# Data Mining - Il processo KDD

Federico Calò

# Contents

# 1 KDD process

L'automazione delle attività economiche produce un incremento dello stream di dati perchè anche singole transazioni (una chiamata telefonica, il credito di una carta, un test medico) sono tipicamente registrate in un computer. Le basi di dati scientifiche e governative sono anche in rapida crescita. C'è un divario crescente tra la generazione di dati e la loro comprensione. Risulta quindi necessario l'utilizzo di computer per analizzare i dati, ma questo non è sufficiente.

Necessitiamo di una metodologia matura che spieghi come grandi strutture di dati possono essere analizzate. Questa metodologia è stata studiata in un'area di ricerca conosciuta come KDD. Lo scopo è investigare come tecniche di Machine Learning possono essere applicate a estratti di "conoscenza" di una grande massa di dati disponibili. All'inizio vi era una certa confusione sull'area di interesse ricoperta dal Machine learning, Data Mining e Knowledge Discovery. Ora, si è giunti alla conclusione che il KDD denota l'intero processo di estrazione della conoscenza, dalla raccolta dei dati e dalla preelaborazione all'interpretazione dei risultati.

Il Data Mining è lo step, all'interno del processo KDD, nel quale le informazioni sono estratte dai dati applicando ad essi opportuni algoritmi. Questi algoritmi sono il più delle volte quelli di Machine Learning. In un contesto aziendale il termine Data Mining è ancora utilizzato per denotare il processo di knowledge discoverym e questo causa qualche confusione. Sempre sulla terminologia: KDD è ancora utilizzato anche se la conoscenza non è rigorosamente estratta dai "database".

Il tentativo più importante e completo di definire il Knowledge Discovery come un processo afferma: "La scoperta della conoscenza è l'estrazione non banale di informazioni implicite, precedentemente sconosciute e potenzialmente utili dai dati."

- **Non banale**: si intende che nel processo è coinvolta qualche ricerca o inferenza statistica, quindi non è un semplice calcolo di quantità predefinite;
- **Implicita**: ci si riferisce al fatto che l'informazione è implicita nel dato e non formalmente esplicita, l'informazione esplicita è estratta attraverso altre tecniche;
- **Sconosciuta**: l'informazione deve essere nuova, la novità dipende dal quadro di riferimento assunto;
- **Utile**: l'informazione deve essere utile a raggiungere lo scopo del sistema o dell'utente. I pattern completamente estranei agli obiettivi dati sono di scarsa utilità e non costituiscono conoscenza all'interno della situazione data.

Diamo una definizione formale al KDD:

**Dati**:

- un insieme di fatti (data) F,
- una rappresentazione in un linguaggio L,
- una certa misura di certezza C,

possiamo definire un pattern come una dichiarazione S in L che descrive le relazioni tra un sotto insieme $F_S$ di F con una certezza $\boldsymbol{c}$, tale che S è più semplice (in un certo senso) dell'enumerazione di tutti i fatti in $F_S$.

Un pattern è considerato conoscenza se è interessante e abbastanza certo (o valido). Nel KDD siamo interessati in pattern che sono espressi in un linguaggio di alto livello, come:

Se Età $< 25$ e Corso-di-Educazione = no

Allora Incidente = si

Con probabilità = 0.2 a 0.3

In questo modo alcuni pattern possono essere capiti e usati direttamente dalle persone o possono essere input ad altri programmatori. Definiamo ora il termine **certezza**, senza un livello sufficiente di certezza i modelli diventano ingiustificati e non riescono a diventare conoscenza. La certezza coinvolge diversi fattori, quali:

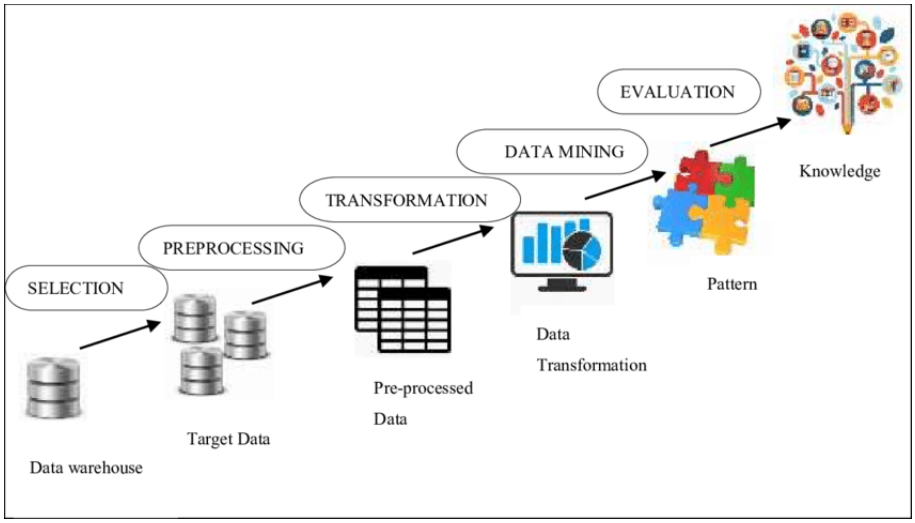
- Integrità dei dati,
- Dimensione del campione su cui è stata effettuata la scoperta
- Il grado di supporto dalla conoscenza del dominio disponibile

Un pattern è definito **interessante** quando è:

- Nuovo
- Utile
- Non banale da calcolare

Vi sono principalmente due differenti interpretazioni dei pattern e dei modelli. La **prima** interpretazione che possiamo dare consiste nel definire un modello come una sintesi globale del data-set, mentre il pattern è una caratteristica locale del data-set, limitato a un sub-set di osservazioni e/o attributi. La **seconda interpretazione** esplica come il data mining implica l'adattamento o la determinazione di pattern da dati osservati. In questo caso il pattern è visto come una istanza del modello. I modelli adattati svolgono il ruolo di conoscenza dedotta

Il kdd è un processo iterativo e interattivo, costituito da molti passaggi che includono molte decisione prese dall'utente.




