UNIVERSITA' DEGLI STUDI DI MODENA E REGGIO EMILIA

Dipartimento di Economia Marco Biagi

Corso di Laurea Magistrale in

Analisi, consulenza e gestione finanziaria

Deep Learning per le scelte di portafoglio

Relatore:

Prof. Francesco Pattarin

Tesi di Laurea di:

Pietro Silvestri

Anno Accademico 2016-2017

Indice

CAPITOLO 1
Introduzione alla tesi

1.1 Motivazione della tesi

I mercati finanziari sono sistemi complessi, non lineari, dinamici e adattivi che necessitano di strumenti e tecniche avanzate per cogliere le relazioni che si generano (Brock e De Lima, 1995). Per molti anni la finanza, per estrarre evidenze dai dati, ha fatto affidamento a tecniche statistiche standard e a modelli costruiti su assunzioni lontane dalla realtà.
Il Machine Learning, sottocategoria dell'Intelligenza Artificiale, promette di rivoluzionare la conoscenza dei mercati, consentendo ai ricercatori di usare tecniche moderne, del tutto analoghe a quelle impiegate nel mondo dell'hard science, che identificano relazioni non lineari e ad alta dimensionalità.
Da sempre gli investitori, quando sono chiamati a scegliere l'allocazione ottimale di risorse tra le asset classes, analizzano le news dei giornali, i reports degli analisti o gli indicatori economici, facendo prevalentemente affidamento sulle proprie intuizioni e giudizi.
La finanza comportamentale ha recentemente contribuito a mettere in evidenza come queste valutazioni, distorte e guidate dalle emozioni, abbiano generato, nei modelli di asset allocation, portafogli poco realistici, con la conseguenza che da Markowitz (1952) a Black-Litterman (1992) i modelli classici hanno massimizzato gli errori di stima.
Negli ultimi decenni, lo sviluppo tecnologico raggiunto dai computer insieme all'aumentata dimensione dei datatset e alle nuove conoscenze nell'analisi dei dati, hanno permesso agli investitori di integrare le strategie d'investimento con strumenti avanzati, consentendo di prendere decisioni più razionali e indirizzate dai dati.
Il Deep Learning, quando applicato a problemi di previsione di serie temporali, raggiunge risultati sorprendenti, tant'è che oggi è il miglior modo disponibile per approssimare una funzione, qualunque sia la sua complessità (Hill, 1996). Il suo forte potere predittivo *out of sample* permette di stimare l'andamento futuro di un asset molto meglio di un essere umano o di una regressione.
Si prospetta che nei prossimi dieci anni l'industria dell'Asset Management verrà "ridisegnata" (PwC, 2017), l'AI è una opportunità per risanare un business in declino, ottenere efficienza operativa e una riduzione dei costi.
L'idea centrale della tesi è di investigare l'uso e i possibili sviluppi dell'Intelligenza Artificiale nell'Asset Management e di implementare un sistema di asset allocation che, utilizzando i segnali generati da una Rete Neurale Artificiale, produca risultati migliori di quelli raggiunti dai benchamark tradizionali.
Il modello di asset allocation preso a riferimento nella fase sperimentale della tesi è quello di Black-Litterman, colonna portante della teoria di portafoglio. Il modello, utilizzando le previsioni generate da una rete neurale, anziché da un essere umano o da un semplice stimatore, è in grado di generare risultati più stabili e migliori. Inoltre, nel modello sviluppato in questo lavoro, invece di sfruttare la teoria del Capital Asset Pricing Model per calcolare il portafoglio di partenza d'equilibrio, si prende in considerazione un portafoglio di (Hierarchical) Risk Parity generato con un algoritmo di Machine Learning.

Lo sviluppo della tesi è stato possibile grazie a uno stage semestrale presso Axyon AI, un'azienda che offre soluzioni per intermediari finanziari attraverso il Deep Learning. Il lavoro è il risultato dello studio di due discipline, quella statistica-finanziaria e quella ingegneristica-informatica e dell'interazione tra due "profili professionali" con background molto diversi, ma che comunicano nello stesso linguaggio e hanno obiettivi comuni.

1.2 Struttura della tesi

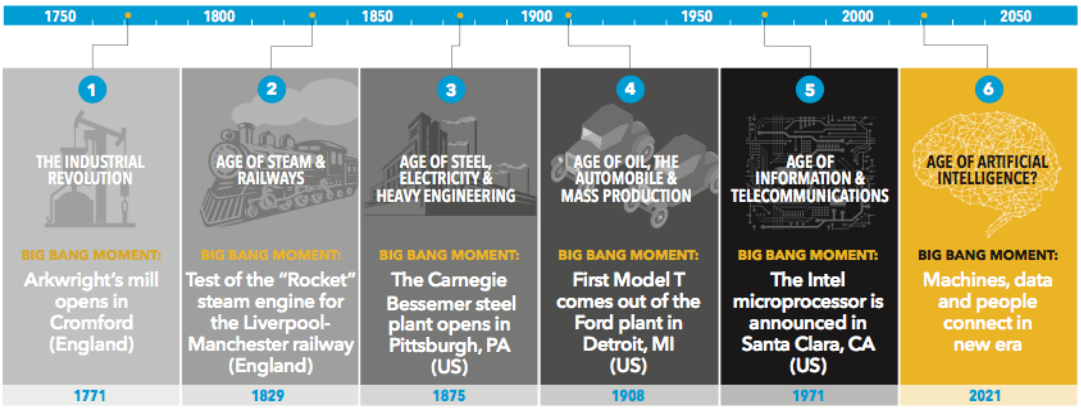
La tesi è strutturata nel seguente modo:

- Il capitolo 2 tratta dell'Intelligenza Artificiale e dei Big Data, dalla storia agli sviluppi futuri. L'obiettivo è introdurre il lettore all'argomento, in termini generali, per capire il contesto in cui si sviluppa la tesi.
- Il capitolo 3 descrive l'uso dell'Intelligenza Artificiale e dei Big Data nell'Asset Management. Vengono riportate le statistiche del settore e le proiezioni future.
- Il capitolo 4 riporta i concetti di base del Machine Learning per capire quali sono le criticità nella strutturazione di un sistema.
- Il capitolo 5 focalizza l'attenzione su una famiglia di algoritmi di Machine Learning, il Deep Learning (reti neurali profonde).
- Il capitolo 6 analizza i modelli più importanti della Modern Portfolio Theory, per capire come si è sviluppata la teoria finanziaria nel tempo.
- Nel capitolo 7 viene descritto l'esperimento e ne vengono riportati i risultati.

CAPITOLO 2
Un'introduzione all'Intelligenza Artificiale e ai Big Data

Nel 1971 la Intel Corporation immette sul mercato il primo microprocessore, il precursore dei chip per computer. Bob Noyce e Gordon Moore, fondatori di Intel, avevano scoperto come miniaturizzare i circuiti elettrici stampati su silicio, un congegno che fino a quel momento occupava una stanza ed era proprietà esclusiva dell'industria militare. Da lì a poco sarebbe nata l'era dell'informatica.
Ogni rivoluzione tecnologica è contraddistinta da una innovazione fondamentale che apre un universo di opportunità (Perez, 2002): nel 1771 Richard Arkwright crea il primo filatoio automatico, che dà inizio alla prima rivoluzione industriale; nel 1829 la prima locomotiva Rocket segna l'inizio dell'epoca delle ferrovie e delle macchine a vapore; nel 1875 la prima fabbrica d'acciaio a Pittsburgh inaugura l'era dell'acciaio e dell'elettricità; nel 1908 la Ford lancia la prima automobile prodotta da una catena di montaggio, dando il via all'era della produzione di massa (figura 2.1). Da oltre 250 anni le rivoluzioni tecnologiche sono i drivers fondamentali del progresso. Ogni rivoluzione ha portato profondi cambiamenti nella società e nell'economia, consentendoci di raggiungere nuovi livelli di benessere.

Figura 2.1. Le sei rivoluzioni tecnologiche



Fonte: Capital Group (2017), p. 1.

Secondo Perter Thiel (2016), fondatore di Pay Pall, il progresso può assumere due forme: orizzontale o verticale. Per progresso orizzontale Theil intende la riproduzione di "cose" che funzionano; a livello macro possiamo estendere il concetto in "globalizzazione". La Cina è l'esempio più calzante: da oltre vent'anni sta seguendo il percorso degli Stati Uniti, imitando tutto ciò che ha funzionato. Poi esiste il progresso verticale, che Theil identifica con la tecnologia. In un mondo di risorse limitate la globalizzazione senza la tecnologia sarebbe insostenibile. Per 10.000 anni i nostri antenati hanno vissuto in società statiche, impadronendosi dei beni degli altri e non creando mai nuove fonti di ricchezza. Dall'avvento del motore a vapore fino alla fine degli anni '70 abbiamo

sperimentato un progresso inarrestabile, ereditando una società infinitamente più ricca di quelle precedenti. Dagli anni '70 poco è stato fatto, abbiamo assistito a una rapida globalizzazione unita a uno sviluppo tecnologico limitato. Ci si aspettava che il progresso continuasse inarrestabilmente, in modo automatico, invece soltanto l'informatica, la comunicazione e la finanza sono progredite. Perfino il nostro linguaggio lascia intendere che ci siamo accontentati: la divisione tra paesi "sviluppati" e non, non è altro che un modo per giustificare la staticità del progresso, sottintendendo che il mondo è già sviluppato ed ha raggiunto il raggiungibile.
Oggi la sfida è quella di riaccendere il motore del progresso e grazie all'Intelligenza Artificiale (AI) questo è possibile. Secondo Dave Coplin, Microsoft's chief envisioning officer, siamo di fronte alla più importante innovazione della nostra era. L'AI fa già parte della nostra vita e sta rimodellando la società: dai Robot-advisor alle macchine che si guidano da sole, l'era dell'Intelligenza Artificiale è arrivata.

2.1 Che cos'è l'AI?

Nonostante l'ampia convergenza di opinioni sul ruolo decisivo dell'AI nel determinare i sentieri di sviluppo futuri, non è ancora possibile trovare una definizione condivisa di cosa sia AI. Il dibattito ha coinvolto negli ultimi sessant'anni più discipline che hanno via via contribuito a metterne in luce diverse sfaccettature.
La domanda "possono le macchine pensare?" ha interessato da sempre gli studiosi, dai filosofi agli scienziati. La riflessione sull'AI è stata una delle aree accademiche più esplorate e sono state proposte diverse interpretazioni di "macchine intelligenti", nessuna delle quali è tuttavia riuscita a racchiudere il pensiero della comunità scientifica in un'unica definizione. Se, infatti, da un lato nessuno può confutare l'abilità logica di un computer, dall'altro questo non ci permette di definirla di per sè come una macchina intelligente.
Nel 1950 Alan Turing, nel suo famoso paper "Computing Machinery and Intelligence", propone un test per identificare una macchina "intelligente". Il test di Turing consiste nel capire se l'interlocutore con cui si sta dialogando a distanza è un computer o una persona.
Nel 1978 Bellman definisce AI come l'automazione di attività che si associano generalmente al pensiero umano mentre nel 1991 Rich e Knight indicano AI come lo studio riguardante l'insegnamento ai computer di attività in cui le persone umane sono migliori.
A seconda che si riponga l'interesse sul pensiero, inteso come processo di creazione e ragionamento, o sul comportamento, inteso come azione e performance, possiamo ottenere diverse definizioni.
Nel 1984 lo scienziato Edsger Dijkstra afferma: "*The question of whether a computer can think is no more interesting than the question of whether a submarine can swim*" sottolineando l'importanza della interpretazione e definizione delle parole.
L'intelligenza, di per sé, è un concetto difficile da definire e che muta nel tempo: nel 1932 il New English Dictionary ha definito l'intelligenza così "*The exercise of understanding; intellectual power; acquired knowledge; quickness of intellect*", mentre nel 1995 "*Intelligence is the ability to reason and to profit by experience. An individual's*

level of intelligence is determined by a complex interaction between their heredity and environment". Oggi è un concetto legato anche alle emozioni (intelligenza emotiva).
E' possibile ottenere altre definizioni di AI in base alle ricerche applicate poste in essere nella comunità scientifica che, a seconda dell'obiettivo, si è concentrata su aspetti diversi: sul ragionamento logico, sulla rappresentazione della conoscenza, sulla pianificazione, sull'elaborazione del linguaggio naturale, sull'apprendimento, sul movimento e la manipolazione. Ancora, le definizioni variano in base al metodo applicato nell'impostazione del problema (Domingos, 2015): il "simbolismo" usa il ragionamento logico basato su simboli astratti; il "connessionismo" costruisce strutture ispirate dalla struttura del cervello umano; gli "evoluzionisti" utilizzano metodi ispirati alla teoria dell'evoluzione di Darwin; i "bayesiani" utilizzano l'inferenza probabilistica e gli "analogizer" estrapolano dai comportamenti passati.
Recentemente è stata data una definizione anche a livello istituzionale. Il Parlamento europeo, nella Proposta di Risoluzione sulla robotica del 2015, ha definito alcune caratteristiche importanti dell'AI quali: "l'ottenimento di autonomia grazie a sensori e/o mediante lo scambio di dati con il suo ambiente (interconnettività) e lo scambio e l'analisi di tali dati; l'autoapprendimento dall'esperienza e attraverso l'interazione; almeno un supporto fisico minore; l'adattamento del proprio comportamento e delle proprie azioni all'ambiente; l'assenza di vita in termini biologici" (Angelini, 2017).
Parte della difficoltà nel trovare una definizione unica deriva anche dal fatto che ogni tentativo descrittivo del fenomeno appare destinato alla transitorietà: le nuove conoscenze continuano a spingere la frontiera dell'innovazione sempre in avanti, è quindi per sua stessa natura che una codificazione troppo specifica rischia di diventare subito obsoleta e di essere sostituita.
Di recente la comunità scientifica ha dato largo credito all'interpretazione di Stuart Russel e Peter Norvig in "*Artificial Intelligence. A Modern Approach*" (2010). Gli autori raccolgono le otto definizioni più importanti, emerse nel dibattito degli ultimi cinquanta anni, riguardanti la sfera del comportamento, del ragionamento e del processo del pensiero, e le riconducono a quattro approcci (Figura 2.2).

1.Pensare come un umano (The cognitive modeling approach): per capire se un computer pensa come un umano, secondo questo approccio, bisogna prima determinare come il cervello umano elabora l'informazione, creando una teoria della mente. Questo può essere fatto attraverso esperimenti psicologici o l'osservazione del funzionamento del cervello umano o l'analisi del pensiero. Una volta che è disponibile una teoria sufficientemente articolata della mente è possibile simularne il funzionamento a computer. Questo approccio è sviluppato dalla scienza cognitiva, il settore che unisce psicologia e computer science per creare modelli e teorie su come il cervello umano elabora l'informazione.
2. Agire come un umano (The Turning Test approach): secondo il test di Turning un computer agisce come un essere umano se, dopo aver sottoposto alcune domande, non è possibile distinguere se le risposte provengono da un uomo o da una macchina. Secondo questo approccio un computer, per poter agire come un umano, deve possedere le seguenti capacità: apprendimento automatico, ragionamento automatico, elaborazione del linguaggio naturale, rappresentazione della conoscenza, visione artificiale, robotica.

3. Pensare razionalmente (The "law of thought" approach): questo approccio raffigura il pensiero come un processo dove le conclusioni sono tratte in base a una corretta sequenza logica.

Figura 2.2. Le quattro categorie di Russel e Norvig

Thinking Humanly	**Thinking Rationally**
"The exciting new effort to make computers think . . . *machines with minds*, in the full and literal sense." (Haugeland, 1985) "[The automation of] activities that we associate with human thinking, activities such as decision-making, problem solving, learning . . ." (Bellman, 1978)	"The study of mental faculties through the use of computational models." (Charniak and McDermott, 1985) "The study of the computations that make it possible to perceive, reason, and act." (Winston, 1992)
Acting Humanly	**Acting Rationally**
"The art of creating machines that perform functions that require intelligence when performed by people." (Kurzweil, 1990) "The study of how to make computers do things at which, at the moment, people are better." (Rich and Knight, 1991)	"Computational Intelligence is the study of the design of intelligent agents." (Poole *et al.*, 1998) "AI . . . is concerned with intelligent behavior in artifacts." (Nilsson, 1998)

Fonte: Russell e Norvig (2010), p. 2.

4. Agire razionalmente (The rational agent approach): per agire razionalmente si intende l'intraprendere azioni che consentano di raggiungere il migliore risultato (o miglior risultato atteso). Un agente è intelligente se agisce razionalmente rispetto all'obiettivo da raggiungere. Non essendo possibile prendere sempre le migliori decisioni possibili, i nuovi modelli di AI cercano di ottenere il meglio dalle informazioni e dalle risorse che hanno (si è passati dalla "optimizing" al moderno approccio "satisfying").

L'ultimo è l'approccio preferito dagli autori; esso permette di ottenere risultati più generali del terzo approccio (il processo logico è uno dei meccanismi possibili per raggiungere la razionalità) ed è più facilmente integrabile con le nuove scoperte perché non si concentra solamente sul pensiero (primo approccio) o sul comportamento (secondo approccio).
In generale, l'Intelligenza Artificiale può dunque essere considerata una disciplina che raccoglie le conoscenze di ingegneria e scienza, con l'obiettivo di creare entità intelligenti, cioè capaci di imparare, risolvere problemi e di agire razionalmente.

2.2 Storia e sviluppi

AI è un settore ancora fortemente in via di sviluppo nonostante le prime ricerche risalgano agli anni '40-'50: nel 1943, viene pubblicata la prima ricerca sull'Intelligenza Artificiale (Warren McCulloch e Walter Pitts); nel 1950 viene creata da M.Minsky e D. Edmonds la prima rete neurale, SNARC, e nel 1956 McCarthy conia l'espressione "Intelligenza Artificiale" che dà vita alla disciplina. Negli anni '60-'70, dopo l'entusiasmo dei primi anni e le prime applicazioni, si ha una battuta d'arresto. Il fallimento di alcuni progetti, uno su tutti quello della comprensione del linguaggio, il taglio dei fondi del governo americano ai progetti di AI e le critiche da parte di alcuni scienziati (H. L. Dreyfus nel libro "What computers can't do" e Minsky e Papert con l'articolo "Perceptrons"), gettano sconforto. A partire dagli anni '80, attraverso l'applicazione industriale, l'Intelligenza Artificiale vive un periodo d'intenso sviluppo. Nel 1982 viene utilizzato il primo sistema in ambito commerciale, R1, impiegato dalla Digital Equipment come sistema esperto per generare configurazioni hardware di computer. In questo periodo, molte aziende americane incominciano a investire per lo sviluppo delle conoscenze di sistemi esperti. Nella seconda metà degli anni ottanta l'industria dell'AI vale miliardi di dollari e comprende centinaia di aziende che sviluppano sistemi esperti, robot, software. In questi anni ritornano d'interesse le reti neurali, prima attraverso la creazione dell'algoritmo d'apprendimento *backpropagation*, poi con lo sviluppo dell'approccio connessionista, consacrato dal libro del 1986 "*Parallel distributed processing*" di McClelland J. e Rumelhart D. Oggi è possibile far andare una macchina senza pilota, creare robot che intrattengono una discussione, sviluppare un algoritmo che batte il campione mondiale di scacchi oppure semplicemente dividere le e-mail dallo spam.

Il recente ritrovamento d'interesse per queste tecniche, trova fondamento nelle tre componenti che hanno generato ciò che è stato definito dagli esperti "*Big Data Revolution*": l'esponenziale incremento di dati disponibili, l'aumento della potenza e della capacità d'archiviazione dei computer a costi ridotti e il perfezionamento delle tecniche d'analisi dei dati, rafforzato dagli sviluppi del ML. La produzione sistematica di dati prodotta nel decennio passato ha portato alla nozione di "Big Data". Per Big data intendiamo l'archiviazione e l'analisi di un grande volume di dati usando varie tecniche tra cui AI. Nel mondo finanziario, i Big Data hanno la possibilità di cambiare profondamente lo scenario attuale: i fund managers, per esempio, in cerca di alpha e strategie incorrelate, cercano sempre più di trarre vantaggi competitivi dall'informazione. Vengono cercati dataset alternativi (alternative data), poco conosciuti, per creare nuove strategie con alpha elevati.

Si stima che il mercato globale della robotica e dei sistemi che sfruttano AI crescerà dai 48,6 miliardi euro del 2014 ai 128 miliardi nel 2020 (Angelini, 2017), dominato dagli investimenti delle grandi società della tecnologia. Compagnie come Google stanno scommettendo milioni dollari sullo sviluppo del settore (Mckinsey Global Institute, 2017). Nel 2016 hanno investito dai 20 ai 30 miliardiimpres di dollari. Anche gli investimenti dagli investitori privati (private equity e venture capital) sono triplicati dal 2013. La maggior parte degli investimenti sono rivolti alla R&D interna (90%) e all'acquisto di startup (10%). Le grandi imprese della tecnologia stanno gareggiando per acquisire i migliori talenti e brevetti in circolazione. Google recentemente ha investito 4.5 milioni di dollari nel "Montreal Institute for learning Algorithms", un laboratorio di ricerca dell' università di Montreal.

L'AI è considerata un fattore di crescita dell'economia perché ha il potenziale per superare le limitazioni fisiche del capitale e del lavoro, creando nuove fonti di valore e ricchezza.
Tradizionalmente i fattori produttivi come il capitale e il lavoro hanno guidato la crescita economica, ma negli ultimi decenni non sono stati capaci di sostenere la prosperità delle economie sviluppate. Secondo un report di Accenture (2016), lo sviluppo dell'AI, nuovo fattore produttivo, ci conduce a ridefinire le relazioni fondamentali dell'economia e come il valore viene creato. Come nuovo fattore produttivo aiuta la crescita attraverso tre determinanti: la creazione di una nuova forza lavoro virtuale (definita "automazione intelligente"), l'aumento del capitale fisico e l'innovazione. L'automazione intelligente permette di replicare molte attività lavorative con maggiore velocità e scala e di ottenere risultati migliori. Ha l'abilità di imparare più velocemente di un essere umano e per alcune attività riesce a essere più preciso. Gli avvocati, per esempio, possono fare analizzare contratti e documenti vari da software, risparmiando denaro e ottenendo risultati migliori. Un recente studio illustra come un algoritmo di ML creato per prevedere le decisioni della Corte Suprema americana possa raggiungere livelli di precisione nella predizione del 70% gestendo oltre 240 variabili.
Il nuovo capitale fisico, macchine intelligenti e robot, permette di raggiungere nuovi livelli d'efficienza: diversamente dal valore del capitale tradizionale che si consuma nel tempo, il valore del capitale investito in macchine intelligenti aumenta; infatti, attraverso il self-learning le prestazioni migliorano all'aumentare dell'esperienza.
Le ore necessarie e i costi per produrre un'unità di output diminuiscono. International Bar Association in "*Artificial Intelligence and Robotics and Their Impact*" (2017) mette in evidenza come un'ora di lavoro di un robot costi 8 volte in meno di quella di un lavoratore umano nel settore automobilistico, senza contare che non necessita di formazione e non è soggetto a infortuni o malattie. Questo rischia di rendere l'uomo non competitivo. Per tali considerazioni, la Proposta di Risoluzione sulla robotica, un insieme di raccomandazioni del Parlamento Europeo concernenti norme di diritto civile sulla robotica, prospetta l'introduzione di meccanismi di tassazione sul lavoro dei robot.
Infine l'AI aiuta la crescita economica perché l'innovazione genera altra innovazione.
Uno studio condotto da Accenture (2016) sull'impatto dell'AI in dodici economie sviluppate riporta come i Paesi che hanno adottato questa tecnologia nei processi abbiano una crescita maggiore. Si stima che AI porterà il maggior beneficio economico in termini assoluti (espresso in GVA, valore aggiunto lordo) negli Stati Uniti, raggiungendo per il 2035 tassi di crescita del 4.6%, e aiuterà, seppure in misura minore, le economie in ritardo come Italia, Spagna e Belgio.
La crescita si distribuisce in maniera diversa anche all'interno dell'economia: Il settore finanziario, insieme a quello tecnologico e delle telecomunicazioni, risultano oggi essere quelli più d'avanguardia, al contrario di altri settori come quello della sanità e dell'educazione. Secondo le stime di McKinsey (2017), il settore della sanità è uno dei più promettenti del prossimo futuro in quanto potrà beneficiare più di altri di un incremento dei profitti derivante dall'adozione dell'Intelligenza Artificiale: si prevede una riduzione del 5-9% delle spese grazie a cure più personalizzate e un aumento della produttivià del 30-50% grazie a processi ottimizzati e automatici. Negli Stati Uniti si stima che si risparmieranno 300 miliardi di dollari all'anno.
Dal report "*The future of jobs*" del 2016 di World Economic Forum, si evince come nei prossimi anni i fattori tecnologici influenzeranno profondamente anche il mondo del

lavoro. Verranno creati due milioni di nuovi posti di lavoro, ma allo stesso tempo, ne verranno distrutti sette per un saldo totale negativo di cinque milioni. E' ragionevole pensare che molti lavori vedranno una riduzione: si ipotizza che ad essere maggiormente colpiti saranno gli impieghi a basso reddito, ovvero lavori ripetitivi e facilmente automatizzabili.

L'impatto sul mercato del lavoro è difficilmente prevedibile e varierà da settore a settore, ma si pensa che a risentirne negativamente in misura maggiore saranno le aree amministrative e di produzione, al contrario di quelle finanziarie, ingegneristiche e informatiche. In molte professioni l'automatizzazione delle attività più ripetitive permetterà di impiegare più tempo in quelle attività che richiedono più creatività. Secondo il World Economic Forum (2016) le prime tre competenze legate allo sviluppo tecnologico che diventeranno indispensabili saranno: il complex problem solving, il critical thinking e la creatività. In questo contesto l'uomo sarà chiamato a fare la differenza attraverso la sua capacità di affrontare problemi complessi in materie sempre più trasversali e aree interconnesse. Per esempio, il data analyst dovrà saper interpretare i numeri e renderli disponibili al management in un linguaggio comprensibile, avendo conoscenze di statistica, programmazione ed economia. Ci sarà domanda di nuove figure dedicate: le aziende avranno bisogno di personale con competenze nuove e dovranno riqualificare quei lavoratori coinvolti in attività investite dall'AI.

Il governo italiano, seguendo l'esempio di Stati Uniti, Inghilterra, Germani e Giappone, ha attivato il progetto Impresa 4.0 volto a incentivare gli investimenti funzionali alla trasformazione tecnologica. Il piano prevede di utilizzare 11 miliardi di euro in ricerca e sviluppo, 10 miliardi in investimenti privati e 2,6 miliardi in investimenti privati early stage (start up).

Probabilmente siamo di fronte a una nuova rivoluzione industriale. Dopo attese eccessive e numerose delusioni, AI sembrerebbe pronta a diffondersi definitivamente e a contribuire allo sviluppo della quarta rivoluzione industriale. La potenza aumentata dei computer, l'enorme quantità di dati disponibili, i sofisticati algoritmi, la necessità dei manager di cercare nuove opportunità di profitto in un contesto di alta competitività e di riduzione dei margini di profitto, permetterebbe la diffusione su larga scala dell'Intelligenza Artificiale. Dopo l'industria 1.0 (1700-1800), l'industria 2.0 (fine 19 secolo), l'industria 3.0 (digitalizzazione fine anni '70), siamo ora davanti all'industria 4.0. Le nuove parole chiave saranno smart production, smart devices, smart energy, ovvero nuove tecnologie produttive e nuove tecniche d'integrazione dei sistemi, tenendo sempre in considerazione i paradigmi dell'energia sostenibile. L'industria 4.0 si muove su principi completamente diversi rispetto al passato. Essa si evolve grazie all' Intelligenza Artificiale, Internet of things, Data network effect o Cloud computing.

Per Internet of things si intende l'estensione di internet al mondo delle "cose". Gli oggetti acquisiscono intelligenza e comunicano dati su se stessi: la sveglia suona prima se rileva traffico per strada oppure il contenitore dei medicinali ti avvisa in caso di dimenticanza. Grazie al collegamento alla rete, gli oggetti acquisiscono un ruolo attivo. Il mondo reale viene mappato da quello virtuale, dando identità a luoghi e cose.

Il Data network effect, invece, nasce dall'idea di aumentare la prestazione di un prodotto all'aumentare del numero di utenti che lo utilizzano: più l'utenza aumenta, più dati sono disponibili, più le prestazioni del prodotto aumentano. Più delle economie di scala conteranno l'ampiezza delle connessioni a cui si può accedere.

Attraverso il Cloud computing le risorse informatiche non vengono configurate dal fornitore appositamente per il cliente, ma vengono assegnate grazie a procedure automatizzate a partire da un insieme di risorse condivise con altri utenti lasciando spazio all'utente nella configurazione.
Nonostante le grosse aspettative l'Intelligenza Artificiale tarda ad essere adottata dalle aziende ed è ancora in una fase iniziale: come riporta il MIT Sload Management Review in "*Reshaping business with artificial intelligence*", 4 su 5 manager riconoscono l'utilità dell'Intelligenza Artificiale, ma solo 1 su 4 ha incorporato nella propria strategia di business questa tecnologia. Quello che emerge è un quadro di luci e ombre: l'85% percento dei manager intervistati (su un campione di 3 mila intervistati in 112 paesi) ritiene AI uno strumento che permette di ottenere un vantaggio competitivo importante, ma soltanto il 5% delle imprese censite la sta sfruttando. Come riporta Jacopo Brunelli, Partner & Managing Director di The Boston Consulting Group per l'Italia, sono ipotizzabili tempistiche simili a quelle della stampa 3D, inventata nel 1983, ma commercializzata su larga scala solo recentemente. Le barriere per adottare l'Intelligenza Artificiale possono essere elevate da superare. La maggior parte dei problemi non deriva da limitazioni tecnologiche, ma piuttosto dalle difficoltà e dai costi di integrazione dell'AI nel business poichè cambiano le attività generatrici di valore, cambiano le procedure e i ruoli, possono sorgere problemi normativi.

2.3 Algoritmi

L'evoluzione del concetto di AI è stato accompagnato dalla sviluppo di algoritmi sempre più potenti. Le prime tecniche di AI osservavano dall'esterno il lavoro del cervello e tentavano di riprodurre la sua performance. Questi sistemi, definiti "sistemi esperti", provavano a imitare il procedimento con cui un esperto affronta un problema. I risultati erano buoni quando si affrontavano problemi ben definiti e circoscritti da un insieme di regole, ma scadenti in situazioni nuove. In questi anni i computer, considerati macchine "intelligenti", sono riusciti a ottenere risultati straordinari in alcune attività come il calcolo, ma non in quelle ritenuti più semplici per il cervello umano, quali il riconoscimento d'immagini. Questo aspetto limitante dei sistemi esperti, molto importante quando si affronta la realtà, è per molte creature viventi una caratteristica vitale dell'intelligenza. La capacità di imparare dall'esperienza, estraendo soluzioni dal passato per utilizzarle nel presente, ci permette di trovare soluzioni migliori e di affrontare al meglio nuove situazioni.
Oggi AI è certamente qualcosa in più dell'automazione di un processo, è un sistema che apprende e migliora sulla base della propria esperienza. Arthur Samuel nel 1959 definisce il Machine Learning (ML) "una sottocategoria del computer science che dà ai computer l'abilità di imparare senza essere esplicitamente programmati". Questo approccio più generale, in contrasto con il vecchio approccio dei sistemi esperti, permette di trovare molteplici soluzioni partendo dall'analisi dei dati. L'utilità del ML è massima quando è difficile esplicitare regole; attraverso queste tecniche è infatti possibile estrarre schemi e relazioni nascoste. Nei sistemi esperti, invece, è necessario definire regole e criteri per risolvere un problema e trasportarli in linguaggio informatico.

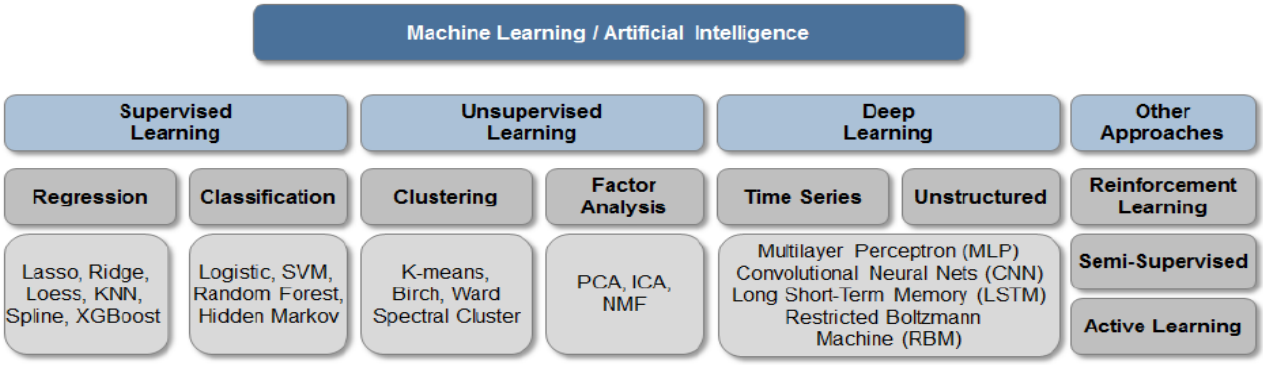
Dagli organismi biologici fino ad arrivare alle macchine, la chiave è sempre l'apprendimento. L'apprendimento è un prerequisito per la sopravvivenza. Tutti gli organismi hanno necessità di percepire l'ambiente in cui vivono, capire i rischi e rispondere di conseguenza. I neuroni, che siano localizzati nel cervello umano o in un lombrico, processano e trasmettono informazione e ci permettono d'imparare. Secondo Volker Tresp, uno dei massimi esperti di ML della società Siemens, esistono tre modi per apprendere: la memorizzazione, cioè la capacità di ricordare le cose; l'abilità, intesa come capacità di imparare; l'astrazione, intesa come capacità di costruire delle regole dalle osservazioni. I computer, fin dalle origini, sono stati sempre molto abili a memorizzare grosse quantità di dati, ma hanno faticato ad apprendere negli altri due modi. Negli ultimi decenni è stato possibile, grazie alle nuove tecnologie e conoscenze, sviluppare nelle macchine nuovi procedimenti d'apprendimento, simili a quelli seguiti dal cervello umano. Il team di Tresp, ad esempio, sta sviluppando una rete neurale artificiale che riesce a fare 10^4 predizioni sulle relazioni tra 10 milioni di oggetti, numeri che corrispondono al numero di sinapsi di un cervello adulto.
Questi nuovi algoritmi permettono di trovare complesse relazioni non lineari, difficili da riconoscere dall'occhio umano, utili per risolvere problemi complessi, come "decifrare" i mercati finanziari. Ad esempio, grazie agli algoritmi di Deep Learning, sottocategoria del ML, oggi è possibile risolvere problemi difficili da descrivere formalmente: le prime applicazioni di AI affrontavano problemi che erano intellettualmente difficili da risolvere per la mente umana, ma relativamente facili per un computer.
Oggi la vera sfida è risolvere compiti intellettualmente facili, ma difficili da descrivere formalmente, problemi che la mente umana risolve intuitivamente. Il Deep Learning analizza i dati attraverso una successione di livelli d'apprendimento, comprendendo il mondo in termini di gerarchia di concetti, in cui ogni concetto è definito attraverso la relazione con un concetto più semplice. Haykin (2009) afferma che il maggior vantaggio di questo modello è la possibilità di trovare la funzione che meglio si adatta ai dati senza esplicitamente definirla. Recentemente, il Deep Learning ha raggiunto risultati che fino a pochi anni fa erano inimmaginabili in diversi settori. Per esempio, nel campo del riconoscimento d'immagini è possibile creare un algoritmo che, attraverso l'analisi dell'immagine satellitare di un parcheggio di un negozio, è capace di stimare la probabilità di un certo ammontare di vendita in un particolare periodo.

2.4 Tecniche di Machine Learning (ML)

Per poter utilizzare al meglio l'enorme e complessa mole di dati prodotta quotidianamente c'è bisogno di sviluppare nuove tecniche di analisi dei dati. Non è possibile analizzare grandi quantità di dati utilizzando strumenti standard come la regressione lineare. Sebbene la regressione lineare possa essere considerata un algoritmo di ML semplice, non è adatta ad affrontare efficacemente outliers, variabili correlate e problemi ad alta dimensionalità. Il ML permette di analizzare efficientemente i Big Data. Il ML è un settore nato dall'intersezione tra la statistica e l'informatica e fa riferimento a un insieme di algoritmi che permettono ai computer di apprendere senza essere esplicitamente programmati. E' possibile dividere gli algoritmi di ML/AI in quattro famiglie (figura 2.3): *Supervised Learning, Unsupervised Learning, Deep Learning, Altri approcci*.

Supervised Learning e *Unsupervised Learning* vengono definiti metodi classici e sono una naturale estensione dei metodi statistici. Il termine Supervised deriva dal fatto che il data scientist guida l'algoritmo nella calibrazione dei parametri.
Nel *Supervised* Learning, infatti, vengono immessi coppie di dati (x,y) nell'algoritmo affinché esso possa imparare la relazione con maggior potere predittivo. La regressione e la classificazione sono due delle tecniche più utilizzate. Attraverso la regressione prediciamo un output a partire dall'analisi di una serie di variabili input, con la classificazione invece identifichiamo a quale classe appartiene una osservazione nuova sulla base dell'analisi di un dataset (training set). Esistono varie estensioni della regressione lineare tra cui la regressione parametrica Lasso e la regressione non parametrica K-nearest neighbors (KNN).

Figura 2.3. Classificazione degli algoritmi di ML



Fonte: Kolanovic e Krishnamachari (2017), p. 18.

Gli algoritmi *Supervised Learning* vengono spesso classificati in parametrici e non: i modelli parametrici sono modulati da un insieme di parametri e di assunzioni restrittive; i modelli non parametrici, al contrario, non essendo vincolati da ipotesi identificano schemi nella storia dei dati e assumono che questi si ripeteranno nel futuro.
La regressione Lasso stabilisce una relazione scegliendo le variabili input più rilevanti: se per esempio due variabili di una regressione sono molto correlate, l'algoritmo le penalizzerà o ne eliminerà una. Questo modello permette di creare risultati più robusti in presenza di un gran numero di variabili potenzialmente correlate e di evitare problemi di overfitting.
Attraverso la regressione K-nearest neighbors, invece, identifichiamo un numero K di esempi simili e facciamo la predizione attraverso una media dei rispettivi K risultati. Dato un training set $\{((x_1, y_2), \ldots . (x_n, y_n)\}$ e un valore K, il modello KNN stima una previsione $\hat{f}(x_0)$ per un punto x_0 individuando l'insieme delle K osservazioni più vicine a x_0 , $\{x_1, \ldots . . x_n\}$ e calcolando la media di tutte le risposte relative $\{y_1, \ldots . . y_n\}$. In sintesi, $\hat{f}(x_0) = \frac{1}{K}\sum y_i$.
Le tecniche non parametriche permettono di estrapolare proprietà dagli eventi passati. Al contrario della regressione, KNN può modellare fenomeni complessi come quelli non lineari e proprio per questo è più sensibile a problemi di overfitting.

I metodi di classificazione del *Supervised Learning* hanno l'obiettivo di classificare in categorie le varie osservazioni ed è possibile, per esempio, ottenere segnali di buy/sell oppure di high/low volatility. La regressione logistica, il Support Vector Machine e Hidden Markov Model sono tra gli algoritmi più utilizzati. La regressione logistica è un estensione della regressione lineare che permette di ottenere un output binario ed è utilizzato per prevedere la probabilità di un evento date le osservazioni passate. Attraverso una funzione logistica trasformiamo una combinazione lineare degli input in un valore tra 0 e 1.Il Support Vector Machine è uno dei classificatori più utilizzati per la sua facilità d'implementazione; permette di modellare fenomeni lineari complessi e attraverso il "Kernel trick" anche non lineari.
Hidden Markov Models è un modello molto utilizzato perché ci permette di descrivere alcuni eventi che non sono direttamente osservabili, ma sono collegati ad altre variabili che possiamo controllare. In questo modello la probabilità di uno stato futuro è collegata con quello dello stato attuale. In finanza è stato spesso usato per inferire il regime in cui si trova il mercato, crescita o flessione, utilizzando variabili che osserviamo come la volatilità e i rendimenti.
Gli algoritmi *Unsupervised learning*, al contrario di quelli *Supervised*, non distinguono tra variabili dipendenti e indipendenti; essi esaminano un dataset e individuano relazioni tra le variabili e i driver comuni. I due metodi più famosi sono il Clustering e l'Analisi fattoriale: attraverso il primo è possibile dividere il dataset in tanti gruppi di elementi omogenei; attraverso il secondo è possibile identificare i principali drivers dei dati o la miglior rappresentazione dei dati.
Uno dei più noti algoritmi di analisi fattoriale è il Principal Component Analysis (PCA): considerato un insieme di dati di mercato, è possibile decomporre ogni serie temporale in una combinazione lineare di fattori incorrelati e capire l'impatto di ogni fattore. La maggior parte delle applicazioni del PCA è nella decomposizione dei rischi di un portafoglio.
Gli algoritmi di *Deep Learning* analizzano i dati attraverso livelli multipli d'apprendimento. Partendo da un concetto semplice, è possibile collegarne altri per ottenere un concetto più difficile. Il Deep Learning è, in essenza, più simile al processo d'apprendimento del cervello umano, e può essere visto come un tentativo di ricreare artificialmente l'architettura di una rete neurale biologica. Per esempio, un bambino impara a riconoscere il concetto di viso riconoscendo alcune sue caratteristiche come il naso, la bocca, gli occhi. Mettendo insieme caratteristiche semplici posso creare un concetto più difficile.
Questi modelli hanno un forte potere predittivo *out of sample* e per questo vengono molto usati in ambito di Asset Management. Le architetture più importanti di Deep Learning sono: *Multilayer Pereptron* (o feedforward), *Long short term memory* e *Convolutional neural network*. *Multilayer Pereptron* è una rete di almeno 3 livelli. Ogni nodo della rete è un neurone che utilizza una funzione d'attivazione non lineare. Il *Long short term memory* è un modello di rete ricorrente (una rete che, al contrario di quelle tradizionali, permette all'informazione di persistere nel tempo attraverso l'uso di loop) capace di imparare relazioni di lungo periodo. Questo tipo d'architettura conserva una memoria degli inputs utilizzati e permette di ricordare eventi lontani. Le *Convolutional neural network* sono uno speciale tipo di Multilayer Pereptron disegnato specificamente per gestire immagini, infatti sono modelli inspirati all'organizzazione della corteccia visiva animale. Diversamente dalla forma tradizionale, i neuroni di questa rete sono

connessi a una piccola regione degli input e condividono i pesi con altri neuroni vicini. Queste reti permettono di alleggerire la fase di pre processamento. Poichè la maggior parte delle applicazioni si rivolge al settore delle immagini, ancora oggi si sa poco dei risultati ottenibili nel mondo del trading.
Per concludere, nella quarta famiglia che raccoglie i rimanenti algoritmi abbiamo quello molto importante del *Reinforcement Learning*. L'obiettivo di questa tecnica è di scegliere, passo dopo passo, il percorso che porterà a massimizzare il compenso. Il modello non conosce la risposta corretta ogni passo (come nel *Supervised Learning*), ma impara nel tempo qual' è il percorso da compiere per massimizzare il compenso. Ha attributi sia del *Supervised Learning* che del *Unsupervised*. Attraverso l'allenamento la macchina migliora la performance finché non arriva alla perfezione. Quando usato insieme al Deep Learning può dare risultati molto importanti come far andare una macchina senza pilota. Nella finanza è particolarmente usata nel trading, soprattutto nelle strategie di high-frequency.

2.5 Big Data

Per Big Data, originariamente, si intendeva un grande datasets che non poteva essere archiviato, gestito e analizzato. La definizione è profondamente cambiata nel tempo ed oggi non fa più solo riferimento alla disponibilità di dati, ma anche alla tecnologia utilizzata per estrarre valore (Deutsche Bank, 2014). L'idea di utilizzare dati per guidare le decisioni di business non è nuova, ma più che mai è diventata la base della competizione. Oggi i Big Data sono quello che il petrolio è stato nel secolo scorso e, come per tutte le risorse limitate, la competizione è sfrenata. L'attuale mercato mondiale dei Big Data e della relativa tecnologia è stimato avere un valore di 130 miliardi di dollari e ci si aspetta cresca a 200 per il 2020 (International Data Corporation, 2016). L'industria finanziaria è una delle determinanti principali di questa crescita e, secondo le stime di JP Morgan (2017), la spesa attuale di 2-3 miliardi di dollari in tecnologia, acquisto di datasets, assunzione di personale qualificato è destinata a crescere a tassi elevati (10-20% annuale).
E' stimato che il 90% dei dati oggi disponibili nel mondo sono stati creati negli ultimi due anni. Ci si aspetta un incremento notevole di dati, dai 4,4 trilioni di gigabytes registrati nel 2015 ai 44 trilioni di gigabytes nel 2020 (Kolanovic e Krishnamachari, 2017).
La necessità delle aziende di avere a disposizioni analisi sempre più approfondite ha alimentato la crescita dei dati. Il cambiamento più profondo è derivato dall'utilizzo sempre maggiore di dati non strutturati come il web, i social media, le e-mail e i sensori. Attualmente, l'80% dei dati mondiali non sono strutturati. Anche la quantità e la velocità dei dati è cambiata: non sono più misurati in terabytes ma in zettabytes e il flusso è continuo. Il volume dei dati archiviati è elevatissimo e in rapido sviluppo, basti pensare che nel 2000 i dati immagazzinati erano 800.000 petabytes e si prospetta un aumento a 35 zettabytes per il 2020. Twitter ogni giorno genera 7 terabytes di data, Facebook 10 (Zikopoulos, 2012).
Per parlare di Big Data il volume deve essere considerato in relazione alla capacità del sistema di acquisire informazioni e affinché un sistema possa diventare big, volume e velocità devono aumentare entrambe. Nel 2001 Doug Laney ha delineato tre caratteristiche chiave dei Big Data: il volume, la velocità e la varietà. Per volume si

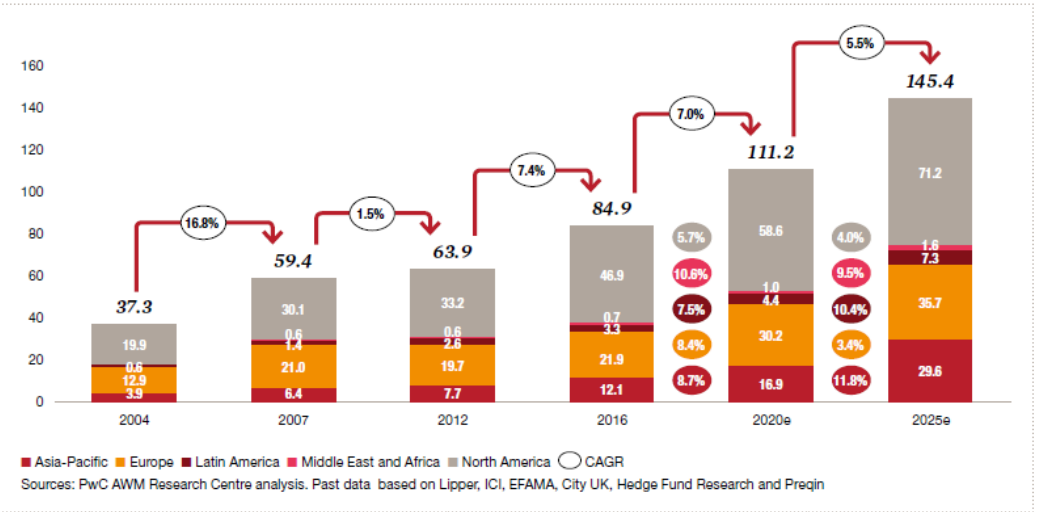
intende la quantità dei dati generati e archiviati. Per varietà, la tipologia di dato (grezzo, semi-strutturato, strutturato). Infine per velocità, la rapidità con cui i dati circolano.
Il più importante cambiamento però deriva da come i dati vengono analizzati. Grazie alle nuove tecniche di analisi, proprie del ML, i dati non sono più una massa indecifrabile, ma una fonte di conoscenza utile a guidare le decisioni.
I Big Data hanno cambiato il modo di fare business e il vantaggio competitivo creato da questo tipo di tecnologia è enorme. Attraverso la loro analisi è possibile aumentare l'efficienza e la velocità dei processi decisionali. Il processo decisionale viene rimodellato e diviso in 5 fasi: generazione e acquisizione dei dati, estrazione e pulizia dei dati, immagazzinamento ed integrazione, modellazione e analisi, interpretazione. Ogni fase modifica lo stato dei dati contribuendo alla creazione di valore. Il vantaggio competitivo viene raggiunto attraverso una combinazione di tecnologia, processi, persone (EY, 2014). E' possibile in questo modo prendere decisioni migliori, basate su evidenze e non su intuizioni, e ridurre il costo dell'informazione.

