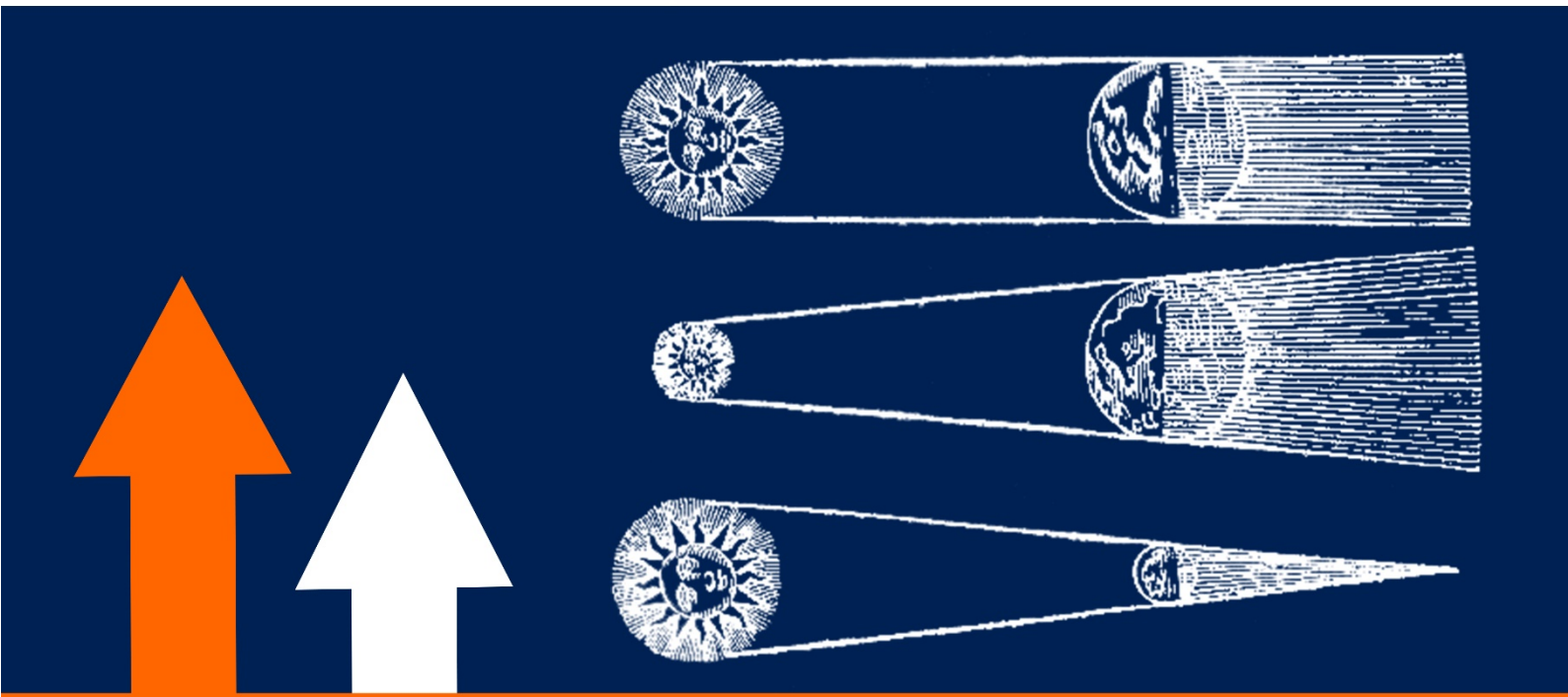
Itinerari per l'alta formazione

IRCrES

UNA BREVE INTRODUZIONE ALLE TECNICHE DI DATA MINING

Greta Falavigna



CNR - Consiglio Nazionale delle Ricerche

IRCrES - Istituto di Ricerca sulla Crescita Economica Sostenibile

Direttore Emanuela Reale

Direzione CNR-IRCRES
Istituto di Ricerca sulla Crescita Economica Sostenibile
Via Real Collegio 30, 10024 Moncalieri (Torino), Italy
Tel. +39 011 6824911 / Fax +39 011 6824966
segreteria@ircres.cnr.it
www.ircres.cnr.it

Sede di Roma Via dei Taurini 19, 00185 Roma, Italy
Tel. +39 06 49937809 / Fax +39 06 49937808

Sede di Milano Via Corti, 12, 20121 Milano, Italy
Tel. +39 02 23699501 / Fax +39 02 23699530

Sede di Genova Università di Genova Via Balbi, 6 - 16126 Genova
Tel. +39 010 2465459 / Fax +39 010 2099826

Redazione Emanuela Reale
Giancarlo Birello
Antonella Emina
Serena Fabrizio
Anna Perin
Isabella Maria Zoppi
redazione@ircres.cnr.it
www.ircres.cnr.it/index.php/it/produzione-scientifica/pubblicazioni

Itinerari per l'alta formazione 2, febbraio 2021
ISBN: 978-88-98193-23-3

Una breve introduzione alle tecniche di Data Mining

GRETA FALAVIGNA

Istituto di Ricerca sulla Crescita Economia e Sostenibile del Consiglio Nazionale delle Ricerche

corresponding author: greta.falavigna@ircres.cnr.it

ABSTRACT

The aim of this publication is to expose students to use basic tools for the analysis of big amount of data. The first section starts presenting the definition of Data Mining and Knowledge Discovery in Database explaining the more common techniques and listing the main operational applications.

A second paragraph illustrates the first three phases preceding the application of Data Mining techniques: Selection/Sampling, Pre-processing/Cleaning and Transformation/Reduction of data. These prelaminar data analysis techniques are essential as the results of the Data Mining models depend on the correctness of the data.

The third paragraph presents some applications of methodologies. In this section, the technical aspect has less relevance than the operational one with the aim to explain the use of these techniques. However, the more common Data Mining models are listed and explained.

The fourth paragraph is addressed to the Text Mining and Web Mining, which are two methodologies used to analyze texts and websites. This section presents the main problems related to textual analysis and the techniques that can be used to obtain effective searches.

Finally, two appendices have been added: the Statistical Appendix reports some technical insights that may be useful for understanding the Data Mining systems; in a second appendix, a Short Glossary containing the main terms related to Data Mining used in the text is proposed.

KEYWORDS: Artificial Neural Networks; Data Mining; Text Mining; Classification; Clustering

DOI: 10.23760/978-88-98193-2021-02

ISBN: 978-88-98193-23-3

HOW TO CITE THIS BOOK

Falavigna, G. (2021). *Una breve introduzione alle tecniche di Data Mining*. Moncalieri: CNR-IRCrES (Itinerari per l'alta formazione). http://dx.doi.org/10.23760/978-88-98193-2021-02

Indice

1. INTRODUZIONE

Il presente testo ha come obiettivo principale quello di rispondere a una domanda che spesso studenti e ricercatori si trovano ad affrontare: cosa faccio con tutti questi dati? Le tecniche di *data mining* offrono una risposta a questo quesito soprattutto perché la maggioranza dei *software* statistici disponibili al giorno d'oggi incorporano delle *routine* che rendono l'applicazione di questi modelli molto semplice.

La breve guida proposta si rivolge a studenti che non hanno confidenza con l'analisi dei dati e che necessitano di farsi un'idea di cosa possono dire i *database* e soprattutto di quali sono gli strumenti più idonei per rispondere alle domande di ricerca.

Il testo si compone di diversi paragrafi che attraverso definizioni ed esempi mostreranno al lettore come estrarre informazioni dai *database* in modo semplice e scoprire pertanto alcuni meccanismi che vengono utilizzati nel mondo della ricerca, ma non solo, per comprendere al meglio il comportamento degli attori delle realtà economiche.

Nel **primo paragrafo** viene presentata la definizione di *Data Mining* e di *Knowledge Discovery in Database*. In questa sezione vengono descritte le principali tecniche ed elencate le principali applicazioni operative.

Il **secondo paragrafo** illustra le prime tre fasi del processo di elaborazione dati che precede l'applicazione delle tecniche di *Data Mining*: la Selezione/Campionamento, la Pre-elaborazione/Pulizia e la Trasformazione/Riduzione dei dati. Queste tecniche di analisi preliminare dei dati sono essenziali in quanto la bontà dei risultati dei modelli di *Data Mining* dipende proprio dalla qualità dei dati.

Le tecniche di *Data Mining* sono presentate nel dettaglio nel **terzo paragrafo**. Le metodologie sono descritte soprattutto per quanto riguarda il loro funzionamento e la loro applicazione. L'aspetto tecnico ha meno rilevanza di quello operativo in quanto si è voluto dare un'idea concreta all'utilizzo dei differenti modelli. Tuttavia, anche se trattati non nello specifico, in questo paragrafo sono elencate e spiegate le principali tecniche di *Data Mining* e le loro applicazioni.

Il **quarto paragrafo** si occupa di presentare il *Text Mining* e il *Web Mining* che sono due metodologie utilizzate per analizzare i testi e il *web*. In questa sezione vengono presentati i principali problemi legati all'analisi testuale e alle tecniche che possono essere utilizzate per ottenere delle ricerche efficaci.

Infine, sono state inserite due appendici: l'**Appendice statistica** riporta alcuni approfondimenti statistici che possono essere utili per capire il funzionamento dei sistemi di *Data Mining* ma anche dei modelli in generale; un **Breve Glossario** in cui vengono riportati i principali termini legati al *Data Mining* ed utilizzati nel testo.

2. IL *DATA MINING*

La maggiore capacità dei moderni dispositivi di memorizzazione permette oggi di collezionare sempre maggiori quantità di dati che possono essere organizzati ed analizzati con differenti scopi. In contrasto con il tradizionale paradigma scientifico per cui i dati vengono

raccolti con il preciso intento di testare le ipotesi definite all'inizio dell'indagine, oggi sono i dati stessi a costituire il punto di partenza e le ipotesi nascono dalla loro stessa analisi attraverso delle tecniche apposite.

L'area di ricerca che si occupa di indagare i *database* costruiti si chiama *Data Mining* o *Knowledge Discovery*. L'espressione KDD (*Knowledge Discovery in Database*) si riferisce a tutto il processo di estrazione di conoscenza applicato ai *database* o anche in generale a delle informazioni o a dati non strutturati. Si tratta dunque del processo attraverso il quale si estrae conoscenza da una banca dati.

Perché sia più facilmente comprensibile la strategia di analisi delle tecniche di *Data Mining* e di *Knowledge Discovery* in *database*, Fayyad et al. (1996) hanno ritenuto necessario sottolineare delle distinzioni tra i seguenti concetti fondamentali:

- **Dati**: sono un insieme di registrazioni nate da fatti. Ad esempio l'acquisto di un bene causa la registrazione di un fatto;
- **Pattern**: è una espressione in un linguaggio specifico che descrive una relazione ricorsa in un insieme di dati. Ad esempio, se abbiamo rilevato una relazione tra l'indebitamento e il rischio di fallimento delle imprese, questa rappresenta il *pattern* trovato;
- **Processo**: rappresenta l'insieme delle fasi operative del modello (ad esempio: la preparazione dei dati, la ricerca di *pattern*, il processo di scoperta, la valutazione e la successiva iterazione del modello);
- **Validità**: misura la bontà dei *pattern* identificati;
- **Novità**: è una misura valutata rispetto alle variazioni dei dati ad esempio attraverso il confronto del valore attuale con quello passato o con quello previsto;
- **Utilità**: è una misura che viene associata ad ogni *pattern*;
- **Comprensibilità**: il *Data Mining* ha come obiettivo quello di fare in modo che i *pattern* identificati spieghino nel migliore dei modi i dati che li hanno generati.

Sulla base di quanto esposto in precedenza e di quanto espresso da Fayyad et al. (1996) e da Dulli et al. (2009) con il nome di *Data Mining* "si intende l'applicazione di una o più tecniche che consentono l'esplorazione di grandi quantità di dati individuando i *pattern* più significativi". Allo stesso modo, il *Knowledge Discovery* in *Database* è quel processo che "usa i metodi (algoritmi) di *Data Mining* per estrarre e identificare conoscenza dai *pattern*, in accordo con le misure e le soglie usate sui fatti".

2.1. Il *Data Mining*: le principali fasi

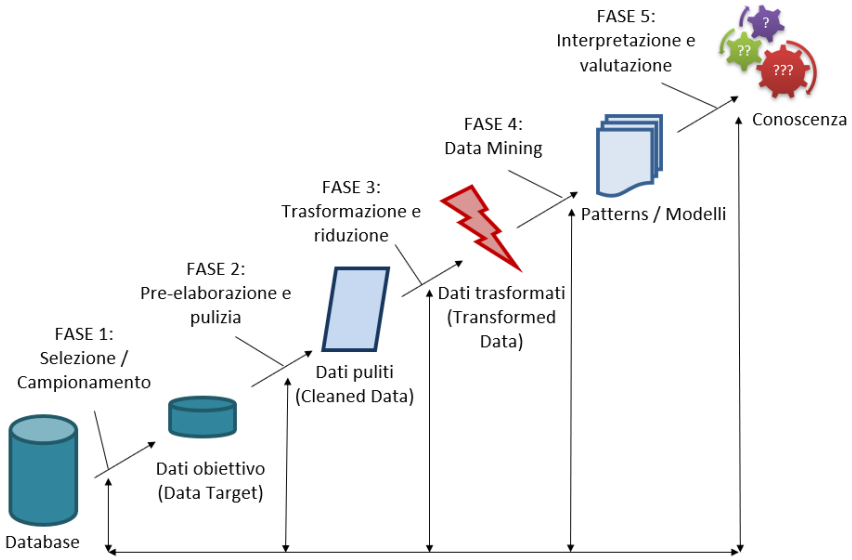
La condizione fondamentale per una efficace identificazione dei *patterns* riguarda l'organizzazione del *database* che deve contenere dati integrati che rispecchino tutta la realtà che si vuole indagare. Se ciò avviene, allora i dati potranno efficacemente essere analizzati per recuperare le informazioni che interessano. A questo punto, sarà necessario procedere a elaborare quanto ottenuto in modo da renderlo velocemente comprensibile e fruibile.

Il *database* deve dunque essere costruito in modo da eliminare tutte le ridondanze o le informazioni che non sono necessarie in quanto possono essere un "disturbo" all'analisi e alla

valutazione dei dati. Inoltre, l'obiettivo di un sistema di *Data Mining* è anche quello di scoprire le informazioni nascoste nei dati (i.e., *discovery driven*) in modo da poter migliorare la conoscenza.

Osservando la Figura 1 è possibile definire il processo di KDD in base a quanto espresso da Fayyad et al. (1996):

- **Selezione**: basandosi sull'obiettivo dell'analisi, si procede alla definizione di specifici criteri e alla successiva estrazione dei dati. In statistica questa fase viene chiama di "campionamento";
- **Pre-elaborazione**: durante questa fase i dati vengono analizzati e ripuliti da quelli che possono rallentare l'elaborazione e la definizione di *patterns*. Inoltre, i dati, in questa fase, possono essere trasformati in modo tale che sia possibile utilizzarli tramite algoritmi matematici. Ad esempio, se in un *database* abbiamo dei dati su delle persone e una variabile registra il sesso, questa generalmente viene trasformata in una variabile numerica 0 per maschio e 1 per femmina (i.e., questo tipo di variabile dicotomica viene chiamata *dummy*);
- **Trasformazione**: durante questa fase i dati vengono arricchiti con altre informazioni. Ad esempio possono essere accorpati (*merge*) con altri *database* in modo che sia reso disponibile il maggior numero di informazioni possibile;
- ***Data Mining***: durante questa fase i dati vengono analizzati con lo scopo di estrarre delle regole e dei modelli;
- **Interpretazione e Valutazione**: i modelli vengono valutati ed interpretati in modo che la conoscenza derivante possa essere di aiuto nelle decisioni. Ad esempio, sarà possibile prevedere l'andamento di un fenomeno, classificare un elemento nuovo in una classe, descrivere gli elementi di un *database* o dare un senso alle relazioni evidenziate tra i dati.



Fonte: elaborazione da Fayyad et al. (1996).
Figura 1. Processo di estrazione della conoscenza.

A questo punto, il processo di *Data Mining* si articola in due fasi. Si parte dall'**esplorazione dei dati** che prevede una prima valutazione, descrittiva a visiva, dei dati così come raccolti nel *database* e si prosegue con la **generazione di *pattern.*** Questa seconda fase prevede l'elaborazione di algoritmi di scoperta di associazioni utili per generare modelli. Durante questa fase avvengono inoltre la convalida e l'interpretazione dei modelli scoperti.

2.2. Il *Data Mining*: le tecniche

Gli strumenti che possono essere considerati *Data Mining tools* sono quelli che, contrariamente ai molti *software* in circolazione, supportano la scoperta automatica dei *pattern*. Da questo punto di vista, esistono due tecniche che fungono da guida per la creazione di applicazioni per il *Data Mining* (Dulli et al., 2009):

- **Metodo di verifica**. Con questo metodo si parte dalle ipotesi che l'utente ha elaborato per poi verificare se nei dati queste ipotesi sono verificate. Il problema di questo metodo è che non viene creata nessuna informazione, ma la risposta dei dati serve solo a confermare o smentire le ipotesi definite dall'utente. Per ovviare a questa carenza, in statistica è stata elaborata una tecnica chiamata *Exploratory Data Analysis*;
- **Metodo di scoperta**. Attraverso questa tecnica, il sistema è in grado di scoprire le informazioni nascoste in modo completamente automatico. Questi vengono analizzati con l'obiettivo di ricercare similitudini e generalizzazioni senza che l'utente intervenga.

2.3. Il *Data Mining*: applicazioni operative

Seguendo quanto proposto da Dulli et al. (2009), qui di seguito vengono elencate le più frequenti applicazioni di *Data Mining:*

- ***Scoring system* (*predictive modelling*)**. Questa tecnica prevede di assegnare dei punteggi in base alla probabilità che un certo evento si verifichi. Uno degli esempi più "famosi" di questa applicazione è chiamata *credit scoring.* Il punteggio assegnato in questo caso rappresenta la capacità di ripagare un debito del soggetto richiedente (i.e., la solvibilità finanziaria). Utilizzano queste metodologie le agenzie di rating (Standard and Poor's, Fitch Rating o Moody's) nel momento in cui emettono un giudizio circa la solvibilità di un paese o di un'impresa;
- **Segmentazione della clientela (*customer profiling*)**. Partendo dalle informazioni comportamentali e/o socio-demografiche degli utenti, un algoritmo di *clustering* sarà in grado di suddividere i soggetti in gruppi omogenei. A questo punto le strategie di marketing potranno essere definite in base alle caratteristiche dei differenti gruppi in modo da poterne massimizzare l'efficacia;
- ***Market basket analysis* (*affinity analysis*)**. Questa tecnica analizza le vendite di prodotti acquistati insieme. Da questa analisi, vengono estratte le informazioni necessarie a capire come collocare i prodotti sugli scaffali;
- **Rilevazione di frodi (*fraud detection*)**. Questa tecnica consente di individuare profili che sono più propensi alla frode/morosità da parte di nuovi clienti in fase di contratti/transazioni;

- **Analisi degli abbandoni (*churn analysis*)**. Attraverso queste tecniche si individuano i clienti che potenzialmente potrebbero abbandonare un prodotto e quindi non riacquistarlo più. Si pensi, ad esempio, ai messaggi privati che arrivano dalle compagnie di telefonia mobile con offerte personalizzate;
- ***Text mining***. Si tratta dell'applicazione di tecniche di *Data Mining* a documenti che risiedono su file testuali (articoli, libri, rapporti, cartelle cliniche, relazioni, questionari, e-mail, forum, chat etc.). Queste tecniche permettono di raggruppare diversi documenti in base all'argomento trattato.
  Quando queste tecniche sono applicate al *web* si parla di *Web Mining* e due sono le tecniche più interessanti:
  - ***Click-stream analysis***. Questa tecnica analizza l'attività che avviene nei siti *web* in modo da capire quali sono le pagine di un sito che portano i maggiori acquisti di prodotti. Non solo, pensando ai *social networks* (facebook, twitter, linkedin, etc.…) può essere utilizzato per capire quali siano gli argomenti di interesse maggiore tra le persone e la relazione con le variabili qualitative delle stesse (età, sesso, professione, nazionalità, etc.). Si tratta soprattutto di tecniche utili per la definizione delle strategie pubblicitarie sul *web*;
  - ***Dynamic content targeting***. Si tratta di algoritmi che permettono di adattare la visualizzazione delle pagine *web* di un sito in base al tipo di visitatore che le consulta.

3. I DATI… CHI SONO COSTORO?

In questo paragrafo saranno trattate le prime tre fasi di cui si è parlato nella precedente paragrafo: selezione/campionamento, pre-elaborazione/pulizia, trasformazione/riduzione.

Questi stadi verranno analizzati mostrando le principali tecniche utilizzate.

3.1. FASE 1: Selezione e Campionamento

Prima di procedere alla modellizzazione è necessario affrontare due temi particolarmente spinosi per il metodologo in quanto necessitano di una buona conoscenza sia della statistica sia dell'argomento indagato. È infatti frequente l'errore per cui vengono escluse alcune variabili che apparentemente non sono significative ma che invece ad una più attenta analisi risultano essere rilevanti nell'indagine. Per questo motivo, la prima fase che bisogna affrontare è quella che comprende la selezione o campionamento dei dati, cioè l'estrazione di una selezione di dati (campione) dal *database* iniziale (popolazione). Successivamente viene effettuata della statistica inferenziale orientata ad una preliminare analisi delle relazioni esistenti tra i dati.

3.1.1. Campionamento

Il problema principale del *Data Mining* è la definizione di modelli in grado di rappresentare il più verosimilmente possibile i dati. Per questo motivo, un modello va inizialmente

creato e modellato in un insieme ristretto di dati derivanti da un *campionamento*. Una volta testato il modello sul campione, questo viene esteso a tutti i dati a disposizione.

La caratteristica principale di un efficace modello di *Data Mining* è quella di essere in grado di "generalizzare", cioè di saper associare un nuovo elemento al corretto gruppo. Il numero di errori per un modello è l'errata attribuzione di un'osservazione ad un gruppo. Tuttavia, se il modello è troppo flessibile il rischio è che si adatti perfettamente ai dati del campione e non sia in grado di riconoscere le osservazioni che non vi appartengono. In questo caso si dice che il modello soffre di *overfitting* e non è in grado di predire/classificare correttamente l'universo da cui il campione è stato estratto. Esistono degli algoritmi di "regolarizzazione" (come la *cross-validation*) che permettono di evitare questo problema attraverso l'immissione all'interno del modello di osservazioni in modalità random oppure l'applicazione di tecniche di "bootstrap" che permettono invece di creare dei soggetti simulati sulla base di quelli appartenenti al campione con l'introduzione di uno scarto prestabilito. Queste tecniche consentono di raffinare il modello ed evitare il problema della "super-specializzazione" (*overfitting*) del modello.

Si definisce *tecnica di campionamento* ogni procedura di scelta di unità da una popolazione statistica (Dulli et al., 2009) e lo *spazio campionario* è l'insieme dei campioni estratti. Esistono diverse tecniche che permettono di ottenere dei campioni che effettivamente rappresentano la popolazione iniziale, considerando attentamente il problema dell'*overfitting*[1].

Un algoritmo di apprendimento (*learner*) apprende le relazioni tra i dati da un sotto-campione che viene chiamato *training set*. Una volta raccolte le informazioni che legano le differenti variabili, l'algoritmo applica quanto appreso sul rimanente sotto-campione, chiamato *testing* (o *validation*). L'algoritmo smetterà di apprendere solo se sarà in grado di minimizzare l'errore di classificazione su questo secondo campione.

Come si è accennato poco prima, può accadere che il modello si superspecializzi a riconoscere i dati del *training set* in quanto ha molte informazioni per poche osservazioni, cadendo dunque in problemi di *overfitting*. La successiva Figura 2 rappresenta la schematizzazione del procedimento di apprendimento e validazione delle tecniche di *Data Mining* e permette di sottolineare quanto sia importante la fase di campionamento al fine di ottenere un modello performante.

Durante la fase di *training* una parte del *database*, circa di 2/3 del totale, viene utilizzato per definire i parametri del modello di *Data Mining*. Degli elementi del *database* si conoscono i risultati, ad esempio, se stiamo considerando un *database* di imprese e vogliamo creare un modello per capire il rischio di fallimento, sapremo quali di queste sono fallite e quali no. Questi risultati si chiamano output teorici e vengono inseriti nel modello insieme alle altre variabili relative ai soggetti del *database* (ad esempio, indici di performance economico-finanziari). Il sistema di *Data Mining* definisce dei parametri e assegna degli output ad ogni elemento, cioè l'output empirico (nel caso in esame lo stato di impresa fallita o sana). Dal confronto tra output empirico e teorico vi sono degli algoritmi che fanno in modo di ricalibrare i parametri del modello finché questo non generi i risultati attesi. Queste tecniche considerano anche il problema dell'*overfitting*.

[1] Si parla di *overfitting* (eccessivo adattamento) quando un modello statistico si adatta ai dati osservati (il campione) perché ha un numero eccessivo di parametri rispetto ai dati osservati.

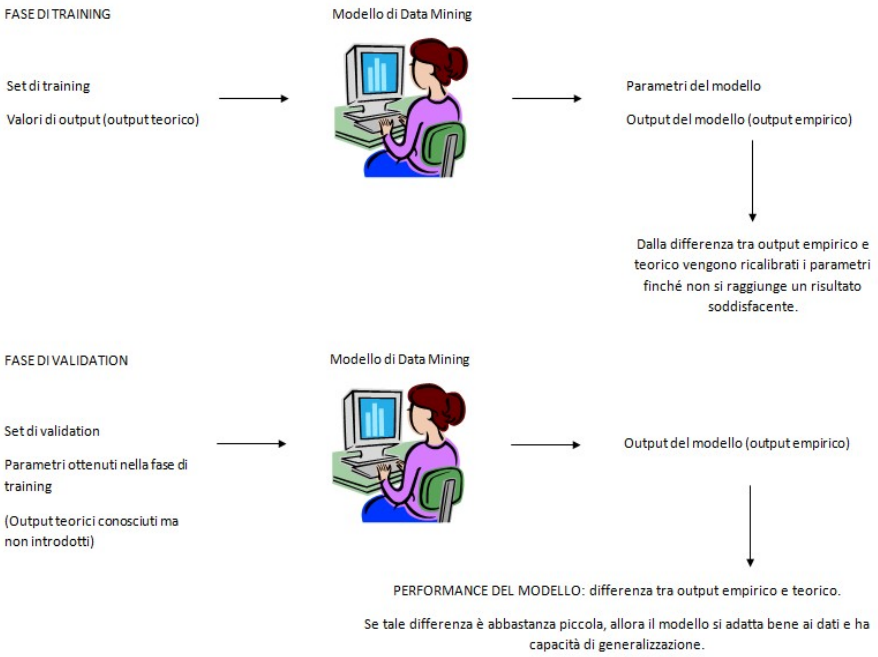


Figura 2. Logica di funzionamento "apprendimento e validazione" dei sistemi di *Data Mining*.

Una volta definiti i parametri e la forma del modello, questi vengono applicati ai dati contenuti nel *validation set*. Anche se si conoscono gli output teorici di queste osservazioni, questi non vengono introdotti ma solo utilizzati per una valutazione finale del modello. Se infatti nella prima fase, la procedura può richiedere del tempo perché gli algoritmi di riaggiustamento dei parametri richiedono di inserire i dati iterativamente più volte all'interno del modello, in questa seconda fase, il modello viene fatto "girare" una sola volta, come se fosse quello definitivo, con l'intento di testarne le performance. Come nel caso precedente, dagli scostamenti tra output empirici e teorici, otteniamo la performance del modello e a questo punto potrà essere applicato a tutti i soggetti che sono interessati al risultato del modello.

Appare dunque chiaro quanto sia determinante campionare correttamente perché un *training set* inadeguato porta alla costruzione di un modello e relativi parametri che soffrono di *overfiting* oppure che non sono rappresentativi della popolazione.

Una tecnica che permette di migliorare l'affidabilità dei risultati ed eliminare il problema dell'*overfitting* è l'algoritmo di *k-fold cross validation* che può essere considerato un tipo di *bootstrap*. Questa tecnica prevede di suddividere l'insieme dei campioni in K partizioni delle stesse dimensioni indipendenti tra loro. Si stabilisce un numero di ripetizioni del modello e ogni volta che quest'ultimo viene fatto girare si utilizza una partizione come *validation* e le restanti come *traning*. Per ogni soggetto si avranno dunque tanti output del modello quante sono le ripetizioni. Il valore medio di queste stime viene chiamato "errore di *cross-validation*". Questa tecnica può essere complicata inserendo delle stratificazioni sui dati. Se ad esempio stiamo considerando delle imprese di cui vogliamo conoscere la probabilità di insolvenza in un certo periodo, potrebbe essere utile stratificare per la dimensione oppure per il settore di attività economica. È infatti noto che, per esempio, le piccole imprese riscontrano più fatica a ottenere credito dalle banche così come vi sono settori industriali più a rischio di altri.

Naturalmente, maggiore è il numero di stratificazioni che inseriamo e maggiore è l'accuratezza della stima. Tuttavia, questo procedimento, oltre a richiedere molte osservazioni (bisogna pensare al numero di stratificazioni e al fatto che dobbiamo avere un *set* di *training* e uno di *validation*), può richiedere anche parecchio impegno computazionale. Allo stesso tempo però, questa metodologia rende possibile affinare il modello e permette di ottenere dei risultati soddisfacenti.

Esistono poi altri algoritmi che servono per migliorare la fase di campionamento ma sono delle specificazioni di questi due precedentemente illustrati e raramente vengono utilizzati. Per conoscenza si tratta delle seguenti tecniche: *leave one out* (Fukunaga e Hummels, 1989), *bagging* (Breiman, 1996) e *boosting* (Freund et al., 1999).

3.1.2. Inferenza

L'inferenza è un insieme di analisi statistiche che viene condotto sul campione al fine di trarre delle deduzioni sulla popolazione intera. Quest'ultima è quasi sempre difficilmente studiabile nella sua interezza ma l'inferenza permette di dedurre delle informazioni partendo da un campione della stessa.

La prima analisi che viene condotta è sulla distribuzione dei dati in quanto, una volta conosciuta, diviene possibile studiare alcune caratteristiche dei dati campionari e trasferirle a quelli della popolazione.

La distribuzione più comune utilizzata per rappresentare i dati è quella "normale" o "gaussiana" poiché la maggioranza dei dati di una variabile si presenta distribuita normalmente e se così non fosse, vi sono delle trasformazioni matematiche che consentono di riportarli alla forma gaussiana.

La distribuzione normale ha un ruolo fondamentale nel *Data Mining* in quanto viene molto utilizzata nelle tecniche di *clustering* e in diverse stime.

Consideriamo una variabile casuale[2] X che ha media μ_x e varianza σ_X^2:[3]

$$\mu_X = E[X] = \sum_{-\infty}^{+\infty} x_i - p_i \qquad \text{per variabili discrete} \qquad (3.1)$$

$$\mu_X = E[X] = \int_{-\infty}^{+\infty} x \cdot p_x(x) \cdot dx \qquad \text{per variabili continue} \qquad (3.2)$$

$$var(X) = \sigma_X^2 = E[X^2] - E[X\}^2 \qquad (3.3)$$

La media e la varianza bastano per definire la forma a campana della variabile casuale normale. L'area sotto la curva rappresenta la densità di probabilità e il suo valore è pari a 1. Inoltre tale forma funzionale è simmetrica rispetto alla media. In Figura 3 sono raffigurate più curve normali, diverse in base a media e varianza.

[2] Una variabile casuale è una variabile che assume determinati valori in modo casuale (non deterministico). Ad esempio: l'esito di una estrazione del Lotto, il risultato di una partita di calcio, il voto di un esame, etc.

[3] Si ricordi che la Deviazione Standard è pari a $\sigma_X = \sqrt{var(X)}$.

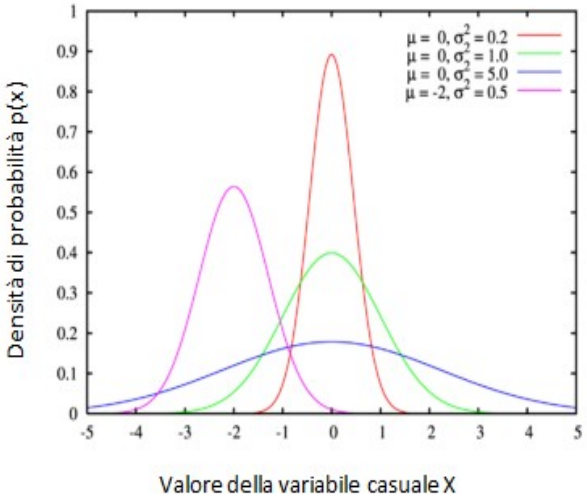


Figura 3. Alcune funzioni di densità di probabilità normali.

3.2. FASE 2: Pre-elaborazione e Pulizia

Questa fase è una delle più critiche poiché permette di stabilire la qualità dei dati a disposizione e soprattutto prevede alcune operazioni che permettono di "pulire" il campione da analizzare. Diversi sono i problemi che si possono incontrare quando si ha a che fare con i dati, come ad esempio la mancanza di alcune informazioni (i.e., *missing value*) e la conseguente necessaria eliminazione dell'osservazione dal campione.

La fase di pre-elaborazione serve dunque ad eseguire le seguenti operazioni:

- **Pulitura dei dati**: questa fase viene usata per riempire i vuoti del *database*, cioè per assegnare dei valori ai già citati *missing values*. Serve inoltre per eliminare i dati che introducono solo del rumore ma che non servono all'analisi e per rimuovere dati non realistici (cosiddetti *outliers*). Generalmente queste operazioni possono essere eseguite in modo automatico attraverso degli specifici algoritmi; tuttavia, solo l'analisi a priori di esperti può effettivamente dire se una variabile può portare o meno informazione. Pensiamo ad esempio a uno studio medico in cui in base all'osservazione di alcuni pazienti, si vuole cercare di capire quali fattori predispongano i pazienti a una certa malattia. Solo se siamo medici possiamo escludere a priori alcune variabili, così come invece sostenere che non si può rinunciare a qualcuna di esse.