In una prima fase si sviluppa una **comprensione del dominio** dell'applicazione, delle relative conoscenze di base e degli obiettivi dell'utente finale. Si prosegue creando **un data set target**, quindi si seleziona un dataset o ci si focalizza su un sotto

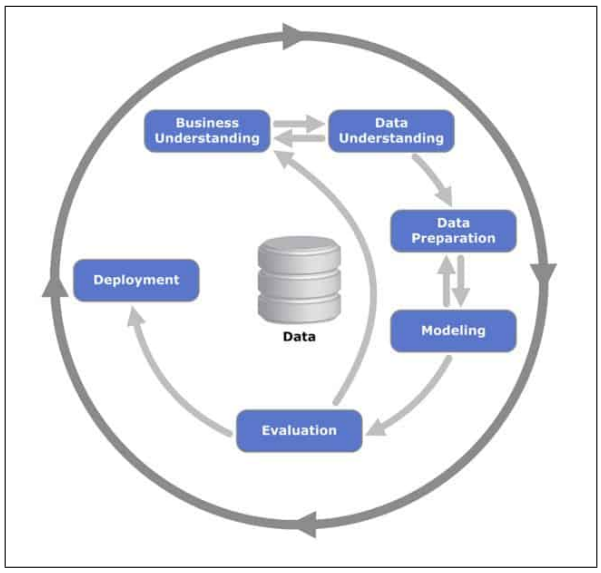
insieme di variabili o di esempi di dati sui quali deve essere eseguita la scoperta. Una volta selezionati questi dati si passa alla fase di **pulizia dei dati e preprocessing** nella quale si effettuano svariate operazioni volte a rimuovere il rumore o dei valori anomali, collezionando le informazioni necessarie per modellare o tenere conto del rumore, decidere strategie per gestire i campi di dati mancanti o tenere conto delle informazioni sulla sequenza temporale e delle modifiche note. Successivamente si entra in una fase di **riduzione e proiezione dei dati**, nella quale si trovano funzioni utili per rappresentare i dati a seconda dell'obiettivo dell'attività. Vengono utilizzati metodi di riduzione o trasformazione multidimensionale per ridurre l'effettivo numero di variabili da considerare o per trovare rappresentazioni di dati invarianti. Conclusa questa attività, si passa alla **selezione del compito del data mining**, decidendo qual è l'obiettivo del processo KDD, scegliendo tra classificazione, regressione, clustering.

Definito l'obiettivo si passa alla **scelta degli algoritmi** di data mining. Vengono selezionati i metodi che verranno utilizzati per ricercare i pattern frequenti all'interno dei dati. Vengono definiti i modelli e i parametri più appropriati e si cerca di far corrispondere particolari metodi di data mining con i criteri globali dei processi di KDD. Quindi si entra nella fase di **Data Mining**, nella quale si ricercano modelli di interesse in una particolare forma rappresentativa. L'utente può contribuire significativamente in questa fase eseguendo correttamente i passaggi precedenti. Alla fine del processo di data mining, vi è la fase di **interpretazione dei pattern minati**, nella quale si effettuano le valutazioni sui risultati ottenuti ed eventualmente si ritorna ad iterare su uno degli step precedenti. Infine si **consolida la conoscenza scoperta**, la conoscenza ottenuta viene incorporata nel sistema per migliorarne le prestazioni o semplicemente si documenta per essere segnalata ad altre parti di interesse. Questa fase include anche il controllo e la risoluzione di particolar conflitti con conoscenze precedentemente estratte o scoperte.

## 1.1 Steps del KDD

La maggior parte del lavoro nel KDD è focalizzata sullo step del data mining. Anche se gli altri step sono considerati importanti per il successo dell'applicazione del KDD nella pratica. La necessità per una standardizzazione del processo della scoperta della conoscenza ha portato alla definizione dello standard industriale CRISP-DM. L'obiettivo di questo standard è quello di sviluppare un processo neutrale per condurre il KD, e di definire i compiti, i loro outputs, la terminologia e i problemi tipici di caratterizzazione.

Il modello di processo CRISP-DM è per la scoperta i conoscenza consiste in sei fasi.




La sequenza delle fasi non è rigida. È possibile spostarsi avanti e indietro tra le diverse fasi a seconda dell'esito di ciascuna fase. Le frecce indicano le dipendenze più importanti/frequenti. Il cerchio esterno simboleggia la natura ciclica di un processo KDD che può continuare dopo l'implementazione di una soluzione. Ogni fase contiene un numero di task che produce specifici output.

### 1.1.1 Business Understanding

La **prima fase** di questo modello viene definita **Business Understanding** (comprensione degli affari), il cui primo step è quello di **determinare gli obiettivi commerciali**, step ha molti aspetti in comune con la fase iniziale di quasiasi impegno progettuale significativo. I requisiti minimi sono:

- un problema o un'opportunità commerciale percepita
- un certo livello di sponsorizzazione esecutiva

Sviluppare una definizione chiara e comprensibile dei bisogni aziendali non è un compito semplice. E' richiesta la collaborazione dei business analyst e dei data analyst. Questo passaggio del processo è anche il momento in cui iniziare a definire le aspettative. Alla fine di questo step vengono prodotti:

- **Il background**, che descrive le informazioni note sulla situazione aziendale all'inizio del processo
- **Gli obiettivi di business**, che descrivono i principali obiettivi dalle prospettive di business
- **I criteri economici di successo**, che definiscono le misure per risultati di alta qualità del progetto dal punto di vista del business

Dopo aver definito gli obiettivi commerciali, si passa alla **valutazione della situazione**, con lo scopo di raccogliere informazioni sulle risorse, sui vincoli e sulle assunzioni. Alla fine di questo sotto processo verrà prodotto un inventario contenente:

- le risorse: personale, dati, calcoli

- vincoli: schede di compilazione, questioni legali, comprensibilità
- assunzioni: disponibilità dei dati

Inoltre non dimentichiamo che vengono prodotti anche **un glossario di termini**, che copre la terminologia di business e di data mining, e **un'analisi costo-beneficio**, cioè un documento contenente le spese del progetto dovrebbero essere confrontate con i potenziali guadagni.