CAPITOLO 3
Intelligenza Artificiale e Big Data nell'Asset Management

3.1 L'Intelligenza Artificiale nell'Asset Management

Il settore dell'Asset Management è in grande evoluzione, si prospetta (PwC, 2017) che nei prossimi dieci anni l'industria verrà "ridisegnata". I maggiori cambiamenti riguarderanno le commissioni, i prodotti, la regolamentazione, la tecnologia e le abilità dei lavoratori.
Gli Asset Under Management (AUM) continuano a crescere (PwC, 2017); viene stimato che si passerà dagli attuali 84,9 trillioni di dollari ai 145.4 trillioni del 2025, con tassi di crescita maggiori per le asset classes alternative (figura 3.1). Le aree geografiche più in crescita sono quella asiatica (tassi di crescita dell'8,7% all'anno fino al 2020) e sud americana (7,5%). Questo trend positivo viene sostenuto dalla crescita della ricchezza personale, soprattutto dei paesi in via di sviluppo.

Figura 3.1. Stima della crescita degli Asset Under Management (2004-2025) in trillioni di dollari



Fonte: PwC (2017).

I ricercatori di PwC prospettano una diminuzione delle gestione attive dal 71% dei AUM globali attuali al 60% per il 2025. Le gestioni passive, invece, continuano il trend positivo, incrementando la quota di mercato dal 17% attuale al 25% (2025), mentre gli alternatives (real estate, private equity…) sono dati in crescita dal 12% al 15%.
Nonostante le prospettive di crescita incoraggianti, dal 2007 gli asset manager sono sotto pressione a causa della riduzione dei margini di profitto. Le commissioni sono state compresse dall'aumentata regolamentazione, dalla competizione e dai players passivi.

Secondo il Boston Consulting Group (BCG, 2017) dal 2013 i ricavi netti (espressi come quota degli AUM) sono calati da 29.3 punti base a 26.7 a causa soprattutto della riduzione delle commissioni e degli investimenti verso le gestioni passive.
BCG stima che le commissioni nette degli hedge funds sono calate in generale del 1% annuo dal 2010.
Nonostante la crescita degli AUM dei prodotti passivi (mandati, fondi, ETFs), dai 6 miliardi del 2008 ai 14 del 2016, questi contribuiscono ancora poco nel bilancio dell'industria mondiale, generando il 6% dei ricavi totali.
Oggi il maggiore contributo ai ricavi proviene dagli "alternatives" (42% dei ricavi totali nonostante rappresentino soltanto il 15% degli AuM), e a seguire le "actives specialties" (21%).
In un contesto d'incertezza e riduzione dei margini, la necessità primaria è quella di investire in nuovi prodotti innovativi, nuove tecnologie e nuove abilità.
La tecnologia impatterà ogni aspetto dell'Asset Management. L'AI è una opportunità per risanare un business in declino, ottenere efficienza operativa e una riduzione dei costi.
Oggi il ML è utilizzato da un piccolo sottoinsieme di fondi quantitativi.
Il Quantitative investing è una strategia d'investimento a gestione attiva (prevalentemente) che impiega modelli matematici e statistici per trovare opportunità d'investimento (Morgan Stanley, 2017*a*). I fondi quantitativi, utilizzando algoritmi sofisticati e grandi dataset, sfruttano le inefficienze di mercato a brevissime termine per ottenere extra rendimenti.

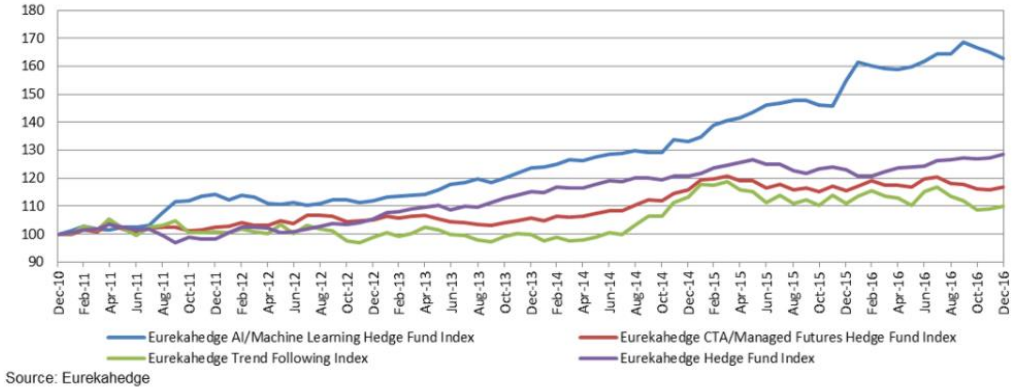
Figura 3.2. Crescita degli asset under management dei fondi quantitativi

Fonte: Wigglesworth (2017a).

Dagli anni novanta, il ricorso a questa strategia è cresciuto molto, vivendo fasi alternate (figura 3.2), grazie a vari fattori (Chincarini , 2010): il miglioramento della tecnologia e la crescita del volume dei dati, il miglioramento degli strumenti quantitativi per l'analisi dei mercati finanziari, l'aumentata domanda da parte dei fondi pensione e degli investitori istituzionali degli stessi, i rendimenti (promessi) superiori rispetto ai metodi tradizionali.

Secondo Hedge Fund Research (Wigglesworth, 2017*a*), quest'anno l'ammontare di denaro gestito dai fondi quantitativi sorpasserà i 1.000 miliardi di dollari, il doppio rispetto il 2010.
Soltanto nel primo trimestre del 2017 sono stati registrati 179 nuovi fondi d'investimento, di cui 55-60 quantitativi.
E' difficile quantificare la diffusione di questa tecnologia per mancanza di informazione pubblica, il Financial Stability Board (2017) stima che i fondi che utilizzano l'AI abbiano in gestione 10 miliardi di dollari, ma che la cifra sia in netta crescita.
Il fondo Eurekahedge è stato tra i primi a rendere pubbliche statistiche sull'AI e big data (figura 3.3): nel 2017 hanno raccolto dati, relativi alle performance di 23 fondi dal 2011 al 2016, mettendo a confronto i rendimenti annui dei fondi che utilizzano ML (Eurekahedge AI) con i fondi quantitativi (Eurekahedge CTA/Managed Futures Hedge Fund Index) e con quelli tradizionali (*Eurekahedge Hedge Fund Index*).
Come si vede dalla figura 3.3, dal 2011 i fondi che hanno utilizzato AI/ML hanno performato meglio (rendimento annuale del 8.44%) rispetto a quelli quantitativi (2.62%) e a quelli tradizionali(4.27%).

Figura 3.3. AI/Machine Learning Hedge Fund vs. Quants vs. Traditional Hedge Funds



Fonte: Eurekahedge (2017).

Riassumendo le statistiche principali (figura 3.4):
-i fondi AI/ML hanno ottenuto sempre rendimenti superiori tranne che nel 2012.
-i fondi AI/ML hanno ottenuto rendimenti aggiustati per il rischio (Sharpe Ratio) superiori nell'orizzonte temporale di due e tre anni.
-i fondi AI/ML sono più volatili rispetto ai fondi tradizionali.

In contrasto con i tradizionali algoritmi quantitativi che devono essere programmati per svolgere un preciso compito, il ML individua patterns senza specificatamente esserlo e migliora la performance all'aumentare dei dati. Queste caratteristiche, oltre a ottenere risultati migliori, ottimizzano la struttura dei costi automatizzando parte delle attività gestite da esseri umani. Optimas (Pierron, 2017*b*) stima per il 2025 un miglioramento del 28% del rapporto costi-ricavi delle aziende che adottano l'AI, ma allo stesso tempo, 230.000 posti di lavoro in meno dal mercato dei capitale, di cui 90.000 nel settore dell'Asset Management, il più colpito. Secondo il BCG (2017) l'automazione dei processi ridurrà del 20-30% i costi generali. Quest'anno, per esempio, a JP Morgan i bots hanno

gestito 1.7 milioni richieste d'accesso, per compiti come il ripristino di una password, facendo il lavoro di 140 persone (Son e Surane, 2017). I lavori più a rischio sono quelli più facilmente automatizzabili, ma anche lavori a stretto contatto con i dati, come quello del junior investment banker. Infatti, come riporta la società di consulenza Kognetics (Son e Surane, 2017), gli analisti finanziari impiegano molto tempo ad analizzare dati che potrebbero essere trattati da macchine.

Figura 3.4. Le statistiche a confronto

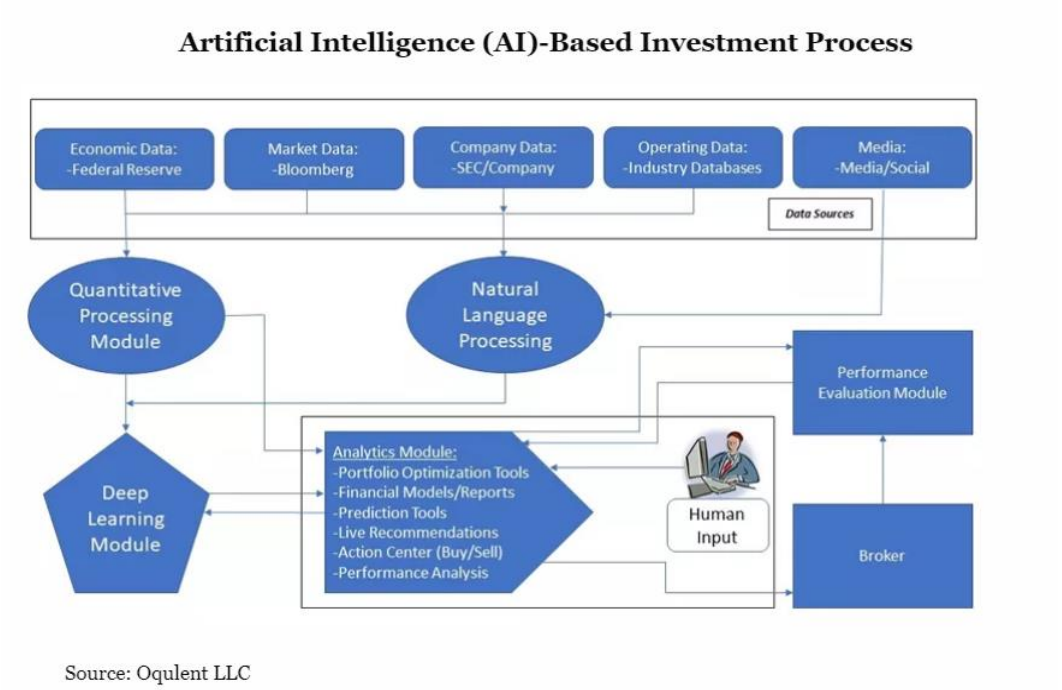
	Eurekahedge AI/Machine Learning Hedge Fund Index	Eurekahedge CTA/Managed Futures Hedge Fund Index	Eurekahedge Trend Following Index	Eurekahedge Hedge Fund Index
2011	14.10%	2.33%	0.71%	(1.75%)
2012	(1.80%)	2.66%	(1.86%)	7.34%
2013	10.34%	0.55%	1.02%	9.24%
2014	7.64%	9.66%	13.44%	4.89%
2015	16.40%	(0.31%)	(2.18%)	1.78%
2016	5.01%	1.15%	(0.62%)	4.48%
5 year annualised returns	7.35%	2.68%	1.80%	5.51%
5 year annualised volatility	4.95%	4.18%	7.13%	3.20%
5 year Sharpe Ratio (RFR=1%)	1.28	0.40	0.11	1.41
3 year annualised returns	9.57%	3.41%	3.31%	3.71%
3 year annualised volatility	5.61%	4.63%	7.78%	3.03%
3 year Sharpe Ratio (RFR=1%)	1.53	0.52	0.30	0.89
2 year annualised returns	10.56%	0.42%	(1.40%)	3.12%
2 year annualised volatility	6.31%	4.90%	8.07%	3.31%
2 year Sharpe Ratio (RFR=1%)	1.51	(0.12)	(0.30)	0.64

Fonte: Eurekahedge (2017).

Con parte del lavoro diretto dalle macchine l'errore umano e le distorsioni cognitive vengono minimizzate; inoltre, il ruolo del quantitative analyst passa a un livello superiore, permettendo di impiegare più tempo in attività che richiedono creatività, come lo sviluppo di nuove strategie d'investimento.
Un altro vantaggio del ML, soprattutto del Deep Learning, è la capacità di individuare relazioni molto complesse in dati ad alta dimensionalità e non strutturati (news, social media, ecc.), consentendo ai manager di trovare alpha nascosti, e nel rappresentare appropriatamente i dati senza processi d'ingegnerizzazione, creando modelli senza assunzioni fondati sull'evidenza estratta dai dati.
Nelle banche d'investimento e nei fondi, l'AI, cambia il processo d'investimento (figura 3.5). Scomponendo il processo in cinque fasi principali -1. raccolta dei dati, 2. processamento, 3. analisi degli investimenti, 4. "decisione", 5. valutazione della performance-, oggi le fasi 1, 2, 5 sono già sostituibili da algoritmi (Saidov, 2018).
Secondo un report di Barclays (Rosov, 2017), che ha intervistato 64 hedge funds e 25 investitori istituzionali, l'88% dei managers utilizza il ML come strumento per processare i dati, mentre soltanto il 25% nella fase "decisione".

In alcuni casi estremi, come Wealthfront, Batterment o Charles Schwab, tutto il processo viene sostituito da una "macchina", dando inizio a un nuovo trend, la ricerca del miglior algoritmo. Questo nuovo servizio, chiamato *robo advisory*, abbatte i costi di commissione attraverso l'automatizzazione del processo d'investimento (Deloitte, 2016). In breve, il robo-advisor transforma gli input del cliente (avversione al rischio, capitale, orizzonte temporale, ecc) in logiche d'investimento e propone diverse soluzioni compatibili con le preferenze dell'investitore.
Secondo le stime di Deloitte (2016), analizzando un campione esteso tra USA, Europa e UK, 80% dei robot-advisors non integrano ancora algoritmi di autoapprendimento e non sono completamente automatizzati (modelli ibridi).
Si prospetta che nel 2020 gestiranno dai 2.2 ai 3.7 trillioni di assets, che per il 2025 saliranno a 16.

Figura 3.5. Un possibile processo d'investimento che utilizza l'AI



Fonte: Saidov (2018).

Grandi società come Bridgewater Associates, Goldman Sachs, Deutsche Bank, UBS, BlackRock, Wells Fargo hanno investito molto (Crosman, 2017) sull'AI.
Le principali applicazioni nel settore sono legate alla predizione di trend o di titoli oppure dei comportamenti degli investitori e, a seguire, alla gestione del rischio, alla costruzione e all'ottimizzazione di portafoglio, al trading algoritmico, all'analisi di dataset alternativi (*sentiment*), all'esecuzione di ordini, al robot advisory.
Bridgewater Associates, famoso fondo d'investimento americano, ha sviluppato un software che fa uso dell'AI per automatizzare la gestione della società. L'obiettivo è quello di far prendere i ¾ delle decisioni dagli algoritmi.

BlackRock, la più grande società d'investimento del mondo, ha sviluppato una piattaforma operativo per asset managers che, sfruttando l'AI, permette di controllare moltissima informazione real-time e prendere decisioni consapevoli.
Aladdin, sviluppato inizialmente per l'uso interno, utilizza algoritmi per il riconoscimento naturale del linguaggio per leggere centinaia di articoli, documenti, report ed estrarre una valutazione sul *sentiment*. BlackRock crede che il ML possa permettere ai manager di prendere decisioni migliori, ma rifiuta l'idea di un processo d'investimento interamente fondato sull'AI. Kochansky, Head of the Aladdin Product Group, dice:" As a fiduciary, you have to always understand why you are making a particular investment decision. We have to be careful about letting loose artificial intelligence".
Goldman Sachs invece usa Kensho, una piattaforma che utilizza il ML per trovare la correlazione tra un evento accaduto nel mondo e l'impatto su un asset. Attraverso questo software è possibile rispondere in brevissimo tempo a domande del tipo "Come reagiscono i titoli difensivi a un attacco terroristico in Europa?" che tradizionalmente impegnavano gruppi di analisti per mesi.
Wells Fargo, UBS, Deutchs Bank invece utilizzano Sqreem per analizzare i comportamenti degli investitori e predire le loro preferenze, dando consigli personalizzati a clienti facoltosi.
In conclusione, sono tempi difficili per l'industria dell'Asset Management, perché il business cambia rapidamente surclassando le società senza una fonte di vantaggio sostenibile. Data la competizione intensa che spinge gli asset managers a cercare efficienza operativa e nuove opportunità d'investimento, non sorprende che Wall Street abbia chiesto aiuto alla Silicon Valley per sviluppare un approccio guidato dai dati. L'AI estrae valore da dati strutturati e non, derivanti da varie fonti, per migliorare le decisioni di business. Gestire il flusso dell'informazione e convertirlo in strategie d'investimento sarà la più grande sfida dell'industria dell'Asset management nei prossimi anni.
Oltre alla tecnologia e ai dati, gli imperativi strategici per il futuro saranno la gestione dei costi e l'offerta di nuovi prodotti/servizi innovativi.
Molti professionisti del settore prevedono un cambiamento radicale, come Luke Ellis, Chief Executive Officer di Man Group, che afferma in un' intervista per Bloomberg (2017): "If computing power and data generation keep growing at the current rate, then machine learning could be involved in 99 percent of investment management in 25 years".
Le grandi società d'investimento hanno iniziato a integrare i propri business models e a investire significativamente su tecnologia, dati e competenze. Optimas (Pierron, 2017*a*) si aspetta che per il 2021 le spese relative all'AI raggiungeranno i 2.8 miliardi di dollari, un aumento del 75% rispetto ad oggi. Le figure lavorative più ricercate vengono dal mondo della scienza- fisica, statistica, matematica, ingegneria-. Molto richiesti sono i data scientists, esperti nella gestione e analisi dei dati, con conoscenze avanzate in informatica e in statistica.
Il futuro dell'Asset Management è sempre più connesso a quello tecnologico e c'è chi, come Hiromichi Mizuno, chief investment officer del più grande fondo pensioni del mondo (Government Pension Investment Fund Japan), si chiede quando Google e Amazon entreranno nel mercato (Takeo e Nozawa, 2017).

3.2 I Big Data nell'Asset Management: i dati alternativi (Alternative data sets)

Al centro della rivoluzione dei Big Data dell'industria finanziaria ci sono nuovi datasets estratti da fonti alternative che forniscono un vantaggio informativo notevole. Nuovi dataset derivanti dalle tecnologie, come i cellulari, i satelliti e i social media, permettono di ottenere informazioni non presenti nelle fonti tradizionali di dati. Avere accesso alle immagini satellitari, per esempio, permette di ottenere informazioni più rapidamente rispetto a un notiziario e di anticipare il mercato.
Gli *alternative data* provengono spesso da fonti esterne al mondo finanziario o dal business dei data provider. Negli ultimi anni il mercato dei dati si è trasformato in un mercato altamente sofisticato, attraendo società provenienti da tutti i settori e creando nuovi ruoli come quello del ricercatore di dati alternativi. Tutto questo interesse ha trasformato i dati in assets di valore. Oggigiorno un investitore professionale, utilizzando l'enorme volume di dati aggiornati in tempo reale e le tecniche sofisticate di analisi dei dati quali il ML, può ottenere un vantaggio competitivo notevole rispetto a un tradizionale investitore. L'abilità delle macchine di analizzare istantaneamente gli earnings delle società quotate, i posts dei social media, i dati delle carte di credito e i search trends sta erodendo il vantaggio accumulato nel tempo dei tradizionali analisti finanziari e dei investitori macro. Per questo motivo è diventato fondamentale acquisire e capire grandi volumi di dati e usare tecnologie sofisticate per l'analisi. Più i datasets alternativi si diffondono, più il mercato reagisce velocemente e anticipa i segnali generati dalle fonti tradizionali, rendendo obsoleti e privi di valori gli attuali datasets. Come riporta JP Morgan nel suo dossier (2017), I dataset tradizionali perdono il loro potenziale predittivo, i vecchi strumenti d'analisi dei dati vengono sostituti e i fondi d'investimento assumono sempre più data scientists. I processi d'investimento dei fondi cambiano: maggior tempo e risorse sono spese nell'acquisizione, processamento e analisi dei dati. In cerca di strategie incorrelate e alpha, i manager dei fondi adottano strategie quantitative basate sui Big Data. Il vantaggio informativo non sarà più generato dagli esperti di settore o dagli insiders, ma piuttosto dall'abilità di analizzare grandi quantità di dati in tempo reale.

3.3 Classificazione dei dati alternativi

E' possibile classificare i dati secondo chi li ha generati (Kolanovic e Krishnamachari, 2017): (i) individui, (ii) processi di business, (iii) sensori.
(i) I dati generati dagli individui sono spesso non strutturati e possono avere come origine i social media, web searches, oppure siti specializzati (es: Twitter, Linkedln, Google search). L'analisi *sentiment* dei social media è molto popolare ed economica rispetto ad altri datasets, per questo motivo sono nate molte società ponte che agiscono da intermediarie tra il mondo dei social network e quello dell'industria finanziaria, come ad esempio Gnip, che permette l'accesso e l'analisi dei dati di Facebook, You tube, Google; oppure iSentium che genera segnali di buy/sell sullo S&P500 analizzando il *sentiment* di molti social media
(ii) I dati generati da processi di business, tipo quelli prodotti da entità pubbliche, governi o aziende (es: WTO, World Bank, Federal Reserve), sono dati altamente strutturati e prodotti con una frequenza minore rispetto a quelli prodotti dagli individui. Price Stats, per esempio, produce delle stime sull'inflazione giornaliere tenendo

controllati i prezzi di 1000 commercianti di 70 nazioni diverse; oppure Dun and Bradstreet, monitorando i pagamenti (ammontare, ritardi, inadempienze..) tra società e valutando il rischio di credito, riesce a creare portafogli long-short
(iii) Infine, I dati generati dai sensori, quali ad esempio, le immagini stellitari, geolocalizzatori, sensori per l'inquinamento e il meteo, sono dati tipicamente non strutturati e di dimensioni molto grandi. Orbital Insights è una stratup che utilizza e analizza i big-data generati dalle immagini satellitari per dare agli investitori una panoramica sempre aggiornata dell'economia mondiale: per esempio, attraverso l'ombra generata dai barili di olio Orbital Insight predice il livello delle scorte. Altre società, tra cui AirSage e Placed, tracciando la localizzazione degli smartphone attraverso gps o wifi monitorano l'affluenza nei grandi magazzini.

3.4 Caratteristiche importanti di un dataset alternativo

E' possibile individuare delle caratteristiche importanti che mostrino l'utilità di un dataset (Kolanovic e Krishnamachar, 2017):
1-Asset class: un dataset alternativo acquisisce valore quando raccoglie informazioni non presenti nei dataset tradizionali. Nella Asset Management, ad esempio, i nuovi dataset non raccolgono informazioni solo sugli strumenti tradizionali quali azioni e bonds, ma anche su asset class alternative (poco correlate con le asset class tradizionali), tipo hedge fund o i fondi di private equity.
2-Stile d'investimento: la maggior parte dei dati disponibili sul mercato sono rilevanti per gli investitori tradizionali quali i macro-investors o stock-specific. Utilizzando I dati alternativi è possibile generare dei segnali nuovi rilevanti soltanto per pochi investitori (es high frequency trader) e sfruttare opportunità nascoste.
3-Contenuto alpha: la caratteristica più importante di un dataset alternativo è la presenza di alpha. E' importante valutare se nei dati è possibile trovare degli alpha che giustifichino i costi di acquisto ed elaborazione del dataset.
4-Diffusione del dataset: capire quanto noto è il dataset all'interno del mercato finanziario è d'importanza fondamentale: più è diffuso e più è probabile che le strategie generate non portino a Sharpe Ratio importanti. I dataset molto conosciuti hanno un basso contenuto alpha e difficilmente potranno portare a strategie vincenti.
5-Livello di elaborazione dei dati: gli investitori preferiscono ricevere segnali piuttosto di dati grezzi perché il processo d'elaborazione dell'informazione ha costi elevati. I dati grezzi non possono essere utilizzati direttamente per creare una strategia, devono essere processati. Ci sono quattro livelli di qualità dei dati: il livello migliore è il report o un'idea di trading strutturata, a seguire abbiamo i segnali, che devono essere processati, poi i formati semi elaborati, dati in formato csv o xml (spesso sono presenti gaps e outliers), infine i dati grezzi.
6-Qualità: questa caratteristica è molto importante perché serie storiche corte, gap e oiutliers rendono difficile l'estrapolazione di informazioni. Più i dati sono puliti, meno tempo, lavoro e denaro è necessario.
7-Aspetti tecnici: frequenza, latenza, formato, rischi legali dei dati ecc.

CAPITOLO 4
Machine Learning Basics

Il ML fa parte del settore più grande del "computer science and statistics". Un algoritmo viene definito di ML se è capace di apprendere un compito dai dati, migliorando la performance all'aumentare dell'esperienza. Mitchell, nel 1997, definisce il concetto di apprendimento: "A computer program is said to learn from experience E with respect to some class of task T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E".

4.1 Features, Target, Training Example, Training Set, Funzione d'Ipotesi

E' importante stabilire un linguaggio di base che possa essere utilizzato per capire le principali criticità del ML. Per questione di semplicità, ho deciso di esporre I successive argomenti facendo riferimento all'algoritmo più semplice di ML, la regressione. E' possibile applicare i principi del ML alla regressione e considerarla un algoritmo di Supervised Learning. Chiamiamo l'input (o features) $x^{(i)}$e l'output (o variabile target da predire) $y^{(i)}$. Un esempio (training example) è una coppia ($x^{(i)}$, $y^{(i)}$) (in un algoritmo di unsupervised learning è un vettore $x \in R^n$). Una lista di m esempi è chiamato training set. Indichiamo X per indicare lo spazio degli input e Y quello degli output. Definiamo funzione di ipotesi la funzione $h: X \to Y$ utilizzata per predire y dato x.

4.2 Il compito (The Task), T

Con il termine "compito" (Task) intendiamo come il sistema di apprendimento elabora un esempio. L'apprendimento non rappresenta il compito, ma il processo per imparare "l'abilità" utilizzata per compiere il task. Il ML può essere impiegato per risolvere differenti tasks, tra cui:

- Classificazione: attraverso la classificazione viene insegnato all'algoritmo ad associare un input a una categoria. Un esempio di classificazione è il riconoscimento di oggetti, dove l'input è un immagine descritta come un insieme di valori distribuiti in base alla brillantezza del pixel, e l'output è un codice numerico identificativo dell'oggetto nell'immagine.
- Regressione: in questo tipo di task, viene chiesto all'algoritmo di predire un valore, dato un input (es: predire il prezzo di un azione).
- Trascrizioni: l'algoritmo osservando una rappresentazione non strutturata di dati trascrive l'informazione in forma testuale; per esempio, fornendo all'algoritmo un messaggio audio esso può trasformalo in una sequenza di parole.
- Imputazione di valori mancanti: attraverso questo task è possibile dare un nuovo esempio $x \in R^n$ mancante di alcune sue entrate x_i e ricevere una predizione dei valori mancanti.

- Denoising: inserendo come input un corrupted example l'algoritmo ci restituisce una predizione dell'esempio giusto (clean example).
- Density estimation: nel applicazione di questo task viene chiesto all'algoritmo di imparare una funzione (probability density function) definita nello spazio delimitato dagli esempi forniti alla macchina. In breve, la density estimation ci permette di catturare la distribuzione di probabilità dei dati e utilizzare la struttura imparata per calcolare altri tasks.

4.3 L'esperienza, E

Gli algoritmi di ML possono essere suddivisi tra unsupervised o supervised in base al tipo d'esperienza utilizzata nel processo d'apprendimento. Gli algoritmi supervised, osservando la relazione tra inupt e output in vari esempi, imparano a predire l'output utilizzando nuovi input mai visti precedentemente. L'obiettivo è insegnare alla macchina a trovare la migliore relazione che collega l'input all'output affinché essa possa in maniera automatica, dato un nuovo input, predire nuovi output. Più formalmente, dato un training set in cui ogni esempio è costituito da un vettore di caratteristiche (features) x e il corrispondente output y, vogliamo trovare la migliore funzione h: X→Y (funzione d'ipotesi) che approssimi la funzione target anche per valori non presenti nel training set.
Al contrario, gli algoritmi unsupervised imparano dai dati analizzando soltanto il vettore di input senza le uscite corrispondenti; l'algoritmo, in maniera automatica, trova schemi, estrapola la distribuzione di probabilità e varie proprietà significative dei dati.

4.4 La misura di performance, P

Per valutare la precisione dell'algoritmo nel eseguire un task, bisogna progettare una misura quantitativa della performance. Generalmente, la misura di performance P è specifica per ogni task: per tasks come la classificazione con input mancanti e la trascrizione si usa come misura l'accuratezza del modello (accurracy, ovvero la proporzione degli esempi per il quale il modello produce l'output corretto) o l'error rate (la proporzione di esempi per il quale il modello produce un output incorretto). Per tasks come la density estimation utilizziamo, invece, metriche diverse che danno un valore continuo per ogni esempio (log-probability).
Affinché l'algoritmo dia risultati accurati anche su esempi mai visti prima, bisogna valutare la performance della macchina anche sul test set, un insieme di dati differenti da quelli utilizzati per allenare la macchina (training set).

4.5 Funzione di Costo (o Funzione di Perdita)

La funzione di costo misura l'accuratezza della ipotesi che vogliamo verificare e, in caso di regressione, viene definita cosi:

$$J(\theta)=\frac{1}{2m}\sum_{i=1}^{m}(h(x^{(i)})-y^{(i)})^2$$

Questa funzione è anche chiamata “Mean squared error”.
Se rappresentassimo graficamente le nostre osservazioni come puntini su un piano x-y, il nostro obiettivo sarebbe quello di tracciare una linea retta tale che la discrepanza quadratica media fra i valori dei dati osservati ed i valori dei dati stimati sia minima.
La funzione costo è generata dalla media dei costi generati da ogni singolo esempio del training set.

4.6 Ottimizzazione: discesa del gradiente Batch

La maggior parte degli algoritmi di Intelligenza Artificiale coinvolgono problemi di ottimizzazione. Per ottimizzazione intendiamo la massimizzazione/minimizzazione di una funzione f(x) di costo modificando il valore di x. Possiamo impiegare varie tecniche di ottimizzazione tra cui, la più utilizzata, è la discesa del gradiente. Attraverso la discesa del gradiente, una tecnica che ci permette di ridurre la f(x) muovendo x a piccoli passi in direzione opposta al segno della derivata, possiamo trovare il minimo della funzioni obiettivo facilmente. L’idea di fondo è quella di minimizzare una funzione di costo J(θ) formata da n parametri θ, aggiornando il valore dei parametri in base alla differenza tra il gradiente negativo di J(θ) e il parametro considerato.
A seconda che l’algoritmo usi, per aggiornare θ_j , tutti gli esempi del training set o soltanto alcuni, il processo di ottimizzazione può essere chiamato diversamente: nel primo caso “batch gradient descent”, nel secondo “stochastic gradient descent”.

Formalmente la discesa del gradiente batch è descritta dalla seguente formula

$\theta_j := \theta_j - a\frac{\partial}{\partial\theta_j}J(\theta_0, \dots, \theta_n)$ (simultaneamente aggiornate per ogni j=0,…,n)
Calcolando $\frac{\partial}{\partial\theta_j}J(\theta_0, \dots, \theta_n)$ otteniamo
$\theta_j := \theta_j - a\frac{1}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}$ per j=o….n

Il segno della derivata parziale è negativo poiché vogliamo spostarci nel verso opposto al gradiente (un vettore che punta nel verso in cui la pendenza della funzione aumenta) per raggiungere il minimo. Modificando il learning rate a possiamo modifiare la “lunghezza del passo”, incrementare oppure diminuire la discesa verso il minimo: learning rate piccoli comportano tempi lunghi di convergenza, learning rate grandi invece rischiano di non trovare il minimo. Quando siamo in presenza di minimi locali l’algoritmo tende a riavviarsi e ad aumentare i tempi di calcolo per verificare se il punto finale di convergenza è un minimo globale .

4.7 Discesa del gradiente stocastica

La discesa del gradiente stocastica è una estensione della discesa del gradiente batch e può essere utilizzato per ottimizzare qualunque funzione convessa su dominio convesso. A differenza del modello precedente, la versione stocastica aggiorna tutti i coefficienti dopo aver esaminato un singolo campione permettendo di convergere verso il minimo

più velocemente; mentre nella versione batch occorre analizzare tutto il training set per compiere un passo, con la versione stocastica si aggiorna leggermente il punto per ogni esempio che si analizza, risparmiando cosi tempo e molti calcoli; in questo modo lo spostamento non va sempre verso il minimo, ma prova a fare un fitting migliore, creando una traiettoria verso il minimo non diritta ma a zig-zag. Lo spostamento verso il minimo per ogni ciclo intero (training set) è più intenso rispetto alla versione batch perché per ogni esempio vi è un aggiornamento del punto.

Ripeti fino a convergenza {
per i:=1,.....,m{
$\theta_j := \theta_j - a(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}$ per ogni j=o....n

4.8 Iperparametri, Train-Validation-Test Sets, Scelta del Modello

E' possibile modificare il comportamento dell'algoritmo modificando alcuni parametri chiamati iperparametri e ottenere, così, diversi modelli utilizzabili per verificare la nostra ipotesi. Per esempio, potremmo essere indecisi sul grado del polinomio da utilizzare, sul coefficiente di penalizzazione del parametro di regolarizzazione λ ,oppure se usare un modello di regressione logistica o una rete neurale. Se i vari modelli venissero allenati sul training set, molto probabilmente si andrebbe incontro al problema del overfitting, infatti venendo addestrati a minimizzare l'errore quadratico generebbero un errore molto basso (sottostima) quando testati per misurare l'accuratezza della performance. Per evitare questo problema è possibile dividere il training set S in tre parti (per esempio, 60% $S_{training}$, 20% $S_{validation}$, 20% S_{test},) e procedere alla scelta del modello (assumiamo un insieme finito di modelli M={M_1, M_2, M_3,.... M_d} dove M_i è un modello polinomiale di ordine i) , nel seguente modo:

1. Allenare ogni modello M_i, sul $S_{training}$ per ottenere la funzione d'ipotesi h_i
2. Scegliere la funzione di ipotesi h_i che produce il minore errore nel $S_{validation}$.
3. Testare la capacità del modello scelto a generalizzare nel test set

In sintesi, ottimizzo i parametri nel training set per ogni modello minimizzando la funzione costo, utilizzo il validation set per scegliere il modello migliore, infine testo la capacità di generalizzare nel test set.
E' importante sottolineare come il test set su cui valutiamo le prestazioni del modello debba essere distinto dalla restante parte del dataset e non avere alcuna influenza su come il modello è stato scelto.

4.9 K-Fold Cross-Validation

Lo svantaggio del dividere il nostro campione di dati in tre parti è quello di testare la performance del modello in un sottoinsieme e avere quindi meno esempi a disposizione. Avere un test set piccolo, per esempio, può creare problemi di robustezza nella stima dell'errore. Il metodo più utilizzato per risolvere questo problema è il k-fold cross, una tecnica statistica che suddivide il dataset totale in k parti uguali e a turno considera la k-esima parte come il validation set e la restante come il training set.

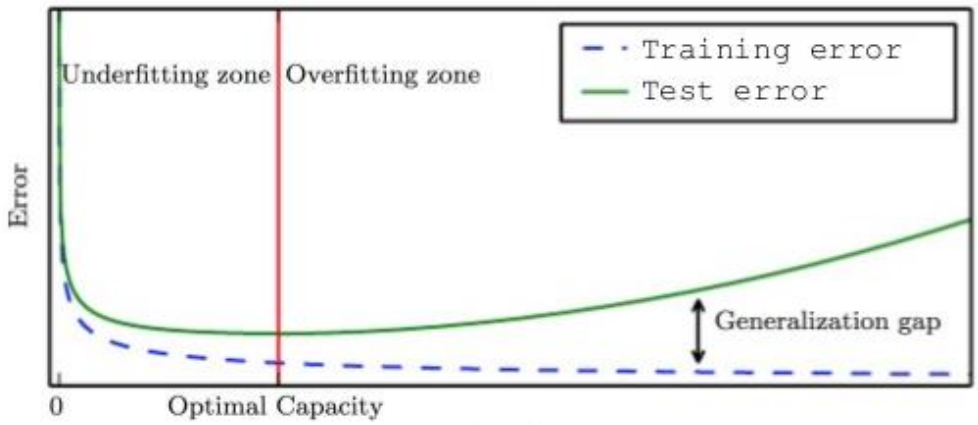
In generale, si suddivide il campione osservato in gruppi, si esclude iterativamente un gruppo alla volta e lo si cerca di predire con i gruppi non esclusi. Più formalmente, si divide casualmente l'insieme delle n osservazioni in k gruppi, o folders, all'incirca di uguale dimensione. Il primo folder viene considerato come un validation set e f è stimata sui restanti k-1 folder. L'errore quadratico medio, MSE, è poi calcolato sulle osservazioni del folder tenuto fuori. Questa procedura è ripetuta K volte; ogni volta scegliendo un folder differente per la validazione ottenendo K stime del test error, $MSE_1, MSE_{2,\dots,} MSE_K$. La stima k-fold CV viene calcolata facendo la media questi valori.

4.10 Capacità, Overfitting e Underfitting

Definiamo "generalizzazione" la capacità dell'algoritmo di eseguire bene il proprio compito utilizzando input su cui non è stato allenato. Affinché un'applicazione abbia successo nel mondo reale vi è la necessità di ottenere risultati accurati utilizzando input mai visti in precedenza, infatti il solo risultato ottenuto nel training set non è un indicatore sufficiente a stabilire l'accuratezza dell'algoritmo.
L'errore generato nell'insieme in cui i parametri sono stati allenati sarà sempre minore dell'errore generato in un altro insieme di dati. Tipicamente alleniamo la macchina su un insieme di esempi (training set) e ne valutiamo la prestazione su un altro mai visto precedentemente (test set). Minimizzando l'errore nel training set (training error) e nel test set (generalization error) otteniamo performance migliori. Il generalization error viene definito come il valore atteso dell'errore su un nuovi esempi non visti nel training set. Due fattori determinano quanto bene funzionerà un algoritmo: la capacità di minimizzare il training error e la capacità di minimizzare la differenza tra il traing error e il generalization error.
Le due situazioni in cui si può incorrere nell'obiettivo di minimizzare gli errori sono l'underfitting e l'overfitting (figura 4.1). L'underfitting (o high bias) fa riferimento a un modello che non è capace di ottenere un errore piccolo sul training set e che di conseguenza non ottiene una buona prestazione. La forma della funzione di ipotesi non si adatta ai dati. Dall'altro estremo, l'overfitting (o high variance), si verifica quando la differenza tra il training error e il test error è troppo grande e la macchina impara informazioni non importanti (noise) che impattano negativamente sulla abilità a generalizzare. Per ridurre il problema dell'overfitting possiamo semplificare il modello riducendo il numero delle features oppure regolarizzare. Alterando la capacità del modello, ovvero la abilità di adattare una grande varietà di funzioni, possiamo controllare questi due aspetti. In generale, i modelli con un alta capacità, memorizzando proprietà del training set non determinanti per la generalization, incorrono in situazioni di overfitting, mentre quelli con bassa capacità, modelli che non si adattano bene ai dati, in underfitting. La capacità del modello deve essere appropriata alla complessità del task e all'ammontare dei training data forniti. Il modello di Vapnik-Chervonenkis (1971) descrive la relazione esistente tra capacità ed errore: all'interno del regime di underfitting, il training error e il generalization error assumono valori alti perché il grado polinomiale è troppo basso e la funzione non si adatta ai dati. Aumentando la capacità del modello, il training error diminuisce, ma aumenta la differenza tra training error e il generalization error entrando cosi nel regime di overfitting; in questo caso, essendo il

grado polinomiale troppo alto, la funzione imparerà caratteristiche del training set non importanti (che generano però un errore basso nel training set) per la generalizzazione.

Figura 4.1 Underfitting vs Overfitting



Fonte: Goodfellow e Bengio (2016), p. 113.

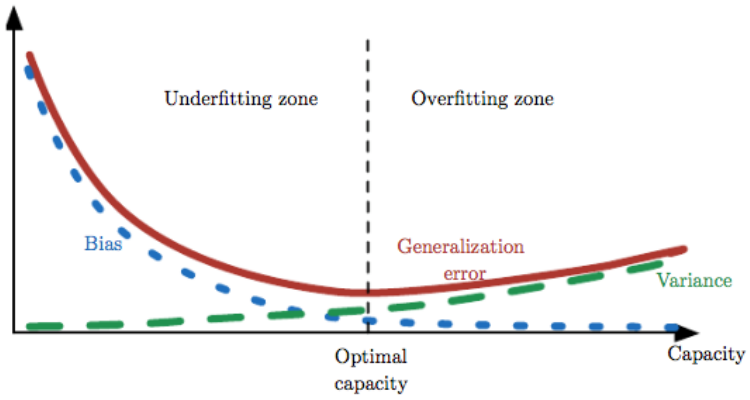
4.11 Bias e Varianza

Il bias di uno stimatore è la differenza tra il valore atteso dello stimatore e il suo vero valore.

$$bias(\widehat{\theta_m}) = E(\widehat{\theta_m}) - \theta$$

Uno stimatore con bias elevato è indice di un modello troppo semplice che non ha buone capacità predittive, come ad esempio un polinomio di grado basso.
La varianza di uno stimatore fornisce un giudizio di quanto la stima che calcoliamo dai dati varia al variare del campione utilizzato. Uno stimatore con varianza elevata è indice di un modello troppo complesso che soffre del problema dell'overfitting e che manca di capacità di generalizzazione.
Varianza e bias costituiscono due misure di errore dello stimatore che devono essere bilanciate in quanto la somma di entrambe ci indica l'errore totale della nostra previsione. Quando analizziamo la performance di un algoritmo di ML, dobbiamo sempre chiederci come ridurre il bias senza aumentare la varianza e viceversa. La maggior parte delle volte la riduzione di una aumenterà conseguentemente l'altra. Uno stimatore desiderabile è quello che ha un MSE (mean squared error) piccolo, ovvero una misura dell'errore della predizione rispetto al valore vero basso. Il MSE incorpora sia la varianza che il bias.
La relazione tra bias e varianza e legata ai concetti di capacità, overfitting e underfitting (figura 4.2). All'aumentare della capacità, il bias tende a diminuire e la varianza ad aumentare e il generalization error ad assumere una forma ad "u".

Figura 4.2. Bias vs Varianza



Fonte: Goodfellow e Bengio (2016), p. 127

4.12 Curva d'Apprendimento

In generale, per curva d'apprendimento intendiamo il rapporto tra il tempo necessario per l'apprendimento (ovvero l'esperienza, intesa come numero di esempi usati nel training set) e le informazioni apprese (apprendimento, inteso come l'errore dell'algoritmo d'apprendimento).
La curva d'apprendimento è uno strumento utilizzato per diagnosticare problemi e migliorare la performance dell'algoritmo.

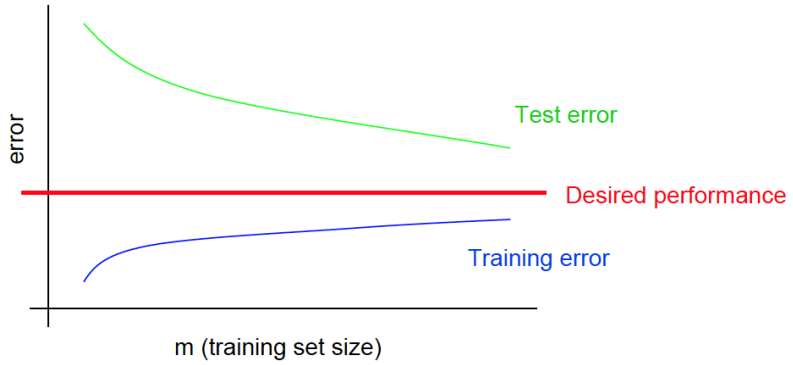
Figura 4.3. Learning curve High Bias

Fonte: NG (2017a).

In regime di high bias (es: un polinomio di grado basso tipo una retta) (figura 4.3), all'aumentare del numero di esempi del training set (più il numero di esempi è ampio più una retta fa fatica ad adattarsi), il training error aumenta e il test error diminuisce fino quasi a coincidere, ottenendo una performance scadente. Un algoritmo che soffre il

problema di high bias non migliorerà aumentando il numero di esempi perché il problema risiede nella forma della funzione che non si adatta ai dati.
In regime di high variance (es: polinomio di grado alto) (figura 4.4) all'aumentare del numero di esempi del training set, il training error aumenta (rimanendo in un intervallo di valori bassi) e il test error diminuisce mantenendo una certa distanza tra di loro (la distanza rappresenta l'errore di generalizzazione). Un algoritmo che soffre il problema di high variance, siccome fatica a generalizzare, migliorerà aumentando il numero di esempi.