  Per quanto concerne il problema dei *missing values* la prima operazione da effettuare è quella di capire perché ci sono dei dati mancanti. Le cause più comuni sono una errata rilevazione del dato, una cancellazione errata dei dati, l'inserimento di un dato volutamente errato, un errore di misura, oppure la non obbligatorietà dell'informazione. Per risolvere questi problemi si può procedere in diversi modi. I principali sono i seguenti:

  1. Ignorare le osservazioni che hanno valori mancanti tenendo presente che se il numero di *missing values* è elevato si rischia di ridurre notevolmente il campione e se stiamo classificando, è necessario assicurarsi che esista un numero sufficiente di osservazioni

per ogni classe, altrimenti la parametrizzazione del modello potrebbe risentirne. Tuttavia, questa è la soluzione tecnicamente più corretta.

2. Riempire i valori mancanti stimandoli da quelli presenti. Tuttavia questa tecnica non sempre è fattibile e soprattutto rischia di introdurre comunque un dato non corretto;
3. Usare un valore costante come 0, n.a., null, 999 ma anche in questo caso è necessario porre molta attenzione. Difatti, questa soluzione altera l'algoritmo di analisi. Esistono dei *software* come Matlab che creano delle combinazioni lineari o meno (a discrezione dell'utente) con le altre osservazioni. Tuttavia il rischio è di avere un dato non realistico;
4. Usare la media della variabile. Ad esempio, consideriamo una classe di liceo di cui conosciamo i voti per ogni alunno tranne che per la materia matematica per l'alunno X. Secondo questa tecnica il voto di matematica dell'alunno X sarà pari alla media dei voti di matematica dell'intera classe. L'esempio fa capire che questa tecnica può dare delle informazioni distorte perché magari l'alunno X è il più bravo di matematica oppure il contrario... Come si può risolvere questo problema?
    - Usare la media della variabile ma considerando il gruppo di appartenenza del soggetto. Considerando l'esempio precedente si considera la media del voto di matematica di alunni che assomigliano (entro un certo grado) all'alunno X nelle altre materie/caratteristiche;
    - Predire il valore mancante in base alle altre variabili. In questo caso si usano modelli come quelli della regressione lineare, alberi decisionali o di classificazione oppure veri e proprio sistemi di *data mining*.

- **Integrazione dei dati**: arricchire il *database* con informazioni provenienti da altre fonti;
- **Trasformazione dei dati**: trasformare i dati in modo che siano pronti per essere elaborati dal modello di analisi;
- **Riduzione dei dati**: verificare che il *database* contenga le informazioni necessarie per ottenere dei validi risultati senza però che ci sia un eccesso di variabili che invece possono introdurre dei disturbi nella validazione del modello. Una tecnica interessante che ha come obiettivo quello di ridurre la variabilità tra i dati è quella della discretizzazione o *binning*. Questa operazione consente di eliminare il rumore e cioè quell'informazione che non migliora la portata informativa dei dati ma semplicemente li sporca. Esistono molteplici tecniche che permettono di fare *binning* ma le più comuni sono lo *smoothing by bin means*, lo *smoothing by bin medians* e lo *smoothing by bin boundaries*[4]. In sostanza gli elementi delle variabili vengono ordinati, suddivisi in intervalli (*depth*) e si sostituiscono i valori della variabile con la media o la mediana dell'intervallo a cui gli stessi valori appartengono.
  Consideriamo il seguente esempio, basato su quanto proposto da Dulli et al. (2009): data la variabile *prezzo* si determini per ogni *bin* il corrispondente dato da assegnare ad ogni elemento della variabile:

[4] Si veda l'appendice per l'approfondimento degli algoritmi (p. 74).

Prezzo: 5 9 10 16 22 22 25 26 27 29 30 35

- Partizioniamo in intervalli di 4 elementi (d=4):[5]
  - Bin 1: 5, 9, 10, 16 →μ=10; *Me*=10 (arrotondata per eccesso)
  - Bin 2: 22, 22, 25, 26 →μ=24 (arrotondata per eccesso); *Me*=24 (arrotondata per eccesso)
  - Bin 3: 27, 29, 30, 35 →μ=30 (arrotondata per difetto); *Me*=30 (arrotondata per eccesso)

- Poiché Media e Mediana coincidono le tecniche *Smoothing by bin means* e *Smoothing by bin medians* coincidono:

  - Bin 1: 10, 10, 10, 10;
  - Bin 2: 24, 24, 24, 24;
  - Bin 3: 30, 30, 30, 30

- *Smoothing by bin boundaries*[6]:

  - Bin 1: 5, 5, 5, 16;
  - Bin 2: 22, 22, 26, 26;
  - Bin 3: 27, 27, 27, 35

Tuttavia, non è sempre possibile avere intervalli di esattamente d elementi. Per questo motivo esistono gli algoritmi *Equi-Width Binning* e *Equi-Depth Binning* che permettono comunque di assegnare valori ottenuti da intervalli più o meno della stessa ampiezza.

3.3. FASE 3: Trasformazione e Riduzione

La fase di trasformazione e riduzione consiste nell'applicare al campione alcune operazioni che consentono di ottenere dei risultati più precisi e corretti. Durante questa fase si procede ad un'analisi esplorativa dei dati, alla loro rappresentazione e ad una prima valutazione statistica delle variabili.

---

[5] La media μ di una variabile x è il valore medio che si ottiene da:

$\mu = \frac{1}{N} \cdot \sum_{i=1} N x_i$

Si dice mediana di N numeri e si indica con *Me*, il valore che occupa la posizione centrale nella successione dei numeri ordinati in senso non decrescente e precisamente (Orsi, 1995):

- se n è dispari, il termine che occupa la posizione $(n + 1)/2$;
- se n è pari, per convenzione, la semisomma dei termini che occupano le posizioni
- $n/2$ e $(n/2 + 1)$

Indicando con F_i la generica frequenza cumulata dell'osservazione i-esima, la mediana è:

$Me = x_i + (x_{i+1} - x_i) \cdot \frac{0.5 - F_{i-1}}{F_i - F_{i-1}}$

[6] Per spiegare i valori prendiamo in considerazione il Bin 1.

– Per quanto concerne il primo valore ci dobbiamo chiedere [5 - il min (cioè 5)] è minore del [Max (cioè 16) - 5]? In questo caso sì e quindi il valore da assegnare è il minimo, cioè 5.
– Consideriamo il secondo valore: 9 - 5 è minore di 16 - 9? Sì, quindi assegniamo 5.
– Consideriamo il terzo valore: 10 - 5 è minore di 16 - 10? Sì, quindi assegniamo 5.
– Consideriamo infine l'ultimo valore: 16 - 5 è minore di 16 - 16? No, quindi assegniamo 16.

Come precedentemente asserito, generalmente non si conosce molto riguardo la popolazione che si va a studiare e quindi inizialmente si raccolgono informazioni sul *database* che possono essere adottate per rappresentare l'intera popolazione.

I *database* vengono costruiti in forma tabellare: sulle righe vi sono i soggetti o le istanze che si vogliono analizzare mentre sulle colonne si trovano le variabili che descrivono gli elementi del *database*.

Queste ultime possono essere di diverso tipo: numeriche/testuali; nominali/ordinali; etc. Considerando la successiva tabella (Tabella 1) possiamo notare che sono presenti delle variabili di tipo categorico e non. Si definiscono "categoriche" quelle variabili che possono assumere un insieme finito di simboli. Ad esempio o si è maschio o si è femmina, o si è pensionati o no, etc. Questi attributi possono essere di due tipi:

- **Nominale**: se i suoi valori non possono essere ordinati. Non si può stabilire se maschio viene prima o dopo femmina;
- **Ordinale**: se i valori della variabile possono essere ordinati in base a una regola. Ad esempio, per la variabile istruzione è possibile definire che il dottorato è il più alto titolo, seguito dalla laurea e poi dal diploma.

ID	Età	Sesso	Istruzione	Pensionato	Reddito (€)
1	34	M	Laurea	Sì	1.300
2	59	F	Diploma	Sì	1.100
3	*null*	M	Dottorato	No	1.500
4	17	M	Altro	*null*	800

Tabella 1: Esempio di *database*, basato su quanto proposto da Dulli et al. (2009).

Non sempre è semplice stabilire se una variabile categorica è nominale o ordinale.

Un attributo numerico invece ha un dominio a valori reali come, nell'esempio considerato, il reddito o l'età.

Nella statistica univariata, accanto ai concetti base di media e varianza che riguardano comunque solo variabili numeriche, sono già presenti alcuni strumenti che permettono di raggruppare le informazioni senza perdere in correttezza (il concetto di percentile, di moda, di mediana etc.). Nel caso multivariato, diventa fondamentale riuscire a capire quali variabili sono necessarie per la corretta analisi dei dati e pertanto si utilizzano delle tecniche che permettono di aiutarci a capire quali relazioni esistono tra i dati.

Se le variabili sono solo due è possibile incrociarle in un piano cartesiano (*scatter-plot*) iniziando a visualizzare una ipotetica relazione ma se invece il numero di attributi è maggiore, questa visualizzazione è impossibile e pertanto è necessario utilizzare altre metodologie per valutare se esistono delle dipendenze tra di esse.

Nel tempo sono emerse diverse tecniche per la sintesi e l'esplorazione dei dati e tecniche per classificare gli elementi in gruppi in qualche modo omogenei.

Considerando l'esempio di Tabella 1, è possibile creare delle tabelle di sintesi per ogni variabile. Per esempio potremmo dire che per la variabile Sesso, considerando le osservazioni presentate, abbiamo rilevato 3 maschi e 1 femmina; per la variabile Pensionato ritroviamo 2 Sì e via dicendo.

In sostanza, l'analisi multivariata consente di fare analisi più sintetiche ma con una portata informativa maggiore rispetto alle analisi effettuate tramite analisi univariata.

Strumenti molto utili per l'analisi dei *database* sono le misure di dispersione che consentono di valutare quanto cambiano i dati di una variabile. Se la variabilità è alta significa che i soggetti analizzati hanno valori di quella variabile diversi; se è bassa significa al contrario che i valori dell'attributo variano poco tra i soggetti.

La variabilità è l'attitudine di un carattere a presentarsi con modalità diverse (Dulli et al., 2009). Per quanto concerne i dati qualitativi, più la distribuzione delle frequenze tra le modalità tende ad uniformarsi e minore è la variabilità; per i caratteri quantitativi, più i valori della variabile si allontanano dalla media, maggiore è la variabilità.

I principali indici che vengono utilizzati per valutare l'eterogeneità dei dati sono:

1. Il **campo di variazione** (γ) è dato:
   $$\gamma = Max(x_i) - min(x_i) \quad (3.4)$$
2. La **Varianza** è: $\sigma^2 = \frac{\sum_{i=1}^{N}(x_i-\mu)^2}{N}$ (3.5)
3. La **Deviazione Standard** è:
   $$\sigma^2 = \sqrt{\frac{\sum_{i=1}^{N}(x_i-\mu)^2}{N}} \quad (3.6)$$

Tornando all'esempio in Tabella 1 la media, la varianza e la deviazione standard della variabile Reddito si ottengono eseguendo il seguente calcolo:

$$\mu_{reddito} = \frac{1300+1100+1500+800}{4} = 1175 \quad (3.7)$$

$$\sigma^2 = \frac{(1300-1175)^2+(1100-1175)^2+(1500-1175)^2+(800-1175)^2}{4} = 66875 \quad (3.8)$$

$$\sigma_{reddito} = \sqrt{66875} = 259 \quad (3.9)$$

Quando è necessario però analizzare variabili qualitative, esistono indicatori di eterogeneità e di similarità che ci permettono di studiarne la dispersione.

Uno dei più famosi indicatori è l'indice di Gini che viene calcolato partendo da una variabile qualitativa x_i che può assumere *k* distinti livelli (ad esempio, per Tabella 1 la variabile istruzione). Le frequenze relative per ogni livello vengono definite p_i. L'**indice di eterogeneità di Gini** è:

$$G = 1 - \sum_{i=1}^{k} p_i^2 \quad (3.10)$$

Per normalizzare questo indicatore e farlo variare tra 0 e 1, viene applicata la seguente trasformazione:

$$G^N = \frac{G}{(k-1)/k} \tag{3.11}$$

Valori pari a 0 indicano assenza di eterogeneità mentre più si avvicina ad 1 e maggiore è l'eterogeneità.

Un ulteriore indicatore utilizzato è quello di **entropia di Shannon** che prende spunto dal concetto di entropia utilizzato nella teoria dell'informazione e si calcola in questo modo:

$$E = \sum_{i=1}^{k} p_i \cdot log p_i \tag{3.12}$$

Normalizzato diventa:

$$E^N = \frac{E}{log k} \tag{3.13}$$

L'interpretazione dei risultati non varia rispetto all'indice di Gini.

A titolo d'esempio consideriamo la variabile sesso, distribuita come mostrato nella tabella seguente (Tabella 2):

Sesso	**N**	p_i
M	12	0,60
F	8	0,40
Totale	20	1

Tabella 2: Esempio di variabile qualitativa

La tabella ci dice che nel *database* sono presenti 12 uomini e 8 donne (k=2, uomo o donna) e quella presentata è dunque una tabella che sintetizza le informazioni che possiamo dedurre dalla variabile sesso. Calcoliamo ora l'indice di Gini e quello di entropia, puri e poi normalizzati.

$$G = 1 - (0{,}6^2 + 0{,}4^2) = 0{,}48 \tag{3.14}$$

$$G^N = \frac{0{,}48}{(2-1)/2} = 0{,}96 \tag{3.15}$$

$$E = [0{,}6 \cdot log 0{,}6 + 0{,}4 \cdot log 0{,}4] = 0{,}2923 \tag{3.16}$$

$$E^N = \frac{0{,}2923}{0.3} = 0{,}971 \tag{3.17}$$

Si può quindi concludere che essendo entrambi gli indici vicini a 1, la distribuzione è molto eterogenea: entrambe le modalità son presenti e con frequenze equilibrate tra loro.

Una fase di pre-processamento utile al corretto funzionamento del modello è la trasformazione delle variabili in range di valori confrontabili.

Le tecniche più comuni sono la **normalizzazione** e la **standardizzazione** che hanno come obiettivo quello di ottenere un insieme di valori che hanno specifiche proprietà.

- **Normalizzazione Z-score**. Consideriamo μ_x la media di una variabile e σ_x la sua deviazione standard, la trasformazione è:

$$x' = \frac{(x-\mu_x)}{\sigma_x} \tag{3.18}$$

  In questo modo si crea una nuova variabile che ha media 0 e deviazione standard pari a 1 (cioè i parametri di una distribuzione normale standard).

- **Normalizzazione per scalatura decimale**

$$x' = \frac{x}{10} \quad \text{dove j è il più piccolo intero tale che Max|x'|<1} \tag{3.19}$$

  Se per esempio la variabile che stiamo trasformando varia da -896 a 835, normalizziamo dividendo tutto per 1000 (10^{100}). I nuovi valori varieranno da -0,896 a 0,835.

- **Normalizzazione min-Max.** I valori della variabile vengono scalati in modo che i nuovi valori cadano tra new_{min} e new_{Max}:

$$x' = \frac{x-min_x}{Max_x-min_x} \cdot (new_{Max} - new_{min}) + new_{min} \tag{3.20}$$

  Tuttavia questa trasformazione è molto influenzata dagli *outliers*.

Quando il *database* da analizzare è molto grande, si possono utilizzare delle tecniche per **ridurne le dimensioni** in modo che i dati richiedano meno spazio in memoria e siano più facili da modellare. Le principali tecniche prevedono: l'aggregazione attraverso un *data cube*[7] al più alto livello di aggregazione o la riduzione della dimensionalità con la selezione di variabili rilevanti[8].

[7] Termine legato alla teoria del *warehouse*. Un "*data cube*" è anche detto cubo, *hypercube*, vettori multidimensionali, database multidimensionali. Si tratta di una struttura dati multidimensionale, un gruppo di celle di dati organizzate secondo le dimensioni. Per esempio un foglio elettronico esemplifica un vettore a due dimensioni con le celle organizzate in righe e colonne essendo ognuna una dimensione. Un vettore tridimensionale può essere visualizzato come un cubo, dove ogni dimensione è un lato del cubo. Vettori di dimensioni maggiori non hanno metafore fisiche, ma organizzano i dati nel modo in cui gli utenti vedono l'azienda. Tipiche dimensioni aziendali sono il tempo, i prodotti, le locazioni geografiche, canali di vendita, etc. Non è raro incontrare più di 20 dimensioni; Tuttavia bisogna considerare che maggiore è il numero delle dimensioni, maggiore è la complessità nel manipolare e nel fare *Data Mining* sul cubo e maggiore può essere la sparsità del cubo. Si veda Dulli et al. (2008).

[8] Si veda in appendice per un approfondimento (p. 75).

Spesso per capire le relazioni tra i dati può risultare utile rappresentare visivamente coppie di variabili poiché questo può servire per farsi un'idea sintetica, anche se non precisa, delle principali statistiche delle variabili di un *database*.

Come suggerito da Fisher (1992) le principali motivazioni che dovrebbero spingere un analista a fare dei grafici sono le seguenti:

- Un esame preliminare delle caratteristiche della distribuzione;
- Un suggerimento per il *test* statistico da scegliere adeguato ai dati raccolti;
- Un aiuto alla comprensione delle conclusioni.

Gli **istogrammi** sono grafici che hanno sull'asse delle ascisse le misure della variabile casuale e sull'asse delle ordinate il numero assoluto, oppure la frequenza relativa o quella percentuale, con cui compaiono i valori di ogni classe.

In queste rappresentazioni ogni dato è rappresentato dalla superficie di un rettangolo. I rettangoli hanno tutti uguale base e il confronto fra le loro superfici è possibile grazie alle diverse altezze. Come per i diagrammi a linee, i rettangoli possono essere posizionati in verticale o in orizzontale. Inoltre, possono essere disegnati uno di seguito all'altro senza spazi intermedi, in modo da formare un'unica superficie (istogramma) oppure separata- mente a strisce verticali (ortogramma). Nel lessico quotidiano però si parla di istogramma in ambedue i casi.

Gli istogrammi (ortogrammi) sono utili quando bisogna rappresentare numerosi dati di cui bisogna considerare sia ogni singolo valore isolatamente sia la loro somma. Servo- no anche quando, nello stesso grafico, si vogliono operare confronti tra dati ottenuti in rileva- zioni diverse: non si fa altro che disegnare affiancati rettangoli di colori diversi per facilitare il raffronto.

La Figura 4 mostra l'istogramma dei dati presentati in Tabella 1. In particolare sull'asse delle ascisse troviamo l'ID mentre sulle ordinate il reddito.

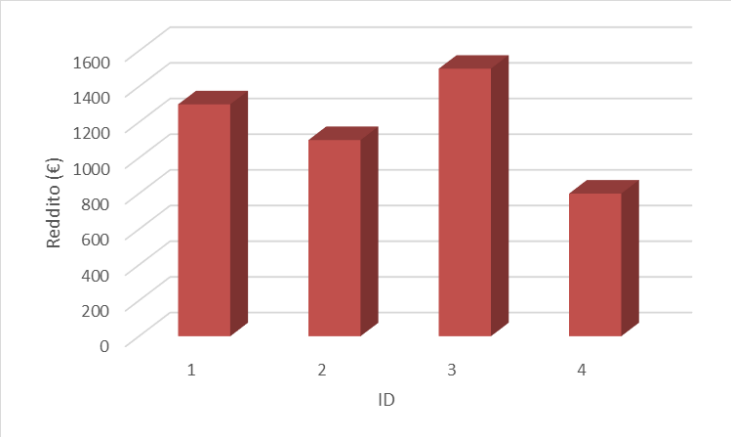


Figura 4. Istogramma su ID e Reddito, da Tabella 1

I **diagrammi a rettangoli distanziati**, o **grafici a colonne**, sono formati da rettangoli con basi uguali ed altezze proporzionali alle intensità (o frequenze) dei vari gruppi consi- derati. A differenza degli istogrammi, sull'asse delle ascisse non vengono riportati misure ordinate ma nomi, etichette o simboli, propri delle classificazioni qualitative.

La Figura 5 mostra un Diagramma a rettangoli distanziati su Istruzione e Reddito da Tabella 1.

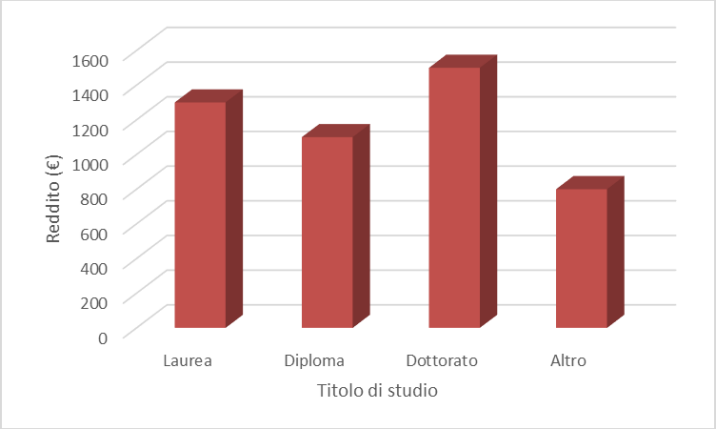


Figura 5. Diagramma a rettangoli distanziati su Istruzione e Reddito, da Tabella 1.

Il **diagramma a punti è** simile al grafico precedente ma utilizza dei punti.
La Figura 6 illustra un diagramma a punti su Istruzione e Reddito da Tabella 1.

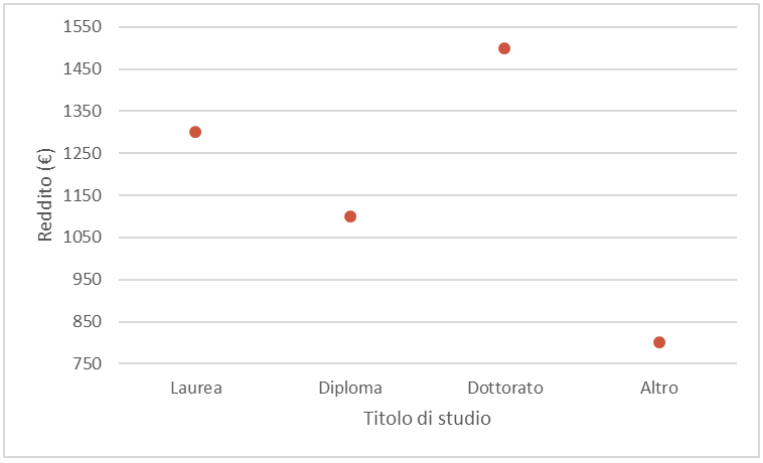


Figura 6. Diagramma a punti su Istruzione e Reddito, da Tabella 1.

Se le frequenze o le quantità di una variabile sono rappresentate da superfici, stiamo utilizzando gli areogrammi.
I più utilizzati sono i diagrammi circolari o a torta che sono costruiti dividendo un cerchio in parti proporzionali alle classi di frequenza. Vengono impiegati soprattutto per rappresentare frequenze percentuali.
La Figura 7 mostra un aerogramma su istruzione e Reddito da Tabella 1.

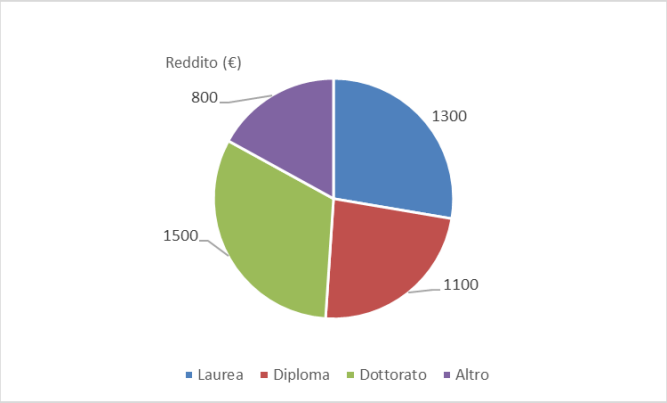


Figura 7. Aerogramma su Istruzione e Reddito, da Tabella 1.

Il **diagramma a Coordinate Polari** mostra il cambiamento e le frequenze dei dati in relazione ad un polo. Sono costruiti su una sezione circolare più o meno ampia e la loro costruzione avviene in 4 fasi:

1. Dal punto centrale si traccia una serie di cerchi (o quadrati) concentrici aventi distanza dal centro esprimente la misura dell'intensità del fenomeno;
2. Si divide l'angolo giro in tante parti uguali quante sono le modalità della serie;
3. Si segnano dei punti nei cerchi che individuano la modalità e la frequenza o l'intensità del fenomeno;
4. Se il fenomeno è continuo si uniscono i punti con segmenti di retta

La Figura 8 mostra un diagramma a coordinate polari su Istruzione e Reddito da Tabella 1.

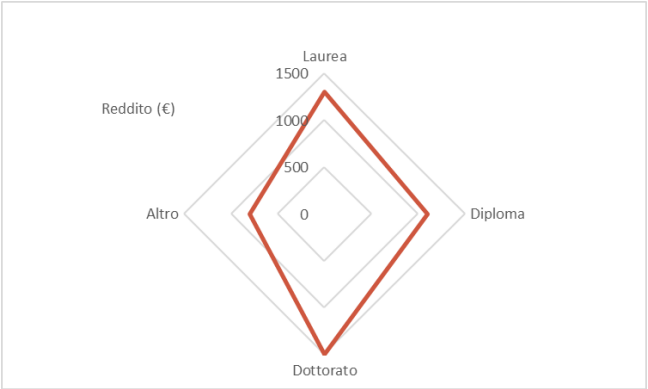


Figura 8. Diagramma a coordinate polari su Istruzione e Reddito, da Tabella 1.

Infine, è ancora possibile valutare alcune **misure di distanza e similarità** sulle variabili presenti nel *database* in modo da poter identificare differenze e somiglianze tra gli attributi.

La distanza/somiglianza tra due variabili viene valutata in termini matematici e quindi ritroviamo diverse formulazioni che variano in base al tipo di dato o al tipo di analisi.

Seguendo Dulli et al., (2009), se S è la rappresentazione simbolica di uno spazio di misura e x, y e z sono tre punti qualsiasi appartenenti ad S, si definisce una **misura di**

dissimilarità* o *semimetrica una funzione $d(x,y) : S\,xS \rightarrow$R che soddisfa le seguenti condizioni:

1. **Proprietà riflessiva**: $d(x,y) = 0$ se e solo se $x = y$;
2. **Non negatività della distanza**: $d(x,y) \geq 0 \quad \forall x,y \in S$;
3. **Proprietà simmetrica**: $d(x,y) = d(y,x) \quad \forall x,y \in S$
4. $d(x,y) \leq d(x,z) + d(y,z) \quad \forall x,y,z \in S$. Se è soddisfatta questa condizione significa che la funzione distanza è una *metrica*. Questa condizione viene chiamata **disuguaglianza triangolare** e richiede che la distanza tra i punti x ed y sia minore o al massimo uguale alla somma delle distanze tra i due punti ed un terzo punto z distinto dai precedenti.

Se S è la rappresentazione simbolica di uno spazio e x e y sono due punti qualsiasi appartenenti ad S, definiamo una **misura di *similarità*** una funzione $s(x,y) : S\,xS \rightarrow$ R che soddisfa le seguenti condizioni:

1. $s(x,y) = 1$ massima similarità se e solo se $x = y$;
2. $s(x,y) = s(y,x) \quad \forall x,y \in S$ (proprietà simmetrica).

Queste definizioni sono essenziali quando si applicano tecniche di *clustering* (si veda il paragrafo 4.2, p. 26) in quanto permettono di quantificare le somiglianze e le distanze tra elementi che appartengono a gruppi differenti.

A titolo esemplificativo, si veda la Figura 9 in cui i due *cluster* (cioè i due raggruppamenti) sono facilmente identificabili grazie a una variabile (caso semplicissimo, 10 soggetti di cui la variabile sesso è così distribuita 6 femmine e 4 maschi).

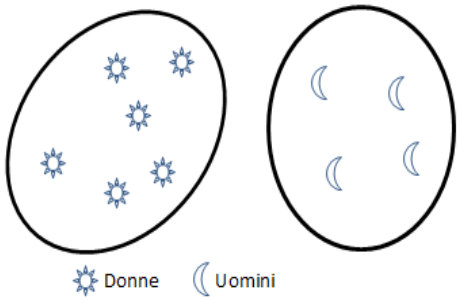


Figura 9. Esempio semplificato di *cluster*.

Le principali misure di distanza e similarità vengono qui elencate ed illustrate matematicamente in appendice (p. 75):

- Distanza Euclidea;
- Distanza di Minkowski;
- Distanza di Lagrange-Tchebychev;
- Distanza di Mhalanobis
- Distanza di Jaccard;
- Correlazione

4. LA MODELLAZIONE DEI SISTEMI DI *DATA MINING*

In questo paragrafo vengono illustrati i principali strumenti di *Data Mining* a disposizione per l'analisi dei dati ed in particolare dei modelli per la classificazione.

Tuttavia, è necessario sottolineare che la maggior parte di queste tecniche (ed in particolare, alberi decisionali e reti neurali artificiali) vengono utilizzate anche per la previsione. La "classificazione" è una filosofia che punta a realizzare algoritmi in grado di apprendere e di adattarsi alle mutazioni dell'ambiente.

Sulla base di quanto illustrato nella Figura 2 del precedente paragrafo, la classificazione si basa sul presupposto di potere ricevere, dopo aver addestrato un modello, degli stimoli dall'esterno a seconda delle scelte dell'algoritmo. Per questo motivo si parla di "apprendimento supervisionato" cioè, della capacità del modello di apprendere in modo intelligente e in seguito applicare ciò che ha appreso su una realtà diversa da quella da cui ha imparato. Questa è la capacità di "generalizzare" che è fondamentale quando si vogliono estrarre delle regole direttamente dai dati.

Le tecniche di "apprendimento non supervisionato" mirano invece ad estrarre in modo automatico da delle basi di dati le informazioni. Queste vengono estratte senza una specifica conoscenza dei contenuti che si dovranno analizzare.

4.1. Regole associative

Le regole di associazione sono delle formule logiche che permettono di associare delle scelte o dei fatti a altri fatti o scelte (Agrawal, 1993; Berry e Browne, 2006).

Queste regole permettono di individuare collegamenti in ampi insiemi di dati infatti vengono spesso utilizzate per analizzare grandi quantità di dati con l'obiettivo di ricercare associazioni utili. Strutture come i supermercati da molti anni analizzano gli acquisti dei clienti che utilizzano tessere a punti per realizzare pubblicità mirate e per migliorare l'organizzazione dei prodotti (Kantardzic, 2011).

Le regole associative partono dal presupposto di possedere un insieme di dati e che l'utente voglia cercare eventuali correlazioni interne. Spesso si vuole individuare se un certo numero di scelte possano portare con una certa probabilità minima ad effettuare una specifica scelta. Le regole hanno due parametri che definiscono la loro bontà: il *supporto* e la *confidenza*. Il primo identifica quante volte la specifica regola appare percentualmente nell'insieme di dati di test. La confidenza invece identifica la percentuale nell'insieme di dati di esempio la regola fa una previsione corretta.

Un esempio semplice per capire di che cosa si occupano le regole associative viene dal commercio. Pensiamo a due *basket market* cioè a due collezioni di oggetti acquistati o da acquistare; in particolare alle liste della spesa rappresentate in Figura 10.

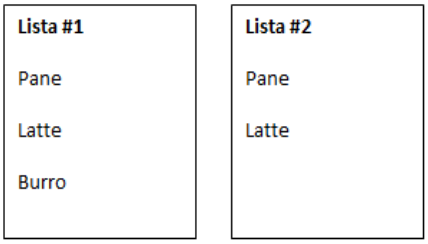


Figura 10. Liste della spesa #1 e #2.

Definiamo le coppie di associazione per queste due liste della spesa:

- Per la prima lista della spesa:
  1. Pane + Latte e Latte + Pane;
  2. Latte + Burro e Burro + Latte;
  3. Pane + Burro e Burro + Pane.

- Per la seconda:

  1. Pane + Latte e Latte + Pane.

A questo punto definiamo due parametri:

- SUPPORTO (S)= La percentuale di transazioni con x e y;
- CONFIDENZA (C)= La percentuale delle transazioni con x che hanno anche y.

Calcoliamo questi parametri per le possibili associazioni (*itemset*):

1. Pane + Latte. S=2/2 perché sia x sia y, cioè il pane e il latte sono presenti sia nella prima che nella seconda lista. C=2/2 perché x, cioè il pane, è presente nelle due liste della spesa che hanno anche y, cioè il latte;
2. Latte e Pane. S=2/2; C=2/2. La spiegazione è identica alla precedente;
3. Pane e Burro. S=1/2 perché solo in una lista della spesa sono presenti entrambi. C=1/2 perché y, cioè il burro, compare in una sola delle due liste della spesa in cui compare però la x, cioè il pane;
4. Burro e Pane. S=1/2: la spiegazione è uguale al punto precedente. C=1/1 perché y, cioè il pane, compare nell'unica lista in cui compare anche la y, cioè il burro;
5. Latte e Burro. S=1/2; C=1/2. La spiegazione è uguale a "Pane e Burro";
6. Burro e Latte. S=1/2; C=1/2. La spiegazione è uguale a "Burro e Pane";

Questi risultati ci dicono che c'è il 100% delle probabilità che chi compera pane comperi anche latte (S=2/2) e che il 100% di coloro che comperano pane, acquistano anche latte. Questo risultato è applicabile anche al secondo caso (C=2/2).

Per quanto riguarda invece Pane + Burro, possiamo dire che pane e burro sono comperati insieme dal 50% dei clienti (S=1/2) e che il 50% di coloro che comperano pane, acqui- stano anche burro (C=1/2).

Il quarto caso ci dice che burro e pane son comperati insieme dal 50% delle persone (S=1/2), mentre il 100% delle persone che acquistano burro acquista anche pane (C=1/1). Questo è un caso ovviamente semplificato che però rende facilmente l'idea di quanto siano complesse le tecniche associative e allo stesso tempo molto utili.

Gli algoritmi funzionano sempre in due fasi: nella prima vengono individuati gli itemset, che ricordiamo possono essere molti e non facilmente individuabili; nella seconda fase si valutano invece le regole e quindi il supporto e la confidenza in modo da poter generare delle vere e proprie regole associative tra gli oggetti analizzati[9].

