = y_train_arr,
  batch_size = 1,
  epochs = 10,
  verbose = 1,
  shuffle = FALSE,
  validation_split = 0.2)

# HOG #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 6,
             activation = 'relu',
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dropout(rate = 0.2) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
          optimizer = 'adam',
          metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(

```

```

x = x_train_arr,
y = y_train_arr,
batch_size = 1,
epochs = 10,
verbose = 1,
shuffle = FALSE,
validation_split = 0.2)

# ING #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 10,
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dropout(rate = 0.1) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
         optimizer = 'adam',
         metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(
  x = x_train_arr,
  y = y_train_arr,
  batch_size = 1,
  epochs = 10,
  verbose = 1,
  shuffle = FALSE,
  validation_split = 0.2)

# K #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 8,
             activation = 'relu',
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
         optimizer = 'adam',
         metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(
  x = x_train_arr,

```

```

y = y_train_arr,
batch_size = 1,
epochs = 10,
verbose = 1,
shuffle = FALSE,
validation_split = 0.2)

# LNVGY #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 8,
             activation = 'relu',
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
          optimizer = 'adam',
          metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(
  x = x_train_arr,
  y = y_train_arr,
  batch_size = 1,
  epochs = 10,
  verbose = 1,
  shuffle = FALSE,
  validation_split = 0.2)

# PG #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 6,
             activation = 'relu',
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
          optimizer = 'adam',
          metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(
  x = x_train_arr,
  y = y_train_arr,

```

```

batch_size = 1,
epochs = 10,
verbose = 1,
shuffle = FALSE,
validation_split = 0.2)

# SPG #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 4,
             activation = 'relu',
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dropout(rate = 0.35) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
          optimizer = 'adam',
          metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(
  x = x_train_arr,
  y = y_train_arr,
  batch_size = 1,
  epochs = 10,
  verbose = 1,
  shuffle = FALSE,
  validation_split = 0.4)

# VOD #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 4,
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
          optimizer = 'adam',
          metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(
  x = x_train_arr,
  y = y_train_arr,
  batch_size = 1,

```

```

epochs = 10,
verbose = 1,
shuffle = FALSE,
validation_split = 0.25)

# WDC #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 6,
             batch_input_shape = c(1, 5, 1),
             return_sequences = TRUE,
             stateful = TRUE) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'maevv',
         optimizer = 'adam',
         metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(
  x = x_train_arr,
  y = y_train_arr,
  batch_size = 1,
  epochs = 10,
  verbose = 1,
  shuffle = FALSE,
  validation_split = 0.05)

# XOM #

lstm_model <- keras_model_sequential()
lstm_model %>%
  layer_lstm(units = 8,
            activation = 'relu',
            batch_input_shape = c(1, 5, 1),
            return_sequences = TRUE,
            stateful = TRUE) %>%
  layer_dropout(rate = 0.2) %>%
  layer_dense(units = 1)

lstm_model %>%
  compile(loss = 'mse',
         optimizer = 'adam',
         metrics = 'mse')

summary(lstm_model)

history <- lstm_model %>% fit(
  x = x_train_arr,
  y = y_train_arr,
  batch_size = 1,
  epochs = 10,

```