Successivamente si passa alla **determinazione degli obiettivi del Data Mining**. In questa fase si trasformano gli obiettivi commerciali in obietti del processo di Data Mining e si costruiscono i relativi criteri di successo. Possiamo classificare gli obiettivi del Data Mining in:

- Classification
- Estimation (produrre una stima, Estimation)
- Prediction
- Affinity grouping
- Clustering
- Description e Profiling

I due obiettivi primari del Data Mining tendono ad essere:

- **Predizione**, che include l'uso di alcune variabili indipendenti per predire valori sconosciuti o futuri che dipendono da altre variabili
- **Descrizione**, nella quale non si fa una distinzione tra variabili dipendenti e indipendenti e si concentra sulla ricerca di modelli interpretabili dall'uomo che descrivono i dati.

L'obiettivo della classificazione è apprendere una funzione che mappa, o classifica, un dato in una predefinita classe. Questa tecnica è molto utilizzata per i database. Invece quando si parla di regressione, si ha come obiettivo apprendere una funzione che mappa un dato in una variabile di previsione reale. Quando invece si cerca di creare un procedimento per trovare le associazioni tra gruppi di variabili, si utilizza il metodo Affinity Grouping. Il clustering è un'attività descrittiva in cui si cerca di identificare un insieme finito di categorie o cluster per descrivere i dati. Le categorie possono essere mutuamente esclusive ed esaustive, oppure consistere in una rappresentazione più ricca come una gerarchia o categorie sovrapposte. La clasterizzazione viene utilizzata spesso per scoprire sotto-popolazioni omogenee, identificare sotto categorie, o analisi di dati. Strettamente correlato al clustering è il compito della stima della densità di probabilità, che consiste in tecniche per stimare dai dati la funzione di densità di probabilità multivariata congiunta di tutte le variabili/campi nel database. Per **Summarization** (riassunto o descrizione o profilazione), si intendono tutti quei metodi per trovare una descrizione compatta per un sottoinsieme di dati. Invece il task **Dependency modeling** (modellare le dipendenze), consiste nel trovare un modello che descrive dipendenze significative tra variabili. Esistono due tipi di modelli dipendenti:

- il livello strutturale di specifici modelli le cui variabili sono localmente dipendenti tra loro
- il livello quantitativo il cui modello specifica quanto le variabili sono dipendenti usando una scala numerica.

Il task di rilevamento di modifiche e deviazioni si focalizza sulla scoperta delle modifiche più significanti nei dati rispetto a valori misurati o normativi in precedenza. Consiste nel trovare un modello che descrive significanti dipendenze tra variabili.

Dopo aver definito il task del Data Mining si passa a **produrre un piano progettuale** per il raggiungimento egli obiettivi di data mining e quindi il raggiungimento degli obiettivi di business. L'output di questa fase è ovviante un piano progettuale, che specifica l'insieme degli step per la restante parte del progetto, la sua durata, le risorse richieste, gli input, gli output e le dipendenze.

### 1.1.2 Data Understanding

Dopo aver contestualizzato il business in cui si andrà a progettare il sistema, si passa a una fase di **Data Understanding**, cioè di compresone dei dati, che inizia con una fase di raccolta iniziale dei dati, durante la si accede ai data rilevanti nell'inventario delle risorse e si produce un report iniziale dei dati raccolti, nel quale si elenca la posizione dei dati, i metodi usati per acquisirli e i problemi incontrati. Successivamente si passa a descrivere i dati esaminando le loro proprietà e creando un report nel quale si descrive il formato dei dati, i potenziali valori, la quantità, l'identificatore dei campi e tutte le caratteristiche che vengono scoperte.

Ci sono due metodi principali per descrivere le variabili:

- **in maniera categorica**: i possibili valori finiti e i differenti tipi che una variabile può assumere. Questa categoria si può suddividere in:
  - **variabili nominali** che denominano il tipo di oggetto a cui si riferiscono, ma non esiste un ordine tra i valori possibili. (stato del materiale, genere, livello di educazione)
  - **variabili ordinali** che assumono un ordine tra i possibili valori. (valutazione del cliente)
- **quantitativi**: sui quali sono consentite operazioni aritmetiche, e si suddividono a loro volta in:
  - **variabili discrete**, i cui valori sono interi
  - **variabili continue**, i cui valori sono numeri reali.

Dopo aver descritto la tipologia di dati si passa alla verifica della loro qualità ispezionando e affrontando diverse caratteristiche quali:

- **Accuratezza**: conformità del valore memorizzato rispetto a quello effettivo
- **Completezza**: nessun valore mancante
- **Consistenza**: rappresentazione uniforme
- **Attualità**: i dati storicizzati non sono obsoleti.

Scarsa qualità dei dati e scarsa integrità dei dati sono i maggiori problemi all'interno dei progetti KDD.Si vedrà come la maggior parte dei dati operativi non è mai stata acquisita o modellata per scopi di data mining. I dati selezionati vengono generalmente raccolti da numerosi sistemi operativi, incoerenti e scarsamente documentati. È importante comprendere la **sensibilità temporale** dei dati. Lo specialista della gestione dei dati è responsabile della raccolta e dell'integrazione dei dati nell'ambiente informativo.

