POLITECNICO DI MILANO
Scuola di Ingegneria Industriale e dell'Informazione
Corso di Laurea Magistrale in Ingegneria Informatica
Dipartimento di Elettronica, Informazione e Bioingegneria

Database exploration: una tecnica statistica basata sull'entropia

Relatore: Prof. Manuel Roveri
Correlatori: Prof.ssa Letizia Tanca
Prof.ssa Elisa Quintarelli

Tesi di laurea di: Filippo Molteni
Matricola: 798292

Anno accademico 2014-1015

Ringraziamenti

Un doveroso ringraziamento va ai miei genitori Rosella e Claudio, e ai miei fratelli Fabio e Viola, che mi hanno sempre sostenuto durante questi anni di studio.

Desidero ringraziare Anika, che ha sempre creduto in me e che mi ha sopportato anche nei momenti più difficili.

Infine, desidero ringraziare i miei compagni di università Andrea, Luca, Mattia, Alessandro, Giulia, Simone, Simone e Salvatore, grazie ai quali ho un bellissimo ricordo di questi anni di studio.

Indice

1 Introduzione 1
1.1 Cos'è la data exploration? 3
1.2 Tecniche di Exploratory Computing 5
1.3 Metodo di exploratory computing proposto 7
1.4 Aspetti innovativi e domini di applicazione 9
2 Exploratory computing: idee e meccanismi 11
2.1 Definizione di database 11
2.2 Esempio motivante 12
2.3 Iniziare la conversazione 13
2.4 Modello di rilevanza 14
2.5 Proseguimento della conversazione 17
3 Analisi della letteratura 20
3.1 Subgroup Discovery 20
3.1.1 Definizione di Subgroup Discovery 20
3.1.2 Elementi principali di un algoritmo di Subgroup Discovery 21
3.1.3 Classi di algoritmi di Subgroup Discovery 23
3.2 Algoritmo CN2-SD 24
3.2.1 Misura di qualità 26
3.2.2 Probabilistic classification dell'algoritmo CN2-SD 27
3.3 Algoritmo APRIORI-SD 28
3.3.1 Misura di qualità 30
3.3.2 Probabilistic classification dell'algoritmo APRIORI-SD 30
3.4 Algoritmo SD-MAP 31
3.4.1 Misura di qualità 33
3.5 Confronto tra CN2-SD, APRIORI-SD e SD-MAP 35
3.6 Data mining 37
3.7 Exploratory Computing attraverso tecniche statistiche 38

4 Soluzione proposta 41
4.1 Commento generale 41
4.2 Contributo innovativo 43
4.3 Idea di base dell'algoritmo proposto 45
4.4 Meccanismi di campionamento 47
4.5 Calcolo degli Indicatori Statistici 49
4.5.1 Entropia di una variabile aleatoria 49
4.5.2 Entropia congiunta di due variabili aleatorie 50
4.5.3 Entropia condizionata di due variabili aleatorie 51
4.5.4 Mutua informazione di due variabili aleatorie 51
4.5.5 Relazione tra entropia e mutua informazione 52
4.6 Descrizione metodo di bootstrap utilizzato 54
4.7 Valutazione sovrapposizione degli intervalli di confidenza 56
4.8 Generalizzazione della tecnica proposta 59
5 Parte sperimentale 61
5.1 Analisi dataset dati sintetici 61
5.1.1 Informazioni preliminari 61
5.1.2 Analisi probabilità congiunta casuale 64
5.1.3 Analisi probabilità congiunta uniforme 70
5.1.4 Analisi probabilità congiunta sbilanciata 75
5.1.5 Riassunto delle osservazioni sui risultati ottenuti sul dataset sintetico 81
5.2 Analisi database dati medici 88
5.2.1 Informazioni preliminari 88
5.2.2 Descrizione della soluzione basata su two-sample chi square test 90
5.2.3 Risultati ottenuti 92
5.2.4 Osservazioni finali sui risultati 105
6 Conclusioni 107
6.1 Sviluppi futuri 108
Bibliografia 109

Indice delle tabelle

Tabella 3.1 Classificazione degli algoritmi di subgroup discovery esistenti 23
Tabella 3.2 Caratteristiche degli algoritmi CN2-SD, APRIORI-SD e SD-Map 36
Tabella 5.1 Dati sintetici, probabilità congiunta casuale: Probabilità congiunta di A, B, C in input 64
Tabella 5.2 Dati Sintetici, probabilità congiunta casuale: valori indicatori statistici ottenuti con l'approccio teorico 65
Tabella 5.3 Dati Sintetici, probabilità congiunta casuale: valori indicatori statistici ottenuti con l'approccio sperimentale 65
Tabella 5.4 Dati sintetici, probabilità congiunta casuale, coppia A-B: Entropia condizionata H(A|B=x) e H(B|A=x), con x=1,2,3 65
Tabella 5.5 Dati sintetici, probabilità congiunta casuale, coppia A-B: Entropia condizionata H(A|C=x) e H(C|A=x), con x=1,2,3 65
Tabella 5.6 Dati sintetici, probabilità congiunta casuale, coppia A-B: Entropia condizionata H(B|C=x) e H(C|B=x), con x=1,2,3 66
Tabella 5.7 Dati sintetici, probabilità congiunta casuale: valori medi indicatori statistici calcolati con bootstrap 66
Tabella 5.8 Dati sintetici, probabilità congiunta casuale: varianze indicatori statistici calcolati con bootstrap 66
Tabella 5.9 Dati sintetici, probabilità congiunta casuale: valor medio e varianza H(A|B=x) e H(B|A=x), con x=1,2,3 ottenute con bootstrap 67
Tabella 5.10 Dati sintetici, probabilità congiunta casuale: sovrapposizione intervalli di confidenza H(A) - H(A|B=b), con b=1,2,3 e H(B) – H(B|A=a), con a=1,2,3 al variare di lambda 67
Tabella 5.11 Dati sintetici, probabilità congiunta casuale: valor medio e varianza H(A|C=x) e H(C|A=x), con x=1,2,3 ottenute con bootstrap 67
Tabella 5.12 Dati sintetici, probabilità congiunta casuale: sovrapposizione intervalli di confidenza H(A) - H(A|C=c), con c=1,2,3 e H(C) – H(C|A=a), con c=1,2,3 al variare di lambda 68
Tabella 5.13 Dati sintetici, probabilità congiunta casuale: valor medio e varianza H(B|C=x) e H(C|B=x), con x=1,2,3 ottenute con bootstrap 68
Tabella 5.14 Dati sintetici, probabilità congiunta casuale: sovrapposizione intervalli di confidenza H(B) - H(B|C=c), con c=1,2,3 e H(C) – H(C|B=b), con b=1,2,3 al variare di lambda 68
Tabella 5.15 Dati sintetici, probabilità congiunta uniforme: Probabilità congiunta di A,B,C in input 70

Tabella 5.16 Dati Sintetici, probabilità congiunta uniforme: valori indicatori statistici ottenuti con l'approccio teorico 71
Tabella 5.17 Dati Sintetici, probabilità congiunta uniforme: valori indicatori statistici ottenuti con l'approccio sperimentale 71
Tabella 5.18 Dati sintetici, probabilità congiunta uniforme, coppia A-B: Entropia condizionata H(A|B=x) e H(B|A=x), con x=1,2,3 71
Tabella 5.19 Dati sintetici, probabilità congiunta casuale, coppia A-B: Entropia condizionata H(A|C=x) e H(C|A=x), con x=1,2,3 71
Tabella 5.20 Dati sintetici, probabilità congiunta casuale, coppia A-B: Entropia condizionata H(B|C=x) e H(C|B=x), con x=1,2,3 72
Tabella 5.21 Dati sintetici, probabilità congiunta uniforme: valori medi indicatori statistici calcolati con bootstrap 72
Tabella 5.22 Dati sintetici, probabilità congiunta uniforme: varianze indicatori statistici calcolati con bootstrap 72
Tabella 5.23 Dati sintetici, probabilità congiunta uniforme: valor medio e varianza H(A|B=x) e H(B|A=x), con x=1,2,3 ottenute con bootstrap 73
Tabella 5.24 Dati sintetici, probabilità congiunta uniforme: sovrapposizione intervalli di confidenza H(A) - H(A|B=b), con b=1,2,3 e H(B) – H(B|A=a), con a=1,2,3 al variare di lambda 73
Tabella 5.25 Dati sintetici, probabilità congiunta uniforme: valor medio e varianza H(A|C=x) e H(C|A=x), con x=1,2,3 ottenute con bootstrap 73
Tabella 5.26 Dati sintetici, probabilità congiunta uniforme: sovrapposizione intervalli di confidenza H(A) - H(A|C=c), con c=1,2,3 e H(C) – H(C|A=a), con c=1,2,3 al variare di lambda 74
Tabella 5.27 Dati sintetici, probabilità congiunta uniforme: valor medio e varianza H(B|C=x) e H(C|B=x), con x=1,2,3 ottenute con bootstrap 74
Tabella 5.28 Dati sintetici, probabilità congiunta uniforme: sovrapposizione intervalli di confidenza H(B) - H(B|C=c), con c=1,2,3 e H(C) – H(C|B=b), con b=1,2,3 al variare di lambda 74
Tabella 5.29 Dati sintetici, probabilità congiunta sbilanciata: Probabilità congiunta di A,B,C in input 75
Tabella 5.30 Dati Sintetici, probabilità congiunta sbilanciata: valori indicatori statistici ottenuti con l'approccio teorico 76
Tabella 5.31 Dati Sintetici, probabilità congiunta sbilanciata: valori indicatori statistici ottenuti con l'approccio sperimentale 76
Tabella 5.32 Dati sintetici, probabilità congiunta sbilanciata, coppia A-B: Entropia condizionata H(A|B=x) e H(B|A=x), con x=1,2,3 76

Tabella 5.33 Dati sintetici, probabilità congiunta sbilanciata, coppia A-B: Entropia condizionata H(A|C=x) e H(C|A=x), con x=1,2,3 77
Tabella 5.34 Dati sintetici, probabilità congiunta sbilanciata, coppia A-B: Entropia condizionata H(B|C=x) e H(C|B=x), con x=1,2,3 77
Tabella 5.35 Dati sintetici, probabilità congiunta sbilanciata: valori medi indicatori statistici calcolati con bootstrap 77
Tabella 5.36 Dati sintetici, probabilità congiunta sbilanciata: varianze indicatori statistici calcolati con bootstrap 78
Tabella 5.37 Dati sintetici, probabilità congiunta sbilanciata: valor medio e varianza H(A|B=x) e H(B|A=x), con x=1,2,3 ottenute con bootstrap 78
Tabella 5.38 Dati sintetici, probabilità congiunta sbilanciata: sovrapposizione intervalli di confidenza H(A) - H(A|B=b), con b=1,2,3 e H(B) – H(B|A=a), con a=1,2,3 al variare di lambda 78
Tabella 5.39 Dati sintetici, probabilità congiunta sbilanciata: valor medio e varianza H(A|C=x) e H(C|A=x), con x=1,2,3 ottenute con bootstrap 79
Tabella 5.40 Dati sintetici, probabilità congiunta sbilanciata: sovrapposizione intervalli di confidenza H(A) - H(A|C=c), con c=1,2,3 e H(C) – H(C|A=a), con c=1,2,3 al variare di lambda 79
Tabella 5.41 Dati sintetici, probabilità congiunta sbilanciata: valor medio e varianza H(B|C=x) e H(C|B=x), con x=1,2,3 ottenute con bootstrap 79
Tabella 5.42 Dati sintetici, probabilità congiunta sbilanciata: sovrapposizione intervalli di confidenza H(B) - H(B|C=c), con c=1,2,3 e H(C) – H(C|B=b), con b=1,2,3 al variare di lambda 80
Tabella 5.43 Dati sintetici, probabilità congiunta sbilanciata: probabilità congiunta coppia attributi A-B 80
Tabella 5.44 Dati sintetici, probabilità congiunta sbilanciata: probabilità congiunta coppia attributi A-B 86
Tabella 5.45 Dati sintetici, probabilità congiunta sbilanciata: probabilità congiunta coppia A-C 87
Tabella 5.46 Dati sintetici, probabilità congiunta sbilanciata: probabilità congiunta coppia B-C 88
Tabella 5.47 Database dati medici: valori indicatori statistici ottenuti considerando tutte le tuple 93
Tabella 5.48 Database dati medici, coppia Reparto-Sesso: valori entropia H(X|Y=y) ottenuti dalla duplice analisi 94
Tabella 5.49 Database dati medici, coppia Reparto-Sesso: comparazione risultati ottenuti con i due test per H(X) – H(X|Y=y) 94

Tabella 5.50 Database dati medici, coppia Reparto-Giornate Degenza: valori entropia H(X|Y=y) ottenuti dalla duplice analisi 95
Tabella 5.51 Database dati medici, coppia Reparto-Giornate Degenza: comparazione risultati ottenuti con i due test per H(X) –H(X|Y=y) 95
Tabella 5.52 Database dati medici, coppia Reparto-Età assistito: valori entropia H(X|Y=y) ottenuti dalla duplice analisi 96
Tabella 5.53 Database dati medici, coppia Reparto-Età assistito: comparazione risultati ottenuti con i due test per H(X) –H(X|Y=y) 97
Tabella 5.54 Database dati medici, coppia Sesso-Giornate Degenza: valori entropia H(X|Y=y) ottenuti dalla duplice analisi 97
Tabella 5.55 Database dati medici, coppia Sesso-Giornate Degenza: comparazione risultati ottenuti con i due test per H(X) –H(X|Y=y) 98
Tabella 5.56 Database dati medici, coppia Sesso-Età assistito: valori entropia H(X|Y=y) ottenuti dalla duplice analisi 98
Tabella 5.57 Database dati medici, coppia Sesso-Età assistito: comparazione risultati ottenuti con i due test per H(X) –H(X|Y=y) 99
Tabella 5.58 Database dati medici, coppia Giornate Degenza-Età assistito: valori entropia H(X|Y=y) ottenuti dalla duplice analisi 100
Tabella 5.59 Database dati medici, coppia Giornate Degenza-Età assistito: comparazione risultati ottenuti con i due test per H(X) –H(X|Y=y) 100
Tabella 5.60 Database dati medici, coppia Giornate Degenza-Età assistito: valori entropia H(Y|X=x) ottenuti dalla duplice analisi 101
Tabella 5.61 Database dati medici, coppia Giornate Degenza-Età assistito: comparazione risultati ottenuti con i due test per H(Y) –H(Y|X=x) 102
Tabella 5.62 Database dati medici, coppia Giornate Degenza-Cmdc: valori entropia H(Y|X=x) ottenuti dalla duplice analisi 103
Tabella 5.63 Database dati medici, coppia Giornate Degenza-Cmdc: comparazione risultati ottenuti con i due test per H(Y) –H(Y|X=x) 103
Tabella 5.64 Database dati medici, coppia Età assistito-Cmdc: valori entropia H(Y|X=x) ottenuti dalla duplice analisi 104
Tabella 5.65 Database dati medici, coppia Età assistito-Cmdc: comparazione risultati ottenuti con i due test per H(Y) –H(Y|X=x) 104

Indice delle figure

Figura 1.1 Esempio sovrapposizione intervalli di confidenza entropia di due attributi: (a) intervalli sovrapposti, (b) intervalli disgiunti 8
Figura 3.1 Pseudocodice algoritmo CN2-SD 26
Figura 3.2 Pseudocodice algoritmo APRIORI-SD 29
Figura 3.3 Pseudocodice algoritmo SD-Map 32
Figura 3.4 Data Mining Process 38
Figura 4.1 Pseudocodice algoritmo proposto 47
Figura 4.2 Relazione tra entropia e mutua informazione 52
Figura 4.3 Pseudocodice della funzione utilizzata per conditionalEntropy, utilizzata per calcolare l'entropia condizionata di due attributi 54
Figura 4.4 Esempio sovrapposizione intervallo di confidenza: intervalli sovrapposti (a), intervalli disgiunti(b) 59

Sommario

Negli ultimi anni il settore dei cosiddetti Big Data ha visto un interesse sempre più crescente da parte della comunità scientifica. Aiutare le persone a dare un senso ai dati salvati in basi di dati di grandi dimensioni è al giorno d'oggi considerato un importante argomento di ricerca.

Dal momento che gli strumenti attualmente disponibili per l'analisi dei dati i riferiscono, tipicamente, ad utenti esperti del settore, si rende necessario sviluppare degli strumenti che possano essere di supporto a tutti i possibili utenti.

In letteratura sono state presentate molte tecniche per l'esplorazione delle basi di dati, le quali si basano sull'identificazione di sottoinsiemi interessanti a partire da una basi di dati nota, attraverso la costruzione di regole.

In questo lavoro viene presentata una nuova tecnica statistica per l'esplorazione di informazioni, basata sull'entropia degli attributi, la quale opera direttamente sui dati memorizzati in una base di dati.

Per testare la solidità del metodo proposto è stata condotta una vasta campagna sperimentale su una base di dati sintetica, la quale ha permesso di verificare la validità della tecnica statistica basata sull'entropia.

Dopo aver terminato gli esperimenti sui dati sintetici, abbiamo rivolto l'attenzione sull'applicazione della tecnica proposta ad una base di dati reale, contenente dati biomedici.

Utilizzando questa base di dati, l'approccio proposto è stato confrontato con un altro metodo utilizzato per la database exploration, il two-sample chi square test; analizzando i risultati ottenuti abbiamo messo in evidenza la validità della tecnica statistica basata sull'entropia proposta.

Capitolo 1

Introduzione

Al giorno d'oggi le grandi quantità di informazioni non strutturate (i 'Big Data') stanno ricevendo un attenzione sempre più crescente in tutte le aree dell'economia globale. Per decenni, le compagnie hanno basato le loro decisioni di business su dati transazionali salvati in basi di dati relazionali. Dietro la criticità dei dati, tuttavia, c'è un potenziale tesoro di dati non tradizionali e meno strutturati: come weblogs, social media, sensori e fotografie, i quali possono essere analizzati per estrarre informazioni utili. Il decremento sia dei costi delle memorie sia della potenza computazionale ha fatto sì che fosse possibile collezionare questi dati, che sarebbero stati sprecati solo pochi anni fa. Come risultato, molte aziende stanno cercando il modo di includere questi dati non tradizionali, ma potenzialmente molto preziosi, nei loro dati tradizionali e di conseguenza anche nelle loro analisi dei dati.

Con il termine Big Data ci si riferisce, tipicamente, ai seguenti tipi di dati:

- Dati aziendali tradizionali: includono informazioni sui clienti derivanti dai sistemi CRM: data transazionali derivanti dall'ERP, database delle transazioni online, etc.
- Machine generated/sensor data: includono dati derivanti da weblogs, sensori, smart meters, digital exhaust;
- Social data: includono i feedback dei clienti, dati derivanti da social media come Facebook, Twitter.

Il McKinsey Global Institute ha stimato che il volume di dati crescerà del 40% all'anno da qui fino al 2020.

Nonostante il volume sia la caratteristica dei dati più evidente, questa non è l'unica.

Infatti ci sono quattro caratteristiche chiave per definire i Big Data:

- Volume. I dati generati dalle macchine sono prodotti in maggiore quantità rispetto ai dati non tradizionali. Per esempio, il motore di un jet può generare fino a 10TB di dati in 30 minuti, mentre una raffineria può produrre un tale volume di dati in un giorno.
- Velocità. La velocità con il quale i dati cambiano è cresciuta vertiginosamente nell'ultimo decennio, così come è cresciuta la velocità con la quale essi vengono scambiati e salvati.
- Varietà. I formati di dati tradizionali tendono ad essere ben definiti da uno schema dei dati e cambiano lentamente. Contrariamente, i formati di dati non tradizionali hanno una vertiginosa rapidità di cambiamento.
- Valore. Il valore economico di dati differenti varia in maniera significativa. Tipicamente ci sono buone informazioni nascoste nei dati non-tradizionali, la sfida è quella di identificare quello che ha valore e poi trasformare e estrarre i dati per l'analisi.

Quando i Big Data vengono analizzati combinandoli con i dati tradizionali, le organizzazioni possono sviluppare una conoscenza più approfondita del loro business, che può portare a una maggiore produttività e competitività.

Per esempio, nel settore dei servizi medici, la gestione di malati cronici o a lungo termine è costosa. L'uso di "in-home devices" per monitorare i parametri vitali, e tener sotto controllo i progressi è solo uno dei modi in cui i sensori possono essere usati per migliorare la salute dei pazienti.