Figura 4.4. Learning curve High Variance



Fonte: NG (2017a).

4.13 Consistenza

All'aumentare della numerosità del nostro dataset, lo stimatore converge verso il vero valore. All'aumentare del numero di esempi presi in considerazione nel nostro dataset, la consistency assicura che il bias dello stimatore diminuisca.

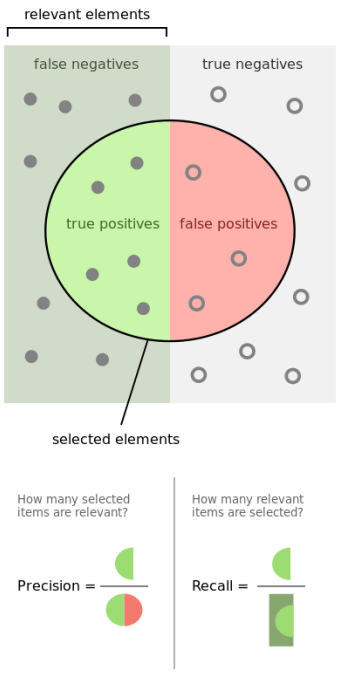
4.14 Più dati o modelli migliori?

Capire se è meglio costruire algoritmi migliori oppure avere a disposizione più dati è un problema dibattuto a lungo nel mondo scientifico. Alcuni, come Peter Norvig (2009), direttore dell'unità di ricerca di Google, sostengono che il semplice aumento del volume dei dati processato dal sistema possa aumentare l'accuratezza della performance , altri sostengono invece che ci sia bisogno di modelli migliori.
Come spiega Andrew NG (2017*a*), professore di Stanford, la verità stà nel mezzo: se possiamo assumere che il vettore di features all'ingresso ha abbastanza informazioni per predire l'output, allora l'aumento dei dati rende la performance migliore. Per ottenere un trade-off ottimale tra bias e varianza dobbiamo quindi utilizzare un algoritmo complesso che si adatti bene ai dati del training set e allo stesso tempo abbia un volume elevato di dati che ci permetta di evitare l'overfitting. In sintesi, si, c'è bisogno di molti dati, ma allo stesso tempo anche di modelli che li interpretino al meglio.

4.15 Precision, Recall, F-score, ROC

Nei problemi di classificazione skewed può succedere che una classe sia più frequente dell'altra. Quando un test non distingue, in maniera netta, i malati dai sani, è necessario calcolare il grado di incertezza della classificazione. E' possibile che in un problema di classificazione di una malattia, per esempio, l'evento y=1, che corrisponde alla casistica in cui il cliente ha contratto la malattia, abbia una probabilità dello 0,5%, mentre il caso opposto y=0 del 99,5%. Questo potrebbe far pensare che il modello possa predirre y=0 con un errore dello 0,5% (accuracy). Per evitare questo problema sono state introdotte due misure alternative all'accuracy (è una misura della distanza tra il valore medio campionario e il vero valore di riferimento), la precision (una misura di esattezza) e il recall (una misura di completezza) (figura 4.5): la precision è il rapporto tra il numero di casi di malattia veri (true positives) e il numero di casi predetti positivi (true positives+false positives), il recall è il rapporto tra il numero di casi di malattia veri (true positives) e il numero di casi di malattia attuali (true positives+false negatives). Se vogliamo aumentare la precisione del modello, in un problema di regressione logistica, possiamo aumentare il livello di confidenza (threshold) ottenendo però un effetto inverso sul recall, viceversa se vogliamo aumentare la completezza e scartare meno casi, abbassiamo il livello di confidenza rinunciando alla precisione. E' possibile ottenere una unica misura che tenga conto della precisione e completezza chiamata F score: $F\ score = 2\frac{PR}{P+R}$

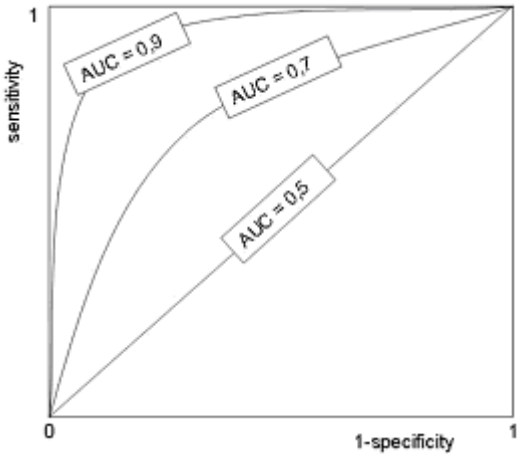
Figura 4.5. Precisione e Recall



Fonte: https://en.wikipedia.org/wiki/Precision_and_recall

Una misura più accurata è la curva ROC (figura 4.6), un grafico che illustra la performance di un classificatore binario al variare del threshold (cutoff). La curva è ottenuta rappresentando in un grafico la frazione di veri positivi (ordinata) e quella di falsi positivi (ascissa) per ogni valore della soglia. L'area sottostante alla curva è una misura dell'accuratezza. Tutti i punti ottenuti nello spazio FP-TP descrivono la curva ROC.

Figura 4.6. Curva ROC



Fonte: https://acutecaretesting.org/en/articles/diagnostic-accuracy--part-1brbasic-concepts-*sensitivity-and-specificity-roc-analysis-stard-statement*

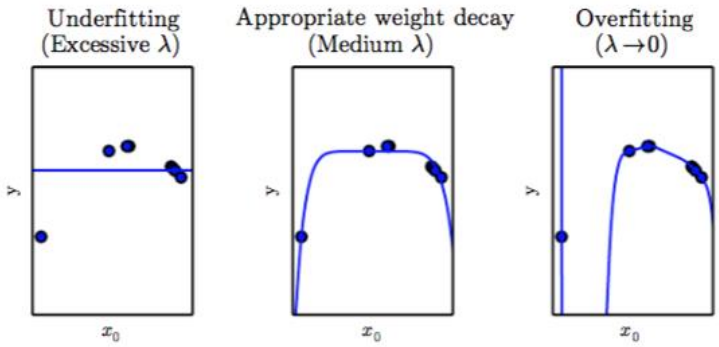
4.16 Regolarizzazione

Nel ML, un obiettivo importante è ottenere modelli che si adattino il più possibile ai dati. Quando il modello si adatta molto bene agli esempi del training set, ma non ha una buona capacità di generalizzazione, siamo in presenza di overfitting. Attraverso la regolarizzazione (figura 4.7) possiamo evitare questo problema: l'idea è quella di penalizzare i modelli complessi con molti parametri che portano all'overfitting. Per regularization intendiamo ogni modifica apportata a un algoritmo di apprendimento con la finalità di ridurre il generalization error ma non il suo training error. In generale, possiamo regolarizzare un modello aggiungendo una penalità chiamata parametro di regolarizzazione λ alla funzione costo. Nel caso della regressione lineare, la funzione costo diventa:

$$J(\theta)=\frac{1}{2m}\ \sum_{i=1}^{m}(h(x^{(i)})-y^{(i)})^2+\lambda\ \sum_{j=1}^{n}\theta_j^2$$

Sommando questo parametro alla funzione costo possiamo penalizzare alcuni parametri senza agire direttamente sul grado del modello (cambiare la forma della nostra funzione d'ipotesi).
Attraverso la modifica del parametro λ possiamo controllare la capacità del modello e prevenire il problema del overfitting. Non esistono forme di regolarizzazione universali, ma forme di regolarizzazione migliori nella risoluzione di un particolare task.

Figura 4.7. L'Effetto della regolarizzazione



Fonte: Goodfellow e Bengio (2016), p. 118.

4.17 Statistica Bayesiana

Il teorema di Bayes (Goodfellow e Bengio, 2016) esprime la probabilità dell'avvenimento di θ dato $x^{(1)}, \ldots., x^{(m)}$ come rapporto tra la probabilità condizionata dell'avvenimento di $x^{(1)}, \ldots., x^{(m)}$ dato θ per la probabilità di θ e la probabilità di $x^{(1)}, \ldots., x^{(m)}$:

$$p\left(\theta \middle| x^{(1)}, \ldots., x^{(m)}\right) = \frac{p\left(x^{(1)}, \ldots., x^{(m)} \middle| \theta\right) p(\theta)}{p(x^{(1)}, \ldots., x^{(m)})}$$

Dove P(A|B) viene detta probabilità a posteriori.
La statistica Bayesiana utilizza la probabilità per riflettere il grado di certezza della conoscenza. Prima di osservare i dati, rappresentiamo la nostra conoscenza sul parametro θ usando una distribuzione a priori.
Nella statistica frequentista/classica θ è un valore definito (non casuale, non ha una distribuzione di probabilità) ma non noto, ed è nostro obiettivo stimarlo attraverso procedure statistiche (es: massima verosomiglianza). Nella statistica baysiana θ è una variabile casuale il cui valore è non noto; in questo approccio dobbiamo specificare una distribuzione a priori p(θ) che esprime le nostre conoscenza a priori sul parametro (la distribuzione è soggettiva, se un individuo ha più informazioni di un altro riesce a ottenere una dsitribuzione più puntuale). Possiamo quindi, osservando il training set, creare una distribuzione a posteriori. Nella statistica classica, invece, la probabilità di un valore è considerata quasi una legge della natura, non c'è spazio per interpretazioni personali
Due importanti differenze rispetto alla statistica classica (frequentista): l'approccio bayesiano fa predizioni utilizzando l'intera distribuzione del parametro e non utilizzando solo una stima del parametro, inoltre l'utilizzo di una distribuzione a priori permette di aggiornare la distribuzione di probabilità tenendo in considerazione delle preferenze del utente.

4.18 Machine Learning vs Statistica tradizionale

La questione di "quale sia la differenza tra ML e la statistica tradizionale" è stata posta innumerevoli volte negli ultimi anni, dando vita a un dibattito tra le due comunità scientifiche ancora oggi irrisolto. I dubbi vengono generati dal fatto che il ML appartiene a un settore ibrido, che fa uso di metodi statistici. Al riguardo esistono due correnti di pensiero: chi ritiene che ci siano differenze sostanziali tra i due approcci disciplinari e chi ritiene che non siano rilevanti.
In mancanza di una visione comune, in questo lavoro si fa principalmente riferimento alla prospettiva di Fawcett e Hardin (2017), due professionisti ed esperti del settore (il cui orientamento è allineato con quella di Axyon AI), che ritengono che le differenze tra ML e Statistica siano sostanziali. Infatti, secondo Fawcett e Hardin, la differenza non risiede solo negli algoritmi, ma negli obiettivi e nei metodi. Nel ML l'obiettivo principale è di compiere un task nel migliore dei modi. Testare la validità di un modello non ha grossa importanza, in quanto l'attenzione è rivolta alla performance e all'ottimizzazione. I ML engineers non sono preparati a testare la validità di un modello, perché il modello è lo strumento per raggiungere la performance e sono i risultati sul test set che costiutiscono la prova che il modello funziona. Anche per questo ragione, il legame con le assunzioni è meno importante: queste possono essere violate se giustificate da buoni risultati. Grazie a questa maggiore libertà è possibile utilizzare una maggiore varietà di modelli. Il ML è applicato a dataset grandi e ad alta dimensionalità e più dati vengono utilizzati, più accurata è la predizione. I ML engineers destinano, infatti, la maggior parte del tempo nella preparazione dei dati.
La Statistica, invece, è più incentrata sulla validazione dei modelli, sull'inferenza e sulla stima accurata di parametri. L'obiettivo finale è l'analisi, condotta rigorosamente. Ogni assunzione deve essere rispettata, i test condotti riportati e controllati. Questo garantisce che il modello sia appropriato ai dati utilizzati e alle assunzioni proposte.
Una prospettiva opposta è invece assunta da Larry Wasserman, professore del dipartimento di statistica e machine learning dell'universita Carnegie Mellon, che ritiene che non esistano differenze sostanziali tre i due ambiti disciplinari, dato che entrambi sono concentrati sulla stessa domanda: "come è possibile imparare dai dati?". L'unica differenza individuata da Wasserman è che la statistica enfatizza l'inferenza in problemi a bassa dimenisonalità, mentre il ML in problemi ad alta dimensionalità[1]. Nella stessa corrente corrente di pensiero si può collocare anche Robert Tibshirani[2], professore di statistica dell'università di Stanford, che sostiene che il ML e la statistica utilizzano nomi diversi per esprimere gli stessi concetti, come -per esempio- supervised learning invece di regressione/classificazione, learning invece di fitting, pesi invece di parametri ecc. Considerando che la regressione lineare può essere vista come una rete neurale feedforward senza hidden layers, con un neurone d'output e una funzione d'attivazione lineare (Kaastra e Boyd, 1996), il loro punto di vista presenta una certa ragionevolezza.

4.19 Progettazione di un algoritmo di Machine Learning

Possiamo sintetizzare la progettazione di un sistema di ML in cinque fasi:

[1] Cfr. il blog di L. Wasserman: normaldeviate.wordpress.com
[2] Cfr. Il corso in "Modern Applied Statistics: Elements of Statistical Learning" di R. Tibshirani (Stanford, Winter 2018), http://statweb.stanford.edu/~tibs/stat315a/

1. Analisi e pre-processing dei dati
2. Selezione delle features
3. Ottimizzazione dei parametri e addestramento nel training set
4. Scelta del modello nel validation set
5. Valutazione della performance sul test set

Siccome i sistemi di ML imparano dai dati, è essenziale pulire i dati da anomalie e incompletezze. Attraverso alcune operazioni, come la rimozione degli outlier, dei duplicati e del rumore, possiamo ottenere dati affidabili da elaborare. E' molto importante anche pre-elaborarli: attraverso tecniche come la normalizzazione e il re-scaling possiamo adattare i dati a seconda dell'obiettivo che si desidera raggiungere. Per re-scaling s'intende un cambio di scala, mentre per normalizzazione il restringimento della variabilità a un intervallo predefinito di valori. Queste operazioni sono necessarie perché gli algoritmi hanno difficoltà a processare dati troppo eterogenei.
Per ridurre i tempi d'addestramento, evitare problemi di dimensione e ottenere una capacità di generalizzazione migliore (diminuendo il rischio di overfitting) è possibile selezionare le features più significative. Si possono individuare tre tecniche principali di selezione delle features: i filtri, i wrapper methods e gli embedded methods. Attraverso i filtri si utilizza una misura proxy (es: test statistici) per creare un ranking tra le varie features; attraverso i wrapper methods (es: forward selection, backward elimination) s'apprende quali featurs sono più significative utilizzando un modello predittivo; infine attraverso gli embedded methods (es: LASSO, RIDGE) s'impiega l'algoritmo interno al processo d'apprendimento (es: regolarizzatore) per selezionare le features migliori.
Nella terza fase si ottimizzano i parametri dei modelli, ottenendo per ogni modello una funzione d'ipotesi, e successivamente, all'interno del validation set, s'individua il modello che genera l'errore minore. Per concludere, si valuta l'accuratezza della performance del modello scelto sul test set.

4.20 Un primo esempio: la regressione

Come scritto all'inizio del capitolo utilizziamo $x^{(i)}$ per designare la variabile "input", chiamata anche features, e $y^{(i)}$ per designare la variabile "output" che cerchiamo di predire; chiamiamo $(x^{(i)}, y^{(i)})$ l'esempio (training example) (i), e l'insieme degli esempi usati per l'allenamento dell'algoritmo "training set". Utilizziamo X per indicare lo spazio degli input, e Y quello degli ouput.
La regressione lineare stima il valore atteso condizionato di una variabile dipendente, Y, dati i valori di altre variabili indipendenti $X_1, \dots, X_k$: $E[(Y|X_1, \dots, X_k)$. Viene definita lineare perché individua la miglior retta capace di approssimare l'andamento dei dati.
La forma più semplice è la seguente:

$h(x) = \theta_0 + \theta_1 x$
dove θ_0 rappresenta l'intercetta della retta e θ_1 il coefficiente angolare.

Quando il nostro scopo è predire un valore obiettivo Y a partire da un vettore di input $x \in R^n$ allora il nostro algoritmo viene definito regressione multipla lineare. Per

esempio, predire il prezzo di una casa (Y) utilizzando delle "features" $(X_1, \dots, X_k)$ che descrivano la casa.
Supponiamo (NG, 2017) di disporre di molti esempi di case, $(x^{(i)}, y^{(i)})$, il nostro obiettivo è trovare una funzione y=h(x) tale che $y^{(i)} \approx h(x^{(i)})$ per ogni esempio. Vogliamo trovare una funzione h(x) che potrà essere utilizzata come predittore del prezzo di una casa anche quando utilizziamo features per una nuova casa dove il prezzo non è noto.
Utilizziamo una funzione lineare per rappresentare la nostra funzione d'ipotesi:

$h(x) = \sum_{i=0}^{n} \theta_i x_i = \theta^T x$
dove $h_\theta(x)$ rappresenta una famiglia di funzioni parametrizzate dalla scelta di $\theta = (\theta_0, \theta_1, \theta_2 \dots., \theta_n)$

Il nostro obiettivo è trovare un valore di θ tale che $h(x^{(i)})$ è vicina il piu possibile a $y^{(i)}$, quindi cerchiamo un valore di θ che minimizzi la funzione di costo:

$\theta_{min} = \min_{\theta} \mathrm{J}(\theta)$
$\mathrm{J}(\theta) = \frac{1}{2} \sum_{i=1}^{m} \sum_i (h(x^{(i)}) - y^{(i)})^2 = \frac{1}{2} \sum_{i=1}^{m} \sum_i (\theta^T x^{(i)} - y^{(i)})^2$
m è il numero di esempi del training set
$(\theta^T x^{(i)} - y^{(i)})$ è l'errore di predizione sul campione i-esimo, calcolato come differenza tra output del modello e il vero risultato.

Questa funzione è chiamata "mean squared error".
Vogliamo trovare il valore di θ che minimizzi J(θ) utilizzando un algoritmo chiamato discesa del gradiente.

Se, per esempio, rappresentassimo la nostra funzione di costo come in figura 4.7, il nostro obiettivo sarebbe raggiungere il punto di minimo utilizzando la derivata della nostra funzione di costo. Dato che la derivata è l'inclinazione della retta tangente nel punto scelto, essa darà la direzione verso cui muoversi. Facendo piccoli passi di dimensione a (definita learning rate) nella direzione più ripida, scendiamo lungo la funzione di costo, raggiungendo il punto di minimo. A seconda della collocazione del punto iniziale da cui iniziamo a differenziare è possibile raggiungere punti di minimo diversi (frecce rosse in figura 4.8). Nella regressione lineare si raggiunge sempre un punto di minimo globale.
L'idea principale è di partire da θ_j casuali e aggiornarli in modo da raggiungere il minimo, ovvero "scendere "la funzione nel verso opposto (segno negativo prima della derivata) rispetto al gradiente nel punto (il gradiente è un vettore che punta nel verso in cui la pendenza della funzione aumenta).
Il gradiente di discesa è rappresentato dalla seguente funzione:

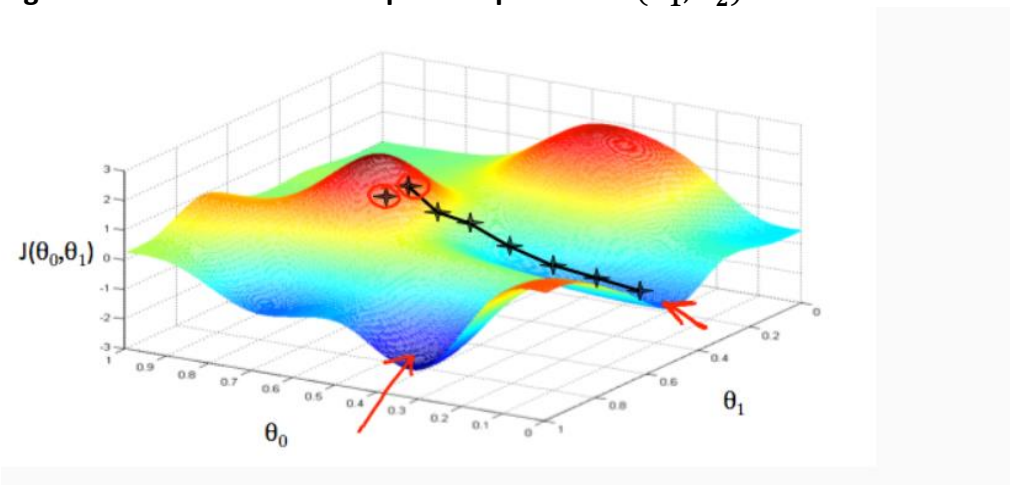
$\theta_j := \theta_j - a \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$ con j=0,1 rappresentati le features

Ad ogni iterazione J, l'algoritmo simultaneamente aggiorna i parametri.
Questo metodo, che utilizza ogni esempio del training set, è chiamato *batch gradient descent*.
Regolarizzando il modello:

$$J(\theta)=\frac{1}{2m}\sum_{i=1}^{m}(h(x^{(i)})-y^{(i)})^2+\frac{1}{2m}\lambda\sum_{j=1}^{n}\theta_j^2$$

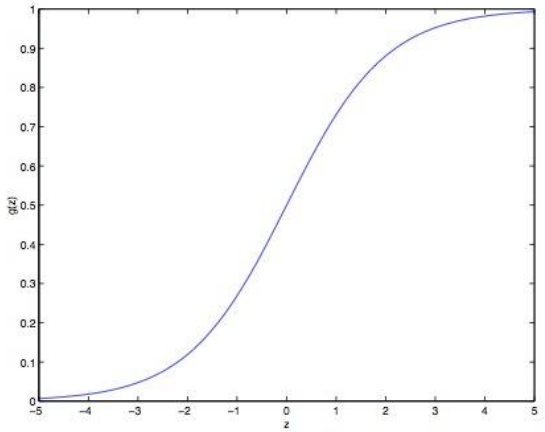
All' aumentare di λ, il nostro modello tenderà a penalizzare tutti i parametri, ottenendo una funzione piatta (underfitting).

Figura 4.8. Funzione di costo per due parametri $(\boldsymbol{\theta_1}, \boldsymbol{\theta_2})$



Fonte: NG (2017b).

Figura 4.9. La funzione Sigmoid



Fonte: NG (2017a).

4.21 Un altro esempio: la classificazione (regressione logistica)

In molti casi è difficile utilizzare modelli lineari per comprendere problemi di tipo economico, per questo motivo applichiamo a ciascun vettore d'ingresso una trasformazione non lineare. Il problema della classificazione è simile a quello della regressione eccetto che i valori che vogliamo predire non appartengono a un insieme continuo, bensì discreto. Prendiamo come esempio un problema di classificazione binaria nel quale y può assumere solo due valori 0 e 1 (NG A., 2017*a*).

Affinché la nostra funzione di ipotesi $h_\theta(x)$ assuma valori tra 1 e 0 ($0 \leq h_\theta(x) \leq 1$) inseriamo $\theta^T x$ all interno di una funzione logistica (detta anche funzione Sigmoid) (figura 4.9):

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1+e^{-\theta^T x}}$$
$$g(z) = \frac{1}{1+e^{-z}}$$

La funzione g(z), collegando qualunque numero reale a (0,1), è utile a trasformare una qualunque funzione in una funzione utile alla classificazione.
Siccome la funzione logistica varia tra 0 e 1, possiamo interpretare le uscite in termini probabilistici.
Dato

$$\begin{cases} p\left(y^{(i)} = 1 \middle| x^{(i)}; \theta\right) = h_\theta\left(x^{(i)}\right) \\ p\left(y^{(i)} = 0 \middle| x^{(i)}; \theta\right) = 1 - h_\theta\left(x^{(i)}\right) \end{cases}$$

riscrivibile in maniera più compatta nel seguente modo

$$p\left(y^{(i)} \middle| x^{(i)}\right); \theta) = h_\theta\left(x^{(i)}\right)^{y(i)} (1 - h_\theta\left(x^{(i)}\right))^{1-y(i)}$$

possiamo trovare un espressione, attraverso il calcolo della funzione di verosimiglianza, della probabilità che, fissati i parametri θ, al set di ingressi $x^{(i)}$ corrispondano le uscite $y^{(i)}$ per tutti gli i=1,…,m:

$$L(\theta) = L(\theta; X; y) = p(y|X); \theta)$$
$$= \prod_{i=1}^{m} p\left(y^{(i)} \middle| x^{(i)}\right); \theta) = \prod\nolimits_{i=1}^{m} h_\theta\left(x^{(i)}\right)^{y(i)} (1 - h_\theta\left(x^{(i)}\right))^{1-y(i)}$$

$L(\theta)$ rappresenta la probabilità che, fissati i parametri θ, agli ingressi $x^{(i)}$ corrispondano le uscite $y^{(i)}$ (assumendo che i campioni siano i.i.d).
Trasformando in logaritmo la precedente formula e massimizzandola otteniamo:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} = [y^{(i)}\log(h\theta(x^{(i)})) + (1 - y^{(i)})\log(1 - h\theta(x^{(i)}))]$$

Quindi, fissata una soglia (threshold), possiamo stabilire la classe d'appartenenza a seconda che $h_\theta(x)$ sia maggiore o uguale alla soglia (classe 1) o minore (classe 0). Per esempio, $h_\theta(x)$ = 0,6 ci dice che la probabilità che l'output sia 1 è del 60%.
Questo tipo di modello viene chiamato anche discriminativo probabilistico, perché in seguito all'addestramento su un insieme di dati, è in grado di valutare quale tra le uscite è la più probabile (stimano la probabilità condizionata $p((y|x)$, ovvero la probabilità che dato l'ingresso x si generi l'uscita y).
In termini non probabilistici, per poter ottenere la nostra classificazione 0-1, possiamo utilizzare la seguente regola:

$$Y = \begin{cases} 1 \; se \; h_\theta(x) \geq \text{threshold} \\ 0 \; se \; \; h_\theta(x) < \text{threshold} \end{cases}$$

Alcuni esempi:
z=0, e^0=1 ⇒ g(z)=1/2
z→∞, $e^{-\infty}$→0 ⇒ g(z)=1
z→−∞, e^{∞}→∞ ⇒ g(z)=0

quindi:

$h_\theta(x) = g(\theta^T x)$≥0.5 quando $\theta^T x$≥0

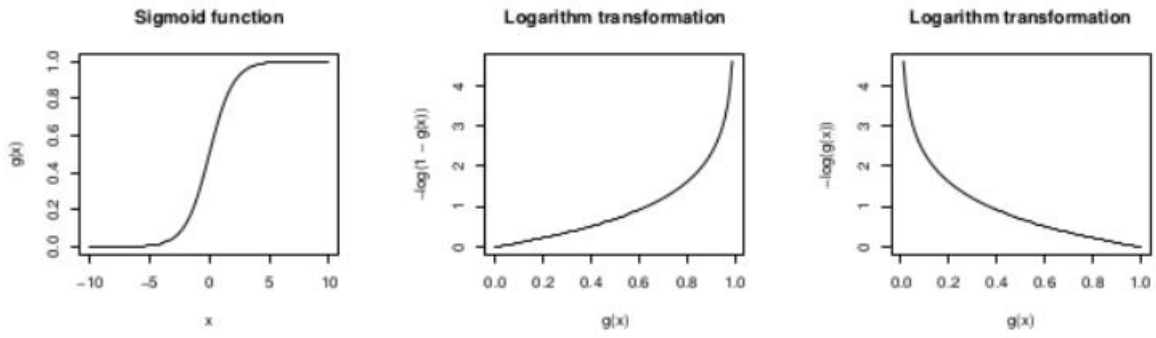
Viene definita decision boundary la linea che separa l'area y=0 da quella in cui y=1
Siccome questo tipo di funzione genera molti punti di ottimo locale (non è una funzione convessa) non possiamo usare la stessa funzione costo della regressione lineare.
Possiamo definire la funzione di costo J(θ) come la sommatoria delle funzione di costo di tutti gli esempi del training set:

J(θ)=$\frac{1}{m}$ $\sum_{i=1}^{m} cost(h(x^{(i)}), y^{(i)})$

In generale la funzione di costo può assumere due forme in base all'output (figura 4.10):

Cost($h_\theta(x), y$) = -log($h_\theta(x)$) se y=1
Cost($h_\theta(x), y$) = -log($1 - h_\theta(x)$) se y=0

Figura 4.10. La funzione di costo in relazione all'output



(a) *Sigmoid function.* **(b)** *Cost for* $y = 0$. **(c)** *Cost for* $y = 1$.

Fonte: NG (2017a).

Se la risposta corretta è 0 (y=0), allora l'errore sarà zero (funzione costo=0) se il nostro algoritmo darà come risultato zero.
Formalmente:

Cost($h_\theta(x), y$)=0 se $h_\theta(x)$=y

Se, invece, il risultato dell'algoritmo è sbagliato, la funzione costo tenderà a infinito:

Cost($h_\theta(x), y$)→∞ se y=0 e $h_\theta(x)$→1
Cost($h_\theta(x), y$)→∞ se y=1 e $h_\theta(x)$→0

Possiamo riscrivere la nostra funzione di costo in maniera piu compatta:

$\text{Cost}(h_\theta(x), y) = -\text{y}\log(h_\theta(x)) - (1 - \text{y})\log(1 - h_\theta(x))$
E di conseguenza la nostra intera funzione di costo:

$$J(\theta) = -\frac{1}{\text{m}}\sum_{\text{i}=1}^{\text{m}}[\text{y}^{(\text{i})}\log(\text{h}\theta(\text{x}^{(\text{i})})) + (1 - \text{y}^{(\text{i})})\log(1 - \text{h}\theta(\text{x}^{(\text{i})}))]$$

Attraverso una successiva implementazione vettoriale otteniamo:

$h = g(X\theta)$
$$J(\theta) = \frac{1}{m}(-y^T\log(h) - (1 - y^T)\log(1 - h)$$

Utilizzando la discesa del gradiente possiamo ottimizzare la funzione costo (la regola di aggiornamento dei pesi è la stessa della regressione lineare):

$\theta_j \coloneqq \theta_j - \frac{a}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}$ (simultaneamente aggiornando tutti θ_j)

Inserendo il parametro di regolarizzazione, invece, la funzione costo diventa:

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m} = \left[y^{(i)}\log\left(h\theta\left(x^{(i)}\right)\right) + \left(1 - y^{(i)}\right)\log\left(1 - h\theta\left(x^{(i)}\right)\right)\right] + \frac{1}{2m}\lambda\sum_{j=1}^{n}\theta_j^2$$

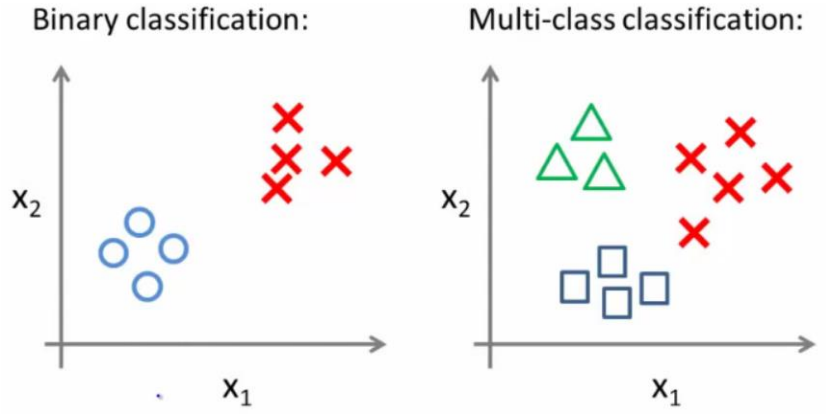
Mentre la regola di aggiornamento dei pesi diventa:

$\theta_0 \coloneqq \theta_0 - \frac{a}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_0^{(i)}$
$\theta_j \coloneqq \theta_j(1 - a\frac{\lambda}{m}) - \frac{a}{m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})x_j^{(i)}$ per j=1,2,…,n

Quando siamo in presenza di più di due categorie y=(0,1….n) (caso multiclasse) (figura 4.11), dividiamo il nostro problema in n+1 problemi di classificazione binaria e prediciamo, per ognuno dei problemi, la probabilità che y sia membro di una delle nostre classi:

$y \in (0,1 \dots, n)$
$h_\theta^{(0)}(x) = P\big((y = 0|x); \theta\big)$
$h_\theta^{(1)}(x) = P\big((y = 1|x); \theta\big)$

Figura 4.11. Classificazione binaria vs Classificazione multiclasse



Fonte: NG (2017a).

In pratica, scegliamo una classe da dividere da tutto il resto e ripetiamo la stessa operazione di classificazione binaria per tutti gli altri casi, infine quando facciamo una predizione su un nuovo x scegliamo la classe che massimizza $h_\theta(x)$.

CAPITOLO 5
Il Deep Learning (reti neurali profonde)

Uno tra i primi algoritmi d'apprendimento creati negli anni '40 è un modello computazionale di neuroni biologici che cerca di ricreare il processo d'apprendimento del cervello umano.
L'architettura mira a riprodurre quello delle reti neurali biologiche, composta da un gran numero di neuroni collegati tra loro in modo diverso secondo l'obiettivo da raggiungere. In generale, la parte centrale del neurone viene chiamato soma, la quale acquisisce informazioni dall'esterno mediante i dendriti ed emette segnali verso l'esterno attraverso l'assone. A seconda che l'input ricevuto superi o meno una certa soglia di attivazione, il neurone si può attivare e emettere un segnale. I neuroni sono interconnessi mediante collegamenti che modulano la rilevanza dei segnali trasmessi; al neurone, quindi, giunge un impulso pesato.

5.1 Modelli non lineari

I modelli lineari, come la regressione lineare o logistica, sono molto utilizzati perché semplici da implementare ma, spesso, risultano essere limitanti perché colgono solo fenomeni lineari. Per esempio, i classificatori lineari, in un problema di classificazione delle immagini, possono dividere lo spazio degli input in poche classi rendendo sensibile il modello a variazioni irrilevanti degli input come lo sfondo, la posizione, le luci e poco sensibili a variazioni piccole rilevanti. Per rappresentare funzioni non lineari di x, possiamo trasformare l'input $\emptyset(x)$ di un modello lineare, dove $\emptyset$ è una trasformazione non lineare (alternativamente è possibile applicare Kernel e ottenere il medesimo risultato). Questa trasformazione non lineare applicata a ciascun vettore d'ingresso ci permette di aumentare il numero di features.
Nella regressione lineare si può rappresentare un singolo ingresso nel seguente modo:

$$h\left(x^{(i)}\right) = \sum_{j=1}^{m} \theta_j x_j^{(i)} = \theta^T x^{(i)}$$

Introduciamo la non linearità applicando $\emptyset(x)$ all'ingresso $x^{(i)}$; il risultato $\emptyset(x^{(i)})$ sarà un vettore di componenti più numeroso, ottenuto dalla combinazione non lineare delle n componenti di partenza:

$$h\left(x^{(i)}\right) = \sum_{j=1}^{m} \theta_j \emptyset(x^{(i)})_j = \theta^T \emptyset(x^{(i)})$$

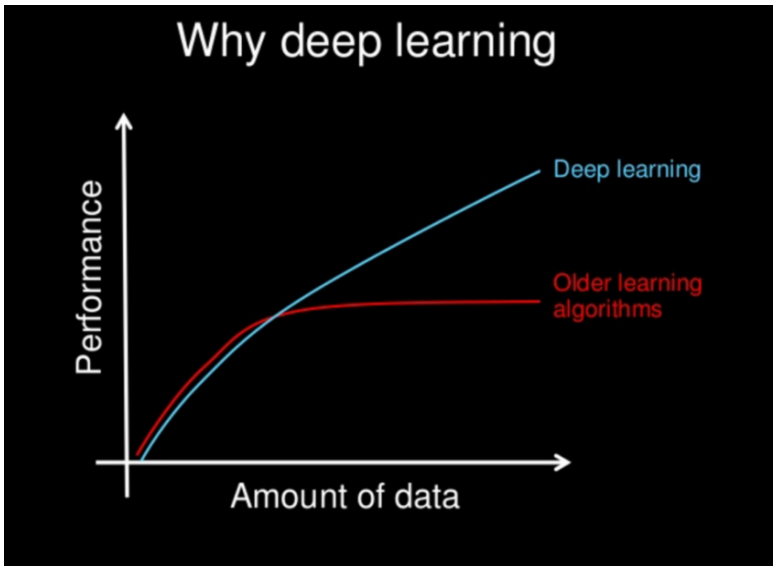
Una delle principali differenze tra i modelli lineari e quelli non, è che, durante la minimizzazione della funzione costo, la convergenza non è garantita.

5.2 Deep Learning (reti neurali profonde)

Il Deep Learning, sottoinsieme del ML, permette a modelli computazionali composti da vari livelli d'apprendimento d'imparare la rappresentazione dei dati attraverso livelli multipli d'astrazione.
Più precisamente Deng e Yu (2014) hanno definito il concetto di Deep Learning come una classe di algoritmi di ML che:
-utilizza diversi livelli di unità non lineari per l'estrazione di features. Ogni livello usa l'output generato dal livello precedente (effetto cascata). Algoritmo può essere supervised o unsupervised.
-sono basati sull'apprendimento di vari livelli di features (unsupervised) o rappresentazione dei dati. Livelli più alti di features derivano dai livelli piu bassi secondo una rappresentazione gerarchica.
-sono parte del settore degli algoritmi di ML del Representation Learning.
-apprendono vari livelli di rappresentazione che corrispondono a diversi livelli d'astrazione. I livelli formano una gerarchia di concetti.
Questi modelli puntano a imparare gerarchie di caratteristiche, in cui le caratteristiche dei livelli superiori sono un insieme di quelle di ordine minore.
Le informazioni vengono elaborate attraverso vari livelli d'apprendimento, partendo da concetti semplici, e combinandoli insieme, si possono apprendere nozioni sempre più complesse, inoltre, più il sistema viene allenato, utilizzando modelli complessi e grandi quantità di dati, più la performance migliora.
Questo processo d'apprendimento permette di rispondere alle nuove sfide proposte dall'Intelligenza Artificiale, problemi difficili da definire ma facili da calcolare.
Negli anni sono stati proposti numerosi approcci per apprendere l'intrinseca struttura dei dati, lineari o non, supervised o unspuervised. I modelli di ML tradizionali non ottengono buoni risultati nel processare dati in forma grezza. Il Deep Learning, sottoinsieme degli algoritmi di Representation Learning, permette di estrarre dai dati le features migliori evitando che vengano progettate dagli ingegneri. Il Representation Learning è un insieme di tecniche che permettono a un sistema di scoprire automaticamente la rappresentazione più appropriata dei dati senza grossi processi d'ingegnerizzazione.
Oggi le architetture Deep sono molto utilizzate in quanto permettono di trovare strutture complesse in dati ad alta dimensionalità e possono essere applicate in molti campi diversi. La profondità della rete permette di implementare funzioni complesse degli input, aumentare la sensibili ai dettagli e l'insensibilità a grandi variazioni (questo, per esempio, permette di distinguere un cane da una volpe rimanendo insensibili a variazioni irrilevanti come lo sfondo, la luce, oggetti).
L'interesse per queste architetture è stato ritrovato nel 2006 (LeCun, Bengio e Hinton, 2015) quando un gruppo di ricercatori del Canadian Instuitute for Advance Research ha introdotto una procedura unsupervised learning che permetteva di creare livelli di features selezionate senza conoscere l'output-target, processo d'apprendimento simile a quello di umani e di animali, i quali scoprono la struttura del mondo osservandolo e non conoscendo già il nome di ogni oggetto. L'idea di base introdotta nel paper di Hinton, Osindero e Yee-Whye Teh " A fast learning algorithm for deep belief nets" era quella di allenare ogni livello attraverso una procedura unsupervised (pre-training) permettendo di ottenere risultati molto migliori di quelli precedenti in cui si sceglieva un peso di partenza casuale. Pre allenando i livelli, il peso poteva essere inizializzato a valori

sensibili. La selezione delle features migliori, attraverso un pre processamento, ha inoltre permesso di ridurre la dimensionalità del problema ("the curse of dimensionality"). Inoltre, come riporta Andrew Ng in un pitch fatto al ExtractConf2015 intitolato "What data scientists should know about Deep Learning" uno dei punti di forza del Deep Learning è la portata dei dati. All'aumentare della profondità della rete e dell'ammontare dei dati, la performance continua a crescere (figura 5.1). Riportando parte del discorso: "for most flavors of the old generations of learning algorithms … performance will plateau. … Deep Learning … is the first class of algorithms … that is scalable. … performance just keeps getting better as you feed them more data".
Il Deep Learning ha portato a risultati eccezionali in settori come la classificazione d'immagini e il riconoscimento vocale, e ha superato le performance di molti altri algoritmi di ML anche nel riconoscimento naturale del linguaggio, in particolare l'analisi del *sentiment*, traduzione automatica e question answering. Questi modelli vengono molto utilizzati nella predizioni di serie temporali grazie al forte potere predittivo out of sample.

Figura 5.1. Deep Learning



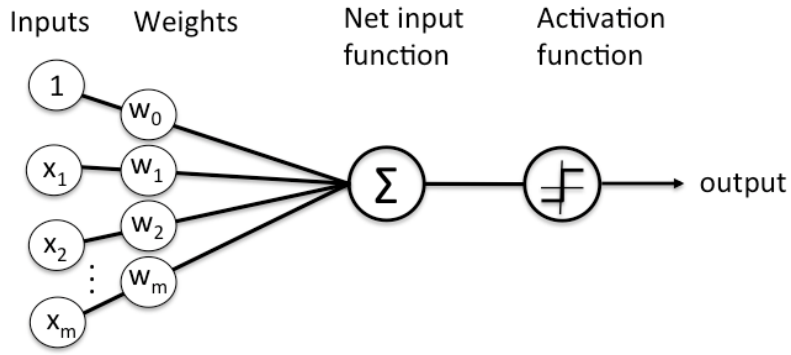
Fonte: NG (2017a).

5.3 Deep FeedForward Network

Le reti neurali feedforward (figura 5.2) sono il modello più famoso e utilizzato nel Deep Learning. Definiscono una funzione $y = f(X;\theta)$ e imparano i valori dei parametri θ che restituiscono la miglior approssimazione della funzione. Vengono definiti feedforward perché l'informazione scorre dal vettore delle entrate x, passando attraverso i livelli intermedi, fino all'output y, senza feedback. Vengono definite "reti" perché sono rappresentate da più funzioni composte $f(x) = f^{(3)}(f^{(2)}\left(f^{(1)}(x)\right))$, in cui $f^{(1)}$ viene definita "input layer" mentre l'ultima funzione viene definita "output layer". Il termine deep indica l'utilizzo di reti profonde con molti layer (livelli) (figura 5.3). L'algoritmo

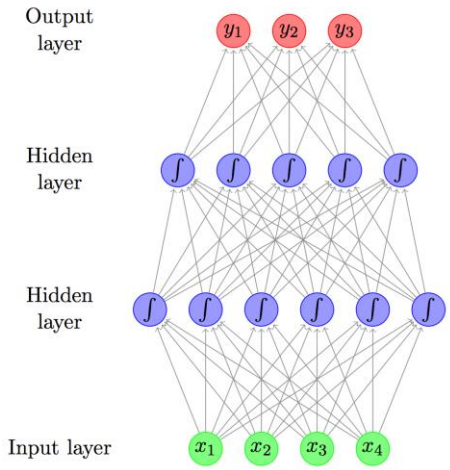
decide come usare i layers per produrre l'output desiderato, senza mostrare i risultati per i layers intermedi (da qui il nome blackbox), rendendo difficile la comprensione dei risultati finali: questi layers vengono definiti "nascosti" (hidden layer).

Figura 5.2. Deep FeedForward Network



Fonte: deeplearning4j.org

Figura 5.3. Deep FeedForward Network (multilayer)



Fonte: Goldberg (2016)

Matematicamente parlando, l'unità di calcolo elementare della rete, il neurone, esegue una trasformazione non lineare di un vettore di ingressi (features) x, fornendo in corrispondenza una uscita scalare y(x). L'architettura più semplice di una rete neurale prevede che gli ingressi siano moltiplicati per dei pesi e che la somma algebrica pesata degli ingressi venga confrontata con un valore soglia per decidere quale uscita fornire: 1, se la somma pesata degli ingressi è maggiore del valore soglia; 0 se minore.
Si consideri il seguente esempio. Date tre features x_1, x_2, x_3 e i relativi output a (activation) appartenenti all'input layer (layer 0), quattro unità nascoste (layer1) e un neurone di output (layer 2), possiamo rappresentare la rete nel seguente modo:

$x_1 = a_1^{(0)}$
$x_2 = a_2^{(0)}$
$x_3 = a_3^{(0)}$
$x^{(i)}$ fa riferimento all'esempio del training set i
$a_j^{(l)}$ fa riferimento all'attivazione della unità j nel layer l

Con una funzione logistica di attivazione g(x):

$$g(x) = \frac{1}{1 + \exp(-w^T x)}$$

La funzione logistica g(x) prende le tre features x_1, x_2, x_3 in input e produce come risultato un valore y.
Possiamo rappresentarla anche cosi:

$z = w^T x + b \approx \theta^T x$
$a = \sigma(z)$
$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

L'output layer finale calcolerà:

$z_1^{(2)} = {w_1^{(2)}}^T a^{(1)} + b_1^{(2)}$
$a_1^{(2)} = g(z_1^{(2)})$

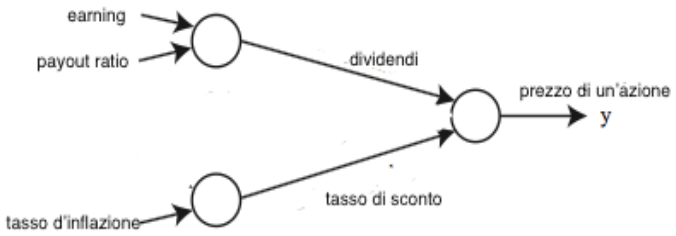
Dove $a^{(1)} = \begin{matrix} a_1^{(1)} \\ a_2^{(1)} \\ a_3^{(1)} \\ a_4^{(1)} \end{matrix}$

E' importante sottolineare come queste reti automaticamente imparano la migliore rappresentazione interna dei dati, permettendo di non creare "manualmente" altri algoritmi interni, ottenendo direttamente una soluzione del problema. Questo processo di apprendimento è chiamato end-to-end learning e permette di ottenere sistemi di dimensione ridotta e con risultati migliori (Bojarski, Del Testa e Dworakowski, 2016): risultati migliori perché non essendoci algoritmi interni creati "manualmente" (sottoposti a criteri e vincoli creati dagli ingegneri) viene massimizzata direttamente la performance generale del sistema, dimensione ridotta perché il sistema impara a risolvere il problema con il minor numero possibile di passaggi.

Per capire meglio il processo d'apprendimento di una rete neurale, possiamo pensare di utilizzare varie features per cercare di predire il prezzo di un'azione (ouput). Per esempio, possiamo utilizzare i dividendi attesi e il tasso di sconto (figura 5.4). La costruzione di una rete neurale è simile a quella di un edificio: prendiamo i singoli neuroni e li mettiamo insieme per ottenere una struttura più complessa. Date le varie

features (dividendi e tasso di sconto), possiamo decidere che i dividendi sono una funzione del payout ratio e degli earning, mentre il tasso di sconto è funzione del tasso d'inflazione. Attraverso l'iterazione di questo procedimento possiamo aggiungere sempre più features e layer e ottenere modelli sempre più complessi.