Al termine della verifica della qualità dei dati viene gerato un report di qualità dei dati che riporta i risultati della verifica e se vi sono dei problemi, sarà possibile discutere di eventuali soluzioni.

A valle della verifica della qualità dei dati, è possibile avviare la fase di **esplorazione dei dati**, alla fine della quale seguirà un relativo report dei risultati ottenuti. Per le *variabili categoriche*, le distribuzioni della frequenza dei dati sono il metodo migliore per capire il contenuto dei dati. Istogrammi e grafici a torta aiutano a identificare gli schemi della distribuzione e i valori mancanti o non validi. Mentre quando lavoriamo con *variabili quantitative*, l'analista dei dati è interessato a misure come il massimo e il minimo, la media, moda, mediana e misure statistiche. Se combinate, queste misure offrono un modo efficace per determinare la presenza di dati non validi e distorti.

### 1.1.3 Data Preparation

Lo step successivo al Data understang vi è lo step di **Data Preparation**, la cui prima fase consiste nel selezionare i dati, fase in cui il problema maggiore consiste nel selezionare dati da tuple di database relazionali. Si possono comunque seguire alcuni principi che includono:

- la rilevanza del dato rispetto all'obiettivo principale.
- vincoli tecnici e qualitativi,
- limiti al volume dei dati o ai tipi di dati

In questa fase si produce un report di inclusione/esclusione dei dati. La selezione dei dati può essere eseguita manualmente o automaticamente (campionamento e selezione delle caratteristiche).

Il più semplice tipo di campionamento è il **campionamento semplice casuale**: ogni gruppo di oggetti della dimensione richiesta ha la stessa probabilità di essere il campione selezionato. È possibile ottenere un campione molto atipico, tuttavia le leggi della probabilità impongono che più ampio è un campione, più è probabile che sia rappresentativo della popolazione da cui proviene. Il metodo tradizionale di scelta di un campione casuale inizia con una numerazione dei membri della popolazione target. L'ordine di numerazione è irrilevante. Una volta che ogni membro della popolazione ha un numero, il campionatore consulta una tabella di numeri casuali per selezionare gli indici dei membri da includere nel campione. Ovviamente il presupposto principale è quello di essere in grado di numerare i membri della popolazione target.

Vi sono due tipi di campionamento semplice casuale:

- con sostituzione
- senza sostituzione

Se la popolazione è grande rispetto al campione, c'è una probabilità molto piccola che qualsiasi membro venga scelto più di una volta e le due tecniche sono essenzialmente le stesse.

Quando la popolazione è divisa in strati o gruppi, è utile invece usare un **campionamento casuale stratificato**. Viene selezionato un campione casule semplice da ciascuno strato separatamente e la loro unione produce un campione stratificato. Un vantaggio di questo campionamento consiste nel fatto che l'analista può controllare il numero di osservazioni all'interno di ogni gruppo o strato e può garantire che particolari gruppi all'interno della popolazione sono adeguatamente rappresentati

nel campione. Quando uno strato ha un'appartenenza molto più piccola degli altri, il semplice campionamento casuale può produrre campioni senza rappresentativi di quello strato. La dimensione del campione è solitamente proporzionale alla dimensione relativa degli strati. Tuttavia, questa non è una regola.

Se i membri all'interno degli strati sono più simili tra loro rispetto ai membri di strati diversi, le stime specifiche per strato saranno più precise di quelle dell'intero campione. Attenzione però, è importante adeguarsi alla sovra rappresentazione quando le inferenze si riferiscono alla popolazione target nel suo insieme.

Alcune regole per una buona rappresentazione di progettazione per un campionamento stratificato, sono:

- gli strati devono essere scelti per:
  - avere dei mezzi che differiscono sostanzialmente tra loro
  - minimizzare la varianza all'interno di uno strato e massimizzarla tra i vari strati
- Le dimensioni del campione devono essere proporzionali alla deviazione standard dello strato.

Un'altra tecnica di campionamento è il **cluster sampling** o campionamento clusterizzato, nel quale i membri della popolazione arrivano naturalmente da cluster, ciò rende possibile campionare i cluster. In questo caso tutti i membri di ogni cluster sono considerati. Nel contesto di grandi basi di dati, un'applicazione comune del cluster sampling è di rendere casuale la scelta di blocchi di dati, e successivamente usare tutti i dati nei blocchi. La motivazione dietro questo approccio è che per recuperare un record di database da un blocco particolare, l'intero blocco deve essere letto in memoria. Questo tipo di campionamento è anche chiamato **block scampling**. I vantaggi del cluster sampling è che riduce i costi richiesti per accedere ai campioni, ma al tempo stesso aumenta la variabilità delle stime campionarie al di sopra di quella del semplice campionamento casuale, a econda di quanto i cluster differiscono tra loro, rispetto alla variazione all'interno del cluster.

Se i membri di un cluster sono più simili dei membri di cluster diversi, gli approcci statistici che presuppongono che i dati siano indipendenti porteranno a inferenze distorte. La modellazione gerarchica può modellare esplicitamente la struttura indotta dall'amplificazione nei dati.

Il **two-stage sampling** (campionamento a due stadi) combina due idee principali: la celta casuale dei cluster e il campionamento all'interno di ogni cluster.