Le compagnie manifatturiere utilizzano sensori nelle loro produzioni per avere un flusso di dati di lavorazione e utilizzarli in maniera tale da essere in grado di migliorare la produzione, la competitività e ridurre i guasti.

Infine, i social media come Facebook e LinkedIn non esisterebbero senza i Big Data. Il loro modello di business richiede un'esperienza personalizzata sul web, che può essere fornita solo catturando e usando tutti i dati disponibili di un utente o un membro.

Per sfruttare i Big Data, le organizzazioni devono quindi evolvere le proprie infrastrutture IT[1] per gestire questo grande volume di dati che varia a grande velocità e poterlo integrare con il precedente sistema di analisi dei dati.

Vista la rapida diffusione dei Big Data è quindi necessario avere a disposizione degli strumenti efficienti per la gestione delle basi di dati di grandi dimensioni. Gli strumenti attualmente disponibili, come 1010data, Actian, Amazon web services, Cloudera, si riferiscono solamente ad una *clientela professionale*, mentre è fondamentale sviluppare dei software che possano essere utilizzati da quelli che in [7] e [8] vengono definiti *data enthusiasts* (giornalisti, investitori, etc.), ossia gente comune che può trarre vantaggio dall'esplorazione dei dati e dalla nuova conoscenza estratta da essi.

Nell'ultimo decennio, la comunità scientifica si è concentrata molto sullo sviluppo di tecniche per l'analisi dei Big Data. Le principali tecniche introdotte si basano sul machine learning, la subgroup discovery e l'association rule learning. Recentemente si è sviluppata una nuova tecnica, la ***data exploration***, la quale ha come scopo il processo di esplorazione dei dati e l'estrazione da essi di informazioni potenzialmente utili ma non esplicite.

1.1 Cos'è la data exploration?

Il termine 'data exploration' si riferisce generalmente ad un utente che è in grado di esplorare i dati, nella maniera a lui più consona, e trovare delle risposte utili interagendo con un elevato volume di dati. Una definizione più tecnica di database exploration, deriva dalla statistica, ed è stata introdotta da Tukey [10]: con l'analisi esplorativa dei dati, il ricercatore, esplora i dati in molti modi possibili, utilizzando anche strumenti grafici come boxplots o istogrammi, ricavando da essi nuove informazioni utili.

[1] Infrastrutture IT (Information Technology): si riferisce all'insieme delle risorse hardware, software e di rete e ai servizi richiesti per l'esistenza e la gestione di un ambiente IT. Permette ad un'organizzazione di fornire servizi IT ai suoi dipendenti, ai suoi partner e ai suoi clienti.

Nonostante l'enfasi sulla visualizzazione, l'analisi esplorativa dei dati si basa sull'assunzione che l'utente abbia delle basi di statistica, mentre in questo lavoro di tesi viene proposto un paradigma per la database exploration che è ispirato all' exploratory computing vision [11].

Possiamo descrivere l'exploratory computing come una "conversazione" *Step-by-Step* di un utente e un sistema, i quali si aiutano a vicenda a migliorare il processo di data exploration, e infine ottengono delle nuove informazioni utili che soddisfano a pieno le necessità dell'utente.

Questo processo è visto come una conversazione perché il sistema provvede a dare un supporto attivo: esso non risponde solamente alle richieste dell'utente, ma suggerisce a quest'ultimo una o più possibili azioni che possono aiutarlo a mettere pienamente a fuoco la sessione esplorativa.

Questa attività può comportare l'uso di una vastità di tecniche differenti, come tecniche statistiche e di analisi dei dati, suggerimento di interrogazioni, strumenti di visualizzazione avanzata, etc.

L'analogia più vicina [11] che si può fare è quella di un dialogo *utente-utente*, nel quale due persone parlano, e fanno continuamente riferimento alla loro vita, alle loro priorità, alla loro conoscenze e credenze, facendo leva su di esse per dare il miglior apporto possibile al dialogo.

In pratica, attraverso la conversazione essi esplorano sé stessi. Questo processo di esplorazione significa quindi investigazione, ricerca di informazioni, comparazione e apprendimento, ed è fondamentale per le collezioni di dati semanticamente ricchi, i quali nascondo informazioni preziose dietro la loro complessità.

In questo contesto ampio e innovativo, questo lavoro di tesi intende fare un significativo passo avanti: viene proposto un modello per compiere concretamente questo tipo di esplorazione su una base di dati. Il modello è sufficientemente generale per comprendere la maggior parte dei modelli di dati e dei linguaggi di interrogazione che sono stati proposti per gestione dei dati negli ultimi anni. Allo stesso tempo, è sufficientemente preciso per fornire una prima formalizzazione del problema e ragionare sulle sfide di ricerca

proposte ai ricercatori in ambito di basi di dati, da questo nuovo paradigma di interazione.

1.2 Tecniche di Exploratory Computing

Sorprendentemente, nonostante anni di ricerca nel campo del data mining, ci sono poche proposte che riguardano il calcolo veloce di parametri statistici dei dati, e che comparino i risultati ottenuti.

Le tecniche di subgroup discovery [6] [9] si sforzano di scoprire sottogruppi di dimensione massima di un insieme di oggetti che sono statisticamente "più interessanti" rispetto ad una proprietà di interesse. La Subgroup Discovery è stata proposta in molti modi differenti, che si differenziano tra loro dal tipo di target variable che usano, dalla misura di qualità che utilizzano e dalla strategia di ricerca implementata. La target variable è un attributo della base di dati che risulta essere interessante per l'analisi.

Un'applicazione tecnica della subgroup discovery include, per esempio, l'analisi di processi di servizio, l'analisi di dati derivanti dai contatori elettrici intelligenti, o l'analisi dei guasti in un processo di produzione. L'ultimo caso è stato analizzato utilizzando diversi algoritmi di subgroup discovery. In particolare, lo scopo principale era quello di identificare i sottoinsiemi (come combinazione di alcuni fattori) che causavano un significativo decremento/aumento nelle tariffe di guasto/riparazione di alcuni prodotti.

Problemi simili nell'industria riguardano il numero di servizi richiesti per un determinato componente, o il numero di chiamate dei clienti al supporto clienti.

Queste applicazioni della subgroup discovery richiedono spesso l'utilizzo di parametri continui, e questo è un problema dato che la maggior parte degli algoritmi di subgroup discovery usano tecniche di discretizzazione standard e

questo provoca una perdita di informazioni. Come conseguenza l'interpretazione dei risultati risulta essere difficile utilizzando strumenti di data mining standard.

Nel corso degli ultimi anni è stato sviluppato un notevole numero di algoritmi per la subgroup discovery. I principali punti deboli di questi algoritmi sono:

- Performance;
- Ridondanza dell'insieme di risultati ritornati dai sottoinsiemi;
- Mancanza di un adeguata comprensione globale dei dati;
- Integrazione e processing di dati eterogenei.

Nelle grandi aree di ricerca, quelle che prendono in considerazione basi di dati multi-relazionali, per esempio quelli utilizzati per i social network, sono richiesti algoritmi efficienti in grado di maneggiare questo tipo di dati.

Inoltre, processare un grande volume di dati con gli algoritmi di subgroup discovery risulta essere problematico. Per risolvere questo problema è quindi necessario utilizzare tecniche di calcolo parallele oppure adottare delle tecniche di campionamento.

Inoltre, l'aggiunta di una conoscenza di fondo in un approccio knowledge-intensive è un prerequisito per l'analisi di basi di dati di grandi dimensioni.

Recentemente è stata proposta in letteratura una nuova tecnica per la database exploration.

L’approccio proposto in questa nuova tecnica si basa sul selezionare due insiemi di tuple corrispondenti ai valori di due attributi della base di dati. Da questi due insiemi, vengono estratti due sottoinsiemi di cardinalità di molto inferiore rispetto a quella degli insiemi di partenza. Per fare questo viene utilizzato un meccanismo di campionamento.

Una volta che il campionamento è terminato, i due sottoinsiemi estratti vengono confrontati attraverso un test di ipotesi, per evidenziate eventuali discrepanze tra i due insiemi [2].

L’analisi della letteratura per le tecniche di data exploration verranno presentate in dettaglio nel Capitolo 3.

1.3 Metodo di exploratory computing proposto

Il metodo di exploratory computing proposto ha come scopo principale quello di estrarre informazioni utili e a priori non esplicite dai dati presenti nelle basi di dati di grandi dimensioni.

In linea con quanto fatto in [1] [2], l’idea di base dell’approccio proposto in questo lavoro di tesi è quella di andare a identificare gli attributi rilevanti per altri attributi della base di dati, e di raggrupparli in sotto gruppi.

La rilevanza degli attributi viene stabilita tramite il calcolo di indicatori statistici, quali entropia, entropia congiunta, entropia condizionata e mutua informazione.

In particolare il parametro statistico su cui si basa la nostra analisi è l’entropia condizionata di un attributo ai valori degli altri attributi della base di dati. Tutte queste entropie condizionate vengono poi confrontate con l’entropia dell’attributo e, a questo punto, vengono confrontati tra loro gli intervalli di confidenza.

Nel caso in cui gli intervalli di confidenza siano sovrapposti, si può concludere che il condizionamento dell’entropia con quel determinato valore dell’altro attributo non porta allo scoperta di informazioni utili.

Nel caso in cui gli intervalli di confidenza siano disgiunti, si può concludere che il condizionamento dell’entropia con quel determinato valore dell’altro attributi porta alla scoperta di nuove informazioni che prima non erano evidenti.

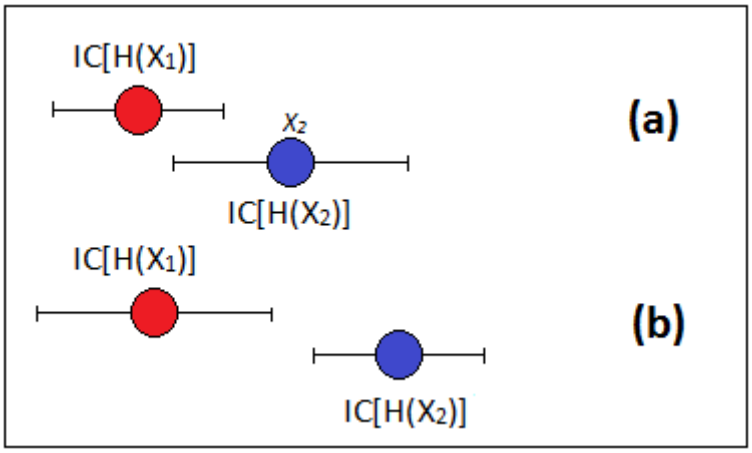

Figura 1.1 Esempio sovrapposizione intervalli di confidenza entropia di due attributi: (a) intervalli sovrapposti, (b) intervalli disgiunti

Nella teoria dell'informazione l'entropia è il valore atteso dell'informazione contenuta in ogni messaggio. In questo caso la parola *messaggio*, sta a indicare un evento, un campione, un carattere o un flusso dati.

L'entropia caratterizza quindi l'incertezza circa la fonte delle informazioni. Quest'ultima è caratterizzata anche dalla distribuzione di probabilità dei campioni ottenuti da essa. L'idea è che meno un evento è probabile, maggiore informazioni porta quando si verifica.

Ad esempio, consideriamo un insieme di gare podistiche miste. Supponiamo che nella maggior parte delle gare la percentuale di donne partecipanti è del 35% circa in ciascuna gara. In una gara tuttavia la percentuale di donne partecipanti risulta essere del 45%. Questo cambiamento di percentuale in questa gara rispetto alle altre nasconde informazioni che possono essere rilevanti come ad esempio: un percorso di gara meno impegnativo, durata del percorso inferiore, etc.

La distribuzione di probabilità degli eventi, congiunta alla quantità di informazione di ogni evento, formano una variabile casuale il cui valor medio è la quantità media di informazione, l'entropia appunto, generata da quella distribuzione.

L’entropia tiene conto della probabilità di osservare uno specifico evento, quindi l’informazione che incapsula è l’informazione riguardante la distribuzione di probabilità sottostante, e non la media degli eventi stessi.

In questo lavoro vengono analizzate le variazioni dell’entropia di un attributo, quando viene condizionato al valore di un altro attributo.

Se dal condizionamento si ottiene una distribuzione statistica dell’attributo considerato, differente rispetto a quella originaria, avremo anche una variazione dell’entropia dell’attributo. Questa variazione ci porta a dire che il condizionamento dell’entropia ha portato alla scoperta di informazioni utili non esplicite.

1.4 Aspetti innovativi e domini di applicazione

Mentre gli algoritmi di subgroup discovery si basano sull'estrazione di regole che possano il più possibile essere rilevanti per l'utente (cercando di coprire nel miglior modo possibile i dati), il metodo proposto in questo lavoro di tesi, si differenzia da questa modalità di data exploration, andando ad operare direttamente sui dati presenti nella base di dati.

Altro elemento fortemente caratteristico dell'approccio proposto è la modalità di esplorazione dei dati, in quanto essa si basa su indicatori statistici mai utilizzati sinora in questo campo quali entropia, entropia condizionata e mutua informazione.

In particolare, in questo lavoro di tesi, viene presentato un metodo di esplorazione dei dati basato sull'analisi dell'entropia condizionata degli attributi ai valori assunti degli altri attributi con il relativo calcolo di sovrapposizione degli intervalli di confidenza in modo tale da poter valutare se il condizionamento ha portato o no a risultati utili. In base al risultato ottenuto attraverso la valutazione degli intervalli di confidenza viene deciso cosa mostrare o non mostrare all'utente.

Il metodo di exploratory computing proposto è stato testato inizialmente su un base di dati di dati sintetici, e successivamente su un base di dati di dati biomedici.

L'analisi effettuata sul sulla base di dati di dati sintetici è risultata necessaria per testare la bontà e la precisione degli indicatori statistici utilizzati per l'esplorazione dei dati. Una volta che i valori ottenuti sulla base di dati sintetica sono risultati coerenti con i valori attesi, abbiamo spostato l'analisi su una base di dati reale contenente dati biomedici.

Struttura della tesi

Nel capitolo 2 verranno presentate le idee e i meccanismi di base dell'exploratory computing.

Nel capitolo 3 verranno presentate le diverse tecniche di database exploration maggiormente utilizzate nell'esplorazione dei dati di grandi dimensioni.

Nel capitolo 4 è presente la descrizione del lavoro svolto. Il capitolo si apre con un commento generale delle tecnica utilizzata per l'exploratory computing. Dopodiché viene presentato il meccanismo per la verifica della rilevanza di un attributo. Vengono inoltre presentati gli indicatori statistici utilizzati e vengono descritti nel dettaglio la tecnica di bootstrap utilizzata e il meccanismo di valutazione di sovrapposizione degli intervalli di confidenza.

Si prosegue poi con la presentazione, nel capitolo 5, dei risultati ottenuti sia sulla base di dati sintetica, sia sul su quella contenente i dei dati biomedici e viene fatto il confronto tra il metodo proposto e un altro algoritmo di exploratory computing esistente. Infine nel capitolo 6 sono racchiuse le riflessioni fatte sul lavoro svolto e i possibili sviluppi futuri.

Capitolo 2

Exploratory computing: idee e meccanismi

2.1 Definizione di database

Un database è una collezione di dati organizzata, in cui le informazioni in essa contenute sono strutturate e collegate tra loro secondo un particolare modello logico, in modo da consentire la gestione efficiente dei dati stessi e l'interfacciamento con le richieste dell'utente attraverso i linguaggi di interrogazione.

In particolare un database relazionale è un database digitale la cui organizzazione è basata sul modello relazione dei dati. Questo modello organizza i dati in una o più tabelle, formate da righe, dette tuple, e da colonne, con una chiave univoca per ogni riga. Generalmente, ogni entità ha una tabella propria nella base di dati, le cui righe corrispondono alle istanze dell'entità e le colonne rappresentano i valori degli attributi descritti da ogni istanza.

Dato uno schema di database $R= \{R_1,\dots,R_k\}$ e un'istanza I di R, un insieme di tuple T è il risultato di ogni interrogazione T^Q su I, cioè $Q(I)$.

L'obiettivo è quello di formalizzare le relazioni tra gli insiemi di tuple identificate dall'utente durante l'esplorazione dei database. Per farlo bisogna ragionare sulle relazioni esistenti sulle interrogazioni.

L'insieme delle tuple $Q(I)$ ritornato da una interrogazione Q su R deriva dall'insieme delle tuple $Q'(I)$ ritornato dalla interrogazione Q' se Q è ottenuto da Q' aggiungendo una o più selezioni, una o più proiezioni, uno o più joint.

Inoltre ogni interrogazione risulta essere indipendente dalle altre, quindi anche i risultati ottenuti da essa sono indipendenti dai risultati ottenuti dalle altre interrogazioni.

Ogni tabella del database rappresenta di per se una interrogazione, e quindi un insieme di tuple.

2.2 Esempio motivante

Per illustrare il processo di esplorazione di una base di dati di grandi dimensioni, è stato utilizzato come esempio, un database di registrazione del fitness tracker *AcmeBand*[1] [2], una smartband da polso che registra continuamente i passi dell'utente, utilizzata per tracciare il sonno di notte.

All'utente che esplora il database, è permesso accedere alle misurazioni di un ampio sottoinsieme degli utenti di *AcmeBand*; in questo esempio, per semplicità è stato assunto che la base di dati utilizzata sia un database relazionale, e che venga utilizzato un linguaggio di interrogazione basato su conjunctive query. Tuttavia, la tecnica discussa di seguito può essere applicata a qualsiasi modello di dati basato su oggetti, attributi e valori degli attributi e object reference: come conseguenza l'approccio proposto può comprendere la maggior parte dei modelli di dati che sono stati proposti negli ultimi anni per modellare i rich data.

Nel caso preso in esame, il database ha la seguente struttura:

- *AcmeUser (id, name, sex, age, cityId)*: tabella contenente i dati degli utenti;
- *Location (id, cityName, state, region)*: tabella contenente i dati circa la locazione degli utenti (l'attributo *region* può assumere i seguenti valori: est, ovest, nord o sud);
- *Activity (id, type, date, start, length, userId)*: tabella per registrare i passi fatti dagli utenti durante le diverse attività (es. camminare, correre, andare in bicicletta etc.);
- *Sleep (id, date, start, length, quality, userId)*: tabella per registrare il sonno degli utenti e la sua qualità (es. sonno profondo, inquieto etc.).

E' possibile notare che il database può avere dimensioni molto grandi, anche se il numero di utenti è basso e l'intervallo di tempo delle attività e del sonno è ridotto.

L'esploratore dei dati intende acquisire delle nuove informazioni riguardanti il livello di fitness e le abitudini del sonno di questa porzione di utenti di *AcmeBand*.

Nell'esempio descritto in [1] e [2], non viene assunta nessuna conoscenza a priori di un linguaggio di interrogazione di database, né la disponibilità di indicatori sintetici precalcolati come quelli che vengono tipicamente utilizzati nelle applicazioni OLAP. Al contrario, il sistema che viene proposto dovrà essere in grado di guidare e supportare i nostri utenti attraverso la loro (casuale) ricerca di informazioni.

2.3 Iniziare la conversazione

L'interazione tra utente e sistema può essere vista come una conversazione tra l'utente che vuole esplorare i dati e il sistema, dove il primo provvede a dare dei suggerimenti iniziali riguardanti i suoi interessi e il secondo suggerisce le "prospettive potenzialmente interessanti" sui dati che possono aiutare a migliorare la ricerca.

Da un punto di vista tecnico, la conversazione è modellata come un reticolo di nodi [1] [2]. Ogni nodo è una vista sopra il database descritta appositamente come una conjunctive query, e quindi rappresenta un sottoinsieme di oggetti del database (tuple, nel nostro caso di database relazionali).

Nell'esempio, viene ipotizzato che l'utente non abbia un obiettivo chiaro, e quindi spetta al sistema suggerire alcuni punti di partenza promettenti per l'esplorazione. Tuttavia, sarà possibile vedere che il modello proposto è in grado di gestire lo scenario alternativo nel quale l'utente inizia la conversazione formulando alcune, seppur vaghe, interrogazioni esplicite.

Dato il database d'esempio, per iniziare la conversazione, il sistema suggerisce alcune features rilevanti. In un contesto relazionale una feature è definita come l'insieme di valori che possono assumere uno o più attributi nell'insieme delle tuple del reticolo di viste, mentre la loro rilevanza è determinata a partire da proprietà statistiche di questi valori.