Figura 5.4. Diagramma di una piccola rete neyrale per predire il prezzo di un'azione



5.4 Funzione di Costo

In confronto alla funzione costo dei modelli precedentemente spiegati viene aggiunto un ulteriore operatore di somma ($\sum_{k=1}^{K}$) che risponda dei multipli nodi di output, mentre nella parte della regolarizzazione, dopo le parentesi quadre, sono presenti delle matrici multiple di theta.

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{k=1}^{K} = [y_k^{(i)}\log(\left(h_\theta\left(x^{(i)}\right)_k\right) + \left(1 - y_k^{(i)}\right)\log\left(1 - h_\theta(x^{(i)})\right)_k] + \frac{\lambda}{2m}\sum_{l=1}^{L-1}\sum_{i=1}^{sl}\sum_{j=1}^{sl+1}(\theta_{j,i}^{(l)})^2$$

$h_\theta(x)_k$è la funzione d'ipotesi che risulta nel k^{th} output
$h_\theta(x)\epsilon R^k$
L è il numero di layers nel network
K è il numero di output
s_l è il numero di unità nel layer l

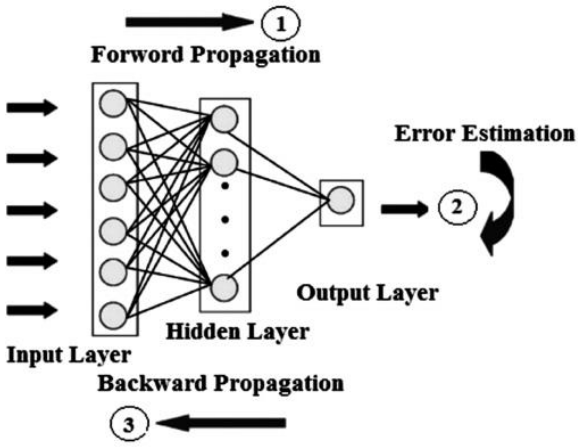
5.5 Backpropagation

Per backpropagation (propagazione all'indietro degli errori) intendiamo un tecnica molto utilizzata per calcolare il gradiente della funzione costo usata in combinazione con un metodo d'ottimizzazione, come ad esempio la discesa del gradiente, per minimizzare la funzione stessa. Nel processo d'apprendimento la backpropagation è usata dall'algoritmo d'ottimizzazione (discesa del gradiente) per aggiustare i pesi dei neuroni calcolando il gradiente della funzione costo. Il gradiente indica, per ogni peso, la direzione che genera il cambiamento più rapido nel valore della funzione costo, in modo tale da minimizzarla e quali pesi contribuiscono maggiormente al cambiamento. In

generale, nella Backpropagation (figura 5.5) l'errore viene propagato all'indietro, dal layer output verso il layer input, per permette di ottenere calcoli più semplici, veloci e meno pesanti per la memoria del computer.
Espresso in termini matematici, vogliamo $\min_{\theta} J(\theta)$ utilizzando l'algoritmo backpropagation:

Dato un training set $\{(x^{(1)}, y^{(1)}), \ldots\ldots, (x^{(m)}, y^{(m)})\}$
$\Delta_{ij}^{(l)} = 0\ (per\ l, i, j)$
Per gli esempi i=1 a m
Fissato $a^{(1)} = x^{(i)}$
Calcolo la forward propagation per ottenere $a^{(l)}$ per l=2,3,….,L
Usando $y^{(i)}$ calcolo l'errore per $a^{(L)}$cosi $\delta^{(L)} = a^{(L)} - y^{(i)}$ dove $a^{(L)}$ è il vettore di outputs nella unità d'attivazione per l'ultimo layer. Più formalmente $\delta^{(L)} = \frac{\partial}{\partial z_j^{(l)}} cost(t)$ è la derivata della funzione costo
Calcolo $\delta^{(L-1)}$, $\delta^{(L-2)}$……,$\delta^{(2)}$ usando $\delta^{(l)} = ((\theta^{(l)})^T \theta^{(l+1)}) g'(z^{(l)})$
$\Delta_{ij}^{(l)} := \Delta_{ij}^{(l)} + a_j^{(l)} \delta_i^{(l+1)}$
$\frac{\partial}{\partial \theta_{ij}^{(l)}} J(\theta) := \frac{1}{m} \Delta_{ij}^{(l)} + \lambda \theta_{ij}^{(l)}\ se\ J$ è $diverso\ da\ 0$
$\frac{\partial}{\partial \theta_{ij}^{(l)}} J(\theta) := \frac{1}{m} \Delta_{ij}^{(l)}\ se\ j$ è $uguale\ a\ 0\ (ovvero\ per\ il\ termine\ bias)$

Figura 5.5. Backpropagation



Fonte: Abedini et al. (2012).

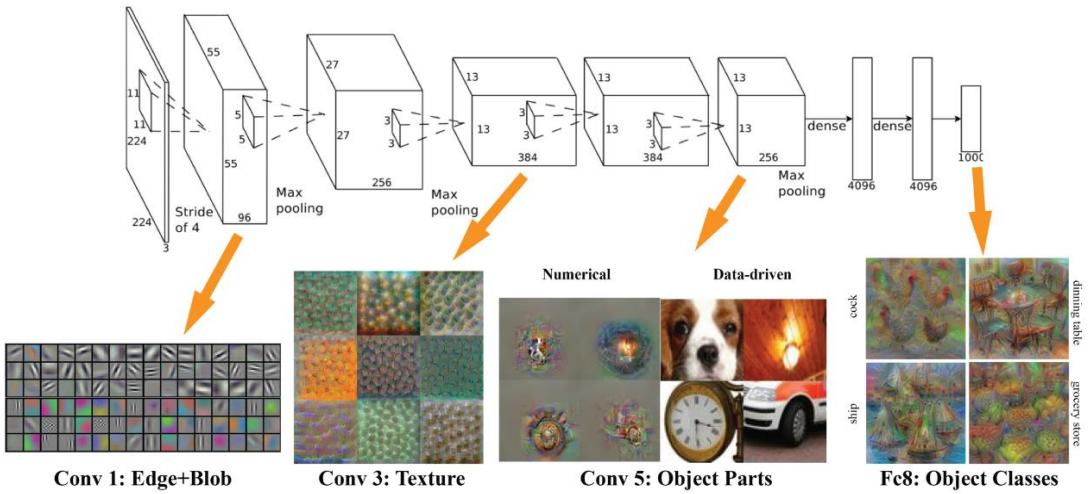
Riassumendo i passaggi chiave da compiere:
1. scegliere una architettura che possa raggiungere l'obiettivo nel migliore dei modi, ovvero: scegliere quanti layers usare, quante unità input (dimensione delle features), quante unità output (numero di classi), quante unità nascoste.
2. allenare la rete a raggiungere l'obiettivo: inizializzare casualmente i pesi, implementare la forward propagation per ottenere la funzione $(h_\theta(x^{(i)})$ per ogni $x^{(i)}$,

implementare la funzione costo, implementare la backpropagation per calcolare le derivate parziali, utilizzare la discesa del gradiente per minimizzare la funzione costo.

5.6 Convolutional Neural Networks (CNNs/ConvNets)

Le Convolutional Neural Networks (LeCun, 1989) sono una particolare rete neurale che usa la convoluzione al posto della moltiplicazione matriciale in almeno uno dei suoi layers (Goodfellow e Bengio, 2016). L'architettura classica, composta almeno da tre layer -Convolutional (CONV), Pooling (POOL), Fully-connected-, è progettata per gestire immagini ad alta dimensionalità, ed è molto usata nel riconoscimento (riconoscimento di cifre manoscritte, riconoscimento facciale, ecc.).
Le CNNs risolvono il problema principale delle reti tradizionali, l'elevato costo computazionale derivante dall'estrazione delle features, riducendo le connessioni tra i neuroni e connettendo soltanto una regione locale delle unità input alle unità hidden.
Utilizzando dati ad alta dimensionalità, infatti, risulterebbe controproducente connettere tutti i neuroni, basti pensare che per un immagine 32x32x3 bisogna calcolare per un neurone nel primo hidden layer 3072 pesi.
Diversamente dalle reti neurale precedentemente spiegate, i neuroni hanno tre dimensioni: lunghezza, altezza, profondità (la dimensione del volume è data dal prodotto delle tre misure).
In generale, un immagine con milioni di pixels viene processata attraverso una sequenza di layers per ottenere un vettore, arrangiato lungo la dimensione profondità, con i valori delle classi cercate. L'immagine viene scomposta nelle features importanti, le quali vengono assemblate per individuare i soggetti all'interno dell'immagine (figura 5.6).

Figura 5.6. CNN



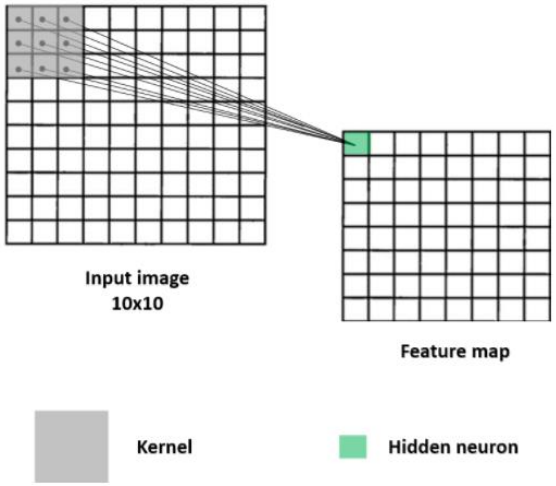
Fonte: MIT's computer vision course

Il cuore della rete è il convolutional layer, il cui obiettivo principale è l'estrazione delle features importanti dall'immagine input. Grazie alla convoluzione è possibile ridurre il

numero di connessioni e mantenere una struttura spaziale che evidenza le relazioni tra i pixel.
Il CONV può essere rappresentato come uno strato neurale, che a differenza dei modelli tradizionali mantiene la stessa struttura "quadrata" (non viene ridotta in forma vettoriale), in cui vengono utilizzate molte "copie" di uno stesso neurone, permettendo di esprimere modelli computazionalmente complessi utilizzando un numero di parametri da apprendere basso. Una CNNs, infatti, può imparare i parametri di un neurone ed utilizzarli in diverse occasioni (parameter sharing), riducendo il numero complessivo. Questo è possibile perché le immagini hanno la proprietà di essere "stazionarie", un feature utile in una posizione molto probabilmente lo è anche in un'altra.
La condivisione di parametri, inoltre, permette alla rete di utilizzare un proprietà importante, soprattutto nel processare serie temporali, chiamata equivariance, in cui la variazione di un oggetto negli input genera una variazione dello stesso ammontare nell'output.
In generale, il meccanismo del CONV è il seguente: applichiamo un filtro/piu filtri all'immagine input per estrarre la feature/le fetures desiderata/e, ottenendo cosi uno/più hidden layer (feature map o activation map) (figura 5.7). Siccome sia il filtro che l'immagine input sono due matrici, è possibile spostare il filtro, per il lungo e largo, sull'immagine e calcolare la mappa d'attivazione come il prodotto tra le entrate del filtro e l'input in ogni posizione.
Ogni filtro ha una dimensione ridotta rispetto alla lunghezza e larghezza dell'immagine iniziale (ma stessa profondità) e attraverso lo spostamento sul volume dell'input produce una mappa d'attivazione a due dimensioni. La mappa d'attivazione viene poi sistemata lungo la dimensione "profondità" e produce l'output.
La convoluzione di altri filtri (cambiando i valori della matrice) sulla stessa immagine genera più mappe d'attivazione. Più filtri utilizziamo (figura 5.8), più features estraiamo e più la nostra rete impara a riconoscere patterns.
I neuroni del primo hidden layer, ottenuti filtrando l'immagine originale, sono cosi connessi soltanto a una regione dell'immagine input. Ogni neurone è il risultato della convoluzione tra una matrice dei pesi, chiamata filtro (kernel), e una regione della stessa dimensione dell'immagine input (receptive field).
Durante la fase dell'allenamento la rete impara i valori dei filtri. Tre iperparametri controllano la dimensione del volume dell'output: profondità, stride, zero-padding. La profondità del volume di output corrisponde al numero di filtri che vogliamo usare, ognuno dei quali impara una caratteristica. Lo stride indica di quanti pixel il filtro si sposta sull'immagine. Lo zero-padding, invece, è utilizzato per riempire il volume d'input con degli zero per assicurare che il volume input e output abbiano la stessa dimensione.
E' possibile calcolare la dimensione del volume di output come una funzione del volume di input (W), la dimensione del filtro (F), lo stride (S) e lo zero-padding (P): $\frac{W-F+2P}{S+1}$.
Per esempio, un input 7x7 e un filtro 3x3 con stride 1 e pad 0 generano un output 5x5, in cui ogni neurone nel CONV ha 5x5=25 pesi.
Formalmente, se definiamo la k-esima mappa d'attivazione in un preciso layer h^k e assumiamo che i filtri siano determinati da W^k e b_ke che la non linearità è rappresentata dalla funzione tanh, allora:

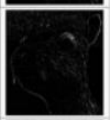
$$h_{ij}^k = tanh((W^k x)_{ij} + b_k)$$

Figura 5.7. Applicazione di un filtro (Kernel)



Fonte: www.kdnuggets.com

Figura 5.8. Vari filtri

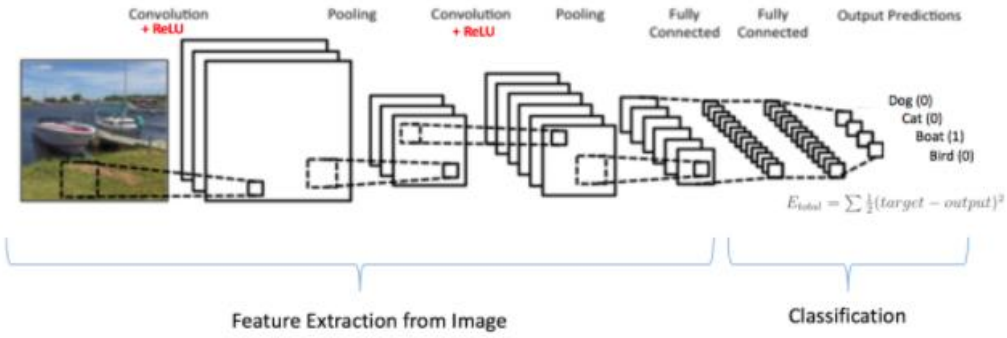
Operation	Filter	Convolved Image
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9}\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Gaussian blur (approximation)	$\frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	

Fonte: ujjwalkarn.me

In sintesi:
-il convolutional layer accetta un volume di dimensione $W_1 x H_1 x D_1$;

-utilizza quattro parametri, il numero di filtri K, lo stride S, l'ammontare di padding P, la dimensione del filtro F;
-produce un volume di dimensione $W_2 x H_2 x D_2$, dove $W_2 = \frac{W-F+2P}{S+1}$ $H_2 = \frac{H-F+2P}{S+1}$ $D_2 = K$);
-utilizza $FxFxD_1$ pesi per filtro, per un totale di $(FxFxD_1)xK$ pesi e K biases;
-nel volume d'output la profondità d-esima (di dimensione $W_2 x H_2$) è il risultato della convoluzione del d-esimo filtro sul volume input con uno stride S.
E' solito utilizzare all'interno di una CNNs un POOL per ridurre la dimensione spaziale (larghezza e altezza) di ogni funzione d'attivazione senza perdere l'informazione importante. Questo permette di ridurre l'ammontare dei parametri e controllare il problema dell'overfitting.
La forma più utilizzata è il Max Pooling il quale riporta l'output massimo all'interno di una zona delimitata. Applicando un filtro di dimensione 2x2 con stride di 2 è possibile sottocampionare ogni profondità in input di 2, lungo l'altezza e la larghezza, diminuendo del 75% le attivazioni.
Il pooling layer opera su ogni profondità del volume d'input, ridimensionandolo spazialmente ma lasciando inalterata la dimensione profondità. Il pooling, inoltre, aiuta a rendere la rappresentazione invariante a piccole variazioni degli input, una proprietà importante quando si è interessati a dove la feature è presente piuttosto di dove esattamente è localizzata. Per esempio, quando una rete determina se una immagine contiene una faccia, non deve sapere esattamente in quale pixel è posizionato l'occhio, ma deve riconoscere l'esistenza dell'occhio in una determinata zona della faccia.
In sintesi:
- il pooling layer accetta un volume di dimensione $W_1 x H_1 x D_1$;
-utilizza due iperparametri (F e S);
-produce un volume di dimensione $W_2 x H_2 x D_2$ dove $W_2 = \frac{W-F}{S+1}$ $H_2 = \frac{H-F}{S+1}$ $D_2 = D_1$;
Infine, è spesso utilizzato un layer classico FC , nel quale i neuroni sono tutti connessi, con l'obiettivo di utilizzare le features estratte nei layer precedenti per classificare l'immagine in classi e imparare combinazioni non lineari delle features.
E' importante sottolineare come la sola differenza tra i layer FC e CONV è data dalla quantità di connessioni, quindi è possibile convertire un layer in un altro.
Per concludere, immaginiamo di voler individuare una barca all'interno di una immagine (figura 5.9). Processando nella rete un insieme di foto è possibile insegnare all'algoritmo a classificare correttamente una barca. Come in una rete tradizionale, le immagini del training set date in input vengono trasformate in probabilità-output. Il vettore target è [0, 0, 1, 0].
Processando il primo esempio, a causa dell'inizializzazione casuale dei pesi, la rete produce delle probabilità-output errate. Misurando l'errore e utilizzando la backpropagation per calcolare il gradienti degli errori rispetto i pesi, i parametri vengono aggiornati, in proporzione al contributo marginale all'errore totale, e la funzione di perdita minimizzata. E' importante notare che se la stessa immagine venisse riutilizzata dopo l'aggiornato dei pesi, si otterrebbe un valore output più vicino al vettore target. Ripetendo lo stesso processo per tutte le immagini del training set, i pesi e i parametri della rete vengono ottimizzati per il riconoscimento della barca. Durante l'allenamento, i pesi della matrice filtro vengono aggiornati mentre altri parametri (come il numero di filtri, la dimensione dei filtri, l'architettura delle reti) rimangono invariati.

Figura 5.9. Applicazione di una CNN per il riconoscimento di una barca all'interno di una immagine



Fonte: www.kdnuggets.com

5.7 Recurrent Neural Networks (RNNs)

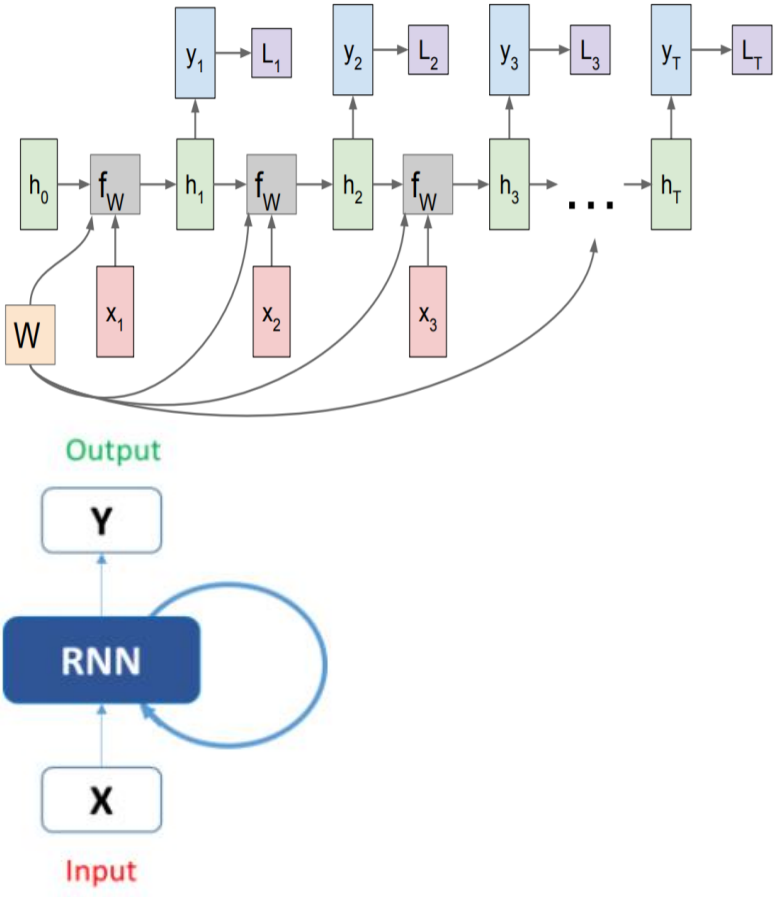
Una rete neurale può predire il prezzo di un titolo utilizzando una serie di features, tipo il volume del giorno o il prezzo d'apertura, ma parte del prezzo è spiegato dai valori dei giorni precedenti, dal trend in cui si trova. E' utili, quindi, considerare le predizioni passate e l'informazione imparata.
Lo stesso approccio è sviluppato dal nostro cervello, infatti, quando leggiamo un testo, capiamo il senso di ogni parola basandoci su quelle precedenti. I nostri pensieri hanno persistenza nel tempo, un elemento mancante nelle reti neurali tradizionali fin qui descritte che assumevano l'indipendenza tra inputs e outputs.
Le RNNs (Rumelhart et al., 1986) risolvono questo problema, sono una famiglia di reti neurali specializzata nel processare dati sequenziali e che utilizza loops per permette all'informazione di persistere nel tempo (figura 5.10).
A differenza delle reti neurali tradizionali, in cui ogni layer è caratterizzato da propri pesi e da un comportamento indipendente da quello del layer precedente/successivo, queste reti condividono gli stessi pesi durante la forward propagation e hanno memoria di ciò che è stato calcolato precedentemente, ad ogni intervallo temporale (o ad ogni posizione della sequenza) lo stato memorizza quello precedente e lo combina con l'informazione nuova.
Nonostante siano state largamente utilizzate nel riconoscimento vocale, traduzione automatica, modellazione del liguaggio/generazione di testo, le RNNs hanno fallito nel diventare uno strumento fondamentale per gli specialisti del ML a causa della difficoltà riscontrate nell'allenamento. La causa principale è l'instabilità del gradiente, il quale, su lunghi intervalli temporali, tende a esplodere o a svanire.
L'architettura più semplice è chiamata Vanilla e permette di mappare una sequenza di vettori input a una sequenza output della stessa lunghezza applicando una formula ricorrente ad ogni intervallo di tempo:

$$h_t = f_W(h_{t-1}, x_t)$$

$h_t = tanh(W_{hh}h_{t-1} + W_{xh}x_t)$
$\widehat{y_t} = softmax(W_{hy}h_t)$
dove h_tè lo stato (è la ”memoria”, l’hidden layer o output features, o l’attivazione) del sistema al tempo t; x_t è il vettore input al tempo t; W_{hh}= peso Hidden to Hidden; W_{hy}= peso hidden to output; W_{xh}=peso input to hidden; $tanh$ è la funzione d’attivazione non lineare (tangente iperbolica);
$\widehat{y_t} = softmax(W_{hy}h_t)$è l’output predetto trasformato dall’operatore softmax in un vettore di probabilità normalizzate.

A ogni intervallo temporale, l’output dello step precedente h_{t-1} insieme al vettore x_t contenente l’informazione nuova diventano gli inputs del nuovo hidden layer, utilizzati per produrre un nuovo stato h_t (output features) e una predizione $\widehat{y_t}$.

Figura 5.10. RNN (unfolded graph vs circuit diagram)



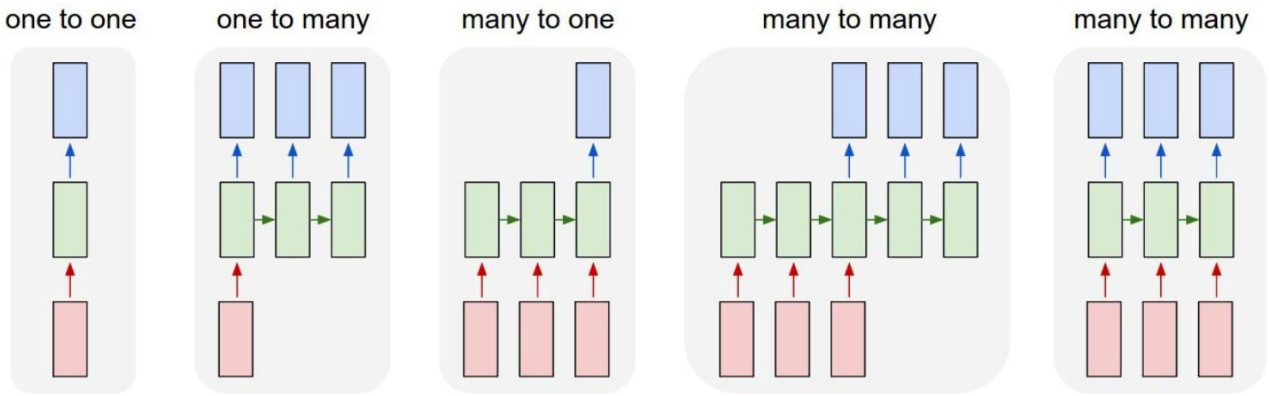
Fonte: Fei-Fei, Johnson e Yeung (2017b).

Esistono altre architetture famose (figura 5.11):

- la “one to one” utilizzata, per esempio, per la classificazione d’immagine;
- la “one to many” utilizzata per il riconoscimento d’immagine, ha come input una immagine e come output delle parole;

- la “many to one” utilizzata per la l’analisi del *sentiment*, ha come input delle parole e come output il *sentiment*;
- la “many to many” utilizzata nella traduzione automatica, ha come input delle parole in una lingua e come output delle parole in un'altra lingua;
- un ‘altra versione della “many to many” usata per la classificazione di video, ha come input il frame di un video e come output delle parole;

Figura 5.11. Architetture di RNNs



Fonte: Fei-Fei, Johnson e Yeung (2017b).

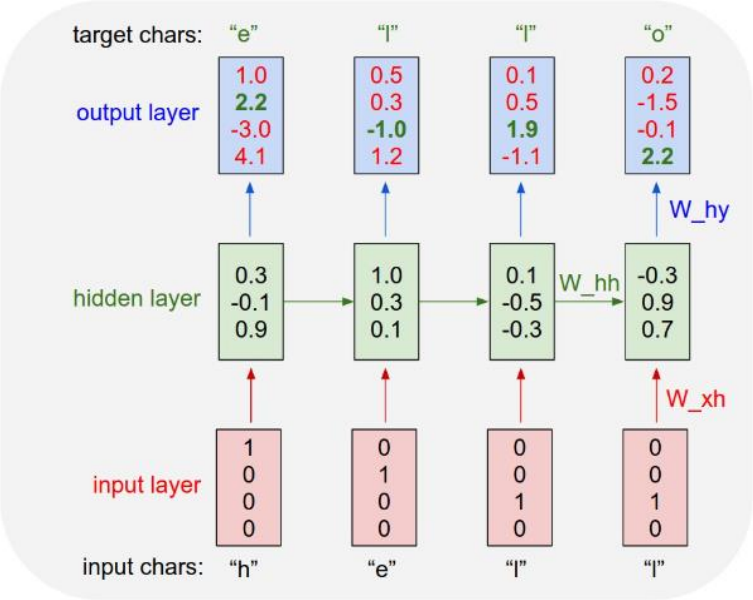
Come nelle reti neurali tradizionali viene utilizzata una funzione di perdita per misurare la differenza tra il valore predetto e il rispettivo target (l’errore). La funzione di perdita è la somma delle funzioni di perdita generate ad ogni intervallo di tempo. Se la funzione di perdita al tempo t è rappresentata dal logaritmo negativo della probabilità di y_t dato $x_1, \dots, x_t$, allora:

$$L = \sum_t L_t = -\sum_t log\, p_{modello}(y_t|(x_1, \dots, x_t))$$

L’errore generato viene “rimandato” all’inizio della rete e usato dalla discesa del gradiente per aggiustare i pesi. La funzione di perdita trasforma l’output attraverso l’operatore di softmax (assumiamo che l’output venga descritto dai logaritmi di probabilità non normalizzate) e compara il risultato con il rispettivo target.
Calcolare il gradiente della funzione di perdita è un operazione costosa a livello computazionale e richiede molto tempo. Durante la backpropagation (back-propagation through time BPTT) l’errore generato in un intervallo temporale può imporre il cambiamento a un altro errore distante molte intervalli, quindi, per semplificare il calcolo, soprattutto quando vengono utilizzate sequenze molto lunghe, la backpropagation può venire troncata (Truncated BPTT) nel tempo.
Le RNNs hanno alcuni vantaggi: usano la stessa dimensione di input a prescindere dalla lunghezza della sequenza, permettono di usare lunghezze (one to many, many to many ecc.) e sequenze diverse (per esempio y è una sequenza mentre x no) tra x e y, utilizzano

la stessa funzione con gli stessi parametri W ad ogni intervallo temporale. Questi vantaggi permettono di imparare un unico modello f che opera indifferentemente in vari setting, in qualunque intervallo temporale e con qualunque lunghezza (della sequenza), senza la necessità di usare modelli separati.
Utilizzare un unico modello condiviso permette di migliorare la generalizzazione, di utilizzare meno dati durante l'allenamento e di non dover imparare nuovi parametri ogni volta che si presenta un semplice variazioni delle features.
Un esempio (Fei-Fei, Johnson e Yeung, 2017) (figura 5.12) può semplificare la comprensione del modello: supponiamo di avere un vocabolario (training set) di quattro lettere (h,e,l,o) e di volere allenare una rete a comporre la parola/sequenza "Hello". Per ogni lettera, chiediamo al modello la distribuzione di probabilità della lettera successiva, data la sequenza delle lettere precedenti.

Figura 5.12. Un applicazione di una RNN



Fonte: Fei-Fei (2017b).

Per allenare la rete, a ogni intervallo temporale inseriamo un vettore input (la lettera target desiderata) e osserviamo la risposta data, la quale può essere interpretata come la probabilità prevista dalla rete per la lettera successiva della sequenza. E' possibile vedere in figura 5.12 come la rete, dopo aver ricevuto la lettera "h", assegna una probabilità di 2,2 alla lettera corretta "e", mentre 4.1 alla lettera sbagliata "o". Attraverso l'allenamento aggiustiamo i pesi fino a quando la rete non predice le probabilità giuste. I passaggi sono gli stessi utilizzati con una rete tradizionale. Attraverso l'algoritmo di backpropagation capiamo in quale direzione (gradiente) bisogna aggiustare i pesi della rete per migliorare le probabilità predette e, ripetendo l'allenamento molte volte, miglioriamo lo score fino a quando le predizioni sono consistenti con la lettera target. Durante l'allenamento i pesi vengono rispettivamente

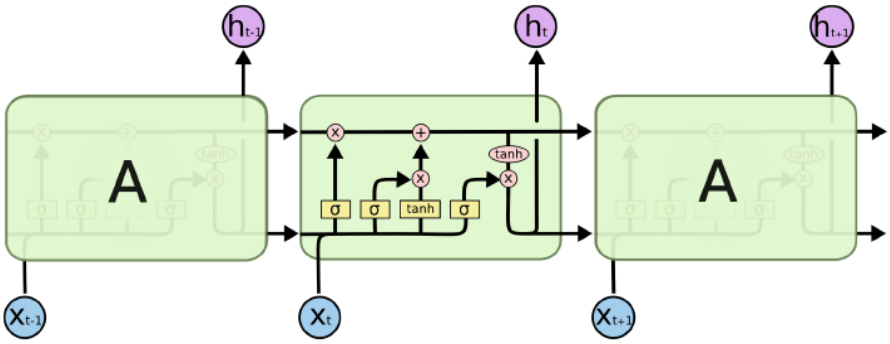
condivisi fino all'aggiornamento, durante il test invece, siccome i parametri sono già stati definiti, rimangono fissi.

5.8 Long Short Term Memory (LSTMs)

Le LSTMs sono una tipologia di RNN capace di imparare dipendenze di lungo periodo. Introdotte nel 1997 (Hochreiter e Schmidhuber), sono oggi usate moltissimo perché permettono di essere applicate in molteplici contesti, dalla previsione di serie temporali al riconoscimento vocale, dalla traduzione automatica al riconoscimento di cifre manoscritte.
Queste reti sono state progettate per superare i problemi generati dal gradiente durante l'allenamento, l'esplosione e lo svanimento del gradiente. Hochreiter (1991) e Bengio (1994) evidenziano come il gradiente declina esponenzialmente quando viene propagato nel tempo per molti stadi, non permettendo di imparare relazioni di lungo periodo. In aggiunta, essendo i layers e gli intervalli temporali collegati tra di loro attraverso moltiplicazioni, perché le RNN coinvolgono la composizione della stessa funzione nel tempo, è possibile che il gradiente esploda rendendo l'apprendimento instabile.
Le LSTMs, aggiungendo unità che memorizzano, imparano relazioni di lungo periodo e gestiscono l'informazione in maniera efficiente, selezionando i patterns più importanti e cancellando quelli inutili. Utilizzando l'esempio riportato in precedenza, il prezzo di un titolo, attraverso l'uso di queste reti, può essere predetto utilizzando il trend, il prezzo del titolo del giorno precedente e i fattori che influenzano il prezzo oggi. Le LSTMs sono molto adatte a predire serie temporali finanziarie perché imparano e ricordano relazioni di lungo periodo, come il regime di mercato, e allo stesso tempo relazioni di breve periodo.
Una rete RNN composta da unità LSTM è chiamata rete LSMT. Le unità sostituiscono gli hidden layer delle reti RNN tradizionali. Una unità è solitamente composta da una cella stato, una porta input, una porta output e una porta "dimenticata". La cella stato è il cuore delle LSTM: ha il compito di ricordare nel tempo i pattern importanti (ricordata il trend del titolo); ha gli stessi inputs x_t (fattori che influenzano il prezzo oggi) e h_{t-1}(prezzo del giorno precedente) e outputs h_t di una RNN ma utilizza più parametri e un sistema di porte per controllare il flusso dell'informazione.
Le porte invece sono dei neuroni convenzionali, come nei modelli tradizionali multi-layer, calcolano una attivazione e attraverso ciò regolano il flusso dell'informazione. L'informazione può essere immagazzinata, cancellata o letta e viene gestita dalle celle attraverso le porte. Le porte interagiscono con gli inpulsi ricevuti, similarmente ai nodi delle reti tradizionali, bloccandoli o facendoli passare in base alla loro importanza e filtrandoli con i propri pesi. I pesi vengono aggiustati nel processo d'allenamento in base all'obiettivo da raggiungere.
Le LSTMs, come le RNNs, sono strutturate a catena ma invece di avere all'interno di un modulo un singolo layer ne hanno quattro $(\sigma, \sigma, tanh, \sigma)$ per permettere all'informazione di essere gestita (figura 5.13).
L'idea principale è la seguente: la linea nera orizzontale superiore funge da "nastro trasportatore" e rappresenta la cella stato. L'informazione può essere aggiunta e rimossa attraverso l'uso delle porte. La prima porta (forget gate) è composto da un layer sigmoid (produce un output tra zero e uno) e un operatore prodotto punto per punto

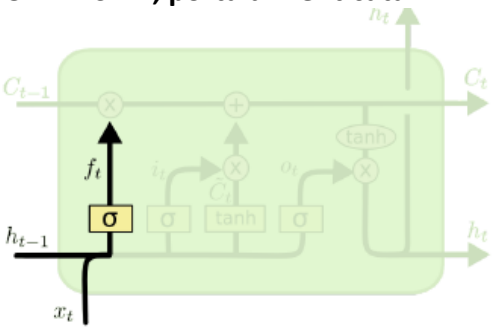
(pointwise). La sua funzione è decidere quanta dell'informazione passata è utile per lo stato attuale. Risponde alla domanda: c_{t-1} dovrebbe essere dimenticato? Viene prodotto un numero tra 0 e 1 (un valore di f_t di zero indica che l'informazione è stata bloccata, mentre un valore di uno che tutta l'informazione è stata utilizzata) per ogni numero presente nella cella stato c_{t-1}, utilizzando l'informazione di x_t *e* h_{t-1} (figura 5.14).

Figura 5.13. LSTM



Fonte: Olah (2015).

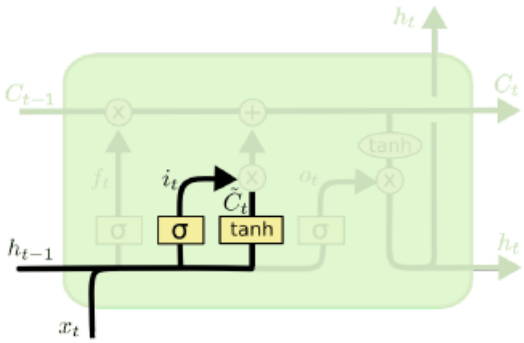
Figura 5.14. LSTM, porta dimenticata



$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] \; + \; b_f\right)$$

Fonte: Olah (2015).

Figura 5.15. LSTM, porta input

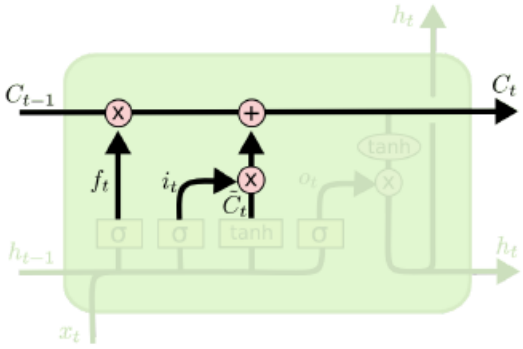


$$i_t = \sigma\left(W_i \cdot [h_{t-1}, x_t] \; + \; b_i\right)$$
$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] \; + \; b_C)$$

Fonte: Olah (2015).

Il secondo passaggio è decidere quanta nuova informazione immagazzinare nella cella stato attraverso l'uso di due layers, il primo sigmoid (input gate layer) e il secondo tanh (new memory generation). Il primo layer attraverso i_t utilizza l'informazione nuova x_te l'hidden state passata h_{t-1} per valutare se è utile preservare la nuova informazione x_t; il secondo, invece, crea un vettore di nuovi valori candidati $\tilde{c}_t$ (nuova memoria) utilizzando sempre x_t *e* h_{t-1} (figura 5.15).
Ora, è possibile prendere i consigli della porta "dimenticata" f_t e cancellare parte della memoria passata c_{t-1} e ,similarmente, prendere i consigli della porta input i_t e aggiungere la nuova memoria $\tilde{c}_t$. La somma di questi due valori produce la memoria finale c_t.
Matematicamente, moltiplichiamo il vecchio stato per la funzione f_t (cancellando l'informazione non utile) e aggiungiamo l'informazione nuova, il prodotto tra i_te $\tilde{c}_t$ (figura 5.16).

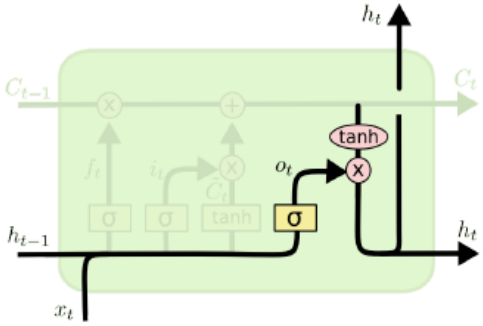
Figura 5.16. LSTM, cella stato



$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t$$

Fonte: Olah (2015).

Figura 5.17. LSTM, porta output



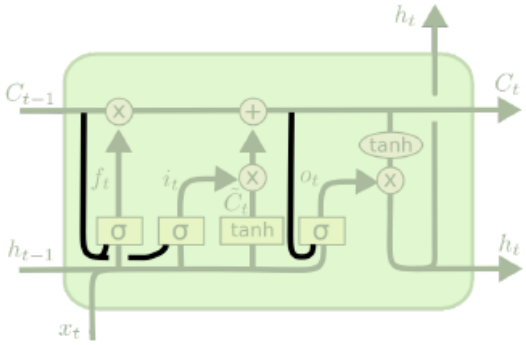
$$o_t = \sigma\left(W_o\left[h_{t-1}, x_t\right] + b_o\right)$$
$$h_t = o_t * \tanh\left(C_t\right)$$

Fonte: Olah (2015).

Infine, è possibile decidere l'output filtrando la cella stato, ovvero separando la memoria finale dall'hidden state. Questo viene fatto perché la memoria finale c_t contiene molta informazione che non necessita di essere salvata nel hidden state.
A livello matematico trasformiamo la cella stato attraverso la funzione tanh per ottenere un valore tra -1 e 1 e moltiplichiamo il valore per il sigmoid layer o_t, il quale funge da filtro (figura 5.17).

Esistono molte architetture diverse, una delle più popolari è quella introdotta da Gers e Schmidhuber nel 2000 che permette alle porte di utilizzare la cella stato attraverso le "Peephole Connections" (figura 5.18)

Figura 5.18. LSTM, Peephole Connections



$$f_t = \sigma\left(W_f \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \; + \; b_f\right)$$
$$i_t = \sigma\left(W_i \cdot [\boldsymbol{C_{t-1}}, h_{t-1}, x_t] \; + \; b_i\right)$$
$$o_t = \sigma\left(W_o \cdot [\boldsymbol{C_t}, h_{t-1}, x_t] \; + \; b_o\right)$$

Fonte: Olah (2015).

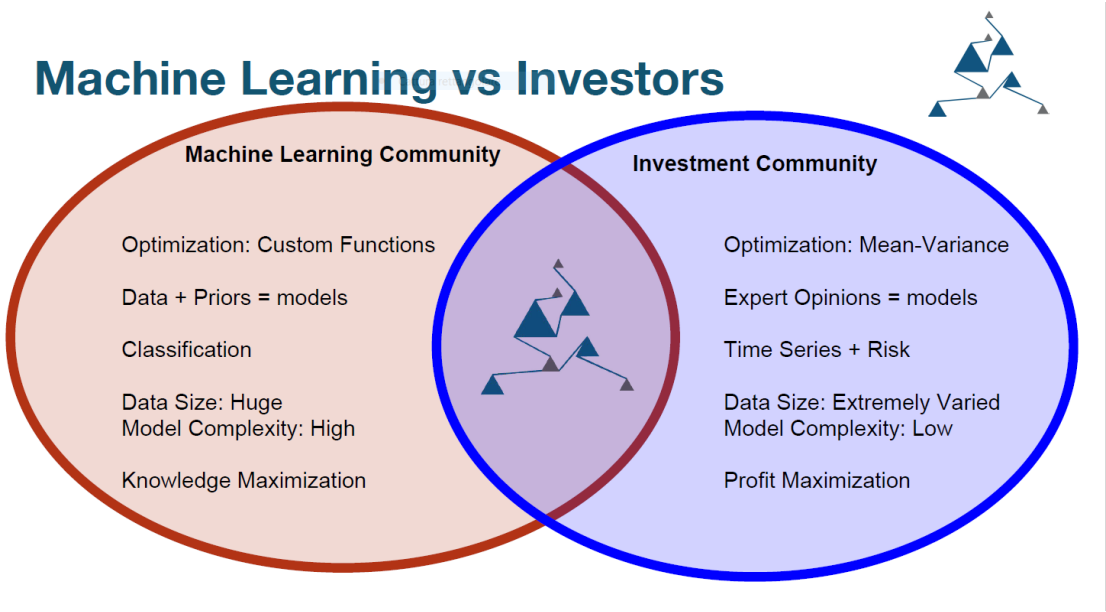
Altre possibili varianti molto famose sono Depth Gated RNNs (Yao *et al*., 2015) e Gated Recurrent Unit (Cho *et al*., 2014b)

CAPITOLO 6
Modern Portfolio Theory (MPT)

La teoria moderna di portafoglio (MPT) nasce nel 1952 da un articolo di Harry Markowitz, "Portfolio Selection", in cui viene descritto un modello matematico (chiamato di media-varianza) per costruire un portafoglio di assets massimizzando il ritorno atteso per un dato profilo di rischio. Fino ad allora, nessuno aveva mai proposto una teoria che analizzasse i portafogli con un approccio matematico/statistico. La teoria in quegli anni si concentra sull'analisi dei fondamentali dei titoli. Nel 1938 John Burr Williams scrive un volume chiamato "The Theory of investment value" che racchiude le conoscenze e il pensiero del tempo e sviluppa l'idea che il valore intrinseco di un titolo possa essere stimato dall'attualizzazione dei dividendi. Sempre in questi anni un professore della Columbia University, Benjamin Grahm, sviluppa la teoria del "Value Investing", descritta nelle sue due opere più note "Security Analysis" (1934) e "The intelligence investor" (1949). Secondo il Value Investing un investitore compra un titolo se ha un margine di sicurezza, ovvero se il prezzo di mercato è inferiore al prezzo intrinseco (stimato attraverso un'analisi dei fondamentali).
Ciò che manca fino al 1952, oltre a un'impostazione matematica del problema della selezione di un portafoglio, è una teoria sull'investimento che, analizzando la relazione tra rischio e rendimento in un portafoglio, distingua tra portafogli efficienti e inefficienti e descriva l'effetto della diversificazione quando il rischio è correlato. Il modello sviluppato da Markowitz però, non esprimendo nessun giudizio sull'efficienza dei mercati, permette agli investitori di possedere diversi portafogli in base alla propria propensione al rischio e di considerare efficienti portafogli diversi.
Nel 1964 Sharpe utilizzando l'impostazione del modello di Markowitz, sviluppa il Capital Asset Pricing Model (CAPM), un modello di selezione di portafoglio sotto l'assunzione di mercati efficienti e aspettative omogenee. In questo modello d'equilibrio tutti gli investitori possiedono lo stesso portafoglio efficiente, quello di mercato. Grazie a questa impostazione del problema, il premio al rischio di un titolo può essere spiegato in base alla esposizione del titolo al rischio sistematico Beta (rischio non diversificabile) e può essere confrontato con gli altri premi al rischio. Il CAPM permette, a differenza di quello di media-varianza, di essere impiegato come modello di valutazione e ancora oggi è molto utilizzato dai professionisti. Nonostante il valore di questa teoria sia stata riconosciuta fin dall'inizio dalla comunità finanziaria, in particolar modo l'idea di scomporre il rischio in sistematico (Beta) e idiosincratico, i ricercatori hanno trovato grosse discrepanze tra le assunzioni proposte e i dati empirici. I test empirici condotti sul CAPM non hanno portato ai risultati attesi a causa soprattutto delle numerose anomalie riscontrate nella verifica delle ipotesi di mercati efficienti (Basu, 1977; Banz, 1981).
Nel 1976 Ross sviluppa una teoria alternativa di asset pricing, l'Arbitrage Price Theory (APT), in cui i rendimenti degli assets sono spiegati da fattori di rischio come i tassi d'interesse, l'inflazione, la produzione industriale. APT spiega la relazione tra rischio e rendimento attraverso l'uso di modelli fattoriali e assume che non ci siano opportunità d'arbitraggio nei mercati finanziari. In seguito vengono proposti anche diversi modelli

multi-fattoriali, tra cui i più importanti sono quello di Chen, Roll e Ross (1986) e quello di Fama e French (1996).
Le numerose ricerche proposte negli anni, che avvalorano l'inefficienza dei mercati finanziari, portano a sviluppare una teoria di gestione di portafoglio attiva che sostiene la possibilità di ottenere extrarendimenti dallo sfruttamento delle inefficienze dei mercati. Il modello di Treyor e Black (1973) e quello di Black-Litterman (1990) sono i due più famosi.
Il primo modello consiglia di investire in parte in un portafoglio a gestione passiva (un portafoglio che riproduce un indice di mercato) e in parte in un portafoglio a gestione attiva (un portafoglio composto da titoli mispriced). Il modello di Black-Litterman, invece, attraverso un impostazione baeysiana del problema, permette di unire la visione dell'investitore sul mercato a dei rendimenti d'equilibrio (distribuzioni a priori), per ottenere una distribuzione di probabilità a posteriori più precisa e che tenga in considerazione l'informazione privata. Quest'ultimo modello, permettendo di ottenere portafogli più "robusti", è stato molto utilizzato dai professionisti del settore e ha visto lo sviluppo di numerose estensioni (Satchell e Scowcroft, 2000; Meucci, 2005; Krishnan e Mains, 2005; Cheung, 2009).
A distanza di quasi settant'anni dalla pubblicazione del paper "Portfolio Selection" molti principi sono stati rivisitati per tener conto delle nuove informazioni disponibili, delle nuove tecnologie e conoscenze, ma le basi di questa teoria si sono mantenute e sono diventate la prassi dell'asset management. La crisi del 2008 ha però messo in luce i limiti della MPT. I mercati si sono dimostrati molto più complessi e volatili di quanto previsto. Tre elementi hanno contraddistinto questa crisi: la paralisi del mercato del credito, il deleveraging diffuso e la crisi di liquidità. La MPT semplifica molto la realtà, attraverso le sue assunzioni, trascurando problemi importanti nella realtà come la liquidità, i costi di transazione e le fat-tails.
Le dinamiche presenti nei mercati finanziari sono ancora oggi in una prima fase di comprensione. I mercati sono luoghi difficili da decifrare, in cui convergono informazioni eterogenee provenienti da diverse fonti, con complesse interazioni. Sono sistemi adattivi, perennemente alterati dalle forze che agiscono al loro interno, e proprio per questo motivo, hanno regole che cambiano ogni giorno.
Ogni scienza empirica sviluppa la teoria basandosi sulle osservazioni ma, diversamente dalla fisica, in finanza, non potendo ripetere gli esperimenti in laboratori sotto condizioni controllate, vengono utilizzate assunzioni che spesso si discostano dalla realtà. Inoltre, se gli strumenti utilizzati per modellizzare le osservazioni sono quelli standard proposti dall'econometria classica, allora difficilmente i ricercatori potranno capire la complessità dei dati e creare teorie utili. Il settore finanziario, per progredire, ha la necessità di sviluppare teorie fondate sulla sperimentazione, e di integrare la propria conoscenza con quella scientifica (Prado, 2017*a*)
I nuovi metodi di analisi dei dati proposti dal ML non rimpiazzano la teoria finanziaria, la guidano. Come riporta riporta Lisa Goldberg, Execuive Director of analytic initiatives and talent alla MSCI Barra di New York, anche se i modelli di finanza tradizionali hanno fallito nel riprodurre la realtà, non bisogna rifiutarli, ma integrarli con le nuove tecnologie e con i nuovi dati disponibili: "This so-called failed theory has a lot of brilliant elements as well as material that needs revision or rethinking".
L'idea della Goldberg è ben rappresentata nella fig. 6.1 che riassume le caratteristiche più importanti nella costruzione di un modello nelle due comunità.