Quando è possibile numerare in qualsiasi modo gli individui di una popolazione, si può effettuare il **systematic sampling** (campionamento sistematico), conosciuto anche come **every k-th sampling**. Questo tipo di campionamento si sviluppa scegliendo un membro in maniera casuale da quelli numerati tra 1 e k, successivamente include ogni k-th membro dopo il campione. Il vantaggio maggiore di questo tipo di campionamento è la sua facile implementazione,anche nel caso in cui la dimensione della popolazione è inizialmente sconosciuta o il conteggio dei membri della popolazione è computazionalmente espansivo. Al contempo bisogna prestare attenzione, dato che la selezione non è casuale, campioni sistematici possono non essere rappresentativi della popolazione devono essere utilizzati con attenzione. È particolarmente vulnerabile alle periodicità nell'elenco dei membri della popolazione. Se la periodicità è presente e il periodo è un multiplo di k, risulterà una distorsione.

Quando vogliamo organizzare il nostro campionamento basato sul valore di una o pù variabili, ma non conosciamo l'intervallo o la distribuzione di queste variabili nella popolazione target, possiamo avvalerci del **two-phase sampling** o campionamento in due fasi. Un campione iniziale può facilitare prendere decisioni più consapevoli sulle strategie di campionamento da utilizzare. Un campione iniziale può aiutare a determinare la dimensione del campione. I calcoli delle dimensioni del campione spesso richiedono stime di determinati parametri della popolazione, come la forza della relazione tra due variabili. In assenza di

conoscenze e/o dati pregressi, un campione iniziale fornirebbe stime per queste quantità e queste stime determinerebbero la dimensione del campione per il campione della seconda fase.

Una domanda che ci si pone spesso è quanto un campione deve essere grande. In statistica vi sono semplici meccanismi per stimare la dimensione del campione necessaria per avere una certa probabilità di rilevare un effetto di una dimensione prespecificata o superiore. Questi meccanismi, che raggruppati prendono il nome di analisi di potenza, si basano su stime delle medie e varianze delle variabili nella popolazione da campionare. E' importante capire il sistema di cause che determinano popolazione e garantire che tutte le fonti di variazione siano prese in considerazione. Un gran numero di osservazioni non ha alcun valore se le principali fonti di variazione vengono trascurate nello studio.

Un'alternativa ad impostare una dimensione del campione predeterminata consiste nel lasciare che i dati "scelgano" la dimensione del campione. L'idea di base è continuare ad aumentare la dimensione del campione fino a quando i risultati o i riepiloghi non cambiano più molto (dove "molto" è impostato in anticipo). Le varianti di questa idea sono note come campionamento progressivo, campionamento adattivo o campionamento sequenziale.

Bisogna però far attenzione al fatto che molti degli strumenti comuni per l'inferenza statistica, inclusi i t-test, presuppongono che i dati comprendano un semplice campione casuale di una popolazione e che i singoli punti dati siano quindi statisticamente indipendenti. Molte delle tecniche di campionamento violeranno questa ipotesi. Esistono tecniche statistiche specializzate per trattare i dati generati da un numero qualsiasi di schemi di campionamento complessi. Il campionamento casuale da un database fornisce stime imparziali delle caratteristiche del database. Tuttavia, se il database stesso rappresenta un campione casuale o sistematicamente distorto dalla popolazione reale di interesse, nessuna tecnica statistica può salvare le inferenze risultanti.

I database in tempo reale contengono degli attributi (chiamati anche **features** o caratteristiche). Il problema della selezione delle caratteristiche sorge perchè la complessità di ricerca nello spazio delle ipotesi deve essere ridotta per ragioni pratiche, e caratteristiche ridondanti o irrilevanti possono avere effetti significativi sulla qualità dei risultati del metodo di analisi (**maledizione della dimensionalità**). L'idea alla base della maledizione della dimensionalità è che dati di grande dimensione sono difficili da processare, per una serie di motivi: il numero di esempi cresce esponenzialmente con il numero di variabili e non ci sono abbastanza osservazioni per ottenere buone stime.

Il **feature selection** è un processo che scegli un subset ottimo di features seguendo alcuni criteri. Vi sono sostanzialmente 3 approcci:

- **wrapper models**
- **filter models**
- **embedded methods**

Il wrapper model si basa su un algoritmo di data mining per determinare se un sottoinsieme di funzionalità è valido. L'algoritmo viene utilizzato come parte della funzione di valutazione e anche per indurre i patterns o il modello finale. L'algoritmo DM può risolvere task predittivi o task descrittivi. Se vogliamo avere un buon insieme di funzionalità per migliorare l'accuratezza di un classificatore, possiamo utilizzare proprio questa misura per basare le evoluzioni del classificatore. Però sorgono diversi problemi, il principale consiste nel determinare veramente l'accuratezza predittiva evitando l'overfitting. Altri problemi consistono nel fatto che un classificatore richiede tempo per apprendere i dati, oppure i dati sono troppo grandi per eseguire un algoritmo di apprendimento, quindi è necessario ridurne la dimensionalità. Vi è quindi la necessità di definire alcuni criteri di stop per garantire che il processo di valutazione termini e che non entri in un loop infinito.

Il filter models è indipendente dall'algoritmo di data mining che sarà utilizzato sul subset di caratteristiche. I subset sono valutati utilizzando proprietà intrinseche dei dati quali le misure informative e di distanza. Se consideriamo l'accuratezza stimata da un classificatore come un'altra misura, possiamo unificare i modelli di filtro e wrapper in un modello generale. Ogni componente può avere diverse scelte. Le varie combinazioni di queste scelte sono alla base di molti algoritmi di selezione delle caratteristiche esistenti.

Alcune strategie per la selezione delle caratteristiche dei subset sono:

- **enumerazione** di tutte i possibili subset e selezione dei migliori
- **generazione casuale** dei subset e selezione del migliore.
- **generazione sequenziale** dei subsets.

Possiamo inoltre selezionare i subset in due modi: **foward selection**, iniziando con un subset vuoto e gradualmente aggiungere una caratteristica alla volta, o **backward selection**, nel quale si inizia con un insieme completo e si rimuove una caratteristica alla volta. Queste strategie sono basate sulla **strategia di ricerca greedy** in uno spazio di subset grande $2^N$, dove N è il numero di caratteristiche.