Elenchiamo qui i principali algoritmi che vengono utilizzati senza scendere nel dettaglio tecnico. La maggioranza dei *software*[10] hanno implementato direttamente delle *routine* capaci di utilizzare gli algoritmi citati: algoritmo originario, algoritmo apriori (Agrawal et al., 1996; Han e Kamber, 2001), algoritmo Direct Hashing Pruning (Park et al., 1995), Dynamic Itemset Counting (Brin et al., 1997), algoritmo partition (Savasere et al., 1995), algoritmi con campionamento (Toivonen, 1996), algoritmi paralleli (Agrawal e Shafer, 1996), algoritmi incrementali (Cheung et al., 1996), algoritmi Pincer-search (Lin e Kedem, 1998), algoritmo seq (Meo, 1999).

4.2. *Cluster Analysis*

La *Cluster Analysis* viene anche chiamata Segmentation Analysis o *Taxonomy Analysis* e ha come obiettivo quello di creare gruppi (o *clusters*) di dati (Gan et al., 2007). Questi sono formati in modo tale che gli oggetti nello stesso cluster siano molto simili tra di loro, mentre oggetti in cluster differenti siano differenti. Le misure di similarità/dissimilarità vengono calcolate in base alle metriche che sono state illustrate nel precedente paragrafo e nell'appendice (p. 75).

Le principali tecniche di *clustering* che verranno di seguito illustrate sono[11]:

1. ***Hierarchical Clustering***: raggruppa i dati in base a diverse metriche per creare dei *cluster tree* o *dendrogram*. L'albero (*tree*) non rappresenta un unico insieme di gruppi (*cluster*) ma piuttosto una gerarchia multilivello in cui i *cluster* ad un livello sono collegati a quelli del livello successivo (Johnson, 1967);
2. ***k-means Clustering***: è un "partitioning method". Attraverso questa tecnica si creano delle partizioni di dati in *k* gruppi mutualmente esclusivi. Diversamente dal *hierarchical clustering*, il *k-means clustering* opera sulle osservazioni attuali e crea un unico livello di gruppi. L'algoritmo *k-means* è dunque più adatto quando ci si trova ad analizzare una grande quantità di dati (Hartigan e Wong, 1979);
3. ***Gaussian Mixture Models***: formano i *cluster* usando funzioni di densità di probabilità normali partendo dall'intera popolazione. I *cluster* sono assegnati selezionando il componente che massimizza la probabilità a posteriori che indica per ogni osservazione del *database* la probabilità di appartenenza ad ogni *cluster*. Come *k-means*, questo metodo è iterativo e converge a un ottimo locale. Questa tecnica è più appropriata di *k-means* quando i *cluster* hanno dimensioni e correlazioni diverse tra loro (Banfield e Raftery, 1993).

---

[9] In appendice (p. 78) viene illustrato l'algoritmo apriori che risulta essere una delle procedure più utilizzate.

[10] I principali: ARMiner (http://www.cs.umb.edu/~laur/ARMiner/); Azmy SuperQuery (http://www.azmy.com/); SPSS Modeler (http://www-01.ibm.com/software/analytics/spss/ products/modeler/); LPA *Data Mining Toolkit* (http://www.lpa.co.uk/dtm.htm); Magnum Opus (http://www.giwebb.com/); PolyAnalyst (http://www.megaputer.com/site/index.php); SamrtBundle (http://www.decidyn.com/SmartBundle.php?id=91005); WizSoft (http://www. wizsoft.com/); XpertRule *Data Mining* (http://www.xpertrule.com/); XAffinity (http://www. xore.com/); Arule (open source, pacchetto per R-statistic, http://cran.at.r-project.org/web/ packages/arules/index.html). Sul sito di Christian Borgelt (http://www.borgelt.net//index. html) nel menù *software* ci sono molte applicazioni open source per il *Data Mining*, per *l'Intelligent Data Analysis* e per le regole associative.

[11] L'esempio riportato è stato creato con Matlab R2010a.

4.2.1. *Hierarchical Clustering*

La procedura per creare un *Hierarchical Cluster* è la seguente:

1. **Trovare le similarità/dissimilarità tra ogni coppia di oggetti del *dataset***.
   In questo step si calcola la distanza tra gli oggetti in base a una delle metriche illustrate in precedenza e in appendice (p. 75 dell'appendice).
2. **Raggruppare gli oggetti in un *cluster tree* gerarchico e binario**.
   In questa fase, si collegano le coppie di oggetti che sono più vicine attraverso una funzione (chiamata *linkage*) che usa le informazioni sulle distanze. Appena gli elementi sono accoppiati in *cluster* binari, i gruppi formati vengono nuovamente raggruppati finché l'albero non viene costruito.
3. **Determinare dove "potare" il *hierarchical tree* in *clusters***.
   In questa fase viene utilizzata una funzione che "pota" i rami dell'albero partendo dal più alto e assegnando tutti gli oggetti sotto ogni "potatura" a un solo *cluster*. Questo crea una vera e propria partizione dei dati.

Qui di seguito un esempio di questa tecnica:

- Generiamo un database di numeri casuali formato da 5 elementi di cui conosciamo due caratteristiche (Tabella 3).

0,814	0,097
0,905	0,278
0,127	0,546
0,913	0,957
0,632	0,964

Tabella 3. Campione casuale x di 5 elementi e 2 variabili.

- Calcoliamo la distanza euclidea[12] tra gli elementi e visualizziamola in forma matriciale (Tabella 4), affinché si capisca meglio l'interpretazione. Questa matrice indica la distanza euclidea tra ogni elemento e per questo è simmetrica con diagonale principale nulla.

0	0,202	0,821	0,863	0,886
0,202	0	0,823	0,679	0,738
0,821	0,823	0	0,887	0,655
0,863	0,679	0,887	0	0,281
0,886	0,738	0,655	0,281	0

Tabella 4: Distanze euclidee

  La distanza tra un elemento e sé stesso è zero ma, partendo dalle colonne, possiamo vedere che la distanza tra il primo elemento (prima colonna) e il secondo (seconda riga) è pari a 0,202. La distanza tra il terzo elemento (terza colonna) e il quinto (quinta riga) è pari a 0,655.

- Applichiamo l'algoritmo *linkage* ed otteniamo il risultato presentato nella Tabella 5. Questa funzione parte dalle distanze ottenute nello *step* precedente e in particolare dalla

[12] Si veda l'appendice a p. 75 per la descrizione delle differenti metriche.

più piccola. Minore è la distanza e più i due oggetti sono simili, quindi raggruppabili in un *cluster*.

1	2	0,202
4	5	0,281
3	7	0,655
6	8	0,679

Tabella 5. Algoritmo Linkage.

Nelle prime due colonne si trovano gli elementi che sono stati uniti, mentre nella terza il valore della distanza.

Nel nostro caso sono stati formati 4 *clusters*: il primo costituito dall'elemento 1 e dall'elemento 2; il secondo composto dall'elemento 4 e dall'elemento 5; e così via. Tuttavia, si noti che il terzo *cluster* risulta formato dal terzo elemento e dal settimo... ma il *database* è formato da 5 soli elementi... chi è il numero 7? L'algoritmo *linkage* genera dei *cluster* partendo dai primi raggruppamenti fatti e li numera m+1 (dove m è il numero di elementi del *database*, cioè 5 nel nostro caso). Guardando i risultati vediamo che vi sono ben 3 elementi nuovi, il 6 che è dato dal *cluster* 1+2, il 7 che è dato da 4+5 e l'8 che è dato da 3 + (4+5). La Figura 11 mostra come *linkage* numera i *clusters*. Il "dendrogramma" relativo ai *clusters* è rappresentato in Figura 12.

Come si può notare dal dendrogramma, la distanza tra gli oggetti 1 e 2 è quella inferiore.

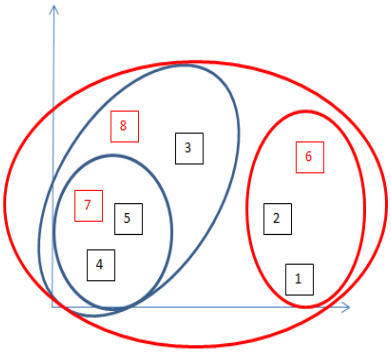


Figura 11. *Linkage* e *cluster*.

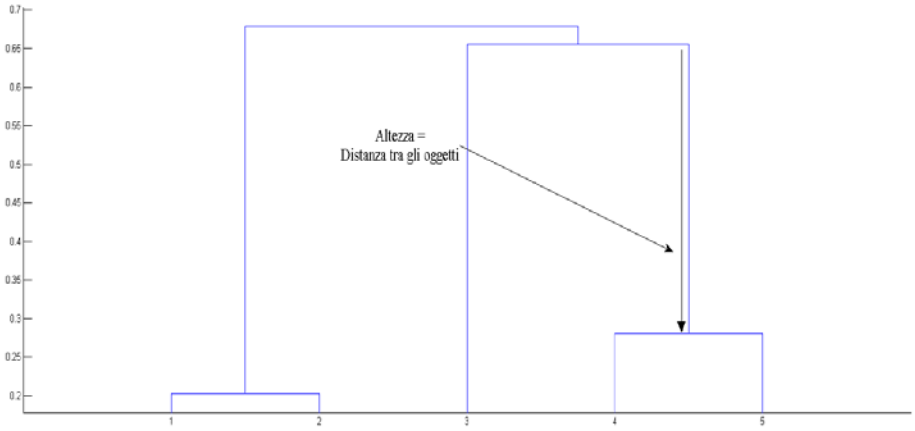


Figura 12. Il dendrogramma.

- Si passa quindi alla verifica della dissimilarità. L'altezza del "baffo" (o ramo) che unisce due oggetti del dendrogramma stabilisce la distanza tra gli stessi e prende il nome di *cophenetic distance*. Un primo modo per capire se il dendrogramma rappresenta bene i dati originari è quello di confrontare questi baffi con le distanze iniziali calcolate nel primo *step* (ricordiamoci che abbiamo usato la distanza euclidea e che il risultato potrebbe essere differente utilizzando una diversa metrica). Il *cophenetic correlation coefficient* calcola la correlazione tra le distanze: maggiore è la correlazione e migliore è il *clustering* ottenuto. Nel nostro caso questo indicatore è pari a 0,9459 e quindi possiamo essere soddisfatti del *clustering* effettuato.
  Questo indicatore può essere utilizzato per capire quale metrica sia la migliore da usare. Difatti, provando ad utilizzare la *distanza di Mahalanobis*, che come è mostrato in appendice (p. 75), genera risultati decisamente differenti da quella euclidea; difatti, in questo caso si ottiene il dendrogramma di Figura 13 con un indice di Cophenet pari a 0,9395 che è inferiore a quello ottenuto con la distanza euclidea.

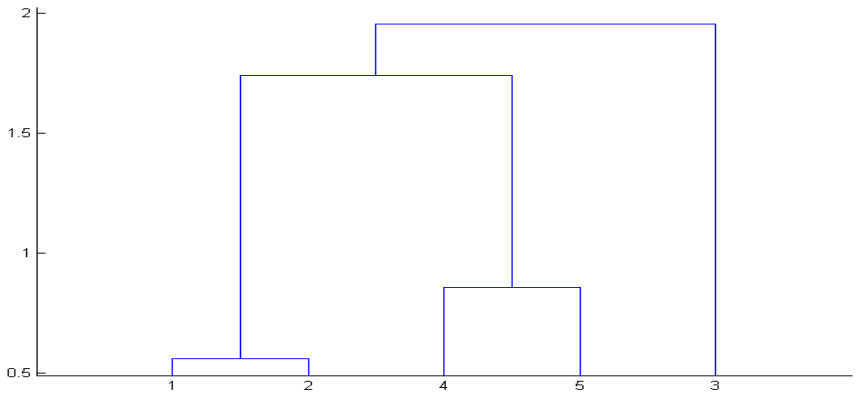


Figura 13. Il dendrogramma, distanza di Mahalanobis.

- Si calcola l'indice di inconsistenza. Questo indicatore confronta l'altezza di un baffo in un *cluster* con la media delle altezze dei baffi che stanno sotto. I *link* (baffi) che legano *cluster* distinti avranno un elevato indice di inconsistenza, mentre se i *cluster* uniti sono indistinti, presenteranno un basso valore.
  Nel nostro esempio l'indice ha i seguenti valori di inconsistenza: 0; 0; 0,7071; 0,6200. Guardando la figura del dendrogramma (Figura 14) si nota come gli elementi 1, 2, 4 e 5 non presentino dei baffi sotto di loro e per questo vengono chiamati *leaf nodes*. I due gruppi formati da questi elementi (cioè il 6 e il 7) avranno un indice di inconsistenza pari a 0. Il terzo *cluster*, formato dall'elemento 3 e dal 7 (4+5), presenta un indice di inconsistenza pari a 0,7071. Anche per l'ultima coppia (6+8) l'indice di inconsistenza è maggiore di 0 e raggiunge il valore di 0,6200.
  Questo indicatore, che apparentemente è difficile interpretazione, ci consentirà nel prossimo *step* di potare il nostro albero in base a una soglia stabilita.
- A questo punto, non abbiamo ancora terminato perché dobbiamo potare l'albero rappresentato nel dendrogramma per costruire gli effettivi *cluster* non binari (ricordiamoci che per ora i nostri gruppi son formati da due elementi).
  Ora esistono due possibilità: trovare la naturale divisione dei dati oppure specificare arbitrariamente il numero di *clusters* in cui vogliamo che i dati siano suddivisi.

1. Il primo metodo prevede di stabilire una soglia (indice di inconsistenza) secondo cui tagliare l'albero e creare i *cluster*. Se ad esempio stabiliamo una soglia di 0,7 avremo tre *cluster* così formati:
   - Elemento 1 + Elemento 2;
   - Elemento 4 + Elemento 5;
   - Elemento 3

   La Figura 14: mostra che cosa fa idealmente questo algoritmo. Taglia in orizzontale ad una certa soglia i rami (o baffi) dell'albero. Rimangono i due *cluster* formati da due elementi e il *cluster* formato da uno solo.

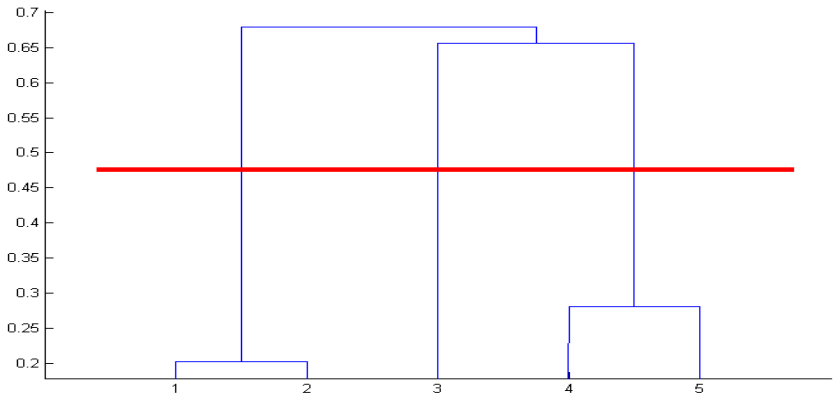


Figura 14. Il dendrogramma e la potatura.

2. La seconda soluzione prevede di decidere in partenza quanti *cluster* vogliamo (per esempio, vogliamo ridurre la numerosità del *database*). Impostando un numero

massimo di *cluster* pari a 2, l'algoritmo crea un gruppo formato dai primi due elementi e un secondo gruppo formato dagli ultimi tre.

4.2.2. *k-means Clustering*

Il *k-means clustering* è un algoritmo di partizionamento che suddivide i dati in *k* gruppi mutualmente esclusivi.

Attraverso questa tecnica ogni osservazione del *dataset* viene considerata come un oggetto che ha una precisa posizione nello spazio. In particolare ricerca la suddivisione in modo tale che gli oggetti all'interno di ogni *cluster* siano il più vicino possibile gli uni gli altri. Anche in questo caso, come si può intuire, la definizione della metrica (distanza) utilizzata per costruire il *cluster* è determinante per la buona riuscita.

Ogni *cluster* è definito non solo dai suoi oggetti ma anche dal suo "centroide" (o centro) che rappresenta il punto in cui la somma delle distanze tra tutti gli oggetti di quel *cluster* è minimizzata. Ovviamente il centroide cambia il base alla metrica utilizzata per calcolare la distanza tra i punti.

In pratica, l'algoritmo *k-means* calcola iterativamente il centroide per ogni *cluster*, spostando gli oggetti da un gruppo ad un altro finché la somma delle distanze non è sufficientemente piccola. Il risultato è dato da *cluster* compatti e molto differenti tra di loro.

Per mostrare come funziona questo algoritmo utilizziamo un *dataset*[13] random formato da 560 righe e 4 colonne. Significa che stiamo analizzando 560 elementi e che per ognuno conosciamo 4 caratteristiche.

Utilizziamo la distanza euclidea e chiediamo all'algoritmo *k-means* di creare 3 gruppi:

1. Il primo *cluster* raccoglie 214 elementi;
2. Il secondo gruppo è formato da 144 soggetti;
3. Il terzo *cluster* raggruppa 202 elementi.

Nella Figura 15 vediamo una rappresentazione dei *cluster* insieme al loro valore di "silhouette". Questo valore indica quanto sono indipendenti i *cluster*. Dala figura si evince che il terzo *cluster* ha il valore più elevato di silhouette (maggiore di 0,6) e questo indica che il *cluster* è abbastanza separato dai vicini.

Il primo *cluster* invece contiene molti punti con basso valore di silhouette e il secondo gruppo ha degli elementi con valori negativi. Questo risultato indica che i due *cluster* non sono ben separati. Per decidere quale sia il numero migliore di *cluster* bisogna procedere a tentativi. Ad esempio, con 4 *cluster* la *silhouette* è rappresentata in Figura 16.

[13] Disponibile all'interno del *software* Matlab.

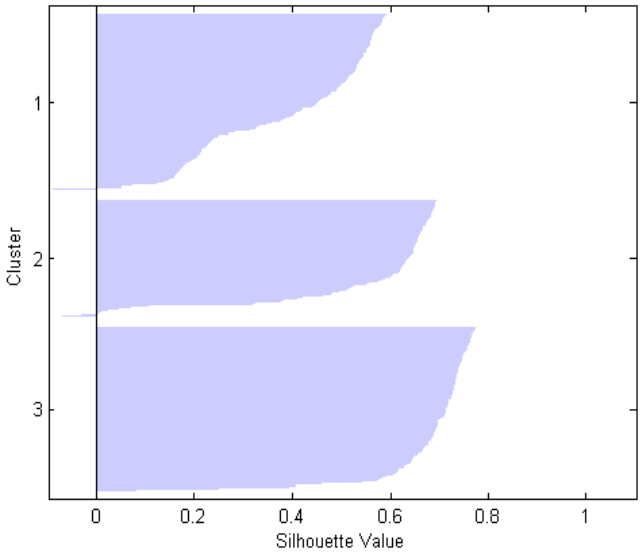


Figura 15. I *cluster* e la loro "silhouette" (3 gruppi).

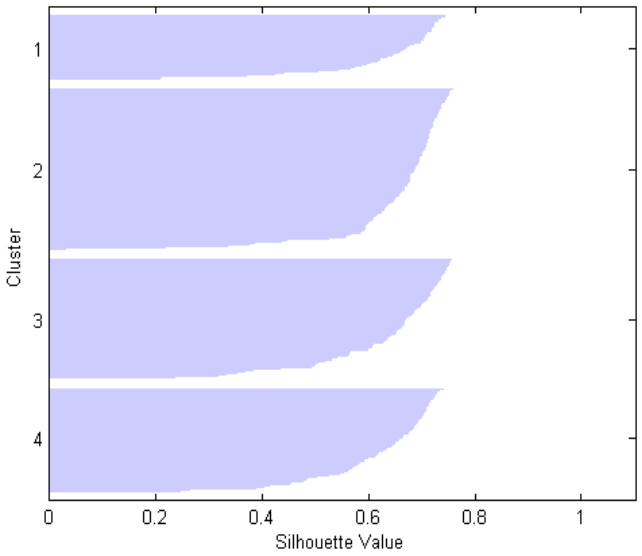


Figura 16. I *cluster* e la loro "silhouette" (4 gruppi).

Come si può vedere non vi sono valori negativi ma un modo per capire quale delle due strategie sia la migliore è quello di calcolare la media della silhouette per i 3 e i 4 gruppi. La media sui 3 gruppi è 0,5352, sui 4 è 0,64. Essendo maggiore quella per i 4 gruppi, questa seconda suddivisione è da considerarsi migliore della prima.

Si ricordi che il risultato cambia anche in base alla metrica utilizzata per calcolare la distanza tra gli elementi. Se infatti proviamo ad utilizzare la metrica legata all'indice di correlazione e formiamo quattro gruppi, avremo una media della silhouette pari a 0,6514, quindi ancora meglio.

4.2.3. *Gaussian Mixture Models*

Questo algoritmo è piuttosto complesso e utilizza le funzioni normali per creare i *cluster*. Creiamo una combinazione di due bivariate (formata da due variabili) distribuite normalmente di 200 componenti ciascuna. I punti vengono rappresentati nella Figura 17. A questo punto attraverso il metodo della massima verosimiglianza (*maximum likelihood*) vengono iterativamente stimate le densità di probabilità (linee colorate in Figura 18). Si può notare con facilità che i dati si distribuiscono principalmente in due gruppi. A questo punto ogni punto viene assegnato ad un *cluster*, ottenendo il risultato presentato in Figura 19. L'algoritmo utilizzato per creare i *cluster* assegna i punti ad ogni gruppo basandosi sulla probabilità posteriore stimata[14]. Ogni punto è assegnato al *cluster* che corrisponde al più alto valore di probabilità posteriore. In Figura 20 vengono mostrati i *cluster* colorati in base alla loro probabilità posteriore.

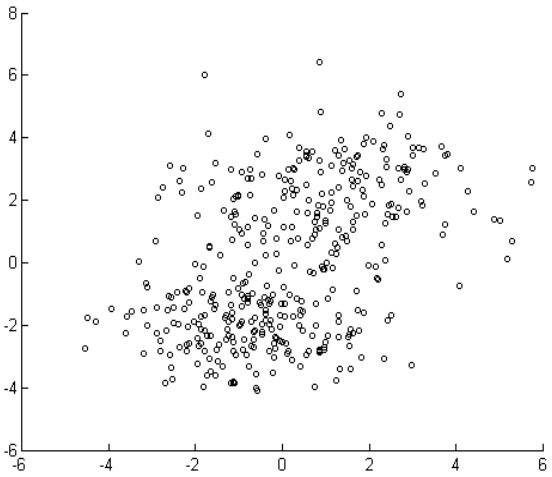


Figura 17. Distribuzione delle bivariate.

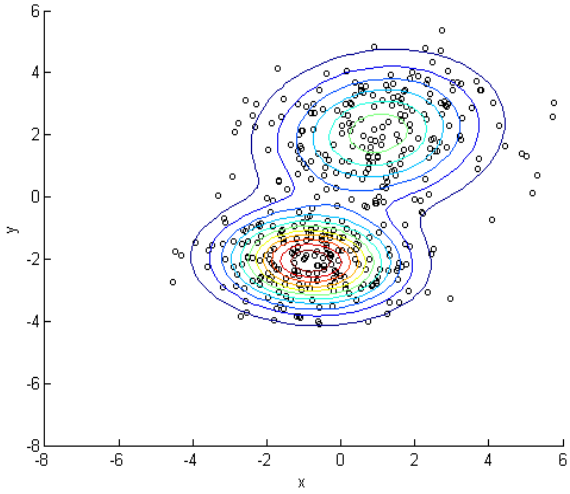


Figura 18. Densità di probabilità.

[14] In statistica Bayesiana, la "probabilità a priori" è il numero di azioni che (teoricamente) portano all'evento rapportato al numero totale di azioni. La "probabilità a posteriori" è il numero di azioni favorevoli rapportato al numero di azioni totali, si tratta quindi di una frequenza.

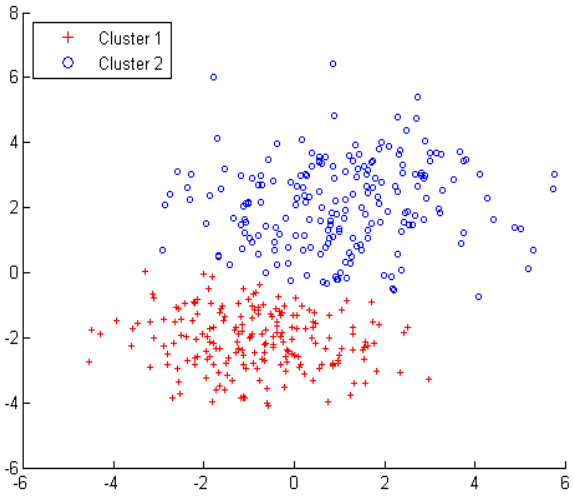


Figura 19. I due *clusters*.

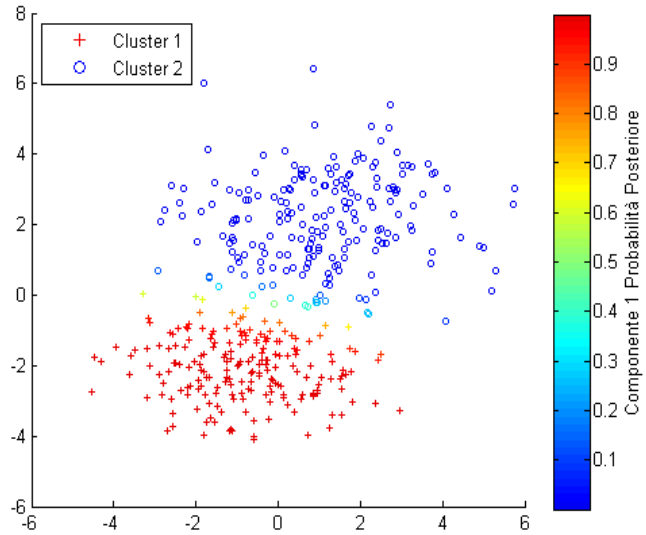


Figura 20. I *cluster* e la probabilità posteriore.

4.3. Classificatori Bayesiani

La metodologia che si basa sui classificatori Bayesiani sfrutta il calcolo probabilistico e pertanto la stima delle probabilità devono essere le più precise possibili.

Supponiamo di essere dei bambini in un laboratorio di dolci. Il macchinario che stiamo osservando glassa le torte con cioccolato bianco o cioccolato fondente. Questo affascinante impianto fa uscire le torte glassate in modo casuale e pertanto non si sa con certezza se uscirà un dolce nero o bianco. Identifichiamo con ω la variabile "glassa"; questa può assumere due valori ω_1 per la glassa bianca e ω_2 per quella fondente. Ricordiamoci che non sappiamo i valori reali di questa variabile e pertanto sarà necessario identificarli in termini probabilistici: con $P(\omega_1)$ identifichiamo la probabilità a priori che il prossimo dolce sia bianco, con $P(\omega_2)$ che sia

fondente. Essendo le glasse solo due, la somma delle due probabilità sarà pari a 1. A questo punto, è possibile definire una regola di questo tipo: se la probabilità che esca una torta glassata con cioccolato bianco è maggiore o uguale alla probabilità che sia prodotta una torta fondente, allora uscirà una torta bianca, altrimenti nera. Formalmente, la regola può essere scritta come:

$$\omega_1 \text{ se } P(\omega_1) \geq P(\omega_2) \text{ altrimenti } \omega_2 \tag{4.1}$$

Tuttavia, siccome vogliamo massimare la probabilità di indovinare il tipo di torta che uscirà cerchiamo qualche informazione che ci aiuti a stimare il più precisamente possibile la probabilità di ω_1 e ω_2. Per questo motivo, la nostra decisione sarà basata anche su una variabile casuale *x* che sarà distribuita in modo condizionato. La sua probabilità infatti sarà "condizionale": $P(x|\omega)$, cioè la probabilità di *x* dato un determinato valore della variabile ω.

Ipotizziamo che la probabilità a priori $P(\omega_j)$ e la probabilità condizionale $P(x|\omega_j)$.siano note, la probabilità di un modello ω_j può essere definita come la ricerca di una probabilità a posteriori che serve da ipotesi per la classificazione nel restituire la classe corrispondente all'ipotesi vincente:

$$P(\omega_j,x) = P(\omega_j|x) \cdot P(x) = P(x|\omega_j) \cdot P(\omega_j) \tag{4.2}$$

Bayes riformula la (4.2) nel modo seguente (Berger, 1985; Bernardo e Smith, 2009):

$$P(\omega_j|x) = \frac{P(\omega_j|x) \cdot P(\omega)}{P(x)} \tag{4.3}$$

e quindi:

$$probabilità\ a\ posteriori = \frac{probabilità\ a\ priori \cdot probabilità\ condizionale}{probabilità\ di\ x} \tag{4.4}$$

La Tabella 6[15] riporta i dati di 10 clienti che hanno risposto ad alcune proposte promozionali. Per ogni soggetto si conosce il sesso (caratteristica individuale) e l'accettazione (Sì) o il rifiuto (No) alla promozione suggerita.

In termini tecnici possiamo dire che il *training set* contiene 10 coppie dati-classe per cui possiamo dire che il primo cliente è un maschio (classe) che ha accettato solo la promozione sulle riviste e ha rifiutato tutte le altre proposte.

Poniamoci come obiettivo quello di sapere, date le risposte alle promozioni, se il cliente è maschio o femmina.

[15] L'esempio è ispirato a quanto proposto da Dulli et al. (2009).

Rivista (R)	Auto (A)	Assicurazione vita (V)	Carta di credito (CC)	Sesso
Sì	No	No	No	Maschio
No	Sì	No	Sì	Femmina
Sì	Sì	Sì	Sì	Maschio
Sì	No	Sì	No	Maschio
Sì	No	Sì	No	Femmina
No	No	No	No	Maschio
Sì	Sì	Sì	Sì	Femmina
No	No	No	No	Femmina
Sì	No	No	Sì	Maschio
No	Sì	Sì	No	Femmina

Tabella 6. Esempio di classificazione di Bayes.

Per calcolare la probabilità che il nuovo cliente sia maschio o femmina in base alle risposte date alle offerte promozionali, utilizziamo quanto proposto dal teorema di Bayes.

Per calcolare le probabilità a priori e condizionata, è necessario costruire una tabella con le frequenze delle classi rispetto alle variabili.

La Tabella 7 mostra per esempio che quasi tutti i clienti di sesso maschile hanno risposto positivamente alla promozione legata alla rivista, mentre solo 1 ha accettato quanto proposto per l'auto mentre le femmine non sembra esserci un'evidenza chiara. Tuttavia, anche se la tabella sembra dare alcune indicazioni, queste non sono ancora sufficienti per capire, date le risposte del nuovo cliente, quale sarà la sua classe (i.e., maschio o femmina).

Risposta	R(M)	R(F)	A(M)	A(F)	V(M)	V(F)	CC(M)	CC(F)
Sì	4	2	1	3	2	3	2	2
No	1	3	4	2	3	2	3	3
Sì/Tot,	4/5	2/5	1/5	3/5	2/5	3/5	2/5	2/5
No/Tot.	1/5	3/5	4/5	2/5	3/5	2/5	3/5	3/5

Tabella 7. Frequenze delle classi rispetto agli altri attributi.

Supponiamo che il nuovo cliente sia femmina (ω') e prendiamo in considerazione le sue risposte (*x'*): Rivista = Sì; Auto = Sì; Vita = Sì; Carta di credito = No.

La probabilità a priori $P(\omega)$ corrisponde al rapporto tra la frequenza della classe rispetto al totale degli individui di quella classe e quindi, essendo i maschi 5, la $p(\omega)$ sarà pari a 5/10 = 0,5. Per misurare la probabilità condizionale $P(x|\omega)$ iniziamo a calcolare quanti maschi hanno comprato in seguito alla prima promozione, quanti alla seconda, alla terza e alla quarta:

P(Rivista = Sì |Sesso = Maschio) = 4/5
P(Auto = Sì |Sesso = Maschio) = 1/5
P(Vita = Sì|Sesso = Maschio) = 2/5
P(Carta di credito = No|Sesso = Maschio) = 3/5

$$P(x|\text{Sesso=Maschio}) = \frac{4}{5}\cdot\frac{1}{5}\cdot\frac{2}{5}\cdot\frac{3}{5} = \frac{24}{625}$$

Ripetiamo lo stesso procedimento per le femmine:

P(Rivista = Sì |Sesso = Femmina) = 2/5
P(Auto = Sì|Sesso = Femmina) = 3/5
P(Vita = Sì|Sesso = Femmina) = 3/5
P(Carta di credito = No|Sesso = Femmina) = 3/5

$$P(x|Sesso=Femmina)=\frac{2}{5}\cdot\frac{3}{5}\cdot\frac{3}{5}\cdot\frac{3}{5}=\frac{54}{625}$$

I risultati ottenuti mostrano che dei dati per un maschio è minore che per una femmina, poiché 24/625 < 54/625. Tuttavia, Bayes ci ricorda che è necessario considerare anche la probabilità a priori:

$P(Sesso = Maschio) = 5/10 = ½$ e $P(Sesso = Femmina) = 5/10 = 1/2$

e quindi:

$$P(x|Sesso=Maschio)\cdot P(Sesso=Maschio)=\frac{24}{625}\cdot\frac{1}{2}=0{,}0192$$

$$P(x|Sesso=Femmina)\cdot P(Sesso=Femmina)=\frac{54}{625}\cdot\frac{1}{2}=0{,}0432$$

La conclusione è che:

$$P(Sesso=Maschio|x)=\frac{0{,}0192}{P(x)}$$

$$P(Sesso=Femmina|x)=\frac{0{,}0432}{P(x)}$$

Quanto ottenuto sembra indicare che il prossimo cliente sarà femmina e se volessimo calcolare la probabilità, basta procedere moltiplicando le frequenze totali relative alle risposte del cliente ideale. Dalla Tabella 7, i soggetti totali che hanno risposto positivamente alla promozione sulle riviste sono 6 su 10 e procedere così per le altre risposte[16]:

$$P(x)=\frac{6}{10}\cdot\frac{4}{10}\cdot\frac{5}{10}\cdot\frac{6}{10}=\frac{720}{10000}=0{,}072$$

e usarla come denominatore, ottenendo:

P(Sesso=Maschio|x) = 0,267 < P(Sesso=Femmina|x) = 0,6

[16] Queste frazioni sono state ottenute sommando i risultati delle probabilità condizionali per maschio e femmina appena calcolate.