Figura 6.1. Machine Learning community vs Investment community



Fonte: Kruzel (2016).

6.1 Il modello di Markowitz (media-varianza)

Markowitz sostiene che, per ottenere portafogli con profili di rischio-rendimento migliori, bisogna studiare le relazioni tra gli assets e combinare assets caratterizzati da bassi livelli di correlazione. Questo importantissimo concetto, la diversificazione, permette all'investitore, attraverso la combinazione di titoli con rendimenti meno che perfettamente correlati, di ottenere una riduzione del rischio senza sacrificare il ritorno atteso del portafoglio.
Il modello presuppone che gli investitori siano avversi al rischio ovvero che, dati due portafogli con lo stesso rendimento atteso, viene sempre preferito il portafoglio meno rischioso (non verrà mai scelto un portafoglio dominato). Secondo questa assunzione, gli investitori assumeranno più rischio soltanto se compensato da un rendimento atteso maggiore.
Consideriamo un portafoglio con n assets in cui l'asset i ha un rendimento R_i (i rendimenti vengono considerati variabili casuali). w_i è l'ammontare investito nell'asset i, μ_i e σ_i^2 corrispondono al rendimento atteso e alla varianza dei rendimenti dell'asset i, mentre σ_{ij} alla covarianza tra i rendimenti dell'assets i e quello j.
Definiamo il rendimento atteso del portafoglio $E[R_p]$ e la varianza di portafoglio σ_p^2:

$$E[R_p] = \sum_{i=1}^{n} \mu_i w_i$$

$$\sigma_p^2 = Var[R] = \sum_{i=1}^{n}\sum_{j=1}^{n} \sigma_{ij} w_i w_j = \sum_{i=1}^{n} w_i^2 \sigma_i^2 + \sum_{i=1}^{n}\sum_{j=1\, j\neq i}^{n} w_i w_j \sigma_i \sigma_j \sigma_{jk}$$

$$\sum_{i=1}^{n} w_i = 1$$
$$w_i \geq 0, i = 1,2 \dots n$$

Un portafoglio composto da due assets rischiosi, per esempio bond e equity, può essere rappresentato cosi:

$$E(R_p) = w_a E(R)_a + w_b E(R)_b$$
$$\sigma_p^2 = w_a^2 \sigma_a^2 + w_b^2 \sigma_b^2 + 2 w_a w_b \sigma_a \sigma_b \sigma_{ab}$$
$$w_a + w_a = 1$$

Per più di due assets rischiosi possiamo utilizzare la forma matriciale che risulta essere più compatta:

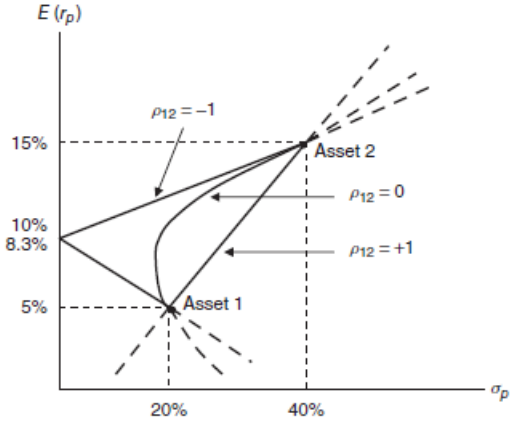
$$E(R_p) = \begin{cases} w'\mu \ \ (senza\ risk\ free) \\ w'\mu + (1 - w'e)\mu_f \end{cases}$$
$$\sigma_p^2 = w' \Sigma \mathrm{w}$$
$$w'e = \sum w_i = 1$$

Dove Σ rappresenta la matrice di varianza-covarianza dei rendimenti degli asset definita positiva.

6.2 Opportunity Set

L'opportunity set rappresenta l'insieme di tutti gli assets e dei portafogli che si possono creare. Graficamente può essere rappresentato in uno spazio in cui l'ascissa è la deviazione standard e l'ordinata è il rendimento atteso.

Figura 6.2. L'effetto della diversificazione con diversi coefficienti di correlazione



Fonte: Francis e Kim (2013), p. 122.

6.3 Diversificazione

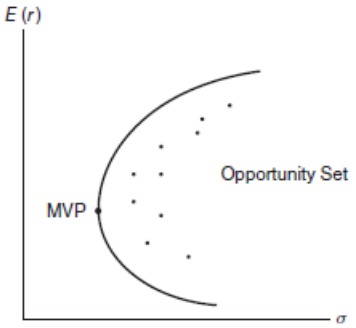
La diversificazione di un portafoglio di titoli (figura 6.2) consiste in una riduzione della rischiosità grazie alla combinazione di titoli i cui rendimenti non sono perfettamente correlati. A livello grafico, minore è la correlazione tra due assets, maggiore è l'estensione del grafico in zona nord-ovest, ovvero dove la preferenza dell'investitore è maggiore.

6.4 Frontiera efficiente e Portafoglio a Varianza Minima

Un portafoglio efficiente è un portafoglio che ha un rendimento atteso maggiore rispetto agli altri appartenenti alla sua stessa classe di rischio.
In formula:

$$E(r_p) > E(r_q) \ per\ ogni\ Q: \sigma_q = \sigma_p$$

Figura 6.3. L'opportunity set, la frontiera efficiente, il portafoglio a varianza minima



Fonte: Francis e Kim (2013), p. 119.

L'insieme dei portafogli efficienti composti da assets rischiosi definiscono la frontiera efficiente. E' importante notare come nel modello di Markowitz ogni singolo asset rischioso venga considerato inefficiente.
E' possibile individuare il portafoglio sulla frontiera efficiente a varianza minima con la seguente formula:

$$w_{mvp} = \min_{w} \frac{1}{2} w' \Sigma \mathrm{w} \ s.t. w'e = 1$$

Con soluzione:

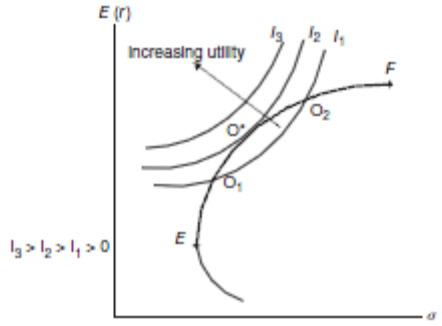
$$w_{mvp} = \frac{\Sigma^{-1} e}{e' \Sigma^{-1} e}$$

Tutti i portafogli al di sotto del portafoglio a varianza minima sono dominati, quindi la parte superiore è chiamata frontiera efficiente (figura 6.3)

6.5 Portafoglio Ottimo

Il portafoglio ottimo per un investitore avverso al rischio è il portafoglio di tangenza tra la frontiera efficiente e la curva di indifferenza con valore maggiore. La composizione del portafoglio dipende dall'avversione al rischio dell'investitore. Questo può essere provato facendo notare come le curve d'indifferenza di un investitore avverso al rischio raggiungono una utilità attesa maggiore spostandosi in alto a sinistra (figura 6.4); siccome l'investitore preferisce la curva d'indifferenza con valore maggiore, l'opportunity set è concavo e la frontiera efficiente è la parte più in alto a sinistra dell'opportunity set, allora il portafoglio ottimale sarà il portafoglio efficiente tangente alla curva d'indifferenza con utilità maggiore.

Figura 6.4. Il portafoglio ottimo



Fonte: Francis e Kim (2013), p. 119.

6.6 Introduzione di un risk free asset: Capital Allocation Line (CAL) e Capital Market Line (CML)

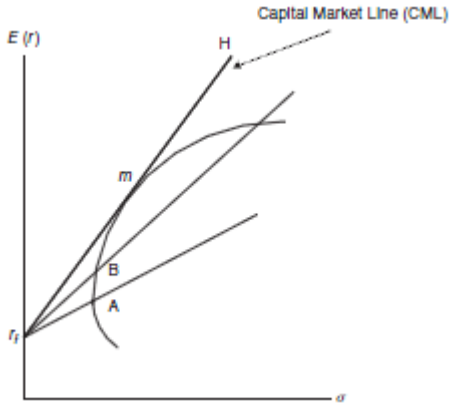
E' possibile introdurre la possibilità di prendere a prestito e prestare a un tasso risk-free. L'asset risk-free (r_f) viene considerato senza rischio, ovvero a varianza zero.
Un portafoglio composto da un asset rischioso e uno risk-free ha un rendimento atteso $E(r_P) = (1 - w)r_f + wE(r)$ e una varianza $\sigma_P^2 = w^2\sigma^2$.
Combinando un asset rischioso (o un portafoglio) con un asset risk-free otteniamo una retta chiamata Capital Allocation Line (CAL).
La CAL descrive una relazione lineare tra rischio e rendimento e può essere definita matematicamente cosi:

$$E(r_P) = r_f + (\frac{E(r) - r_f}{\sigma})\sigma_P$$

L'inclinazione della retta viene definita reward to risk (RR) o Sharpe Ratio (SR).

Esiste una CAL particolare, la Capital Market Line (CML), che rappresenta l'insieme delle combinazioni lineare tra il portafoglio ottimale di attività rischiose (chiamato anche portafoglio di tangenza, e rappresenta il portafoglio di attività rischiose sulla frontiera efficiente nel punto in cui la CML è tangente alla frontiera efficiente) e l'asset risk free. Tutti i portafogli appartenenti alla CML hanno un profilo di rischio-rendimento maggiore di quelli sulla frontiera efficiente, con eccezione del portafoglio di tangenza. Questa retta diventa la nuova frontiera efficiente e rappresenta la CAL con l'inclinazione più ripida (ovvero con lo SR maggiore).
Nel caso avessimo n assets rischiosi avremmo n CAL, ma soltanto una viene considerata efficiente, la CML (figura 6.5).

Figura 6.5. Capital Market Line



Fonte: Francis e Kim (2013), p. 129.

6.7 Selezione del Portafoglio Ottimo

Nel caso di due soli assets, uno rischioso e uno risk-free, un investitore sceglie la combinazione tra i due che massimizza l'utilità attesa.
In formula:
Max $U = \mu_p - \frac{1}{2}A\sigma_p^2$ s.t. $u_p = w\mu \quad + (1-w)r_f \; e \; \sigma_p^2 = w^2\sigma^2$
dove A identifica il coefficiente di avversione al rischio dell'investitore, definita in modo tale che la varianza sconta l'utilità a tassi maggiori per livelli di tolleranza al rischio minore.

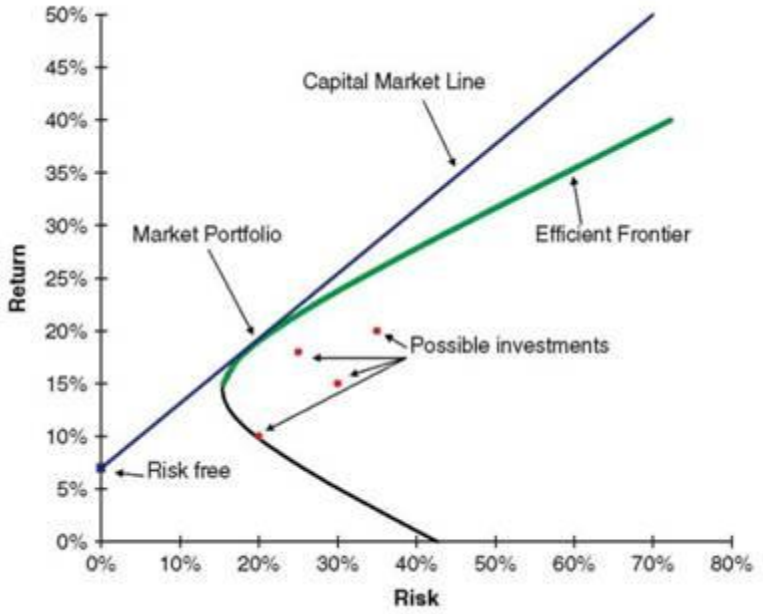
Risolvendo il problema di massimizzazione otteniamo la combinazione ottimale $w = \frac{\mu - r_f}{A\sigma^2}$
Nel caso di n assets rischiosi e uno risk free, il portafoglio ottimo per un investitore sarà sempre quello che massimizza la sua utilità attesa e risulterà quel portafoglio che si trova nel punto di tangenza tra la curva di utilità e la frontiera d'efficienza (CML).
Assumendo di poter prestare/prendere a prestito al tasso risk-free e di poter vendere allo scoperto, è possibile formalizzare il problema nel seguente modo:

$$\max_{w} U = r_f + w'(\mu - er_f) - \frac{A}{2} w'\Sigma w$$

Con soluzione:

$w = \frac{1}{A}\Sigma^{-1}(\mu\text{-}er_f)$ in cui 1-w'e è investito nell'asset risk-free

Figura 6.6: Il portafoglio ottimo includendo un'attività non rischiosa (risk free)



Si può ottenere lo stesso risultato mettendo in evidenza una proprietà importante, la separation property, che afferma l'indipendenza del portafoglio ottimo di attività rischiose dalla attitudine al rischio dell'investitore. Secondo questa proprietà, il problema della selezione del portafoglio può essere diviso in due passaggi: trovare il portafoglio ottimale di attività rischiose (uguale per tutti gli investitori) e combinare un asset risk-free con tale portafoglio, a seconda dell'attitudine al rischio, per massimizzare l'utilità attesa, trovando cosi il portafoglio ottimo completo. Di conseguenza è possibile affermare che tutti gli investitori ottengono lo stesso portafoglio di attività rischiose a prescindere dall'attitudine al rischio.

Il portafoglio ottimo rischioso può essere ottenuto massimizzando lo $\mathrm{SR_p} = \frac{\mathrm{E(R_p)-R_f}}{\sigma_p}$:

$$\max_{w} SR \quad s.t. \;\; w'e = 1$$

$$w = \frac{\Sigma^{-1}\mu}{e'\Sigma - 1\mu} \; con \; \mu = E(r) - e'r_f$$

La teoria di Markowitz può essere rappresentata graficamente come in figura 6.6.

6.8 Single-Index Model

La procedura introdotta da Markowitz richiede un numero molto elevato di stime da utilizzare nella matrice di varianza-covarianza. Per risolvere questo problema nel 1963 Sharpe, rivisitando il modello del 1959 di Markowitz, propone il single-index model, un modello fattoriale che utilizza un indice di mercato (di solito viene utilizzato lo S&P 500) come proxy del rischio sistematico. Semplificando il modo in cui il rischio è rappresentato, è possibile stimare e utilizzare un numero ridotto di input. La semplificazione è possibile perchè le covarianze tra i rendimenti degli assets sono affette da fattori comuni tipo i tassi di interesse o il ciclo economico. E' possibile quindi scomporre il tasso di rendimento di un titolo i in una componente di rendimento attesa più un rendimento derivante da un fattore macroeconomico non atteso (m) e un rendimento derivante da un fattore specifico non atteso (e). Questo modello, definito fattoriale, può essere descritto cosi:

$r_i = E(r_i) + \beta_i m + e_i$
$\sigma_i^2 = \beta_i \sigma_m^2 + \sigma^2(e_i)$
$cov(r_i, r_j) = \beta_i \beta_j \sigma_m^2$

m è distribuito con media zero e deviazione standard σ_m
m genera correlazioni tra i titoli mentre e_i no
m e e_i sono incorrelati
Beta rappresenta la sensibilità del titolo i ai cambiamenti macroeconomici del mercato.

Utilizzando il modello fattoriale sopra spiegato e considerando come rischio sistematico/macroeconomico un indice di mercato (S&P500), è possibile definire il Single-Index Model:

$R_i(t) = \alpha_i + \beta_i R_M(t) + e_i(t)$
dove $R_M = r_m - r_f$ e $R_i = r_i - r_f$

il Beta rappresenta la sensitività del rendimento del titolo i alle variazioni dell'indice di mercato
$\beta_i R_M$ rappresenta il rischio sistematico
$e_i(t)$ rappresenta il rischio specifico e alpha un nonmarket premium.

E' possibile stimare il modello attraverso una regressione, regredendo l'extra rendimento atteso di un titolo con l'extra rendimento atteso del portafoglio di mercato, utilizzando le serie storiche relative. Il rendimento è cosi spiegato da due componenti, una sistematica e l'altra idiosincratica.
La varianza e covarianza possono essere espresso nel seguente modo:

$\sigma_i^2 = \beta_i^2 \sigma_m^2 + \sigma^2(e_i)$
$cov(r_i, r_j) = \beta_i \beta_j \sigma_m^2$

Come nel modello di Markowitz, all'aumentare del numero di assets inclusi nel portafoglio il rischio idiosincratico diminuisce, mentre quello sistematico rimane invariato.
Il principale vantaggio di questo modello (Bodie e Kane, 2011), oltre alla diminuzione delle numero di input stimati, è il framework fornito per l'analisi dei titoli in preparazione della lista degli input. Le stime dei premi a rischio richieste dal modello di Markowitz dipendono dalle previsioni fornite dagli analisti sui fattori macroeconomici e sui titoli. Il single-index model, separando queste due fonti di rendimento, permette di ottenere risultati più consistenti: gli analisti specializzati in previsioni macro stimano il premio al rischio e il rischio dell'indice di mercato, mentre gli analisti specializzati nell'analisi dei singoli titoli stimano gli alpha.

6.9 Capital Asset Pricing Model (CAPM)

Nel 1964 William Sharpe, nell'articolo "Capital Asset Prices: a Theory of Market Equilibrium Under Conditions of Risk", espone un modello d'equilibrio dei mercati che spiega la relazione tra il rendimento atteso dell asset i-esimo e il rischio sistematico dell'asset β (Beta). Il CAPM afferma che un titolo ricompensa l'investitore per il valore del tempo (r_f) e per il rischio non diversificabile che si è assunto (definito premio al rischio e ottenuto moltiplicando la misura di rischio β per il premio al rischio benchmark $E(r_m)$-r_f). Secondo questo modello non tutto il rischio di un asset viene ricompensato dal mercato, ma soltanto quella parte che non può essere ridotta dalla diversificazione, perciò un asset con un alto rischio sistematico deve avere un rendimento atteso alto per indurre l'investitore ad assumere del rischio non diversificabile.
Il CAPM predice il rendimento atteso di assets rischiosi in condizioni di equilibrio sotto alcune assunzioni:

1- Gli investitori sono price-taking, le loro azioni non influenzano l'andamento generale del mercato;
2- Esiste un solo periodo di investimento;
3- Gli investitori sono razionali, avversi al rischio e usano il metodo di media varainza per prendere decisioni;
4- Tutti gli assets sono scambiati sul mercato;
5- Il mercato del capitale è perfetto, tutta l'informazione è libera e istantaneamente disponibile a tutti;
6- È possibile prendere a prestito e prestare al tasso risk-free;
7- Tutti gli assets sono infinitamente divisibili e in offerta fissa;
8- Tutti gli investitori hanno aspettative omogene.

Siccome tutti gli investitori ottengono la stessa CML e frontiera efficiente (per le assunzioni imposte), è possibile considerare il portafoglio di tangenza, definito di mercato, il portafoglio di attività rischiose efficiente detenuto da tutti gli investitori. Aggregando tutti portafogli degli investitori per ottenere il portafoglio di mercato, i capitali presi a prestito e prestati, grazie alla loro corrispondenza, vengono eliminati, e il valore del portafoglio ottenuto è uguale alla ricchezza dell'intera economia.
Il portafoglio di mercato contiene tutti gli assets rischiosi scambiati nella proporzione in cui sono offerti (valore di mercato) e ogni investitore possiede un portafoglio in proporzioni che replica quello di mercato. La non inclusione di un asset all'interno del

portafoglio di mercato, a causa della mancanza di domanda, genera un abbassamento del prezzo a un livello tale per cui gli investitori, ritenendo l'acquisto vantaggioso, includono l'asset nel portafoglio.
In questo contesto, ricordando che ogni investitore investe nel portafoglio rischioso $y_h = E(r_m - r_f)/A_h\sigma_m^2$, e che per quanto appena esposto è possibile approssimare y=1 e A=$\bar{A}$, è possibile calcolare il premio al rischio del portafoglio di mercato in questo modo:

$$E(r_M) - r_f = \bar{A}\sigma_M^2$$

Il premio al rischio domandato dagli investitori per investire nel portafoglio di mercato dipende dall'avversione media e dal rischio del portafoglio di mercato.
Siccome il rischio di un titolo all'interno di un portafoglio viene calcolato misurando l'apporto di rischio del titolo al portafoglio (covarianza tra rendimento del titolo e il rendimento del portafoglio), è possibile calcolare il RR di un titolo $E(r_i) - r_f/cov(r_i r_m)$.
Dato che il mercato è in equilibrio e che tutti gli investimenti devono offrire lo stesso RR, allora il RR del titolo i deve essere uguale a quello del mercato:

$$\frac{E(r_i) - r_f}{cov(r_i r_m)} = \frac{E(r_M) - r_f}{\sigma_M^2}$$

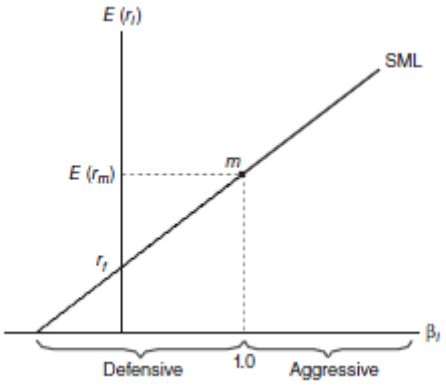
Da questa relazione è possibile ricavare il premio al rischio equo del titoli i:

$$E(r_i) - r_f = \beta_i(E(r_m) - r_f)$$

$\mathrm{E(r_i)}$ è il rendimento atteso dell'asset rischioso i
$\mathrm{r_f}$ è il rendimento dell'asset risk-free
$\mathrm{E(r_m)}$ è il rendimento atteso del portafoglio di mercato
$\mathrm{(E(r_m)\text{-}r_f)}$ rappresenta il premio al rischio ottenibile per aver investito in titoli rischiosi
$\beta_i = \frac{cov_{im}}{\sigma_m^2}$ è una misura del rischio sistematico (non diversificabile) e rappresenta il contributo dell'asset i alla varianza del portafoglio di mercato come frazione della varianza totale del portafoglio di mercato.
La pendenza della retta è misurata da $SR = E(r_m) - rf/\sigma_m^2$
Se la covarianza tra l'asset i e il portafoglio è positiva allora l'inserimento all'interno del portafoglio aumenterà il rischio e viceversa.
Siccome tutti gli investitori possiedono lo stesso portafoglio rischioso, allora tutti troveranno che la relazione tra il Beta di un asset con il portafoglio di mercato è uguale al Beta con il proprio portafoglio rischioso, di conseguenza tutti gli investitori saranno d'accordo sull'appropriato premio al rischio di un titolo.
Questo modello può essere rappresentato tracciando la security market line (SML) (figura 6.7): una retta che mette in relazione il premio al rischio di un asset singolo e il contributo dell'asset alla varianza di portafoglio (β_i).
In generale, dato il rischio di un investimento, la SML fornisce il tasso di rendimento necessario per compensare l'investitore del rischio e del valore del tempo. Secondo questa metrica di valutazione, se un azione fornisce un rendimento atteso maggiore del

ritorno equo proposto dal CAPM, allora il titolo è sottovalutato e viceversa. La differenza tra il valore equo e il ritorno atteso viene definito alpha.

Figura 6.7. CAPM, Security Market Line



Fonte: Francis e Kim (2013), p. 298.

E' importante notare come, a differenza della CML, che mette in relazione il premio al rischio di un portafoglio efficiente (composti da una combinazione tra un asset risk-free e uno rischioso a seconda della propensione al rischio) con la deviazione standard, la SML mette in relazione il premio al rischio di un singolo asset con il rischio dell'asset, inteso come contributo alla varianza di portafoglio (Beta).
Nonostante il CAPM abbia fallito i test empirici, è un modello largamente accettato e utilizzato. La SML viene spesso utilizzata come benchmark per la valutazione delle performance degli investimenti e spesso per prendere decisioni di capital budgeting.

6.10 Modelli Fattoriali

E' possibile e vantaggioso considerare diverse fonti di rischio invece di utilizzarne una unica (rischio sistematico di mercato) come nel CAPM. I modelli fattoriali sono strumenti (non è una teoria) che ci permettono di quantificare le "fonti" di rischio che influenzano il rendimento di un titolo e forniscono un framework utile per pianificare strategie di copertura.
Il modello più semplice, che considera un fattore di rischio, può essere rappresentato cosi:

$$r_i = E(r_i) + \beta_i F + e_i$$

F è la deviazione del fattore macroeconomico dal suo valore atteso (sorpresa)
β_i è la sensitività del titoli i al fattore
e_i incertezza derivante dal rischio specifico, assunto incorrelato con il fattore F e incorrelato con gli altri rischi specifici.

Il modello afferma che il ritorno dell'asset i è uguale al valore che ci si attende ($E(r_i) = rf + \beta_i RP$) più un ammontare casuale (con valore atteso uguale a zero perché misura la nuova informazione) attribuibile a degli eventi sistemici non anticipati, più un ammontare casuale (con valore atteso uguale a zero) attribuibile al rischio specifico.
Il rendimento e il rischio di un portafoglio sono cosi definiti:

$$\sigma^2(r_p) = \beta_p^2 \sigma^2(F) + \sum_{i=1}^{n} w_i^2 \sigma(e_i^2)$$

$$r_p = E(r_p) + \beta_p F + \sum_{i=1}^{n} w_i e_i$$

6.11 Modelli Multi Fattoriali

Confinare il rischio sistematico/macroeconomico in una sola variabile, come il rendimento di un indice di mercato, può risultare riduttivo, è quindi possibile dividere il rischio in diversi fattori per ottenere una descrizione migliore del rendimento di un titolo. I modelli multifattoriali danno la possibilità di misurare l'esposizione ai vari fattori macroeconomici e costruire un portafoglio che copra i rischi (hedging).
Nei modelli multi fattoriali il premio al rischio è determinato dalla somma delle esposizioni dei fattori di rischio, mentre il rendimento di un portafoglio e la varianza cosi:

$$r_p = E(r_p) + \sum_{j=1}^{k} \beta_{ij} F_j + \sum_{i=1}^{n} w_i e_i$$

$$w'\Sigma \mathrm{w} = \mathrm{w}'\mathrm{B}\Omega\mathrm{B}'\mathrm{w} + \mathrm{w}'\Sigma_e w$$

Dove B=[β_{ij} e $\Sigma_e = \mathrm{diag}[\sigma_i^2]$ per i=1,2,…,n assets e j=1,2,…,k fattori
I fattori possono essere cross correlati ma non correlati con e_i.

6.12 No Arbitrage Condition

E' possibile che nei mercati finanziari si creino opportunità di profitto derivanti dal mispricing dei titoli e che, attraverso operazioni di arbitraggio (acquisto e vendita simultanea dello stesso titolo per ottenere un profitto senza rischio dalla discrepanza tra i prezzi), alcuni investitori ottengano profitti privi di rischio. La condizione di non arbitraggio (no arbitrage condition) deriva dall'intervento dei arbitraggisti presenti sul mercato, i quali assorbendo l'opportunità di profitto in brevissimo tempo, riportano i prezzi in equilibrio.
La condizione di non arbitraggio è più stringente dell'argomento di dominanza presente nel CAPM perché tutti gli investitori, a prescindere dalla propensione al rischio, sfruttano l'opportunità di profitto.

6.13 Arbitrage Pricing Thoery (APT)

L'APT è un modello di pricing sviluppato nel 1976 da Stephen Ross che descrive il rendimento atteso di un titolo azionario in funzione di una serie di fattori di rischio.
Ross deriva la relazione del APT combinando un modello fattoriale insieme alla condizione di non arbitraggio.
Il modello multifattoriale APT può essere matematicamente descritto cosi:

$$r_i = E(r_i) + \beta_{i1}F_1 + \beta_{i2}F_2 + \cdots .. + \beta_{ik}F_k$$

E' importante notare come due titoli esposti in ugual modo allo stesso fattore di rischio possono, ex post, portare a rendimenti divergenti. Questo è possibile perché i rendimenti dipendono da due determinati: il rendimento atteso a inizio periodo di valutazione e la variazione inattesa.
A differenza del CAPM, che rappresenta il modello di pricing per eccezione, l'ATP è più robusto, flessibile e realistico: i rendimenti dei titoli sono sensibili a più fattori di rischio, non vengono imposte assunzioni rigide sulla funzione d'utilità degli investitori e il portafoglio di mercato può essere rappresentato da un qualunque portafoglio diversificato. Entrambi possono essere usati per fare valutazioni di capital budgeting e valutazione d'investimenti, però la condizione di non arbitraggio permette di ottenere una relazione senza restrizioni vincolanti. Infatti, diversamente dal CAPM, APT non richiede che il portafoglio benchmark sul SML sia il portafoglio di mercato (portafoglio difficilmente osservabile nella realtà), ma un portafoglio ben diversificato. APT, quindi, non assume ipotesi sulla propensione al rischio degli investitori.
In un modello di APT ogni portafoglio ben diversificato sulla SML può rappresentare il portafoglio benchmark, permettendo di usare indici passivi.
La relazione sviluppata da Ross però, a differenza del CAPM che usa un portafoglio di mercato unico, non vale per tutti i titoli, ma soltanto per un numero limitato. L'ATP risulta quindi essere più adattabile a mercati nei quali coesistono diverse classi d'investitori con funzioni d'utilità e obbiettivi diversi fornendo un quadro di riferimento concettuale utile per organizzare strategie di asset allocation.

6.14 Il modello di Chen, Roll, Ross

Un modello multifattoriale molto utilizzato dalla comunità finanziaria è il modello del 1986 di Chen, Roll e Ross descritto nel paper "Economic Forces and the Stock Market". Essi individuano 5 fattori che tracciano un quadro macroeconomico generale e aiutano a predire i rendimenti dei titoli: la variazione percentuale della produzione industriale, la variazione percentuale dell'inflazione attesa, la variazione percentuale dell'inflazione inattesa, lo spread tra corporate bonds di lungo periodo e bond governativi di lungo periodo, lo spread tra bond governativi di lungo perido e t-bills. Gli autori trovarono che la produzione industriale, l'inflazione inattesa e lo spread sui corporate bond sono fattori significativi con molto potere predittivo.

6.15 Modello a tre fattori di Fama French

Un contributo importante ai modelli fattoriali è stato dato da Fama e French nel 1993. Usando variabili fondamentali piuttosto che macroeconomiche, costruiscono un modello che spiega il rendimento di un titolo.
Questo modello è stato utilizzato da molti ricercatori come base per ulteriori approfondimenti e applicazioni.
A livello matematico la relazione può essere descritta nel seguente modo:

$$r_{it} = \alpha_i + \beta_{iM} R_{Mt} + \beta_{iSMB} SMB_t + \beta_{iHML} HML_t + e_{it}$$

SMB indica il ritorno in eccesso di un portafoglio di azioni di piccola dimensione (small cap) con un portafoglio di grande dimensione (large cap).
HML indica il ritorno in eccesso di un portafoglio di azioni con un alto P/BV con un portafoglio con basso P/BV.
Indice di mercato ($\mathrm{R_{Mt}}$) ha il ruolo di catturare il rischio sistematico d'origine macroeconomico.
Fama e French giustificano il potere predittivo di SMB e HML considerandoli come proxy di valori fondamentali: valori alti di P/BV sono indicativi di problemi finanziari, mentre rendimenti in eccesso delle small cap sono indicativi di una maggior sensibilità ai cambiamenti delle condizioni del business.

6.16 Gestione di Portafoglio Attiva

I fondi d'investimento sono una delle creazioni di maggior successo degli ultimi decenni, permettono di investire in portafogli molto diversificatia costi contenuti.
La gestione può essere condotta in maniera attiva o passiva. Per gestione attiva s'intende una strategia in cui il manager cerca di sfruttare inefficienze di mercato per ottenere extra rendimenti (definiti alpha), dove per extra intendiamo maggiori del benchmark di riferimento. Al contrario dei fondi a gestione passiva, che replicano un indice di mercato, quelli a gestione attiva comprano titoli sottovalutati e vendono quelli sopravalutati. La teoria della gestione attiva è applicabile soltanto nel caso in cui ci siano opportunità da sfruttare all'interno dei mercati finanziari, ed è basata sull'ipotesi di non efficienza dei mercati.
L'ipotesi dei mercati efficienti (efficient-market hypothesis EMH) è una teoria sviluppata da Eugene Fama in cui i prezzi degli assets riflettono tutta l'informazione disponibile. E' quindi possibile affermare che, se si assume l'ipotesi per vera, è impossibile battere il mercato sistematicamente perché gli assets sono scambiati al fair value.
Esistono tre forme di efficienza: debole, semi-forte, e forte. La forma debole afferma che i prezzi degli asstes scambiati riflettono tutta l'informazione passata disponibile; la semi forte che i prezzi riflettono l'informazione pubblica; la forma forte l'informazione privata.
I fondi passivi, invece, non hanno obbiettivi di extra rendimento: sono pensati per ottenere il rendimento del mercato e offrono agli investitori bassi costi ottenuti grazie a un turnover limitato dei titoli (ribilanciamenti di lungo termine) e bassi costi operativi. L'obiettivo dei manager a gestione passiva è quello di minimizzare il tracking error, una misura della bontà della gestione, ottenuta dalla differenza tra il rendimento del fondo e il benchmark di riferimento.

Il dibattito tra i sostenitori della gestione attiva e quelli della gestione passiva è ancora oggi molto forte; il confronto più intenso è nel capire se vale la pena pagare commissioni più alte per cercare di ottenere extra rendimenti attraverso una gestione attiva. Sono state trovate diverse evidenze contrastanti, a favore di una tesi o dell'altra: nel 1991 Sharpe dimostra che la gestione attiva ottiene risultati peggiori di quella passiva considerando i costi (gestione, turnover, tasse ecc.) mentre nel 1996 Elton, Gruber e Blake trovano che è possibile ottenere performance migliori dello S&P500 sfruttando un portafoglio di alpha.
In generale, è dimostrato che è possibile battere la gestione passiva per periodi brevi ma è molto difficile trovare strategie che sistematicamente battono il mercato per periodi lunghi.
Negli ultimi anni la gestione passiva ha avuto molto successo, grazie anche alla evidenza empiricha emersa dalle ricerche che rilevano come i costi sono una componente importante della performance (Bogle, 2002; Malkiel, 1995; Carhart, 1997; Daniel-Grinblatt, Wemers, Titman, 1997)

6.17 Il modello Treynor e Black (TB)

Treynor e Black nel 1973, considerando i mercati finanziari parzialmente efficienti (semi-forte), introducono per la prima volta la possibilità di utilizzare informazioni non riflesse nei prezzi per ottenere extra rendimenti (alpha).
Il modello dimostra che l'utilizzo di un portafoglio passivo con uno attivo crea combinazioni più efficienti di rischio-rendimento. Il portafoglio ottimale, cosi, diviene composto da due parti: una parte passiva, composta da un portafoglio di mercato e una parte attiva, composta da titoli mispriced.
L'elemento distintivo del modello è la complementarietà tra la diversificazione e la selezione dei titoli: l'investitore ha la possibilità di mitigare il rischio (e maggiori rendimenti) generato dalla selezione dei singoli titoli attraverso lo sfruttamento del beneficio di diversificazione prodotto dal portafoglio passivo.
Il modello, nonostante sia stato riconosciuto dalla comunità finanziaria come avvincente, non è stato molto utilizzato in ambito di Asset Management perché la performance, dipendendo molto dall'abilita dell'analista di predire gli extra rendimenti, è sempre sottoposta a test di verifica (Kane, Kim eWhite, 2003).
Treynor e Black formulano alcune assunzioni per descrivere il modello: tutti gli investitori utilizzano il criterio di media varianza (Sharpe Ratio), viene utilizzato un specifico portafoglio di mercato per replicare la strategia passiva, l'analisi per trovare gli extra rendimenti è limitata a un numero ridotto di titoli (i restanti titoli riflettono tutta l'informazione disponibile). In generale, il modello fa uso della maggior parte delle assunzioni del CAPM e del Single-Index (diagonal) model.
L'obiettivo finale del modello è massimizzare lo Sharep Ratio del portafoglio ottimo completo:

$$S_p = \frac{E(R_p)}{\sigma_p}$$

E' possibile riscrivere lo Sharp Ratio del portafoglio completo per mettere in risalto come il contributo del portafoglio attivo sia determinato dal Informatio Ratio, il rapporto tra l'Alpha e la deviazione standard dei residui:

$$S_P^2 = S_M^2 + \frac{\alpha_A^2}{\sigma_A^2}$$

dove $\frac{\alpha_A^2}{\sigma_A^2}$ indica l'Information Ratio, l'extra rendimento ottenuto dalla selezione dei titoli in rapporto al rischio specifico apportato e S_M^2 lo Sharpe Ratio (alla seconda) del portafoglio passivo.

La massimizzazione dello Sharpe Ratio del portafoglio completo comporta la massimizzazione dell' information ratio. Per ottenere ciò, bisogna investire in ogni titolo in proporzione al contributo alpha apportato $\frac{\alpha_i}{\sigma^2(e_i)}$. Cambiando la scala, in modo tale che la posizione totale nel portafoglio attivo è W_A^*, il peso di ogni titolo diventa $W_i^* = W_A^* * \frac{\alpha_i/\sigma^2(e_i)}{\sum_{i=1}^n \alpha_i/\sigma^2(e_i)}$.
E' possibile notare come il contributo positivo generato da ogni titolo all'interno del portafoglio è misurato dall'alpha, mentre quello negativo dall'aumento della varianza del portafoglio a causa dell'aumento del rischio specifico. All'aumentare del numero di titoli con alpha positivi, aumenta la diversificazione all'interno del portafoglio attivo e allo stesso tempo aumenta il peso allocato al portafoglio attivo all'interno del portafoglio completo.
Per ottenere la posizione iniziale da adottare nel portafoglio attivo $W_A^0 = \frac{\alpha_A/\sigma^2(e_A)}{E(R_M)/\sigma_M^2}$ bisogna calcolare $\alpha_A = \sum_{i=1}^n w_i\, \alpha_i$ e $\sigma^2(e_A) = \sum_{i=1}^n w_i^2\, \sigma^2\,(e_i)$ e successivamente aggiustare la posizione in base al $\beta_A = \sum_{i=1}^n w_i \beta_i$ attraverso la seguente formula $W_A^* = \frac{W_A^0}{1+(1-\beta_A)W_A^0}$.

Questo aggiustamento viene fatto perché aumenta la correlazione tra il portafoglio attivo e passivo all'aumentare del Beta del portafoglio attivo.
Il premio al rischio e la varianza del portafoglio ottimale completo diventano:

$$E(R_P) = (W_M^* + W_A^* \beta_A) E(R_M) + W_A^* \alpha_A$$
$$\sigma_P^2 = (w_M^* + w_A^* \beta_A)^2 \sigma_M^2 + (W_A^* \sigma \quad (e_A))^2$$

Alcune considerazioni sul modello: l'obiettivo della security selection è quello di aumentare il rendimento del portafoglio passivo in maniera più che proporzionale rispetto al rischio introdotto.
E' importante sottolineare come il beneficio di diversificazione del portafoglio ottimale completo è maggiore quando è bassa la correlazione tra il portafoglio attivo e quello passivo.
Siccome il portafoglio ottimo è molto sensibile alle previsioni fatte dall'analista sui rendimenti attesi, è possibile rivedere il modello in una impostazione bayesiana per ottenere risultati più robusti. E' possibile combinare una distribuzioni a priori per gli alpha con l'informazione derivante dagli analisti finanziari per ottenere una distribuzione a posteriori. Siccome in assenza delle previsioni degli analisti gli alpha verrebbero

assunti uguali a zero, è possibile stabilire la media della distribuzioni degli alpha uguale a zero e calcolare la varianza dalla serie storica $\alpha \sim N(0, \sigma_\alpha^2)$. Questa distribuzione, chiamata a priori, può essere aggiornata con l'informazione ottenuta dall'analista $\alpha_F = \alpha + e \ \ dove\ e \sim (0, \sigma_e^2 | \alpha)$ per ottenere una distribuzioni a posteriori $E(\alpha|\alpha_F) \sim \frac{\sigma_\alpha^2}{\sigma_e^2 + \sigma_\alpha^2} \alpha_F$ più precisa.

6.18 Il Modello di Black e Litterman (The Canonical Black Litterman Reference Model)

In due articoli importanti dei primi anni novanta, Black e Litterman (1990; 1992) descrivono un modello di asset allocation che permette di superare i limiti del tradizionale modello di Markowitz e di generare allocazioni più sensate. Come dimostrano i due autori, attraverso un approccio bayesiano, è possibile creare una framework di portfolio selection più intuitivo, portafogli meno sensibili agli input (Best e Grauer, 1991), e "mitigare" il problema della massimizzazione degli errori (Lee, 2000), generando portafogli meno concentrati (Litterman, 1992).
La loro soluzione propone di unire dei rendimenti (impliciti) d'equilibrio di mercato con le convinzioni dell'investitore sull'andamento futuro dei rendimenti, per generare una nuova stima più robusta e precisa da utilizzare come input nel processo di media-varianza. Oltre a risolvere molti problemi della teoria di Markowitz, permette all'investitore di tenere in considerazione delle proprie opinioni (views) e del grado di fiducia riposto su di esse. Il modello ha riscontrato grande successo, tant'è che ancora oggi è molto utilizzato nel industria dell'Asset Mangement.
Formalmente, utilizzando la statistica bayesiana, i rendimenti attesi sono considerati variabili casuali non osservabili di cui è possibile inferire la distribuzione di probabilità. Secondo questa impostazione, l'inferenza inizia con una distribuzione a priori, le conoscenze del ricercatore, che viene aggiornata con la nuova informazione addizionale per ottenere una distribuzione a posteriori. In Black Litterman (BL), i rendimenti CAPM d'equilibrio rappresentano la distribuzione a priori, mentre le views dell'investitore l'informazione addizionale. Il teorema di Bayes inferisce la distribuzione di probabilità dei rendimenti attesi usando l'informazione proveniente sia dalle views che dai rendimenti d'equilibrio.
L'obiettivo di BL è riuscire a "modellare" i rendimenti attesi $E(r) \sim N(\mu, \sum)$, che vengono assunti distribuiti come una normale, da utilizzare poi insieme alla matrice di varianza-covarianza come inputs nel processo di ottimizzazione di Markowitz.
Consideriamo μ, la media dei rendimenti sconosciuta, come una variabile casuale distribuita nel seguente modo:

$\mu \sim N(\pi, \sum_\pi)$
dove π è la stima della media e $\sum_\pi$ è la varianza della media ("precisione" della stima).

Questa rappresenta la distribuzione a priori del modello di BL.
E' possibile riscrivere la distribuzione, $\mu = \pi + \varepsilon$, dove $\varepsilon \sim N(0, \sum_\pi)$ e definire la varianza dei rendimenti della stima π cosi:

$$\sum_r = \sum + \sum_\pi.$$

In definitiva, i rendimenti attesi sono distribuiti come una normale:

$E(r) \sim N(\pi, \sum_r)$.