Come esempio concreto, viene considerato che il possa sistema iniziare l'esplorazione con una delle seguenti possibilità:

- (S_1) "Potrebbe essere interessante esplorare i tipi di attività. Infatti: la corsa è l'attività più frequente (supera il 50%), andare in bicicletta è la meno frequente (meno del 20%)";
- (S_2) "Potrebbe essere interessante esplorare il sesso degli utenti che corrono. Infatti: il 65% dei corridori sono maschi";
- (S_3) "Potrebbe essere interessante esplorare le differenze di distribuzione della lunghezza della corsa tra uomini e donne. Infatti: generalmente gli uomini corrono di più delle donne".

2.4 Modello di rilevanza

Prima di proseguire con l'esempio, è necessario discutere come le features rilevanti vengono estratte.

La nozione di rilevanza è basata sulla distribuzione di frequenza degli attributi nel reticolo di view [1] [2]. Per iniziare la conversazione, il sistema popola il reticolo con un insieme iniziale di viste. Queste corrispondono tipicamente alle tabelle stesse del database, e le unioni di esse ottenute grazie all'utilizzo delle chiavi esterne. In generale, ogni nodo di una vista può essere visto come un concetto della conversazione, descritto da una frase nel linguaggio naturale. Per esempio il nodo corrispondente alla tabella *Activity* rappresenta il concetto *"attività" (*un concetto di base, dal quale il sistema può pro-attivamente iniziare la conversazione), mentre l'unione tra *AcmeUser* e *Location* rappresenta il concetto di *"user location"*, e così via.

Il sistema costruisce istogrammi per gli attributi di queste viste, e cerca quelli che possono avere interesse per l'utente. Nel metodo presentato in[1] e in [2], viene utilizzata una nozione comparativa di rilevanza che proviene dall'informatica teorica, e si cercano gli attributi che mostrano una deviazione significativa rispetto alla sua distribuzione. Da un punto di vista pratico, per ogni attributo vengono identificate un insieme di distribuzioni di riferimento che corrispondono a quelle attese (considerate quindi non interessanti), e si considerano rilevanti solo quegli attributi che hanno una distribuzione differente da quella di riferimento.

Per fare questo si assume la disponibilità di un meccanismo di similarità per le distribuzioni dei valori d, d'.

In questo lavoro di tesi, il meccanismo di similarità utilizzato è basato sul calcolo iterativo dell'entropia degli attributi, utilizzando un meccanismo di bootstrap.

Il meccanismo di bootstrap consente ad ogni sua iterazione di andare a campionare un sottoinsieme di tuple dall'insieme totale di tuple del dataset, sul quale andare poi a calcolare l'entropia degli attributi. Alla fine delle iterazioni viene calcolata media e varianza dell'entropia di ciascun attributo considerato. Una volta ottenuti questi valori si procede al confronto delle entropie per identificare eventuali similarità.

Grazie al meccanismo di bootstrap e al campionamento, siamo in grado di gestire l'analisi di database di grandi dimensioni.

Per un concetto base, la distribuzione di riferimento può essere quella uniforme, sebbene scelte differenti possono essere fatto in relazione alla semantica dell'attributo. Nell' esempio riportato l'attributo *type* della tabella *Activity* viene considerato come potenzialmente rilevante (suggerimento (S_1) sopra) dal momento in cui la sua distribuzione assume significative differenze statistiche, secondo il meccanismo di rilevanza utilizzato, rispetto alla distruzione. Si noti che (S_1) fornisce, insieme a una descrizione della feature rilevante, anche alcuni valori come spiegazione di queste differenze.

Per le viste che sono più di basso livello nel reticolo dei nodi, la nozione di rilevanza è leggermente più sofisticata. Per introdurla, è necessario formalizzare la struttura del reticolo di viste e di insiemi di tuple. Nel database relazionale d'esempio viene fatto tenendo conto delle relazioni tra le conjunctive query: abbiamo detto che l'insieme delle tuple T^Q ritornato da una interrogazione Q su R deriva dall'insieme delle tuple $T^{Q'}$ ritornato dall' interrogazione Q' se ottenuto da Q' aggiungendo una o più selezioni, una o più proiezioni, una o più unioni. Indichiamo questo con $Q \preceq Q'$, e a sua volta $T^Q \preceq T^{Q'}$.

Nell' esempio preso in considerazione, la vista *AcmeUser* ⋈ *Location* deriva dalle viste *AcmeUser* e *Location*.

Allo stesso modo, la vista $\sigma_{type='running'}$ (AcmeUser ⋈ Activity) deriva da *AcmeUser* e *Activity*, ma anche dalla vista $\sigma_{type='running'}$(Activity).

In [1] e [2], viene inoltre fatto notare che il reticolo codifica una tassonomia: ogni volta che l'utente impone una restrizione sulla vista corrente(concetto), cioè equivale in qualche modo a identificare uno dei suoi "sub-concetti". Il reticolo non necessità di essere precalcolato, e sarà tipicamente costruito in modo dinamico.

Dato un insieme di tuple *T*, e un attributo *A* in *T*, è possibile calcolare le distribuzioni dei valori di *A* in *T*. Per formalizzare questa nozione di rilevanza è necessario notare che, ogni volta che una interrogazione *Q* deriva da *Q'*, è possibile tenere traccia della relazione tra gli attributi di *Q(I)* e quelli di *Q'(I)*. Per identificare univocamente gli attributi tra le viste, l'attributo *A* nella tabella *R* viene denotato con il nome *R.A* . Si può quindi dire che l'attributo *R.A* in *Q(I)* matcha l'attributo *R.A* di qualsiasi *Q'(I)* che sia un antenato o un discendente di *Q(I)* nel reticolo delle interrogazioni.

Alla fine di questo ragionamento, viene formalizzata la nozione di rilevanza. E' stato definito che l'attributo *R.A* di un nodo *Q(I)* è rilevante se la sua distribuzione *d* è statisticamente differente dalla distribuzione di qualsiasi attributo corrispondente di nodi antecedenti *Q(I)*.

In [1] e [2], viene fatto notare anche che il caso della radice è un caso speciale: in questo caso si assume la distribuzione di un attributo nel nodo radice come quella di default per l'attributo; tuttavia l'amministratore di sistema può

modificare questo parametro basandosi sulla sua conoscenza del dominio di interesse.

La tecnica di database exploration presentata in questo lavoro di tesi, utilizza la valutazione delle variazioni dei valori delle entropie degli attributi per verificare la rilevanza di un attributo. In particolare le entropie considerate per questa analisi vengono calcolate una su tutte le tuple considerate mentre l'altra sulle tuple condizionate ad un valore di un altro attributo.

2.5 Proseguimento della conversazione

Abbiamo ora tutti gli strumenti per presentare un esempio di conversazione complessa. Assumiamo come primo passo che l'utente è presente con gli item (S_1)-(S_3) presentati precedentemente, e seleziona l'item (S_1) [1] [2]:

(S_1)"Potrebbe essere interessante esplorare i tipi di attività. Infatti: correre è l'attività più frequente (supera il 50%) mentre l'andare in bicicletta è l'attività meno frequente (meno del 20%)".

Questa scelta è interpretata dal sistema come un'interazione con il reticolo delle viste. L'utente ha selezionato la vista *Activity*, e come feature rilevante *type*. Il sistema mostra un sottoinsieme di valori per l'attributo *type*, quelli che giustificano la sua rilevanza. L'utente può scegliere uno o più di questi valori: assumiamo che lui/lei selezioni il valore *running*. Questo è interpretato dal sistema come un interesse verso l'insieme delle tuple che corrispondono all'attività *running*. Come conseguenza, una nuova vista viene generata e aggiunta al reticolo:

$$\sigma_{type=running}(\text{Activity})$$

L'aggiunta di un nuovo nodo al reticolo lancia una nuova interazione, con il sistema che cerca di trovare nuovi suggerimenti rilevanti. Per fare questo può aggiungere ulteriori nodi. In questo caso, trova che una feature rilevante è la

regione dei corridori (utenti che corrono).
Questo comporta il calcolo di una nuova vista:

$$\sigma_{type=running}(\text{Location} \bowtie \text{AcmeUser} \bowtie \text{Activity})$$

Il sistema si rende conto che la distribuzione dell'attributo *regione*, con questa vista, è significativamente differente sia dalla distribuzione uniforme, sia da quelle calcolate precedentemente.

Assumiamo, infatti, che gli utenti sono ugualmente distribuiti tra le varie regioni. Al contrario, i corridori sono specialmente attivi a ovest. Tra le possibili features rilevanti, il sistema suggerirà:

($S_{1.1}$) "Potrebbe essere interessante esplorare la regione degli utenti che praticano la corsa. Infatti: mentre gli utenti sono equamente distribuiti tra le regioni, il 65% degli utenti che praticano la corsa si trovano a ovest e solo il 15% a sud".

L'utente seleziona il suggerimento($S_{1.1}$) e sceglie il valore sud. Il processo viene innescato nuovamente, e il sistema si accorge che:

($S_{1.1.1}$) "Potrebbe essere interessante esplorare il sesso degli utenti che praticano la corsa nella regione sud. Infatti: mentre i valori del sesso sono, di solito, equamente distribuiti tra gli utenti, solo il 10% di essi che praticano la corsa nel sud sono donne".

Bisogna quindi ora calcolare una nuova vista e aggiungere il relativo nodo al reticolo esistente:

$$\sigma_{type\ =running,\ region=south,\ \ sex=female}(\text{Location} \bowtie \text{AcmeUser} \bowtie \text{Activity})$$

Dopo che lui/lei ha selezionato il suggerimento ($S_{1.1.1}$), l'utente sta esplorando esattamente l'insieme delle tuple che corrispondono alle donne che corrono nella regione sud. Può a questo punto decidere se chiedere al sistema ulteriori consigli, navigare i record del database selezionati, o usare una distribuzione dei valori calcolata esternamente per cercare features rilevanti.

Assumiamo, ad esempio, che lui/lei sia interessato/a a studiare la qualità del sonno, e esegue quindi il download da un sito web di un foglio Excel il quale afferma che il 60% degli utenti si lamentano della qualità del loro sonno. Quando questi dati vengono importati nel sistema, quest'ultimo suggerisce, inaspettatamente, che l'85% il sonno delle donne del sud che pratica la corsa è di buona qualità.

Avendo tratto nuova conoscenza dai dati, l'utente è soddisfatto.

Da un punto di vista tecnico, il sistema ha costruito un complesso reticolo di viste, e la conversazione consiste in un cammino in questo reticolo.

Capitolo 3

Analisi della letteratura

3.1 Subgroup Discovery

La subgroup discovery è una tecnica di data mining utilizzata per scoprire relazioni interessanti tra oggetti differenti dello stesso insieme, rispetto ad una proprietà specifica verso cui si ha interesse.

I pattern estratti sono rappresentati tipicamente sotto forma di regole e vengono chiamati subgroups.

3.1.1 Definizione di Subgroup Discovery

Nella subgroup discovery, si assume che vengono date una popolazione di individui (che possono essere oggetti, clienti, ...) e una proprietà appartenente a questi individui, verso cui si ha dell'interesse.

L'obiettivo della subgroup discovery è quello di scoprire i sottoinsiemi della popolazione che sono "più interessanti" statisticamente, cioè mostrano delle caratteristiche statistiche più inusuali rispetto alla proprietà di interesse.

La subgroup discovery tenta di scoprire delle relazioni tra le differenti proprietà o variabili dell'insieme rispetto ad una variabile obiettivo. Queste relazioni sono descritte in forma di singole regole.

Una regola *R*, può essere formalmente come:

$$R: Cond \rightarrow Target_{Value}$$

dove $Target_{Value}$ è un valore della variabile di interesse per il processo di subgroup discovery, e *Cond* è una congiunzione di caratteristiche (attributo –

coppia valori) che è in grado di descrivere una distribuzione statistica inusuale rispetto al $Target_{Value}$.

La subgroup discovery si colloca a metà strada tra la predictive e la descriptive induction, e il suo compito è quello di generare in modo unico e interpretabile i sottogruppi per descrivere le relazioni tra le variabili e un determinato valore della variabile obiettivo. L'algoritmo genera un subgroup per ogni valore assunto dalla target variable, quindi deve essere fatta un'esecuzione per ogni valore della variabile.

I seguenti tre fattori sono fondamentali per la subgroup discovery:

- Dimensioni limitate consentono una comprensione migliore;
- Un'ampia copertura delle regole significa un supporto maggiore;
- Un alto livello di importanza significa che le regole descrivono dei sottogruppi che presentano delle caratteristiche distribuzionali differenti in confronto all'intera popolazione.

3.1.2 Elementi principali di un algoritmo di Subgroup Discovery

Gli elementi principali che bisogna considerare in un algoritmo di subgroup discovery sono:

- Tipo della target variable. Possono essere trovati tipi differenti per la variabile: binaria, nominale o numerica. Per ogni tipo, devono essere effettuate delle analisi differenti:

    - Analisi binarie: le variabili possono assumere solo due valori (vero o falso), e il processo si basa sul provvedere sottogruppi interessanti per ogni possibile valore;
    - Analisi nominali: la target variable può assumere un numero non determinato di valori, ma la filosofia delle analisi è similare a quelle binarie, cerca sottogruppi per ogni valore;

    - Analisi numeriche: questo tipo di analisi sono le più complesse perché la variabile può essere rappresentata con diversi formati.

- Description language: la rappresentazione dei sottogruppi deve essere adatta per ottenere delle regole interessanti.

- Misure di qualità: queste sono un fattore chiave per l'estrazione della conoscenza perché l'interesse dipende esclusivamente da esse. Si dividono in:

    - Misure di complessità: sono legate all'interpretabilità dei sottogruppi, ossia alla semplicità della conoscenza estratta da essi;
    - Misure di generalità: sono usate per quantificare la qualità delle singole regole in accordo con i pattern individuali di interesse che coprono;
    - Misure di precisione: mostrano la precisione dei sottogruppi e sono molto usate nell'estrazione delle association rules e nella classificazione;
    - Misure di interesse: sono utilizzate per selezionare e ordinare i patterns in accordo con il loro potenziale interesse per l'utente;
    - Misure ibride. In questo gruppo troviamo un grande numero di misure di qualità perché la subgroup discovery cerca di ottenere un compromesso tra generalità, interesse e precisione del risultato ottenuto.

- Strategia di ricerca: questo elemento è molto importante, dato che lo spazio di ricerca ha una relazione esponenziale al numero di caratteristiche e valori considerati. Al momento vengono utilizzate numerose strategie di ricerca come, ad esempio, beam search, evolutionary algorithms, etc.

3.1.3 Classi di algoritmi di Subgroup Discovery

Nel corso degli anni sono stati sviluppati numerosi algoritmi che usano la subgroup discovery.

Per classificare questi algoritmi, si può fare una distinzione tra:

- Estensioni dei classification algorithms;
- Estensioni degli association algorithms;
- Evolutionary fuzzy systems.

Nella seguente tabella sono riassunti i principali algoritmi per la subgroup discovery in base alla precedente classificazione.

Estensioni dei classification algorithms	***Estensioni degli association algorithms***	***Evolutionary algorithms***
EXPLORA	APRIORI-SD	SDIGA
MIDOS	SD-MAP	MESDIF
CN2-SD	Merge-SD	NMEEF-SD

Tabella 3.1 Classificazione degli algoritmi di subgroup discovery esistenti

Per quanto riguarda la prima categoria di algoritmi possiamo distinguere tra gli algoritmi pionieri, cioè quelli sviluppati per primi, e di cui fanno parte EXPLORA e MIDOS, e quelli basati sui classification algorithms, di cui fa parte l'algoritmo CN2-SD. I primi sono estensioni dei classification algorithms e usano i decision trees[2]. Essi possono utilizzare due diverse strategie per la ricerca, esaustiva o euristica, e molte misure di qualità per valutare la qualità dei subgroups. I secondi invece sono stati sviluppati compiendo degli adattamenti agli algoritmi di classification rule learning.

La seconda categoria di algoritmi deriva dagli association rule algorithm, i quali tentano di ottenere relazioni tra variabili dello stesso data set. In questo caso,

[2] Decision trees: sono strumenti di supporto alle decisioni che usano grafi a forma di albero o modelli di decisioni e delle loro possibili conseguenze. Sono comunemente utilizzati in ricerca operativa, specialmente in analisi decisionale, per aiutare ad identificare una strategia piuttosto che il raggiungimento di un obiettivo.

alcune variabili possono comparire sia nell'antecedente sia nel conseguente della regola. Questo nella subgroup discovery non è possibile, in quanto il conseguente della regola, costituito dalla proprietà di interesse, è prefissato. Bisogna quindi applicare degli adattamenti agli association rule algorithm per utilizzarli per la subgroup discovery. Questi adattamenti variano da algoritmo ad algoritmo.

Gli evolutionary algorithms imitano i principi dell'evoluzione naturale per formare dei processi di ricerca. I principali evolutionary algorithms utilizzati sono gli algoritmi genetici. L'euristica utilizzata in questi algoritmi è detta fitness function, la quale determina quali regole verranno selezionate per formare la nuova conoscenza.

Di seguito verranno presentati tre degli algoritmi più utilizzati nel processo di subgroup discovery.

3.2 Algoritmo CN2-SD

Il CN2-SD [4] è un algoritmo che nasce dalla modifica del CN2 rule learning algorithm.

L'approccio proposto dal CN2-SD algorithm esegue la subgroup discovery attraverso le seguenti modifiche del CN2:

- Sostituisce la ricerca euristica accuracy-based con una nuova weighted relative accuracy heuristic che è un compromesso tra la generalità e l'accuratezza della regola;
- Incorpora i pesi nel covering algorithm;
- Incorpora i pesi nell'euristica utilizzata;
- Usa una classificazione probabilistica basata sulla class distribution degli esempi ricoperti dalle singole regole.

Se usati per la subgroup discovery, uno dei problemi dei rule learner standard, come il CN2, è l'uso del covering algorithm per la costruzione dell'insieme delle regole. La mancanza principale del covering algorithm è che solo le prime regole potrebbero essere di interesse per la descrizione dei subgroups con

sufficiente significato. Nelle iterazioni seguenti del covering algorithm, le regole sono indotte da sottoinsiemi contenenti solamente esempi positivi che non sono ricoperti dalle regole indotte in precedenza. Questo fatto influenza in modo inopportuno il processo di subgroup discovery.

Come rimedio a questo problema viene utilizzato un weighted covering algorithm, in cui le regole indotte dalle ultime iterazioni dell'algoritmo rappresentano anch'esse subgroup sufficientemente grandi ed interessanti della popolazione.

Il CN2-SD modifica l'algoritmo classico in modo tale che gli esempi positivi già ricoperti non vengano cancellati dal training set corrente. Invece, ad ogni iterazione, l'algoritmo salva insieme ad ogni esempio un contatore che indica quante volte l'esempio è stato ricoperto sino a quel momento. I pesi derivati da questo contatore appariranno poi nel calcolo di *WRAcc*. I pesi iniziali di tutti gli esempi positivi e_j sono uguali a 1, $w(e_j, 0) = 1$, questo significa che non sono ancora stati coperti da nessuna regola. I pesi più bassi, compresi tra 0 e 1, invece, indicano che l'esempio è già stato coperto da una o più regole. Ne consegue che, gli esempi coperti da una o più regole decrementano il loro peso in modo tale che gli esempi non ancora coperti, i cui pesi non sono ancora stati decrementati, abbiano maggiori chance di essere coperti nella prossima iterazione dell'algoritmo.

E' possibile adottare due schemi di pesatura delle regole:

- Multiplicative weights: dato un parametro $0<\gamma<1$, i pesi degli esempi positivi coperti decresce come segue:

  $$w(e_j, i) = \gamma^i$$

  dove $w(e_j, i)$ è il peso dell'esempio e_j coperto da i regole.
- Additive weights: in questo caso i pesi degli esempi positivi coperti decrescono in accordo con la seguente formula:

  $$w(e_j, i) = \frac{1}{1+i}$$

  Alla prima iterazione tutti gli esempi hanno lo stesso peso $w(e_j, 0)=1$ mentre nelle iterazioni seguenti il contributo degli esempi è inversamente proporzionale alla loro copertura delle regole precedentemente indotte.