```
verbose = 1,  
shuffle = FALSE,  
validation_split = 0.2)
```

## Appendice C

Trading system generato dalle reti neurali.

AEP	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	08/11/2012	46,48	42,07	-2	200	LONG	8.416,00		PEGGIORE
2	14/11/2012	46,13	41,37	-2	-200	SHORT	8.272,00	144,00	
3	29/05/2013	50,20	46,16	-2	200	LONG	9.234,00		
4	31/05/2013	50,04	45,97	-2	-200	SHORT	9.192,00	42,00	
5	16/08/2013	47,55	43,63	-2	200	LONG	8.728,00		
6	21/08/2013	47,20	42,77	-2	-200	SHORT	8.552,00	176,00	
7	10/02/2015	61,68	58,26	-2	200	LONG	11.654,00		
8	24/02/2015	62,71	58,99	-2	-200	SHORT	11.796,00	142,00	
9	26/08/2015	58,69	53,53	-2	200	LONG	10.708,00		
10	11/09/2015	58,30	53,29	-2	-200	SHORT	10.656,00	52,00	
11	17/08/2016	70,21	65,75	-2	200	LONG	13.152,00		
12	19/08/2016	70,62	67,11	-2	-200	SHORT	13.420,00	268,00	
13	27/12/2017	76,08	73,21	-2	200	LONG	14.644,00		
14	04/01/2018	75,31	71,68	-2	-200	SHORT	14.334,00	310,00	
OPERAZIONI								314,00	482,00

AIRI	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	29/03/2011	2,41	2,95	-2	200	LONG	592,00		UGUALE
2	05/04/2011	2,43	3,00	-2	-200	SHORT	598,00	6,00	
3	23/12/2011	2,41	2,95	-2	200	LONG	592,00		
4	29/02/2012	2,63	3,10	-2	200	LONG	622,00	50,00 €	
5	20/03/2012	2,53	3,17	-4	-400	SHORT	1.264,00		
6	16/08/2012	4,87	5,60	-2	200	LONG	1.122,00		
7	31/08/2012	5,33	6,07	-2	-200	SHORT	1.212,00	90,00	
8	07/12/2012	5,71	6,54	-2	200	LONG	1.310,00		
9	10/12/2012	5,70	6,55	-2	-200	SHORT	1.308,00	2,00	
10	18/06/2013	5,16	5,70	-2	200	LONG	1.142,00		
11	19/06/2013	5,26	5,79	-2	-200	SHORT	1.156,00	14,00	
12	30/08/2013	5,25	5,97	-2	200	LONG	1.196,00		
13	09/09/2013	5,29	6,00	-2	-200	SHORT	1.198,00	2,00	
14	19/11/2014	7,42	10,01	-2	200	LONG	2.004,00		
15	01/12/2014	7,39	10,00	-2	-200	SHORT	1.998,00	6,00	
16	07/10/2016	3,13	4,00	-2	200	LONG	802,00		
17	12/10/2016	3,09	4,04	-2	-200	SHORT	806,00	4,00	
OPERAZIONI								158,00	158,00

BMRN	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	07/05/2010	28,75	20,57	-2	200	LONG	4.116,00		UGUALE
2	12/05/2010	29,06	21,11	-2	-200	SHORT	4.220,00	104,00	
3	09/08/2011	32,60	24,33	-2	200	LONG	4.868,00		
4	24/08/2011	34,88	27,71	-2	-200	SHORT	5.540,00	672,00	
5	31/03/2014	70,71	66,66	-2	200	LONG	13.334,00		
6	04/04/2014	67,75	66,72	-2	-200	SHORT	13.342,00	8,00	
7	25/08/2015	100,49	124,74	-2	200	LONG	24.950,00		
8	26/08/2015	101,08	124,95	-2	-200	SHORT	24.988,00	38,00	
9	24/10/2017	85,90	86,63	-2	200	LONG	17.328,00		
10	03/11/2017	84,41	82,52	-2	-200	SHORT	16.502,00	826,00	
11	12/02/2018	83,91	81,10	-2	200	LONG	16.222,00		
12	16/02/2018	85,32	85,30	-2	-200	SHORT	17.058,00	836,00	
OPERAZIONI								832,00	832,00

CFRUY	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	10/05/2010	5,37	3,55	-2	200	LONG	712,00		UGUALE
2	11/05/2010	5,37	3,56	-2	-200	SHORT	710,00	2,00	
3	05/08/2011	6,47	5,65	-2	200	LONG	1.132,00		
4	09/08/2011	6,32	5,25	-2	-200	SHORT	1.048,00	84,00	
5	10/06/2015	8,57	8,41	-2	200	LONG	1.684,00		