Indipendentemente dal metodo di generazione dei sottoinsiemi di funzionalità adottato, è necessaria una misura per decidere quale funzionalità deve essere aggiunta o rimossa o quale sottoinsieme deve essere mantenuto. Secondo la **misura dell'informazione**, data una funzione di incertezza **U** e le probabilità delle classi precedenti $\mathbf{P}(c_i)$, l'informazione ottenuta da una caratteristica X, **IG(X)**, è definita come la differenza tra la precedente incertezza e l'incertezza posteriore attesa usando X. In formula

$$IG(X) = \sum_i U(P(c_i)) - E[\sum_i U(P(c_i|X)) \tag{1}$$

Una regola di valutazione delle caratteristiche derivata dal concetto di guadagno di informazioni afferma che la caratteristica X è preferita alla caratteristica Y se $IG(X) > IG(Y)$. Cioè, una caratteristica dovrebbe essere selezionata se può ridurre più incertezza. Se U(x)=$-x * log(x)$ allora $\sum_i U(P(c_i))$ è una misura di entropia.

***Misure della distanza.*** Se l'obiettivo è la classificazione, una misura alternativa può essere la distanza tra le funzioni di densità classe-condizione. Se *P(X|C)* è la funzione di densità condizionata dalla classe della caratteristica X, nei due casi della clase (C=$c_1$ o C=$c_2$),P($X|C$) è definita da P($X|c_1$) e P($X|c_2$). Se D(X) è la distanza tra P($X|c_1$) e P($X|c_2$), una regola di valutazione basata su P afferma che X è preferito a Y se D(X)>D(Y). Questo perchè stiamo cercando una feature che può separare due classi il più possibile. Maggiore è la distanza, più facile separare le due classi.

La divergenza Kullback-Leibler è una misura della distanza tra distribuzioni probabili, P e Q su un dominio V. Questa divergnza è definita come:

$$m_{KL}(P,Q) = \sum_{v \in V} q(v) log(\frac{q(v)}{p(v)}) \tag{2}$$

Questa divergenza misura in quale misura la distribuzione P è un'approssimazione della distribuzione Q o, più precisamente, la perdita dell'informazione se noi prendessimo P invece di Q. In sintesi si misura quanto P diverge da Q. LE proprietà di questa misura sono:

- Assimetrica, ovvero $m_{KL}(P,Q) \neq m_{KL}(Q,P)$
- non definita quando p(v)=0

- nel caso specifico di q(v)=0, q(v)log(q(v)/p(v))=0
- il range dei valori non è limitato. Quindi, possiamo utilizzare questo valore per stabilire quale delle due distribuzioni Q e Q' è una migliore approssimazione di P, ma non ci permette di determinare in termini assoluti se Q è una buona approssimazione di P osservando $m_{KL}(P,Q)$

La divergenza del $\chi^2$ è definita come:

$$m\chi^2(P,Q) = \sum_{y \in Y} \frac{|p(y) - q(y)|^2}{p(y)} \tag{3}$$

è rigorosamente topologicamente più forte della divergenza KL data la disuguaglianza $m_{KL}(P,Q) \leq m\chi^2(P,Q)$, la convergenza nella funzione di divergenza $\chi^2$ implica la convergenza nella divergenza KL, ma non il contrario. Inoltre è assimetrica e non definita quando p(y)=0.

Un ultimo tipo di misura della distanza eè la distanza di variazione, data dalla formula $m_1(P,Q) = \sum_{y \in Y} |p(y) - q(y)|$, detta anche distanza di Manhattan per funzioni di probabilità p(y) e q(y) e coincide con la distanza di Hamming quando tutte le features sono binari. Similarmente si può utilizzare la distanza Euclidea data da $m_2(P,Q) = \sum_{y \in Y} |p(y) - q(y)|^2$. Queste due metriche soddisfano la proprietà di simmetria e le proprietà metriche.

***Dipendenza dalle misure***. Si verifica con quanta forza una caratteristica è associata alla classe. Denotando attraverso R(X) una misura di dipendenza tra la feature X e la classe C, preferiamo la feature X alla feature Y se R(X)>R(Y). Un problema con le tre misure precedenti è che non possono rompere i legami tra due features ugualmente buone. Pertanto queste funzionalità non possono rilevare se una di esse è ridondante.

***Inconsistenza delle misure***. Si tenta di trovare un numero minimo di funzionalità che separano le classi in modo coerente come può fare l'intero set di funzionalità. In altre parole, le misure di incoerenza mirano a raggiungere $P(C|FullSet) = P(C|SubSet)$ Le regole di valutazione delle funzionalità derivate dalle misure di incoerenza affermano che è necessario selezionare il sottoinsieme minimo di funzionalità in grado di mantenere la coerenza dei dati mantenuti dall'insieme completo di funzionalità.

Per **selezionare le feature** ci sono 3 componenti necessari: un generatore di subset, un valutatore e un criterio di stop. Vi sono diversi metodi. ***Approcci completi ed esaustivi***: il *focus* è applicato su una misura di consistenza e valuta esaustivamente tutti i subset partendo da una feature, mentre il *branch and bound* consiste in una enumerazione sistematica di tutte le soluzioni, dove ampi sottoinsiemi di candidati infruttuosi sono scartati in massa utilizzando dei limiti superiori e inferiori della quantità da ottimizzare. Inizia con un insieme completo di feature e valuta l'accuratezza stimata. ***Approcci Euristici***: ***SFS*** (sequential foward search) e ***SBS***(sequential backward search) possono essere applicati a ciascuna delle misure, ***DTM*** è la più semplice versione di una modalità di weapper - impara un classificatore una volta e usa qualsiasi caratteristica trovata nel classificatore. ***Approcci non deterministici***: ***LVF*** (Las Vegas Filter) e ***LVW*** (Las Vegas wrapper), generano subset di feature casualmente e li testano in maniera differente, LVF applica una misura inconsistente, LVW usa una stima accurata attraverso un classificatore; ***algoritmi generici*** e ***ricottura simulata*** sono anche usati nella selezione di feature. Il primo può produrre più sottoinsiemi, il secondo produce un singolo sottoinsieme. ***Approcci basati sulle instanze***: ***Relief***, molti piccoli campioni di dati sono campioni proveniente dai dati. Le funzionalità vengono ponderate in base ai loro ruoli nella differenziazione di istanze di classi diverse per un campione di dati. È possibile selezionare funzioni con pesi maggiori.