Procedure UnsortedCN2 (*all_Examples*, *classes*)

1. *Rules_Set* ← ∅
2. For each *class* in *classes*
   3. Generate *rules* with OneClassCN2(*all_Examples*, *class*)
   4. *Rules_Set* ← *Rules_Set* ⋃ *rules*
5. Return *Rules_Set*

Procedure OneClassCN2 (*examples*, *class*)

1. *rules* ← ∅
2. Repeat
   3. *Best_Condition* ← *Find_Best_Condition*(*examples*, *class*)
   4. If (*Best_Condition* is not null) then
      5. Add rule "if *Best_Condition* then *class*" to *rules* and removes all examples that belongs to class *class* covered by *Best_Condition*
6. Until *Best_Condition* is null
7. Return *rule*

Figura 3.1 Pseudocodice algoritmo CN2-SD

3.2.1 Misura di qualità

La funzione euristica utilizzata, come detto in precedenza, è la ***WRAcc*** (weighted relative accuracy). Nel calcolo della *WRAcc* tutte le probabilità sono calcolate a partire dalle frequenze relative. Il peso di un esempio misura quanto è importante coprire quell'esempio nella prossima iterazione. La funzione euristica utilizzata risulta quindi essere la seguente:

$$WRAcc\ (Class \leftarrow Cond) = \frac{n'(Cond)}{N'} * \left(\frac{n'(Class.Cond)}{N'} - \frac{n'(Class)}{n'(Cond)} \right)$$

In questa equazione, *N'* è la somma di tutti i pesi di tutti gli esempi, *n'(Cond)* è la somma dei pesi di tutti gli esempi coperti, e *n'(Class.Cond)* è la somma dei pesi di tutti gli esempi coperti correttamente. Per aggiungere una regola all'insieme delle regole generate viene scelta quella con il *WRAcc* massimo, tra quelle non ancora presenti nell'insieme prodotto fino a quel momento. Quindi tutte le regole nell'insieme finale sono distinte, senza duplicati.

3.2.2 Probabilistic classification dell'algoritmo CN2-SD

In generale le regole indotte possono essere trattate come ordinate o non ordinate. Nel caso di regole ordinate possono essere interpretate come una decision list: quando si classifica un nuovo esempio, le regole vengono sequenzialmente estratte e la prima regola che copre l'esempio viene usata.

Nel caso invece di un insieme non ordinato di regole, la distribuzione degli esempi ricoperti tra le classi è attaccata ad ogni regola. Regole nella forma:

If ***Cond*** then ***Class*** [***ClassDistribution***]

sono indotte, dove i numeri nella lista *ClassDistribution* indicano, per ogni singola classe, la percentuale degli esempi di quella classe coperti dalla regola.

Quando viene classificato un nuovo esempio tutte le regole vengono provate e quelle che coprono l'esempio sono collezionate. Quando si verifica uno scontro, cioè più di una regola copre un esempio, viene utilizzato un meccanismo di voto per ottenere la predizione finale: le class distribution assegnate alle regole sono considerate per determinare la classe più probabile.

Se nessuna regola copre l'esempio, viene invocata una regola di default.

3.3 Algoritmo APRIORI-SD

L'algoritmo APRIORI-SD [3] è stato sviluppato adattando l'association rule learning alla subgroup discovery. Questo è stato possibile costruendo un classification rule learner(APRIORI-C), potenziato con un nuovo meccanismo di post-processing, una nuova misura di qualità per le regole indotte e usando una classificazione probabilistica delle istanze.

Per utilizzare l'APRIORI-SD per la subgroup discovery le modifiche che devono essere fatte nell'algoritmo APRIORI-C sono le seguenti:

- Uso di uno schema pesato nel post-processing delle regole;
- Usa della weighted relative accuracy come nuova misura di qualità delle regole nel passo di post-processing quando vengono selezionate le migliori regole;
- Classificazione probabilistica basata sulla distribuzione delle classi degli esempi coperti da regole individuali;

La post-processing procedure utilizzata dall'APRIORI-SD può essere implementata come segue:

Repeat

- Ordina le regole dalla migliore alla peggiore in accordo con la misura di qualità adottata (weighted relative accuracy)
- Decrementa i pesi degli esempi coperti.

Until

- Tutti gli esempi sono stati coperti o non ci sono nuove regole

Lo schema utilizzato per gestire i pesi tratta gli esempi in modo tale che gli esempi positivi che vengono coperti non vengano eliminati quando la migliore regola viene selezionata dalla procedura di post-processing. Invece, ogni volta che una regola viene selezionata, l'algoritmo salva, insieme ad ogni esempio, un contatore che indica quante volte l'esempio è stato coperto da una regola fino a quel momento.

I pesi iniziali di tutti gli esempi positivi *e* sono uguali a *1*, $w(e_j, 0) = 1$. Questo sta a indicare che l'esempio non è ancora stato coperto da nessuna regola, pesi minori invece stanno a indicare che l'esempio è già stato coperto da una o più regole.

I pesi degli esempi positivi che vengono coperti dalle regole sono decrementati seguendo la formula:

$$w(e_j, i) = \frac{1}{1+i}$$

Alla prima iterazione dell'algoritmo tutti gli esempi positivi contribuiscono con lo stesso peso, mentre nelle iterazioni successive i contributi degli esempi sono inversamente proporzionali alla copertura che hanno dalle regole selezionate fino a quel momento. In questo modo gli esempi coperti da una o più regole selezionate decrementano il loro peso in modo tale che ad ogni iterazione successiva possano essere selezionate regole che coprono esempi non ancora coperti.

1. algorithm $APRIORI - SD(Examples, Classes, minSup, minConf, k)$
2. Ruleset= $APRIORI - C(Examples, Classes, minSup, minConf)$
   set all example weights of Examples to 1)
3. Majority= the majority class in Examples
4. Resultset= {}
5. Repeat
   6. BestRule= rule with the highest weighted relative accuracy in Ruleset.
   7. Resultset= Resultset $\cup$ BestRule
   8. Ruleset= Ruleset \ decrease the weights of examples covered by BestRule remove from Examples the examples covered more than k-times
9. until Examples={} or Ruleset={}
10. return Resultset= Resultset $\cup$ true $\rightarrow$ Majority

Figura 3.2 Pseudocodice algoritmo APRIORI-SD

3.3.1 Misura di qualità

La misura di qualità utilizzata nell'algoritmo è la ***WRAcc*** (weighted relative accuracy) che è definita come segue

$$WRAcc(X \rightarrow Y) = \frac{n'(X)}{N'} * \left(\frac{N'(Y\ X)}{n'(X)} - \frac{n'(Y)}{N'} \right)$$

dove *N'* è la somma dei pesi di tutti gli esempi, *n'(X)* è la somma dei pesi di tutti gli esempi coperti dalle regole, e *n'(Y X)* è la somma dei pesi di tutti gli esempi coperti dalle regole correttamente.

3.3.2 Probabilistic classification dell'algoritmo APRIORI-SD

In generale le regole indotte possono essere trattate come ordinate o non ordinate. Nel caso di regole ordinate possono essere interpretate come una lista di IF-THEN-ELSE: quando si classifica un nuovo esempio, le regole vengono sequenzialmente estratte e la prima regola che copre l'esempio viene usata.

Nel caso invece di un insieme non ordinato di regole, la distribuzione degli esempi ricoperti tra le classi è attaccata ad ogni regola. Regole nella forma:

$$X \rightarrow Y[ClassDistribution]$$

sono indotte, dove i numeri nella lista *ClassDistribution* indicano, per ogni singola classe, la percentuale degli esempi di quella classe coperti dalla regola.

Quando viene classificato un nuovo esempio tutte le regole vengono provate e quelle che coprono l'esempio sono collezionate. Quando si verifica uno scontro, cioè più di una regola copre un esempio, viene utilizzato un meccanismo di voto per ottenere la predizione finale: le class distribution assegnate alle regole sono considerate per determinare la classe più probabile.

Se nessuna regola copre l'esempio, viene invocata una regola di default.

3.4 Algoritmo SD-MAP

SD-Map [5] è un metodo di ricerca esaustiva, dipendente da una soglia minima di sopportazione: se il minimo viene settato a zero, l'algoritmo esegue la ricerca esaustiva su l'intero spazio di ricerca.

L'algoritmo SD-Map nasce dall'estensione e dall'adattamento del metodo FP-growth per il processo di subgroup discovery. L'FP-growth algorithm è un approccio efficiente per l'estrazione di pattern frequenti, e il suo funzionamento è simile a quello dell'algoritmo APRIORI. L'FP-growth opera su un insieme di item ottenuti tramite un insieme di selettori. Il miglioramento più importante dell'FP-growth rispetto all'APRIORI è la caratteristica di eliminare le scansioni multiple del database per testare ogni pattern frequente. Questa operazione viene sostituita da una tecnica *divide-and-conquer*[3] ricorsiva.

Come una struttura dati speciale, l'albero che contiene i pattern frequenti(FP-tree) è implementato come una struttura ad albero prefissa estesa che salva le informazioni riguardanti i patterns frequenti. Ogni nodo dell'albero è una tupla (selettore, contatore, node-link): il contatore misura il numero di volte che il corrispondente selettore è contenuto nei casi raggiunti da un determinato percorso, e il node-links collega i nodi dell'FP-tree con lo stesso selettore.

La costruzione dell'FP-tree prevede solo due passaggi attraverso l'insieme dei casi: durante il primo passaggio viene scansionato il caso base collezionando l'insieme *L* dei selettori più frequenti, il quale viene ordinato in ordine di supporto decrescente. Nel secondo passaggio sono inseriti nell'albero in accordo con l'ordine di *L*, i selettori le cui chance di condividere dei prefissi comuni sono incrementate. Dopo la costruzione, i percorsi dell'albero contengono le informazioni riguardo i selettori e i pattern più frequenti, tipicamente in un formato più compatto rispetto al caso base.

Per determinare l'insieme dei pattern frequenti, l'algoritmo FP-growth applica un metodo di divide-and-conquer, prima estraendo i frequent pattern

[3] Divide-and-Conquer: in informatica, è un paradigma utilizzato nel design degli algoritmi basato sulla ricorsione multi-ramificata. Un algoritmo divide-and-conquer lavora dividendo in problema in due o più sotto problemi dello stesso tipo, che possono essere risolti direttamente.

contenenti un selettore e poi estraendo ricorsivamente i pattern condizionati dall'occorrenza di un 1-selector.

Per il passo ricorsivo, viene costruito un FP-tree condizionale, a partire da un pattern condizionale base di un selettore frequente dell'FP-tree e i suoi nodi corrispondenti dell'albero. Il pattern condizionale base consiste di tutti i percorsi prefissi di tale nodo *v*, cioè, considerando tutti i percorsi *P* a cui partecipa il nodo *v*. Dato il pattern condizionale base, viene generato un FP-tree più piccolo, l'FP-tree condizionale di *v* conta i nodi. Se l'FP-tree è formato da un solo percorso, allora viene generato il frequent pattern considerando tutte le combinazioni dei nodi presenti nel percorso. Altrimenti, il processo viene ripetuto ricorsivamente.

L'algoritmo SD-Map adotta un approccio ingenuo per la subgroup discovery: utilizza il metodo FP-growth per collezionare i patterns frequenti, dopodiché testa i pattern con una misura di qualità.

Require: target variable t, quality function q, set of selectors E (search space)

1: Scan 1 – Collect the frequent set of selectors, and construct the frequent node list L for the main FP-tree:
   1. For each case for which the target variable has a defined value, count the (tp_e, fp_e) for each selector $e \in E$.
   2. Prune all selectors which are below a minimum support $\mathcal{T}_{Supp}$, i.e., $tp(e) = frequency(e \wedge t) < \mathcal{T}_{Supp}$.
   3. Using the unpruned selectors $e \in E$, construct and sort the frequent node list L in support/frequency descending order.

2: Scan 2 – Build the main FP-tree:
For each node contained in the frequent node list, insert the node into the FP-tree (according to the order of L), if observed in a case, and count the number of (tp, fp) for each node

3: Scan 3 – Build the Missing-FP-tree (This step can also be integrated as a logical step into scan 2):
   1. For all frequent attributes, i.e., the attributes contained in the frequent nodes L of the FP-tree, generate a node denoting the missing value for this attribute.
   2. Then, construct the Missing-FP-tree, counting the (tp, fp) for each (attribute) node.

4: Perform the adapted FP-growth method to generate subgroup patterns:
5: **repeat**
6: **for** each subgroup s_i that denotes a frequent subgroup pattern **do**
7: Compute the adjusted population counts TP', FP' as shown in Equation 2
8: Given the parameters tp, fp, TP', FP' compute the subgroup quality $q(s_i)$ using a quality function q
9: **until** FP-growth is finished
10: Post-process the set of the obtained subgroup patterns, e.g., return the k best subgroups, or return the subgroup set $S = \{s \mid q(s) \geq q_{min}\}$, for a minimal quality threshold $q_{min} \in \mathbb{R}$ (This step can also be integrated as a logical filtering step in the discovery loop (lines 5-9)).

Figura 3.3 Pseudocodice algoritmo SD-Map

3.4.1 Misura di qualità

Per il calcolo della qualità della subgroup discovery vengono utilizzati quattro parametri:

- True positive(t_p), casi contenenti la variabile obiettivo *t* nel subgroup *s* dato;
- False positive(f_p), casi che non contengono la variabile obiettivo *t* nel subgroup *s*;
- *TP* e *FP*, riguardano la variabile *t* nella popolazione totale di dimensione *N*.

Applicando il metodo FP-growth, si possono calcolare i parametri come segue:

- $n = count(s)$
- $t_p = support(s) = count(s \wedge t)$
- $f_p = n - t_p$
- $p = \frac{t_p}{(t_p + f_p)}$
- $TP = count(t)$
- $FP = N - TP$
- $p_0 = \frac{TP}{TP+FP}$

Tuttavia, persiste un problema molto diffuso per il data mining: il problema dei valori mancanti.

Se mancano dei valori nel caso base, allora non tutti i casi avranno un valore definito per ogni attributo. Se il valore mancante non può essere eliminato, deve essere considerato per il calcolo della qualità del subgroup. E' necessario quindi aggiustare il conteggio della popolazione identificando i casi dove le variabili o il target non sono definiti.

Per migliorare questa situazione, e per ridurre lo spazio dell'FP-tree costruito vengono considerate le seguenti osservazioni:

- Per stimare la qualità della subgroup discovery è necessario solamente determinare i quattro parametri principale t_p, f_p, *TP* e *FP* con un potenziale adattamento per i valori mancanti. Dopodiché gli altri parametri possono essere derivati: *n=tp+fp, N=TP+FP*

- Per la subgroup discovery, il concetto di interesse è fissato: la target variable. Quindi, i parametri necessari descritti sopra possono essere calcolati direttamente.

Se il conteggio di t_p e f_p è salvato direttamente nei nodi dell'albero FP, è possibile calcolare la qualità dei subgroups direttamente mentre si generano i frequent patterns: se la target variable è presente nel nodo t_p viene incrementato, altrimenti viene incrementato f_p. Questo calcolo viene fatto solamente per i nodi in cui la target variable ha un valore definito.

Per gestire i valori mancanti, l'SD-Map prevede la costruzione di una seconda struttura ad albero, denominata Missing-FP-tree. L'albero FP per contare i valori mancanti, può essere ristretto all'insieme degli attributi frequenti del FP-tree principale, in quanto solo quelli possono formare dei subgroups che poi verranno testati per verificare la presenza di valori mancanti. Dopodiché il Missing-FP-tree deve essere valutato per ottenere il corrispondente conteggio dei valori mancanti. Tenendo conto dei valori mancanti si ha la necessità di aggiustare il numero di *TP*, *FP* corrispondente alla popolazione, in quanto i conteggi con per il subgroup FP-tree sono ottenuti per i casi dove il target è definito, e la subgroup description prende in considerazione solamente i casi in cui il selettore è definito.

Usando il Missing-FP-tree, si possono identificare le situazione dove qualche attributo contenuto nella subgroup description non è definito:

Per una subgroup description $sd = (e_1 \wedge e_2 \wedge \ldots \wedge e_n), e_i = (a_i, V_i), V_i \subseteq dom(a_i)$ si calcolano i conteggi dei valori mancanti $missing(a_1 \vee a_2 \vee \ldots \vee a_n)$ per l'insieme degli attributi $M = \{a_1, a_2, \ldots, a_n\}$. Questo può essere ottenuto applicando la seguente trasformazione

$$missing(a_1 \vee \ldots \vee a_n) = \sum_{i=1}^{n} missing(a_i) - \sum_{m \in 2^M} missing(m)$$

dove $|m| \geq 2$. Quindi, per ottenere l'aggiustamento dei valori mancanti rispetto all'insieme M contenente gli attributi del subgroup, è necessario aggiungere le voci degli header dei nodi del Missing-FP-tree corrispondenti agli attributi individuali, e eliminare le voci di ogni percorso suffisso terminante in un elemento di M contenente almeno un altro elemento di M.

Considerando *(t_p, f_p)* contenuti nel Missing-FP-tree si ottiene il numero di *($TP_{missing}$, $FP_{missing}$)* dove i casi non possono essere valutati statisticamente in quanto alcune delle subgroup variables non sono definite, cioè almeno un attributo contenuto in *M* è mancante. Per sistemare i conteggi della popolazione, si utilizzano le seguenti formule:

$$TP' = TP - TP_{missing}, \qquad FP' = FP - FP_{missing}.$$

Dopodiché è possibile calcolare la qualità del subgroup basandosi sui parametri t_p, f_p, *TP'*, *FP'*.

SD-Map include un passo di post-processing per la selezione e la gestione delle potenziali ridondanze dei subgroup ottenuti. Solitamente, l'utente è interessato solo ai *k* migliori subgroups e non vuole vedere visualizzati gli altri. Per fare questo è possibile selezionare i *k* migliori subgroups basandosi sulla misura di qualità o, alternativamente, selezionare i *k* gruppi che hanno qualità migliore di una soglia prefissata.

3.5 Confronto tra CN2-SD, APRIORI-SD e SD-MAP

Confrontando gli algoritmi presentati si può dedurre che l'APRIORI-SD produce un insieme di regole più piccolo, in cui le singole regole hanno una maggior copertura, ma un minor importanza delle regole ottenute con il CN2-SD.

L'algoritmo SD-Map risulta avere una maggior efficienza in termini di tempo rispetto all'algoritmo APRIORI-SD.

L'algoritmo SD-Map è l'unico algoritmo esistente per la subgroup discovery che gestisce il problema dei valori mancanti all'interno del dataset. Questo approccio è fondamentale per dataset che contengono un elevato numero di dati mancanti. Senza applicare questa tecnica non è possibile effettuare una valutazione efficiente della qualità del subgroup, in quanto un numero troppo grande di subgroup andrebbe considerato nel passo di post-processing, oppure subgroups con una qualità più alta potrebbero essere esclusi erroneamente.

L'SD-Map algorithm presenta una scalabilità migliore rispetto all'APRIORI-SD, per il quale il tempo di esecuzione cresce esponenzialmente per una crescita esponenziale dello spazio di ricerca.

Il tempo di esecuzione dell'SD-Map, invece, cresce in modo più conservativo. Questo è dovuto al fatto che l'APRIORI-SD applica la generazione dei candidati e la strategia di test basandosi sulle scansioni multiple del data set e quindi sulle dimensioni del data set. L'SD-Map algorithm beneficia dal suo approccio divide-and-conquer adattato dal metodo FP-growth eliminando un elevato numero di passi di generazione e testing.

Algoritmo	***Tipo della target variable***	***Description Language***	***Misura di qualità***	***Strategia di ricerca***
CN2-SD	Categorical	Conjunctions of pairs. Operators =, <, > and ≠	Unusualness	Beam search
APRIORI-SD	Categorical	Conjunctions of pairs Attribute-value. Operators =, < and >	Unusualness	Beam search with minimum support pruning
SD-Map	Binary	Conjunctive languages with internal disjunctions. Operator =	Unusualness, Binomial test	Exhaustive search with minimum support pruning

Tabella 3.2 Caratteristiche degli algoritmi CN2-SD, APRIORI-SD e SD-Map

3.6 Data mining

Il data mining è il processo computazionale per inferire pattern in database di grandi dimensioni utilizzando metodi ibridi che coinvolgono tecniche di intelligenza artificiale, machine learning, statistica. Lo scopo principale del processo di data mining è quello di estrarre informazioni implicite da un data set e trasformare queste informazioni in conoscenza utile.

Il compito del data mining è quello di effettuare un'analisi automatica o semi-automatica di grandi quantità di dati per estrarre pattern interessanti che in precedenza erano sconosciuti, quali gruppi di records (cluster analysis), record inusuali (anomaly detection) e dipendenze (association rule mining). Questo, tipicamente, coinvolge l'uso di tecniche proposte nell'ambito delle basi di dati, come ad esempio gli indici spaziali. Questi pattern possono essere poi visti come un sommario dei dati in input, e possono essere utilizzati per ulteriori analisi più approfondite.