4.4. Alberi decisionali

Gli alberi decisionali (o *decision tree*) sono sistemi che creano delle classificazioni di un set di osservazioni sulla base di alcune caratteristiche determinanti.

Le caratteristiche principali di un albero decisionale sono i *nodi* che rappresentano i sottoinsiemi direttamente derivanti dalla prima classificazione e le *foglie* che invece rappresentano l'output finale dell'albero.

Ogni nodo rappresenta una variabile (o attributo), mentre i rami dell'albero (o archi) assumono i possibili valori dell'attributo. Seguendo la ramificazione, l'osservazione sotto analisi trova la propria classificazione una volta raggiunta la foglia.

Per capire come funziona un albero decisionale analizziamo un *dataset* di 150 misurazioni di iris di cui si conosce la specie (setosa, versicolor, virginica), la lunghezza e la larghezza dei sepali e la lunghezza e la larghezza dei petali[17].

La Figura 21 riporta le foto di tre specie con indicazione di petalo e sepalo, mentre i dati sono rappresentati nella Figura 22. Un albero decisionale funziona seguendo regole tipo la seguente: "se la lunghezza del sepalo è inferiore a 5,45, classifica la specie come setosa". Inoltre, essi non richiedono alcuna assunzione circa la distribuzione delle variabili in ogni classe e pertanto hanno gli stessi vantaggi delle tecniche non parametriche. Trattandosi di algoritmi di classificazione supervisionati, si ricorda che noi conosciamo di che tipo è l'iris che analizziamo e che utilizzeremo questa informazione per migliorare il più possibile l'algoritmo e fare in modo che l'albero sia in grado di classificare correttamente ogni iris analizzata.

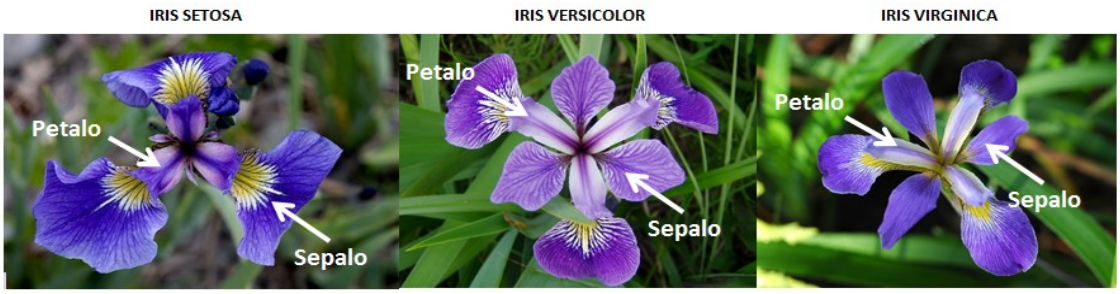


Figura 21. Le tre specie di iris.

[17] Il *dataset* Iris è stato creato da Ronald Fisher e da Edgar Anderson nel 1936. Questo *database* viene largamente utilizzato nelle applicazioni di *machine learining* ed è disponibile in diverse versioni nella maggioranza dei *software* statistici, tra cui Matlab che è l'applicazione utilizzata in questo esempio).

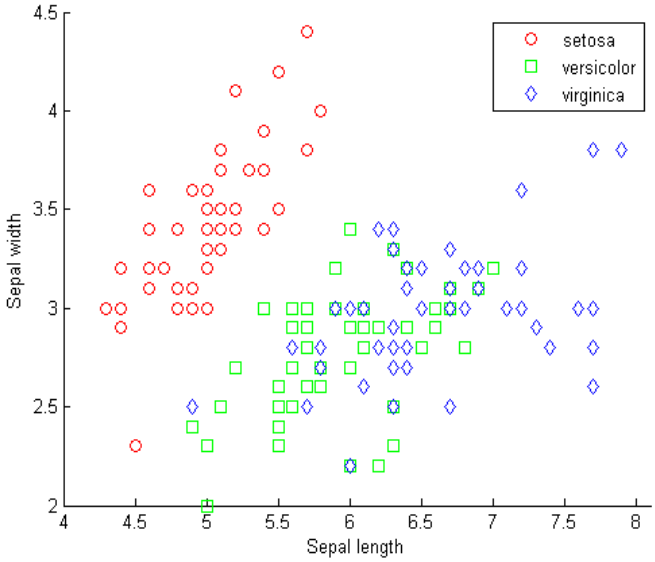


Figura 22. Statistica su *database* iris.

Attraverso un processo iterativo di regressione l'algoritmo crea un albero stabilendo delle regole che bisogna seguire nella classificazione. L'albero che ne deriva è mostrato in Figura 23 in cui SLU rappresenta la lunghezza del sepalo e SLAR la larghezza. L'albero parte dalla regola definita prima ed è formato da 19 nodi. Se l'osservazione soddisfa la relazione, allora si prende il ramo sinistro; altrimenti il destro.

A questo punto, si applica un algoritmo di *cross-validation* per migliorare e semplificare la procedura di classificazione. Inoltre si attua contemporaneamente la potatura dell'albero cercando il numero di nodi minimo che consenta di ottenere il risultato migliore.

La Figura 24 mostra il migliore albero potato.

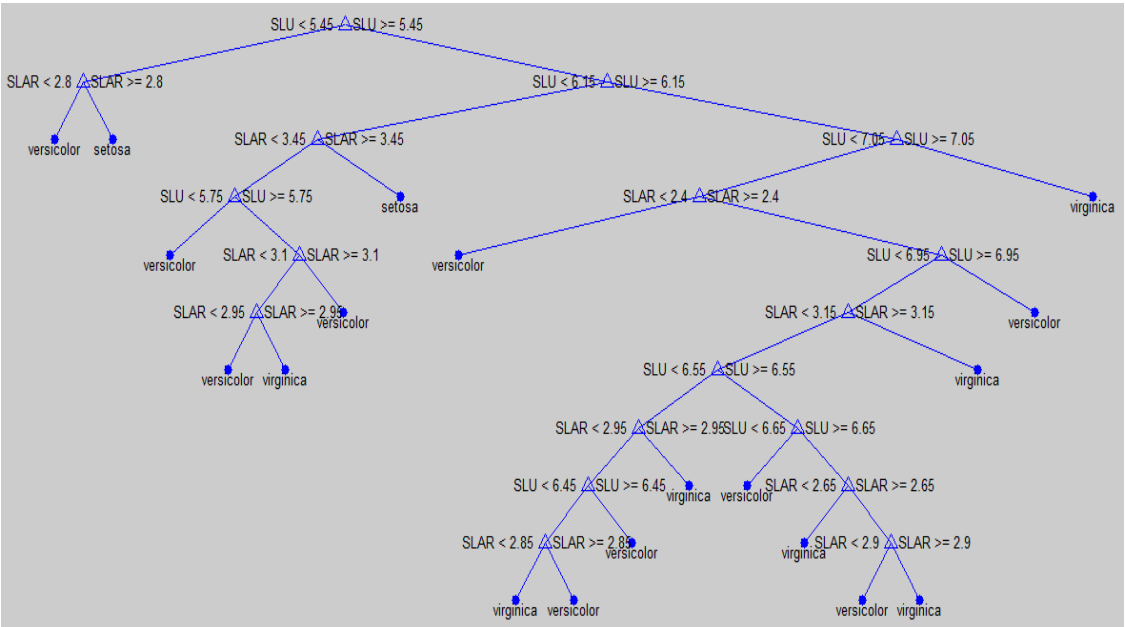


Figura 23. L'albero decisionale iniziale.

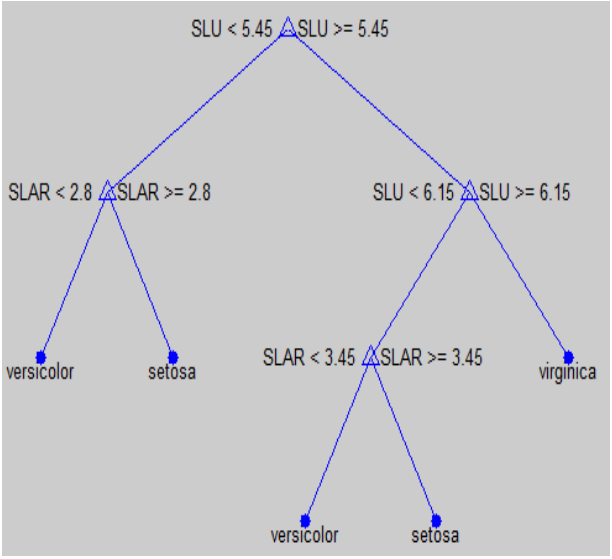


Figura 24. L'albero decisionale finale.

4.5. *k-Nearest Neighbor*

Per capire come funziona l'algoritmo *k-Nearest Neighbor* riprendiamo l'esempio delle iris ma questa volta consideriamo le informazioni sui petali e non sui setali. In Figura 25 vengono mostrati i dati in base alla loro specie e a lunghezza e larghezza dei petali. A questo punto dobbiamo classificare una nuova osservazione per capire a quale specie appartenga, visualizziamolo con il simbolo x all'interno della Figura 26. Cerchiamo ora i 10 punti più vicini a x utilizzando una delle metriche presentate. Nell'esempio viene utilizzata la distanza euclidea. La Figura 27 mostra uno zoom delle osservazioni selezionate vicine al punto che deve essere classificato. Andando a vedere a quale delle specie appartengono i 10 punti, questi sono: 2 virginica e 8 versicolor. Per questo motivo, la nuova iris sarà versicolor.

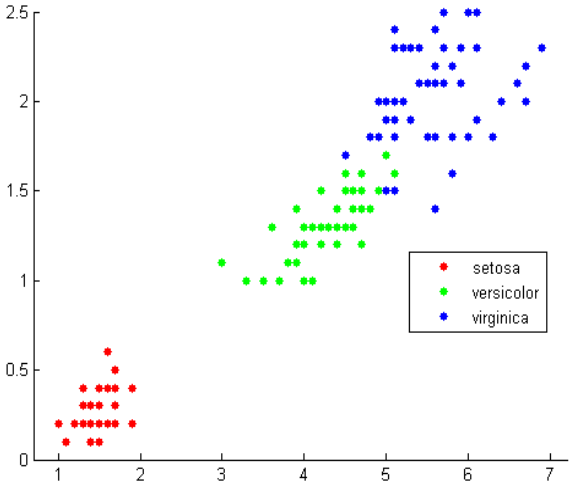


Figura 25. Rappresentazione del *dataset* sulle iris in base a larghezza e lunghezza dei petali.

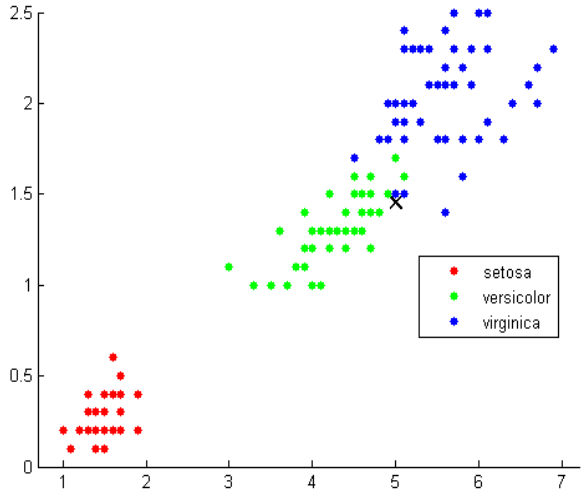


Figura 26. *Dataset* sulle iris alle prese con un nuovo punto

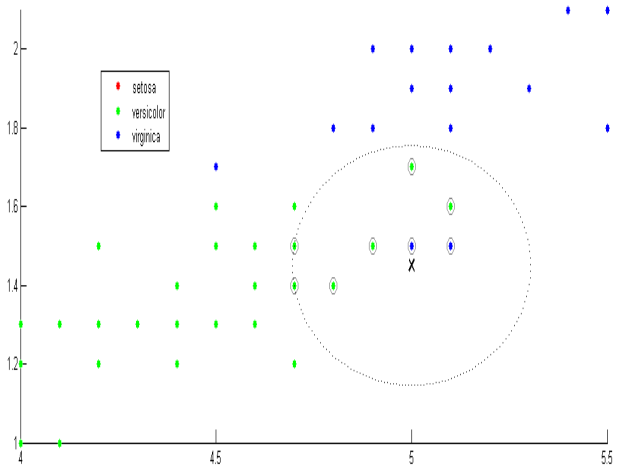


Figura 27. I 10 punti più vicini.

4.6. Analisi Discriminante (*Discriminant Analysis*)

L'analisi discriminante nasce alla fine degli anni '60 per opera di Edward Altman (1968) che per primo ideò un sistema discriminatorio per capire la probabilità di insolvenza di un'impresa.

Questa metodologia ha preso anche il nome di *Z-score* (Altman, 2000) in quanto Altman chiamò proprio *z* il punto che avrebbe permesso di stabilire la salute economico-finanziaria dell'impresa. Attorno alla *z* esiste comunque una zona chiamata *grigia*, di incertezza, e le imprese che vi si trovano sono difficilmente classificabili e quindi il giudizio assegnato soffre di incertezza.

Il modello multivariato (Altman, 1968) prevede la definizione di una funzione lineare che permette di separare due aree e quindi di selezionare i soggetti in base al valore della funzione ottenuto. Nel caso specifico immaginiamo di aver già ottenuto la funzione *z* che avrà la forma lineare

$$z = \beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n \tag{4.5}$$

e supponiamo di avere una nuova impresa da valutare. $x_1, \ldots, x_n$ rappresentano le variabili che servono per stimare la z per l'impresa analizzata, ad esempio: numero di dipendenti, capitale sociale, valore delle immobilizzazioni e così via. Poiché abbiamo già stimato il modello conosciamo anche i β e non manca che inserire i valori di bilancio all'interno della nostra funzione. Ebbene, ottenuta la nostra $z_{impresa}$ la confronteremo con il nostro z *score* e se maggiore, l'impresa sarà di un tipo (ad esempio, alto rischio di fallimento), se inferiore, l'azienda verrà valutata al contrario.

Riprendiamo l'esempio sulle iris e per costruire il modello usiamone un campione formato da 51 elementi con le informazioni sui sepali delle specie versicolor e virginica. La rappresentazione dei dati è presentata in Figura 28 in cui le ordinate rappresentano la larghezza delle foglie (SLARG) e le ascisse la lunghezza (SL). Creiamo ora 10.000 coppie di valori (x, y) che simulino le variabili SL e SLARG e, dopo aver definito la funzione *z-score* in base ai dati iniziali, le classifichiamo. La funzione discriminante è definita dall'equazione successiva:

$$z = -4{,}80 + 6{,}15x - 5{,}34y - 1{,}03x^2 + 1{,}67xy - 0{,}94y^2 \tag{4.6}$$

e l'analisi discriminante è illustrata in Figura 29.

Come si può notare la linea rosa delimita le due aree che definiscono i due tipi di iris. L'analisi discriminante può essere fatta utilizzando più variabili (multivariata) ma diventa impossibile darne una rappresentazione grafica.

Tuttavia, uno dei principali svantaggi dell'analisi discriminante è che facendo parte dei modelli econometrici parametrici è necessario assumere che la distribuzione dei dati di partenza sia normale. Per questo motivo prima di applicarla è necessario fare questo tipo di analisi ed eventuali successive trasformazioni sui dati del *database*.

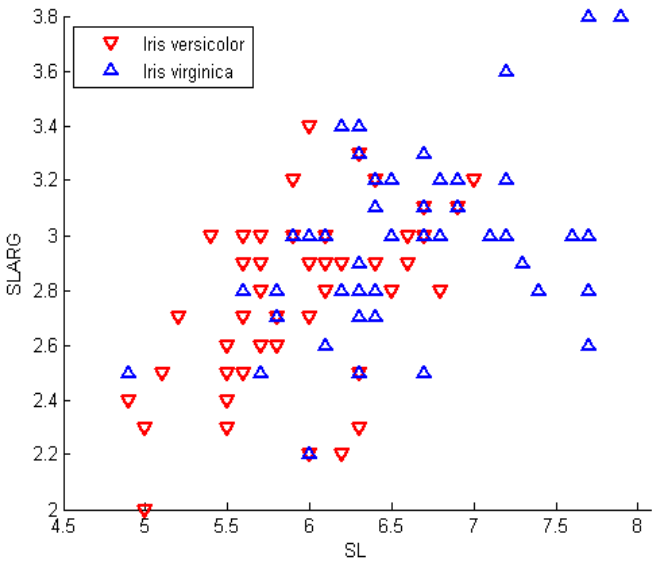


Figura 28. Rappresentazione dei sepali.

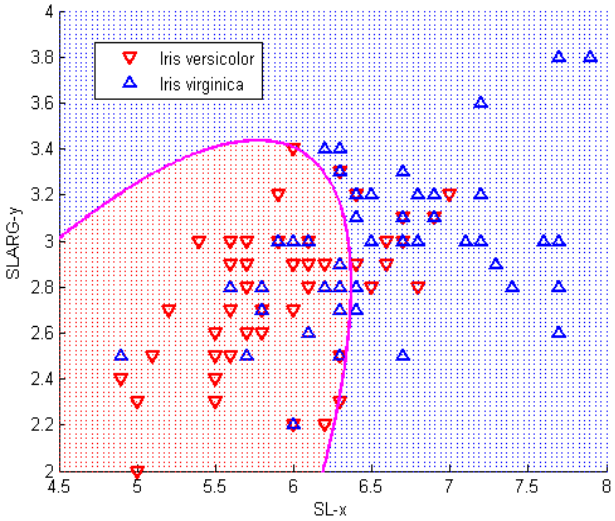


Figura 29. Z-score per le iris.

4.7. Analisi di regressione e serie temporali

Quando in un *database* vi sono dati che non considerano un preciso momento temporale (ad esempio, l'utile netto dell'impresa al 31/12/t) ma delle serie temporali (ad esempio l'utile netto dell'impresa al 31/12/t - al 31/12/t-1 - al 31/12/t-2 e al 31/12/t-3), vengono utilizzati dei modelli per l'analisi delle serie storiche generalmente orientati alla previsione.

Vi sono numerose tecniche utilizzate per l'analisi dei "*database* dinamici"[18] tra cui l'analisi della similarità nei dati e delle sotto-sequenze, il *Dynamic Time Warping (DTW)*, che viene utilizzato per il riconoscimento vocale e il filtro di Kalman che è un filtro ricorsivo per l'analisi di un sistema dinamico a partire da una serie di misure soggette a rumore ed infine l'analisi di regressione. Quest'ultima si occupa della ricerca di relazioni tra due variabili statistiche (qualitative o quantitative). Inoltre questo tipo di analisi può essere effettuata sia su serie storiche sia su dati *pooling* (cioè che non hanno una dinamica temporale) ma nel primo caso non solo l'analisi di regressione individua una relazione ma un effetto delle variabili chiamate indipendenti sulla variabile dipendente. Poiché la regressione è molto utilizzata, in questo paragrafo ci focalizzeremo su questa tecnica.

4.7.1. La regressione

La "funzione di regressione" esprime il legame di dipendenza di una variabile da una o più variabili ed è molto utile perché permette di valutare il valore della variabile dipendente al variare di quella/e indipendente7i. Ad esempio, se di un bene, non di prima necessità, si sono rilevate, al variare del prezzo, le relative quantità domandate, si può determinare, mediante il metodo "dei minimi quadrati" (*ordinary least squares*) o attraverso la stima della "massima verosimiglianza" (*maximum likelihood*), la funzione della domanda che esprime il legame fra il prezzo e la quantità domandata dai consumatori e quindi il produttore ha la possibilità di prevedere, per un prezzo prefissato, la corrispondente quantità di bene domandata.

[18] Dinamico significa che i dati variano con il passare del tempo.

La funzione più utilizzata è quella lineare e allora si parla di "regressione lineare" ma possono esservi anche relazioni diverse (logaritmiche, esponenziali, quadratiche, etc.).

Siano X e Y due variabili statistiche, in cui Y rappresenta la variabile dipendente e X la variabile indipendente; se esiste una relazione lineare, i punti si distribuiscono vicino a una retta; se invece i punti sono molto dispersi, non esiste alcuna relazione lineare.

Consideriamo un *dataset* formato da 488 individui[19] di cui conosciamo l'età (variabile "Età", di tipo discreto), lo stato civile (variabile "Non sposato"; dicotomica, può assumere due valori: 0 = Sposato e 1 = Non_Sposato), se vivono in città o campagna (variabile "Campagna"; dicotomica, può assumere due valori: 0 = Città e 1 = Campagna), se hanno terminato il corso di studi (variabile "Istruzione"; dicotomica, può assumere due valori: 0 = Non terminati e 1 = Terminati), quanto tempo hanno tenuto il lavoro (in termini di anno; variabile "Lavoro", di tipo continuo), salario annuale (espresso in migliaia di euro; variabile "Salario", di tipo continuo).[20]

Vogliamo vedere se esiste una relazione tra il tempo di lavoro e il salario: chi lavora da più tempo, guadagna di più? In termini di regressione vogliamo vedere se il salario dipende dal tempo di lavoro e quindi la variabile salario sarà la nostra dipendente (y), mentre il lavoro sarà l'indipendente (x).

In Figura 30 viene mostrata la relazione tra le due variabili ed è stata inserita anche la retta di regressione lineare stimata rappresentata dalla linea continua. La funzione ottenuta non ci dice molto ma nel calcolo della regressione vengono stimati anche i *p-value* il cui valore ci indica se la variabile considerata è significativa o no. La Tabella 8 riporta i parametri stimati nel modello di regressione lineare con il metodo dei minimi quadrati. Gli asterischi ($*$) indicano se la variabile è risultata significativa e cioè se la relazione indicata dalla funzione è statisticamente rilevante. Il numero degli asterischi ci dice a che livello la variabile è significativa. I risultati indicano che effettivamente esiste una relazione significativa al 95% (due $*$; $(1 - p{-}value) \cdot 100$) tra il lavoro e il salario. Il segno del coefficiente, in questo caso positivo, indica il segno della relazione. Che cosa ci dice quindi l'analisi? All'aumentare di un anno del tempo di lavoro (l'unità di misura di un anno è data da come sono stati raccolti i dati), il salario aumenta di 1,68 cioè di €1.680.

In fondo alla tabella abbiamo degli altri indicatori che ci dicono la qualità del modello. In particolare abbiamo la numerosità del campione e l'indice $Adjusted - R^2$ che, a parte essere un aggiustamento dell'R^2, indica quanta variabilità è spiegata nel modello considerando il numero delle osservazioni. Maggiore è questo indice e maggiore è la variabilità spiegata, quindi il risultato è migliore; la F di Fisher che è un *test* che viene fatto sul modello per stabilire se effettivamente i dati possono essere rappresentati da una funzione lineare. Accanto a questo valore si associa un *p-value* che, in questo caso, ci assicura che il modello ben rappresenta i dati.

[19] Il *dataset* qui utilizzato si chiama genxmpl3 ed è un *database* fornito da Stata per effettuare prove di codice o esempi e non analisi di dati o di ricerca. Pertanto i risultati ottenuti non riflettono la realtà ma sono da considerarsi un esempio del tipo di considerazioni che si possono ottenere effettuando una regressione. Il *dataset* è disponibile al seguente indirizzo: http://www.stata-press.com/data/r10/genxmpl3.dta

[20] Gli esercizi sono stati svolti con il *software* STATA/SE 12.0.

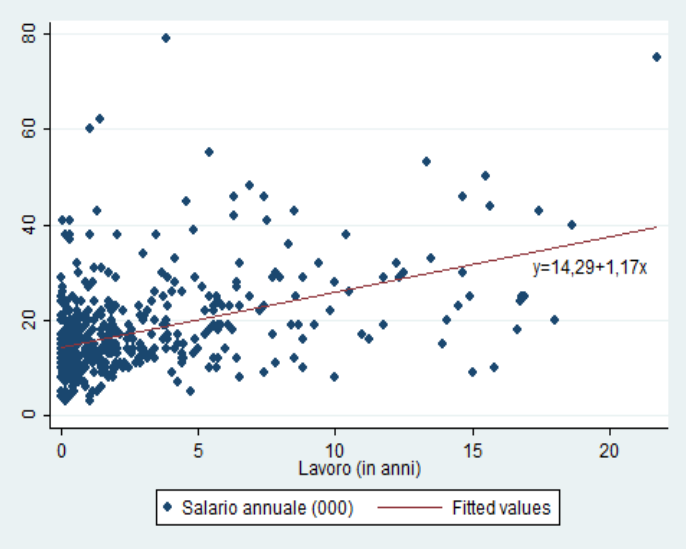


Figura 30. La funzione di regressione lineare.

Prima di applicare il modello di regressione su tutte le variabili, la Figura 31 mostra la relazione tra le variabili a coppie. In sostanza questo grafico calcola la regressione a coppie di variabili, cioè la correlazione tra gli attributi del campione.

L'interpretazione visiva di questi grafici è intuitiva per quanto riguarda le variabili continue o discrete, mentre per quelle dicotomiche, i valori sono tutti distribuiti sul valore 0 o 1 ed è quindi impossibile vedere una correlazione.

La variabile Età è correlata positivamente con Istruzione, Lavoro e Salario (e viceversa). Questo significa che all'aumentare dell'Età aumenta la probabilità che il soggetto abbia terminato gli studi (variabile Istruzione) che abbia lavorato per più tempo (Lavoro) e che guadagni di più (Salario).

La variabile Istruzione è correlata positivamente a Lavoro e Salario e anche queste due ultime variabili evidenziano un legame positivo.

Quanto ottenuto implica che esiste una relazione tra i due attributi ma non consente di fornire un nesso di causalità, cosa che invece sarebbe stato possibile valutare se avessimo avuto a disposizione dati temporali (serie storiche).

VARIABILI	**Salario (Dev. Std.)**
Lavoro	1,168***
	(0,107)
Intercetta	14,29***
	(0,498)
N	488
Adj-R^2	0,198
$F_{(1,486)}$	119,15

*** p<0,01, ** p<0,05, * p<0,1

Tabella 8. I risultati della regressione salario e lavoro.

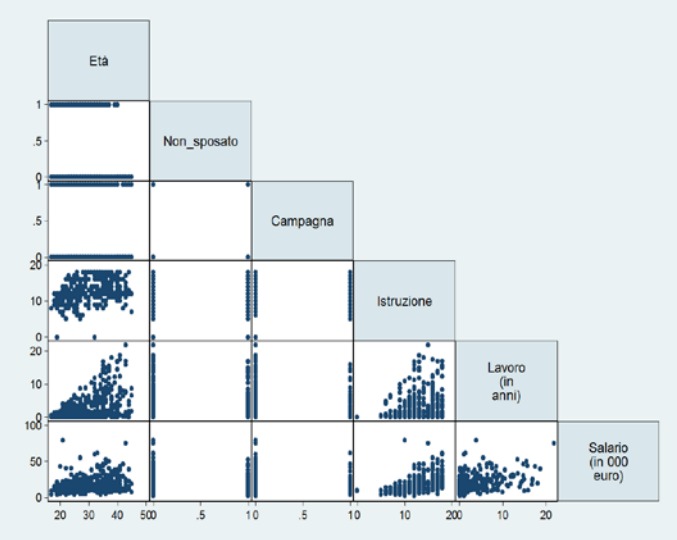


Figura 31. Le matrici di correlazione

La Tabella 9 mostra i risultati della regressione considerando tutte le variabili a disposizione. Come si può vedere, le variabili Età e Sposato non sono significative, cioè non vi è una relazione (lineare) con il salario.

La variabile Campagna è significativa al 95% e diversamente dalle altre ha un coefficiente con segno negativo che significa che passando da 0 a 1, cioè dalla città alla campagna, il salario diminuisce di 2,967 (che in termini di mila euro sono €2.967).

Attenzione: quando si analizzano i risultati della regressione, bisogna ricordarsi che vanno interpretati uno alla volta a parità delle altre variabili. Ad esempio, nel caso della variabile Campagna, possiamo dire che il salario diminuisce di 2.967 euro a parità delle altre variabili.

VARIABILI	**Salario (Dev. Std.)**
Lavoro	0,910***
	(0,111)
Età	-0,0329
	(0,0637)
Non_sposato	-0,463
	(0,862)
Campagna	-2,967***
	(0,843)
Istruzione	1,379***
	(0,143)
Intercetta	-0,573
	(2,333)
N	479
Adj-R^2	0,356
$F_{(5,471)}$	52,209

*** p<0,01, ** p<0,05, * p<0,1

Tabella 9. Il modello di regressione per tutte le variabili.

L'*Ad justed* $- R^2$ è migliorato rispetto a prima in quanto abbiamo più variabili e quindi la varianza totale è aumentata.

La *F* di Fisher conferma la bontà del modello. La numerosità campionaria è un pochino diminuita in quanto alcuni dati sono mancanti.

La funzione di regressione in questo caso avrà la seguente forma:

$$Salario = -0{,}57 + 0{,}91 \cdot Lavoro - 0{,}03 \cdot Età - 0{,}46 \cdot Non\_sposato - 2{,}97 \cdot Campagna + 1{,}38 \cdot Istruzione \quad (4.7)$$

Se invece avessimo voluto usare come variabile dipendente la variabile Non_sposato, che è dicotomica, avremmo dovuto usare un modello *logit* o *probit* che sono funzioni che in uscita hanno 0 o 1 e sono stimate con il metodo della massima verosimiglianza.

I risultati sono presentati in Tabella 10 in cui possiamo vedere che le uniche variabili significative sono Età e Lavoro (l'intercetta non viene considerata). Prima di commentare i risultati è necessario sottolineare che i coefficienti qui rappresentati non sono direttamente interpretabili in quanto devono essere oggetto di una trasformazione[21].

La trasformazione dei due coefficienti significativi è pari a -0,028 per l'Età e di 0,013 per il Lavoro. L'interpretazione è la seguente: all'aumentare di un anno di età la probabilità di incontrare nel *database* un soggetto non sposato diminuisce del 2,8%, mentre all'aumentare del tempo passato a lavoro, la probabilità di trovare un soggetto non sposato aumenta dell'1,3%.

VARIABILI	**Non_sposato (Dev. Std.)**
Età	-0,149***
	(0,0227)
Campagna	-0,405
	(0,260)
Istruzione	0,0556
	(0,0471)
Lavoro	0,0674*
	(0,0396)
Salario	-0,00602
	(0,0139)
Intercetta	2,568***
	(0,678)
N	479
Log-likelihood	-255,613
$\chi^2_{(5)}$	65,796

*** p<0,01, ** p<0,05, * p<0,1

Tabella 10. Modello Logit.

[21] Essi sono infatti i *log-odds* che sono dati da: $\frac{probabilità}{1-probabilità}$.
L'equazione del modello stimato è dunque:

$$L = 2{,}57 - 0{,}15 \cdot Età - 0{,}41 \cdot Campagna + 0{,}06 \cdot Istruzione + 0{,}07 \cdot Lavoro - 0{,}01 \cdot Salario$$

dove L è il log-odds. Per trasformare i coefficienti in modo che possano essere interpretati in termini percentuali bisogna procedere attraverso una trasformazione inversa dei log-odds: probabilità=$\frac{e^{logit}}{1+e^{logit}}$.

Quanto esposto vale anche per le serie storiche o dati *panel*. In questo caso è possibile non solo dire che esiste una relazione ma evidenziare un nesso di casualità.

Supponiamo che gli alunni di una classe vengano intervistati una volta al mese per 5 mesi. Noi avremo quindi lo stesso soggetto presente 5 volte nel campione, una per ogni intervista.

Supponiamo di avere delle informazioni sui soggetti: sesso, media dei voti, età, giudizio sulla qualità dell'insegnamento. Queste sono le nostre variabili.

Ipotizziamo di voler creare un modello per capire se la variabile sesso influenza il giudizio sulla qualità dell'insegnamento ed assumiamo inoltre di avere ottenuto un coefficienze statisticamente significativo al 95%.

Se non avessimo dei dati panel, l'interpretazione sarebbe che effettivamente esiste una relazione tra il sesso dell'allievo e il giudizio espresso sull'insegnamento. In questo caso invece, possiamo dire che essere maschio o femmina influenza il giudizio espresso.

Esistono inoltre diversi *test* che permettono di capire inoltre quali siano le variabili indipendenti che più influenzano la dipendente.

I modelli di regressione possono essere utilizzati per fare classificazione o previsione esattamente come accade per i sistemi di *Data Mining* utilizzati prima.

In una prima fase, si stima il modello e quindi si calcolano i coefficienti (quelli descritti nelle tabelle precedenti) e successivamente questi coefficienti vengono applicati a dati nuovi, non appartenenti al campione. Se abbiamo bisogno di sapere un risultato preciso, come un dato finanziario, utilizzeremo la regressione; se vogliamo classificare un soggetto in uno di due gruppi useremo un modello *logit* o *probit*; se invece dobbiamo classificarlo tra più gruppi si utilizzeranno due specificazioni dei modelli *logit* e *probit*, cioè il *multinomial logit*/*probit* o l'*ordered logit*/*probit.*

Per approfondimenti su questo argomento si veda Greene (2000).

4.8. Reti neurali artificiali (RNA) - *Artificial Neural Networks* (ANN)

I modelli basati sulle Reti Neurali Artificiali o *Artificial Neural Networks* (RNA o ANN) prendono spunto dalle ricerche in ambito biologico ed in particolare da quelle fondate sulla struttura del cervello. Alcuni ricercatori si sono focalizzati sul modo in cui l'uomo compie gesti e/o prende decisioni. A questo proposito si è studiata la forma, le caratteristiche e l'organizzazione dei neuroni in modo da costruire un nodo artificiale che, attraverso la programmazione informatica, fosse in grado di rappresentare l'attività del neurone biologico (Figura 32). Un neurone può essere considerato l'unità computazionale elementare del cervello.