Per le attività di data mining non di classificazione (nessuna etichetta di classe disponibile), dovrebbero essere presi in considerazione metodi alternativi. Ad esempio, una misura di entropia può essere introdotta per classificare in sequenza le

caratteristiche. L'idea di base è che le caratteristiche sono rilevanti se possono descrivere le istanze in termini di cluster relativamente chiaramente definiti.

La **scalabilità** è un altro problema. In LVS la parte più dispendiosa in termini di tempo di un processo di selezione delle funzionalità viene identificata e ritardata fino a quando non è necessario. LVS è un'estensione di LVF che utilizza una misura di incoerenza (IC) con una complessità di runtime di controllo O(n), dove n è il numero di istanze. Se n è enorme, è costoso calcolare IC molte volte. Si noti inoltre che il componente di generazione di sottoinsiemi di funzionalità genererà sempre più sottoinsiemi non validi che non soddisfano IC man mano che la cardinalità di un sottoinsieme valido diminuisce. Pertanto, ha senso separare il calcolo di IC come cardinalità per tutti i dati dalla generazione di sottoinsiemi di funzionalità. Ma abbiamo bisogno di dati per generare sottoinsiemi di funzionalità. Il compromesso è che invece di utilizzare l'intero set di dati, ne utilizziamo solo una parte per la generazione di sottoinsiemi di funzionalità. Quando un sottoinsieme viene testato come valido sulla porzione di dati, viene calcolato l'IC per l'intero dato. Successivamente se IC è stato soddisfatto, allora la selezione della feature è completata, altrimenti se IC non è stato soddisfatto, le istanze inconsistenti vengono aggiunte alla porzione di dati e viene eseguito un altro ciclo di generazione di sottoinsiemi di funzionalità sulla porzione di dati ingrandita. Questo metodo è particolarmente efficace solo quando n è sufficientemente largo a causa del sovraccarico nel LVS.

**Wrapper models** cercano di risolvere uno specifico problema, quindi il criterio può realmente essere specificato. Invece consuma molto tempo se bisogna valutare uno schema a ogni iterata. **Filter models** sono molto più veloci ma non incorporano algoritmi di data mining usati per la generazione di un modello o pattern, quindi il modello può essere subottimale.

A differenza dei modelli di filtri e wrapper, nei metodi embedded la parte di selezione delle caratteristiche non può essere separata dall'algoritmo di data mining. È parte integrante dell'algoritmo. Per esempio, nell'algoritmo di decision tree, la selezioni delle caratteristiche che contribuiscono alla creazione dell'albero finale è parte della costruzione dell'algoritmo di decision tree. Poiché siamo interessati a selezionare le funzionalità nella fase di trasformazione dei dati, non consideriamo gli embedded methods.

La **pulizia dei dati** aumenta la qualità dei dati al livello richiesto. Ciò comporta la selezione di sottoinsiemi di dati puliti, l'inserimento di impostazioni predefinite adeguate o tecniche più ambiziose come la stima della modellazione dei dati mancanti. Alla fine di questa fase vi è un rapporto sulla pulizia dei dati, che descrive le decisioni e le azioni per affrontare i problemi di qualità dei dati ed elenca le trasformazioni dei dati per la pulizia e i possibili impatti sull'analisi dei risultati. I due problemi più comuni sono la mancanza di valori e dati di rumore. I **dati di rumore** consistono in variabile che hanno valori che non sono conformi con quelli che ci aspettiamo dalle variabili. Le osservazioni in cui si verificano questi valori rumorosi sono chiamate valori anomali (*outliers*). Differenti tipi di outliers possono essere trattati in differenti modi. Vi possono essere errori umani, i quali possono essere corretti o cancellati dall'analisi, oppure le distribuzioni simmetriche spesso indicano valori anomali. La mancanza di dati, invece, include valori che non sono presenti nei dati selezionati e questi valori non validi bisogna eliminarli durante il rilevamento del rumore. Questo caso si verifica se viene commesso un errore durante l'inserimento dei dati, oppure se l'informazione non era disponibile al momento dell'inserimento oppure i dati selezionati all'interno di risorse eterogenee hanno creato dei mismatch. Per correggere questo tipo di errore vengono eseguite diversi tipi di azioni, l'inserimento di un valore predefinito come il termine "none" è l'ideale. Si potrebbe cancellare le righe che presentano valori mancanti, però questa tecnica, per quanto facile da implementare, può generare la perdita di dati che possono essere valutati. Un'altra tecnica consiste nell'eliminazione della variabile dall'analisi se presnta un significativo numero di osservazioni con valori mancanti per la stessa variabile. Infine, un'ultima tecnica, consiste nel rimpiazzare il valore mancante con un altro valore, che nel caso di variabili quantitative può consistere nella media o nella mediana, mentre per le variabili categoriche può essere rappresentato dalla moda o dal valore "sconosciuto". Si potrebbe anche pensare di attuare un approccio più sofisticato è quello di predire il valore più probabile delle variabili all'interno delle osservazioni.