Le tecniche di data mining possono essere applicate da due prospettive differenti:

- **Predictive induction**, il cui obiettivo è quello della scoperta della conoscenza per la classificazione della predizione. Le principali tecniche che ricorrono a questa tecnica sono classification, regression e temporal series.
- **Descriptive induction**, il cui obiettivo principale è l'estrazione di nuova conoscenza interessante dai dati. Le principali tecniche che fanno parte di questa categoria sono: association rules, summarisation, subgroup discovery.

Di particolare importanza per il caso preso in esame è la subgroup discovery che verrà descritta di seguito.

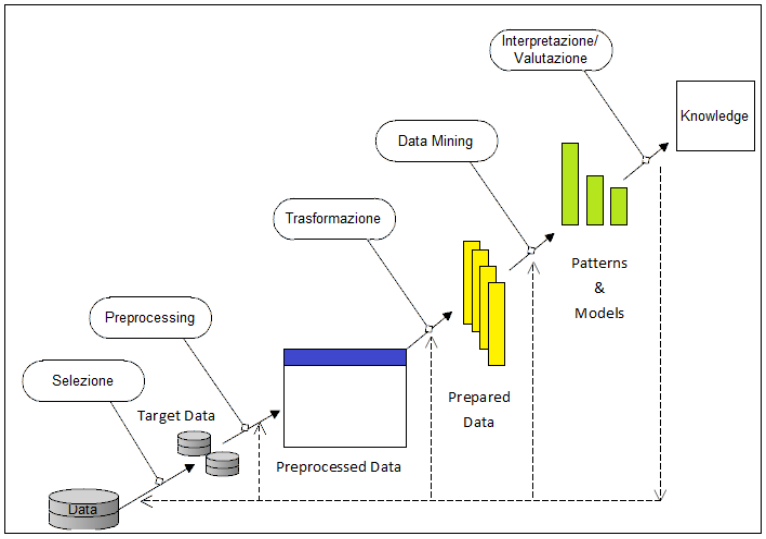

Figura 3.4 Data Mining Process

3.7 Exploratory Computing attraverso tecniche statistiche

In letteratura è stato recentemente presentato [1] un algoritmo statistico per misurare la differenza tra due insiemi di tuple T^{Q1} e T^{Q2}, i quali hanno in comune una feature al fine di calcolare la loro relativa rilevanza. I due insiemi di tuple possono avere origini differenti, non necessariamente devono essere il risultato di interrogazioni che sono una il raffinamento dell'altra. Questa generalità è necessaria per far fronte alle diverse necessità descritte nei capitoli precedenti, cioè l'acquisizione di un concetto di rilevanza flessibile che possa essere usato per le diverse aspettative degli utenti.

Il metodo si basa su un insieme di test di ipotesi che operano su sottoinsiemi dell'insieme originale di tuple, i quali vengono estratti randomicamente.

La principale intuizione è che i test di ipotesi possono essere condotti incrementalmente in modo tale da aumentare la scalabilità, mentre allo stesso tempo si tiene sotto controllo i risultati falsi positivi attraverso la media della Bonferroni correction standard.

Il processo è composto da tre fasi che sono condotte iterativamente:

- **Fase 1 – Sampling**: vengono campionati gli insiemi di tuple T^{Q1} e T^{Q2} e vengono estratti i sottoinsiemi q_1 e q_2 di cardinalità molto inferiore rispetto a T^{Q1} e T^{Q2}. Questo può essere fatto utilizzando diverse strategie di campionamento: sequenziale, random o ibrido.
- **Fase 2 – Comparison:** assumiamo che X_1 e X_2 siano le proiezioni di q_1 e q_2 sopra un specifico attributo(feature). I dati in X_1 e X_2 possono essere sia numerici che categorici. La fase di comparazione permette di valutare la discrepanza tra X_1 e X_2 attraverso test d'ipotesi, nella forma $test_i(X_1, X_2)$. Esempi di test possono essere [12]: (i) il *two-sided t test*, valuta le variazioni del valor medio di due sottoinsiemi gaussiani; (ii) il *two-sided Wilcoxon rank sum*, valuta le variazioni del valor medio di due sottoinsieme senza distribuzione; (iii) il *one/two-sample Chi-square test*, valuta la distribuzione di sottoinsiemi con valori discreti rispetto a una distribuzione di riferimento o valuta le variazioni in proporzione tra due sottoinsiemi con valori discreti; (iv) il *one/two-sample Kolmogorov-Smirnov test*, per valutare se un sottoinsieme deriva da una funzione di probabilità continua di riferimento o per valutare se due sottoinsiemi sono stati generati da due funzioni di probabilità continue differenti.
- **Fase 3 – Query ranking:** la procedura descritta sopra può essere applicata a differenti coppie di insiemi di tuple. La differenza tra le loro distribuzioni empiriche può essere calcolata usando la *distanza di Hellinger*[4]. Basandoci su questo, possiamo ordinare gli insiemi di tuple per trovare quelli che mostrano le differenze più marcate.

Questa tecnica può essere applicata all'esempio (S_1)-(S_3) fatto nel paragrafo 2.5, considerando i seguenti test d'ipotesi:

[4] Distanza di Hellinger: in probabilità e statistica è utilizzata per quantificare la similarità tra due distribuzioni di probabilità.

(i) Il *one-sample Chi square test* può essere usato per valutare se i *types* delle *activities* seguono una distribuzione discreta uniforme. Questo permetterebbe di respingere l'ipotesi nulla che tutti i tipi delle attività sono equamente distribuiti;

(ii) Il *two-sample Chi square test* può essere usato per valutare se il *sex* degli *users* che fanno *running* è equamente distribuito. Questo permetterebbe di respingere l'ipotesi nulla di probabilità equamente distribuita tra maschio e femmina;

(iii) Il *two sample Kolmogorov-Smirnov test* può essere usato se esiste una differenza statistica nella distribuzione della *length* della *running activity* tra maschio e femmina. Questo permetterebbe di valutare che la distribuzione delle durate per i corridori maschi sono statisticamente differenti da quelle delle donne.

Si noti, che il meccanismo presentato sopra può essere visto come una tecnica di subgroup discovery con le seguenti caratteristiche distintive:

- Gestisce sia attributi numerici, sia attributi categorici;
- Rappresenta i sottogruppi come query SQL;
- La classificazione degli attributi in interessanti o insolito comprende l'uso di test d'ipotesi statistici e della *Hellinger distance*;
- La ricerca di attributi rilevanti si basa nell'unione tra campionamento e meccanismi incrementali per i test di ipotesi statistici.

Capitolo 4

Soluzione proposta

4.1 Commento generale

Lo scopo di questo lavoro è quello di presentare una tecnica di database exploration per basi di dati di grandi dimensioni, dedita all'estrazione di informazioni rilevanti, che in un primo momento non sono esplicite, dai dati.

Si vuole evitare di fornire all'utente un numero elevato di dati irrilevanti. Per fare questo è quindi necessario andare ad analizzare gli attributi della base di dati presa in considerazione, e scoprire quali di essi sono rilevanti.

In questo lavoro di tesi, viene presentata una nuova tecnica basata sull'entropia degli attributi di una base di dati, la quale è in grado di stabilire se il valore di un attributo è in grado di cambiare la distribuzione di un altro attributo, e in caso positivo è anche in grado di valutare di quanto cambia questa distribuzione. Ad esempio, riprendiamo il suggerimento (S_2) presentato nel Capitolo 2:

> (S_2) "Potrebbe essere interessante esplorare il sesso degli utenti che corrono. Infatti: il 65% dei corridori sono maschi".

In questo caso per arrivare a mostrare il seguente suggerimento all'utente, il sistema basato sull'approccio proposto in questa tesi, deve calcolare la distribuzione dell'attributo *'sex'* degli utenti, che indichiamo con *D(S)*. Dopodiché seleziona dall'insieme di tuple quelle relative all'attributo *'sex'* degli utenti che praticano il '*type = running*'. Su questo sotto insieme selezionato di tuple, calcola nuovamente la distribuzione dell'attributo *'sex'*, che indichiamo con *D(S|A)*. Ottenute le due entropie, le confronta per verificare se *D(S|A)* ha

una distribuzione diversa da *D(A)*. In caso affermativo, mostra il suggerimento (S_2) all'utente.
La query corrispondente utilizzata dal sistema per fare il procedimento descritto è:

SELECT sex FROM AcmeUser JOIN Activity WHERE type='running'

Per valutare le variazioni delle distribuzioni degli attributi considerati, la tecnica di database exploration proposta si basa sull'entropia degli attributi. Riprendendo l'esempio qui sopra, quindi, viene prima calcolata l'entropia dell'attributo *'sex'* degli utenti, la quale viene indicata con *H(S)*, poi viene calcolata l'entropia dell'attributo *'sex'*, condizionato al valore del tipo di attività *'running'*. Ottenute le due entropie vengono confrontate, se esiste una variazione del comportamento dell'entropia dell'attributo *'sex'*, se viene condizionato o meno al tipo di attività.

Di seguito viene presentato il metodo utilizzato per verificare se il valore di un attributo condiziona o meno un altro attributo di una base di dati.
Dato un insieme di tuple *T*, e una coppia di attributi *A* e *A'* in *T*, calcoliamo l'entropia dell'attributo *A*, *H(A)*, a partire dai valori che assume in *T*.
Andiamo ora a individuare tutti i valori assunti dall'attributo *A'* in *T*.
Per ciascuno dei valori di *A'* andiamo a calcolare l'entropia congiunta $H(A|x_i)$ dove *x* è l'insieme dei valori assunti da *A'* e *i* è l'attributo i-esimo considerato.
Per rendere più robusta la fase di analisi, la tecnica proposta non si limita ad una semplice comparazione tra i due valori di entropia calcolati. Infatti, il calcolo di questi valori di entropia viene ripetuto utilizzando un meccanismo di bootstrap. Al termine delle iterazioni di questo meccanismo, per ogni entropia calcolata si ha una serie di valori stimati. Per ciascuna di queste entropie vengono calcolati media e varianza. Su queste vengono costruiti degli intervalli di confidenza, i quali vengono poi analizzati per prendere delle decisioni che sono più robuste rispetto alle analisi sui singoli valori.
Detto questo, procediamo a confrontare gli intervalli di confidenza tra *H(A)* e $H(A|x_i)$. Se essi risultano essere sovrapposti significa che il valore x_i

dell'attributo *A'* non è rilevante per l'attributo *A*. Viceversa se i due intervalli di confidenza risultano essere disgiunti, significa che il condizionamento dell'entropia dell'attributo *A* con il valore x_i provoca una variazione nella distribuzione dell'attributo *A* non trascurabile, e quindi possiamo dire che x_i dell'attributo *A'* è un valore rilevante per l'attributo *A*.

Dato che l'analisi ha rivelato che il valore x_i dell'attributo *A'* è un valore rilevante per l'attributo *A*, viene formulata la seguente query:

> (Q) "Potrebbe essere interessante esplorare l'attributo *A* quando l'attributo *A'* assume il valore x_i"

la quale viene mostrata nella corrispondente conversazione tra utente e sistema.

Di fondamentale importanza nella tecnica proposta è il meccanismo di bootstrap, grazie al quale è possibile fare un sotto campionamento delle tuple del database e grazie a questa capacità di ridurre il numero di tuple selezionate per l'analisi, è possibile gestire database di grandi dimensioni.

4.2 Contributo innovativo

Il metodo di analisi dei dati presentato in questo lavoro si differenzia sotto numerosi aspetti rispetto ai metodi utilizzati negli algoritmi presentati in precedenza.

La prima fondamentale differenza tra l'algoritmo proposto e quelli presenti in letteratura, è il metodo con il quale si vanno a identificare i subgroup.

Mentre la maggior parte degli algoritmi esistenti si basa sull'estrazione di regole che possano essere il più possibile rilevanti per la loro analisi, andando a coprire nel migliore dei modi i dati, il metodo proposto opera direttamente sui dati presenti nel database.

Il secondo fattore innovativo è rappresentato dall'approccio utilizzato. Infatti, in questo lavoro viene condotta un'analisi basandosi sull'entropia, l'entropia condizionata e la mutua informazione di variabili aleatorie discrete, che nessun altro algoritmo esistente ha preso in considerazione.

Di fondamentale importanza in questo lavoro, è l'analisi dell'entropia condizionata di un attributo in base ai valori assunti dagli altri attributi. Così facendo viene implementato un meccanismo di query, su un attributo, proiettate in base al valore assunto da un altro attributo, e possiamo utilizzare il valore ritornato dalla query per stabilire l'importanza che ha quel determinato attributo, quando assume un determinato valore, per l'attributo sul quale viene fatta la query.

Quando si applicano delle tecniche di data mining sui problemi reali, questi hanno solitamente grandi dimensioni. Quanto un algoritmo di data mining non lavora in modo appropriato con database di grandi dimensioni, le scelte possibili sono: ridisegnare l'algoritmo in modo tale che lavori in modo efficiente con dataset di grandi dimensioni dati in input, o ridurre la dimensione dei dati senza cambiare drasticamente i risultati ottenuti.

Il sampling è una delle tecniche più utilizzate nel data mining per ridurre la dimensione dei dati e consiste in una selezione di particolari istanze del data set in accordo con alcuni criteri. L'applicazione di tecniche di campionamento al database iniziale senza considerare dipendenze e relazioni tra i dati porta a una importante perdita di conoscenza per il processo di data exploration. Se è necessario applicare delle tecniche di ridimensionamento dei dati, è quindi di fondamentale importanza essere sicuri che non vengano perse informazioni potenzialmente importanti.

L'algoritmo presentato ha una tecnica di campionamento strutturata in modo tale da eliminare la perdita delle informazioni potenzialmente importanti.

L'algoritmo proposto presenta delle similarità con quanto fatto in [1] ma, anche in questo caso, le differenze sono rilevanti:

- in questo lavoro viene presentata una tecnica basata su entropia;
- in questo lavoro viene effettuata un'analisi tramite bootstrap, mentre in [1] viene utilizzato un test d'ipotesi;

- l'analisi degli intervalli di confidenza utilizzata in questo lavoro permette una granularità di analisi più fine, mentre il test di ipotesi utilizzato in [1] lavora in modo on/off (sovrapposto/non sovrapposto).

4.3 Idea di base dell'algoritmo proposto

L'algoritmo proposto può essere suddiviso nei seguenti sei passi:

- **Passo 1 – Campionamento**

  In questa fase vengono estratte dal dataset completo le tuple che verranno poi utilizzate nel proseguimento dell'analisi. La tecnica di campionamento scelta per l'estrazione delle tuple è quella casuale con ripetizione, cioè una tupla può essere estratta e far quindi parte del campione più di una volta.

  Il campionamento con ripetizione dei dati con la tecnica di bootstrap, permette di estrarre dall'insieme totale di tuple del database un primo subset di dati, dai quali vengono poi estratti i subset utilizzati per calcolare l'entropia condizionata.

  Con il bootstrap non è necessario che il secondo livello di estrazione dei contenga dati indipendenti.

  Consideriamo l'insieme di tutte le tuple T^Q, definiamo come T^{Q1} e T^{Q2} l'insieme totale delle tuple dell'*attributo 1* e dell'*attributo 2* rispettivamente.

  I due attributi sono dati in input all'algoritmo attraverso la query che va a interrogare il database. Oltre ai due attributi l'algoritmo riceve in input il valore *v* del secondo attributo, *attributo 2* nel nostro esempio, per il quale vogliamo andare a condizionare l'entropia del primo attributo, *attributo 1* nel nostro esempio.

  Estraiamo, applicando il campionamento casuale, da T^{Q1} e T^{Q2} due sottoinsiemi q_1 e q_2 con cardinalità di molto inferiore rispetto a T^{Q1} e T^{Q2}, cioè $|q_1| << |T^{Q1}|$ e $|q_2| << |T^{Q2}|$.

È importante notare che i due sottoinsiemi q_1 e q_2 sono paralleli cioè contengono in posizioni uguali i valori degli attributi corrispondenti alla stessa tupla del database iniziale.

- **Passo 2 – Calcolo entropia**

  In questa fase viene calcolata l'entropia dei singoli attributi. Per fare questo vengono utilizzati i due sottoinsiemi q_1 e q_2.

  I due subset q_1 e q_2 contengono rispettivamente i valori relativi alle tuple estratte per l'*attributo 1* e per l'*attributo 2*.

  Il calcolo dell'entropia di ciascun attributo utilizzando la funzione *entropy*(q_i) [13].

  Supponiamo di volerci concentrare nell'analisi dell'*attributo 1*, il quale viene considerato come attributo di riferimento.

  Andiamo quindi a calcolare la sua entropia, utilizzando il subset di valori estratti dal campionamento. Per fare ciò invochiamo la funzione: *entropy*(q_1).

  L'entropia calcolata in questo passo verrà poi utilizzata per fare il confronto con quella calcolata nel passo successivo.

- **Passo 3 – Calcolo entropia condizionata**

  In questa fase viene calcolata l'entropia dell'attributo condizionata ai valori assunti dall'altro attributo. In particolare vogliamo verificare l'andamento dell'entropia dell'*attributo 1* rispetto al valore *v* dell'*attributo 2*.

  Siano X_1 e X_2 i sottoinsiemi di q_1 e q_2 rispetto al valore *v* dell'*attributo 2*. Andiamo ora a calcolare l'entropia condizionata di X_1 e X_2 utilizzando la funzione *conditionalEntropy*(X_i, X_j) [13].

  In particolare nel nostro esempio la funzione assumerà la seguente forma: *conditionalEntropy*(X_1 , X_2).

  L'entropia calcolata in questa fase è quella che verrà utilizzata per il confronto con quella calcolata al passo precedente.

- **Passo 4 – Iterazione(Bootstrap)**

  I passi precedenti vengono ripetuti *M* volte. Ad ogni iterazione nuovi sottoinsiemi q_1, q_2, X_1 e X_2 vengono estratti e vengono eseguiti i passi 2 e 3.

Terminate le *M* iterazioni, abbiamo *M* valori per ciascuna delle entropie considerate. Per ciascuna di esse andiamo a calcolare media e varianza. Una volta calcolate, l'algoritmo prosegue con il passo 5.

- **Passo 5 – Calcolo intervalli di confidenza**

  In questa fase vengono calcolati gli intervalli di confidenza sia delle entropie dei singoli attributi, sia delle entropie condizionate calcolate nelle fasi precedenti.

  L'intervallo di confidenza è costruito a partire dalla media dell'entropia condizionata, aggiungendo e sottraendo la sua deviazione standard moltiplicata per un coefficiente λ.

- **Passo 6 – Verifica sovrapposizione intervalli di confidenza**

  In questa fase dell'algoritmo, viene fatto il confronto di sovrapposizione tra gli intervalli di confidenza dell'entropia di un attributo con le relative entropie condizionate.

```
per ogni iterazione M

    1. estrai le tuple dal dataset
    2. calcola entropie attributi
    3. calcola entropie condizionate ai valori degli attributi
fine

5. calcolo intervalli di confidenza
6. Verifica sovrapposizione intervalli di confidenza
```

Figura 4.1 Pseudocodice algoritmo proposto

4.4 Meccanismi di campionamento

Esistono diverse tecniche di campionamento, che possiamo suddividere in tre grandi categorie:

- Sequenziali;

- Casuali o randomiche;
- Ibride.

Il ***campionamento sequenziale*** è una tecnica di campionamento non probabilistica, inizialmente sviluppata come strumento di controllo di qualità dei prodotti. La dimensione del campione, *n*, non è fissata a priori, né è fissato il tasso di tempo per la raccolta dei dati.

Il processo inizia con il campionamento di un campione o di un gruppo di campioni. Questi vengono poi testati per verificare se il test d'ipotesi può essere o no rifiutato. Se il test d'ipotesi non viene rifiutato, allora un altro campione o un altro gruppo di campioni vengono campionati e testati di nuovo. In questo modo il test continua fino a che il ricercatore non è fiducioso dei risultati ottenuti.

Questa tecnica può ridurre i costi di campionamento riducendo il numero di campioni richiesti.

Tuttavia presenta problemi nel fare inferenza statistica.

Il ***campionamento casuale*** è il metodo probabilistico più utilizzato per creare un campione di una popolazione. Ogni campione è composto da elementi estratti completamente a random dalla popolazione iniziale. Come risultato, si ha che ogni elemento della popolazione ha la stessa probabilità di essere estratto durante il processo di campionamento.

Il random sampling può essere di due tipi:

- con ripetizione, cioè un elemento della popolazione può essere estratto più di una volta per ciascun campione;
- senza ripetizione, cioè un elemento della popolazione può essere estratto solo una volta per ciascun campione.

Il vantaggio principale del campionamento casuale è che il ricercatore ha la garanzia che il suo campione è rappresentativo della popolazione, e quindi analizzandolo possono essere tratte delle conclusioni statistiche valide.