La caratteristica principale del neurone è quella di generare un potenziale elettrico che si propaga lungo l'"assone"[22] quando l'attività elettrica, a livello del corpo del neurone, supera una determinata soglia. L'input in ingresso nel neurone, è un insieme di fibre chiamate "dendriti": esse sono in contatto con gli assoni di altri neuroni dai quali ricevono i potenziali elettrici. Il punto di connessione fra un assone di un neurone e il dendrite di un altro neurone è chiamato "sinapsi".

[22] L'assone corrisponde all'output del neurone.

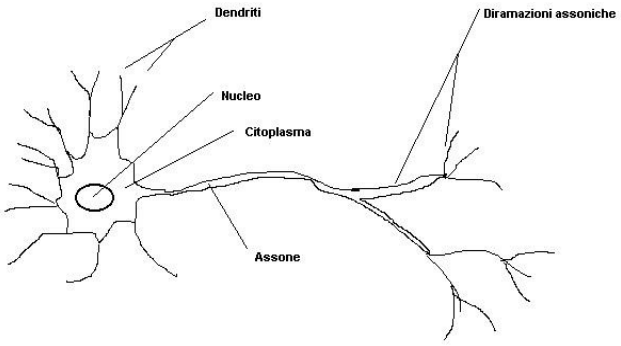


Figura 32. Rappresentazione di un neurone biologico.

Quest'ultima ha la proprietà di modulare l'impulso elettrico proveniente dall'assone. Il potenziale elettrico generato da un neurone è di tipo tutto-o-nulla (1 o 0). Infatti, se l'attività elettrica del neurone supera una certa soglia, si innesca l'impulso, altrimenti no e la scarica non differisce per intensità da un neurone all'altro. Il potenziale si propaga lungo l'assone e giunge alla sinapsi con il dendrite di un altro neurone. Il potenziale post-sinaptico sul dendrite dipende dalle caratteristiche biochimiche della sinapsi. In presenza dello stesso potenziale pre-sinaptico, due sinapsi diverse generano potenziali post-sinaptici differenti. In altre parole, la sinapsi pesa il potenziale in ingresso (input) modulandolo. I potenziali post-sinaptici si propagano attraverso i dendriti del neurone e, a livello del soma[23], si sommano. Solo se il risultato di tale somma è superiore ad una certa soglia, il neurone innesca il potenziale che si propagherà attraverso il suo assone, altrimenti ciò non accade.

Per le caratteristiche peculiari del funzionamento del neurone, lo stesso può essere rappresentato come nella Figura 33. Le linee di ingresso, corrispondenti ognuna ad un dendrite, terminano su di un modulo somma (Σ), corrispondente al soma del neurone. In questo modulo, i segnali vengono sommati (stimolatori) o sottratti (inibitori) per dare un segnale che perverrà al modulo A. Il modulo A è un elemento decisionale che trasmette sulla sua linea di uscita (assone) un segnale con certe caratteristiche e che uscirà solo se l'uscita è maggiore di un certo valore corrispondente alla soglia del neurone stesso.

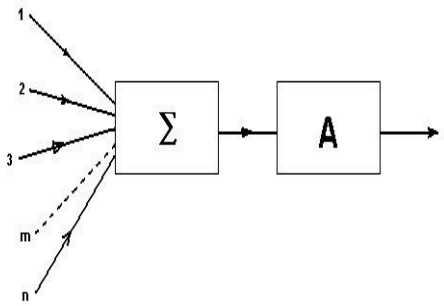


Figura 33. Rappresentazione del funzionamento di un neurone biologico.

[23] Corpo di varia forma della cellula nervosa che ne contiene il nucleo.

Data questa rappresentazione, sono state trasformate tutte queste caratteristiche in un modello informatico capace di descrivere il funzionamento dei neuroni. Nascono così, i nodi artificiali.

La connessione di più neuroni artificiali ha fatto nascere la cosiddetta Rete Neurale Artificiale. Questa semplice rappresentazione ha permesso di poter pensare a neuroni artificiali in grado di risolvere problemi complessi in diversi ambiti applicativi.

4.8.1. La definizione informatica delle RNA

Le reti neurali artificiali sono una classe di modelli composti da strati di unità elementari di elaborazione, *Processing Elements* (PEs) che elaborano l'informazione per mezzo di una funzione non lineare. Tali unità sono anche dette neuroni o nodi per analogia con il sistema connessionista biologico.

Questi modelli fanno parte dei sistemi di intelligenza artificiale ma, a differenza di quelli classici, nelle reti neurali non esiste un decisore centrale. La decisione risulta dunque essere il risultato di un lavoro cooperativo distribuito tra tante molecole. Le differenze fondamentali rappresentano le caratteristiche principali della rete e sono:

- Le reti neurali sono potenzialmente adattive in quanto apprendono;
- Le reti neurali sono in grado di generalizzare;
- Le reti neurali sono resistenti al rumore[24];
- Le reti neurali sono resistenti alle lesioni[25];
- Le reti neurali possono essere rappresentate simbolicamente.

È possibile individuare le componenti di un nodo e verificare quali problemi sono emersi nella costruzione e/o implementazione dello stesso, nel contesto della ricerca empirica presente nella letteratura.

Gli elementi fondamentali del neurone sono i seguenti e sono rappresentati nella Figura 34:

- Inputs o strati di attivazione (x_i);
- Connessioni o pesi o sinapsi (w_i);
- Stato di attivazione netto (Net);
- Valore soglia o *bias* (θ)[26];
- Funzione di attivazione ($f(Net)$);
- Output(s) o stato di attivazione ($y(_i)$).

Per comprendere le potenzialità di questo strumento, è necessario spiegare la fase più importante della costruzione di una rete neurale per la classificazione e/o la previsione dell'insolvenza.

[24] Nel mondo reale, non si ricevono input puliti, ad esempio, quando si parla al cellulare in una strada molto affollata, si sente pochissimo di quello che la persona dall'altra parte sta dicendo. Tuttavia, è sufficiente per capire quello che l'altro sta dicendo. Un organismo, dunque, o un modello artificiale che lo vuole riprodurre e spiegare, deve riuscire a percepire un segnale attraverso un rumore enorme. Nell'intelligenza artificiale classica, questo non è possibile mentre nelle reti neurali ciò è riproducibile.

[25] Se viene danneggiata una parte della rete, il danno che viene misurato può essere piccolissimo e quindi la stessa rete può continuare a funzionare quasi come prima.

[26] Nella rete a percettrone si utilizza un determinato valore soglia che definisce lo stato di attivazione o inibizione di ogni singolo nodo. Questo *bias*, nelle versioni più recenti di rete, è rappresentato dal peso applicato all'input e che assume un valore costante pari a 1. Questa variabile, sempre presente, arricchisce i gradi di libertà della funzione.

Tale fase è quella dell'"apprendimento" o "training" che avviene attraverso un algoritmo ben specificato da chi definisce la struttura della rete. A questo punto, operano le "leggi di apprendimento" che fissano regole per attuare le variazioni alle connessioni della rete.

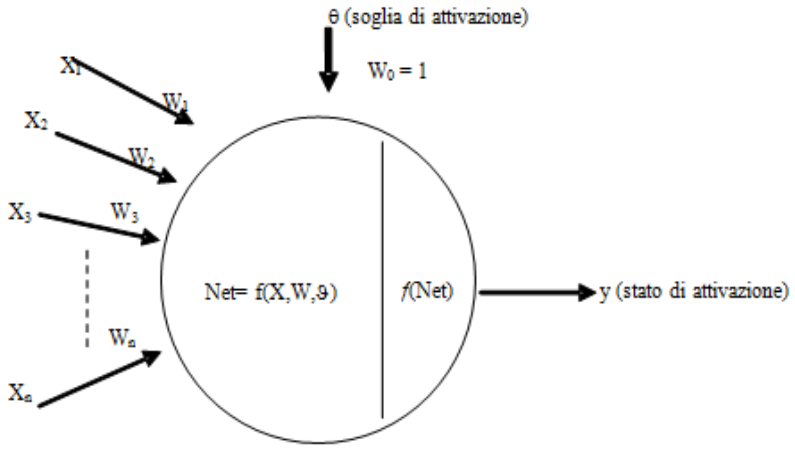


Figura 34. Rappresentazione grafica di un nodo o neurone.

Queste regole possono essere di tipo:

- "Non supervisionato": la rete utilizza degli stimoli *S* provenienti dal mondo reale che fornisce delle risposte *R* sulla base degli stessi stimoli. In questo caso la rete si auto-organizza in modo da favorire una adeguata rappresentazione della realtà;
- "Supervisionato": la rete interagisce indirettamente con il mondo reale attraverso l'utilizzo sia degli stimoli *S* che delle risposte *R* derivate dall'ambiente e fornite da un intermediario esterno. In questo caso è necessario seguire la seguente procedura operativa:
  - Stabilire quale valore ogni unità di output deve assumere quando la rete classifica correttamente un'osservazione sottoposta sulle unità di input;
  - Presentare sulle unità di input degli esempi di apprendimento (*training set*) di cui si conosce l'appartenenza o meno ad uno dei gruppi oggetto di studio;
  - In seguito si valuta la risposta della rete. Se questa è:
    - Corretta, allora la fase di apprendimento prosegue prendendo in esame un'altra osservazione;
    - Diversamente, si provvede a calcolare l'errore tra la risposta ottenuta dalla rete e quella desiderata e modificando i pesi si cerca di ottenere dalla rete la risposta corretta.

La tecnica matematica più utilizzata recentemente per il *training* nell'apprendimento supervisionato è quella dell'algoritmo "back-propagation" che fa apprendere la rete sulla base della minimizzazione dell'errore nella definizione del risultato.

La Figura 35 riporta uno schema che rappresenta il funzionamento di questo tipo di apprendimento che è simile a quello presentato nel paragrafo 3 riguardo al funzionamento dei sistemi di *Data* Mining (Figura 2). Nel caso dell'apprendimento non supervisionato, è la rete stessa che è lasciata libera di auto-organizzarsi sulla base delle somiglianze esistenti tra i vari esempi, modificando autonomamente le proprie connessioni sulla base di precise

indicazioni che definiscono come tenere conto delle varie distanze intercorrenti tra i differenti esempi.

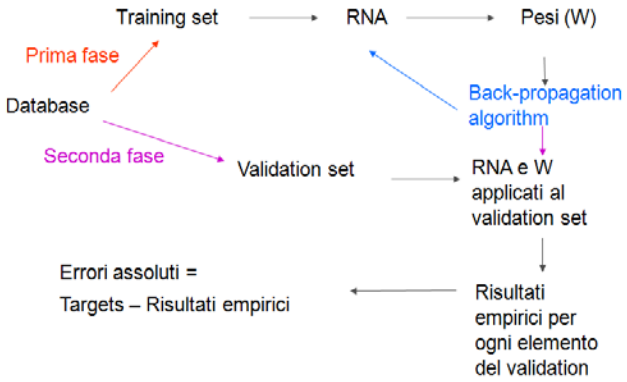


Figura 35. Apprendimento supervisionato.

In entrambe i casi, l'apprendimento avviene mediante "cicli" o "epoche" in quanto gli esempi appartenenti al *training set* vengono presentati più volte.

In conclusione, la definizione di RNA che meglio rappresenta i modelli che vengono utilizzati per fare classificazione/previsione/simulazione è la seguente ed è rappresentata in Figura 36:

> "una RNA è un modello complesso formato da neuroni (anche chiamati percettroni) che sono collegati tra di loro da sinapsi (pesi) e che consentono di simulare o prevedere un evento o il comportamento di un agente".

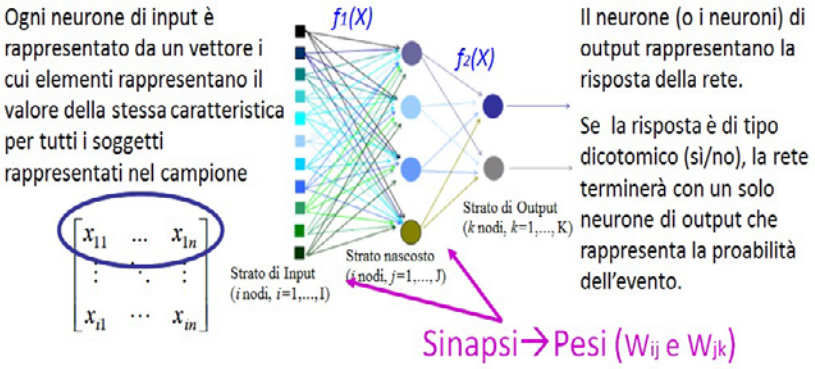


Figura 36. RNA-ANN.

In Figura 36 si vede che i neuroni di una rete sono organizzati in più strati:

- Lo strato di input in cui i neuroni rappresentano le variabili del *database*;
- Lo strato nascosto che è dato da un numero di neuroni che va trovato empiricamente (si noti che anche il numero di strati nascosti deriva da prove empiriche);
- Lo strato di output che genera l'uscita.

Il passaggio da uno strato all'altro avviene tramite delle funzioni che vengono chiamate "di attivazione" o "di trasferimento" (*Transfer functions*). Queste funzioni sono generalmente non lineari[27] e poiché sono quelle attraverso le quali si determinano i pesi è fondamentale trovare quelle che meglio rappresentano le relazioni tra i nostri dati. Se per esempio ci aspettiamo che la rete ci dica se un soggetto è di tipo 0 o 1 allora in uscita avremo una funzione che varia tra 0 e 1, quindi una logaritmica.

Non essendo una tecnica parametrica, non esistono *test* statistici da applicare e quindi la valutazione è solo ex-post in base ai risultati empirici ottenuti.

Come è già stato sottolineato precedentemente, le tecniche che si basano sull'apprendimento, se da un lato riescono a simulare efficientemente ed efficacemente la realtà, dall'altro possono soffrire di problemi di *overfitting* oppure di incapacità di valutare correttamente un soggetto molto diverso dal campione di *training*. Bisogna dunque fare molta attenzione alla definizione del *database* iniziale in quanto da esso deriva la bontà della stima del modello.

Esistono molte strutture di RNA in base a come sono fatti gli strati, alle funzioni utilizzate al tipo di apprendimento e anche al tipo di dati da analizzare. Qui di seguito vengono presentati brevemente i tipi più comuni.

Nei prossimi paragrafi verranno illustrate più dettagliatamente il tipo di rete *Feed-forward neural network* e *Self-Organizing Feature Map*, che rispettivamente si caratterizzano la prima per l'apprendimento supervisionato (algoritmo *back-propagation*) e la seconda per il non-supervisionato, in quanto sono le più utilizzate.

Infine, si farà un breve accenno alle *Support Vector Machine* che rappresentano un metodo di apprendimento supervisionato per la regressione e la classificazione di *pattern*. Queste metodologie sono state sviluppate da Vapnik (1998) e sono molto utilizzate nel campo *text mining*.

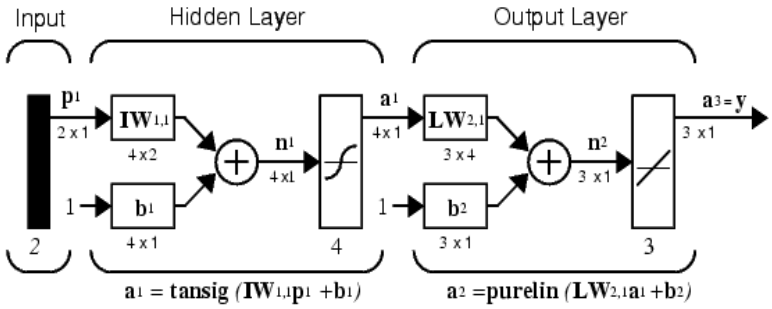
Per le altre topologie si rimanda all'appendice (p. 86).

4.8.2. *Feed-Forward Neural Network* (FFNN)

La topologia di questa rete è molto flessibile in quanto è quella più simile al reticolo neurale del cervello. Si tratta infatti di un insieme di neuroni collegati tra loro da legami (funzioni di attivazione - assoni) che trasmettono le informazioni (sinapsi - pesi).

La caratteristica principale di questa topologia di rete è l'essere *feed-forward* (si veda Figura 37) e cioè presentare legami che vanno solo in una direzione: dallo strato di input allo strato nascosto (che ricordiamo, può essere più di uno) e da questo allo strato di output. Le informazioni viaggiano solo in questa direzione e non è possibile che dallo strato nascosto parta dell'informazione verso quello di input.

[27] Si veda l'appendice a p. 79 in cui vi è la descrizione delle principali funzioni di trasferimento.



Fonte: Beale et al. (2010).
Figura 37. FFNN.

In questa sezione si tralascia la matematica[28] che è piuttosto impegnativa ma si mostra un esempio molto semplice ma che mostra come le reti neurali *feed-forward* siano degli strumenti efficaci.

Supponiamo di avere un campione di 10 soggetti di cui conosciamo alcune variabili. Di questi elementi conosciamo anche l'output. La Figura 38 mostra il confronto tra gli output reali (+ in rosso) e quelli empirici in uscita dalla rete (* blu). Come si può notare vi è una grande differenza tra i valori, questo perché la rete non è stata ancora addestrata. Se addestriamo la rete per 150 epoche attraverso l'algoritmo della *back-propagation* è possibile vedere che la precisione nei risultati migliora (Figura 39). Si noti che ora sono poche le osservazioni che non riescono ad essere simulate dalla rete. Per migliorare le performance possiamo aumentare il numero di epoche, cambiare le funzioni di attivazione (in questo esempio abbiamo usato delle *logsig*, si veda l'appendice a p. 79), aumentare il numero degli strati nascosti, modificare il numero di neuroni negli strati nascosti, cambiare le funzioni per inizializzare le matrici dei pesi, modificare altre funzioni per le definizioni dei massimi… Insomma, ci sono molti modi in cui possiamo modificare la topologia della RNA per far migliorare il risultato.

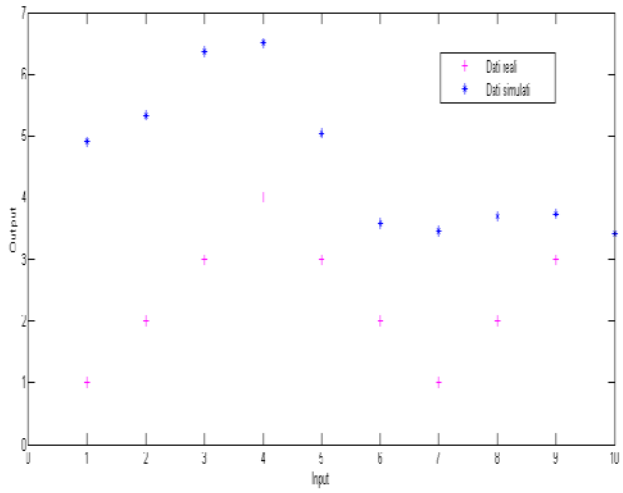


Figura 38. Esempio FFNN - pre train.

[28] Per quanto concerna la trattazione matematica delle reti neurali artificiali si consiglia Haykin (2009) e Haykin (1994).

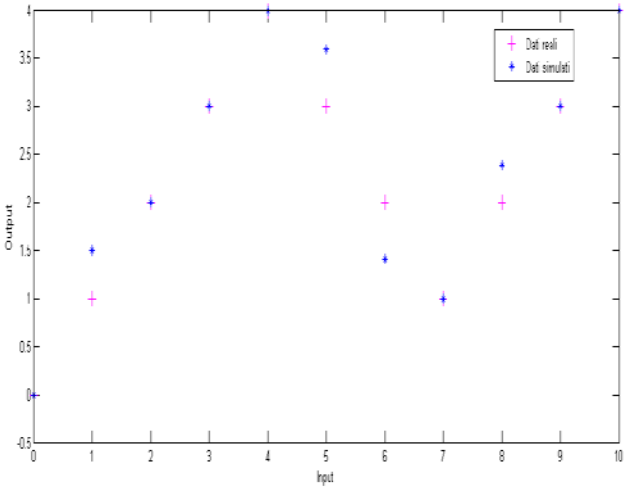


Figura 39. Esempio FFNN - post train.

4.8.3. *Self-Organizing Feature Map* (SOFM)

Le SOFM sono RNA che consistono di due strati di neuroni e hanno un tipo di apprendimento non supervisionato, come illustrato in Figura 40.

Il primo strato non è uno strato di neuroni veri e propri ma serve come ricettore e trasmettitore dei dati di input verso il secondo strato.

Consideriamo il caso più semplice, cioè quello in cui il secondo strato è costituito da due dimensioni: ogni neurone del secondo strato si connette con ogni neurone del primo (il numero di neuroni del secondo strato è scelto arbitrariamente).

Ogni neurone del secondo strato ha un suo vettore di pesi uguale alla dimensione dello strato di input. I neuroni sono connessi tra di loro in base a (relazioni di vicinanza[29]), che determinano la struttura della mappa. Queste relazioni sono definite attraverso delle apposite funzioni (*topological neighborhood functions*). L'obiettivo dell'apprendimento nelle *self-organizing map* è di specializzare parti differenti del secondo strato a rispondere similmente a particolari *pattern* d'ingresso, così come le informazioni sensoriali visive, uditive o di altro tipo sono gestite da parti separate della corteccia cerebrale nel cervello umano.

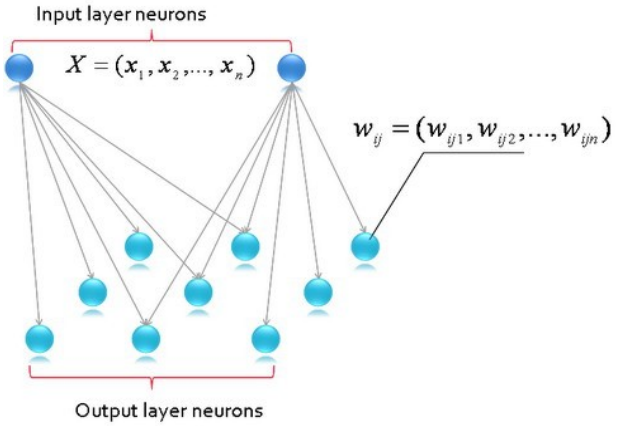


Figura 40. *Self-Organizing (Feature) Map*: apprendimento non supervisionato.

[29] Per le metriche si veda l'appendice a p. 75.

L'addestramento (*training*) utilizza l'apprendimento competitivo (*compet function*). Quando viene passato un campione di *training* in ingresso alla rete, si calcola la sua distanza euclidea da tutti i vettori dei pesi. Il neurone col vettore dei pesi più simile all'ingresso è chiamato *Best Matching Unit (BMU)*. I pesi della BMU e dei neuroni vicini a questo nel reticolo SOFM vengono avvicinati al vettore d'ingresso. L'intensità dell'avvicinamento decresce nel tempo e in funzione della distanza dei neuroni dalla BMU.

Questo processo viene ripetuto per ogni vettore d'ingresso per un numero di cicli variabile. Le SOFM hanno due modalità di funzionamento:

1. ***Batch Training***. Durante il processo di addestramento si costruisce una mappa e la rete neurale si organizza usando un processo competitivo. È necessario dare in ingresso alla rete un numero elevato di vettori d'ingresso, che rappresentino il più possibile il tipo di vettori che ci si aspetta durante la seconda fase (se ce ne sarà una). Altrimenti, gli stessi vettori d'ingresso devono essere "somministrati" più volte.
2. ***Incremental Training***. Durante il processo di *mapping* un nuovo vettore può essere dato in ingresso alla mappa; questo viene automaticamente classificato o categorizzato. Ci sarà un solo neurone vincitore: quello il cui vettore dei pesi giace più vicino al vettore d'ingresso che viene facilmente individuato calcolando la distanza euclidea fra il vettore d'ingresso e il vettore dei pesi.

Si consideri l'esempio in cui vi sono come input 1000 osservazioni di cui si conoscono 2 variabili. Per ognuna delle 1000 osservazioni abbiamo 4 output. Lo strato di output è formato da due dimensioni (due variabili) e ogni dimensione è costituita da 6 neuroni (lo spazio sarà quindi 6*x*6). La precedente informazione ci dice anche che essendo uno spazio bidimensionale, il numero di *clusters* che vogliamo ottenere alla fine è pari a 36. Le iterazioni (o "epoche") vengono impostate a 200.

La Figura 41 rappresenta la posizione dei dati di input (in verde), dei neuroni (pallini blu-grigio) e dei vettori dei pesi (linee rosse), mentre la Figura 42 attraverso il colore (più chiaro è il colore e minore è la distanza), visualizza la distanza tra i neuroni (è stata utilizzata la metrica euclidea).

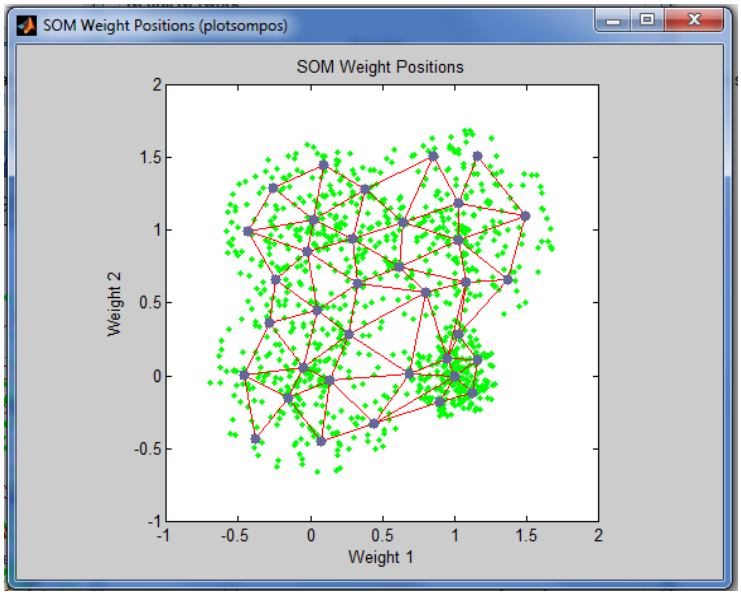


Figura 41. Posizione dei dati di input e dei vettori dei pesi.

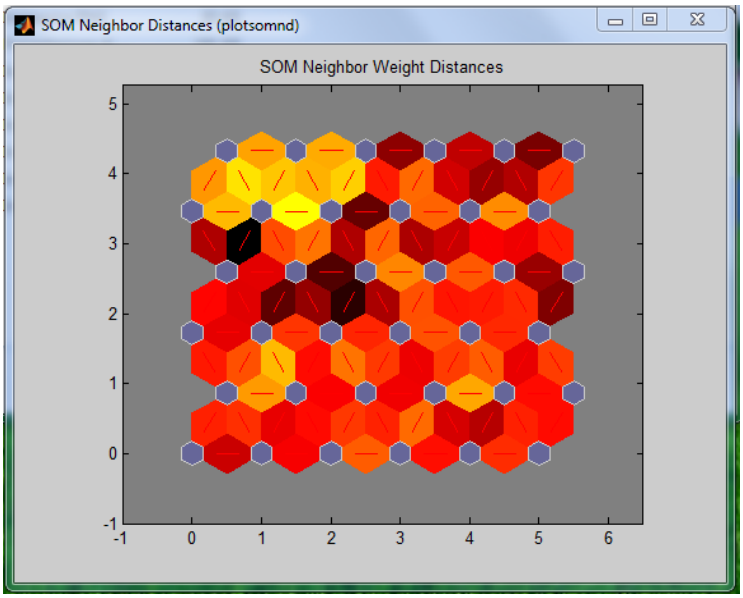


Figura 42. Visualizzazione della distanza tra i neuroni.

La Figura 43 mostra i neuroni e la loro dimensione mostra quanti input sono raccolti in ogni neurone.

Infine, in Figura 44 è raffigurata per ogni variabile (e quindi per ogni input) la mappa dei pesi (distanze) tra i neuroni. Come prima, i colori più chiari indicano neuroni più vicini. Come si può vedere c'è molta differenza tra le due variabili.

Se la suddivisione in 36 (6x6) *cluster* non risulta la più efficace, possono essere modificate le funzioni di distanza, le funzioni per la definizione dei pesi iniziali, la struttura della griglia di partenza, nonché la sua dimensione. Oltre a ciò, come nel caso delle altre topologie, vi sono dei parametri tecnico-matematico che sono soggetti a cambiamento e possono influenzare i risultati.

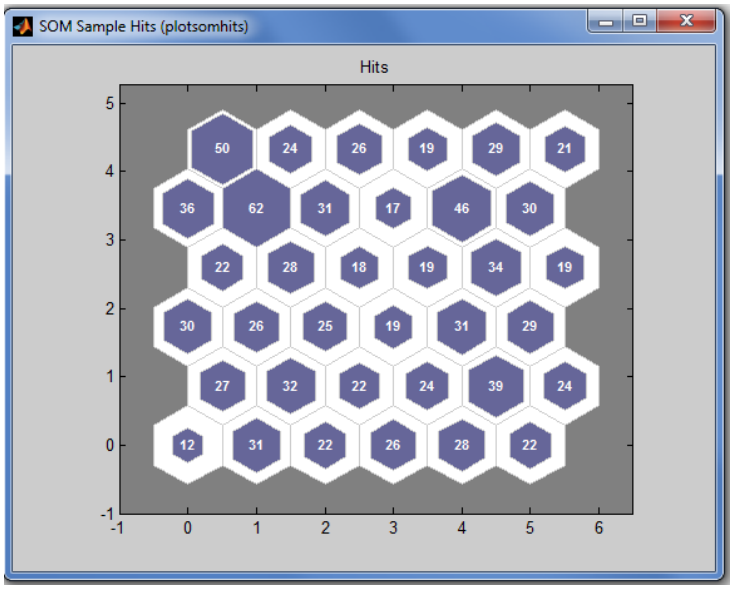


Figura 43. Elementi raggruppati in base ai neuroni.

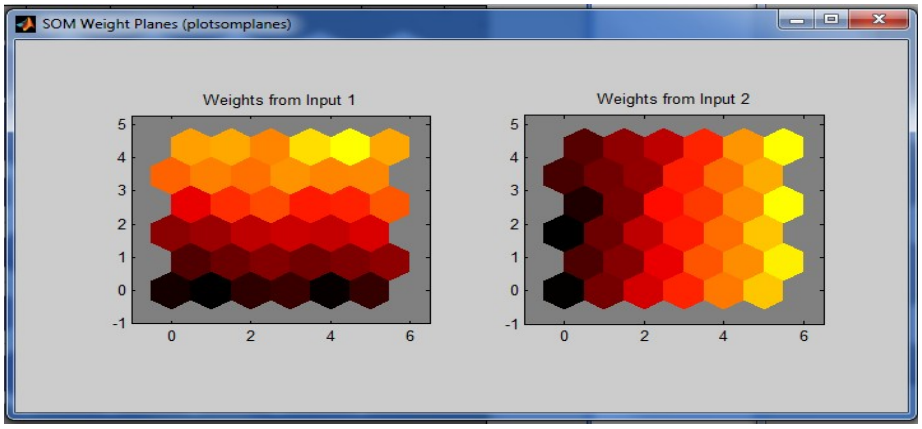


Figura 44. Visualizzazione dei pesi per ogni input.

È necessario inoltre ricordare che le SOF M vengono spesso chiamate "mappe di Kohonen", dal nome di colui che per primo ne formulò il funzionamento e la topologia (Kohonen, 1982; Kohonen, 2001).

4.8.4. *Support Vector Machine* (SVM)

Questa metodologia è una delle più utilizzate nel campo del *Text Mining* in quanto rappresenta una tecnica che possiamo definire "ibrida". Essa infatti associa un tipo di apprendimento "neurale" (supervisionato) e un modello di regressione (che può essere lineare, quadratica, gaussiana, polinomiale o definita dall'utente).

Le SVM classificano gli elementi del *dataset* minimizzando l'errore empirico ma anche massimizzando il margine geometrico intorno alla *kernel function*[30] che discrimina tra gli elementi.

Per spiegare come funzionano le SVM riconsideriamo il campione di iris (Figura 21, pag. 38). Il *dataset* è formato da 150 rilevazioni di cui si conoscono larghezza e lunghezza di petali e sepali. In questo esempio consideriamo i sepali (SL = lunghezza e SLARG = larghezza). Per quanto concerne l'output, per ognuna delle 150 iris sappiamo se appartiene alla specie "setosa", "versicolor" o "virginica". Per semplicità, trasformiamo la variabile di output in un attributo dicotomico 0 = non-setosa e 1 = setosa; procediamo suddividendo in modo casuale il *set* di *training* e di *validation* nella proporzione di 2/3 il primo e 1/3 il secondo. In Figura 45 sono mostrate le performance sul *set* di *training*. Le due tipologie (setosa e non-setosa) sono separate nettamente da una retta lineare. Infatti, è stata utilizzata una *kernel function* lineare per definire le due regioni. I cerchi vuoti rotondi definiscono i *support vector* che delimitano la massima distanza intorno alla linea che permette di avere una corretta classificazione. Come si può notare, se volessimo unire con una linea i *support vector*, questa non sarebbe affatto lineare, anche se la retta da cui si parte ha una forma lineare. Proprio il fatto che da un lato si parte da un modello noto (regressione lineare per la minimizzazione dell'errore) e dall'altro si utilizzano i dati empirici per disegnare dei confini (*support vector*, massimizzazione dei margini), rende le *support vector machine* delle tecniche decisamente efficaci ed efficienti nella classificazione.

[30] Semplificando, le funzioni *kernel* calcolano una stima della densità di probabilità di un campione. Per un approfondimento si veda Bergman (1970).

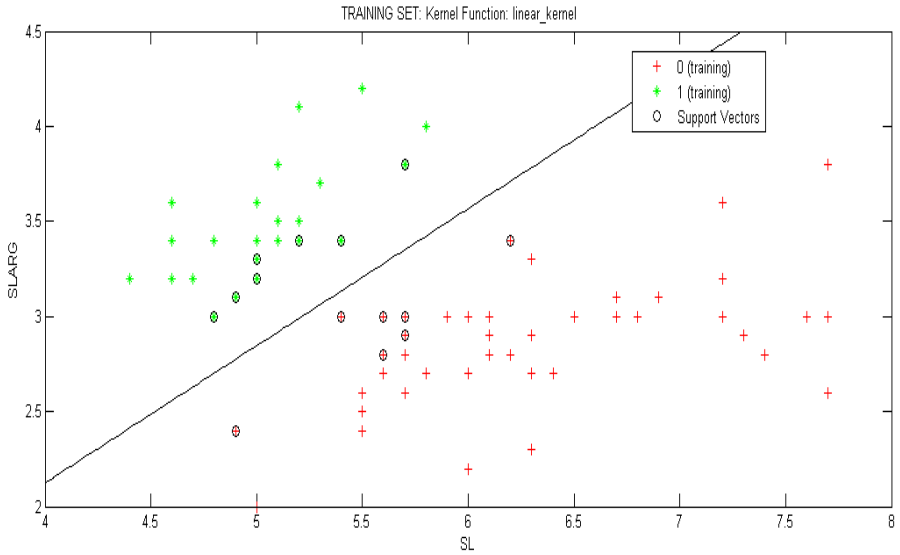


Figura 45. Support Vector Machine (*training*).