Il modello, sfruttando la teoria d'equilibrio dei mercati finanziari, utilizza come portafoglio d'equilibrio di partenza quello di mercato del CAPM e attraverso un processo d'ottimizzazione inversa individua i rendimenti d'equilibrio di mercato. La relazione tra i rendimenti attesi e il Beta del CAPM, basata su una presunta allocazione efficiente (capitalizzazione di mercato degli assets), viene utilizzata come base per il processo di ottimizzazione inversa.
Partendo dalla massimizzazione della funzione di utilità quadratica (come nel CAPM) è possibile stimare i rendimenti d'equilibrio di mercato attraverso i seguenti passaggi:

$U(w) = w\Pi - \delta 2w\sum w$
$U'(w) = 0$
$0 = \Pi - \delta\sum w$
$\Pi = \delta\sum w$
Π è il vettore dei rendimenti di equilibrio di mercato (Nx1 vettore colonna) in eccesso rispetto al risk-free
δ è il coefficiente di avversione al rischio $\frac{E(R_m)-r_f}{\sigma^2}$
$\sum$ è la matrice di varianza covarianza (matrice NxN) stimata dai rendimenti storici.
w rappresenta i pesi del portafoglio di equilibrio (Nx1 vettore colonna), ovvero le capitalizzazione di mercato degli assets del portafoglio di equilibrio CAPM.

Per ottenere la distribuzione a priori BL viene assunto $\sum_\pi = \tau\sum$, in cui τ è un coefficiente di proporzionalità che riflette l'incertezza nei confronti del portafoglio d'equilibrio o nell'accuratezza della stima di Π (valori minori corrispondono a un alto grado di confidenza nelle stime), e cosi otteniamo:

$P(A) \sim N(\Pi, \tau\sum)$
$E(r) \sim N(P(A), \sum)$

Se l'investitore non ha views, a causa dell incertezza nelle stime, investe una quantità $\frac{1}{1+\tau}$ nel portafoglio d'equilibrio.
Il portafoglio ottenuto può essere modificato in base alle informazioni private in possesso e alla fiducia riposta in esse. E' possibile esprimere le views in maniera assoluta, ovvero rispetto a un rendimento passato, oppure relativa, ovvero rispetto a un rendimento di un altro asset. Assumendo che ogni views è incorrelata con le altre views, otteniamo una matrice di covarianza semplificata (matrice diagonale) che permette di ottenere risultati più robusti.
E' possibile esprimere lo stesso concetto a livello matematico attraverso combinazioni lineari dei rendimenti attesi.
Ipotizziamo k views e n assets, allora è possibile esprimere le views nel seguente modo

$P\mu = Q + \varepsilon$

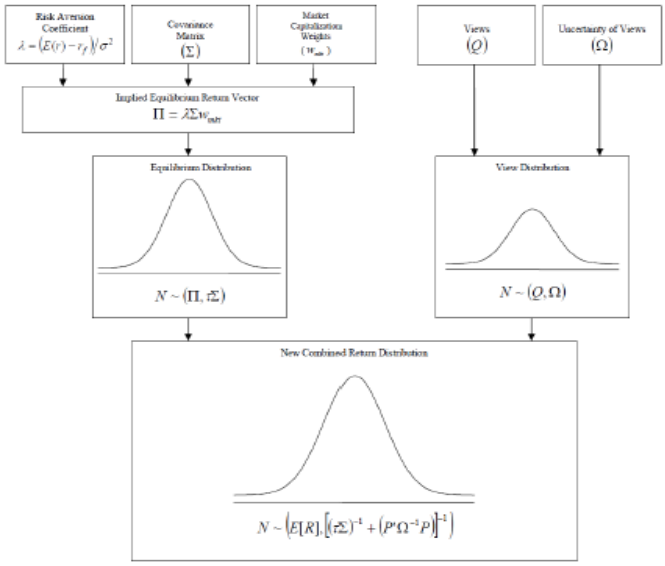
-P è una matrice (k x n) che contiene il peso di ogni views ed è composta da k righe, una per ogni views, e n colonne, una per ogni asse. Se il peso è 0 allora l'investitore non ha opinione e se la view è relativa allora la somma dei pesi è uguale a 0 (se assoluta la somma dei pesi è uguale a 1)
-μ è il vettore (n x 1) che contiene la media dei rendimenti attesi
Q è un vettore (k x 1) che contiene il valore della combinazione lineare ovvero i rendimenti previsti per ogni views
-ε è un vettore (k x 1) che rappresenta la confidenza dell'investitore che si distribuisce $\varepsilon \sim N(0,\Omega)$,
-Ω è la matrice diagonale (k x k) di covarianza delle views che rappresenta l'incertezza dell'investitore per ogni views.

Data queste specificazioni delle views è possibile formulare la distribuzione condizionale della media e varianza nel seguente modo:

$$P(B|A) \sim N(Q, \Omega)$$

La varianza delle views (Ω) è inversamente connessa alla confidenza nelle views; ad ogni modo BL non forniscono un modo per calcolarla, lasciando all'investitore il compito. Esistono vari modi per calcolarla (Walter, 2014), tra cui quello proposto da Black Litterman proporzionale alla varianza della distribuzione a priori.

Figura 6.8. Una sintesi del modello di Black Litterman



Fonte: Idzorek (2005).

Utilizzando la distribuzione a priori e la distribuzione condizionale, attraverso il teorema di Bayes, è possibile derivare la distribuzione a posteriori

$$P(A|B) \sim N([(\tau\sum)^{-1}\Pi + P^T\Omega^{-1}Q][(\tau\sum)^{-1} + P^T\Omega^{-1}P]^{-1}, [(\tau\sum)^{-1} + P^T\Omega^{-1}P]^{-1})$$

L'utilizzo di una distribuzione a posteriori e d'informazioni aggiornate porta a ottenere risultati più precisi (Blamont e Firoozye, 2003) perché l'incertezza data dall'unione di informazioni diverse è minore.
E' importante notare che, aggiungendo l'informazione nuova, variano anche i rendimenti attesi su cui l'investitore non aveva espresso nessuna opinione esplicita perché le views sono espresse come combinazione lineare dei rendimenti attesi, della matrice di varianza covarianza dei rendimenti di equilibrio e della matrice di varianza covarianza delle views. A variare maggiormente sono i pesi dei titoli sui quali l'investitore ha espresso la propria visione.
Infine, l'output ottenuto dal modello di BL viene utilizzato come input del modello di ottimizzazione di media-varianza per ottenere un portafoglio efficiente.
In figura 6.8 si può osservare uno schema rias
suntivo.

6.19 Risk Parity

La Risk Parity è un modello di asset allocation, sviluppato a metà degli anni novanta, che alloca il rischio, definito come volatilità, invece del capitale. Il modello è stato sviluppato secondo i principi della teoria di Markowitz, ma utilizzato per la prima volta dal fondo All Weather nel 1996.
Utilizzando il linguaggio della MPT, la risk parity è un portafoglio a varianza minima in cui ogni asset contribuisce ugualmente alla volatilità del portafoglio.
Il principio di base è che in un portafoglio ben diversificato, ogni asset classes dovrebbe avere lo stesso contributo marginale al rischio totale del portafoglio. E' stato infatti provato (Kazemi, 2012) che in un portafoglio tradizionale 60/40, il rischio dell'equity in realtà conti per quasi il 90% del rischio totale del portafoglio. Utilizzando la risk parity il capitale viene allocato maggiormente verso le asset classes meno rischiose per bilanciare il rischio delle asset classes più rischiose.
Il modello più diffuso di risk parity è l'equally-weighted, che può matematicamente essere descritto nel seguente modo, per un portafoglio di N assets dove il peso dell'asset i è dato da w_ie la matrice di covarianza da Σ:

$$\min_{w} \sum_{i=1}^{N} [w_i - \frac{\sigma(w)^2}{(\Sigma \mathrm{w})_i N}]^2$$

In cui:

$$\sigma(w) = \sqrt{w' \Sigma \mathrm{w}} = \sum_{i=1}^{N} \sigma_i(w)$$

$$\sigma_i(w) = \frac{w_i (\Sigma \mathrm{w})_i}{\sqrt{w' \Sigma \mathrm{w}}} = \frac{\sigma(w)}{N}$$

6.20 Critica della Modern Portfolio Theory: dalla teoria alla pratica, Modern Portafolio Theory 2.0

Nonostante la teoria di portafoglio sviluppata da Markowitz sia ancora oggi la base su cui si sviluppa la finanza moderna, il suo utilizzo nel mondo reale è molto limitato a causa delle assunzioni imposte nella modellizzazione.
Molti esperti del settore (Rockel, 2010) criticano l'impostazione teorica proposta. Ad esempio, Rachev S. (Chief Scientist a FinAnalytica) sostiene che è una teoria sviluppata per mercati "tranquilli", con volatilità e correlazioni costanti. Egli afferma: "The correlations that are embedded as the main assumption -the normality of returns - are telling you the chance of an avalanche affecting everybody in the village below is zero (Rockel, 2010). A sua volta, Brown A. (Risk Manager a AQR Capital Management) ritiene che sia fondata su assunzioni ingannevoli, dando l'impressione all'investitore che gli assets sono sempre rischiosi allo stesso modo, e dunque rendendolo "cieco" alle bolle speculative e diffondendo l'idea che sia sempre possibile comprare e vendere a prezzo di mercato (quando invece la crisi ha evidenziato l'opposto) (Rockel, 2010).
Nel paradigma classico, alla Markowitz, il rischio viene identificato con la varianza dei rendimenti. Questa impostazione ha senso se i rendimenti sono distribuiti come una normale, ma è stato empiricamente dimostrato che nella realtà questa assunzione è sistematicamente violata.
La piu grande limitazione nell'utilizzare una distribuzione normale è la trascuratezza dei movimenti estremi del mercato, le *fat-tails*. Uno studio di Morningstar (2011) fornisce evidenza di questo fenomeno: tra il 1926 e il 2011 ci sono stati 10 mesi in cui i rendimenti sono stati più di tre deviazioni standard sotto la media quando, secondo la distribuzione normale, soltanto 1.3 mesi avrebbero dovuto esserlo. Inoltre, la condizione di normalità considera la varianza una misura di rischio simmetrica, senza distinguere tra movimenti in rialzo o al ribasso: i rendimenti con una deviazione positiva risultano cosi più rischiosi di quanto in realtà sono e viceversa quelli negativi, generando allocazioni sub-ottimali.
Gli investitori non sono interessati al grado di dispersione dei rendimenti, ma piuttosto alla quantità di ricchezza che possono perdere. Le nuove misure di rischio (VaR, Expected shortfall, Conditional VaR) considerano l'asimmetria della distribuzione e le fat-tails. Xiong e Idzorek (2011) dimostrano come le nuove misure di rischio impattano positivamente sull'allocazione ottimale.
Kritzman, professore del MIT Sloan School of Management, spiega invece come la correlazione, intesa come nella MPT, statica e lineare, porta a risultati falsi e spesso opposti da quelli voluti: egli riporta, per esempio, che quando il mercato è in fase toro, la correlazione tra le azioni US e non-US è del 17%, generando un elevato beneficio di diversificazione, mentre quando è in fase orso è del 76%, generando portafogli concentrati. (Rockel, 2010)
Oggi la correlazione viene considerata in evoluzione nel tempo e tutt'altro che lineare, soprattutto nei periodi di stress acuti dei mercati.
Roger Ibbotson, professore alla Yale School of Management, invece sottolinea l'importanza della liquidità nei mercati, un problema ignorato dai modelli di finanza classica: " That liquidity component of what affects returns is every bit as important as the risk component" (Rockel, 2010).
Durante un crisi, come quella del 2007, i titoli possono diventare illiquidi molto velocemente ed essere scambiati, quando possibile, a costi molto elevati; per di più questa situazione è spesso peggiorata dai problemi di leverage.

Anche il tradizionale modello d'ottimizzazione media-varianza è stato messo in discussione dai professionisti del settore. Innanzitutto necessita della stima di una grande quantità di dati da utilizzare come input; poi tratta le stime dei rendimenti e della matrice di covarianza come vere, non considerando la possibilità che ci siano degli errori. Siccome la bontà dei risultati finali deriva principalmente dalla precisione nella stima dei rendimenti, deviazione standard e correlazioni, il problema centrale di questo modello risiede nel fatto che viene massimizzato l'errore di stima (nel 1998 Michaud ha definito il modello di Markowitz "error maximizer"). Tra gli input utilizzati, le stime dei rendimenti futuri sono il fattore più importante, ma anche i meno stabili: nel 1993 Chopra e Ziemba dimostrano che un errore di stima dei rendimento atteso influisce negativamente undici volte di più di più di un errore di stima di varianza.
Un'altra criticità importante è che il portafoglio ottimale ottenuto risulta essere molto instabile: a piccole variazioni negli input corrispondono grande variazioni nella composizione del portafoglio; inoltre, quando non sono presenti vincoli, si creano portafogli con pesi negativi molto estremi, perché l'ottimizzazione sottopesa gli assets con rendimenti attesi bassi, correlazione positiva alta e varianza elevata.
Quando non è possibile vendere allo scoperto, invece, a causa delle limitazioni imposte, i portafogli tendono a essere molto concentrati su pochi assets.
Un'altra limitazione importante è che il modello considera soltanto un orizzonte temporale, quando nella realtà i gestori di fondi, avendo obbiettivi multipli, considerano più orizzonti.
Infine, l'impossibilità di confrontare i portafogli di individui con propensione al rischio differente rende il modello non utilizzabile per fare delle valutazioni assolute. Il modello delinea un insieme di portafogli efficienti, ma non fornisce istruzioni chiare nella scelta del portafoglio ottimo, se non attraverso la curva d'indifferenza dell'investitore. Essendo molto riduttivo e poco intuitivo considerare gli investitori massimizzatori di una funzione d'utilità composta da tre parametri (media, varianza, tolleranza al rischio), oggi molti fondi, articolano gli obiettivi d'investimento e ne valutano l'efficienza in termini di capacità di raggiungimento. Per esempio, lo Yale Investments Office (Lam, 2016) ha stabilito due obiettivi -cash-flow stabili su un orizzonte intermedio per il budget operativo dell'università e mantenimento nel lungo periodo del potere d'acquisto- e due metriche di valutazione -la media su due anni del declino della spesa utilizzando il peggior 10% dell'anno e il fallimento nel preservare metà del potere d'acquisto su 50 anni.
Negli anni sono state proposti modelli alternativi a quello di media-varianza: il Resampled Efficiency, proposto nel 1999 da R. Michaud, tiene conto degli errori di stima utilizzando una simulazione di Monte Carlo che considera i possibili scenari di mercato futuri e Il modello "full-scale optimization", proposto da Kritzman nel 2007 che, al contrario del processo di ottimizzazione di media-varianza che considera una media dei rendimenti degli assets, elabora ogni singolo rendimento nella storia di un asset, permettendo di distinguere tra rialzi e ribassi dei prezzi tenendo in considerazione la maggior avversione alle variazioni negative dei rendimenti. Questa impostazione permette di riconsiderare in maniera diversa le correlazioni tra gli assets soprattutto in periodi di turbolenza sui mercati e di stimare l'esposizione al rischio pensando ai rendimenti come provenienti da regimi di rischio diversi

6.21 Altri stili d'investimento: Analisi Tecnica

L'analisi tecnica tenta di sfruttare patterns ricorrenti nei prezzi dei titoli per generare performance superiori. Secondo questo approccio è possibile prevedere l'andamento futuro del prezzo di un titolo analizzando la storia passata. Gli investitori che utilizzano questo stile d'investimento, basandosi sull'assunto che i comportamenti degli operatori si ripetono nel tempo, cercano di sfruttare gli aggiustamenti graduali dei prezzi ai livelli d'equilibrio e di identificare il cambiamento del trend rispetto a uno stadio iniziale, mantenendo la posizione fino a quando non si riceve un segnale di conferma.
Questa tecnica, sviluppata negli Stati Uniti negli anni trenta, oggi rispecchia un insieme di regole frutto dell'esperienza operativa di migliaia di operatori. I principi di base di questa teoria derivano dalla Dow Theory: il prezzo di un titolo è scomponibile in tre trend, uno di lungo periodo (mesi-anni), uno intermedio (mesi), e uno di breve (giornaliero); esistono tre fasi all'interno di un trend, accumulo, partecipazione del pubblico e distribuzione; i mercati scontano tutte le notizie; i trend devono essere confermate dal volume; l trend esistono fino a che dei segnali non dimostrano il contrario; il trend deve essere confermato anche dall'andamento dei settori a lui collegati.
Le informazioni di base analizzate sono il prezzo, il volume e l'open interest.
Numerosi indicatori di analisi tecnica vengono oggi utilizzati per cercare di predire i prezzi dei titoli: media mobile, bande di Bollinger, MACD, Momentum, RSI, ROC.
La maggior critica mossa a questa impostazione, soprattutto dagli esponenti del mondo accademico, è che la storia dei prezzi non costituisce un'indicazione affidabile di quelli futuri perché l'andamento seguito è aleatorio (random walk).

6.22 Altri stili d'investimento: Analisi Fondamentale

Analisi fondamentale è uno stile d'investimento utilizzato per stabilire il prezzo di un titolo in base alle caratteristiche intrinseche economico finanziarie della società cui fa riferimento. Questo tipo di analisi cerca di individuare il fair value di un titolo attraverso due tipi d'informazione: indicatori relativi al sistema economico nel suo complesso (macro), come il PIL o il tasso d'inflazione, e indicatori relativi alla redditività e alla solidità patrimoniale della società in relazione al prezzo di mercato e alle prospettive di crescita, come il price-earning (rapporto tra il prezzo di mercato e gli utili), il price-book value (rapporto tra il prezzo di mercato e il valore libro).
A differenza dell'analisi tecnica, che è prevalentemente utilizzata per orizzonti temporali brevi, l'analisi fondamentale viene applicata per selezionare le opportunità d'investimento migliori rispetto ai prezzi di mercato nel lungo periodo.
I modelli di valutazione più utilizzati sono il Discounted Cash Flow model (valutazione assoluta), che attualizza i futuri flussi di cassa, e i multipli (valutazione relativa), che analizzano le variabili fondamentali di una società come gli utili, cash flow, book value, dividendi.

CAPITOLO 7
Un'applicazione del modello di Black-Litterman con il Deep Learning

7.1 Finalità e obiettivi del capitolo

La finalità del capitolo è investigare, simulando un sistema di tactical asset allocation, se è possibile, attraverso l'AI, raggiungere risultati migliori di quelli ottenuti dai benchmark classici.
A partire dal modello "canonico" di asset allocation di Black-Litterman (Walters, 2007), sostituendo il portafoglio d'equilibrio con uno ottenuto attraverso un algoritmo di ML e sostituendo le views con quelle predette da un ANN, vengono analizzati i risultati rispetto ai portafogli benchmark suggeriti dalla teoria finanziaria.
Più in dettaglio ci si propone di verificare se il modello presentato possa raggiungere i seguenti quattro risultati.

1. Alpha: extra rendimenti rispetto ai benchmark. Il valore generato dai segnali delle reti neurali dovrebbe riflettersi nei risultati economici attraverso rendimenti corretti per il rischio maggiori dei benchmark.
2. Stabilità del portafoglio d'equilibrio: risultati meno sensibili alle variazioni delle condizioni di mercato. Il portafoglio d'equilibrio, rappresentando il portafoglio detenuto dall'investitore quando non ha visioni sul futuro andamento dei mercati finanziari, dovrebbe essere ben diversificato e robusto in tutte le condizioni di mercato.
3. Flessibilità: un minorricorso a vincoli e assunzioni.
4. Decisioni razionali: riferimento a decisioni maggiormente informate e prive di emozioni.

Prima di presentare le caratteristiche e i risultati del modello viene condotta una breve rassegna della letteratura sul modello di BL e sull'utilizzo delle ANNs per la predizione di serie temporali finanziarie e la costruzione di portafoglio, con l'obiettivo di mostrare che ad oggi l'integrazione tra BL e le ANNs è ancora poco sviluppata.

7.2 Letteratura sul modello di Black Litterman

Il primo modello pubblicato da Fischer Black e Robert Litterman nel 1990 è una research note interna di Goldman Sachs, estesa l'anno successivo in un paper (1991a) pubblicato sul Journal of Fixed Income. Il paper non offre tutte le formule impiegate nel modello, ma soltanto l'impostazione della metodologia. Una rivisitazione del modello è pubblicata nel 1992 sul Financial Analysts Journal. Nel 1999 i due autori forniscono nuovi dettagli, ma le formule rimangono incomplete e l'esempio proposto è difficile da riprodurre.
L'interesse suscitato dal modello di BL, dopo la sua comparsa all'inizio degli anni novanta, ha stimolato numerosi contributi che hanno cercato di estenderne la portata.

In un paper del 1998, Bevan e Winkelman offrono i dettagli della loro esperienza all'interno di Goldman Sachs riguardante l'implementazione del modello in un processo di asset allocation reale, includendo la calibrazione dei parametri.
Satchell e Scowcroft (2000) introducono una nuova espressione non-Bayesiana del problema. Il loro modello alternativo utilizza uno stimatore puntuale invece di una distribuzione, e utilizza τ e Ω per controllare il mix tra prior e views. Questa versione non è molto utilizzata in letteratura e presto viene sostituita da quella di Meucci, che nel 2003, introduce un altro modello non Bayesiano in cui è rimosso il parametro τ, per poi nel 2005 coniare il termine "Beyond Black Litterman" per indicare un nuovo approccio a metà tra quello tradizionale e quello non Bayesiano (Meucci 2003; 2005).
Nel 2003 Herold fornisce una nuova visione del problema cercando di ottimizzare gli alpha delle gestioni attive. Egli integra il modello alternativo di Black con indicatori di rischio attivi per determinare il mix tra le views e il portafoglio d'equilibrio.
Idzorek (2005) introduce una tecnica per specifiare Ω in modo tale che l'impatto possa essere misurato in termini di percentuale di cambiamento dei pesi. Krishnan e Mains (2005) propongono il Two factor Black Litterman, un modello che utilizza fattori che sono incorrelati con il mercato. Nel 2006 Meucci presenta un approccio che utilizza una distribuzione non normale. Cheung nel 2009 introduce il concetto di modello aumentato, una versione che offre l'integrazione con un modello fattoriale.

7.3 Letteratura sull'utilizzo delle ANNs per la predizione di serie temporali finanziarie e la costruzione di portafoglio

La letteratura relativa alle applicazioni finanziarie delle ANNs si presenta estremamente eterogenea. Trattandosi di una materia "ibrida", a metà tra la finanza e l'ingegneria, è difficile trovare lavori di rassegna e/o individuare articoli di particolare rilievo che agevolino la sua sistematizzazione. Scopo di questa rassegna è di individuare le pubblicazioni più importanti in relazione al problema da risolvere e, più in particolare, di mostrare che sono ancora pochi i tentativi di integrare le ANNs con il modello di BL.
Negli ultimi vent'anni, alla luce delle ricerche sulla non linearità delle serie temporali finanziarie (Brock e De Lima, 1995; Zhang, 1998), il ML è stato molto utilizzato per risolvere problemi di previsione. I mercati finanziari sono sistemi dinamici complessi, evolutivi e con molto rumore, difficili da interpretare, tant'è che oggi sono una delle più grandi sfide raccolte dall'AI (Anish e Majhi, 2015).
Le ANNs, grazie all'impostazione non-parametrica e data-driven, raggiungono risultati stupefacenti, migliori di altri modelli statistici come l'analisi discriminante multipla (Yoon e Swales, 1991), la regressione lineare (Enke e Thawornwong, 2005; Comrie, 1997), l'ARIMA (Kohzadi, 1996; Adebiyi et al., 2014) o i GARCH (Desai, 1998; Luna e Ballini, 2012).
Nel 1988 White, uno dei primi ricercatori ad impiegare le ANNs nei mercati finanziari, pubblica un paper in cui utilizza i rendimenti giornalieri di IBM per testare l'ipotesi di efficienza dei mercati confrontando i risultati con quelli ottenuti da un modello autoregressivo lineare. Nonostante White ritenga che le tecniche lineari non siano in grado di cogliere relazioni complesse e abbiano favorito lo sviluppo della teoria di mercati efficienti, non è riuscito a rifiutare l'ipotesi utilizzando le ANNs.

Negli anni a seguire le ricerche si sono intensificate, e i risultati raggiunti hanno dimostrato come i modelli non lineari ottengano risultati più accurati nella predizione dei prezzi degli assets finanziari (Kimoto et al., 1990; Jang e Lai,1994; Kryzanowski, 1992; O'Conner e Madden, 2006; Niaki e Hoseinzade, 2013; Mingyue, Cheng e Yu, 2016).
Già dagli anni novanta è possibile trovare papers che utilizzano le reti neurali ricorrenti (RNNs) e le Long-Short Term-Memory (LSTMs): nel 1990, per esempio, Kamijo e Tanigawa allenano una RNN a individuare patterns nei mercati finanziari, come i triangoli, raggiungendo un'accuratezza nel riconoscimento di quindici prove su sedici. Altri studi relativi all'uso di reti neurali ricorrenti sono proposti da Kuan e Liu (1995), da Wang e Leu (1996), Prokhorov, Saad e Wunsch (1998), da Hsieh, Hsiao e Yeh (2011) e da Chen, Zhou e Dai (2015).
Le reti neurali convoluzionali (CNNs), invece, sono d'applicazione più recente. Nel 2014 un ricercatore di Stanford, Siripurapu, utilizza una rete neurale convoluzionale (CNN) per predire l'andamento futuro di titoli finanziari a partire da una "immagine" delle variazioni passate dei prezzi. Altre ricerche più recenti sono proposte da Ding, Zhang e Duan (2015), da Di Persio e Honchar (2016), da Chen e Huang (2016) e infine da Zhou et al. (2018).
Le ANNs sono utilizzate anche per la costruzione di portafoglio; il Pension and Investment Department of Deere & Company (Hall, 1994), per esempio, costruisce vari portafogli attraverso lo stile d'investimento dei titoli che hanno ottenuto le performance migliori nel breve periodo. Freitas (2001), invece, stima i rendimenti attesi da utilizzare nel modello di media-varianza a partire dalle previsioni stimate dalle ANNs sul futuro prezzo dei titoli.
Fernandez e Gomez (2007) tracciano la frontiera efficiente secondo la teoria di Markowitz grazie all'uso di una particolare rete, l'Hopfield network, concludendo che è possibile ottenere risultati migliori dei modelli classici.
Ko e Lin (2008) analizzando i prezzi, le covarianze e varianze di 21 compagnie taiwanesi, creano un portafoglio che ottiene rendimenti maggiori del Taiwan 50 Index. Altri paper rilevanti sono quelli di Thim e Seah (2010) e quello di Heaton, Polson e Witte (2016).
Poche ricerche, invece, utilizzano l'impostazione di Black-Litterman (BL) per la costruzione del portafoglio. Zimmermann (2002) propone un'applicazione di BL che utilizza le previsioni proposte da una error correction neural networks (ECNN) per deviare, entro certi limiti, dal portafoglio d'equilibrio (CAPM). Il modello è diviso in tre fasi: costruzione del modello di previsione ECNN; calcolo della profittabilità di un asset rispetto agli altri; ottimizzazione e allocazione entro certi limiti. Rispetto al modello tradizionale le previsioni vengono fatte con un ECNN, le decisioni d'investimento vengono prese ottimizzando le previsioni, l'esposizione al rischio del portafoglio è implicitamente controllata nel tempo.
Creamer nel 2015 presenta un modello di BL che utilizza dati non strutturati -il *sentiment* di news, raccomandazioni degli analisti e indicatori corporate- per generare le views. Il portafoglio ottenuto raggiunge risultati migliori del benchmark.
Nel 2016 Mudasir, Subekti e Kusumawati propongono invece di utilizzare un radial basis neural network (RBFNN) per predire le views, ottenendo livelli buoni d'accuratezza.
Recentemente sono state pubblicate numerose ricerche sui vantaggi generati dalla analisi del *sentiment* di news, social network e siti nella predizione di assets finanziari. Il paper più citato è quello di Bollen, Mao e Zeng ("Twitter mood predicts the stock market" del 2011) che mostra che la predizione sul DJIA migliora significativamente

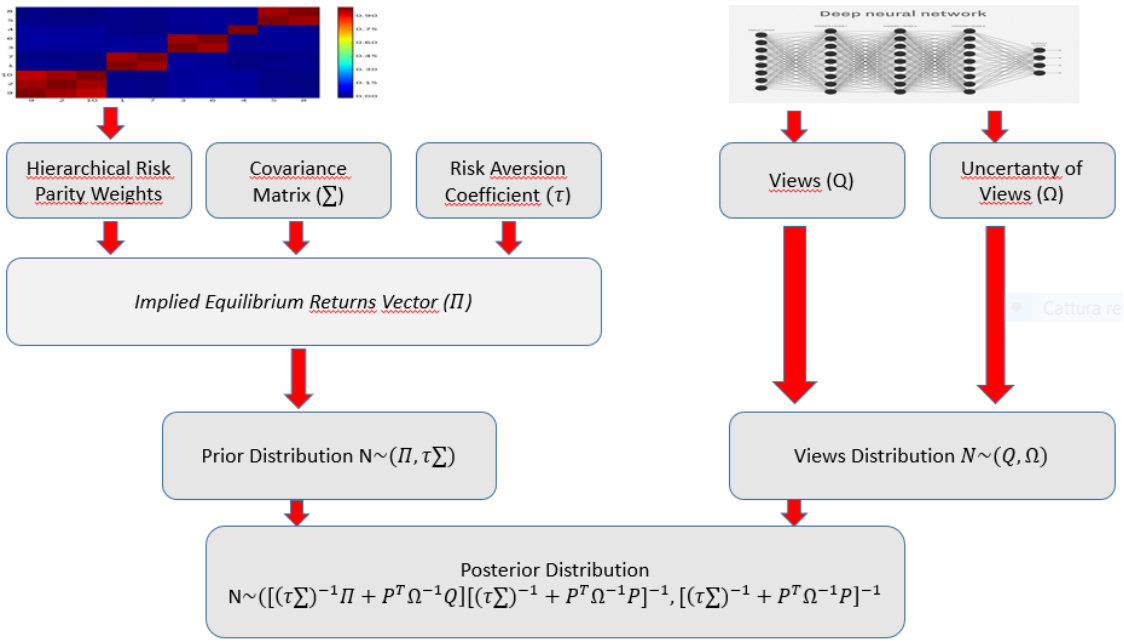
analizzando "l'umore" dell'opinione pubblica. Altre ricerche relative all'uso di dati non strutturati sono quelle di: Peng e Jiang (2015); Xiong, Nichols e Shen (2015); Ding, Zhang e Duan (2014); Li, Bu e Wu (2017); Sohangir, Wang e Pomeranets (2018).
In conclusione, non esistendo in letteratura significative applicazioni del modello di Black-Litterman con il Deep Learning, è interessante capire quanto queste reti profonde, che controllano moltissima informazione, possono ottenere predizioni più accurate e superare i problemi tradizionali di costruzione del portafoglio.

7.4 Il processo d'investimento

Il modello utilizzato per questa sperimentazione presso Axyon AI utilizza i rendimenti d'equilibrio Π come punto di partenza. Questi possono essere interpretati come i rendimenti di lungo periodo forniti dai mercati finanziari, calcolati attraverso un processo d'ottimizzazione inversa nel quale il vettore è estratto utilizzando l'informazione nota, il parametro d'avversione al rischio δ, i pesi w del portafoglio ottenuti attraverso l'Hierarchical Risk Parity e la matrice di varianza-covarianza dei rendimenti storici $\sum$. Invece di utilizzare, come pesi, la capitalizzazione di mercato degli assets che compongono un portafoglio di mercato CAPM, sono utilizzati quelli prodotti da un portafoglio di Risk Parity ottenuto con un algoritmo di ML. Dal punto di vista della teoria finanziaria, siccome la Risk Parity è un portafoglio efficiente, può sostituire il portafoglio d'equilibrio suggerito da BL. La decisione è favorita dal fatto che è possibile ottenere un portafoglio perfettamente diversificato in termini di rischi, meno concentrato, più robusto (Haesen, 2017) e, secondo alcuni studi (Asness, Frazzini e Pedersen, 2012), con performance migliori rispetto ai tradizionali market-cap o media-varianza.
Il portafoglio ottenuto diventa il centro gravitazionale del modello, ovvero quel portafoglio che tutti gli investitori detengono nel caso in cui non ci siano views. L'idea fondamentale è che l'equilibrio esistente nei mercati finanziari, rappresentato dai pesi di un portafoglio efficiente d'equilibrio (Risk Parity), serva come base per l'allocazione ottimale.
All'interno del framework sviluppato da BL, i rendimenti impliciti Π insieme alla matrice di varianza-covarianza moltiplicata per una costante di proporzionalità $\tau\sum$ rappresentano la distribuzione a priori $P(A) \sim N(\Pi, \tau\sum)$.
L'allocazione di base viene poi ribilanciata in base alle views dell'asset manager. Queste sono generate da un sistema di ANNs profonde e, attraverso il teorema di Bayes, sono integrate con il portafoglio d'equilibrio per generare una nuova allocazione.
Nel dettaglio, per ciascun asset del portafoglio, le ANNs producono un segnale che rappresenta la probabilità che tale strumento ottenga, sull'orizzonte temporale stabilito, una performance maggiore della media degli strumenti (o di un benchmark). Il segnale viene convertito, secondo l'impostazione di BL, in un vettore di rendimenti Q e in una matrice di varianza-covarianza delle views Ω.
Attraverso le specificazioni delle views vengono formulate la media e la varianza della distribuzione condizionale nello spazio delle views $P(B|A) \sim N(Q, \Omega)$. Data la distribuzione a priori e la distribuzione condizionale, applicando il teorema di Bayes, è possibile ottenere la distribuzione a posteriori $P(A|B) \sim N([(\tau\sum)^{-1}\Pi + P^T\Omega^{-1}Q][(\tau\sum)^{-1} + P^T\Omega^{-1}P]^{-1}, [(\tau\sum)^{-1} + P^T\Omega^{-1}P]^{-1})$.

Dalla distribuzione è possibile ricavare i nuovi rendimenti attesi ($[(\tau\sum)^{-1}\Pi + P^T\Omega^{-1}Q][(\tau\sum)^{-1} + P^T\Omega^{-1}P]^{-1}$), una media pesata tra la distribuzione a priori e le stime condizionali, da utilizzare nel processo di media-varianza per ottenere una nuova allocazione ottimizzata.

Figura 7.1. Il processo d'investimento



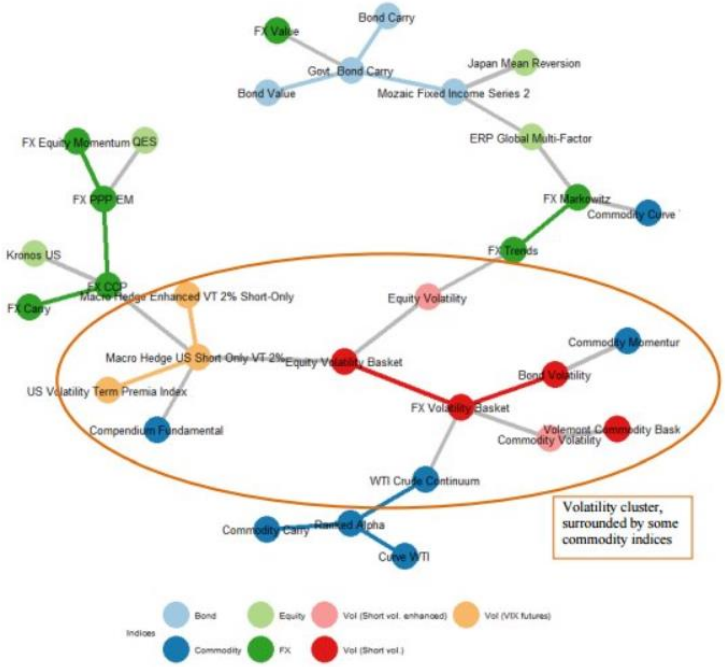
La nuova allocazione privilegia gli assets su cui l'asset manager è più confidente ed è una combinazione di due portafogli, quello d'equilibrio e quello di un portafoglio composto da una somma pesata delle views. La figura 7.1 schematizza il processo d'investimento descritto.

7.5 Problem Framing: Hierarchical Risk Parity (HRP)

Il portafoglio d'equilibrio viene composto seguendo la teoria Hierarchical Risk Parity (HRP) sviluppata da Prado (2016*b*). Secondo l'Autore, l'inversione della matrice di correlazione richiesta dalla programmazione quadratica genera errori di magnitudo tale da compensare il beneficio di diversificazione. Piccoli errori di stima vengono "ingranditi", generando soluzioni sbagliate. Questo è un problema per i portafogli di media varianza, i quali ottengono buone performance in-sample, ma pessime out-of-sample. Uno dei motivi principali per cui l'ottimizzatore genera risultati instabili è che lo spazio vettoriale è modellato come un grafico in cui ogni nodo è collegato ad un altro, in cui tutti gli investimenti sono possibili sostituti di altri.
La matrice di correlazione manca della nozione di gerarchia. L'approccio suggerito da Prado, generando un albero di cluster tra le variabili che sono simili (figura 7.2), elimina i collegamenti non necessari ottenendo portafogli più robusti. Le relazioni tra gerarchie, anche nei mercati finanziari, sono la chiave per capire fenomeni complessi (Papenbrock,

2016). La teoria dei grafi e il ML, introdotti da Prado, aiutano a capire le vere relazioni tra gli assets.
L'idea principale è quella di creare una gerarchia di cluster all'interno della matrice di correlazione dei rendimenti e allocare ad ognuno lo stesso ammontare di capitale, sfruttando l'informazione contenuta all'interno della matrice senza invertirla. Il modello, inoltre, permette di utilizzare matrici singolari. Prado riporta nel suo paper che per invertire una matrice di dimensione 50 è necessario utilizzare 5 anni di dati giornalieri, un intervallo temporale troppo esteso affinchè la correlazione rimanga costante.
Per le caratteristiche appena delineate risulta interessante capire il contributo di questo approccio in un sistema di tactical asset allocation.

Figura 7.2. Hierarchical Risk Parity, l'albero di cluster



Fonte: Kolanovic et al. (2017).

7.6 Problem Framing: Deep Neural Networks

Le predizioni sono generate da una rete neurale profonda (Deep Neural Network) utilizzando, per ciascun titolo, i dati storici raccolti alla chiusura dei mercati ogni giorno. Nei dati relativi ad ogni titolo (inclusi prezzi e volumi di scambio, indicatori tecnici e fondamentali, indicatori riguardanti il settore di ogni titolo, oltre che il contesto finanziario e economico complessivo) le reti sono allenate a riconoscere pattern che anticipano andamenti più o meno performanti del titolo considerato rispetto all'indice di riferimento. Tutto ciò è possibile attraverso un processo di applicazione di avanzati algoritmi di apprendimento, durante il quale le reti eseguono un elevato numero di operazioni su mercati simulati (in media tra i 5 e i 10 milioni di operazioni), apprendendo dai propri errori.
La struttura della rete varia a seconda dei dati processati, della tipologia di mercato affrontato e degli obiettivi di predizione, ma in generale è ideata per risolvere una serie

di problemi di classificazione binaria (0 sottoperforma il mercato e 1 overperforma il mercato), uno per ogni orizzonte di predizione. Per una specifica azione, su ciascun orizzonte temporale supportato, ciascuna predizione rappresenterà dunque la probabilità che tale azione ottenga (sull'orizzonte temporale) una performance maggiore dell'indice stesso. Utilizzando il linguaggio del modello di Black Litterman, la views è definita relativa.
Ad esempio, una predizione di 54% su FSTE MIB per l'orizzonte temporale "dieci giorni" significa che il sistema attribuisce una probabilità pari al 54% a FSTE MIB di ottenere una performance superiore a quella dell'indice tra oggi e dieci giorni in avanti, mentre una predizione di 46% su FSTE MIB per l'orizzonte temporale "dieci giorni" significa che il sistema attribuisce una probabilità pari al 54% ad FSTE MIB di ottenere una performance minore a quella dell'indice tra oggi e dieci giorni in avanti.
Il processo d'allenamento può essere rappresentato come un ciclo, scandito in 4 fasi, che punta a migliorare i risultati ad ogni iterazione:

1. Preparazione del dataset: aggiunta di dati, nuove features predittive, ecc.
2. Evoluzione dei modelli: viene migliorata l'architettura della rete
3. Genetics: vengono usati algoritmi genetici per scegliere le migliori features e eliminare quelle inutili
4. Miglioramento dei modelli: i modelli vengono modificati tenendo in considerazione il numero ridotto di features.

Terminata la quarta fase, si è raggiunto un risultato intermedio e il ciclo può ricominciare dall'inizio.
La predizione per un asset viene poi convertita in Q mediante una funzione che associa ad ogni segnale il rendimento delta ottenuto tra azione e indice. Questa funzione è ottenuta analizzando tutti i segnali (200.000) generati nella storia della rete per l'orizzonte temporale di un mese. Dai risultati dell'analisi dati (tabella 7.1) è osservabile che i segnali generati dalla rete hanno una dispersione che va da 0,35 a 0,54, dove 0,50 è il valore soglia tra un segnale d'acquisto e uno di vendita (l'intervallo $0{,}50 \pm 0{,}01$ viene escluso perché è composto da segnali troppo "rumorosi").

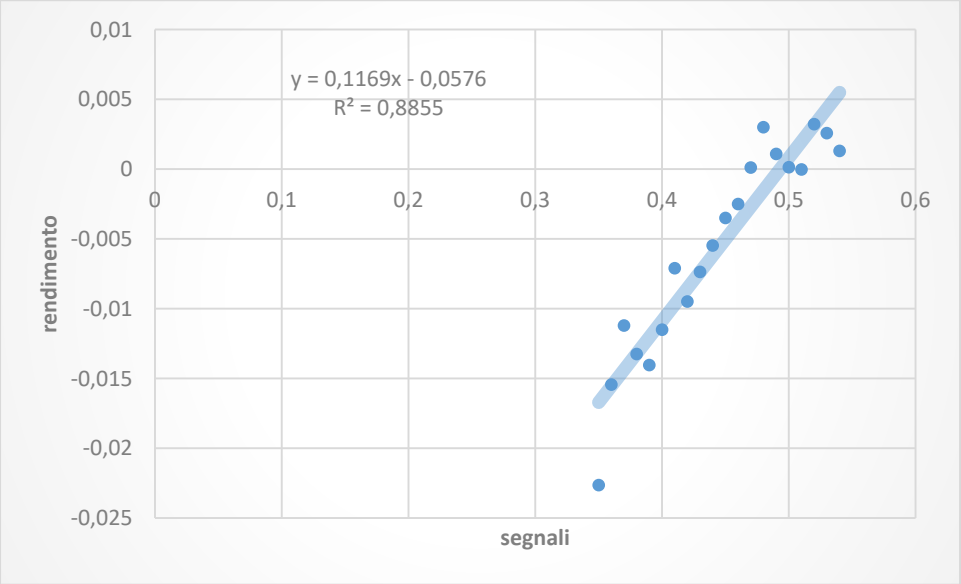
Tabella 7.1. L'analisi dei segnali

centro	min	max	media	dev standard
0,35	0,35	0,355	-0,023	0,041
0,36	0,355	0,365	-0,015	0,039
0,37	0,365	0,375	-0,011	0,033
0,38	0,375	0,385	-0,013	0,037
0,39	0,385	0,395	-0,014	0,035
0,4	0,395	0,405	-0,012	0,033
0,41	0,405	0,415	-0,007	0,034
0,42	0,415	0,425	-0,009	0,034
0,43	0,425	0,435	-0,007	0,033
0,44	0,435	0,445	-0,005	0,033

0,45	0,445	0,455	-0,004	0,033
0,46	0,455	0,465	-0,003	0,033
0,47	0,465	0,475	0,000	0,035
0,48	0,475	0,485	-0,003	0,037
0,49	0,485	0,495	-0,001	0,035
0,5	0,495	0,505	0,000	0,033
0,51	0,505	0,515	0,000	0,032
0,52	0,515	0,525	0,001	0,034
0,53	0,525	0,535	0,002	0,032
0,54	0,535	0,545	0,003	0,030

E' importante notare (tabella 7.1) che la rete genera molti più segnali di vendita che di acquisto come se avesse imparato a vendere ma non a comprare. Probabilmente la rete è stata allenata in un periodo in cui hanno prevalso i segnali di vendita su quelli d'acquisto. Questo comportamento, di difficile interpretazione, meriterebbe un approfondimento maggiore, ad ogni modo, questo tipo di analisi, non essendo in linea con gli obiettivi della tesi, non viene compiuta.
Per convertire il segnale in Q viene utilizzata la relazione y=0,1169x-0,0576. Come auspicabile, la relazione (grafico 7.1) tra il segnale generato dalla rete e il rendimento delta è lineare. Al crescere della confidenza del segnale, il profitto aumenta. Per ottenere invece Ω vengono utilizzate le deviazioni standard degli intervalli. Questo approccio, a differenza di quello di Black-Litterman, è guidato dai dati.

Grafico 7.1. L'analisi dei segnali. Sull'asse della ascissa sono rappresentati i segnali generati dalle reti neurali, sull'asse delle ordinate il rendimento delta ottenuto rispetto al mercato.



7.7 Metodologia: data collection (Deep Neural Networks)

Il successo di una rete neurale dipende dalla comprensione del problema che si sta affrontando. Selezionare le variabili d'input da utilizzare è un passaggio critico da supportare con la teoria economica. Il ricercatore interessato a predire i prezzi di mercato deve decidere quali indicatori utilizzare, tecnici o fondamentali (o entrambi) e per quale mercato, considerando le diverse caratteristiche dei dati.
Per esempio, la frequenza dei dati da utilizzare dipende dagli obiettivi da raggiungere; generalmente i dati giornalieri sono utili per predire movimenti intragiornalieri (trading), i dati mensili o trimestrali, invece, per strategie di lungo termine tipo la Buy and Hold.
E' importante considerare anche il costo e la disponibilità dei dati, oltre che la qualità.
Nel modello preso in considerazione vengono fornite alla rete ogni giorno, per ogni asset, per ogni orizzonte temporale supportato i dati che seguono.