Le ***campionamento ibrido*** viene utilizzato facendo cooperare tra loro due o più tecniche di campionamento, in modo tale che si possano ottenere risultati migliori rispetto all'uso singolo di una delle tecniche utilizzate.

4.5 Calcolo degli Indicatori Statistici

Gli indicatori statistici utilizzati per l'analisi delle correlazioni tra gli attributi sono:

- Entropia di una variabile aleatoria;
- Entropia congiunta di due variabili aleatorie;
- Entropia condizionata di due variabili aleatorie;
- Mutua informazione di due variabili aleatorie.

4.5.1 Entropia di una variabile aleatoria

Data una variabile aleatoria *X*, che possa assumere valori $x_1, x_2,\ldots,x_n$ con probabilità $p_1, p_2,\ldots,p_n$, si definisce l'entropia della variabile X come:

$$H(X) = E[I(X)] = -E[logP(X)]$$

Se il numero di valori che la variabile *X* è finito allora il calcolo dell'entropia si riduce ad una media dell'auto informazione di ogni simbolo x_i pesata con la propria probabilità $P(x_i)$:

$$H(X) = -\sum_{i=1}^{n} P(x_i) * \log P(x_i)$$

L'entropia può essere vista come la quantità media di informazione che riceviamo quando la variabile *X* assume un determinato valore x_i.

Nel caso in cui, invece, *X* sia una variabile aleatoria continua con distribuzione di probabilità *P(X)*, la sua entropia o entropia differenziale è definita come:

$$H(X) = -\int_{-\infty}^{+\infty} P(x) * \log_2 P(x)\, dx$$

se tale integrale esiste.

L'entropia in questo caso può assumere qualsiasi valore tra (-∞, +∞). A differenza del caso discreto, per una variabile aleatoria continua non si può interpretare l'entropia differenziale come una misura dell'incertezza media.

4.5.2 Entropia congiunta di due variabili aleatorie

Abbiamo definito l'entropia di una singola variabile aleatoria *X*. Possiamo estendere questa definizione considerando due variabili casuali e definendo l'entropia congiunta.

Date due variabili *X* e *Y*, con distribuzione di probabilità *P(X)* e *P(Y)* rispettivamente, l'entropia congiunta *H(X, Y)* è

$$H(X,Y) = -\sum_x \sum_y P(x,y) * log_2[P(x,y)]$$

dove *x* e *y* sono valori di *X* e *Y*, rispettivamente, mentre *P(x,y)* è la probabilità che questi due valori vengano assunti contemporaneamente.

L'entropia congiunta è una misura dell'incertezza associata ad un insieme di variabili casuali.

4.5.3 Entropia condizionata di due variabili aleatorie

Date due variabili aleatorie *X* e *Y*, con distribuzione di probabilità *P(X)* e *P(Y)* rispettivamente, l'entropia condizionata *H(X|Y)* è

$$H(X|Y) = -\sum_{y\in Y}\sum_{x\in X} p(x,y) * \log(p|y) = \sum_{x,y} p(x,y) * log\frac{p(y)}{p(x,y)}$$

dove *x* e *y* sono valori di *X* e *Y*, rispettivamente, *P(X,Y)* è la probabilità che questi due valori vengano assunti contemporaneamente, mentre *p(x|y)* è la probabilità condizionata dei due valori.

4.5.4 Mutua informazione di due variabili aleatorie

La mutua informazione di due variabili aleatorie è una quantità che misura la mutua dipendenza delle due variabili.

Date due variabili aleatorie *X* e *Y*, con distribuzione di probabilità *P(X)* e *P(Y)* rispettivamente, la mutua informazione *I(X; Y)* è

$$I(X;Y) = \sum_{y\in Y}\sum_{x\in X} p(x,y) * \log\left(\frac{p(x,y)}{p(x)p(y)}\right)$$

dove *x* e *y* sono valori di *X* e *Y*, *p(x, y)* è la probabilità congiunta dei due valori assunti da *X* e da *Y*, mentre *p(x)* e *p(y)* sono le probabilità che la variabile *X* e la variabile *Y* assumano i valori *x* e *y* rispettivamente.

La mutua informazione è sempre non negativa e simmetrica, cioè

I(X; Y) = I(Y; X).

4.5.5 Relazione tra entropia e mutua informazione

Si può utilizzare un diagramma di Venn per rappresentare la relazione tra entropia e mutua informazione.

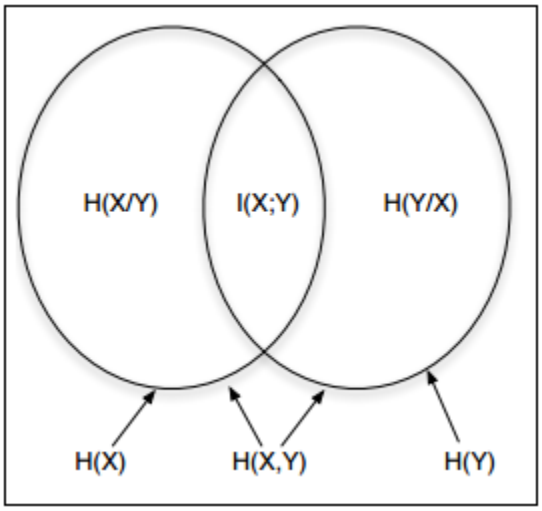

Figura 4.2 Relazione tra entropia e mutua informazione

Partendo dal seguente teorema

$$I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$$

è possibile effettuare una serie di osservazioni:

- $I(X;X) = H(X)$

  Dimostrazione Per il teorema precedente

  $$I(X;Y) = H(X) - H(X|Y)$$

  poiché abbiamo Y=X e ricordando che H(X|X)=0 si ottiene:

  $$I(X;X) = H(X) - H(X|X) = H(X)$$

- $I(X;Y) \geq 0$
- $I(X;Y) = 0 \leftrightarrow X\ e\ Y\ sono\ stocasticamente\ indipendenti$

- $H(X|Y) \leq H(X)$

  $\leftrightarrow H(X|Y) - H(X) \leq 0$
  $\leftrightarrow H(X) - H(X|Y) \geq 0$
  $\leftrightarrow I(X;Y) \geq 0$
  Questo significa che il condizionamento riduce l'entropia.

L'analisi dei database di grandi dimensioni effettuata con la tecnica statistica basata su entropia presentata in questo lavoro di tesi, non utilizza i valori teorici in quanto non è possibile sapere la distribuzione di probabilità degli attributi. Il calcolo degli indicatori statistici presentati nei paragrafi precedenti, viene fatto a partire dai dati, quindi per applicare la tecnica proposta è prima necessario andare a fare una stima di essi. Nella seguente figura viene riportato il codice della funzione matlab utilizzata per il calcolo dell'entropia condizionata di una coppia di attributi. Si può notare che l'entropia condizionata della coppia di attributi non viene calcolata direttamente, ma viene ottenuta dalla sottrazione tra l'entropia congiunta dei due attributi e l'entropia dell'attributo con il quale vogliamo condizionare.

```
function z = conditionalEntropy (x, y)
% Compute conditional entropy H(x|y) of two discrete variables x and y.

    assert(numel(x) == numel(y));
    n = numel(x);
    x = reshape(x,1,n);
    y = reshape(y,1,n);

    l = min(min(x),min(y));
    x = x-l+1;
    y = y-l+1;
    k = max(max(x),max(y));

    idx = 1:n;
    Mx = sparse(idx, round(x), 1,n,k,n);
    My = sparse(idx, round(y), 1,n,k,n);
    Pxy = nonzeros(Mx'*My/n); %joint distribution of x and y
    Hxy = -dot(Pxy,log2(Pxy+eps));

    Py = mean(My,1);
    Hy = -dot(Py,log2(Py+eps));

    % conditional entropy H(x|y)
    z = Hxy-Hy;
end
```

Figura 4.3 Pseudocodice della funzione utilizzata per conditionalEntropy, utilizzata per calcolare l'entropia condizionata di due attributi

4.6 Descrizione metodo di bootstrap utilizzato

In questo paragrafo viene descritta nel dettaglio la tecnica di bootstrap utilizzata nell'algoritmo. Per inquadrare in modo preciso il funzionamento è necessario provvedere a una formalizzazione del problema.

Consideriamo un insieme di dati Z_n ottenuto estraendo n campioni $x_1,...,x_n$ indipendenti e identicamente distribuiti da una variabile casuale x definita su X, cioè, $Z_n = \{x_1,...,x_n\}$ e costruiamo lo stimatore $F_n = F(Z_n)$.

Quello che si vuole fare è fornire un indicazione della qualità *Q* di F_n, cioè si vuole fornire un intervallo di confidenza di F_n.

La procedura ideale dovrebbe compiere la seguente:

1. estrarre gli *m* insiemi di dati indipendenti di cardinalità *n* da *X*, in modo da generare $Z^1_n,...Z^m_n$;
2. calcolare, in corrispondenza dell'*i*-esimo dataset Z^i_n lo stimatore $F^i_n=F(Z^i_n)$. Ripetere la procedura per tutti gli i=*1,...,m*.
3. stimare la qualità $Q(F^i_n,\ldots, F^m_n)$ dello stimatore F_n basato sulle m realizzazioni $F^i_n=F(Z^i_n)$, *i=1,…,m*.

Con il metodo di bootstrap gli *m* insiemi di dati Z^i_n, *i=1,...,m* sono estratti da Z_n con ripetizione. Questo significa che, ogni volta un campione x_j viene estratto e inserito in un generico insieme Z^i_n, esso viene rimesso anche in Z_n che contiene tutti i suoi *n* dati originali. Una volta che tutte le stime sono state generate sono usate per calcolare dello stimatore basata sulle m realizzazioni.

I passi chiave del bootstrap nel nostro caso sono tre:

- Campionamento casuale con ripetizione delle tuple relative agli attributi presi in considerazione;
- Calcolo dell'entropia degli attributi presi in considerazione;
- Calcolo delle entropie degli attributi condizionate ai valori dei restanti attributi.

Questo procedimento viene ripetuto *N* volte, in modo tale che per ogni stimatore calcolato abbiamo *N* realizzazioni.

Al termine delle iterazioni, per ciascuno degli indicatori statistici calcolati, viene calcolate media, varianza e deviazione standard.

La formula utilizzata per il calcolo della media utilizzata è la seguente:

$$\bar{X} = \frac{1}{N}\sum_{i=1}^{N} X_i$$

dove:

- N, è il numero di volte che viene ripetuto il bootstrap;
- X_i, è il valore dell'indicatore all'i-esima iterazione;

- $\bar{X}$, è il valore della media dell'indicatore.

La formula della varianza utilizzata è la seguente:

$$\sigma_X^2 = \frac{1}{N}\sum_{i=1}^{N}(X_i - \bar{X})^2$$

dove:

- N, è il numero di volte che viene ripetuto il bootstrap;
- X_i, è il valore dell'indicatore all'i-esima iterazione;
- $\bar{X}$, è il valore della media dell'indicatore;
- σ_X^2, è la varianza dell'indicatore statistico.

La formula della deviazione standard utilizzata è la seguente:

$$\sigma_X = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(X_i - \bar{X})^2}$$

dove:

- N, è il numero di volte che viene ripetuto il bootstrap;
- X_i, è il valore dell'indicatore all'i-esima iterazione;
- $\bar{X}$, è il valore della media dell'indicatore;
- σ_X, è la deviazione standard dell'indicatore.

Questi valori verranno poi utilizzati per la valutazione della sovrapposizione degli intervalli di confidenza tra l'entropia di un attributo e le sue entropie condizionate ai valori dell'altro attributo.

4.7 Valutazione sovrapposizione degli intervalli di confidenza

Terminato il bootstrap e calcolate media e deviazione standard, dell'entropia degli attributi e delle entropie degli attributi condizionate

ai valori degli altri attributi, possiamo andare a verificare la sovrapposizione degli intervalli di confidenza.

I test di ipotesi, utilizzati dalle altre tecniche di data exploration, portano ad una valutazione binaria: rifiuto o non rifiuto dell'ipotesi nulla.

L'intervallo di confidenza indica l'intervallo di valore dell'attributo considerato, con cui il risultato campionario che viene osservato è compatibile. Questo permette di avere delle valutazioni più robuste e precise.

Il calcolo di un intervallo di confidenza viene effettuato attraverso la seguente formula:

$$IC = \bar{x} \pm \lambda\sigma_X$$

Dove:

- $\bar{x}$, è la media dell'indicatore statistico;
- σ_X , è la deviazione standard dell'indicatore statistico;
- λ , è un parametro. Per l'analisi presentata in questo lavoro, vengono associati a λ i valori 1,2 e 3.

Una volta calcolati gli intervalli di confidenza si passa al loro confronto.

Ad esempio, supponiamo di avere due attributi X_1 e X_2 e supponiamo che l'attributo X_2 possa assumere due valori differenti:*1, 2*.

Definiamo $IC[H(X_1)]$ l'intervallo di confidenza dell'entropia dell'attributo X_1.

Definiamo ora $IC[H(X_1 | X_2=1)]$ e $IC[H(X_1 | X_2=2)]$, gli intervalli di confidenza dell'entropia di X_1 condizionata ai valori assunti dall'attributo X_2.

A questo punto si può procedere con la valutazione di sovrapposizione degli intervalli prendendo in considerazione, ad esempio, $IC[H(X_1)]$ e $IC[H(X_1 | X_2=1)]$.

Se gli intervalli di confidenza sono sovrapposti, allora si può affermare che la scelta del valore dell'attributo non porta nessuna informazione

aggiuntiva riguardo l'altro attributo e quindi non ha senso andare a fare una selezione basandosi su quel valore.

Se invece gli intervalli di confidenza sono disgiunti, si ha un vantaggio importante nell'andare a selezionare il valore dell'attributo, in quanto porta nuova informazione riguardante il secondo attributo che prima non era evidente.

Un possibile esempio di query è:

> (Q) "Potrebbe essere interessante esplorare l'attributo X_1, quando l'attributo X_2 assume il valore *2*".

La query SQL corrispondente, inserita per interrogare il sistema, è:

> SELECT X_1 FROM db WHERE X_2=2

Si fa la stessa cosa per $IC[H(X_1)]$ e $IC[H(X_1 | X_2=2)]$, e così via per tutti i possibili valori dell'attributo.

Essendo la valutazione di sovrapposizione degli intervalli di confidenza un metodo generale, è possibile anche andare a confrontare gli intervalli di confidenza delle due entropie condizionate: $IC[H(X_1 | X_2=1)]$ e $IC[H(X_1 | X_2=2)]$. In questo caso se i due intervalli risultato essere sovrapposti, significa che l'entropia dell'attributo X_1 condizionata ai valori *1* e *2* dell''attributo X_2, ha la stessa distribuzione.

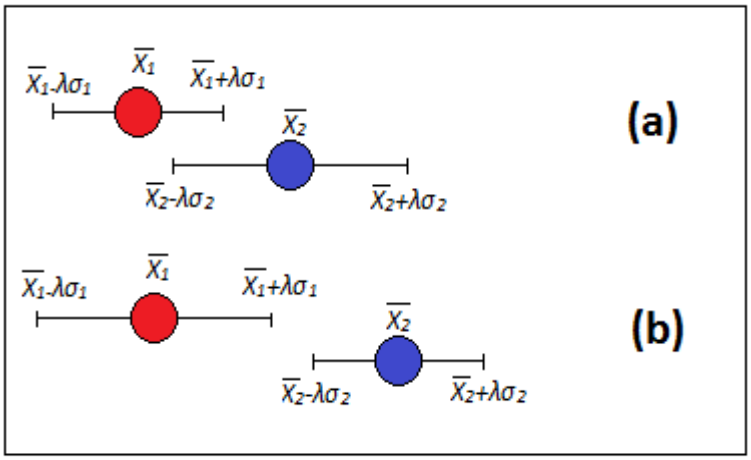

Figura 4.4 Esempio sovrapposizione intervallo di confidenza: intervalli sovrapposti (a), intervalli disgiunti(b)

4.8 Generalizzazione della tecnica proposta

La tecnica per la database exploration basata sull'entropia precedentemente mostrata, oltre ad essere applicata al confronto tra entropie di un attributo e un attributo condizionato al valore di un altro attributo della base di dati, può essere estesa a tutte le coppie attributo-valore nella sequenza di passi della conversazione tra utente e sistema.

Ad esempio, se l'utente è interessato alla *'corsa'*, il sistema controlla tra gli altri attributi della query che contiene il tipo di attività *'corsa'*, quali di essi sono rilevanti e se ne trova alcune che sono rilevanti, allora va a individuare i valori di questi attributi che sono rilevanti.

Nel caso in cui ci siano più attributi che sono rilevanti è possibile andare a ordinarli in base alla loro importanza, effettuando un ranking degli attributi. Per fare questo è sufficiente calcolare gli intervalli di confidenza dei due attributi e confrontarli con l'intervallo di confidenza dell'attributo di riferimento, andando a verificarne le relative distanze. Maggiore è la distanza tra gli intervalli di confidenza maggiore è la differenza tra le entropie e quindi più importante sarà l'attributo.

Questo stesso ragionamento può essere fatto anche considerando valori diversi ma rilevanti di uno stesso attributo per un altro attributo di riferimento. La metodologia usata è la stessa, si calcola l'intervallo di confidenza dell'attributo di riferimento, dopodiché si calcolano gli intervallo di confidenza dell'attributo di riferimento condizionato ai valori risultati rilevanti del secondo attributo. Una volta ottenuti gli intervalli di confidenza per tutti i valori si confrontano con l'intervallo di riferimento dell'attributo di riferimento. Anche in questo caso, maggiore è la distanza tra i due intervalli di confidenza maggiore è la differenza tra le entropie, è più rilevante è il valore dell'attributo per l'attributo di riferimento.

Riprendendo l'esempio di prima, controllando tra gli altri attributi della query il cui attributo di riferimento è il '*tipo di attività*', il sistema individua come attributi rilevanti il *'sesso'* degli utenti e la loro *'età'*.

Quindi calcola l'entropia dell'attributo *'tipo di attività'* condizionandola prima all'attributo *'sesso'* e poi all'attributo *'età'*. Confronta, a questo punto, l'intervallo di confidenza dell'entropia dell'attributo *'tipo di attività'* prima con l'intervallo di confidenza di *'tipo di attività'* condizionato all'attributo *'sesso'* e poi l'intervallo di confidenza di *'tipo di attività'* con l'attributo *'età'*. Guardando le differenze tra i due confronti, stabilisce quale dei due attributi risulta essere più rilevante per l'attributo *'tipo di attività'*.

Da questa analisi, il sistema trova che l'attributo *'età'* è più rilevante rispetto a *'sesso'*.

A questo punto si accorge che ci sono tre valori di *'età'* che risultano rilevanti per l'attributo *'tipo di attività'*. Allora anche in questo caso, procede al calcolo di tutti gli intervalli di confidenza necessari, li confronta e infine valutando le differenze tra le coppie di intervalli di confidenza, identifica quale tra i tre valori di *'età'* è più rilevante per l'attributo *'tipo di attività*.

Capitolo 5

Parte sperimentale

5.1 Analisi dataset dati sintetici

5.1.1 Informazioni preliminari

Abbiamo inizialmente applicato la tecnica proposta di database exploration ad un dataset di dati sintetici per validare sperimentalmente quanto proposto.

Per fare questo è stata condotta un'ampia campagna di esperimenti basati su un database di dati sintetici, composto da tre attributi e da 10 000 tuple.

Un'altra caratteristica importante da tenere in considerazione è la cardinalità degli attributi (cioè il numero di possibili valori diversi che possono avere) che nel nostro caso è tre. I tre attributi possono assumere tutti tre i seguenti tre valori: 1,2 e 3.

I dati vengono generati in base a una matrice di probabilità congiunta degli attributi che è stata definita in base a tre differenti esperimenti. Per testare tutti i possibili casi di distribuzione di probabilità congiunta che si potessero presentare, abbiamo quindi condotto tre esperimenti diversi, variando le caratteristiche della matrice di probabilità congiunta in ingresso. Gli esperimenti effettuati sono:

- Primo esperimento: *probabilità congiunta casuale*. Per svolgere questo esperimento è stato previsto di generare la matrice di probabilità congiunta casuale, estraendo dei numeri random tra 0 e 1 e successivamente normalizzando, in quanto la somma degli elementi della matrice deve essere sempre uguale a 1.
- Secondo esperimento: *probabilità congiunta uniforme*. La scelta di utilizzare una matrice di probabilità congiunta uniforme è stata

fatta per verificare che la mutua informazione per le possibili coppie di attributi ottenuta fosse nulla.
La matrice di probabilità congiunta utilizzata per questo esperimento viene generata seguendo un semplice procedimento: tutte le combinazioni degli attributi sono equiprobabili (cioè vengono calcolate tutte le possibili combinazioni degli attributi e si divide 1 per il numero calcolato. Il numeratore della divisione risulta essere uguale a 1 perché la somma degli elementi della matrice di probabilità congiunta deve essere proprio pari a 1);

- Terzo esperimento: *probabilità congiunta sbilanciata*. In questo caso abbiamo utilizzato una matrice di probabilità congiunta preimpostata, in cui la probabilità nella maggior parte delle combinazioni degli attributi è zero. Anche in questo caso la somma degli elementi della matrice di probabilità congiunta deve essere uguale a 1.