Successivamente si passa alla fase di **costruzione dei dati**, la quale include operazioni di preparazione dei dati, come la generazione di variabili derivate, inserimento di nuovi record o trasformazione di variabili esistenti. I dati possono essere trasformati in *una singola variabile*, per essere ulteriormente perfezionati per soddisfare i requisiti del formato di input dei particolari algoritmi di data mining da utilizzare. Esempi sono la conversione delle variabili di tipo data dal formato US a quello Europeo, oppure il calcolo dell'età data la data di nascita, l'aggregazione dei valori all'interno di un conto corrente per gli ultimi 3,6,12 mesi. Si può effettuare un ridimensionamento o una normalizzazione dei dati. Si parla di normalizzazione dei dati quando colonne numeriche sono trasformate usando funzioni matematiche in dei range. Questo processo è importante perchè tutte le variabili all'interno di una colonna devono essere trattate in maniera uguale e non devono influenzarsi a vicenda, oppure perchè alcuni dati possono ricevere solo alcuni valori all'interno di un range.

Un altro tipo di normalizzazione è la normalizzazione min-max che performa una transformazione lineare sui dati originali. Supponendo che $min_A$ e $max_A$ sono i valori di minimo e di massimo di un attributo A, questa normalizzazione mappa un valore v di A in v' nel range $[newMin_A; newMax_A]$ attraverso la formula:

$$v' = \frac{v - min_A}{max_A - min_A}(newMax_A - newMin_A) + newMin_A \tag{4}$$

Le relazioni tra i valori dei dati originali vengono preservate. Se un caso di input futuro per la normalizzazione non rientra nell'intervallo di dati originale per A, si verifica un errore "out of bounds" (fuori dal limite).

Nella normalizzazione definita z-score, anche chiamata a media zero), i valori per un attributo A sono normalizzati sulla base della media o della deviazione standard di A. Un valore v di A è normalizzato in v' attraverso la formula:

$$v' = \frac{v - mean_A}{standDev_A} \tag{5}$$

dove $mean_A$ e $standDev_A$ sono la media e la deviazione standard dell'attributo A. Questo metodo di normalizzazione è utile quando il minmo e il massimo dell'attributo A sono sconosciuti, o quando ci sono gli outliers che dominano la normalizzazione min-max.

La normalizzazione attraverso il decimal scaling transforma i valori dell'attributo A in punti decimali, per assicurarsi che il range dell'intervallo i cui essi sono compresi sia {-1,+1}. Il numero di punti decimali dipende dal massimo valore assoluto di A. Un valore v di A è normalizzato in v' attraverso la funzione:

$$v' = \frac{v}{10^j} \tag{6}$$

dove j è il più piccolo intero tale che $max(|v'|) < 1$

Si parla di discretizzazione delle variabili quando convertiamo variabili quantitative in variabili categoriche dividendo il valore di input in intervalli. Due metodi di discretizzazione sono l'Equale width e l'Equal depth. In entrambi questi metodi le informazioni sulla classe non vengono utilizzate nel caso in cui le osservazioni siano preclassificate, inoltre la discretizzazione viene applica a ogni attributo indipendentemente dagli altri. Il partizionamento attraverso l'equal width divide il range in N intervalli di uguale lunghezza, se A e B sono il più piccolo e il più grande valore all'interno dell'attributo, l'intervallo di width sarà:

$$W = \frac{B - A}{N} \tag{7}$$

Questo metodo è il più diretto, però ha anche degli aspetti negativi perchè non gestisce bene i dati distorti ed è sensibile ai valori anomali.

Il partizionamento attraverso l'equal-depth divide il range in N intervalli, ognuno contenente approssimativamente lo stesso numero di esempi. Questo tipo di partizionamento ha una buona scalabilità, gestisce gli attributi categorici attraverso alcuni trucchetti, minimizza le informazioni perdute durante il processo di partizionamento.

In entrambi i casi il numero di elementi tralasciati è definito dall'utente. Nel caso di classificazione dei dati, è buona pratica che questo numero non sia minore del numero di classi che vogliamo riconoscere, oppure determinarlo attraverso la formula:

$$N_{bins} = \frac{M}{3 * C} \tag{8}$$

dove M è il numero di esempi di training e C il numero di classi.

Procedure complesse di conversione di dati hanno l'obiettivo di costruire un piccolo insieme di indici da un vasto numero di variabili, in modo tale che solo alcune informazioni vengano perse e il numero totale di funzioni venga ridotto. L'analisi fattoriale e l'analisi del componente principale sono due tecniche di analisi multivariate per la riduzione dei dati. L'analisi fattoriale affronta il problema dell'analisi della struttura delle interrelazioni (correlazioni) tra un GRANDE numero di variabili (ad es. punteggi dei test, elementi del test, risposte al questionario) definendo un insieme di dimensioni sottostanti comuni, note come fattori. Questa non è una tecnica che usa dipendenze, ovvero non vi è nessuna variabile considerata come criterio o variabile dipendente da cui tutte le altre dipendono e sono le variabili predittore o indipendenti. E' una tecnica interdipendente in cui ogni variabile è considerata simultaneamente e ognuna è in relazione alle altre.

Oltre alla costruzione dei dati, vi è l'integrazione dei dati, che ha come obiettivo la combinazione di informazioni provenienti da tabelle multiple o record per creare nuovi record o valori. L'output di questa fase è un insieme di dati uniti, che si riferisce all'unione di due o più tabelle unite che contengono differenti informazioni sugli stessi oggetti, o dati aggregati, sui quali verranno effettuate delle operazioni per produrre nuovi dati. Il risultato è il modello dei dati analitici, che rappresenta una ristrutturazione consolidata, integrata e dipendente dal tempo dei dati selezionati e preelaborati dalle varie fonti operative ed esterne.