- **Dati fondamentali/economici**, ovvero dati macroeconomici e contabili, generalmente con una frequenza di rilascio bassa (trimestrali). Quando utilizzati da soli hanno poco potere predittivo a causa della loro grande diffusione (Prado, 2018). Offrono informazione sul ciclo economico, e conseguentemente su quello finanziario con riferimento alle specifiche aree economico-geografiche (area Euro, Giappone, Usa). Dopo un attenta ricerca, sono utilizzati gli indicatori proposti da Chen, Roll e Ross (1987) e quelli indicati da Bodie e Kane nel libro "Investments" (2013) creati dal Conference Board. L'idea è quella di utilizzare un insieme di indicatori economici standard per ogni area economico-geografica. La scelta di utilizzare un numero ridotto d'indicatori è motivata in parte dai vincoli sulle risorse aziendali disponibili e in parte dalla conoscenza ancora oggi poco diffusa sugli indicatori economici adatti a sistemi di ML.
- La lista degli indicatori utilizzati è la seguente:
  "Industrial confidence indicator", "Consumer confidence index", "Interest rate 0-3 month", "Industrial Production", "CPI", "Benchmark 20 year government", "Unemployment rate", "GDP", Money supply", "Benchmark 10 year government", "M1", "New houseing construction", "Current account balance", "Price house", "Bank lending", "Gross external debt", "Deficit/surplus", "Canada raw materials", "Foreign Reserve assets", "Expected inflation", "Unanticipated inflation","change in expected inflation", "Term strcture", "Bond risk premium", "Manifacturing trade sale", "Producer price index", "Petrolium products refined", "Equiy premium", "Import", "Export", "Unit labor cost", "bank lending", "Economic *sentiment*", "Personal saving", "Average wage manufacturing", "import price index", "Export price index", "Employment","Exchange rate", "Poulation", "Personal saving", "Average weekly hours, manufacturing", "Average weekly initial claims for unemployment insurance", "Manufacturers' new orders, consumer goods and materials", "new orders index ", "Manufacturers' new orders, nondefense capital goods excl" aircraft ", "Stock prices, 500 common stocks", "oil", "Avg" consumer expectations for business conditions ", "Employees on nonagricultural payrolls", "Personal income less transfer payments", "Manufacturing and trade sales ", "Average duration of unemployment ", "Consumer installment credit outstanding to personal income ratio", "Commercial and industrial loans ", "Consumer price index for services "

- **Dati di mercato/indicatori tecnici**, che derivano da attività di trading che hanno luogo nei mercati. Utilizzando in gran parte dati storici vengono considerati indicatori *lagged*, utili per avere conferma di un trend, ma non per prevedere il movimento futuro.
  Vengono utilizzati gli indicatori tecnici creati internamente da Axyon AI: Absolute Price Oscillator (APO), Chande Momentum Oscillator (cmo), Momentum (mom), Rate of Change (ROC), Relative Strenght Index (rsi), Weighted Moving Average (wma)

  “apo_5_10“,“apo_10_20“,“apo_20_40“,“apo_20_40“,“cmo_20“,“cmo_40“,“mo m_5“,“mom_10“,“mom_20“,“mom_40“,“roc_5“,“roc_10“,“roc_20“,“roc_40“,“rs i_5“,“rsi_10“,“rsi_20“,“rsi_40“,“wma_ratio_5“,“wma_ratio_10“,“wma_ratio_20“ ,“Wma_ratio_40.

- **Le serie temporali dei prezzi (giornalieri) e del volume di trading** di tutti gli assets:

  “Datastream US Equity Index”, “S&P 1500”, “S&P 500 COMPOSITE”, “S&P 600 Small Cap”, S&P 400 Mid Cap, “TOPIX”, ”FTSEMIB”, “INDEX”, “IBEX 35 INDEX”, “US Financials Equity Index”, “US Health Care Equity Index”, “US Banks Equity Index”, “US Oil&Gas Equity Index”, “US Technology Equity Index”, “Datastream Developed Markets Equity Index”, “CAC Index”, “DAX Index”, Thomson Reuters Asia Pacific Equity Index”, “Datastream UK Equity Index”, “Datastream Canada Equity Index”, “Datastream Japan Equity Index”, “Datastream Germany Equity Index”, “Datastream Italy Equity Index”, “Datastream China Equity Index, “S&P ASX200”,“Datastream Russia Equity Index”, “Datastream Latin AmericaEquity Index”,“Datastream Pacific Equity Index”, “Datastream World Ex US Equity Index”,“Datastream World Ex EMU Equity Index”, “Thomson Reuters G7 Equity Index”, “Datastream Europe Farest Equity index”, Datastream Europe Ex UK Equity index”, “Datastream Farest Equity index”, “Thomson Reuters Italy 50 Equity Index”, “Datastream Italy Small Cap Equity Index”, “Eurostoxx 50”, “MSCI World”, “MSCI Emerging Market”, “S&P500 Growth”, “S&P500 Value”, “S&P500 High Beta”, “S&P500 Low Volatility”, “S&P500 Composite”, “S&P500 130/30”, “MSCI EAFE”, “MSCI EUROPE”, “MSCI PACIFIC”, “DOW JONES Industrials”, “Datastream Government US 10y”, “Datastream Government US 30y”, “Datastream Government US 5y”, “Datastream Government US Over 10y”, “Datastream Government US 7y-10y”, “Datastream Government US 5y-7y”, “Datastream Government US 3y-5y”, “Datastream Government US 1y-3y”, “Datastream Government EU 10y”, “Datastream Government EU 30y”, “Datastream Government EU 5y-7y”, “Datastream Government EU 3y-5y”, “Datastream Government EU 1y-3y”, “Datastream Government EU 7y-10y”, “Datastream Government EU Over 10y”, “Datastream Government UK 1y-3y”, “Datastream Government UK 5y”, “Datastream Government UK 10y”, “Datastream Government UK Over 10y”, “Datastream Government IT10y”, “Datastream Government IT 30y”, “Datastream Government IT 1y-3y”, “Datastream Government IT over 10y”, “Datastream Government IT 5y”, “Datastream Government GER 10y”, “Datastream Government GER 30y”,

"Datastream Government GER 1y-3y", "Datastream Government GER Over 10y", "Datastream Government GER 5y", "Thomson Reuters Corporate Convertible US Investment Grade ", "Thomson Reuters Corporate Convertible Europe Investment Grade ", "Thomson Reuters Corporate Convertible Global Investment Grade" , "Datastream Government JPY 1y-3y", "Datastream Government JPY 5y", "Datastream Government JPY 10y", "Datastream Government JPY Over 10y", "FTSE Government Emerging Markets 1y-3y", "FTSE Government Emerging Markets Over 10y", "FTSE Government Emerging Markets 3y-5y", "FTSE Government Emerging Markets 5y-7y", "FTSE Government Emerging Markets 7y-10y", "IBOXX Corporate EU High Yield Fixed Rate", "ECAPITAL Corporate EU ", "FTSE Corporate UK All Maturities", "FTSE Corporate EU All Maturities", "EMTS T-Bills EU 6m", "IBOXX Corporate EU All Maturities", "FTSE Corporate UK BBB", "FTSE Corporate UK AAA", "IBOXX Corporate UK All Maturities", "FTSE Corporate EU AAA", "FTSE Corporate EU BBB ", "IBOXX Corporate EU BBB 10+", "S&P500 Bond ", "S&P500 AAA ", "S&P500 BBB", "S&P500 High Yield", "S&P500 Energy Bond", "S&P500 Financials Bond", "S&P500 Health Care Bond", "S&P500 Industrial Bond", "S&P500 Tech Bond", "S&P500 Investment Grade", "S&P500 Materials Bond", "S&P500 Utilities Bond", "Oil", "Gold", "Silver", "Oil&Gas Index", "Commodity Index", "GSCI Commodity Spot", "Datastream Real Estate UK", "Datastream Real Estate US", "Datastream Real Estate China", "Datastream Real Estate Asia", "Datastream Real Estate Developed Market", "Datastream Real Estate EMU", "Datastream Real Estate Emerging Market", "Datastream Real Estate Italy", "Datastream Real Estate US Commercial","Datastream Real Estate US Resident", "USD/EUR", "EUR/GBP", "AUD/EUR", "EUR/CAD", "USD/GBP", "EUR/JPY", "USD/JPY", "GBP/JPY","EUR/YUAN"

7.8 Metodologia: preparazione del dataset (Deep Neural Networks)

Il dataset viene preparato a partire dalle serie temporali, creando nuove features e normalizzando i dati. Durante la fase di "data pre-processing" i dati grezzi vengono analizzati e trasformati per minimizzare il *noise* ed estrarre le relazioni più importanti. Questa fase comporta molte prove ed errori. Una prassi comune è prendere la differenza prima e il logaritmo naturale di una variabile. Attraverso la prima operazione si elimina il trend dai dati; con la seconda invece si trasforma una relazione moltiplicativa in additiva, semplificando i conti e la distribuzione. Il dataset viene poi filtrato per rimuovere le osservazioni che non servono e creare una distribuzione più uniforme. L'obiettivo della normalizzazione, invece, è cambiare i valori del dataset per usare una scala comune, senza perdere informazione o creare distorsioni. Nel modello utilizzato i dati d'input sono standardizzati usando la statistica classica (media e deviazione standard).
Il dataset (tabella 7.2) fa riferimento a un periodo temporale che inizia l'1/01/2000 e termina il 15/04/2017 e contiene 137 assets. Il dataset contiene sia le features che le predizioni target (se un asset sovra-performerà o sotto-performerà il mercato per ogni

orizzonte temporale). Gli orizzonti temporali presi in considerazione sono cinque, ovvero 5-10-20-40-60 giorni.

Tabella 7.2. caratteristiche del dataset generato

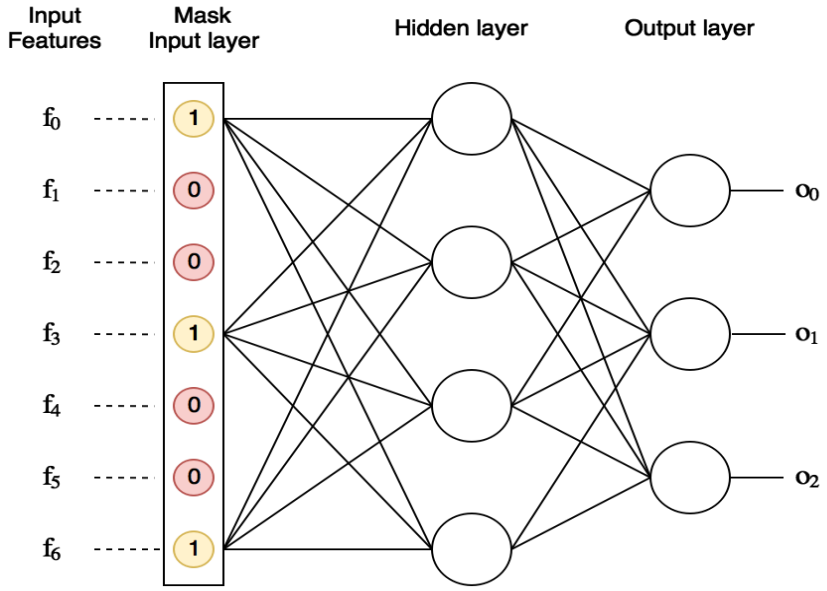
Numero di assets	137
Intervallo temporale	Dal 1/01/2000 all'15/04/2017
Numero di samples	627'286
Numero di input features	224 (3 categorie: geografica, settoriale, classe d'investimento)
Variabile target	0 (sottoperforma il mercato) o 1 (sovraperforma il mercato) per ogni orizzonte temporale. Totale: 5 variabili.
Pre-processamento	Standardizzazione del dataset e riempimento dei valori nulli con il valore mediano

7.9 Metodologia: features Selection (Deep Neural Networks)

Attraverso l'uso di algoritmi genetici vengono estratte le features con valore predittivo. Un metodo per selezionare le variabili d'input più importanti è provare varie combinazioni, per esempio una lista di 20 indicatori tecnici può essere testata selezionando 10 indicatori e provando varie combinazioni con le restanti variabili. Sebbene computazionalmente intensiva, questa procedura riconosce la probabilità che alcune variabili siano eccellenti predittori solo in combinazione con altre variabili.
Axyon Genetics è sviluppato per trovare le migliori combinazioni di features da impiegare come input nei modelli predittivi. Supponiamo che il dataset in input sia composto da n features [f0, f1, , …, fn-1]. La selezione delle features è stabilita da un vettore binario (0 o 1) di lunghezza n, chiamato "maschera". Ad ogni entrata della maschera corrisponde una feature del dataset, e ogni valore 1 o 0 determina se la corrispondente featurs è utilizzata durante l'allenamento del modello.
In figura 7.3 è possibile osservare un esempio di un dataset caratterizzato da 7 features input. Una maschera di lunghezza sette può essere applicata al dataset input per filtrare le features f1, f2, f4 e f5. Le featurs mantenute dalla maschera sono f0, f3 e f6.
Questo metodo permette di rendere piu veloce il paragone tra dataset, infatti partendo dallo stesso vengono selezionate combinazioni multiple delle features d'input senza la necessità di generarne di nuovi.
Analizzando il numero di volte che una features è utilizzata dalla rete, è possibile individuare il valore predittivo. Nella simulazione presa in esame, il tipo di strumento, il settore, la geografia insieme agli indicatori tecnici sull'oro e sul petrolio sono i più utilizzati. Su 224 features in input, 99 sono state utilizzate, di cui 29 fanno riferimento

all'oro/petrolio, 28 al tipo di strumento, 14 alla geografia/settore, mentre la restante parte è divisa tra indicatori tecnici e fondamentali.

Figura 7.3. Axyon Genetics



7.10 Metodologia: training, validation, test sets (Deep Neural Networks)

Nei problemi di previsione su setting supervisionati, un metodo standard per prevenire l'overfitting è dividere il dataset in tre parti: la prima parte di dati (training set) è usata dal modello per aggiustare i parametri (pesi), la seconda parte (validation set) è usata per ottimizzare gli iperparametri del modello (come ad esempio il numero di layer), l'ultima parte per testare il modello. Questo processo viene chiamato cross-validation. Le dimensioni dei vari intervalli devono essere scelte bilanciando l'esigenza di avere un numero elevati di osservazioni su cui allenare la rete e allo stesso tempo avere abbastanza esempi per testare la capacità di generalizzare. Solitamente il test set utilizza le osservazioni più recenti perché ritenute le più importanti. Per allenare la rete sono stati seguiti i seguenti passaggi (*walk-forward sliding windows testing*):

1. Il training set (50% del dataset) è selezionato all'inizio del dataset, il validation (10%) e il test (1%) sono successivi. Una porzione finale del dataset è tenuta da parte e non usata.
2. La suddivisione precedente è ripetuta allargando il training set di un certa frazione, facendo scorrere il validation set e il test set conseguentemente. Facendo ciò, una piccola parte finale di dati non viene utilizzata.
3. I passaggi appena elencati vengono ripetuti varie volte finché il test set raggiunge la fine del dataset.

Siccome la rete sviluppata da Axyon AI è pensata per gli utenti finali, la dimensione del test set è ridotto al minimo per utilizzare nell'allenamento dati aggiornati.

Il test set utilizzato nella simulazione inizia il 1/02/2017 e termina il 1/02/2018, il validation set dal 1/02/2015 al al 1/02/2017, mentre il training dal 1/02/2005 al al 1/02/2015.

7.11 Metodologia: ottimizzazione (Deep Neural Networks)

In questa fase vengono provati nuovi orizzonti temporali, nuove features, nuove regolarizzazioni o nuove architetture per cercare di ottenere performance migliori. Non esiste un metodo standard, o una formula "magica" per trovare il giusto valore di un parametro, se non quello di sperimentare diverse impostazioni. Per esempio, per selezionare il numero di neuroni nascosti ottimali bisogna allenare la rete con diversi valori e valutare la prestazione sul test set. L'incremento nel numero di neuroni provato è libero, non ci sono regole, anche se spesso è vincolato alle risorse computazionali disponibili. Testando la prestazione è, quindi, possibile selezionare l'impostazione che genera l'errore minore. Nonostante questo approccio risulti costoso, per tempo impiegato e per risorse di calcolo, è tra i più utilizzati quando si tratta d'ottimizzazione.

7.12 Metodologia: la rete ottimizzata (Deep Neural Networks)

Come detto in precedenza, le reti neurali sono capaci d'approssimare qualsiasi funzione. Questa flessibilità le rende un ottimo strumento per compiere previsioni ma allo stesso tempo, l'elevato numero di parametri da selezionare, complica il disegno dell'architettura. Nel corso di questa applicazione sono provate diverse architetture di reti neurali, tra cui le *Fully Connected Neural Networks* (FCNNs), le *Recurrent Neural Networks* (RNNs) e le *Convolutional Neural Networks* (CNNs).
Di seguito vengono riportate le caratteristiche principali della FCNN utilizzata per la generazione delle views, la rete che ha ottenuto i risultati migliori tra le architetture sperimentate:

- 3 layers x 32 neuroni
- Weights initialization: He_normal
- Hidden Dropout: 0.6
- Batch Normalization: 1
- Weight Regularizer L2: 0.01
- Optimizer: Adam
- Mini Batch Size: 512
- Learning rate: 0.001
- 100 epoche di training early stopping
- Loss: binary crossentropy

La rete utilizzata nella sperimentazione è composta da 3 layers, ognuno dei quali ha 32 neuroni.

Viene inizializzata attraverso una tecnica chiamata "He_normal" utilizzando dei pesi casuali estratti da una distribuzione gaussiana centrata su zero con deviazione standard $\sqrt{\frac{2}{n° \, unità \, d'input}}$.
In ogni layer vengono spenti il 60% dei neuroni (Hidden Dropout) per evitare problemi d'overfitting.
L'ottimizzatore Adam è una estensione della discesa stocastica del gradiente che recentemente ha trovato molto successo nel settore (Brownlee, 2017) perché è facile da implementare e ottiene ottime prestazioni in problemi particolarmente rumorosi.
Per ottenere una maggior velocità nel training vengono aggiustate e scalate le attivazioni (Batch normalization).
Attraverso l'early stopping la rete, dopo aver valutato 100 epoche, seleziona automaticamente il punto di flessione in cui la performance sul test set inizia a decrescere (mentre la performance sul training continua a migliorare), evitando di overfittare.
Infine la crossentropy loss misura la performance di un modello di classificazione in cui l'ouput è un valore compreso tra 0 e 1.

7.13 Metodologia: criteri di valutazione (Deep Neural Networks)

I modelli predittivi sono valutati attraverso il ROC, una misura di performance, come spiegato nel paragrafo 4.15, più precisa rispetto alla classica accuratezza.
Le metriche riportate in tabella 7.3 fanno riferimento ai dati del training, validation e test e sono intesi come una media di venti allenamenti:

Tabella 7.3. I risultati delle reti

	Accuracy				
	5 days	**10 days**	**20 days**	**40 days**	**60 days**
Training	0.547	0.563	0.581	0.600	0.608
Validation	0.521	0.532	0.536	0.546	0.570
Test	0.510	0.512	0.512	0.521	0.530

	ROC				
	5 days	**10 days**	**20 days**	**40 days**	**60 days**
Training	0.565	0.586	0.615	0.640	0.651
Validation	0.530	0.546	0.556	0.570	0.601
Test	0.512	0.517	0.519	0.535	0.550

Dall'analisi della tabella si evince un pattern significativo: infatti le reti neurali ottengono sistematicamente risultati migliori su orizzonti temporali più lunghi, in qualunque

modello e parte di dati. Questo fenomeno può sembrare contro intuitivo: predire il futuro prossimo non dovrebbe essere più difficile di predire il futuro remoto. Ad ogni modo, se si considera che gli orizzonti più brevi sono affetti da fluttuazione casuali impredicibili, il ragionamento è in linea con i risultati.

7.14 Metodologia: data collection (Black-Litterman)

Per la costruzione del portafoglio di BL vengono presi in considerazione 4 equity indices, che sono assunti essere rappresentativi di un global equity portfolio. I criteri di scelta adottati seguono quelli proposti dal libro CFA "Managing Investment Portfolios" (2007): gli assets di una asset class devono essere omogenei; le asset classes devono essere mutualmente esclusive; le asset classes devono essere il meno correlate possibili per sfruttare il beneficio di diversificazione; le asset classes, come gruppo, devono ricoprire la maggior parte del mondo investibile; le asset classes devono essere liquide.
Gli indici utilizzati sono:

- MSCI Europe (USD), un indice equity rappresentativo delle performance delle medie-grandi società di 15 nazioni europee sviluppate (di cui le più importanti sono UK, Francia, Germania, Svizzera, Olanda, Spagna, Svezia). L'indice copre l'85% della capitalizzazione (aggiustata per il flottante) di ogni paese.
- MSCI Pacific (USD), un indice equity rappresentativo delle performance delle medie-grandi società di 5 mercati sviluppati dell'area del pacifico (Giappone, Australia, Hong Kong, Singapore, Nuova Zelanda). L'indice copre l'85% della capitalizzazione (aggiustata per il flottante) di ogni paese.
- MSCI Emerging Market (USD), un indice equity rappresentativo delle performance delle medie-grandi società di 24 paesi emergenti (di cui le più importanti sono Cina, Corea del Sud, Taiwan, India e Brasile). L'indice copre l'85% della capitalizzazione (aggiustata per il flottante) di ogni paese.
- SPX500 Composite (USD), un indice che segue l'andamento di un paniere azionario formato dalle 500 aziende statunitensi a maggiore capitalizzazione.

Come risk-free viene utilizzata la serie storica del 4-Week Treasury Bill USA che per il periodo preso in considerazione ha oscillato tra 0,50% e 1,27%.
Per ogni indice viene utilizzata una serie storica di rendimenti giornalieri con un orizzonte temporale di tre anni, dal 01-01-2014 al 01-01-2017, ottenuti scaricando dal server di Thomson Reuters Datastream i prezzi giornalieri e calcolando il rendimento semplice $R_t = \frac{P_t}{P_{t-1}} - 1$. Vengono utilizzati i rendimenti semplici e non logaritmici perché su orizzonti temporali brevi la differenza è minima ed essi permettono di calcolare il rendimento di un portafoglio in maniera più semplice. Siccome il portafoglio viene ribilanciato mensilmente, i rendimenti vengono portati a frequenza mensile. Anche se i rendimenti mensili "catturano" meno informazione di quelli giornalieri, in letteratura sono considerati una buona approssimazione.
Infine, i risultati sono annualizzati utilizzando la seguente formula:

$$R_{P,annualizzato} = \prod_{t=1}^{12}\left(1 + R_{P,t}\right) - 1$$

7.15 Metodologia: scelta dei parametri δ, Ω, τ e Assunzioni (Black-Litterman)

Come ricordato nel paragrafo 7.2, a causa della mancanza di precise linea guida sulla implementazione del modello BL, il significato e la calibrazione dei parametri δ, Ω e $\boldsymbol{\tau}$ ha generato grande confusione. Questi parametri, ai fini di questo lavoro, non hanno un impatto molto rilevante, per questo motivo vengono presi fissi, utilizzando valori sperimentati e accettati in letteratura:

- Il coefficiente d'avversione al rischio del mercato δ è calcolato utilizzando i rendimenti e la varianza del portafoglio d'equilibrio $\frac{E(R_m)-r_f}{\sigma^2}$. Nella simulazione viene preso fisso con un valore di 2,5 come suggerito da Black e Litterman.
- τ è interpretato come l'incertezza della stima dei Π, dato una lunghezza temporale T, e quindi $\tau = 1/T$. Siccome le osservazioni mensili sono 36 per asset (3 anni di dati mesnili), il valore preso è di 0,027. E', infatti, logico pensare che più grande è il dataset, minore è il rumore ε e più accurata è la stima dei Π, con la naturale conseguenza che il valore di τ diminuisce. In letteratura sono utilizzati valori compresi nell' intervallo 0.05 e 0.025.
- Ω viene calcolato, come spiegato in precedenza, utilizzando la deviazione standard dell'intervallo del segnale.

La simulazione utilizza quattro assunzioni principali, utili per semplificare la simulazione senza far perdere di significato ai risultati:

- i rendimenti si distribuiscono come una normale;
- non ci sono costi di transazione;
- non viene considerato il rischio di cambio;
- non è possibile vendere allo scoperto.

7.16 Metodologia: indicatori di Performance e benchmark (Black-Litterman)

Minimizzare la funzione d'errore può non essere sufficiente per valutare la capacità delle reti neurali nel generare extra-rendimenti. Avere un errore di previsione basso non necessariamente significa avere un sistema di trading profittevole. Dopo aver trasformato il segnale generato dalle reti neurali in un'azione sui mercati finanziari, è necessario calcolare una misura di rendimento aggiustata per il rischio. In generale, valutare la performance attraverso un rendimento medio non è sufficiente. I rendimenti devono essere aggiustati per il rischio prima di essere comparati. La misura di rendimento aggiustata per il rischio più utilizzata è lo Sharpe Ratio (SR), il rapporto tra il premio al rischio di un determinato intervallo temporale e la deviazione standard dei rendimenti dello stesso periodo:

$$\mathrm{SR_p} = \frac{\mathrm{E(R_p)} - \mathrm{R_f}}{\sigma_\mathrm{p}}$$

Presentata nel 1966 da Sharpe, questa misura rappresenta nella MPT l'inclinazione della capital market line, ed è oggi uno degli indicatori di performance più utilizzati dagli investitori.
Per annualizzare lo SR mensile viene utilizzata la metodologia proposta da Morningstar (2005):

$$SR_A = SR_m\sqrt{12}$$
$$SR_m = \frac{R_m}{\sigma_m}$$
$$R_m = \frac{1}{n}\sum_{i=1}^{n}(R_i - RF_i)$$
$$\sigma_m = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(R_i - RF - \bar{R})^2}$$

Per analizzare la capacità delle reti neurali nel generare extra-rendimenti, è opportuno valutare il rischio e il rendimento in relazione a un benchmark. l'Informatio Ratio (IR) di Treynor e Black (1973), il rapporto tra l'alpha di un portafoglio e il rispettivo rischio non sistematico (Tracking error), misura proprio questo. In formula:

$$IR = \frac{\alpha_P}{\sigma(e_P)}$$

α_P viene definito "alpha", o rendimento attivo, e rappresenta la differenza tra il rendimento (in eccesso al risk free) del portafoglio e il benchmark.
$\sigma(e_P)$ viene definito "tracking error" o rischio attivo e misura la differenza di volatilità tra il portafoglio e il benchmark.

L'alpha viene annualizzato utilizzando la seguente formula:

$$\alpha_A = (1 + \alpha_m)^{12} - 1$$

Siccome l'alpha, tradizionalmente, può essere considerata l'abilità del manager nel generare extra-rendimenti, questa grandezza misura la capacità della rete nel generare rendimenti superiori a quelli del benchamark.
Per calcolare questo indicatore di performance vengono utilizzati il rendimento/rischio residuo, invece che il rendimento/rischio attivo.
Il concetto può essere illustrato osservando la relazione tra il rendimento del portafoglio e il benchmark (single index model):

$$R_P(t) = \alpha_i + \beta_P R_M(t) + e_P(t)$$

Il rendimenti residuo, α_i,e il rischio residuo $e_P(t)$ sono indipendenti dal rendimento del benchmark. Utilizzare i rendimenti residui e il rischio residuo, invece del rendimento attivo e del rischio attivo, permette di valutare la performance delle reti neurali al di fuori del comportamento del benchmark. In altre parole viene misurata la capacità delle

reti in relazione al benchmark, in modo indipendente dai rendimenti generati dal benchmark. Utilizzare il rischio e il rendimento attivo equivarrebbe ad assumere il nostro portafoglio con un beta di 1, cosa difficilmente applicabile in generale (Rasmussen, 2003).
Vengono definiti i seguenti benchmark :

- MSCI World Index, un indice di mercato azionario globale che traccia la performance delle medio- grandi imprese equities di 23 paesi. Rappresenta il mercato azionario mondiale.

- Equally weighted (EW), un portafoglio classico che attribuisce ad ogni asset la stessa ponderazione. Questa strategia, non considerando nessun tipo d'informazione nella costruzione del portafoglio, ha ottenuto i risultati migliori in letteratura (Garlappi e Uppal, 2009).

- Portafoglio di varianza minima (Min-Var), un portafoglio efficiente che minimizza la varianza. Questo viene costruito, come indicato dalla teoria di Markowitz, considerando soltanto la correlazione e la deviazione standard degli assets, senza stimare i rendimenti attesi.

Tutti i risultati vengono annualizzati per facilitare il confronto.

7.17 Lo scenario economico: l'andamento del mercato azionario mondiale

Per una valutazione critica dei risultati è importante analizzare economicamente il periodo storico preso in considerazione. Siccome il periodo out of sample inizia l'1/02/2017 e termina il 31/01/2018, l'analisi economica viene condotta, più in generale, per l'anno 2017.
Nel 2017 il mercato azionario mondiale ha superato le attese prestabilite alla fine del 2016, sostenuto dalla crescita economica mondiale (crescita del 3,7% del Global output nel 2017), da una inflazione contenuta e da prospettive di crescita degli earnings (International Monetary Fund, 2018; Joyce e O'Brien, 2018).
Il mercato equity USA ha continuato a crescere, stimolato dalla riforma fiscale di Trump, proseguendo un percorso positivo di 106 mesi e raggiungendo nuovi massimi, sostenuto in maggior modo dalle Megacap della tecnologia come Amazon, Facebook, Apple.
Oltre i confini americani, la crescita economica mondiale e le politiche monetarie della Banca Centrale Europea hanno favorito l'espansione dei mercati europei. La Banca Centrale Europea ha annunciato che restringerà il suo programma d'acquisto, anche se continuerà a mantenere i tassi bassi fino alla fine del quantitative easing (International Monetary Fund, 2018). La Banca d'Inghilterra ha aumentato i tassi per la prima volta dal 2008. Il clima politico stabilitosi in Europa ha creato fiducia negli investitori, il maggior contributo deriva dalla vittoria alle presidenziali di Macron, sostenitore dell'Euro.
Anche il mercato azionario giapponese ha ottenuto performance positive soprattutto negli ultimi quattro mesi dell'anno, stimolate dai tassi d'interesse negativi e dalle politiche accomodanti della banca centrale del Giappone.

Per quello che riguarda i paesi emergenti, dopo aver lottato negli ultimi anni per rimanere al passo con i paesi sviluppati, nel 2017 hanno ottenuto la performance migliore, grazie in modo particolare alla stabilità del dollaro americano, alla ripresa del commercio e all'aumento del prezzo delle materie prime (Bourbon C., 2018).
La volatilità resta molto bassa, specialmente nei mercati USA. Il Vix Index, indicatore di volatilità del mercato americano, ha raggiunto il minimo storico di 8.54, lo S&P500, per la prima volta dal 1995, non ha raggiunto un drawdown del 3%.
Nella tabella 7.4 sono proposti i rendimenti degli ultimi anni ottenuti dagli indici di Morgan Stanley, benchmark dei mercati azionari USA, Europa, Pacifico e Paesi Emergenti.

Tabella 7.4. I risultati (%) degli indici di Morgan Stanley

	Pacifico	Europa	Paesi Emergenti	USA	Mondo
2017	25,00	26,24	37,28	21,90	23,07
2016	4,46	0,22	11,19	11,61	8,15
2015	3,21	-2,34	-14,92	1,32	-0,32
2014	-2,47	-5,68	-2,19	13,36	5,50
2013	18,43	25,96	-0,26	32,61	27,37
2012	14,60	20,00	18,22	-16,13	16,54
2011	-13,61	-10,50	-18,42	1,99	-5,02
2010	16,08	4,49	18,88	15,45	12,34

7.18 I risultati: il portafoglio d'equilibrio

La prima parte di analisi dei risultati si concentra sulla valutazione del portafoglio d'equilibrio di BL. Questo portafoglio, tradizionalmente, è rappresentato da quello di mercato composto secondo la teoria del CAPM (Capital Asset Pricing Model) nell'ipotesi di mercati in equilibrio. Mentre la MPT suggerisce che il portafoglio di mercato ha il maggior Sharpe Ratio, ci sono situazioni in cui l'allocazione proposta è sub-ottimale. Inoltre, essendoci asset classes che non hanno una capitalizzazione direttamente calcolabile (come per esempio le materie prime), è soggetta a vincoli e approssimazioni. Il metodo proposto nella tesi utilizza un approccio scientifico ("Hierarchical Risk Parity" HRP), piuttosto di una teoria, per valutare la volatilità storica e creare un portafoglio d'equilibrio. Il portafoglio ottenuto viene confrontato con il portafoglio classico di mercato CAPM (ottenuto utilizzando i pesi del Vanguard Total World Market Index 2016) e uno efficiente a varianza minima (Min-Var), due possibili sostituti del portafoglio d'equilibrio. I primi risultati sono ottenuti mantenendo l'allocazione fissa per tutto il periodo out of sample (come se si seguisse una strategia di Buy and Hold) per mettere in risalto le caratteristiche delle strategie di costruzione del portafoglio, a prescindere dai segnali, e capire quale portafoglio è più appropriato a rappresentare il portafoglio d'equilibrio.
Utilizzando i dati storici (tabella 7.5) vengono generate le allocazioni presenti in tabella 7.6.

Tabella 7-5. Matrice di correlazione storica (2014-2017)

2014-2017	Pacifico	Europa	Paesi Emergenti	CAGR (%)	Dev.St (%)
Pacifico				1,21	12,48
Europa	0,75			-5,33	12,80
Paesi Emergenti	0,81	0,78		-5,00	16,30
Usa	0,78	0,77	0,78	7,79	10,83

Tabella 7.6. I pesi dei portafogli e i primi risultati out of sample (CAGR, Dev.St, SR)

Out of sample	Pacifico	Europa	Paesi Emergenti	USA	CAGR (%)	Dev.St (%)	SR
HRP	25,05	25,31	19,84	29,80	29,46	4,60	5,60
CAPM	17,00	21,00	9,00	53,00	28,29	4,69	5,30
Min-Var	10,13	6,88	0,00	82,99	26,53	5,00	4,70

L'algoritmo di HRP, analizzando i cluster della matrice di correlazione, alloca il 29,80% del capitale all'indice USA, il meno rischioso secondo i dati storici; il 25,31% all'indice europeo; il 25,05% all'indice Pacifico e il 19,84% all'indice Paesi Emergenti.
L'allocazione del rischio di questo portafoglio rimane bilanciata anche out of sample (tabella 7.7). Il portafoglio che minimizza la varianza, invece, generando una allocazione molto sbilanciata su pochi assets, ottiene prestazioni buone in sample, ma non out of sample (Prado, 2016*b*). Nella simulazione presa in esame, il rischio è sbilanciato completamente sull'indice USA, il quale contribuisce per 89,93% del rischio totale del portafoglio.

Tabella 7.7. Il contributo al rischio (%)

Contributo al rischio (out of sample)	HRP	CAPM	Min-Var
Pacifico	18,39	12,95	7,00
Europa	27,07	19,13	3,07
Paesi Emergenti	27,03	10,25	0,00
USA	27,51	57,67	89,93

Nonostante l'obiettivo del Min-Var sia quello di minimizzare la varianza, esso è il portafoglio con la volatilità out of sample più elevata. Come illustrato più avanti, è importante sottolineare che in caso di crisi, quando le correlazioni cambiano improvvisamente valore, uno shock idiosincratico penalizza la concentrazione dell'allocazione. Un portafoglio poco diversificato non è buon portafoglio d'equilibrio.
I risultati delle due strategie basate sulla gestione del rischio possono essere in parte spiegati analizzando come le informazioni vengono gestite. Infatti mentre il Min-Var

inverte la matrice di covarianza storica per ottenere i pesi, il HRP trasforma la matrice, creando delle gerarchie e eliminando i legami deboli (tabella 7.8). Per il portafoglio Min-Var tutte le variabili sono interconnesse tra di loro, senza gerarchie, non riconoscendo la complessità dei dati. Parte della differenza dei risultati è spiegato anche da questo, ma siccome una dimostrazione matematica è al di fuori dell'obiettivo della tesi, l'affermazione rimane una supposizione.

Tabella 7.8. Correlazione utilizzata dal portafoglio di HRP vs portafoglio Min-Var

Correlazione HRP	**Paesi Emergenti**	**Europa**	**Pacifico**
Paesi Emergenti			
Europa	0.67		
Pacifico	0.63	0.52	
USA	0.61	0.65	0.57
Correlazione Min-Var	**Paesi Emergenti**	**Europa**	**Pacifico**
Paesi Emergenti			
Europa	0,78		
Pacifico	0,81	0,75	
USA	0,78	0,77	0,78

Continuando il confronto, nonostante il rischio del portafoglio di mercato CAPM sia molto sbilanciato, ottiene risultati simili a quelli del HRP (tabella 7.6), anche se il portafoglio di HRP ottiene i risultati migliori.
I risultati non differiscono molto da portafoglio a portafoglio perché il periodo preso in considerazione è particolarmente positivo per il mercato azionario mondiale e poco volatile.
Per capire come le strategie reagiscono sotto diverse condizioni di mercato, vengono testate in un altro intervallo temporale. Come detto in precedenza, essendo il 2017 un anno estremamente positivo, non è utile a valutare la stabilità dei portafogli in condizioni di stress.
Per questa simulazione viene considerato il periodo storico della bolla delle dot-com e quello successivo fino al 2006. Ho deciso di utilizzare questa crisi finanziaria piuttosto di quella del 2007 perché meno acuta e particolare. Infatti la crisi del 2007, essendo stata la più grande dagli anni '30 dello scorso secolo, non è un buon periodo di prova.
I portafoglio sono costruiti con i dati tra il 1997-1999, il periodo in cui si è sviluppato la bolla, e testati tra il 2000-2006, il periodo della crisi (fino al 2002) e della ripresa economica. Viene utilizzato, soltanto per questa simulazione, un nuovo indicatore di performance, il Sortino Ratio. Questo indicatore utilizza una misura di rischio che tiene in considerazione solamente la parte negativa della volatilità.
Osservando i risultati (tabella 7.9), il portafoglio di HRP performa leggermente meglio dei benchmark ottenendo valori maggiori di CAGR, SR e Sortino Ratio. Il contributo al rischio rimane equilibrato anche se la deviazione standard è maggiore. I risultati sono

simili perché tutti e tre i portafogli sono esposti alla stessa fonte di rischio principale. La deviazioni standard e il drawdown massimo sono pressoché uguali. Questo dimostra che, in caso di crisi, la concentrazione su una sola asset class, anche se diversificata geograficamente, è pericolosa.

Tabella 7.9. I risultati dei portafogli equities per il periodo 2000-2006 (in parentesi il contributo al rischio)

2000-2006	Pacifico	Europa	Paesi Emergenti	USA	CAGR (%)	Dev.St (%)	Max Drawdown (%)	SR	Sortino Ratio
HRP	22.77 (19,81%)	33.45 (33,66%)	15.50 (20,53%)	28.28 (26%)	5.04	15.01	-45,03	0,20	0,28
CAPM	19.09 (16,47%)	28.02 (28,53%)	11.14 (14,84%)	41.75 (40,11%)	4.34	14.71	-44,90	0,16	0,22
Min-Var	6.47 (4,61%)	69.72 (74,19%)	0,00	23.81 (21,20%)	4.56	14.98	-45,57	0,17	0,24

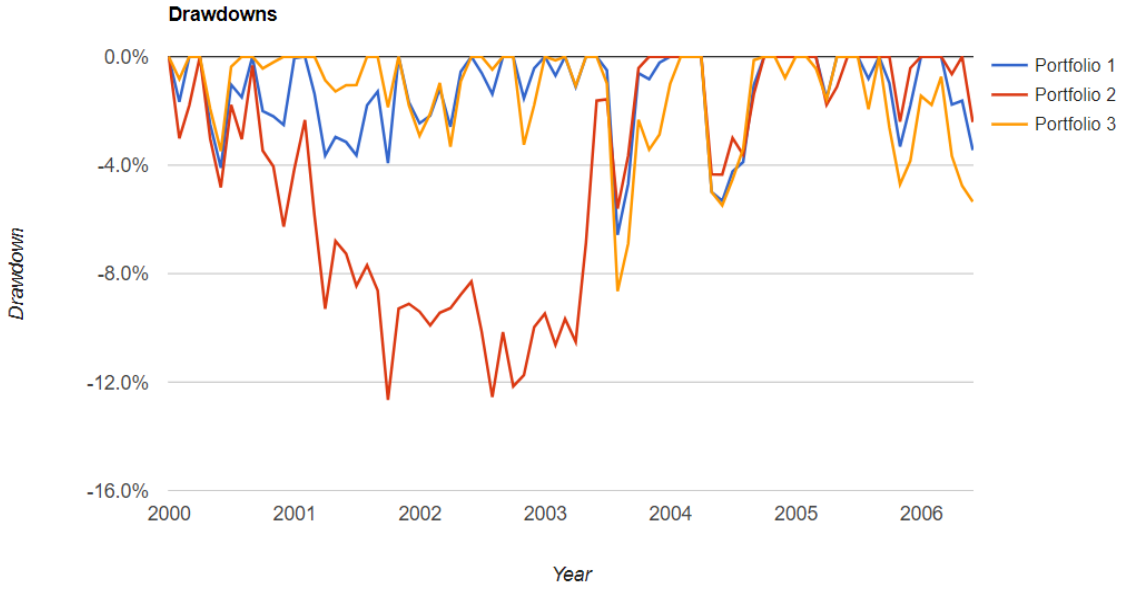
Viene, quindi, simulato per lo stesso periodo un portafoglio composto da più asset classes, equity e bond (tabella 7.10). Vengono aggiunti un indice rappresentativo del mercato mondiale corporate investment grade e uno dei titoli di stato a lungo termine (mondiale). Per il portafoglio CAPM vengono utilizzate le proporzioni tra bond e equity proposte nel paper "Historical Returns of the Market Portfolio" di Doeswijk, Lam e Swinkels, mentre l'allocazione geografica dell'equities segue quella proposta dal Prof. Siegel J. (2005).
Come previsto, i risultati migliorano molto; vengono ridotte in modo particolare le deviazioni standard (del 50%) e i drawdown (tra il 17% e il 40%) di tutti e tre i portafogli.

Tabella 7.10. I risultati dei portafogli multi-asset per il periodo 2000-2006 (in parentesi il contributo al rischio)

2000-2006	Pacifico	Europa	Paesi Emergenti	USA	treasury long term	corporate invest. grade	CAGR (%)	Dev.St (%)	Max Drawdown (%)	SR	Sortino Ratio
HRP	9,24 (13,43%)	12,44 (15,54%)	6,69 (12,63%)	9,36 (10,78%)	32,75 (24,00%)	29,52 (23,00%)	7,22	7,00	-7,49	0,56	0,83
CAPM	11,00 (16,12%)	17,00 (24,44%)	6,00 (12,35%)	26,00 (36,44%)	25,00 (5,68%)	15,00 (4,97%)	6,91	8,63	-18,85	0,45	0,64
Min-Var	6,66 (4,03%)	9,62 (3,81%)	0,00	0,00	0,00	83,72 (92,16%)	6,41	7,15	-8,21	0,46	0,68

L'aspetto più significativo è dato dal drawdown massimo, che raggiunge nel portafoglio CAPM un valore negativo doppio rispetto agli altri due (tabella 7.10).
In figura 7.4 sono rappresentati tutti i drawdown del periodo 2000-2006. Mentre il portafoglio che minimizza la varianza e quello di HRP seguono un andamento simile, il portafoglio CAPM è molto meno stabile.

Figura 7.4. Drawdown (periodo 2000-2006), in azzurro il portafoglio HRP, in giallo quello di Min-Var e in rosso il CAPM



E' da sottolineare la capacità del portafoglio di HRP di resistere agli shock nei periodi di turbolenza e allo stesso tempo di ottenere ottimi risultati nei periodi di espansione; dividendo il periodo della simulazione in due (tabella 7.11 e 7.12) è possibile vedere come il portafoglio durante la crisi (2000-2002) ottiene risultati (Dev.St e Max Drawdown) in linea con il portafoglio che minimizza la varianza, mentre nella fase di recupero/espansione (2002-2006) risultati superiori. Considerando che il portafoglio che minimizza la varianza è composto prevalentemente da bond (83,72%), ottenere tali risultati nel primo periodo è indice di stabilità.

Tabella 7.11. I risultati dei portafogli multi-asset per il periodo 2000-2002 (in parentesi il contributo al rischio)

2000-2002	CAGR (%)	Dev.St (%)	Max drawdown (%)	SR	Sortino Ratio
HRP	3,52	6,60	-4,42	0,00	-0,01
CAPM	-2,34	8,31	-13,36	-0,68	-0,83
Min-Var	7,77	6,52	-3,48	0,61	1,00

Tabella 7.12. I risultati dei portafogli multi-asset per il periodo 2002-2006 (in parentesi il contributo al rischio

2002-2006	CAGR (%)	De.St (%)	Max drawdown (%)	SR	Sortino Ratio
HRP	10,63	6,82	-5,20	1,18	1,96
CAPM	11,03	7,51	-8,69	1,13	1,86
Min-Var	8,68	7,59	-7,75	0,83	1,27

In figura 7.5 e 7.6, nell'intervallo zero e uno, viene delineata l'evoluzione nel tempo della composizione del portafoglio ribilanciato, mentre, nell'intervallo uno e due, la composizione di base del portafoglio. Questa rappresentazione permette di vedere, mese per mese, come i segnali generati hanno cambiato l'allocazione del portafoglio.
Siccome i due portafogli a confronto hanno una composizione simile e utilizzano gli stessi segnali, anche l'allocazione nel tempo segue un andamento comune, a parte per il periodo settembre 2017 in cui vengono utilizzate le allocazioni di partenza perché le reti neurali non hanno generato segnali.

Figura 7.5. L'andamento nel tempo dei pesi del portafoglio di BL (HRP)

.