6	17/06/2015	8,53	8,22	-2	-200	SHORT	1.642,00	42,00	
7	21/08/2015	8,04	7,75	-2	200	LONG	1.552,00		
8	24/08/2015	7,80	7,18	-2	-200	SHORT	1.434,00	118,00	
9	25/11/2015	7,85	7,36	-2	200	LONG	1.474,00		
10	27/11/2015	7,90	7,46	-2	-200	SHORT	1.490,00	16,00	
11	06/08/2019	8,37	7,92	-2	200	LONG	1.586,00		
12	20/08/2019	8,20	7,65	-2	-200	SHORT	1.528,00	58,00	
OPERAZIONI								288,00	288,00

DD	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	10/05/2010	50,40	38,49	-2	200	LONG	7.699,04		UGUALE
2	13/05/2010	51,00	41,37	-2	-200	SHORT	8.272,89	573,85	
3	15/12/2014	68,82	62,07	-2	200	LONG	12.415,76		
4	18/12/2014	71,34	65,40	-2	-200	SHORT	13.077,85	662,09	
5	29/06/2016	75,01	69,83	-2	200	LONG	13.967,12		
6	06/07/2016	74,52	68,70	-2	-200	SHORT	13.738,25	228,88	
7	06/02/2018	97,07	94,83	-2	200	LONG	18.968,49		
8	21/02/2018	103,17	101,76	-2	-200	SHORT	20.350,76	1.382,26	
9	01/10/2018	93,62	92,19	-2	200	LONG	18.439,04		
10	04/10/2018	91,81	91,47	-2	-200	SHORT	18.292,71	146,33	
11	03/05/2019	74,90	73,71	-2	200	LONG	14.744,86		
12	07/05/2019	72,49	70,33	-2	200	LONG	14.067,61	2.759,32	
13	10/05/2019	70,47	65,14	-4	-400	SHORT	26.053,14		
14	11/10/2019	70,57	63,83	-2	200	LONG	12.768,00		
15	15/10/2019	71,58	64,84	-2	-200	SHORT	12.966,00	198,00	
OPERAZIONI								318,32	318,32

HOG	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	09/08/2011	41,47	35,66	-2	200	LONG	7.134,00		UGUALE
2	22/08/2011	40,09	32,99	-2	-200	SHORT	6.596,00	538,00	
3	18/05/2012	47,86	46,13	-2	200	LONG	9.228,00		
4	29/05/2012	48,54	47,98	-2	-200	SHORT	9.594,00	366,00	
5	27/01/2014	57,79	64,27	-2	200	LONG	12.856,00		
6	31/01/2014	57,32	62,34	-2	-200	SHORT	12.466,00	390,00	
7	17/12/2014	57,55	62,96	-2	200	LONG	12.594,00		
8	07/01/2015	57,78	64,30	-2	-200	SHORT	12.858,00	264,00	
9	26/08/2015	54,23	54,68	-2	200	LONG	10.938,00		
10	09/09/2015	54,29	56,27	-2	-200	SHORT	11.252,00	314,00	
11	11/10/2018	44,16	41,07	-2	200	LONG	8.216,00		
12	19/10/2018	42,69	39,39	-2	-200	SHORT	7.876,00	340,00	
13	14/05/2019	40,38	34,20	-2	200	LONG	6.842,00		
14	16/05/2019	40,64	34,95	-2	-200	SHORT	6.988,00	146,00	
OPERAZIONI								178,00	178,00

ING	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	16/12/2014	13,82	12,66	-2	200	LONG	2.534,00		UGUALE
2	19/12/2014	14,19	13,23	-2	-200	SHORT	2.644,00	110,00	
3	13/08/2018	14,26	13,38	-2	200	LONG	2.678,00		
4	16/08/2018	14,50	13,51	-2	-200	SHORT	2.700,00	22,00	
5	10/05/2019	12,13	11,81	-2	200	LONG	2.364,00		
6	14/05/2019	11,94	11,48	-2	200	LONG	2.298,00	118,00	
7	16/05/2019	11,85	11,37	-4	-400	SHORT	4.544,00		
8	02/08/2019	10,84	10,34	-2	200	LONG	2.070,00	50,00	
9	08/08/2019	10,60	10,11	-2	-200	SHORT	2.020,00		
								36,00	36,00

OPERAZIONI 4

K	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	04/11/2011	55,37	49,96	-2	200	LONG	9.994,00		MIGLIORE
2	10/11/2011	55,27	49,40	-2	-200	SHORT	9.878,00	116,00	
3	24/04/2012	55,48	50,68	-2	200	LONG	10.138,00		
4	02/05/2012	55,37	50,04	-2	-200	SHORT	10.006,00	132,00	
5	19/08/2013	63,53	62,92	-2	200	LONG	12.586,00		