Durante l'analisi si considerano coppie di attributi, quindi prima di iniziare il calcolo degli stimatori presentati in precedenza è necessario estrarre dalla matrice iniziale, la matrice di probabilità congiunta specifica dei due attributi.
Una volta ottenuta questa matrice per tutte le possibili coppie di attributi, si può iniziare con lo studio del database.

In ciascun esperimento vengono condotte due analisi differenti:

- *Analisi considerando l'intero insieme di tuple del dataset*: in questa analisi si ha un duplice approccio, in quanto gli indicatori statistici vengono calcolati sia da un punto di vista teorico, partendo dalle matrici di probabilità congiunta calcolate in precedenza, sia da un punto di vista sperimentale, partendo dai dati generati seguendo le matrici di probabilità congiunta calcolate in precedenza.
  La motivazione che ci ha portato a scegliere questo duplice approccio è di avere una verifica della bontà dei risultati ottenuti con l'approccio sperimentale (che è quello che verrà utilizzato poi sui database reali), confrontandoli con quelli ottenuti dall'approccio teorico, i quali costituiscono il vero valore delle misure di entropia/mutua informazione.

Inoltre i dati sperimentali ottenuti considerando l'intero insieme di tuple vengono utilizzati successivamente per verificare la bontà delle stime effettuate con il bootstrap.

- *Analisi considerando un sottoinsieme di tuple del dataset utilizzando il meccanismo di bootstrap* [Presentato in Sezione 4.6]: in questa analisi si ha un singolo approccio ed è quello basato sui dati. I valori ottenuti vengono confrontati con quelli dell'approccio sperimentale condotto sull'intero dataset in modo tale da avere un riscontro sulla loro bontà.

  Per gli esperimenti che seguono, il bootstrap viene ripetuto 10 volte. Ad ogni iterazione vengono campionate randomicamente con ripetizione 500 tuple, con un totale quindi di tuple considerate alla fine delle iterazioni pari a 5000.

  Una volta terminare le iterazioni del meccanismo di bootstrap, vengono calcolate media e varianza degli indicatori statistici, le quali vengono poi utilizzate per calcolare gli intervalli di confidenza di ciascuno di essi.

  Una volta calcolati gli intervalli di confidenza, si procede alla verifica del loro comportamento, controllando se si sovrappongono o no [come mostrato in sezione 4.7]. Questo serve per capire se il valore di un attributo cambia la distribuzione di un altro attributo

Di seguito vengono presentati i risultati ottenuti sul dataset sintetico, divisi in tre sezioni differenti.

Nella Sezione 5.1.2 troviamo i risultati dell'esperimento effettuato a partire dalla matrice di probabilità congiunta in input casuale.

A seguire, nella Sezione 5.1.3, vengono presentati i risultati dell'esperimento effettuato a partire dalla matrice di probabilità congiunta in input uniforme.

Infine, nella Sezione 5.1.4, vengono presentati i risultati dell'esperimento effettuato a partire dalla matrice di probabilità congiunta in input sbilanciata.

In ciascuna di queste Sezioni, troviamo due sotto Sezioni, dove vengono presentati rispettivamente: l'analisi effettuata sul dataset completo, l'analisi effettuata con il bootstrap.

Nella Sezione 5.1.5, sono raccolti i commenti e le osservazioni generali riguardanti i tre diversi esperimenti condotti.

5.1.2 Analisi probabilità congiunta casuale

Per questo esperimento abbiamo utilizzato una probabilità congiunta in input, per il gruppo di attributi, casuale.
La matrice di probabilità congiunta dei tre attributi generata randomicamente è la seguente:

	P (A, B, C)		B		
			1	2	3
C=1	A	1	0.0381	0.0375	0.0551
		2	0.0160	0.0411	0.0558
		3	0.0657	0.0532	0.0152
C=2	A	1	0.0440	0.0262	0.0165
		2	0.0604	0.0709	0.0063
		3	0.0022	0.0261	0.0372
C=3	A	1	0.0604	0.0137	0.0582
		2	0.0111	0.0393	0.0235
		3	0.0276	0.0437	0.0551

Tabella 5.1 Dati sintetici, probabilità congiunta casuale: Probabilità congiunta di A, B, C in input

5.1.2.1 Analisi dataset completo

I risultati ottenuti considerando l'intero dataset sono:

Attributo X	Attributo Y	H(X)	H(Y)	H(X,Y)	H(Y,X)	H(X|Y)	H(Y|X)	I(X; Y)
A	B	1,5841	1,5839	3,1332	3,1332	1,5493	1,5491	0,0348
A	C	1,5841	1,5766	3,1256	3,1256	1,5490	1,5415	0,0350
B	C	1,5839	1,5766	3,1376	3,1376	1,5610	1,5537	0,0228

Tabella 5.2 Dati Sintetici, probabilità congiunta casuale: valori indicatori statistici ottenuti con l'approccio teorico

Attributo X	Attributo Y	H(X)	H(Y)	H(X,Y)	H(Y,X)	H(X|Y)	H(Y|X)	I(X; Y)
A	B	1,5835	1,5836	3,1311	3,1311	1,5475	1,5476	0,0359
A	C	1,5834	1,5767	3,1234	3,1234	1,5468	1,5400	0,0366
B	C	1,5835	1,5737	3,1390	3,1390	1,5654	1,5555	0,0181

Tabella 5.3 Dati Sintetici, probabilità congiunta casuale: valori indicatori statistici ottenuti con l'approccio sperimentale

L'entropia di un attributo condizionata ai valori dell'altro attributo della coppia A-B:

	H(A|B=x) teorica	H(A|B=x) sperimentale	Errore Assoluto	H(B|A=x) teorica	H(B|A=x) sperimentale	Errore Assoluto
x=1	1,5503	1,5524	0,0021	1,5400	1,5425	0,0025
x=2	1,5342	1,5343	0,0000	1,5305	1,5248	0,0056
x=3	1,5646	1,5542	0,0105	1,5772	1,5755	0,0017

Tabella 5.4 Dati sintetici, probabilità congiunta casuale, coppia A-B: Entropia condizionata H(A|B=x) e H(B|A=x), con x=1,2,3

L'entropia di un attributo condizionata ai valori dell'altro attributo della coppia A-C:

	H(A|C=x) teorica	H(A|C=x) sperimentale	Errore Assoluto	H(C|A=x) teorica	H(C|A=x) sperimentale	Errore Assoluto
x=1	1,5809	1,5765	0,0045	1,5599	1,5598	0,0001
x=2	1,5159	1,5243	0,0084	1,5408	1,5459	0,0051
x=3	1,5416	1,5508	0,0091	1,5200	1,5200	0,0044

Tabella 5.5 Dati sintetici, probabilità congiunta casuale, coppia A-B: Entropia condizionata H(A|C=x) e H(C|A=x), con x=1,2,3

L’entropia di un attributo condizionata ai valori dell’altro attributo della coppia B-C:

	H(B|C=x) teorica	H(B|C=x) sperimentale	Errore Assoluto	H(C|B=x) teorica	H(C|B=x) sperimentale	Errore Assoluto
x=1	1,5839	1,5842	0,0004	1,5805	1,5797	0,0008
x=2	1,5259	1,5428	0,0169	1,5729	1,5774	0,0045
x=3	1,5658	1,5747	0,0089	1,5100	1,5100	0,0074

Tabella 5.6 Dati sintetici, probabilità congiunta casuale, coppia A-B: Entropia condizionata H(B|C=x) e H(C|B=x), con x=1,2,3

5.1.2.2 Analisi con bootstrap

L’analisi effettuata con il bootstrap ha prodotto i seguenti risultati:

Attributo X	Attributo Y	H(X)	H(Y)	H(X,Y)	H(Y,X)	H(X|Y)	H(Y|X)	I(X; Y)
A	B	1,5780	1,5821	3,1247	3,1248	1,5427	1,5468	0,0353
A	C	1,5797	1,5707	3,1057	3,1057	1,5350	1,5259	0,0447
B	C	1,5803	1,5707	3,1266	3,1266	1,5559	1,5462	0,0245

Tabella 5.7 Dati sintetici, probabilità congiunta casuale: valori medi indicatori statistici calcolati con bootstrap

Attributo X	Attributo Y	Var (H(X))	Var (H(Y))	Var (H(X,Y))	Var (H(Y,X))	Var (H(X|Y))	Var (H(Y|X))	Var (I(X; Y))
A	B	0,0001	0,0001	0,0001	0,0001	0,0001	0,0001	0,0001
A	C	0,0001	0,0001	0,0003	0,0003	0,0003	0,0003	0,0003
B	C	0,0001	0,0001	0,0003	0,0003	0,0002	0,0002	0,0001

Tabella 5.8 Dati sintetici, probabilità congiunta casuale: varianze indicatori statistici calcolati con bootstrap

L’analisi delle entropie condizionate a ciascun valore degli attributi ha prodotto i seguenti risultati:

- Coppia A-B

	H(A|B=x)	Var (H(A|B=x))	H(B|A=x)	Var (H(B|A=x))
x=1	1,5469	0,0012	1,5360	0,0008
x=2	1,5305	0,0007	1,5330	0,0006
x=3	1,5519	0,0004	1,5725	0,0001

Tabella 5.9 Dati sintetici, probabilità congiunta casuale: valor medio e varianza H(A|B=x) e H(B|A=x), con x=1,2,3 ottenute con bootstrap

La relativa analisi di sovrapposizione degli intervalli di confidenza tra entropia dell’attributo e le sue entropie congiunte condizionate al valore dell’altro attributo della coppia, ha portato i seguenti risultati:

	H(A|B=1)	H(A|B=2)	H(A|B=3)	H(B|A=1)	H(B|A=2)	H(B|A=3)
λ =1	Sì	No	Sì	No	No	Sì
λ =2	Sì	Sì	Sì	Sì	Sì	Sì
λ =3	Sì	Sì	Sì	Sì	Sì	Sì

Tabella 5.10 Dati sintetici, probabilità congiunta casuale: sovrapposizione intervalli di confidenza H(A) - H(A|B=b), con b=1,2,3 e H(B) – H(B|A=a), con a=1,2,3 al variare di lambda

- Coppia A-C

	H(A|C=x)	Var (H(A|C=x))	H(C|A=x)	Var (H(C|A=x))
x=1	1,5659	0,0003	1,5444	0,0005
x=2	1,4968	0,0005	1,5395	0,0007
x=3	1,5309	0,0011	1,4914	0,0008

Tabella 5.11 Dati sintetici, probabilità congiunta casuale: valor medio e varianza H(A|C=x) e H(C|A=x), con x=1,2,3 ottenute con bootstrap

La relativa analisi di sovrapposizione degli intervalli di confidenza tra entropia dell’attributo e le sue entropie congiunte

condizionate al valore dell'altro attributo della coppia, ha portato i seguenti risultati:

	H(A|C=1)	H(A|C=2)	H(A|C=3)	H(C|A=1)	H(C|A=2)	H(C|A=3)
λ =1	Sì	No	No	Sì	Sì	No
λ =2	Sì	No	Sì	Sì	Sì	No
λ =3	Sì	No	Sì	Sì	Sì	Sì

Tabella 5.12 Dati sintetici, probabilità congiunta casuale: sovrapposizione intervalli di confidenza H(A) - H(A|C=c), con c=1,2,3 e H(C) – H(C|A=a), con c=1,2,3 al variare di lambda

- Coppia B-C

	H(B|C=x)	Var (H(B|C=x))	H(C|B=x)	Var (H(C|B=x))
x=1	1,5677	0,0009	1,5730	0,0001
x=2	1,5157	0,0005	1,5687	0,0005
x=3	1,5760	0,00003	1,4922	0,0003

Tabella 5.13 Dati sintetici, probabilità congiunta casuale: valor medio e varianza H(B|C=x) e H(C|B=x), con x=1,2,3 ottenute con bootstrap

La relativa analisi di sovrapposizione degli intervalli di confidenza tra entropia dell'attributo e le sue entropie congiunte condizionate al valore dell'altro attributo della coppia, ha portato i seguenti risultati:

	H(B|C=1)	H(B|C=2)	H(B|C=3)	H(C|B=1)	H(C|B=2)	H(C|B=3)
λ =1	Sì	No	Sì	Sì	Sì	No
λ =2	Sì	No	Sì	Sì	Sì	No
λ =3	Sì	Sì	Sì	Sì	Sì	Sì

Tabella 5.14 Dati sintetici, probabilità congiunta casuale: sovrapposizione intervalli di confidenza H(B) - H(B|C=c), con c=1,2,3 e H(C) – H(C|B=b), con b=1,2,3 al variare di lambda

L'analisi di sovrapposizione degli intervalli di confidenza per questo esperimento ha rivelato che per tutte le coppie di attributi ci sono uno o più casi in cui gli intervalli di confidenza sono disgiunti.

Analizzando la coppia di attributi A-B, si evidenzia gli intervalli di confidenza di che $H(A)$ e $H(A|B=2)$ per λ =1 non si sovrappongono. Passando ad analizzare $H(B)$ e $H(B|A)$ è possibile osservare che per $A=1, 2$ gli intervalli di confidenza per $\lambda = 1$ sono disgiunti.

Passando alla coppia di attributi A-C è possibile osservare che $H(A)$ e $H(A|C=2)$ hanno degli intervalli di confidenza disgiunti per ogni λ.

Quindi è utile andare ad interrogare il sistema con la query:

```
SELECT A FROM db WHERE C=2
```

e analizzarne i risultati ritornati, in quanto questa query potrebbe portare in risalto nuove informazioni utili che andrebbero ad aggiungersi a quelle che già conosciamo.

Continuando nell'analisi della coppia di attributi A-C, è possibile osservare che $H(C)$ e $H(C|A=3)$ presentano intervalli disgiunti per $\lambda = 1, 2.$

Infine, considerando la coppia di attributi B-C è possibile notare che $H(B)$ e $H(B|C=2)$ presentano intervalli di confidenza disgiunti per $\lambda = 1, 2$, mentre $H(C)$ e $H(C|B)$ presentano degli intervalli disgiunti per $B=3$ e $\lambda = 1, 2$.

Consideriamo ora ad esempio la coppia di attributi B-C, e nel caso specifico l'analisi di sovrapposizione degli intervalli fatta per $H(C)$ e $H(C|B)$, con B che assume i valori 1 e 2. Come è possibile vedere dalla tabella corrispondente, gli intervalli di confidenza di $H(C)$ e $H(C|1)$, così come quelli di $H(C)$ e $H(C|2)$, si sovrappongono per ogni lambda. Questo significa che se dovessimo andare ad interrogare il sistema con la seguente query:

```
SELECT C FROM db WHERE B=1 OR B=2
```

i risultati ritornati non aggiungerebbero nuove informazioni rispetto a quelle di cui eravamo già in possesso.

5.1.3 Analisi probabilità congiunta uniforme

Per questa elaborazione abbiamo forzato la matrice di probabilità congiunta in ingresso con valori uniformi per tutti le possibili combinazione tra gli attributi. Quindi in questo caso sapendo che la somma di tutte le probabilità congiunte deve essere 1, abbiamo calcolato la probabilità di una singolo combinazione con la seguente formula:

$$P(X,Y) = 1/(numValoriAttributo1 * numValoriAttributo2 * numValoriAttributo3)$$

La matrice di probabilità congiunta risulta quindi essere:

	P (A, B, C)		B		
			1	2	3
C=1	A	1	0.0370	0.0370	0.0370
		2	0.0370	0.0370	0.0370
		3	0.0370	0.0370	0.0370
C=2	A	1	0.0370	0.0370	0.0370
		2	0.0370	0.0370	0.0370
		3	0.0370	0.0370	0.0370
C=3	A	1	0.0370	0.0370	0.0370
		2	0.0370	0.0370	0.0370
		3	0.0370	0.0370	0.0370

Tabella 5.15 Dati sintetici, probabilità congiunta uniforme: Probabilità congiunta di A,B,C in input

5.1.3.1 Analisi dataset completo

I risultati ottenuti considerando l'insieme totale delle tuple sono:

Attributo X	Attributo Y	H(X)	H(Y)	H(X,Y)	H(Y,X)	H(X|Y)	H(Y|X)	I(X; Y)
A	B	1,5849	1,5849	3,1699	3,1699	1,5849	1,5849	0
A	C	1,5849	1,5849	3,1699	3,1699	1,5849	1,5849	0
B	C	1,5849	1,5849	3,1699	3,1699	1,5849	1,5849	0

Tabella 5.16 Dati Sintetici, probabilità congiunta uniforme: valori indicatori statistici ottenuti con l'approccio teorico

Attributo X	Attributo Y	H(X)	H(Y)	H(X,Y)	H(Y,X)	H(X|Y)	H(Y|X)	I(X; Y)
A	B	1,5849	1,5848	3,1694	3,1694	1,5845	1,5844	0,0004
A	C	1,5848	1,5849	3,1695	3,1695	1,5847	1,5847	0,0002
B	C	1,5849332	1,5849	3,1698	3,1698	1,5849	1,5849	0,0001

Tabella 5.17 Dati Sintetici, probabilità congiunta uniforme: valori indicatori statistici ottenuti con l'approccio sperimentale

L'entropia di un attributo condizionata ai valori dell'altro attributo della coppia A-B:

	H(A|B=x) teorica	H(A|B=x) sperimentale	Errore Assoluto	H(B|A=x) teorica	H(B|A=x) sperimentale	Errore Assoluto
x=1	1,58382	1,584360622	0,000540292	1,58382033	1,584962501	0,00114217
x=2	1,584954	1,583482	0,001472019	1,584954	1,584963	8,25E-06
x=3	1,584349	1,583682	0,000667	1,584349126	1,584963	0,000613

Tabella 5.18 Dati sintetici, probabilità congiunta uniforme, coppia A-B: Entropia condizionata H(A\|B=x) e H(B\|A=x), con x=1,2,3

L'entropia di un attributo condizionata ai valori dell'altro attributo della coppia A-C:

	H(A|C=x) teorica	H(A|C=x) sperimentale	Errore Assoluto	H(C|A=x) teorica	H(C|A=x) sperimentale	Errore Assoluto
x=1	1,584801232	1,584271495	0,00052974	1,583342004	1,584962501	0,0016205
x=2	1,583393	1,584925	0,00153172	1,584726	1,584963	0,0002363
x=3	1,583416071	1,584962	0,001546	1,584672249	1,58E+00	0,00029

Tabella 5.19 Dati sintetici, probabilità congiunta casuale, coppia A-B: Entropia condizionata H(A\|C=x) e H(C\|A=x), con x=1,2,3

L’entropia di un attributo condizionata ai valori dell’altro attributo della coppia B-C:

	H(B|C=x) teorica	H(B|C=x) sperimentale	Errore Assoluto	H(C|B=x) teorica	H(C|B=x) sperimentale	Errore Assoluto
x=1	1,5835	1,5846	0,0011	1,5843	1,5849	0,0007
x=2	1,5846	1,5837	0,0008	1,5842	1,5849	0,0008
x=3	1,5842	1,5826	0,0015	1,5831	1,5800	0,0019

Tabella 5.20 Dati sintetici, probabilità congiunta casuale, coppia A-B: Entropia condizionata H(B|C=x) e H(C|B=x), con x=1,2,3

5.1.3.2 Analisi con bootstrap

L’analisi effettuata con il bootstrap ha prodotto i seguenti risultati:

Attributo X	Attributo Y	H(X)	H(Y)	H(X,Y)	H(Y,X)	H(X|Y)	H(Y|X)	I(X; Y)
A	B	1,5814	1,5795	3,1557	3,1557	1,5762	1,5742	0,0053
A	C	1,5811	1,5831	3,1545	3,1545	1,5713	1,5733	0,0098
B	C	1,5821	1,5813	3,1596	3,1596	1,5783	1,5775	0,0038