In Figura 46 mostriamo i risultati del modello applicato ai rimanenti dati sulle iris e verifichiamo le performance calcolando la percentuale di corretta classificazione. Questa risulta essere pari a 0,9867 cioè il 98,67% delle iris sottoposte alla rete sono stati correttamente classificati in base alla specie.

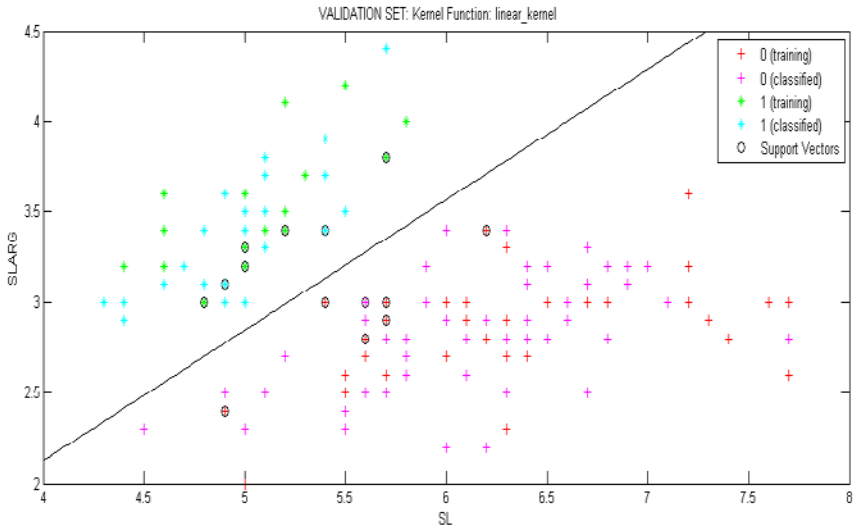


Figura 46. Support Vector Machine (*validation*).

4.8.5. I principali campi di applicazione delle RNA

I più comuni campi di applicazione sono:

- Cancellazione adattiva del rumore nelle telecomunicazioni
- Valutazione dei rischi di ammortamento
- Detector di bombe al plastico
- Controllo di qualità nella produzione dei tubi fluorescenti
- Riconoscimento di parole

- Controllo preventivo della rumorosità dei motori di automobile
- Comprensione e classificazione dei risultati delle ricerche scientifiche
- Tattiche di marketing per le compagnie aeree
- Classificazione dei segnali sonar
- Previsione della vulnerabilità da eroina
- Riconoscimento e inseguimento di obbiettivi mobili
- Compressione di dati e immagini
- Riconoscimento di caratteri manuali
- Validazione firme
- Controllo qualità nei processi di automazione industriale
- Valutazione della probabilità in diversi contesti
- Identificazione dei candidati migliori per specifiche posizioni in una azienda
- Minimizzazione dei *database* delle aziende
- Ottimizzazione del rapporto qualità dei posti/prezzo del biglietto per le linee aeree
- Valutazione dei premi assicurativi
- Selezione di squadre specializzate

Nella finanza i principali campi di applicazione sono:

- Analisi del rischio di credito e valutazione delle perdite
- Lettura e validazione delle forme scritte a mano
- Distribuzione degli investimenti e analisi del portafoglio cliente
- Previsione e simulazione degli scenari di cambio valutari e simulazione di scenari economici
- Previsione dei titoli e delle azioni
- Analisi della evasione e della elusione fiscale

Per la risoluzione di problemi legati all'ambiente, li troviamo applicati a:

- Simulazione dei modelli di sviluppo urbano e Analisi dei movimenti migratori
- Previsione della mobilità lavorativa
- Previsioni del tempo e Previsione dei tassi di inquinamento

Nel settore industriale della manifattura:

- Automazione robots e dei sistemi di controllo multiparametrici
- Controllo dei processi di produzione in linea
- Ispezione qualità
- Selezione parti per le linee di assemblaggio

Nell'ambito giuridico sono impiegate nella:

- Valutazione prove
- Valutazione delle sospensioni di concessione edilizia
- Ottimizzazione delle strategie di istruzione dei processi

Nella biologia:

- Funzionamento danneggiato del cervello
- Modellizzazione della retina e della coclea
- Studio del DNA
- Simulazione nuove molecole

Nel campo medico e socio-sanitario:

- Analisi del parlato nell'aiuto dell'ascolto per i sordi profondi
- Diagnosi e prescrizione di trattamento a partire da sintomi
- Monitoraggio delle operazioni chirurgiche
- Previsione di effetti collaterali di alcuni farmaci su specifici soggetti
- Lettura delle radiografie
- Comprensione delle cause dei fenomeni epilettici
- Previsione della vulnerabilità schizofrenica
- Previsione abbandono scolastico

4.9. Algoritmi genetici

Prima di passare alla valutazione dei sistemi appena illustrati, viene qui brevemente descritto il funzionamento degli algoritmi genetici. Generalmente vengono utilizzati per risolvere problemi di ottimizzazione soprattutto quando si è di fronte a problemi del tipo "se... allora".

Gli Algoritmi Genetici, proposti nel 1975 da J.H. Holland, sono un modello computazionale che si ispira ai modelli dell'evoluzione naturale darwinista. Ogni individuo ha sue caratteristiche e proprietà specifiche che tutti possiamo vedere e che sono quindi "visibili". Queste qualità costituiscono il "fenotipo dell'individuo". Quest'ultimo determina le possibilità e i limiti delle interazioni dell'individuo con l'ambiente in cui vive. Tuttavia, il fenotipo è determinato sostanzialmente dal patrimonio genetico "invisibile" o "genotipo", costituito dai geni. Ad ognuno di questi corrisponde uno specifico fenotipo e quindi la sopravvivenza degli individui con caratteristiche più adatte, significa in realtà la sopravvivenza dei geni più adatti. I due princìpi fondamentali dell'evoluzione sono la "variazione genetica" e la "selezione naturale". A questo punto, è chiaro che, affinché la popolazione possa evolvere, gli individui che la costituiscono devono essere caratterizzati da una ricca varietà di fenotipi e quindi di genotipi.

Da qui in poi opera la selezione, che premia la sopravvivenza, la longevità e la riproduzione degli individui più adatti. I meccanismi generatori della varietà del genotipo sono sostanzialmente due: un "processo combinatorio" dei geni, grazie ai diversi apporti dei genitori e le "mutazioni geniche" casuali. Le mutazioni producono nuovi geni, alcuni dei quali si tramandano alle generazioni successive, mentre altri scompaiono e il cosiddetto pool di geni, nel quale "pesca" la selezione naturale, cambia continuamente. I cambiamenti che avvengono da una generazione all'altra sono molto piccoli ma quelli positivi si accumulano (selezione cumulativa) e, in tempi lunghissimi, danno origine a cambiamenti enormi.

Secondo la moderna versione degli "equilibri punteggiati", l'evoluzione sarebbe fortemente influenzata da eventi eccezionali e soprattutto avverrebbe per salti. Ciò significa che

a periodi di ristagno, che possono essere anche lunghissimi, seguono periodi di accelerazione evolutiva relativamente brevi.

Sono stati dunque introdotti gli operatori fondamentali dell'algoritmo genetico che qui riassumiamo:

- **Selezione genetica e riproduzione**: identifica quali elementi di una popolazione sopravvivono per riprodursi e con la riproduzione ha luogo la ricombinazione dei geni. Si basa sul processo selettivo per cui il principio dell'adeguatezza impone che "sopravviva il più forte". Nei modelli economici, questa fase avviene attraverso una funzione di valutazione dell'adeguatezza degli individui in modo da generarne sempre di migliori e di evitare le duplicazioni degli stessi;
- **Ricombinazione genetica (*crossover*)**: i geni di due individui selezionati per la riproduzione vengono scambiati tra di loro in modo da far evolvere la popolazione e consentire l'esplorazione di nuove porzioni di spazio;
- **Mutazione genetica**: introduce ulteriori cambiamenti che intervengono con maggiore rarità sui geni. In questo modo si arricchisce la varietà degli individui presenti nella popolazione evitando che quest'ultima tenda ad essere troppo uniforme e perda così ogni capacità di evolvere.

La Figura 47 rappresenta il processo che descrive l'operare degli AG che può essere suddiviso nelle seguenti sei fasi:

1. Si genera casualmente la popolazione iniziale di individui (genomi);
2. Per ogni individuo si calcola la fitness rispetto al problema da risolvere;
3. Si applica l'operatore genetico della selezione che, tenendo conto delle singole *fitness*, identifica gli individui destinati a vivere e a morire;
4. Attraverso la ricombinazione gli individui sopravvissuti si riproducono facendo nascere nuove soluzioni;
5. Con la mutazione diventa possibile registrare un'improvvisa modifica di una o più soluzioni;
6. Le soluzioni figlie costituiscono una nuova popolazione di individui e nella nuova popolazione viene ripetuta la sequenza a partire dal secondo punto.

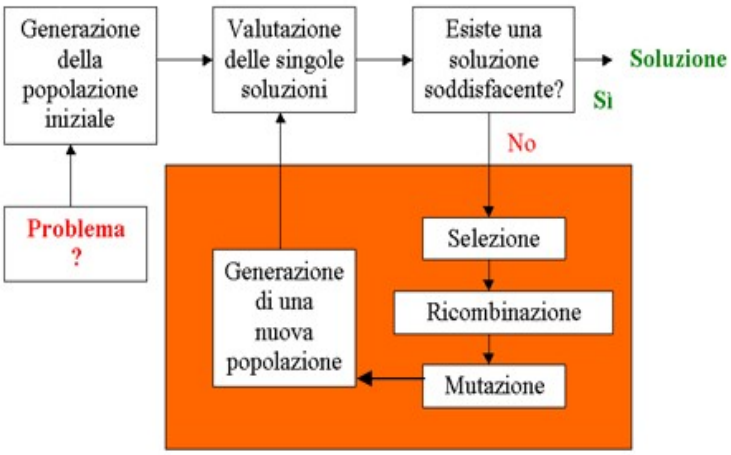


Figura 47. Rappresentazione del funzionamento dell'algoritmo genetico.

Nell'analisi del rischio di insolvenza gli algoritmi genetici sono stati utilizzati su due fronti:

- La generazione genetica di funzioni lineari;
- La generazione genetica di score basati su regole.

Nel primo caso la funzione genetica lineare assume la forma seguente:

$$GLS = a_0 + a_1 R_{a1} + a_2 R_{j2} + \cdots + a_n R_m \qquad (4.8)$$

dove a_0 indica la costante; a_i indica il coefficiente *i*-esimo e R_{ki} indica il *k*-esimo indicatore della *i*-esima famiglia di indicatori.

L'algoritmo genetico deve scegliere la costante a_0, i coefficienti a_i e gli *n* indicatori tratti dalle *n* famiglie, l'utente stabilisce a priori i segni dei coefficienti a_i (ma non quello della costante), il numero (*n*) delle famiglie e la lista degli indicatori appartenenti a ciascuna famiglia.

Nel caso invece di generazione di score basati su regole, gli algoritmi genetici sono stati utilizzati per produrre un insieme di regole basate su *test* riguardanti il segno ed il valore di chiusura degli indicatori selezionati.

Quanto esposto in questo paragrafo non è esaustivo ma dovrebbe bastare per capire cosa sono gli AG ed il loro modo di operare.

Per approfondimenti su questo argomento si vedano: Goldberg e Holland (1988); Conn et al. (1997).

4.10. Valutazione dei metodi di classificazione

In questo paragrafo viene analizzata la bontà di un modello o di più modelli di classificazione/previsione. In generale questo tipo di valutazione, che non è statisticamente legata ai parametri del modello, viene fatta sulle tecniche supervisionate e si tratta delle matrici di confusione e delle curve ROC.

4.10.1. La matrice di confusione

La matrice di confusione (Parker, 2001) confronta in modo molto semplice i risultati empirici e quelli teorici, cioè gli scostamenti tra i risultati reali e quelli ottenuti dal modello. Creiamo un *dataset* di 1.000 elementi di cui conosciamo 2 variabili. Il nostro output, che conosciamo, è di 4 variabili. Costruiamo un modello a rete neurale artificiale *feed-forward* con uno strato nascosto formato da 20 neuroni. La Figura 48 mostra la matrice di con- fusione per i 4 output. Le celle rosse indicano gli errori, mentre quelle verdi indicano la corretta classificazione.

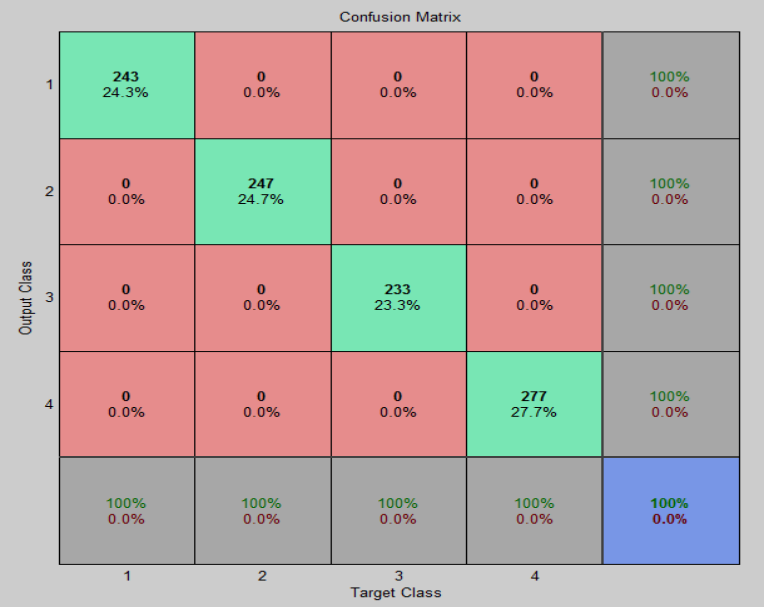


Figura 48. La matrice di confusione.

Come si può vedere, il modello a reti neurali ha perfettamente classificato gli elementi e definito gli output correttamente.

Se consideriamo il caso in cui abbiamo un solo output binario (generalmente è il caso più frequente) la matrice di confusione o di contingenza avrà la forma rappresentata in Tabella 11. Tra le grandezze definite per valutare le prestazioni di un classificatore le più frequenti sono le seguenti[31]:

Classificazione	**Classi effettive**	
	p	**n**
p'	Veri Positivi (VP)	Falsi Positivi (FP)
n'	Falsi Negativi (FN)	Veri Negativi (VN)

Tabella 11. Matrice di confusione - Positivi e Negativi.

$$\text{Sensitivity} = \Pr(p'|p) = \frac{VP}{VP+FN} \tag{4.9}$$

$$\text{Specificity} = \Pr(n'|n) = \frac{VN}{FP+VN} \tag{4.10}$$

$$\%\ \text{VP previsti} = \Pr(p|p') = \frac{VP}{VP+FP} \tag{4.11}$$

$$\%\ \text{VN previsti} = \Pr(n|n') = \frac{VN}{FN+VN} \tag{4.12}$$

$$\text{Tasso di FP per VN} = \Pr(p'|n) = \frac{FP}{FP+VN} \tag{4.13}$$

[31] Poiché spesso in uscita non abbiamo un valore 0 o 1, si utilizza una soglia di *cut-off* che risulta essere, nelle valutazioni standard, pari a 0,5.

$$\text{Tasso di FN per VP} = \Pr(n'|p) = \frac{FN}{VP+FN} \quad (4.14)$$

$$\text{Tasso di FP per classificati positivi} = \Pr(n|p') = \frac{FP}{VP+FP} \quad (4.15)$$

$$\text{Tasso di FN per classificati positivi} = \Pr(p|n') = \frac{FN}{FN+VN} \quad (4.16)$$

La *Sensitivity* (eq. 4.9) indica quanti valori positivi il modello è stato in grado di classificare correttamente su tutti i valori positivi che gli sono stati sottoposti da analizzare. Immaginiamo di stare valutando un campione di pazienti di cui sappiamo se hanno il morbillo (1) oppure no (0). La sensitività ci dice quanti 1 (cioè quanti malati) il modello è riuscito ad identificare rispetto al totale dei pazienti infetti.

La *Specificity* (eq. 4.10) indica quanti valori negativi il modello è stato in grado di classificare correttamente su tutti i valori negativi sottoposti ad analisi. Nell'esempio precedente, quanti non malati il modello è stato in grado di individuare in modo esatto.

La *% VP previsti* (eq. 4.11) confronta il numero dei positivi classificati correttamente con il totale dei classificati positivi. Considerando l'esempio, confronta il numero dei pazienti classificati correttamente infetti sul totale dei pazienti classificati infetti.

La *% VN previsti* (eq. 4.12) indica il numero dei negativi classificati correttamente con il totale dei classificati negativi. Nell'esempio, mostra la percentuale di pazienti classificati correttamente non infetti sul totale dei pazienti classificati non malati.

Il *Tasso di FP per VN* (eq. 4.13) rapporta il numero di classificati non correttamente positivi con il totale dei negativi presenti nel campione. Riportiamolo all'esempio, il tasso ci dice la percentuale dei pazienti che sono stati incorrettamente considerati infetti sul totale dei sani presenti nel *database*.

Il *Tasso di FN per VP* (eq. 4.14) indica il rapporto tra il numero di classificati non correttamente negativi con il totale dei positivi presenti nel campione. Nell'esempio indica la percentuale dei pazienti che sono stati incorrettamente considerati sani sul totale dei malati presenti nel *dataset*.

Il *Tasso di FP per classificati positivi* (eq. 4.15) indica il rapporto tra coloro che sono stati classificati incorrettamente positivi sul totale dei classificati positivi. Mostra la percentuale, considerando l'esempio, dei pazienti che sono stati classificati non correttamente malati sul totale dei soggetti classificati malati.

Il *Tasso di FN per classificati negativi* (eq. 4.16) mostra il rapporto tra coloro che sono stati classificati incorrettamente negativi sul totale dei classificati negativi. Per quanto concerne l'esempio, mostra la percentuale dei pazienti che sono stati classificati non correttamente sani sul totale dei soggetti classificati sani.

4.10.2. La curva *Receiver Operating Characteristics* (ROC)

La curva ROC (Yonelinas, 1994) è una metrica utilizzata per verificare la qualità della classificazione.

Le curve ROC mostrano graficamente il valore di *sensitivity* vs. 1-*specificity* per tutti i possibili *cut-o*ff.

L'area sotto la curva (AUC: *Area Under the Curve*) si usa per valutare la capacità di discriminazione ed è anche detta "statistica c". In sostanza, l'AUC equivale alla probabilità di fare una previsione corretta di effetto/non effetto basata sulle probabilità stimate dal modello.

La regola pratica dice che se:

- AUC=0,50: il modello non ha alcuna capacità discriminatoria;
- AUC compreso tra 0,70 e 0,80: discriminazione accettabile;
- AUC compreso tra 0,80 e 0,90: discriminazione eccellente;
- AUC>0,90: discriminazione fenomenale, quasi mai raggiungibile in pratica.

Supponiamo di avere un *dataset*[32] formato da 189 bambini di cui conosciamo il peso in grammi alla nascita ("Peso", variabile continua); l'età della mamma ("Età", variabile continua); il peso della mamma all'inizio della gravidanza ("Peso_mamma", variabile continua); la razza della mamma ("Razza", variabile qualitativa che può assumere 3 valori: bianca, nera, altro); se la mamma ha fumato in gravidanza ("Fumo", variabile dicotomica: Sì=1 e No=0); se la mamma aveva già avuto parti prematuri ("Prematuro", variabile discreta: numero di parti prematuri precedenti); se la mamma soffriva/soffre di ipertensione ("Ipertensione", variabile dicotomica: Sì=1 e No=0); se la mamma ha (avuto) problemi di irritabilità uterina ("Irritabilità", variabile dicotomica: Sì=1 e No=0); numero di visite nel primo trimestre della gravidanza ("Visite", variabile discreta).

Vogliamo sapere se esiste una relazione tra il peso del neonato e le caratteristiche della mamma.

In particolare vogliamo sapere se esistono dei fattori che possono influenzare la nascita sottopeso o meno e per questo motivo costruiamo una nuova variabile dicotomica "Peso_sotto" che assumerà valore pari a 1 se il neonato peserà meno di 2.500 grammi e pari a 0 se il suo peso sarà superiore.

Creiamo il nostro modello *logit* in cui la variabile "Peso_sotto" è la variabile dipendente (y) e le altre variabili le indipendenti (x_i).

I risultati (Tabella 12) ci dicono che vi sono delle relazioni tra Peso_mamma, Razza, Fumo, Ipertensione e Irritabilità. Tutte relazioni positive tranne la prima.

Calcoliamo la curva ROC che viene presentata in Figura 49. La bisettrice indica il modello inutile mentre la curva indica la capacità del modello di classificare e infatti abbiamo un AUC pari a 0,75 circa che, secondo i nostri parametri, è un risultato accettabile. Esistono altre curve che analizzano l'andamento di *specificity* e *sensibility* contemporaneamente in base ai *cut-off* e che rendono visivamente bene l'idea di quale sia il valore soglia migliore, tuttavia, la curva ROC è un semplice metodo per testare se il modello ha una buona capacità di classificazione.

A questo punto, per esercizio, proviamo a calcolare la matrice di confusione per il nostro modello (Tabella 13) e i successivi indicatori di sensitività e specificità (eq. 4.17 ed eq. 4.18). Come si può notare il modello ha un valore si *sensitivity* piuttosto basso pari al 35,59%. Questo indica la capacità di corretta identificazione e quindi classificazione dei bambini con peso inferiore ai 2,5 kg.

La *specificity* invece è elevata (90,77%) e quindi il modello riesce a individuare più facilmente i bambini che pesano più di 2,5 kg.

[32] Il modello è stato costruito con STATA/SE 12.0.

VARIABILI	Peso_sotto (Dev.Std)
Età	-0.0271
	(0.0365)
Peso_mamma	-0.0152**
	(0.00693)
Razza (nera)	1.263**
	(0.526)
Razza (altro)	0.862**
	(0.439)
Fumo	0.923**
	(0.401)
Prematuro	0.542
	(0.346)
Ipertensione	1.833***
	(0.692)
Irritabilità	0.759*
	(0.459)
Costante	0.461
	(1.205)
N	189
Log-likelihood	-100,724
$\chi^2_{(8)}$	33,224

*** p<0,01, ** p<0,05, * p<0,1

Tabella 12. Modello Logit.

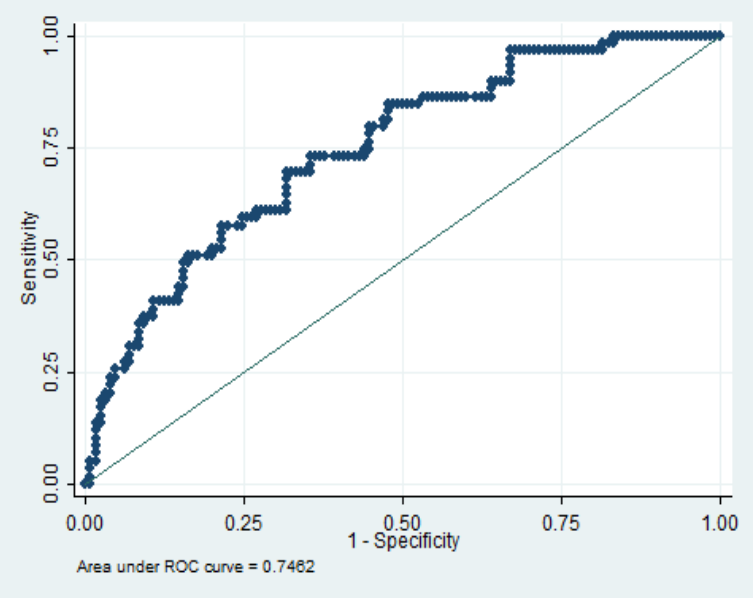


Figura 49. La curva ROC.

Classificazione	Classi effettive	
	p	**n**
p'	21	12
n'	38	118

Tabella 13. Matrice di confusione - Positivi e Negativi.

$$Sensitivity = \Pr(p'/p) = \frac{VP}{VP+FN} = \frac{21}{21+38} = 35{,}59\% \quad (4.17)$$

$$Specificity = \Pr(n'/n) = \frac{VN}{FP+VN} = \frac{118}{12+118} = 90{,}77\% \quad (4.18)$$

5. IL *TEXT MINING* E IL *WEB MINING*

In questo paragrafo[33] verrà trattato l'argomento del *Text Mining* in quanto la statistica può essere applicata non solo a dati numerici ma anche a *dataset* che contengono dati. Inoltre si farà un breve accenno al *Web Mining*.

5.1. Il *Text Mining*

Il *Text Mining* è l'insieme di tecniche e metodi usati per il processamento automatico del linguaggio testuale naturale disponibile in grande quantità su file elettronici. Ha come obiettivo quello di estrarre e strutturare i contenuti e i temi per una rapida analisi (di tipo non letterario), la scoperta di informazioni nascoste e/o per rendere automatiche alcune decisioni.

Diversa è la *stylometry* che studia lo stile dei testi con lo scopo di individuare l'autore dal brano ma ha molto in comune con la *lexicometry* (o statistica lessicale, o linguistica statistica o linguistica quantitativa). Difatti il *Text Mining* è l'estensione di quest'ultima scienza all'utilizzo di metodi statistici multidimensionali. È quindi valida l'espressione:

Text Mining = Lexicometry + Data Mining

Sempre più i ricercatori hanno utilizzato tecniche statistiche e inferenziali per l'analisi testuale e da allora sono stati fatti enormi progressi rispetto al semplice calcolo delle percentuali. Il *Text Mining* nasce nel 1912 quando Estoup e in seguito Zipf (1999) iniziarono a calcolare le frequenze delle parole nei testi, studiandole successivamente con metodi statistici. È famoso l'esempio dell'Ulisse di Joyce la cui decima parola appare 2,653 volte, la centesima 265 volte, la millesima 26 volte e la decimillesima 2. A questo proposito, si può dimostrare che il prodotto tra il "rango" r e la frequenza f è virtualmente costante:

$$r \cdot f = costante \quad (5.1)$$

[33] Quanto esposto in questo paragrafo prende spunto dal web e dal libro di Tufféry (2011).

Questa non è una legge valida con sempre lo stesso grado di accuratezza ma è comunque veramente universale perché si applica a tutti i tipi di testo e per tutti i linguaggi. A questo proposito, Li (1992) ha dimostrato che questa regola può essere applicata al testo in cui le parole sono create in modo random partendo da un alfabeto con distribuzione uniforme.

La formula 5.1 dunque deve essere ora modificata nel seguente modo:

$$r^{\alpha} \cdot f = costante \qquad (5.2)$$

dove α è un esponente che dipende dal linguaggio. Generalmente varia tra 1,1 e 1,3 e si avvicina a 1,6 nel linguaggio dei bambini. Come regola generale ci si può ricordare che decresce con l'arricchirsi del *corpus* testuale, misurato come il rapporto tra il numero di parole diverse V ("vocabolario") e il numero totale di parole N (V è in genere proporzionale alla radice quadrata di N).

Per quanto concerne il *Text Mining* esistono due tipi di metodologie:

1. **Metodi descrittivi**. Questi possono essere utilizzati per la ricerca di temi/argomenti racchiusi in un testo senza conoscerli in anticipo;
2. **Metodi predittivi**. Questi trovano regole per associare automaticamente il documento ad un insieme di temi predefiniti. Questa metodologia potrebbe essere utile nel caso in cui, per esempio, siamo nell'ufficio risorse umane di un'impresa e stiamo raccogliendo dei CV per assumere personale. Potrebbe essere utile creare un algoritmo in modo che il CV finisca direttamente al dipartimento interessato, senza doverlo aprire ed analizzare.

In questo caso il corpus analizzato deve incontrare le seguenti condizioni:

- Deve essere in un formato leggibile dai processori;
- Deve includere un numero minimo di testi;
- Deve essere sufficientemente comprensibile e coerente;
- Non devono esserci troppi temi diversi in ogni testo;
- Deve evitare il più possibile l'uso dell'ironia, delle allusioni e delle antifrasi.

Le principali risorse di testi analizzati sono sondaggi di opinione, raccolte sulla soddisfazione dei clienti, lettere di lamentele, trascrizioni di interviste telefoniche e non, e-mail, raccolta di riviste/giornali in formato elettronico, *database* presenti sul *web*.

Alcune analisi periodiche possono essere effettuate automaticamente attraverso un algoritmo di *Text Mining* purché i file di input siano standard. In questo modo, la tecnica analizzata genera veloci analisi senza che sia necessario ripetere le identiche operazioni.

Come precedentemente accennato, il *Text Mining* può scoprire dell'informazione nasco- sta che può essere "recuperata" (*Information retrieval*) o "estratta" (*Information extraction*). In dettaglio:

1. Il primo tipo (*Information retrieval*) tratta i documenti nella loro totalità insieme ai temi di cui si occupano. Viene utilizzato per confrontare documenti e identificare il loro tipo. L'obiettivo è quello di rilevare tutti i temi che sono presenti. In questo caso l'analisi è globale.
2. La seconda procedura (*Information extraction*) ricerca una specifica informazione all'interno dei documenti senza alcun confronto tra gli stessi. Considera la prossimità tra le parole per discriminare tra diverse asserzioni che hanno identiche parole- chiave. Questo algoritmo si interessa solamente dei temi legati al *database*. L'*information extraction* parte dal linguaggio naturale e lo usa per costruire un data- base strutturato. In sostanza è un modo per esplorare

il linguaggio testuale naturale trovando le parole o le frasi corrispondenti ad ogni campo del *database*. L'analisi in questo caso è locale.
Questo è un processo più complesso perché richiede l'uso di un'analisi morfo-sintattica in grado di riconoscere i costituenti del testo (parole e frasi), la loro natura e le loro relazioni.

5.1.1. Analisi linguistica e relativi problemi

È chiaro a tutti il fatto che quando si è sul web non si parla una sola lingua ma molte e che anche all'interno dello stesso linguaggio esistono dei modi di dire o delle espressioni che hanno significati differenti. Per esempio, per i Canadesi inglesi la scritta "Garage Sale" significa garage in vendita ma se viene letta dai canadesi francesi questa significa "*Dirty Garage*" cioè garage sporco. Molti dei testi che vengono utilizzati per le analisi di *Data Mining* provengono dal web e possono soffrire di questi problemi.

Diventa inoltre fondamentale riuscire ad identificare nomi e verbi, aggettivi ed avverbi nei testi da analizzare e questa fase richiede un'analisi grammaticale. Questo lavoro può essere complicato dalla presenza di termini omografi. Ad esempio, "mi piace leggere" e "queste buste sono leggere".

Non solo, vi sono nei testi molte forme di "disambiguità" che può essere dovuta a "polisemia" delle parole (una parola ha molti significati), ad "ellissi" (si verifica quando si sottintende una parola o un verbo in una frase), ad "anafora" (consiste nel riprendere o ripetere all'inizio di frasi o di versi successivi una parola o un'espressione), ad "antifrasi" (il significato di una parola è l'opposto di quello che assume normalmente), ad ironia e così via.

Un'ulteriore difficoltà si incontra nel formato dei dati da processare in quanto può dare origine ad ambiguità, pensiamo per esempio alla lettera O (o maiuscola) e al numero 0 (zero); alle parole che vengono divise dal trattino "-" ma che a volte sono scritte come una sola parola (algoritmo di *back-propagation* o *backpropagation*); ad errori ortografici ma anche al diverso significato morfologico dei termini (ad esempio "viaggio" può essere un nome maschile singolare ma anche la prima persona singolare dell'indicativo presente del verbo viaggiare); ad abbreviazioni. Anche la struttura sintattica delle frasi pone problemi perché non è ovviamente sempre la stessa ma ognuno sviluppa un personale modo di scrivere/parlare.

Esistono inoltre delle espressioni in cui le parole che le compongono sono separate ma che in realtà hanno significato solo in coppia, ad esempio "conto corrente". Sono due nomi che hanno un loro significato proprio ma usati insieme ne acquistano un altro specifico. Stessa cosa accade con le date: "2 Agosto 2013" può essere vista come l'insieme di 3 parole che hanno un significato a se stante ma che insieme indicano un preciso momento. Per questi motivi, nel momento della definizione di un modello di *Text Mining* è necessario costruire un dizionario che tenga conto anche di queste espressioni. Può essere utile creare dei dizionari in base al tema, ad esempio uno per il business in cui vi sono tutti i termini e le espressioni legate a quel mondo e così via.