6	26/08/2013	63,03	62,77	-2	-200	SHORT	12.552,00	34,00	
7	26/06/2014	65,41	64,99	-2	200	LONG	13.000,00		
8	02/07/2014	65,83	65,80	-2	-200	SHORT	13.158,00	158,00	
9	12/09/2016	70,35	77,28	-2	200	LONG	15.458,00		
10	13/09/2016	70,72	78,60	-2	-200	SHORT	15.718,00	260,00	
11	01/09/2017	65,95	65,47	-2	200	LONG	13.096,00		
12	18/09/2017	66,43	68,31	-2	-200	SHORT	13.660,00	564,00	
13	26/03/2018	64,57	63,75	-2	200	LONG	12.752,00		
14	27/03/2018	65,01	64,40	-2	-200	SHORT	12.878,00	126,00	
15	30/05/2019	56,11	52,91	-2	200	LONG	10.584,00		
16	31/05/2019	55,80	51,45	-2	-200	SHORT	10.288,00	296,00	
								530,00	1.030,00

OPERAZIONI 8

LNVGY	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	10/06/2015	22,48	29,13	-2	200	LONG	5.828,00		UGUALE
2	12/06/2015	22,29	29,30	-2	-200	SHORT	5.858,00	30,00	
3	07/01/2016	14,83	17,72	-2	200	LONG	3.546,00		
4	08/01/2016	15,27	18,57	-2	-200	SHORT	3.712,00	166,00	
5	07/07/2017	11,55	12,20	-2	200	LONG	2.442,00		
6	10/07/2017	11,53	12,16	-2	-200	SHORT	2.430,00	12,00	
7	07/02/2018	10,73	10,57	-2	200	LONG	2.116,00		
8	13/02/2018	10,36	9,77	-2	-200	SHORT	1.952,00	164,00	
								20,00	20,00

OPERAZIONI 4

PG	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	01/12/2010	66,90	61,57	-2	200	LONG	12.316,00		UGUALE
2	15/12/2010	68,91	63,61	-2	-200	SHORT	12.720,00	404,00	
3	29/08/2013	81,50	76,76	-2	200	LONG	15.354,00		
4	06/09/2013	81,62	77,12	-2	-200	SHORT	15.422,00	68,00	
5	01/08/2014	83,15	79,25	-2	200	LONG	15.852,00		
6	08/08/2014	84,28	80,05	-2	-200	SHORT	16.008,00	156,00	
7	03/08/2015	80,45	76,51	-2	200	LONG	15.304,00		
8	11/08/2015	80,42	76,04	-2	-200	SHORT	15.206,00	98,00	
9	12/10/2018	83,44	79,37	-2	200	LONG	15.876,00		
10	17/10/2018	85,72	80,76	-2	-200	SHORT	16.150,00	274,00	
								804,00	804,00

OPERAZIONI 5

SPG	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	08/08/2011	129,98	99,51	-2	200	LONG	19.904,16		UGUALE
2	18/08/2011	133,62	107,42	-2	-200	SHORT	21.482,48	1.578,31	
3	06/06/2013	167,33	153,99	-2	200	LONG	30.799,74		
4	13/06/2013	166,19	150,33	-2	-200	SHORT	30.063,85	735,89	
5	01/05/2015	179,95	181,76	-2	200	LONG	36.354,00		
6	04/05/2015	180,72	184,91	-2	-200	SHORT	36.980,00	626,00	
7	05/10/2018	175,88	170,49	-2	200	LONG	34.100,00		
8	10/10/2018	176,76	174,00	-2	-200	SHORT	34.798,00	698,00	
9	21/12/2018	173,89	168,05	-2	200	LONG	33.612,00		
10	27/12/2018	173,38	165,52	-2	-200	SHORT	33.102,00	510,00	
								1.656,42	1.656,42

OPERAZIONI 5

VOD	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	07/05/2010	23,25	20,43	-2	200	LONG	4.087,63		UGUALE
2	10/05/2010	23,32	21,35	-2	-200	SHORT	4.267,11	179,49	
3	01/12/2010	26,94	26,28	-2	200	LONG	5.257,86		
4	10/12/2010	27,26	26,55	-2	-200	SHORT	5.308,91	51,05	
5	18/05/2012	27,47	26,70	-2	200	LONG	5.341,45		
6	29/05/2012	27,75	27,48	-2	-200	SHORT	5.494,43	152,98	
7	21/02/2013	25,95	25,01	-2	200	LONG	5.003,02		
8	05/03/2013	27,25	25,88	-2	-200	SHORT	5.174,35	171,33	