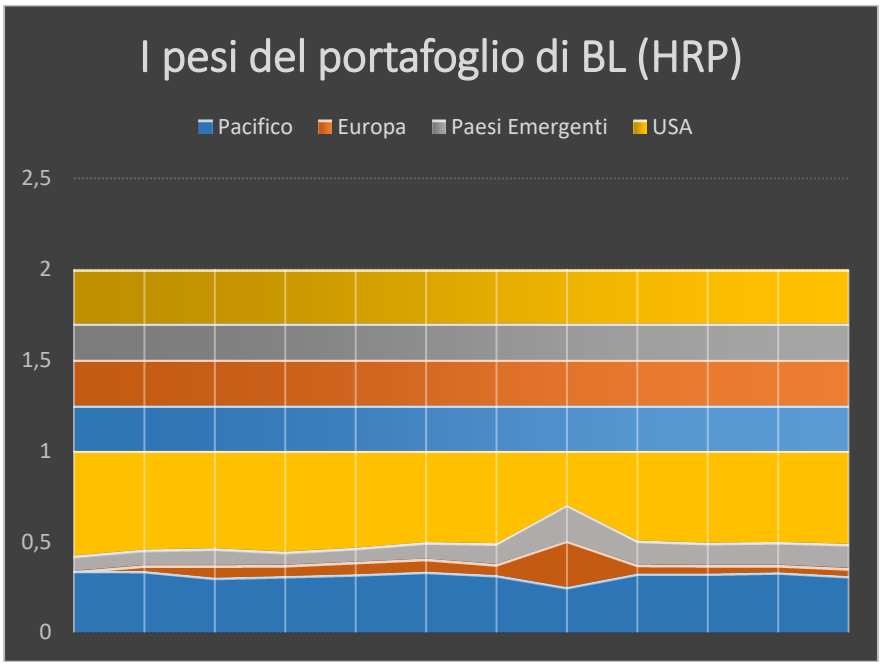


Figura 7.6. L'andamento nel tempo dei pesi del portafoglio di BL (CAPM)

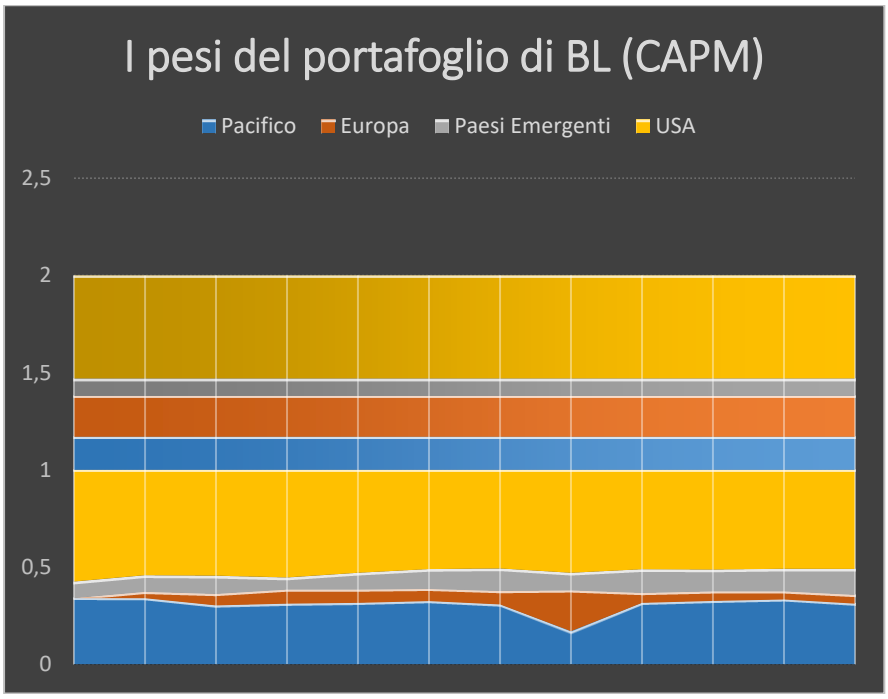


Tabella 7.13. Le statistiche sulla composizione del portafoglio di BL (HRP)

Pesi del portafoglio di BL (HRP)	Media	Dev. St	Max	Min
Pacifico	31,67	2,28	33,97	25,05
Europa	6,76	5,80	25,31	0,00
Paesi Emergenti	11,01	3,41	19,84	7,24
USA	50,54	6,63	57,62	2,98

Tabella 7.14. Le statistiche sulla composizione del portafoglio di BL (CAPM)

Pesi del portafoglio di BL (CAPM)	Media	Dev. St	Max	Min
Pacifico	30,77	4,50	34,02	17
Europa	6,36	5,00	21,00	0,00
Paesi Emergenti	10,04	1,98	13,31	6,29
USA	52,82	2,16	57,47	50,79

Osservando, invece, le statistiche di tabella 7.13 e 7.14, l'allocazione del portafoglio di CAPM è meno volatile. Parte di questo comportamento è spiegato dal fatto che l'allocazione di base del CAPM è più simile alla posizione media del portafoglio ribilanciato. Entrambi i portafogli, nonostante utilizzino input diversi, hanno una allocazione stabile. Siccome il ribilanciamento dipende dalla confidenza del segnale, un'allocazione stabile è la conseguenza di confidenze appropiate. Utilizzare le confidenze generate dalle reti neurali, inoltre, permette di non utilizzare il metodo

classico proposto da Black e Litterman basato sull'assunzione che la varianza della distribuzione a priori sia proporzionale a quella delle views (Walters, 2007).
L'allocazione dell'indice europeo è la più volatile a causa di un minimo di zero ottenuto nel primo periodo di ribilanciamento; probabilmente le reti hanno intercettato il clima d'incertezza che vigeva sui mercati europei nei primi mesi del 2017, derivante soprattutto dalle elezioni francesi. Tra marzo e aprile 2017, infatti, la volatilità dell'indice Eurostoxx ha raggiunto il livello massimo dalla Brexit del 2016. Siccome le reti utilizzano tutta l'informazione a loro disposizione, anche le predizioni a più lungo periodo, è possibile che abbiano deciso di rimanere fuori dal mercato europeo, per poi allocare poco alla volta il capitale.
Nel settembre 2017, invece, le reti neurali non generano nessun segnale. I segnali non hanno superato il cutoff del 1% impostato, quindi la rete non ha suggerito nessuna allocazione. Questo lascia pensare che le informazioni estratte dai mercati finanziari fossero "neutrali", ma è soltanto un ipotesi difficile da confermare.
Analizzando le misure di performance, il portafoglio di BL (HPR) raggiunge risultati migliori (tabella 7.15). Infatti, oltre ad ottenere uno SR superiore, vince il confronto anche nelle misure di performance attive. L'Alpha generato rispetto il benchmark MSCI World e l'IR sono superiori, inoltre il portafoglio di BL (CAPM) è più correlato con l'andamento dei mercati azionari mondiali (Beta 1,048 con t-test 11,20).

Tabella 7.15. I risultati: portafoglio di BL (HRP) vs portafoglio di BL (CAPM). In parentesi la statistica t

	Alpha (%)	IR	SR	Beta	ES (%)	CAGR (%)	Dev.St (%)
BL (HRP)	4,10 (1,46)	2,82	5,86	0,98 (9,78)	1,45	30,57	4,6
BL (CAPM)	1,10 (0,43)	0,81	4,76	1,048 (11,2)	1,35	28,60	4,8

In conclusione, creare un portafoglio d'equilibrio utilizzando l'approccio di HRP può portare a benefici. Anche in diverse condizioni di mercato, questo portafoglio ottiene risultati migliori di un portafoglio che minimizza la varianza o di un CAPM. Come dimostrato, l'allocazione ottenuta è meno concentrata e permette di ottenere buone prestazioni anche in diverse fasi dei mercati. La teoria suggerita da Prado permette, inoltre, di eliminare dei vincoli poiché, non considerando le capitalizzazioni di mercato, consente di utilizzare più asset classes.
Infine, utilizzare un metodo di costruzione del portafoglio che si concentra sul rischio può essere un buon punto d'inizio in assenza di views sui rendimenti attesi. Infatti quando un investitore non ha idea dell'andamento futuro dei mercati finanziari la cosa più intelligente che può fare è diversificare il rischio.

7.19 Risultati: il confronto con i benchmark HRP, Min-Var, EW, MSCI World

La seconda parte dell'analisi si focalizza sul confronto con i benchmark. L'obiettivo è capire se le reti neurali riescono a generare valore in un sistema di (tactical) asset allocation.
Siccome il portafoglio di BL (HRP) domina quello di BL (CAPM), l'analisi si concentra soltanto sul primo portafoglio.
Il primo confronto prende in considerazione il portafoglio di BL con l'allocazione di base per capire se i ribilanciamenti hanno generato profitti. Osservando la tabella 7.16 è possibile notare che il portafoglio di BL, rispetto all'allocazione di base (il portafoglio d'equilibrio di HRP), ottiene extraprofitti 7 volte su 12 (periodi). Investendo in entrambi i portafogli 10.000 dollari, il montante del portafoglio ribilanciato diventa di 13.057,41 (CAGR del 30,57%) contro i 12.946,734 (29,46%), per un extraprofitto di 110,67 dollari (1,10%).

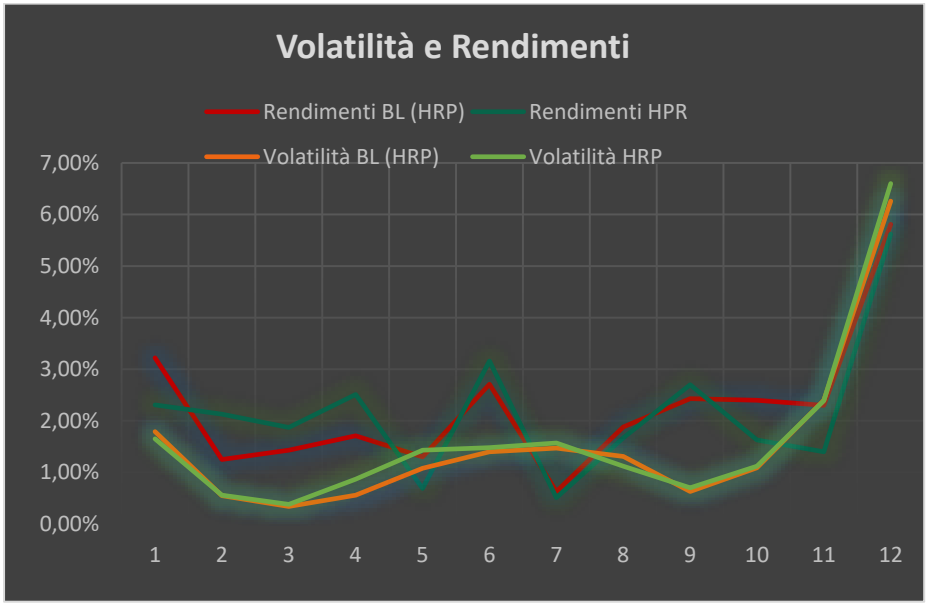
Tabella 7.16. SR e Extrarendimenti generati dal ribilanciamento mensile

	feb-17	**mar-17**	**apr-17**	**mag-17**	**giu-17**	**lug-17**	**ago-17**	**set-17**	**ott-17**	**nov-17**	**dic-17**	**gen-18**
SR BL	**1,80**	2,27	4,21	**3,05**	**1,20**	1,94	**0,43**	1,44	3,86	**2,20**	**0,96**	**0,93**
SR HRP	1,40	3,80	4,92	2,89	0,48	2,14	0,32	1,49	3,86	1,46	0,59	0,85
Rendimenti BL (%)	3,22	1,25	1,43	1,71	1,30	2,71	0,63	1,88	2,43	2,40	2,30	5,81
Rendimenti HRP (%)	2,31	2,13	1,87	2,51	0,69	3,16	0,51	1,67	2,70	1,63	1,40	5,62
Extrarendiementi	**0,91**	-0,88	-0,44	-0,80	**0,61**	-0,45	**0,12**	**0,21**	-0,27	**0,77**	**0,90**	**0,19**

Anche considerando una misura di rendimento aggiustato per il rischio i risultati non cambiano, lo SR è più alto 7 volte su 12. Lo SR annualizzato del portafoglio di BL è 5,86 contro 5,67 di quello di HRP. Analizzando le misure di performance attive, l'IR annuale del portafoglio di BL è 1,8 mentre l'Alpha è di 3,9% (t-test 0,87). Questi valori confermano quelli trovati in precedenza, il portafoglio di BL riesce a ottenere profitti extra (modesti) rispetto al portafoglio non ribilanciato. Entrambi i portafogli sono molto correlati con l'andamento del mercato equity mondiale, BL ha un Beta rispetto l'indice MSCI World di 0,98 (t-test 9,8) mentre HRP di 0,96 (t-test 8,3).
Nonostante sia difficile interpretare i segnali delle reti è possibile individuare un pattern utile per interpretare i risultati economici, dal momento che il modello ottiene risultati inferiori al benchmark nei periodi di bassa volatilità. Come è possibile vedere in figura 7.7, quando la volatilità è ai minimi (periodi 2, 3, 4 e 9), il modello ottiene risultati inferiori al portafoglio benchmark. Questo sembra sottolineare una buona capacità delle reti di estrarre informazioni utili nei periodi di movimento dei mercati.
Procedendo con l'analisi, dal confronto con il portafoglio che minimizza la varianza si attendono degli extra-rendimenti maggiori; il portafoglio benchmark, utilizzando una strategia di costruzione di portafoglio più conservativa, dovrebbe ottenere risultati minori in un periodo d'espansione dei mercati. Lo SR del benchmark è 4,77, minore di quello del portafoglio di BL (5,86). L'IR è di 4,6 mentre l'Alpha è di 5,9% (t-test 2,55), valori molto maggiori di quello ottenuti rispetto al primo benchmark.

Tra tutti i benchmark, è quello che ottiene i risultati peggiori. Questo potrebbe essere spiegato, come detto in precedenza, anche da come la strategia analizza le informazioni per la costruzione del portafoglio.

Figura 7.7. Volatilita e Rendimenti a confronto: BL (HRP) vs HRP



Il benchmark Equally Weighted, invece, assegna lo stesso peso ad ogni asset del portafoglio, non valutando nessun tipo d'informazione. Questa strategia, in letteratura, ha ottenuto i risultati migliori (Garlappi e Uppal, 2009).
Nella simulazione considerata, il portafoglio benchmark EW risulta essere il più volatile, in modo particolare dal periodo 4 al periodo 8. I rendimenti maggiori sono ottenuti nei mesi più "tranquilli" (periodi 2, 3, 4) (figura 7.8). Questo risultato è spiegabile considerando la logica con cui è costruito il portafoglio: a differenza del portafoglio a Min-Var che, essendo costruito per contenere il rischio, ottiene i rendimenti più bassi, il portafoglio EW, non essendo influenzato da nessun tipo d'informazione, ottiene i rendimenti più alti del periodo. Il portafoglio di BL, nello stesso periodo, è il più moderato, di fatto ribilanciando ogni mese e aggiornando l'informazione a disposizione, risulta essere il meno volatile e con rendimenti intermedi tra i due benchmark.
Utilizzando gli indicatori di performance, lo SR del portafoglio benchmark è 5,77 mentre IR è 2,07. L'Alpha ottenuto è 5,4% (t-test 1,050). Il Beta del Benchmark rispetto al MSCI World è 0,96 (t-test 6,7). La deviazione standard dei residui è 2,60%, la più alta tra i portafogli analizzati. Quest'ultimo risultato conferma la buona prestazione del portafoglio EW, infatti il portafoglio di BL per ottenere extraprofitti ha aumentato la componente di rischio attivo di più rispetto agli altri benchamrk.
Infine, il portafoglio di BL ottiene risultati migliori anche rispetto all'indice MSCI World rappresentativo del mercato azionario mondiale. Lo SR dell'indice è 5,22, l'IR rispetto all'indice è 2,82 e l'Alpha 4,1% (t-test 1,46). In tabella 7.17 è proposto un riassunto dei risultati ottenuti dal portafoglio di BL (HRP) rispetto ai Benchmark (Alpha, IR, ES, Beta) e il risultato dei Benchmark (SR, CAGR).

Figura 7.8 Volatilita e Rendimenti a confronto

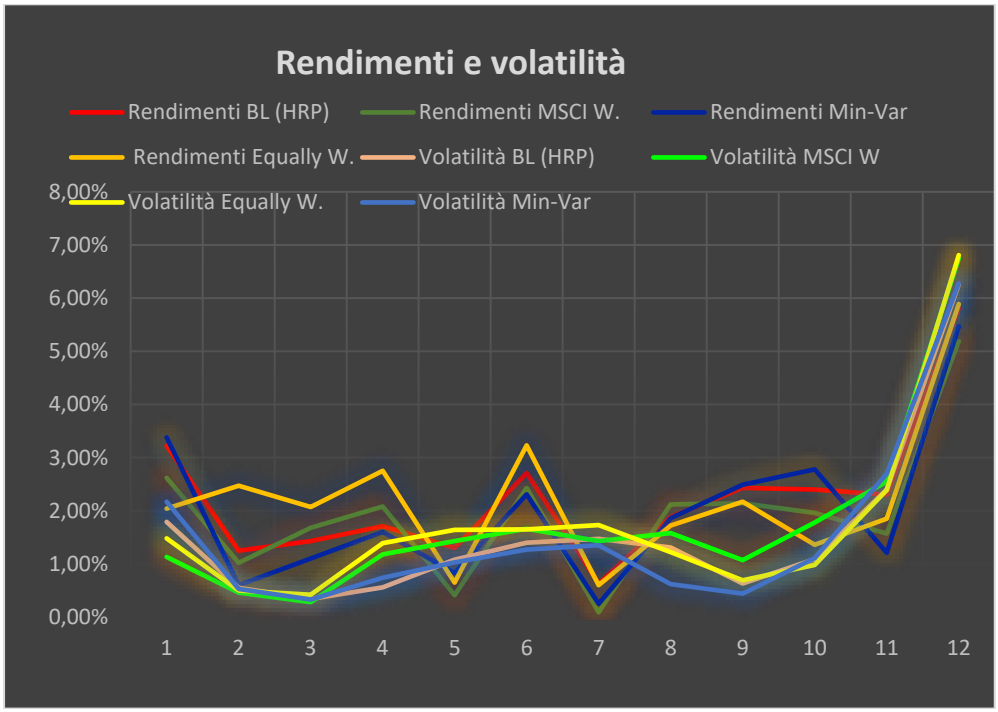


Tabella 7.17. Il rissunto dei risultati: i valori ottenuti da BL(HRP) rispetto ai Benchmark (Alpha, IR, ES, Beta) e i risultati dei Benchmark (SR, CAGR). In parentesi la statistica t.

	Alpha (%)	IR	ES (%)	Beta	SR	CAGR (%)
MSCI World	4,10 (1,46)	2,82	1,45	0,98 (9,78)	5,22	25,85
EW	5,40 (1,05)	2,07	2,60	0,80 (4,90)	5,77	30,20
Min-Var	5,90 (2,55)	4,60	1,28	0,88 (11,30)	4,77	26,53
HRP	3,90 (0,86)	1,80	1,85	0,88 (5,90)	5,67	29,46

7.20 Conclusioni

I risultati hanno raggiunto tre dei quattro obiettivi prefissati (alpha, stabilità del portafoglio d'equilibrio, flessibilità, decisioni razionali). E' importante tenere in considerazione che l'analisi è basata su dati relativi a soli 12 mesi, un periodo troppo breve per garantire la significatività dei risultati; inoltre, non vengono tenuti in considerazione i costi di transazione e il rischio di cambio. Il sistema di *tactical asset allocation* ha ottenuto extraprofitti rispetto tutti i benchmark, confermando che le reti neurali possono contribuire alla generazione di valore. Dalla simulazione è emerso che i risultati migliori sono ottenuti nei periodi più volatili, ma va ancora una volta rimarcato che essi si riferiscono a un periodo temporale particolare, caratterizzato da un andamento particolarmente positivo per il mercato azionario mondiale e con una volatilità ai minimi. Sarebbe stato utile valutare la capacità delle reti anche in un differente periodo temporale, per vedere come cambia il valore generato sotto diverse

condizioni, ma in questo caso non sarebbe stato possibile ottenere risultati out of sample.
Il sistema ideato permette di risolvere uno dei punti più controversi del modello di BL, la determinazione della confidenza delle views. Invece di utilizzare l'assunzione proposta da BL, in cui la varianza della distribuzione a priori viene posta proporzionale a quella delle views, le confidenze vengono stimate direttamente dai dati. Questo può portare a un significativo miglioramento del modello poichè i ribilanciamenti sono una conseguenza di tali valori.
Il portafoglio d'equilibrio costruito seguendo la teoria della HRP ha battuto i benchmark. Questo tipo di strategia, oltre ad ottenere maggiori rendimenti corretti per il rischio, è risultato più stabile sotto diverse condizioni di mercato. Questa caratteristica, in particolare, lo rende un buon portafoglio d'equilibrio in quanto, rappresentando quel portafoglio che l'investitore detiene quando non ha informazioni sul futuro andamento dei mercati finanziari, è necessario che sia pronto a ogni scenario. Inoltre, non utilizzando le capitalizzazioni di mercato come il modello originale, è possibile utilizzare qualsiasi asset classes e creare portafogli diversificati.
Il problema principale del sistema è di riuscire a interpretare i segnali delle reti neurali. Un asset manager, gestendo il capitale di altre persone, ha il dovere di motivare le proprie decisioni. Non capire come un segnale è stato generato ne limita molto l'uso. Questo problema ha in parte origine dall'approccio utilizzato dagli ingegneri che pone l'attenzione (quasi esclusivamente) sulla performance e l'ottimizzazione.
Per trasformare il "lavoro ingegneristico-informatico" in valore economico bisogna creare una strategia che faccia leva sulle determinanti che hanno generato il segnale, perciò avere conoscenza di come il segnale è stato generato. E' in questa circostanza che i due approcci disciplinari devono interagire, comprendersi e avere obbiettivi comuni. C'è la necessità di sviluppare figure professionali interdisciplinari che uniscano soft skills ad hard skills e che permettano ai due mondi di comunicare. Senza questo collegamento la conoscenza non può progredire.
Nonostante questi limiti, ritengo che le reti neurali sono uno strumento utile per integrare le conoscenze di un manager perché sono capaci di valutare moltissima informazione in tempo reale senza farsi influenzare dalle emozioni e dai pregiudizi.

Bibliografia

Abedini R., Estandyari M., Nezhadmoghadam A., Rahmanian B.

2012 *The prediction of undersaturated crude oil viscosity: An artificial neural network and fuzzy model approach*, Petroleum Science and Technology

Adebiyi A.A., Adewumi A., Ayo K.,

2014 *Comparison of ARIMA and Artificial Neural Networks Models for Stock Price Prediction*,

Journal of Applied Mathematics

Angelini R.

2017 *Intelligenza artificiale e governance. Alcune riflessioni di sistema*, Astrid Rassegna

Anish C., Majhi B.

2016 *An ensemble model for Net asset value prediction*, IEEE, Marzo

Asness C., Frazzini A., Pedersen L.

2013 *Quality Minus Junk*, SSrn

Banz R.

1981 *The relationship between Return an Market Value of common stocks*, Journal of Financial Economics, Marzo

Basu S.

1983 *The investment performance of common stocks in relation to their price earning ratios: a test of efficient market hypothesis*, Journal of Finance, Giugno

Bellman R.

1978 *An Introduction to Artificial Intelligence: Can Computer Think?*, Boyd & Fraser

Bengio Y., Simard P., Frasconi P.

1994 *Learning long-term dependencies with gradient descent is difficult,* IEEE Transactions on Neural Networks

Best M.J., Grauer R.R.

1991 *On the Sensitivity of Mean-Variance-Efficient, Portfolios to Changes in Asset Means: Some Analytical and Computational Results,* The Review of Financial Studies, Gennaio

Bevan A., Winkelmann K.

1998 *Using the Black-Litterman Global Asset Allocation Model: Three Years of Practical Experience*, Goldman Sachs Fixed Income Research paper, Giugno

Black F., Litterman R.

1990 *Asset Allocation: Combining Investor Views with Market Equilibrium*, Goldman Sachs Fixed Income Research Note, Settembre

Black F., Litterman R.

1991 *Global Asset Allocation with Equities, Bonds and Currencies*, Goldman Sachs Fixed Income Research Note, Ottobre

Black F., Litterman R.

1991*a* *Global Portfolio Optimization*, Journal of Fixed Income

Black F., Litterman R.

1992 *Global Portfolio Optimization*, Financial Analysts Journal, Settembre

Black F., Litterman R.

1999 *The Intuition Behind Black-Litterman Model Portfolios*, Goldman Sachs Asset Management Working paper

Blamont D., Firoozye N.

2003 *Bayesian Asset Allocation: Black Litterman*, Global Markets Research Deutsche Bank, Dicembre

Bodie Z., Kane A.

2013 *Investments and Portfolio Management*, McGraw Hill, Dicembre

Bogle J.

2002 *An Index Fund Fundamentalist*, The journal of Portfolio Management

Bollen J., Maoa H., Zengb X.

1990 *Stock price pattern recognition-a recurrent neural network approach*, Neural Networks

Bojarski M., Del Testa D., Dworakowski D.

2016 *End to End Learning for Self Driving Cars*, ArXiv, Aprile

Boston Consulting Group

2017 *Global asset management 2017 the innovator's advantage*, Boston Consulting Group, Luglio

Bourbon C.

2018 *2017 Stock Market returns Ring the Bell of Caution*, Morningstar, Gennaio

Brock W. A., De Lima P. J. F.

1995 *Nonlinear time series, complexity theory and finance*, Elsevier

Brownlee J.

2017 *Gentle Introduction to the Adam Optimization Algorithm for Deep Learning*, Machine Learning Mastery, https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/, Marzo

Capital Group

2017 *Artificial Intelligence: it's not the future, it's now*, Capital Group, Maggio

Carhart M.

1997 *On persistence in Mutual Fund Performance*, The Journal of Finance

Chen K., Zhou Y., Dai F.

2015 *A LSTM-based method for stock returns prediction: A case study of China stock market*, IEEE, Dicembre

Chen N., Roll R., Ross A.

1986 *Economic Forces and the Stock Market*, The Journal of Business, Luglio

Mingyue Q., Cheng L., Yu S.

2016 *Application of the Artifical Neural Network in predicting the direction of stock market index*, 10th International Conference on Complex, Intelligent, and Software Intensive Systems

Cheung W.

2009 *The Black-Litterman Model Explained*, Journal of Asset Management, Febbraio

Comrie A. C.

1997 *Comparing Neural Networks and Regression Models for Ozone Forecasting*, Journal of the Air & Waste Management Association, Giugno

Coplin D.

2016 Microsoft exec: *'AI is the most important technology that anybody on the planet is working on today*', Business Insider, http://www.uk.businessinsider.com/microsoft-exec-ai-is-the-most-important-technology-that-anybody-on-the-planet-is-working-on-today-2016-5?IR=T

Chincarini L.

2010 *A Comparison of Quantitative and Qualitative Hedge Funds*, SSRN, Maggio

Cho K., Van Merrienboer B., Gulcehre C., Bahdanau D., Bougares F., Schwenk H., Bengio Y.

2014 *Learning phrase representations using rnn encoder-decoder for statistical machine translation*, Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, Ottobre

Chopra K., Ziemva T.

1993 *The Effect of Errors in Means, Variances, and Covariances on Optimal Portfolio Choice*, Journal of Portfolio Management

Olah C.

2015 *Understanding LSMT networks*, Colah's Blog, http://colah.github.io/posts/2015-08-Understanding-LSTMs/, Agosto

Crosman P.

2017 *Beyond Robo-Advisers: How AI Could Rewire Wealth Management*, American Banker, https://www.americanbanker.com/news/beyond-robo-advisers-how-ai-could-rewire-wealth-management, Gennaio

Deloitte

2016 *The expansion of robot advisory in wealth management*, Deloitte, Agosto

DeMiguel V., Garlappi L., Uppal R.

2009 *Optimal Versus Naive Diversification: How Inefficient is the 1/N Portfolio Strategy?*, Review of Financial Studies

Deng L., Yu D.

2014 *Deep learning: Methods and applications. Found Trends Signal Process*, Now Publisher,

Deutsche Bank

2014 *Big data. How it can become a differentiator*, Deutsche Bank

Desai V., Bharati R.

1998 *The Efficacy of Neural Networks in Predicting Returns on Stock and Bond Indices*, Decision Science

Doeswijk R., Lam T., Swinkels L.

2017 *Historical Returns of the Market Portfolio*, SSRN

Ding X., Zhang Y., Liu T., Duan J.

2014 *Using Structured Events to Predict Stock Price Movement: An Empirical Investigation*, EMNLP, Ottobre

Ding X., Zhang Y., Liu T., Duan J.

2015 *Deep Learning for Event-Driven Stock Prediction*, IJCAI

Di Persio L., Honchar O.

2016 *Artificial neural networks approach to the forecast of stock market price movements*, International Journal of Economics and Management Systems

Domingos P.

2015 *The Master Algorithm: How the Quest for the Ultimate Learning Machine Will Remake Our World,* Hardcover, Basic Book, Settembre

Dreyfus H. L.

1972 *What computers can't do*, Harper & Row

Elton E., Gruber M., Blake C.

1996 *The Persistence of Risk-Adjusted Mutual Fund Performance*, The Journal of Business

EY

2014 *Big Data. Changing the way business compete and operate*, EY, Aprile

Enke D., Thawornwong S.

2005 *The use of data mining and neural networks for forecasting stock market returns*, Expert Systems with application

Eurekahedge

2017 *Artificial intelligence: The new frontier for hedge funds*, Eurekahedge

http://www.eurekahedge.com/Research/News/1614/Artificial-Intelligence-AI-Hedge-Fund-Index-Strategy-Profile, Gennaio

Fawcett T., Hardin D.

2017 *Machine Learning vs. Statistics*, Silicon Valley Data Science, https://svds.com/machine-learning-vs-statistics/

Fama E., French K.

1996 *Multifactor portfolio efficiency and mulktifactor asset pricing*, Journal of Financial and Quantitative Analysis

Fei-Fei L.

2017*a* *Convolutional Neural Networks for Visual Recognition*, Stanford CS231n

Fei-Fei L.

2017*b* *Recurrent Neural Networks*, Stanford CS231

Fernandez A., Gomez S.

2007 *Portfolio selection using neural networks*, ScienceDirect

Freitas F. D. D., Souza A. F. D., Gomez F. J. N., Almeida, A. R. D.

2001 *Portfolio Selection with Predicted Returns Using Neural Networks*, IASTED International Conference on Artificial Intelligence and Applications

Francis J., Kim D.,

2013 *Modern Portfolio theory*, Wiley, Gennaio

FSB

2017 *Artificial intelligence and machine learning in financial services Market developments and financial stability implications*, FSB, Novembre

Goldberg Y.

2016 *A Primer on Neural Network Models for Natural Language Processing*, Journal of Artificial

Intelligence

Goodfellow I., Bengio Y.

2016 *Deep Learning*, M.I.T. Press, Novembre

Graham B.

1934 *Security Analysis*, Mcgraw Hill

Graham B.,

1949 *The intelligence Investor* , Harper

Grinblatt D., Wemers R., Titman S.

1997 *Measuring mutual fund performance with characteristic based benchmarks*, Journal of Finance

Haesen D., Hallerbach W.G., Markwat T., Molenaar R.

2014 *Enhancing Risk Parity by Including Views*, The Journal of Investing", Agosto

Hall J.

1994 *Adaptive Selection of US Stocks with Neural Networks*, Trading on the Edge

Harvey C., Liu Y., Zhu H.

2016 *… and the cross-section of expected returns*, Review of Financial Studies, Gennaio

Haykin S.

2009 *Neural Networks and Learning Machines*, Pearson Education Inc

Hastie T., Tibshirani R., Friedman J.

2017 *The Elements of Statistical Learning. Data Mining, Inference, and Prediction*, Springer, Gennaio

Heaton J., Polson N., Witte J.

2016 *Deep Portfolio Theory*, ArXiv

Herold U.

2003 *Portfolio Construction with Qualitative Forecasts*, Journal of Portfolio Management

Hill T., O'Connor M., Remus W.

1996 *Neural network models for time series forecasts*, Management Science

Hinton G., Osidero S., Teh Y.

2006 *A fast learning algorithm for deep belief nets*, Neural Computation, Luglio

Hochreiter J., Schmidhuber S.

1997 *Long Short Term Memory*, Neural Computation

Hsieh T.J.

2011 *Forecasting stock markets using wavelet transforms and recurrent neural networks: An integrated system based on artificial bee colony algorithm*, Applied Soft Computing, Marzo

Ibbotson Associates

2016 *Strategic Asset Allocation and Commodities*. Commissioned by PIMCO, Marzo

Idzorek T.

2005 *A Step-By-Step guide to the Black-Litterman Model*, Incorporating UserSpecified Confidence Levels

IBA Global Employment Institute

2017 *Artificial Intelligence and Robotics and Their Impact on the Workplace*, IBA GEI-International Bar Association Global Employment Institute, Aprile

International Data Corporation

2016 *Worldwide Semiannual Big Data and Analytics Spending Guide*, International Data Corporation

International Monetary Fund

2018 *World Economic Outlook*, International Monetary Fund, Gennaio

Jang G., Lai F.

1994 *Intelligent Trading of an Emerging market*, Trading on the Edge

Joyce B., O'Brien A.

2018 *Global equity markets produced strong gains in 2017*, Glc Asset Management, https://www.glc-amgroup.com/news-insights/market-reviews/2017-market-review--stocks--bonds-and-the-economy-exceed-expecta.html

J. P. Morgan

2010 *Impact Investments. An emerging asset class*, Global Research, Novembre

Kaastra I., Boyd M.

1996 *Designing a neural network for forecasting financial and economic time series*, Neurocomputing

Kamijo K., Tanigawa T.

1990 *Stock price pattern recognition-a recurrent neural network approach*, Neural Networks

Kane A., Kim T., White H.

2003 *Active portfolio Management: the power of the Treynor Black Model*, Nova Science Publisher, Dicembre

Kazemi H.

2012 *An Introduction to Risk Parity*, Alternative Investment Analyst Review

Kimoto T., Asakawa M., Yoda M., Takeoka M.

1990 *Stock market prediction system with modular neural networks*, Neural Networks

Kolanovic M., Krishnamachari R.

2017 *Big Data and AI strategies*, J.P Morgan, Maggio

Kolanovic M., Lau A., Lee T., Krishnamachari R.

2017 *Cross asset portfolios of tradable risk premia indices. Hierarchical risk parity: Enhancing returns at target volatility*, Global Quantitative & Derivatives Strategy J.P. Morgan, Aprile

Kohzadi N.

1996 *A comparison of artificial neural network and time series models for forecasting commodity prices*, Neurocomputing, Marzo

Ko P.C., Lin P.

2008 *Resource allocation neural network in portfolio selection*, Expert Systems with Applications

Krishnan H., Mains N.

2005 *The Two-Factor Black-Litterman Model*, Risk Magazine, Luglio

Kritzman M.

2007 *Mean-Variance versus Full-Scale Optimisation: In and Out of Sample*, Journal of Asset Managment

Kryzanowski L., Galler M., Wright D. W.

1992 *Using Artificial Neural Networks to Pick Stocks*, Financial Analysts Journal

Kruzel S.

2016 *Machine Learning Based Trading Decisions*, Astrocyte Research, Agosto

Kuan C.M., Liu T.

1995 *Forecasting exchange rates using feedforward and recurrent neural networks*, Journal of Applied Econometrics, Dicembre

Kumar N.

2017 *How AI Will Invade Every Corner of Wall Street*, Bloomberg,

https://www.bloomberg.com/news/features/2017-12-05/how-ai-will-invade-every-corner-of-wall-street, Dicembre

Lam W.

2016 *Robo-Advisors: A Portfolio Management Perspective*, Yale College, Aprile

Laney D.

2001 *3D Data Management: Controlling Data Volume, Velocity, and Variety*, Metagroup, 6 Febbraio

LeCun Y., Boser B., Denker J., Henderson D., Howard R., Hubbard W.

1989 *Backpropagation applied to handwritten zip code recognition*, Neural Computation

LeCun Y., Bengio Y., Hinton G.

2015 *Deep Learning*, Nature, Maggio

Lee W.

2000 *Advanced Theory and Methodology of Tactical Asset Allocation*, Wiley, Agosto

Li J., Bu H., Wu J.

2017 *Sentiment-aware stock market prediction: A deep learning method*, IEEE, Giugno

Luna I., Ballini R.

2012 *Adaptive fuzzy system to forecast financial time series volatility*, Journal of Intelligent e Fuzzy System

Maciel S., Gomide F., Ballini R.

2016 *An Evolving Fuzzy-GARCH Approach for Financial Volatility Modeling and Forecasting*,

Computational Econometrics

Maginn J., Tuttle D., McLeavey D., Pinto J.,

2007 *Managing investment Portfolios: A Dynamic Process*, Wiley, Aprile

Malkiel B.

1995 *Returns from Investing in Equity Mutual Funds 1971 to 1991*, The Journal of Finance, Giugno

Markowitz H.

1952 *Portfolio Selection*, The Journal of Finance, Marzo

McClelland J., Rumelhart D.

1986 *Parallel distributed processing*, M.I.T. Press

Mckinsey Global Institute,

2017 *Artificial Intelligence the next digital frontier?* McKinsey&Company, Giugno

Meucci A., Fusai G.

2003 *Assessing views*, Risk Magazine

Meucci A.

2005 *Beyond Black-Litterman: Views on No-Normal Markets*, SSRN, Novembre

Meucci A.

2006 *Beyond Black-Litterman in Practice: A Five-Step Recipe to Input Views on non-Normal Markets*, SSRN Electronic Journal

Michaud R.

1998 *Efficient Asset Management*, Oxford University Press

Mingyue Q., Cheng L., Yu S.

2016 *Application of the Artifical Neural Network in predicting the direction of stock market index*, 10th International Conference on Complex, Intelligent, and Software Intensive

Minsky M., Papert S.

1969 *Perceptrons. An Introduction to Computational Geometry*, M.I.T. Press

Milliman Reasearch Report

2015 *Risk Factor Portofolio Managemenet*, Milliman, Gennaio

Mitchell T.

1997 *Machine Learning*, McGraw-Hill, Marzo

Morgan Stanley

2017*a* *Will big data be increasingly fundamental to stock picking*?, Morgan Stanley, https://www.morganstanley.com/ideas/quant-fundamental, Dicembre

Morningstar

2005 *Standard Deviation and Sharpe Ratio*, Morningstar,

http://corporate.morningstar.com/IT/documents/MethodologyDocuments/MethodologyPapers/StandardDeviationSharpeRatio_Definition.pdf, Gennaio

Morningstar

2011 *Asset Allocation Optimization Methodology*, Morningstar, Dicembre

Mudasir M., Subekti R., Kusumawati R.

2016 *Radial basis function neural network for views prediction on Black Litterman model*, Journal of Innovative Technology and Education

NG A.

2017*a* *Machine Learning*, Stanford University CS229

NG A.

2017*b* *Machine Learning,* Coursera, https://www.coursera.org/learn/machine-learning/home/welcome

Niaki S. T., Hoseinzade S.

2013 *Forecasting S&P 500 index using artificial neural networks and design of experiments*, Journal of Industrial Engineering International

Norvig P.

2009 *The Unreasonable Effectiveness of Data*, IEEE, Marzo

O'Connor N., Madden M.G.

2006 *A neural network approach to predicting stock exchange movements using external factors*, Knowledge-Based Systems

Papenbrock J.

2016 *Using AI to establish a reliable and objective way of diversification*, Altii, https://www.altii.de/de/blog/using-ai-to-establish-a-reliable-and-objective-way-of-diversification/

Pease A., Nikolaus K.

2014 *From Biological Systems to Machines, Learning is the key*, Siemens, http://www.siemens.com/innovation/en/home/pictures-of-the-future/digitalization-and-software/artificial-intelligence-learning-is-the-key.html

Peng Y., Jiang H.

2015 *Leverage Financial News to Predict Stock Price Movements Using Word Embeddings and Deep Neural Networks*, ArXiv, Giugno

Perez C.

2002 *Technological Revolutions and Financial Capital: The Dynamics of Bubbles and Golden Ages*, Edward Elgar

Pierron A.

2017*a* *AI and Alternative Data: Moving To Trading's Next Model*, Opimas, http://www.opimas.com/research/267/detail/, Luglio

Pierron A.

2017*b* *Artificial Intelligence in Capital Markets: The Next Operational Revolution*, Opimas, http://www.opimas.com/research/210/detail/, Marzo

Podkaminer E. L.

2013 *Risk Factors as Building Blocks for Portfolio Diversification: The Chemistry of Asset Allocation*, in Investment Risk and Performance, CFA Institute

Prado M.

2016*a* *Mathematics and Economics: A Reality Check*, SSRN, Agosto

Prado M.

2016*b* *Building Diversified Portfolios that Outperform Out-of-Sample*, Journal of Portfolio Management

Prado M.

2017 *Finance as an Industrial Science*, Journal of Portfolio Management, Luglio

Prado M.

2018 *Advances in Financial Machine Learning*, Wiley, Febbraio

Purdy M., Daugherty P.

2016 *Why artificial intelligence is the future of growth*, Accenture

PwC

2017 *Asset & Wealth Management Revolution: Embracing Exponential Change*, PwC

Rasmussen M.

2003 *Quantitative Portfolio Optimisation, Asset Allocation and Risk Management: A Practical Guide to implementing Quantitative Investment Theory*, Palgrave Macmillan

Refens A.

1994 *Stock performance modeling using neural networks: A comparative study with regression models*, Neural Networks

Rich E., Knight K.

1991 *Artificial Intelligence*, McGraw-Hill

Rockel N.

2010 *Modern Portfolio Theory's Evolutionary Road*, Institutional Investor, Maggio

Rosov S.

2017 *Behind the Hype: Barclays Report on Machine Learning in Investment Management*, CFA Institute, Luglio

Ross A.

1976 *The Arbitrage Thory of capital Asset Pricing*, Journal of Economic Theory, Maggiol

Rumelhart D. E., Hinton G. E., Williams R. J.

1986 *Learning representations by backpropagating errors*, Nature

Russell S., Norvig P.

2010 *Artificial Intelligence. A Modern Approach*, Prentice Hall

Saad E.W., Prokhorov D.V., Wunsch D.C.

1998 *Comparative study of stock trend prediction using time delay, recurrent and probabilistic neural networks*, IEEE Transactions on Neural Networks, Novembre

Saidov U.

2018 *Data Science Will Transform the Investment Industry: Are You Prepared?*, CFA Institute

Samuel A.

1959 *Some Studies in Machine Learning Using the Game of Checkers*, IBM Journal of Research and Development

Servigny A., Bouzid B.

2016 *What can Machine Learning bring to investing?*, Bramham Gardens, http://www.bramham-gardens.com/wp-content/uploads/2016/03/What-can-Machine-Learning-bring-to-investing_.pdf, Marzo

Satchell S., Scowcroft A.

2000 *A Demystification of the Black-Litterman Model: Managing Quantitative and Traditional Portfolio Construction*, Journal of Asset Management

Sharpe W.F.

1964 *Capital Asset Prices: A Theory of Market Equilibrium*, The Journal of Finance, Settembre

Sharpe W.F.

1991 *The Arithmetic of Active management*, Financial Analyst journal, Gennaio

Siripurapu A.

2015 *Convolutional Networks for Stock Trading*, Stanford University

Siegel J.

2005 *Future for Investors: Why the tired and True Triumph Over the Bold and the New*, Crown Business, Marzo

Siegel J.

2014 *Stocks for the Long Run*, McGraw Hill, Gennaio

Sohangir S., Wang D., Pomeranets A., Khoshgoftaar M.T.

2018 *Big Data: Deep Learning for financial sentiment analysis*, Journal of Big Data, Gennaio

Son H., Surane J.

2017 *How to Survive Wall Street's Robot Revolution*, Bloomberg,

https://www.bloomberg.com/news/articles/2017-09-25/how-to-survive-wall-street-s-robot-revolution-quicktake-q-a, Settembre

Takeo Y., Nozawa S.

2017 *World's Biggest Pension Fund Says AI Will Replace Asset Managers*, Bloomberg, https://www.bloomberg.com/news/articles/2017-12-14/world-s-biggest-pension-fund-sees-ai-replacing-asset-managers, Dicembre

Thiel P.

2016 *Zero to One. Notes on startups, or How to build the future*, Crown Business, Settembre

Treynor J., Black F.

1973 *How to use security analysis to improve portfolio selection*, Journal of Business, Gennaio

Trippi R. R., Turban E.

1993 *Neural Networks in Finance and Investing: Using Artificial Intelligence to Improve Real World Performance*, New York, McGraw-Hill Inc.

Turing A.

1950 *Computing Machinery and Intelligence*, Mind, Ottobre

Vapnik V.N., Chervonenkis Y.

1971 *On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities*, Theory of Probability and its Application

Wang J., Leu Y.J.

1996 *Stock market trend prediction using ARIMA-based neural networks*, Neural Networks,

Giugno

McCulloch W., Pitts W.

1943 *A Logical Calculus of the Ideas Immanent in Nervous Activity*, Bulletin of Mathematical Biophysics

Walter J.

2014 *The Black-Litterman Model in Detail*, SSRN, Giugno

Wasserman L.

2012 *Statistics Versus Machine Learning, Normal Deviate*,

https://normaldeviate.wordpress.com/2012/06/12/statistics-versus-machine-learning-5-2/

White H.

1988 *Economic prediction using neural networks: the case of IBM daily stock*, IEEE

Widrow B., Rumelhart D. E., Zablehr M. A.

1997 *Neural networks: Applications in industry, business and science*, Communications of the ACM

Wigglesworth R.

2017*a* *Quant hedge funds set to surpass $1tn Management mark*, Financial Times, https://www.ft.com/content/ff7528bc-ec16-11e7-8713-513b1d7ca85a,%20Quant%20hedge%20funds%20set%20to%20surpass

Wigglesworth R.,

2017*b* *Fledgling quant funds seek to disrupt Wall Street*, Financial Time, Https://www.ft.com/content/83f4e562-65be-11e7-9a66-93fb352ba1fe, Agosto

World Economic Forum

2016 *The future of jobs*, World Economic Forum, Gennaio

Xiong J.X., Idzorek T.

2011 *The Impact of Skewness and Fat Tails on the Asset Allocation Decision*, Financial Analysts Journal, Aprile

Xiong R., Nichols E.P., Shen Y.

2015 *Deep Learning Stock Volatility with Google Domestic Trends*, ArXiv

Yao J., Tan C. L., Poh Hean L.

1999 *Neural Networks for Technical Analysis: A Study on KLCI*, International Journal of Theoretical and Applied Finance

Yao K., Cohn T., Vylomova K., Duh K., Dyer K.

2015 *Depth-Gated LSTM*, ArXiv

Yoav Goldberg

2016 *A Primer on Neural Network Models for Natural Language Processing*, Journal of Artificial Intelligence

Yoon Y., Swales G.

1991 *Predicting Stock Price Performance: A Neural Network Approach*, IEEE 24th Annual Internal Conference of Systems Sciences

Zikopoulos

2012 *Understanding Big Data: Analytics for enterprise class hadoop and streaming data*, McGraw-Hill

Zhang G., Patuwo E., Hu M. Y.

1998 *Forecasting with artificial neural networks: The state of the art*, International Journal of Forecasting

Zimmermann H.

2002 *Active Portfolio-Managment based on Error Correction Neural Networks*, Proceedings of the 14th International Conference on Neural Information Processing Systems: Natural and Synthetic

Zhou X., Pan Z., Hu G., Tang S., Zhao C.

2018 *Stock Market Prediction on High Frequency Data Using Generative Adversarial Nets*, Hindawi, Novembre

Ringraziamenti

Vorrei ricordare tutti coloro che mi hanno aiutato nel corso del lavoro di tesi, a loro va la mia gratitudine.
Ringrazio innanziatutto il Prof. Francesco Pattarin, per avermi guidato nella impostazione della tesi.
Ringrazio Daniele Grassi, CEO di Axyon AI, per avermi dato la possibilità di sviluppare il progetto di tesi utilizzando le risorse dell'azienda e Jacopo Credi, Chief Scientist, per il supporto tecnico.
Proseguo con i ragazzi di Axyon AI che mi hanno accolto e trattato con grande rispetto e professionalità.
Infine vorrei ringraziare le persone a me più care, i miei genitori che, con il loro sostegno morale ed economico, mi hanno permesso di arrivare fino a qui, contribuendo alla mia formazione personale.

Axyon AI

Axyon AI è un'azienda che offre soluzioni per banche e istituzioni finanziarie attraverso il Deep Learning. Nasce da un'idea di Daniele Grassi, CEO di Axyon AI, a seguito dell'esperienza decennale acquisita attraverso DM Digital, software house fondata nel 2007 dallo stesso.
Axyon AI lavora a stretto contatto con l'Università di Modena e Reggio Emilia e con il centro di ricerca Softech ICT e vanta collaborazioni importanti tra cui ING Bank, rapporto nato a seguito del loro programma d'accelerazione fintech.
Negli ultimi anni ha ricevuto alcuni riconoscimenti importanti; il più recente è il primo posto nel PITCH360 del marzo 2018 a Londra.