A questo punto si procede con la semplificazione del processo che avviene secondo le seguenti fasi:

1. **Lemmatizzazione:** la riduzione di una forma flessa di una parola alla sua forma canonica (non marcata). La forma è la stessa che si trova su un dizionario, in modo che sia più facilmente possibile capire l'argomento trattato nel testo;
2. **Raggruppamento:** la definizione dei gruppi delle varianti che si trovano nel testo. Ad esempio in inglese possiamo trovare "realize" o "realise", entrambe le forme sono corrette

perché la prima è secondo lo stile americano e la seconda segue le regole britanniche. Possono esservi inoltre: varianti sintattiche (ad esempio: “sono di Torino” e “sono torinese”), varianti semantiche (“io ti vendo un libro” e “tu comperi un libro da me”), sinonimi, punti negli acronimi (ad esempio “SPA” e “S.P.A.”). Questi sono tutti problemi che è necessario affrontare per semplificare e migliorare i risultati del processo di analisi;

3. **Raggruppamento per analogie:** la creazione di famiglie di nomi derivati che possono essere seguiti da un segno che definisce se l’intensità è crescente o no. Ad esempio: “un pochino, di meno, pochissimo/-” o “molto, di più moltissimo/+”;
4. **Identificazione degli argomenti/temi**: a questo punto si procede con l’analisi del testo raggruppando in base a livelli. Ad esempio:

   Livello 1:
   Assegno/Carta di credito/Cambiale/Denaro/… →*Mezzi di pagamento*
   Livello 2:
   Mezzi di pagamento/Soldi/Cassa/Conto corrente/…. →*Banca*

5.1.2. Applicazioni statistiche e *Data Mining*

Quando l’analisi dei testi e degli argomenti è terminata, si possono filtrare i temi e i ter- mini perché vengano elaborati utilizzando sia criteri statistici, sia criteri semantici, sia un corpus. Nel primo caso si selezionano i termini e i temi in base alla loro frequenza. In questo caso è possibile definire delle regole per pesare ogni termine, per esempio si può decidere di preferire i termini che compaiono più volte ma in pochi testi e in questo caso il peso sarà uguale alla frequenza del termine o il numero dei testi che lo contengono. I criteri semantici invece sono focalizzati su uno specifico soggetto o su un corpus di lemmi che identifica un preciso insieme di parole come ad esempio quelle offensive ed i loro derivati in modo da pulire il documento.

Essendo stati precedentemente trattati, i termini vengono ora considerati come veri e propri dati su cui è possibile applicare le tecniche di *Data Mining*. Gli individui, cioè gli elementi, sono i testi i documenti; mentre le variabili o attributi sono gli argomenti o i termini dei documenti. Così facendo è possibile creare delle tabelle lessicali in cui ogni cella c_{ij} rappresenta il numero di presenze del termine j (o un indicatore della presenza/assenza) nel documento i. La cella c_i può anche essere il numero di presenze del termine j nell’insieme di documenti relativi all’i-esimo individuo (lettere, report di interviste, etc.).

Queste tabelle possono essere analizzate tramite l’analisi delle corrispondenze che semplifica il problema della riduzione delle variabili iniziali (cioè dei termini) che restano comunque numerose anche in seguito al pre-processamento iniziale. Al termine di questo processo, le variabili continue vengono generalmente trasformate in discrete e si procede quindi all’applicazione delle tecniche standard per la classificazione.

Il *Text Mining* può rispondere a due tipi di richieste:

1. ***Open Requests*** o ***Free Text Requests*** che sono interrogazioni in base a parole- chiave o parti di testo utilizzate per cercare documenti rilevanti compresi in un corpus che si modifica lentamente (ad esempio una biblioteca elettronica).
   Una metodologia utilizzata ed efficace per risolvere questi problemi è rappresentata dalle Catene di *Markov* che descriviamo brevemente. Immaginiamo di avere n scatole, ognuna delle quali contenente delle palle numerate. Estraiamo casualmente una palla dalla prima

scatola, il numero segnato sulla palla indica la scatola da cui verrà estratta la palla successiva. Proseguiamo in questo modo finché le scatole non sono vuote. La sequenza di scatole (il numero assegnato ad ogni scatola), estratta in questo modo casuale, rappresenta la catena di *Markov* del nostro esercizio. La probabilità di estrarre una determinata palla dipende dalla scatola da cui è estratta e quindi da tutte le precedenti estrazioni. Lo stesso procedimento viene applicato alle frasi: la probabilità della comparsa di una parola dipende dalle precedenti parole e non tutte le sequenze di parole hanno la stessa probabilità di comparire. Le catene di *Markov* sono utilizzate per il riconoscimento vocale e della grafia, per la correzione ortografica ed in generale per le interfacce vocali;

2. ***Predefined Requests*** che sono interrogazioni relative al numero di termini fissi, applicate ad un corpus che cambia in un modo dinamico nel tempo (ad esempio la categorizzazione di documenti o il filtro di email).
   Le tecniche più frequentemente utilizzate sono le reti neurali artificiali e gli alberi decisionali. In particolare, le reti d Kohonen (si veda il paragrafo 4.8.3, p. 55) e la metodologia del *Hierarchical clutering* (paragrafo 4.2.1, p. 27) sono le più utilizzate.

5.1.3. *Information Extraction* (IE)

I sistemi di *information extraction* hanno l'obiettivo di estrarre informazione strutturata da un corpus non strutturato di testi.

Questi sistemi sono piuttosto complessi in quanto necessitano di dizionari semantici specifici per l'argomento analizzato che si vuole analizzare ma anche degli algoritmi di ana- lisi sintattica in grado di riconoscere le forme linguistiche generali (verbo, soggetto, complementi, …).

Definiti i testi su cui estrarre l'informazione e fissati i campi[34] da completare, il sistema di *information extraction* evidenzia le frasi rilevanti ed estrae le informazioni desiderate.

Poiché questa tecnica riguarda soprattutto l'analisi del linguaggio naturale (*Natural Language Processing*, NLP), le recenti tecnologie multimediali testuali come l'annotazione automatica o l'estrazione di informazione da video, immagini e audio, possono essere considerate sistemi di *information extraction.*

I principali campi di applicazione di questi sistemi sono:

- Il completamento automatico di forme predefinite estraendo da un testo in forma libera;
- La costruzione automatica di *database* bibliografici da articoli scientifici (ad esempio in questo caso alcuni campi da completare possono essere; titolo, autore, giornale, data di pubblicazione, etc. …);
- La scansione automatica della stampa finanziaria;
- La rilevazione automatica delle idee o delle richieste dei consumatori di uno specifico prodotto/servizio in base ai dati di vendita.

I risultati che si ottengono da questi sistemi sono sintetizzati da due indicatori;

1. L'***accuracy rate*** che è rappresentato dal rapporto:

$$AR=\frac{\text{numero di campi correttamente completati}}{\text{numero totale di campi completati}} \qquad (5.3)$$

[34] Nel linguaggio dei *database*, i "campi" corrispondono alle variabili di una tabella cioè alle colonne mentre i "record" sono le righe.

2. Il ***recall rate*** che invece il risultato della seguente divisione:

$$RR = \frac{\text{numero di campi correttamente completati}}{\text{numero totale di campi (completati e non)}} \quad (5.4)$$

5.2. Il *Web Mining*

Il *Web Mining* è l'applicazione delle tecniche di *Data Mining* ai dati ottenuti dai server Internet sul modo in cui gli utenti utilizzano il web. Può essere utilizzato per analizzare i comportamenti dei *web user* e quindi servire in fase di valutazioni manageriali. Pensiamo ai proprietari di siti commerciali per i quali è importante conoscere quali prodotti sono maggiormente visualizzati e non solo acquistati in quanto indicano interesse da parte dell'utente per i prodotti/servizi mostrati.

Tuttavia, con la sempre maggior diffusione dei social network, quali *Facebook, Twitter*, etc. il *Web Mining* ha assunto un ruolo fondamentale per effettuare analisi sulle idee e opinioni di coloro che frequentano il web.

Utilizzando le regole associative (si veda il paragrafo 4.1, p. 24) e quindi estraendo delle regole di tipo "*if… then…*" è possibile creare delle regole come ad esempio "gli utenti che hanno letto l'articolo a, hanno letto anche il b con la percentuale x. Coloro che non hanno letto l'articolo b, per una percentuale y hanno letto un articolo correlato, i rimanenti no". Questa analisi può essere utilizzata per creare delle tassonomie di utenti ma anche degli stessi siti internet.

Si tratta di indagini globali che si basano su dei *log-file*[35] in cui sono racchiuse tutte le informazioni relative al sito. Questi documenti sono salvati direttamente sul server e le righe contengono le richieste degli utenti (per esempio: per cambiare una pagina, per scaricare un file).

Il formato più comune di *log-file* contiene l'indirizzo IP del computer dell'utente, la data e l'ora della richiesta, il tipo di richiesta, il sito richiesto, il protocollo HTTP, il codice che rappresenta la risposta del server e la dimensione dell'oggetto richiesto.

Una linea di un ipotetico *log-file* è la seguente:

```
130.5.48.74 [22/May/2006:12:16:57 -0100] "GET
/content/index.htm HTTP/1.1" 200 1243
```

Le prime cifre indicano l'indirizzo IP dell'utente, seguono la data e l'ora della richiesta (nel formato ora, minuti, secondi) nonché l'indicazione che rispetto al GMT (*Greenwich Mean Time*) siamo a -1 ora. La stringa *GET* indica che è stato richiesto un download di un oggetto di 1243 bit dal sito */content/index.htm* con protocollo *HTTP versione 1.1*. Inoltre l'operazione è andata a buon fine (codice 200).

La versione estesa del log-file riporta inoltre il nome e la versione del browser utilizzato. Anche se l'esempio è molto semplice, si intuisce che le informazioni contenute nel log-file sono decisamente interessanti e possono essere molto utili per diversi scopi, soprattutto se aggregate in *database* e analizzate con le tecniche di *Data Mining*.

[35] I *log-file* sono dei file di testo e ne esistono di differenti tipi.

Difatti, le analisi possono non solo essere di supporto alle decisioni manageriali ma anche per analizzare il tessuto sociale, i comportamenti degli utenti e le loro opinioni.

6. APPENDICE STATISTICA

6.1. Algoritmi di *Binning*

Questi algoritmi prendono il nome dagli intervalli (*bin*) in cui vengono suddivisi i possibili valori assunti da una variabile. Se infatti consideriamo un attributo a_i dove i rappresenta un indice che va da 1 a N e i cui valori sono ordinati, possiamo procedere fissando un valore *d* (*depth*) che consente di suddividere i valori della variabile in intervalli (i.e., *bin*). Questi intervalli (*I*) contengono dunque all'incirca *d* elementi e pertanto si otterranno *N/d* intervalli. A questo punto diventa fondamentale definire una funzione di *smoothing* che permetterà di sostituire ad ogni elemento della variabile α un valore derivato dall'intervallo in cui si trova.

Le funzioni di *smoothing* più note sono le seguenti (Dulli et al., 2009):

- ***Smoothing by bin means***: che prevede di sostituire ai valori di α la media dell'intervallo in cui si trova:
  $a_i = \mu(I_{\mathrm{u}}(i))$; (6.1)
- ***Smoothing by bin medians***: che prevede di sostituire ai valori di α la mediana del corrispondente intervallo:
  $a_i = Me(I_{\mathrm{u}}(i))$; (6.2)
- ***Smoothing by bin boundaries***: che prevede di sostituire ai valori di α l'estremo più vicino dell'intervallo in cui si trova:

$$\begin{cases} a_i = \min\left(I_{\cup(i)}\right) \text{ se } (a_i - \min\left(I_{\cup(i)}\right) < \mathrm{Max}\left(I_{\cup(i)}\right) - a_i) \\ a_i = \mathrm{Max}\left(I_{\cup(i)}\right) \text{ altrimenti} \end{cases} \quad (6.3)$$

6.2. Le medie

Media quadratica

$$\bar{x}_q = \frac{1}{N} \cdot \sum_{i=1}^{N} x_i^2 \quad (6.4)$$

Media geometrica

$$\bar{x}_{geom} = \sqrt[N]{\prod_{i=1}^{N} x_i} \quad (6.5)$$

Media aritmetica

$$\bar{x}_h = \frac{N}{\sum_{i=1}^{N} \frac{1}{x_i}} \quad (6.6)$$

6.3. Selezione degli attributi rilevanti

Questa fase di analisi dei dati permette di individuare quali sono le variabili rilevanti che rappresentano realmente i dati. In questo paragrafo vengono brevemente descritte le due tecniche più utilizzate per la selezione degli attributi rilevanti (Dulli et al., 2009):

- ***Step-wise forward selection***. Questo algoritmo prevede che si parta inserendo nell'analisi una variabile alla volta. In questo modo, è possibile confrontare il risultato prima e dopo l'inserimento di ogni variabile e verificare se l'attributo aggiunto effettivamente contribuisce alla conoscenza del *database*. Questa operazione viene effettuata finché non si esauriscono tutte le variabili e non si ottengono dei risultati accettabili;
- ***Step-wise backward selection***. Questo approccio invece funziona esattamente al contrario rispetto a quello precedentemente presentato. Si parte infatti dall'insieme completo delle variabili e si prosegue eliminando una per volta le variabili e confrontando i risultati prima e dopo. Anche in questo caso, l'algoritmo si ferma quando si è raggiunto un risultato soddisfacente.

Queste tecniche sono molto utilizzate nel campo medico e nelle analisi di regressione per la valutazione del rischio.

6.4. Misure di distanza e similarità

6.4.1. Distanza Euclidea

Date due variabili *X* e *Y* di lunghezza *m*:

$$X = x_1, x_2, \ldots, x_m \; e \; Y = y_1, y_2, \ldots, y_m \tag{6.7}$$

stabiliamo quanto sono vicine utilizzando la distanza euclidea così definita (Danielsson, 1980):

$$D_E(X, Y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2} \tag{6.8}$$

6.4.2. Distanza di Minkowski

Siano *X* e *Y* due variabili di lunghezza *m*:

$$X = x_1, x_2, \ldots, x_m \; e \; Y = y_1, y_2, \ldots, y_m \tag{6.9}$$

stabiliamo quanto sono vicine utilizzando la distanza di Minkowski così definita (Ichino e Yaguchi, 1994):

$$D_M(X, Y) = \sqrt[r]{\sum_{i=1}^{m} (x_i - y_i)^r} \tag{6.10}$$

r è un parametro che può assumere:

- *r*=1: distanza di Manhattan;
- *r*=2: distanza euclidea;
- $r \to +\infty$: quando la distanza tra le componenti di un vettore è massima.

6.4.3. Distanza di Lagrange-Tchebychev

Siano *X* e *Y* due attributi di lunghezza *m*:

$$X = x_1, x_2, \ldots, x_m \; e \; Y = y_1, y_2, \ldots, y_m \tag{6.11}$$

stabiliamo quanto sono vicini utilizzando la distanza di Lagrange-Tchebychev (Rajola, 2003):

$$D_{LT}(X,Y) = \max_{1 \le i \le +\infty} |X_i, Y_i| \tag{6.12}$$

6.4.4. Distanza di Mahalanobis

Date due variabili *X* e *Y* di lunghezza *m*:

$$X = x_1, x_2, \ldots, x_m \; e \; Y = y_1, y_2, \ldots, y_m \tag{6.13}$$

stabiliamo quanto sono vicine utilizzando la distanza di Mahalanobis (Penny, 1996):

$$D_{Mah}(X,Y) = \sqrt{(X-Y) \cdot \sigma(X,Y)^{-1} \cdot (X-Y)^T} \tag{6.14}$$

dove $\sigma(X,Y)$ è la matrice delle covarianze e *T* è la dimensione del vettore quando *X* e *Y* sono multivariate.

6.4.5. Correlazione

Siano *X* e *Y* due variabili di lunghezza *m*:

$$X = x_1, x_2, \ldots, x_m \; e \; Y = y_1, y_2, \ldots, y_m \tag{6.15}$$

Il *coefficiente di correlazione lineare r* è pari alla covarianza calcolata sulle variabili standardizzate X^* e Y^* (Orsi, 1995).

La sua formulazione risulta dunque essere la seguente:

$$r = \frac{\sigma_{X,Y}}{\sqrt{\sigma_x \cdot \sigma_y}} \tag{6.16}$$

Si dimostra che *r* varia tra −1 e +1. Valori pari a ±1 indicano dipendenza lineare perfetta. Se *r* = −1 i punti sono perfettamente appaiati su retta con inclinazione negativa e è possibile concludere

che le due variabili sono legate da una relazione inversamente proporzionale; se $r = +1$ i punti si sono appaiati su una retta inclinata positivamente e pertanto se una variabile aumenta, aumenterà anche l'altra. Se $r = 0$ le variabili non sono correlate.

6.4.6. Distanza di Jaccard (per variabili dicotomiche)

Date due variabili dicotomiche *X* e *Y* di lunghezza *m*:

$$X = x_1, x_2, \ldots, x_m \; e \; Y = y_1, y_2, \ldots, y_m \tag{6.17}$$

le similarità tra di esse possono essere misurate definendo le seguenti quantità:

- M01 = numero degli attributi dove *X* ha 0 e *Y* ha 1;
- M10 = numero degli attributi dove *X* ha 1 e *Y* ha 0;
- M00 = numero degli attributi dove *X* ha 0 e *Y* ha 0;
- M11 = numero degli attributi dove *X* ha 1 e *Y* ha 1.

L'indice di Jaccard (Jaccard, 1901) può essere calcolato nel seguente modo:

$$J = \frac{M11}{M01+M10+M11} \tag{6.18}$$

In Figura 50[36] vengono mostrate le diverse distanze qui discusse su un *database* formato da 10 soggetti e 3 variabili.

Sono dunque state calcolate (m*(m − 1))/2 (cioè (10*9)/2) distanze, ordinate in modo crescente ed infine riportate su uno stesso grafico.

La distanza di Jaccard è stata esclusa perché riguarda variabili dicotomiche, mentre in questo caso le variabili considerate sono continue.

Si può notare come le metriche Euclidea, di Minkowski (r=3) e di Lagrange-Tchebychev diano dei risultati molto simili mentre la Mahalanobis e di Correlazione danno risultati molto differenti e quindi occorre valutare attentamente quale di queste distanze si utilizza quando si fa analisi dei dati. È bene inoltre sottolineare che esistono molte altre metriche (ad esempio la *cityblock*, la *cosine*, la *spearman* e la *hamming*. Per un approfondimento si consulti Berkhin, 2006) ma in questa appendice si è voluto semplicemente illustrare le più utilizzate. Inoltre è sempre comunque possibile definire delle distanze personalizzate.

[36] Quanto rappresentato nella figura è stato ottenuto con Matlab R2010a.

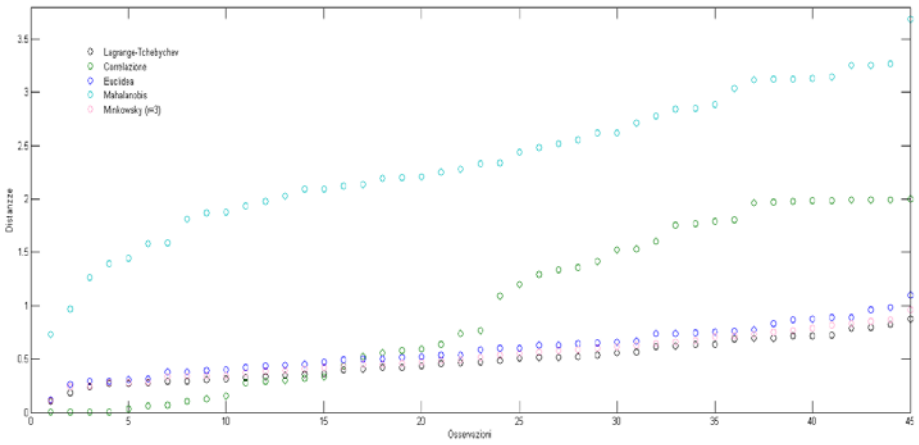


Figura 50. Distanze in base a diverse metriche.

6.5. Regole Associative: algoritmo apriori

Quanto esposto segue il testo di Agrawal et al. (1993, 1996) in cui è stato evidenziato che il problema legato alla scoperta delle regole associative si può decomporre in due fasi, sfruttando l'osservazione che:

$$C(x \to y) = \frac{S(x \cup y)}{S(x)} \tag{6.19}$$

dove ricordiamo che S è il supporto di un insieme, cioè la frazione di transazioni che lo contiene.

Si prosegue determinando l'insieme di item che hanno un supporto almeno pari al minimo supporto ottenuto e questi *itemset* vengono chiamati "*large itemsets*".

L'algoritmo apriori trova gli insiemi frequenti di oggetti (FI = *frequent itemset*) che soddisfino il vincolo sul supporto.

Un sottoinsieme di un FI è a sua volta un FI: se {x,y} è frequente anche {x} e {y} lo sono con supporto maggiore. Infine, iterativamente, l'algoritmo trova i FI con la cardinalità da 1 a K e li usa per generare le regole associative. La principale proprietà dell'algoritmo

Apriori è la seguente: se y è frequente e x ⊆ y, allora anche x è frequente. Ciò significa che ogni transazione che contiene y contiene anche x, quindi S (x) ≥ S (y). La conseguenza è che se x non è frequente, non è necessario generare e valutare gli insiemi che lo includono.

Consideriamo i seguenti item:

{1, 3, 4}

{2, 3, 5}

{1, 2, 3, 5}

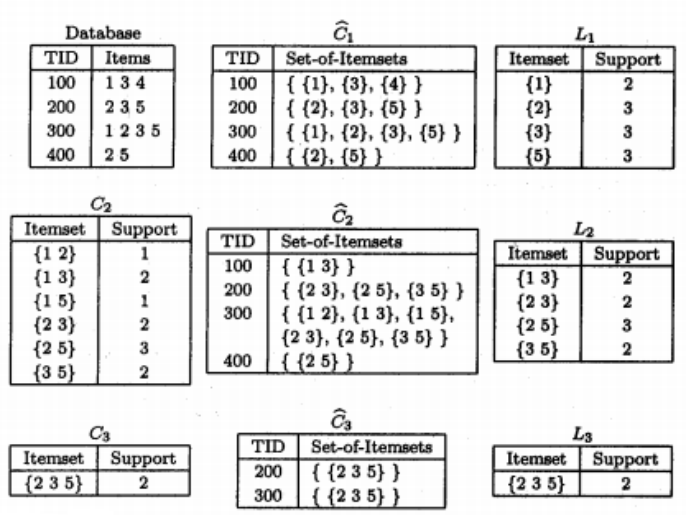
{2,3}

Nella Figura 51 viene rappresentato il modo in cui opera l'algoritmo apriori. L'insieme C1 verifica il supporto per ogni elemento. L1 è il nuovo insieme C1 da cui è stato tolto l'elemento con S inferiore. C2 costruisce gli *itemset* da L1 e ne calcola il supporto, L2 mostra C2 meno gli elementi con supporto più basso. Si costruisce quindi C3 che nel grafico è mostrato in forma

ridotta per motivi di spazio e L3 in cui associa il supporto. L3 rappresenta la regola (che per convenzione chiamiamo A). Da questa figura si evince che:

- ***Join Step***. C_k è generato congiungendo L_{k-1} con se stesso;
- ***Prune Step***. Ogni *(k−1)-itemset* che non è frequente, non può essere sottoinsieme di un *k-itemset* frequente.

Esistono diversi metodi per migliorare l'efficienza dell'algoritmo tra cui il *partitioning* (Savasere, 1995), il *sampling* (Toivonen, 1996) e il *Dynamic Itemset Counting* (Brin et al., 1997).



Fonte: Agrawal et al. (1996).
Figura 51. Apriori – esempio.

6.6. Funzioni di attivazione (*Transfer functions*)

In questo paragrafo vengono illustrate le più frequenti funzioni di attivazione[37]. Le principali funzioni di trasferimento sono:

***Positive Linear Transfer Function* (*poslin*)**. Si tratta di una funzione che assume solo valori positivi (da $-\infty$ a $+\infty$) ed ha una forma lineare. Se definiamo un vettore che va da -5 a +5 con passo 0.1 e costruiamo la funzione *poslin* su questi dati, questa avrà la forma rappresentata in Figura 52. Anche se il vettore inizia a -5, tutti i valori inferiori allo 0 vengono portati a 0. La funzione ha seguente forma matematica

$$f(x) = \begin{cases} x \; se \; x \geq 0 \\ 0 \; altrove \end{cases} \tag{6.20}$$

***Linear transfer function* (*purelin*)**. Questa funzione assume valori sia negativi che positivi ed è lineare (da $-\infty$ a $+\infty$). Considerando il vettore di prima il grafico corrispondente alla funzione è rappresentato in Figura 53.

[37] Quanto rappresentato nella figura è stato ottenuto con Matlab R2010a.

Log-sigmoid transfer function **(logsig)**. La funzione logistica assume valori tra 0 e 1 ed ha la seguente forma funzionale:

$$f(x) = \frac{1}{1+exp^{-x}} \tag{6.21}$$

Considerando l'esempio precedente, la Figura 54 rappresenta la funzione *logsig* per i dati analizzati.

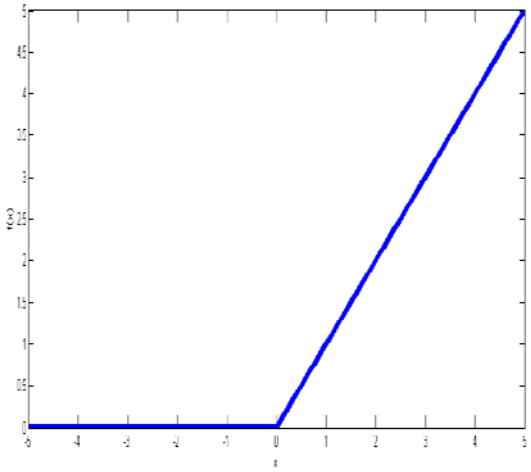


Figura 52. Positive Linear Transfer Function (poslin).

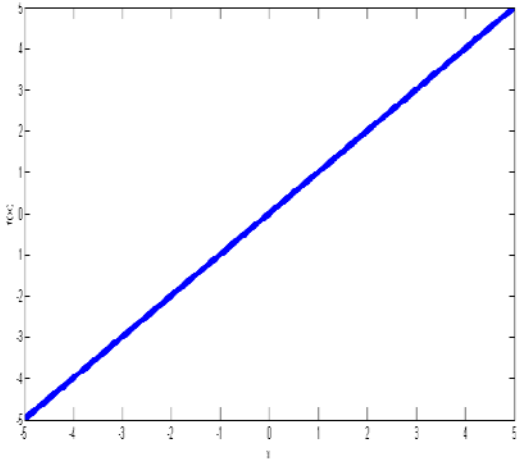


Figura 53. Linear Transfer Function (purelin).

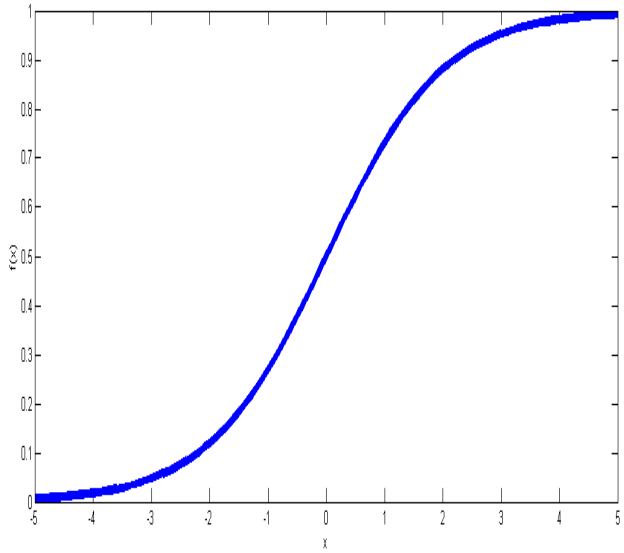


Figura 54. Log-sigmoid Transfer Funztion (logsig).

***Hard-limit transfer function* (*hardlim*).** La funzione *hardlim* ha la forma seguente:

$$f(x) = \begin{cases} 1 \; se \; x \geq 0 \\ 0 \; altrove \end{cases} \tag{6.22}$$

Utilizzando lo stesso vettore definito in precedenza, il grafico della funzione è rappresentato nella Figura 55.

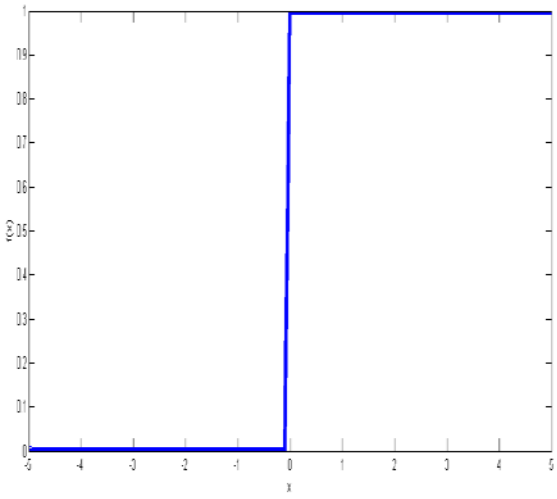


Figura 55. Hard-limit Transfer Function(hardlim).

***Hyperbolic tangent sigmoid transfer function* (*tansig*)**. I valori di questa funzione variano tra -1 e +1 e la forma funzionale è la seguente:

$$f(x) = \frac{2}{1+exp^{-2n}} - 1 \qquad (6.23)$$

La Figura 56 rappresenta il vettore di valori tra -5 e +5.

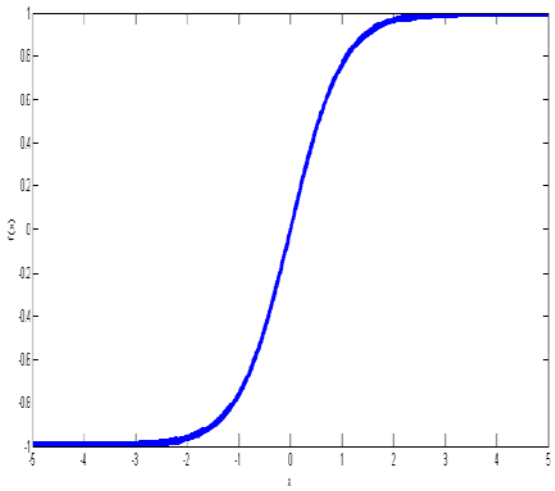


Figura 56. Hyperbolic tangent sigmoid Transfer Function (tansig).

***Symmetric hard-limit transfer function* (*hardlims*)**. Questa funzione è simile alla hardlim ma varia tra -1 e +1. La sua forma funzionale è la seguente:

$$f(x) = \begin{cases} 1 \; se \; x \geq 0 \\ 0 \; altrove \end{cases} \qquad (6.24)$$

Applicando i dati dell'esempio, si determina il grafico rappresentato in Figura 57.

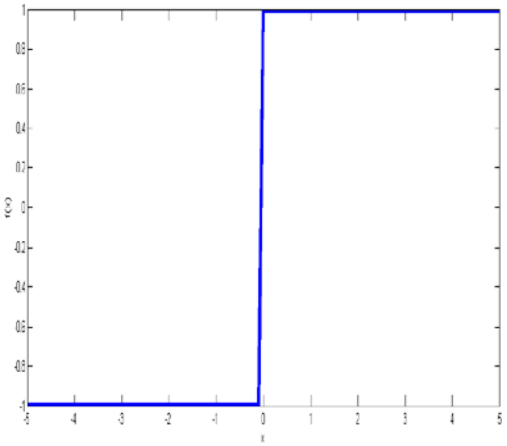


Figura 57. Symmetric hard-limit Transfer Function (hardlims).

***Saturating linear transfer function* (*satlin*)**. Questa funzione assume valori compresi tra 0 e 1 e ha la seguente forma funzionale:

$$f(x) = \begin{cases} 0 \; se \; x \leq 0 \\ x \; se \; 0 \leq x \leq 1 \\ 1 \; se \; x \geq 1 \end{cases} \tag{6.25}$$

La Figura 58 rappresenta la funzione *satlin* per l'esempio utilizzato.

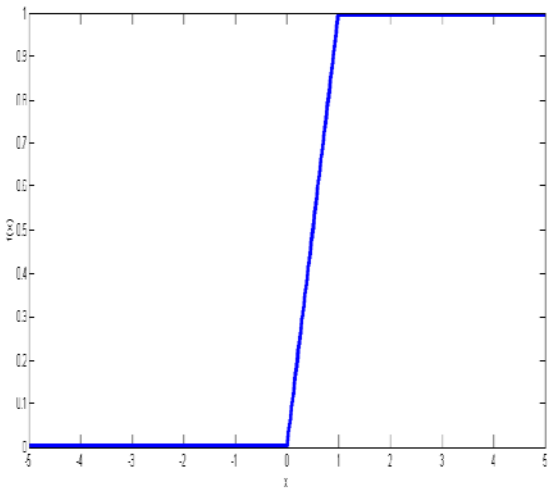


Figura 58. Saturating linear Transfer Function (satlin).

***Symmetric saturating linear transfer function* (*satlins*).** Questa funzione varia tra -1 e 1 e ha la seguente forma funzionale:

$$f(x) = \begin{cases} -1 \; se \; x \leq 0 \\ x \; se \; 0 \leq x \leq 1 \\ 1 \; se \; x \geq 1 \end{cases} \tag{6.26}$$

In Figura 59 vi è la rappresentazione della funzione per i valori di x tra -5 e +5.

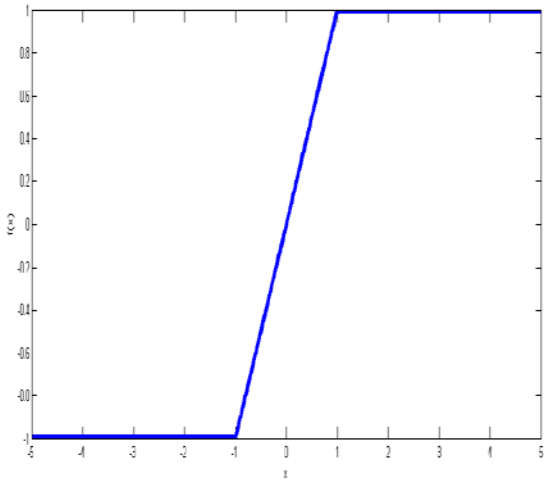


Figura 59. Symmetric saturating linear Transfer Function (satlins).

***Inverse transfer function* (*netinv*)**. Questa funzione calcola l'inverso di x, dunque la sua forma funzionale è semplicemente f (x) = 1/x. Con questa funzione incontriamo problemi quando x assume valori pari a 0 in quanto 1/0 tende a +∞. Supponiamo di avere un vettore che va da 1 a 10 con passo 0,1. La funzione inversa applicata a questo vettore avrà la forma di Figura 60.