9	04/02/2014	33,23	36,46	-2	200	LONG	7.294,56		
10	10/02/2014	33,40	37,21	-2	-200	SHORT	7.439,39	144,83	
11	24/08/2015	32,04	33,49	-2	200	LONG	6.700,00		
12	28/08/2015	32,26	34,47	-2	-200	SHORT	6.892,00	192,00	
13	14/06/2016	29,79	30,36	-2	200	LONG	6.074,00		
14	22/06/2016	30,20	32,06	-2	-200	SHORT	6.410,00	336,00	
15	06/02/2018	28,59	28,81	-2	200	LONG	5.764,00		
16	09/02/2018	28,33	28,47	-2	-200	SHORT	5.692,00	72,00	
17	16/05/2018	27,21	26,91	-2	200	LONG	5.384,00		
18	22/05/2018	27,18	26,98	-2	-200	SHORT	5.394,00	10,00	
								1.165,67	1165,67

OPERAZIONI 9

WDC	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	24/02/2011	34,78	30,82	-2	200	LONG	6.166,00		MIGLIORE
2	02/03/2011	34,82	29,44	-2	200	LONG	5.890,00	40,00	
3	04/03/2011	35,62	30,25	-4	-400	SHORT	12.096,00		
4	28/01/2015	80,96	104,23	-2	200	LONG	20.848,00		
5	03/02/2015	80,69	98,46	-2	-200	SHORT	19.690,00	1.158,00	
6	07/08/2017	72,95	81,16	-2	200	LONG	16.234,00		
7	10/08/2017	72,69	80,51	-2	-200	SHORT	16.100,00	134,00	
8	01/12/2017	71,83	79,27	-2	200	LONG	15.856,00		
9	05/12/2017	71,22	77,26	-2	200	LONG	15.454,00		
10	08/12/2017	72,80	80,12	-4	-400	SHORT	32.044,00	734,00	
11	14/05/2019	42,19	43,34	-2	200	LONG	8.670,00		
12	17/05/2019	41,81	44,35	-2	-200	SHORT	8.868,00	198,00	
								320,00	602,00

OPERAZIONI 5

XOM	Data	Prezzo previsione	Prezzo reale	Costo	Quantità	Ordine	Transazioni	Operazione	Risultato originale
1	07/05/2010	71,69	64,50	-2	200	LONG	12.902,00		UGUALE
2	17/05/2010	71,13	63,74	-2	-200	SHORT	12.746,00	156,00	
3	05/08/2011	76,47	75,01	-2	200	LONG	15.004,00		
4	10/08/2011	75,22	70,74	-2	-200	SHORT	14.146,00	858,00	
5	14/08/2013	87,27	89,34	-2	200	LONG	17.870,00		
6	21/08/2013	86,21	86,95	-2	-200	SHORT	17.388,00	482,00	
7	03/02/2014	88,50	91,92	-2	200	LONG	18.386,00		
8	06/02/2014	87,83	89,51	-2	-200	SHORT	17.900,00	486,00	
9	02/08/2016	86,14	86,45	-2	200	LONG	17.292,00	44,00	
10	11/08/2016	86,33	86,69	-2	-200	SHORT	17.336,00		
11	20/11/2017	81,20	80,23	-2	200	LONG	16.048,00	632,00	
12	05/12/2017	82,75	83,41	-2	-200	SHORT	16.680,00		
13	06/02/2018	79,63	78,51	-2	200	LONG	15.704,00	382,00	
14	15/02/2018	78,32	76,62	-2	-200	SHORT	15.322,00		
15	25/10/2018	79,56	78,49	-2	200	LONG	15.700,00		
16	29/10/2018	79,78	78,07	-2	200	LONG	15.616,00	416,00	
17	31/10/2018	81,01	79,34	-4	-400	SHORT	31.732,00		
18	02/08/2019	75,47	72,61	-2	200	LONG	14.524,00	554,00	
19	07/08/2019	74,69	69,86	-2	-200	SHORT	13.970,00		
							-	1.826,00	1.826,00

OPERAZIONI	9
------------	---

	TS originale	TS previsione
N° operazioni	96	90
N° transazioni	198	186
Costi operativi	-408	-384
<b>PROFITTO</b>	<b>3.139,77</b>	<b>1.885,77</b>

N° di operazioni che non hanno rispettato lo stesso andamento di prezzo	9
---	---

Simbolo	Data	Prezzo
AEP	27/02/2020	90,03
AEP	12/03/2020	86,43
AEP	19/11/2020	81,79
AEP	03/12/2020	84,59
ING	26/01/2020	11,00
ING	02/02/2020	10,91
ING	22/09/2020	6,96
ING	30/09/2020	7,09
K	06/02/2020	63,26
K	19/02/2020	65,78
WDC	27/02/2020	52,25
WDC	04/03/2020	56,88