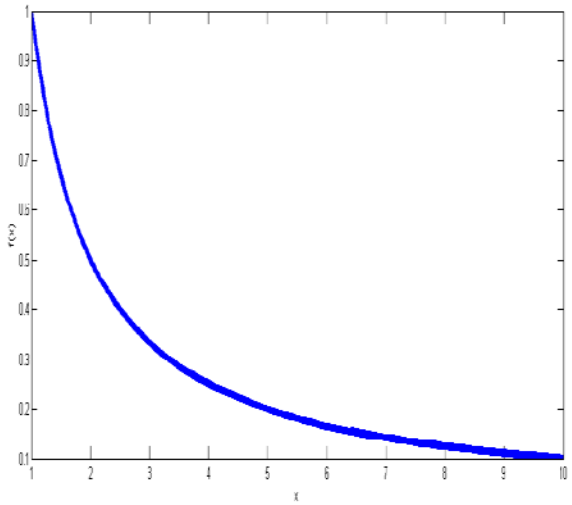


Figura 60. Inverse Transfer Function (netinv).

***Triangular basis transfer function* (*tribas*)**. Questa funzione assume valori compresi tra 0 e 1-|x| in base alla seguente funzione:

$$f(x) = \begin{cases} 1 - |x| se - 1 \leq x \leq +1 \\ 0 \; altrove \end{cases} \tag{6.27}$$

Riprendiamo il vettore che ha valori tra -5 e +5 con passo 0,1. La funzione *tribas* per questi dati è rappresentata nella Figura 61.

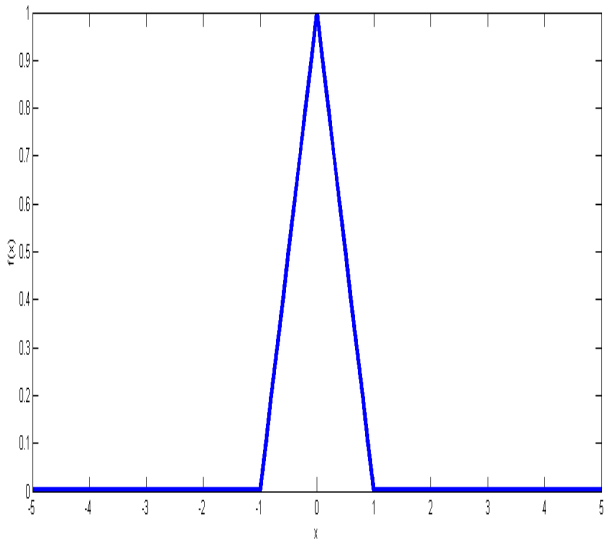


Figura 61. Triangular basis Transfer Function (tribas).

***Radial basis transfer function* (*radbas*)**. Anche questa funzione può assumere valori che variano tra 0 e 1 in base alla seguente forma funzionale:

$$f\ (x)\ =\ exp^{-x^x} \tag{6.28}$$

Per il vettore di valori tra -5 e +5 la funzione *radbas* è rappresentata in Figura 62.

***Competitive transfer function* (*compet*).** Questa funzione non ha una forma continua ma discreta, in base alla seguente forma:

$$f(x) = \begin{cases} 1\ se\ \max(x) \\ 0\ altrove \end{cases} \tag{6.29}$$

Supponendo di avere un vettore con valori: 0; 1; -0,5; +0,5 la f(x) assumerà valore pari a 1 in corrispondenza del secondo elemento e 0 altrove. La Figura 63 rappresenta questo esempio.

***Softmax transfer function* (*softmax*)**. Questa funzione ha seguente formulazione matematica:

$$f(x) = \frac{exp^2}{\sum exp^2} \tag{6.30}$$

Consideriamo di nuovo il vettore precedente formato da 4 valori. La Figura 64 rappresenta la funzione *softmax* per questi dati.

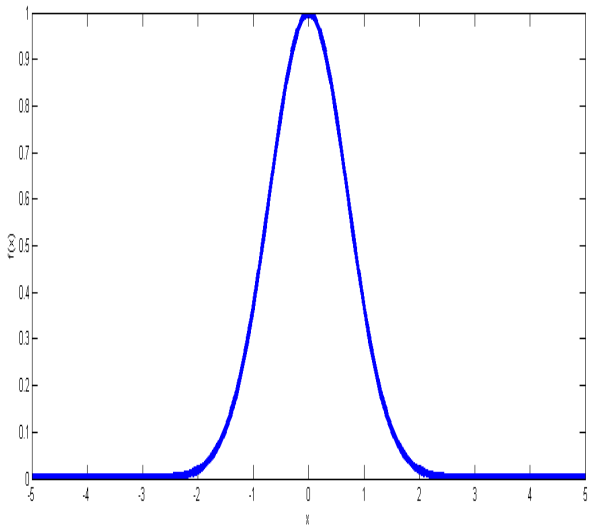


Figura 62. Radial basis Transfer Function (radbas).

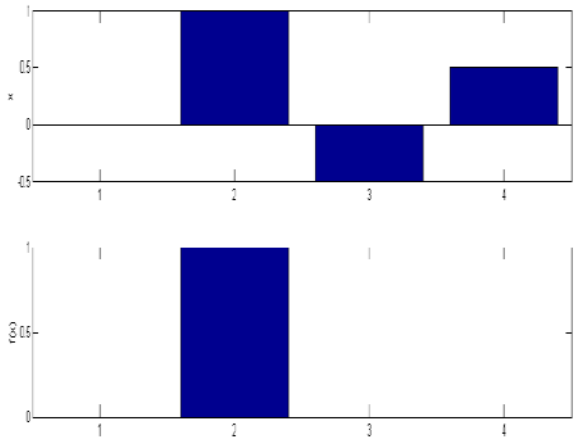


Figura 63. Competitive Transfer Function (compet).

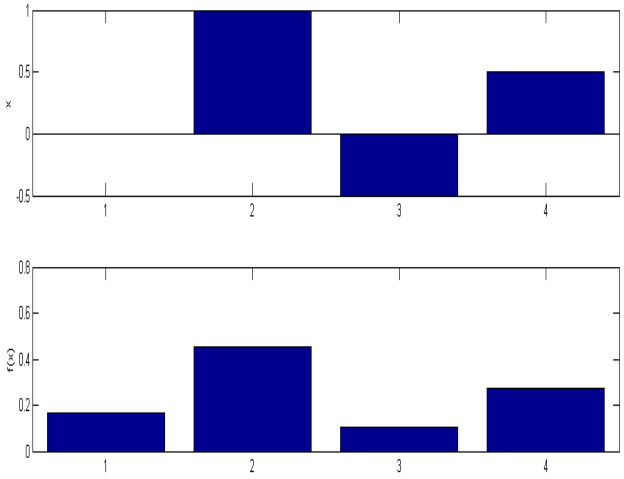
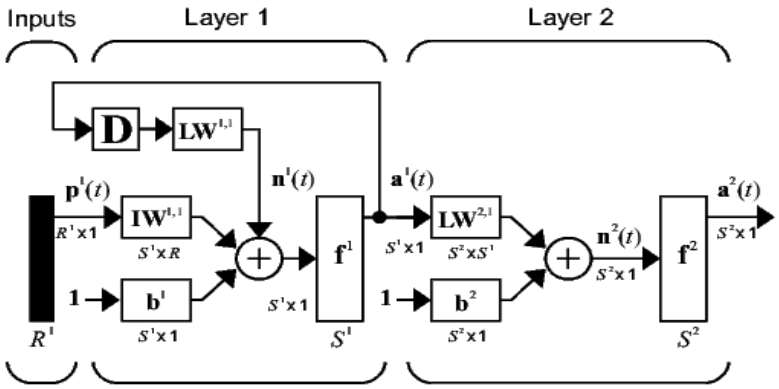


Figura 64. Softmax Transfer Function (softmax).

6.7. Topologie di RNA

6.7.1. *Dynamic Neural Network* (LRN)

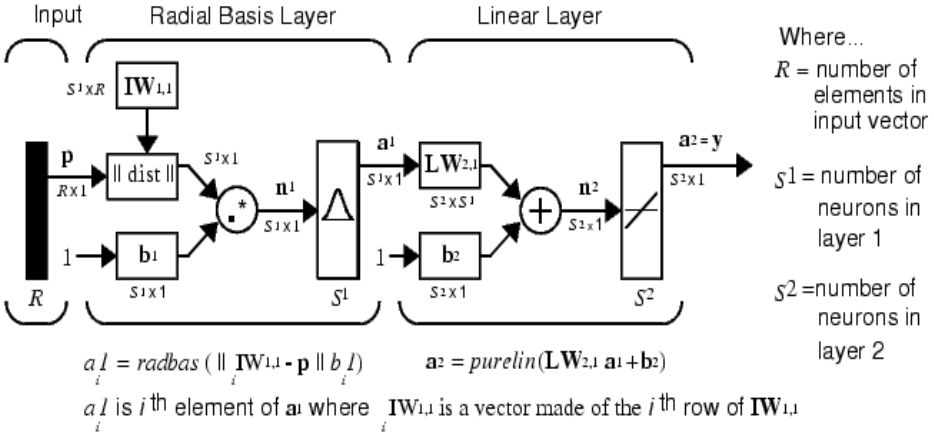
Questo tipo di rete è formato da due strati nascosti e vengono chiamate “dinamiche” perché tra il primo e il secondo strato vengono introdotte delle informazioni all’interno della topologia neurale che non erano state introdotte all’inizio (Figura 65).



Fonte: Beale et al. (2010).
Figura 65. Topologia di una rete dinamica.

6.7.2. *Radial Basis Function* (RBF)

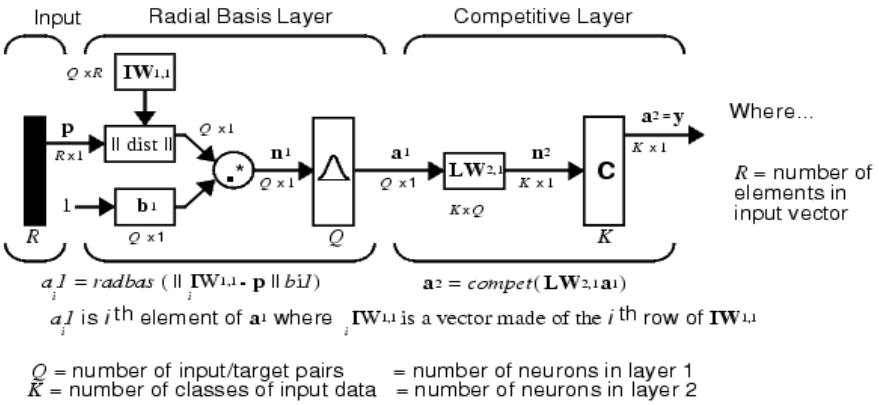
Questo tipo di rete neurale ha come caratteristica quella di avere nel primo strato nascosto una gaussiana e nel secondo presenta una funzione *purelin*, come mostrato in Figura 66. Viene impiegato per la previsione del tempo. Molto simili a queste reti sono le *Generalized Regression Neural Network* (GRNN) che nel secondo strato presentano invece della funzione *purline* un tipo particolare di funzione lineare.



Fonte: Beale et al. (2010).
Figura 66. Radial Basis Function.

6.7.3. *Probabilistic Neural Network* (PNN)

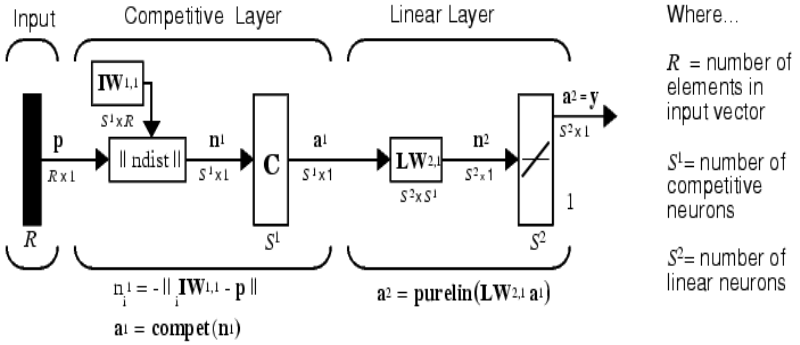
Questa topologia reticolare è una specificazione delle Radial Basis Function in quanto anche essa utilizza nel primo strato una gaussiana. Nell'ultimo strato invece utilizza una funzione *compet* che permette di avere in uscita delle probabilità (Figura 67). Viene spesso impiegata nei problemi di classificazione perché assegna ad ogni elemento la probabilità di appartenere ad un gruppo e per questo motivo è molto simile ai modelli ibridi "neuro-fuzzy".



Fonte: Beale et al. (2010).
Figura 67. Probabilistic Neural Network.

6.7.4. *Linear Vector Quantization* (LVQ)

Questa topologia di rete viene associata alle *SOFM* (vedi paragrafo 4.8.3 a p. 55) e presenta un primo strato competitivo e un secondo lineare, come mostrato in Figura 68.



Fonte: Beale et al. (2010).
Figura 68. Linear Vector Quantization.

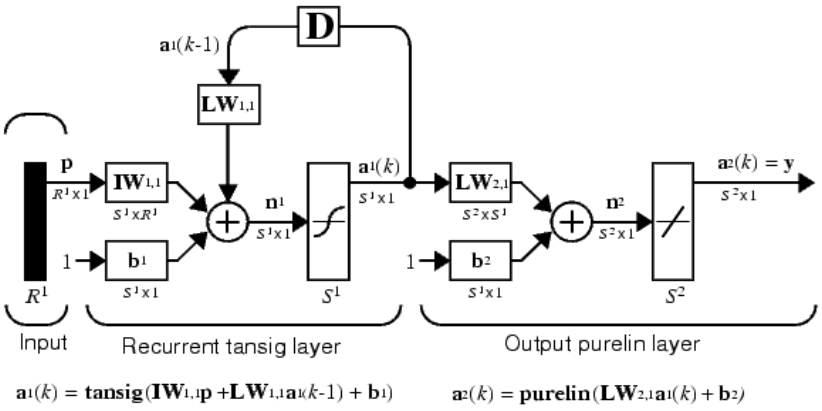
6.7.5. *Elmann Neural Network* (ELMNN)

Questa rete viene utilizzata per analizzare dati storici in quanto inserisce un ritardo all'in- terno della struttura. Difatti dal primo strato si ottiene un output che viene reinserito nella rete.

Per questo motivo questa tipologia di modelli viene anche chiamata rete "ricorrente" o "storica"[38].

La Figura 69 mostra la topologia di questo modello.

[38] Si ricorda che le reti neurali *feed-forward* hanno connessioni che permettono l'invio di informazione solo in un senso.



Fonte: Beale et al. (2010).
Figura 69. Elmann Neural Network.

7. BREVE GLOSSARIO

ACCURATEZZA: esprime la capacità di un modello di ottenere delle previsioni corrette;

AGGREGAZIONE DEI DATI: si tratta di una misura sintetica di più variabili. Per esempio il Prodotto Interno Lordo rappresenta una grandezza che è data dall'aggregazione di più variabili;

ALBERO DI DECISIONE: si tratta di un metodo di classificazione costituito da nodi e foglie;

ALGORITMI AGGLOMERATIVI: si tratta di metodi che permettono di raggruppare i dati in gruppi fino ad arrivare ad un solo *cluster*;

ALGORITMI GENETICI: modello di ottimizzazione basato su combinazioni, mutazioni e selezione naturale;

ALGORITMO APRIORI: si tratta di una tecnica per l'apprendimento di regole d'associazione;

ALGORITMO DI APPRENDIMENTO: si tratta di un metodo per cui il modello apprende su una parte del campione. Basandosi sui risultati, il modello adatta i pesi in modo da minimizzare l'errore. Due sono le tecniche di apprendimento: supervisionato, secondo il quale i pesi si modificano il base al valore di output e quello non supervisionato in base al quale il modello si auto-adatta ai dati;

ALGORITMO *K-MEANS*: si tratta di una tecnica che permette di suddividere i dati, o oggetti, in *k* partizioni sulla base delle loro caratteristiche. L'obiettivo di questo metodo è quello di minimizzare la variabilità tra i gruppi attraverso un procedimento iterativo;

ANALISI ESPLORATIVA DEI DATI: si tratta di utilizzare degli strumenti grafici per descrivere i dati in modo da acquisire il maggior numero di informazioni possibili da *dataset*;

ANALISI MULTIVARIATA: si tratta dell'analisi di *database* con più variabili;

ANALISI UNIVARIATA: si tratta dell'analisi di un *dataset* con una sola variabile;

ASSONE: rappresenta l'output del neurone di una rete neurale artificiale;

ATTRIBUTO: rappresenta una caratteristica qualitativa/quantitativa di un individuo;

AUC (*AREA UNDER the CURVE*): si usa per valutare la capacità di discriminazione di un modello e si riferisce alla curva ROC.

BACK-PROPAGATION: si tratta del più utilizzato algoritmo di apprendimento per la fase di *training* di una rete neurale artificiale;

BANCA DATI (o *DATABASE*): insieme di dati che può essere di dimensioni anche molto grandi e che descrive diverse realtà: sociali, economiche, naturali;

BINNING: consente di trasformare variabili continue in discrete utilizzando degli intervalli ricavati utilizzando un valore soglia (bin);

BOOTSTRAP: si tratta di una metodologia matematica di campionamento con reimmissione che permette di ottenere delle stime di parametri più robuste;

CENTROIDE: rappresenta il punto in cui la somma delle distanze tra tutti gli oggetti di quel *cluster* è minimizzata;

CLASSIFICAZIONE: si tratta di un'operazione che permette di assegnare gli elementi di un *dataset* in classi predefinite;

CLUSTERING: si tratta di una metodologia che permette di suddividere i dati in gruppi mutualmente esclusivi. All'interno dello stesso gruppo, i dati devono essere i più vicini possibili e il più distante possibile da quelli di gruppi diversi;

CROSS-VALIDATION: consiste in una tecnica che prevede che un *dataset* sia suddiviso in più parti. Ogni parte viene iterativamente utilizzata come *test* e le restanti come *training;*

DATA CLEANING: si tratta di una serie di operazioni che permettono di eliminare le ridondanze e le inconsistenze di un *dataset*;

DATA CUBE: (semplificando) consiste in una struttura per la memorizzazione di dati che permette di eseguire analisi in tempi rapidi. I dati sono studiati non solo su due dimensioni ma su tre e pertanto le analisi hanno una portata informativa superiore;

DATA MINING: rappresenta un insieme di tecniche che hanno l'obiettivo di estrarre di informazioni dai *database* allo scopo di individuare *patterns* e correlazioni utili;

DATASET: consiste nell'insieme di dati.

DATI ANOMALI: si tratta di dati molto diversi da quelli del *dataset*. Possono essere il risultato di un errore ma certamente vanno individuati ed esaminati per evitare che sporchino i risultati;

DENDROGRAMMA: si tratta di un grafico albero che illustra le relazioni tra diversi gruppi quando viene effettuata la *cluster analysis;*

DISCRETIZZAZIONE: consiste nella trasformazione di una variabile continua in una discreta (una tecnica è l'algoritmo di *binning*);

ENTROPIA: misura l'eterogeneità dei dati;

ERRORE QUADRATICO MEDIO: si tratta di un indice che misura la variabilità nei dati. Viene calcolato come la media del quadrato degli scarti intorno ad un'origine prestabilita;

ERRORE: consiste nella differenza tra il valore reale e il valore stimato;

FEED-FORWARD: quanto una rete neurale artificiale è *feed-forward* significa che i segnali viaggiano sempre da sinistra verso destra e non esistono dei segnali che tornano indietro;

FUNZIONE di ATTIVAZIONE (o TRASFERIMENTO): sono le funzioni che all'interno delle reti neurali permettono di estrarre e poi veicolare informazioni sotto forma di pesi matematici;

FUNZIONE *KERNEL*: calcola la stima della densità di probabilità di un campione.

GAUSSIAN MIXTURE MODELS: formano i *cluster* rappresentando la funzione di densità di probabilità delle variabili osservate come un mix di densità multivariate normali;

GIGABYTE: un miliardo di *bytes*.

HIERARCHICAL CLUSTERING: raggruppa i dati in base a diverse metriche per creare dei "*cluster tree*" o "dendrogram";

HTTP: *l'HyperText Transfer Protocol* (protocollo di trasferimento di un ipertesto) è usato come principale sistema per la trasmissione d'informazioni sul web.

ID3: è l'algoritmo per la generazione di un albero di decisione;

INDIRIZZO IP: è un'etichetta numerica che identifica univocamente un dispositivo collegato a una rete informatica che utilizza l'*Internet Protocol* come protocollo di comunicazione;

INFERENZA: è un insieme di analisi statistiche che viene condotto sul campione al fine di trarre delle deduzioni sulla popolazione intera;

ISTOGRAMMA: diagramma a blocchi usato per rappresentare una distribuzione di frequenze.

ITEM: denota una modalità di una variabile qualitativa; oppure una variabile appartenente ad un insieme di variabili;

ITEMSET: insieme di *item*;

KNOLEDGE DISCOVERY in *DATABASE* (KDD): è definito come il processo di scoperta di conoscenza.

LEMMATIZZAZIONE: è la riduzione di una forma flessa di una parola alla sua forma canonica (non marcata);

LINGUAGGIO DI INTERROGAZIONE MULTIMEDIALE: si tratta di un linguaggio, (Multi-Dimensional Query Language), che permette di specificare quali dati estrarre da una struttura di dati multidimensionale;

LOG-FILE: è un file di testo.

MAPPA DI KOHONEN: si veda la voce SOFM;

MARKET BASKET ANALYSIS: si tratta dell'analisi delle abitudini di acquisto dei clienti trovando associazioni su diversi prodotti comprati;

MATRICE DI CONFUSIONE: rappresenta gli esiti dell'applicazione di un classificatore binario;

MEDIA: la media è un indicatore di posizione;

MEDIANA: è rappresentata dal valore/modalità assunto dalle unità statistiche che si trovano nel mezzo della distribuzione di una variabile, dopo che gli stessi valori sono stati ordinati;

MISSING VALUES: valori mancanti;

MODA: si tratta del valore che compare più frequentemente in una distribuzione di frequenze;

MODELLO LINEARE: si tratta di un modello che assume relazioni lineari tra le variabili;

MODELLO PREDITTIVO: si tratta di un modello per la previsione di specifici output riferiti a una variabile nel *dataset*;

OUTLIER: consiste in un valore nel *dataset* anomalo;

OVERFITTING: si tratta di un problema legato a un modello che è stato addestrato su un campione nel quale ci sono molte variabili rispetto al numero di osservazioni. Il modello si superspecializza nel riconoscimento di quel *dataset* ma potrebbe sbagliare nella generalizzazione su ulteriori dati;

PAGERANK: consiste in un algoritmo capace di calcolare la probabilità di capitare nella pagina cercata;

PATTERN: indica una regolarità che si osserva nello spazio e/o nel tempo nel fare o generare delle cose;

PROBABILITÀ A POSTERIORI: consiste nel rapporto tra il numero di azioni favorevoli e il numero di azioni totali;

PROBABILITÀ A PRIORI: si tratta del rapporto tra il numero di azioni favorevoli è il numero totale di azioni;

REGOLE ASSOCIATIVE: si tratta di formule logiche che permettono di associare delle scelte o dei fatti ad altri fatti o scelte;

REGRESSIONE LINEARE: consiste in un metodo per la definizione della relazione lineare tra una variabile dipendente e uno o più variabili indipendenti;

RETI NEURALI ARTIFICIALI o *ARTIFICIAL NEURAL NETWORKS*: modelli predittivi non lineari che imparano attraverso il *training* e che richiamano come struttura le reti neurali biologiche;

ROC: *Receiver Operating Characteristic* è una curva che viene calcolata per valutare il risultato di un modello di classificazione;

SCORING: si tratta dell'assegnazione di un punteggio che indica la probabilità che si verifichi un evento;

SENSITIVITY: indica quanti valori positivi il modello è stato in grado di classifica- re correttamente su tutti i valori positivi che gli sono stati sottoposti da analizzare;

SIMILARITÀ: si tratta della distanza tra due oggetti o gruppi di oggetti in ambito di *cluster analysis*;

SINAPSI: si tratta dei pesi nei modelli di reti neurali artificiali;

SMOOTHING: lisciamento dei dati;

SOFM: le *Self-Organizing Feature Map* (SOFM): sono un tipo di rete neurale non supervisionato;

SOMA: è il corpo di varia forma della cellula nervosa che ne contiene il nucleo;

SPECIFICITY: indica quanti valori negativi il modello è stato in grado di classificare correttamente su tutti i valori negativi sottoposti ad analisi;

SUPERVISIONATO: si tratta di un modello di cui si conoscono i risultati;

SUPPORT VECTOR MACHINE (SVM): sono un insieme di metodi di apprendimento supervisionato per la regressione e la classificazione di *pattern*;

SUPPORTO: indica il numero di volte in cui una regola associativa appare in un insieme di dati;

TEXT MINING: si tratta di un insieme di metodi che sono utilizzati per l'analisi automatica dei dati testuali;

TRAINING SET: si tratta del campione che viene utilizzato per addestrare un modello;

VALIDATION SET: si tratta di un campione di dati sul quale vengono testati i parametri di un modello;

VARIANZA: si tratta di un indicatore della variabilità di una variabile;

WEB MINING: consiste nell'elaborazione, attraverso metodi di *Data Mining*, di dati provenienti dal web.

8. BIBLIOGRAFIA

Agrawal, R., & Shafer, J.C. (1996). Parallel mining of association rules. Knowledge and Data Engineering. *IEEE Transactions, 8*(6), pp. 962-969.

Agrawal, R., Imieliński, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. In *Proceedings of the 1993 ACM SIGMOD international conference on Management of data* (pp. 207-216).

Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., & Verkamo A.I. (1996). Fast Discovery of Association Rules. *Advances in knowledge discovery and data mining*, 12, pp. 307-328.

Altman, E. I. (1968). Financial ratios, discriminant analysis and the prediction of corporate bankruptcy. *The journal of finance*, *23*(4), pp. 589-609.

Altman, E.I. (2000). *Predicting financial distress of companies: revisiting the Z-score and ZETA models* (pp. 9-12). New York: Stern School of Business, New York University.

Anderson, E. (1936). The species problem in Iris. *Annals of the Missouri Botanical Garden*, *23*(3), pp. 457-509.

Banfield, J. D., & Raftery, A. E. (1993). Model-based Gaussian and non-Gaussian clustering. *Biometrics*, pp. 803-821.

Beale, M.H., Hagan, M.T., & Demuth, H.D. (2010). *Neural Network Toolbox^{TM} User's Guide*. Natick, MA: MathWorks.

Berger, J.O. (1985). *Statistical decision theory and Bayesian analysis*. New York: Springer.

Bergman, S. (1970). *The kernel function and conformal mapping* (Vol. 5). Providence, Rhode Island: American Mathematical Society.

Berkhin, P. (2006). A survey of clustering data mining techniques. In *Grouping multidimensional data* (pp. 25-71). Berlin, Heidelberg: Springer.

Bernardo, J. M., & Smith, A. F. (2009). *Bayesian theory* (Vol. 405). Chichester, England: John Wiley & Sons.

Berry, M.W., & Browne, M. (2006). *Lecture notes in data mining*, Singapore: World Scientific.

Breiman, L. (1996). Bagging predictors. *Machine learning*, *4*(2), pp. 123-140.

Brin, S., Motwani, R., Ullman, J.D., & Tsur, S. (1997). Dynamic itemset counting and implication rules for market basket data. In *Proceedings of the 1997 ACM SIGMOD international conference on Management of data* (pp. 255-264).

Cheung, D.W., Han, J., Ng, V.T., & Wong, C.Y. (1996). Maintenance of discovered association rules in large databases: An incremental updating technique. In *Proceedings of the twelfth international conference on data engineering* (106-114).

Conn, A., Gould, N., & Toint, P. (1997). A globally convergent Lagrangian barrier algorithm for optimization with general inequality constraints and simple bounds. *Mathematics of Computation*, 66(217), pp. 261-288.

Dulli, S., Furini, S., & Peron, E. (2009). *Data mining: metodi e strategie*. Milano: Springer-Verlag Italia.

Danielsson, P.E. (1980). Euclidean distance mapping. *Computer Graphics and image processing*, 14(3), pp. 227-248.

Fayyad, U., Piatetsky-Shapiro, G., & Smith, P. (1996). From data mining to knowledge discovery: An overview. In *Advances in Knowledge Discovery and Data Mining* (pp. 1-34). Menlo Park, CA: MIT Press.

Fisher, R.A. (1936). The use of multiple measurements in taxonomic problems. *Annals of eugenics*, *7*(2), pp. 179-188.

Fisher, R. A. (1992). Statistical methods for research workers. In *Breakthroughs in statistics* (pp. 66-70). New York, NY: Springer.

Freund, Y., Schapire, R., & Abe, N. (1999). A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, *14*(5), pp. 771-780.

Fukunaga, K., & Hummels, D.M. (1989). Leave-one-out procedures for nonparametric error estimates, Pattern Analysis and Machine Intelligence. *IEEE Transactions*, *11*(4), pp. 421-423.

Gan, G., Ma, C., & Wu, J. (2007). *Data clustering: theory, algorithms, and applications*. Philadelphia, PA-Alexandria, VA: American Statistical Association-Society for Industrial and Applied Mathematics.

Goldberg, D.E., & Holland, J.H., (1988). Genetic algorithms and machine learning, *Machine learning, 3*(2), pp. 95-99.

Greene, W.H. (2000). *Econometric analysis 4th edition. International edition*. (pp. 201-215), New Jersey: Prentice Hall.

Han, J., Kamber, M., & Pei, J. (2011). Data mining concepts and techniques third edition. *The Morgan Kaufmann Series in Data Management Systems*, *5*(4), pp. 83-124.

Wong, M. A., & Hartigan, J. A. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics), 28*(1), pp. 100-108.

Haykin, S.S. (1994). *Neural networks: a comprehensive foundation*, New York: Prentice Hall.

Haykin, S.S. (2009). *Neural networks and learning machines*, New York: Prentice Hall.

Holland, J.H. (1975). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*, University of Michigan Press.

Ichino, M., & Yaguchi, H. (1994). Generalized Minkowski metrics for mixed feature-type data analysis. *IEEE Transactions on Systems, Man, and Cybernetics*, *24*(4), pp. 698-708.

Jaccard, P. (1901). Étude comparative de la distribution florale dans une portion des Alpes et des Jura, *Bulletin de la Société Vaudoise des Sciences Naturelles*, 37, pp. 547-579.

Johnson, S.C. (1967). Hierarchical clustering schemes, *Psychometrika, 32*(3), pp. 241-254.

Kantardzic, M. (2011). *Data mining: concepts, models, methods, and algorithms*. Hoboken, NJ: John Wiley & Sons.

Kohonen, T. (1982). Self-organized formation of topologically correct feature maps, *Biological cybernetics, 43*(1), pp. 59-69.

Kohonen, T. (2001). *Self-organizing maps*. Berlin, Heidelberg, New-York: Springer Science & Business Media.

Li, W. (1992). Random texts exhibit Zipf's-law-like word frequency distribution. *IEEE Transactions on information theory*, *38*(6), pp. 1842-1845.

Lin, D. I., & Kedem, Z. M. (1998, March). Pincer-search: A new algorithm for discovering the maximum frequent set. In *International conference on Extending database technology* (pp. 103-119). Berlin, Heidelberg: Springer.

Meo, R. (1999, August). A new approach for the discovery of frequent itemsets. In *International Conference on Data Warehousing and Knowledge Discovery* (pp. 193-202). Berlin, Heidelberg: Springer.

Orsi R. (1995). *Probabilità e inferenza statistica*, Bologna: Il mulino.

Park, J. S., Chen, M. S., & Yu, P. S. (1995). An effective hash-based algorithm for mining association rules. *Acm sigmod record*, *24*(2), pp. 175-186.

Parker, J. R. (2001). Rank and response combination from confusion matrix data. *Information fusion*, *2*(2), pp. 113-120.

Penny, K. I. (1996). Appropriate critical values when testing for a single multivariate outlier by using the Mahalanobis distance. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *45*(1), pp. 73-81.

Rajola, F. (2003). *Customer relationship management: Organizational and technological perspectives*. Berlin, Heidelberg: Springer.

Savasere, A., Omiecinski, E.R, & Navathe, S.B. (1995). An efficient algorithm for mining association rules in large databases. *The International Journal on Very Large Data Bases*, pp. 432-444.

Toivonen, H. (1996). Sampling large databases for association rules, *The International Journal on Very Large Data Bases*, 96, pp. 134-145.

Tufféry, S. (2011). *Data mining and statistics for decision making*. Chichester, West Sussex, UK: John Wiley & Sons.

Vapnik, V. (1998). The support vector method of function estimation. In *Nonlinear modelling.* (pp. 55-85), Boston, MA: Springer.

Yonelinas, A.P. (1994). Receiver-operating characteristics in recognition memory: evidence for a dual-process model. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *20*(6), pp. 1341-1354.

Zipf, G. K. (1999). *The psycho-biology of language: An introduction to dynamic philology*. London: Routledge.

Itinerari per l’alta formazione

Itinerari per l’alta formazione è una collana dei Volumi IRCrES per la didattica universitaria e terziaria. Gli Itinerari mettono rapidamente a disposizione degli studenti, della comunità scientifica e di un vasto pubblico testi completamente open access, finalizzati alla formazione.

- N. 1. G.G. Calabrese. (2021). *Elementi di organizzazione aziendale*. Moncalieri TO: CNR-IRCRES. http://dx.doi.org/10.23760/978-88-98193-2020-02

Abstract

The aim of this publication is to expose students to use basic tools for the analysis of big amount of data.
The first section starts presenting the definition of Data Mining and Knowledge Discovery in Database explaining the more common techniques and listing the main operational applications.
A second paragraph illustrates the first three phases preceding the application of Data Mining techniques: Selection/Sampling, Pre-processing/Cleaning and Transformation/Reduction of data. These prelaminar data analysis techniques are essential as the results of the Data Mining models depend on the correctness of the data.
The third paragraph presents some applications of methodologies. In this section, the technical aspect has less relevance than the operational one with the aim to explain the use of these techniques. However, the more common Data Mining models are listed and explained.
The fourth paragraph is addressed to the Text Mining and Web Mining, which are two methodologies used to analyze texts and websites. This section presents the main problems related to textual analysis and the techniques that can be used to obtain effective searches.
Finally, two appendices have been added: the Statistical Appendix reports some technical insights that may be useful for understanding the Data Mining systems; in a second appendix, a Short Glossary containing the main terms related to Data Mining used in the text is proposed.

ISBN 978-88-98193-23-3