Tabella 5.21 Dati sintetici, probabilità congiunta uniforme: valori medi indicatori statistici calcolati con bootstrap

Attributo X	Attributo Y	Var (H(X))	Var (H(Y))	Var (H(X,Y))	Var (H(Y,X))	Var (H(X|Y))	Var (H(Y|X))	Var (I(X; Y))
A	B	0,00001	0,00007	0,00009	0,00009	0,00004	0,00006	0,00001
A	C	0,00002	0,00000	0,00011	0,00011	0,00011	0,00008	0,00008
B	C	0,00001	0,00001	0,00002	0,00002	0,00001	0,00002	0,00001

Tabella 5.22 Dati sintetici, probabilità congiunta uniforme: varianze indicatori statistici calcolati con bootstrap

L’analisi delle entropie condizionate a ciascun valore degli attributi ha prodotto i seguenti risultati:

- Coppia A-B

	H(A|B=x)	Var (H(A|B=x))	H(B|A=x)	Var (H(B|A=x))
x=1	1,5779	4,74E-05	1,5739	7,84E-05
x=2	1,5809	9,29E-06	1,5762	7,77E-05
x=3	1,5696	0,0002	1,5722	0,0001

Tabella 5.23 Dati sintetici, probabilità congiunta uniforme: valor medio e varianza H(A|B=x) e H(B|A=x), con x=1,2,3 ottenute con bootstrap

La relativa analisi di sovrapposizione degli intervalli di confidenza tra entropia dell'attributo e le sue entropie congiunte condizionate al valore dell'altro attributo della coppia, ha portato i seguenti risultati:

	H(A|B=1)	H(A|B=2)	H(A|B=3)	H(B|A=1)	H(B|A=2)	H(B|A=3)
λ =1	Sì	Sì	Sì	Sì	Sì	Sì
λ =2	Sì	Sì	Sì	Sì	Sì	Sì
λ =3	Sì	Sì	Sì	Sì	Sì	Sì

Tabella 5.24 Dati sintetici, probabilità congiunta uniforme: sovrapposizione intervalli di confidenza H(A) - H(A|B=b), con b=1,2,3 e H(B) – H(B|A=a), con a=1,2,3 al variare di lambda

- Coppia A-C

	H(A|C=x)	Var (H(A|C=x))	H(C|A=x)	Var (H(C|A=x))
x=1	1,5785	5,13E-05	1,5791	2,15E-05
x=2	1,5649	0,0008	1,5655	0,0004
x=3	1,5702	0,0001	1,5746	1,04E-04

Tabella 5.25 Dati sintetici, probabilità congiunta uniforme: valor medio e varianza H(A|C=x) e H(C|A=x), con x=1,2,3 ottenute con bootstrap

La relativa analisi di sovrapposizione degli intervalli di confidenza tra entropia dell'attributo e le sue entropie congiunte condizionate al valore dell'altro attributo della coppia, ha portato i seguenti risultati:

	H(A|C=1)	H(A|C=2)	H(A|C=3)	H(C|A=1)	H(C|A=2)	H(C|A=3)
λ =1	Sì	Sì	Sì	Sì	Sì	Sì
λ =2	Sì	Sì	Sì	Sì	Sì	Sì
λ =3	Sì	Sì	Sì	Sì	Sì	Sì

Tabella 5.26 Dati sintetici, probabilità congiunta uniforme: sovrapposizione intervalli di confidenza H(A) - H(A|C=c), con c=1,2,3 e H(C) – H(C|A=a), con c=1,2,3 al variare di lambda

- Coppia B-C

	H(B|C=x)	Var (H(B|C=x))	H(C|B=x)	Var (H(C|B=x))
x=1	1,5791	3,23E-05	1,5752	9,70E-05
x=2	1,5778	2,51E-05	1,5786	5,21E-05
x=3	1,5782	2,71E-05	1,5786	2,08E-05

Tabella 5.27 Dati sintetici, probabilità congiunta uniforme: valor medio e varianza H(B|C=x) e H(C|B=x), con x=1,2,3 ottenute con bootstrap

La relativa analisi di sovrapposizione degli intervalli di confidenza tra entropia dell'attributo e le sue entropie congiunte condizionate al valore dell'altro attributo della coppia, ha portato i seguenti risultati:

	H(B|C=1)	H(B|C=2)	H(B|C=3)	H(C|B=1)	H(C|B=2)	H(C|B=3)
λ =1	Sì	Sì	Sì	Sì	Sì	Sì
λ =2	Sì	Sì	Sì	Sì	Sì	Sì
λ =3	Sì	Sì	Sì	Sì	Sì	Sì

Tabella 5.28 Dati sintetici, probabilità congiunta uniforme: sovrapposizione intervalli di confidenza H(B) - H(B|C=c), con c=1,2,3 e H(C) – H(C|B=b), con b=1,2,3 al variare di lambda

Al termine di questo esperimento, è possibile notare che per tutte e tre le coppie di attributi il risultato è lo stesso: per ogni valore possibile dei tre attributi e per ogni lambda, c'è sovrapposizione degli intervalli di confidenza delle rispettive entropie.

Questo comportamento non deve sorprendere, anzi conferma proprio quello che si voleva dimostrare: presa una coppia di attributi, se i due attributi sono

indipendenti, il valore assunto da uno di essi non cambia la distribuzione dell'altro.

Quindi, in questo caso, non avrebbe alcun senso andare ad interrogare il sistema con delle query come ad esempio:

'SELECT A FROM db WHERE C=1'

perché i risultati ritornati non aggiungerebbero nuove informazioni rispetto a quelle che già abbiamo.

5.1.4 Analisi probabilità congiunta sbilanciata

Per quanto riguarda questa elaborazione, abbiamo inserito appositamente una matrice di probabilità congiunta sbilanciata per il gruppo di attributi, cioè dove nella maggior parte dei casi la probabilità che gli attributi assumano contemporaneamente un determinato valore è pari a zero.

In particolare la probabilità congiunta inserita è la seguente:

	P (A, B, C)		B		
			1	2	3
C=1	A	1	0.1000	0	0
		2	0	0.1000	0
		3	0	0	0.1000
C=2	A	1	0.1000	0.1000	0
		2	0	0.1000	0
		3	0	0	0.1000
C=3	A	1	0.1000	0	0
		2	0	0.1000	0
		3	0	0	0.1000

Tabella 5.29 Dati sintetici, probabilità congiunta sbilanciata: Probabilità congiunta di A,B,C in input

5.1.4.1 Analisi dataset completo

I risultati ottenuti considerando l'insieme totale delle tuple sono:

Attributo X	Attributo Y	H(X)	H(Y)	H(X,Y)	H(Y,X)	H(X|Y)	H(Y|X)	I(X; Y)
A	B	1,5709	1,5709	1,8954	1,8954	0,3245	0,3245	1,2464
A	C	1,5709	1,5709	3,1219	3,1219	1,5509	1,5509	0,0199
B	C	1,5709	1,5709	3,1219	3,1219	1,5509	1,5509	0,0199

Tabella 5.30 Dati Sintetici, probabilità congiunta sbilanciata: valori indicatori statistici ottenuti con l'approccio teorico

Attributo X	Attributo Y	H(X)	H(Y)	H(X,Y)	H(Y,X)	H(X|Y)	H(Y|X)	I(X; Y)
A	B	1,5692	1,5705	1,8998	1,8998	0,3293	0,3306	1,2399
A	C	1,5669	1,5744	3,1210	3,1210	1,5466	1,5541	0,0203
B	C	1,5746	1,5724	3,1239	3,1239	1,5516	1,5494	0,0230

Tabella 5.31 Dati Sintetici, probabilità congiunta sbilanciata: valori indicatori statistici ottenuti con l'approccio sperimentale

L'entropia di un attributo condizionata ai valori dell'altro attributo della coppia A-B:

	H(A|B=x) teorica	H(A|B=x) sperimentale	Errore Assoluto	H(B|A=x) teorica	H(B|A=x) sperimentale	Errore Assoluto
x=1	1,5838	1,5843	0,0005	1,5838	1,5849	0,0011
x=2	1,5849	1,5834	0,0014	1,5849	1,5849	8,25E-06
x=3	1,5843	1,5836	0,0006	1,5843	1,5849	0,0006

Tabella 5.32 Dati sintetici, probabilità congiunta sbilanciata, coppia A-B: Entropia condizionata H(A\|B=x) e H(B\|A=x), con x=1,2,3

L'entropia di un attributo condizionata ai valori dell'altro attributo della coppia A-C:

	H(A|C=x) teorica	H(A|C=x) sperimentale	Errore Assoluto	H(C|A=x) teorica	H(C|A=x) sperimentale	Errore Assoluto
x=1	1,5849	1,5845	0,0004	1,5	1,5100	0,0100
x=2	1,5	1,4885	0,0114	1,5849	1,5691	0,01583
x=3	1,5849	1,5596	0,0253	1,5849	1,57	0,0159

Tabella 5.33 Dati sintetici, probabilità congiunta sbilanciata, coppia A-B: Entropia condizionata H(A|C=x) e H(C|A=x), con x=1,2,3

L'entropia di un attributo condizionata ai valori dell'altro attributo della coppia B-C:

	H(B|C=x) teorica	H(B|C=x) sperimentale	Errore Assoluto	H(C|B=x) teorica	H(C|B=x) sperimentale	Errore Assoluto
x=1	1,5849	1,5842	0,0008	1,5849	1,5849	1,5849
x=2	1,5	1,5017	0,0017	1,5841	1,5841	1,5841
x=3	1,5849	1,5639	0,0210	0,0009	0,0009	0,0009

Tabella 5.34 Dati sintetici, probabilità congiunta sbilanciata, coppia A-B: Entropia condizionata H(B|C=x) e H(C|B=x), con x=1,2,3

5.1.4.2 Analisi con bootstrap

L'analisi effettuata con il bootstrap ha prodotto i seguenti risultati:

Attributo X	Attributo Y	H(X)	H(Y)	H(X,Y)	H(Y,X)	H(X|Y)	H(Y|X)	I(X; Y)
A	B	1,5720	1,5720	1,8817	1,8817	0,3097	0,3097	1,2624
A	C	1,5642	1,5728	3,1170	3,1170	1,5442	1,5528	0,0200
B	C	1,5721	1,5645	3,1091	3,1091	1,5446	1,5370	0,0274

Tabella 5.35 Dati sintetici, probabilità congiunta sbilanciata: valori medi indicatori statistici calcolati con bootstrap

Attributo X	Attributo Y	Var (H(X))	Var (H(Y))	Var (H(X,Y))	Var (H(Y,X))	Var (H(X|Y))	Var (H(Y|X))	Var (I(X; Y))
A	B	0,0000	0,0001	0,0008	0,0008	0,0012	0,0009	0,0013
A	C	0,0002	0,0001	0,0005	0,0005	0,0003	0,0001	0,0001
B	C	0,0001	0,0001	0,0004	0,0004	0,0002	0,0002	0,0001

Tabella 5.36 Dati sintetici, probabilità congiunta sbilanciata: varianze indicatori statistici calcolati con bootstrap

L'analisi delle entropie condizionate a ciascun valore degli attributi ha prodotto i seguenti risultati:

- Coppia A-B

	H(A|B=x)	Var (H(A|B=x))	H(B|A=x)	Var (H(B|A=x))
x=1	0	0	0,78863	0,004159
x=2	0,790384	3,62E-03	0	0
x=3	0	0	0	0

Tabella 5.37 Dati sintetici, probabilità congiunta sbilanciata: valor medio e varianza H(A|B=x) e H(B|A=x), con x=1,2,3 ottenute con bootstrap

La relativa analisi di sovrapposizione degli intervalli di confidenza tra entropia dell'attributo e le sue entropie congiunte condizionate al valore dell'altro attributo della coppia, ha portato i seguenti risultati:

	H(A|B=1)	H(A|B=2)	H(A|B=3)	H(B|A=1)	H(B|A=2)	H(B|A=3)
λ =1	No	No	No	No	No	No
λ =2	No	No	No	No	No	No
λ =3	No	No	No	No	No	No

Tabella 5.38 Dati sintetici, probabilità congiunta sbilanciata: sovrapposizione intervalli di confidenza H(A) - H(A|B=b), con b=1,2,3 e H(B) – H(B|A=a), con a=1,2,3 al variare di lambda

- Coppia A-C

	H(A|C=c)	Var (H(A|C=c))	H(C|A=a)	Var (H(C|A=a))
x=1	1,571824	0,000157	1,521754	0,000392
x=2	1,494812	0,001325	1,576597	3,29E-05
x=3	1,579765	1,43E-05	1,573105	8,07E-05

Tabella 5.39 Dati sintetici, probabilità congiunta sbilanciata: valor medio e varianza H(A|C=x) e H(C|A=x), con x=1,2,3 ottenute con bootstrap

La relativa analisi di sovrapposizione degli intervalli di confidenza tra entropia dell'attributo e le sue entropie congiunte condizionate al valore dell'altro attributo della coppia, ha portato i seguenti risultati:

	H(A|C=1)	H(A|C=2)	H(A|C=3)	H(C|A=1)	H(C|A=2)	H(C|A=3)
λ =1	Sì	No	Sì	No	Sì	Sì
λ =2	Sì	Sì	Sì	Sì	Sì	Sì
λ =3	Sì	Sì	Sì	Sì	Sì	Sì

Tabella 5.40 Dati sintetici, probabilità congiunta sbilanciata: sovrapposizione intervalli di confidenza H(A) - H(A|C=c), con c=1,2,3 e H(C) – H(C|A=a), con c=1,2,3 al variare di lambda

- Coppia B-C

	H(B|C=x)	Var (H(B|C=x))	H(C|B=b)	Var (H(C|B=b))
x=1	1,5722	0,0004	1,5764	2,57E-05
x=2	1,5041	0,0011	1,4783	0,0014
x=3	1,5722	0,0002	1,5727	7,31E-05

Tabella 5.41 Dati sintetici, probabilità congiunta sbilanciata: valor medio e varianza H(B|C=x) e H(C|B=x), con x=1,2,3 ottenute con bootstrap

La relativa analisi di sovrapposizione degli intervalli di confidenza tra entropia dell'attributo e le sue entropie congiunte condizionate al valore dell'altro attributo della coppia, ha portato i seguenti risultati:

	H(B|C=1)	H(B|C=2)	H(B|C=3)	H(C|B=1)	H(C|B=2)	H(C|B=3)
λ =1	Sì	No	Sì	Sì	No	Sì
λ =2	Sì	Sì	Sì	Sì	Sì	Sì
λ =3	Sì	Sì	Sì	Sì	Sì	Sì

Tabella 5.42 Dati sintetici, probabilità congiunta sbilanciata: sovrapposizione intervalli di confidenza H(B) - H(B|C=c), con c=1,2,3 e H(C) – H(C|B=b), con b=1,2,3 al variare di lambda

Al termine di questo esperimento quello che si evidenzia è che la per coppia di attributi A-B non si ha mai sovrapposizione degli intervalli di confidenza per ogni possibile caso analizzato. Questo è dovuto alla matrice di probabilità congiunta relativa della coppia, la quale presenta numerosi valori nulli, come è possibile osservare della seguente tabella.

P(A, B)		B		
		1	2	3
A	1	0,300	0,100	0
	2	0	0,300	0
	3	0	0	0,300

Tabella 5.43 Dati sintetici, probabilità congiunta sbilanciata: probabilità congiunta coppia attributi A-B

La presenza di molti valori nulli ha la conseguenza di decrementare la precisione di calcolo e quindi i risultati possono essere diversi rispetto a quelli attesi. Questo non accade invece per le coppie A-C e B-C, come è possibile osservare dalla *Tabella 5.45* e dalla *Tabella 5.46*, per le quali nonostante i numeri valori nulli della matrice di probabilità congiunta originaria, non si hanno valori nulli nelle relative matrici di probabilità.

Osservando i valori delle entropie condizionate ottenute con il bootstrap, è possibile notare che molte di esse assumono valore pari a 0. Questo è dovuto alla matrice di probabilità congiunta in input che assume dei valori sbilanciati e questo fa sì che alcune combinazioni dei valori degli attributi non si verifichino mai. Questo comportamento si ha, però, solamente per la coppia A-B. Infatti andando analizzare i risultati ottenuti per la coppia A-C, è possibile notare che $H(A)$ e $H(A|C=2)$ per $\lambda = 1$ hanno gli intervalli di confidenza disgiunti, così come $H(C)$ e $H(C|A=2)$ per $\lambda = 1$.

Passando all'analisi della coppia B-C, è possibile osservare che presentano degli intervalli di confidenza disgiunti nei seguenti casi: $H(B)$ e $H(B|C=2)$ con $\lambda = 1$, $H(C)$ e $H(C|B=2)$ con $\lambda = 1$.

5.1.5 Riassunto delle osservazioni sui risultati ottenuti sul dataset sintetico

Di seguito vengono riportate le osservazioni sugli esperimenti svolti sul dataset sintetico.

Per quanto riguarda il primo esperimento, quello in cui la matrice di probabilità congiunta in input è casuale, si nota che:

- l'analisi condotta sull'intero insieme di tuple del database evidenzia che i risultati ottenuti possono essere considerati validi. Infatti l'approccio sperimentale ha prodotto risultati in linea a quelli prodotti dall'approccio teorico.
- l'analisi condotta utilizzando il meccanismo di bootstrap ha evidenziato che, nonostante il minor numero di tuple utilizzato, i valori degli indicatori statistici ottenuti sono coerenti con quelli ottenuti con l'analisi sull'intero database.

  L'analisi di sovrapposizione degli intervalli di confidenza tra l'entropia dell'attributo e l'entropia dell'attributo condizionata ai valori dell'altro attributo della coppia, ha prodotto i seguenti risultati:

  - Coppia A-B

    L'unico caso in cui gli intervalli di confidenza di $H(A)$ e $H(A|B = b)$ con $b = 1,2,3$, sono disgiunti è per $H(A|2)$ con $\lambda = 1$.

    I casi in cui gli intervalli di confidenza di $H(B)$ e $H(B|A=a)$ con $a=1,2,3$, sono disgiunti $H(B|1)$ con $\lambda=1$ e $H(B|2)$ con $\lambda=1$.

Questo risulta ragionevole perché andando a verificare il valore assunto dalle entropie condizionate, quella di $H(A|2)$ *è* quella che si discosta maggiormente da $H(A)$, e quindi quello che porta più informazione quando si verifica. La stessa valutazione può essere fatta per l'entropia condizionata *H(B|A),* per la quale per *a=1, 2* abbiamo entropie condizionate che si discostano in maniera sensibile dall'entropia di *H(B)*, suggerendo la possibile scoperta di informazioni a priori non esplicite, se si dovesse andare a condizionare l'entropia di *B* con uno dei due valori. Un esempio di query è:

SELECT B FROM db WHERE A=1

- Coppia A-C

  Gli intervalli di confidenza di $H(A)$ e $H(A|C=c)$ con $c=1,2,3$, risultano essere disgiunti per *H(A/2)* con λ *=1,2,3* e per *H(A/3)* con λ *=1*.

  Gli intervalli di confidenza di *H(C)* e *H(C/A=a)* con *a=1,2,3* sono disgiunti per *H(C/3)* con λ *=1,2*.

  I risultati ottenuti per questa coppia sono ragionevoli ed in linea con quanto atteso. Infatti, andando a verificare le entropie condizionate ottenute per $H(A|C=c)$, si evidenzia che per $c=2$ si ottiene un valor medio dell'entropia sensibilmente differente da $H(C)$. Questo è riconducibile alla probabilità di *P(A/2)*, la quale risulta inferiore rispetto a *P(A/1)* e *P(A/3)*, e quindi quando condizionando l'entropia di A con *c=2*, la possibilità di trovare informazioni utili non esplicite è maggiore.

Lo stesso ragionamento può essere fatto per la coppia di entropie $H(C) - H(C|A)$. Una possibile query per l'interrogazione del sistema potrebbe essere:

SELECT A FROM db WHERE c=2

- Coppia B-C
  Gli intervalli di confidenza di $H(B)$ e $H(B|C = c)$ con $c = 1,2,3$, risultano essere disgiunti per $H(B|2)$ con $\lambda = 1,2$.

  Gli intervalli di confidenza di $H(C)$ e $H(C|B=b)$ con $b=1,2,3$ sono disgiunti per $H(C|3)$ con $\lambda = 1,2$.

  I risultati riguardanti questa coppia sono ragionevoli. Infatti controllando le entropie condizionate ottenute si evidenzia che $H(B|C=2)$ si discosta maggiormente da $H(B)$. Questo è dovuto alla probabilità congiunta della coppia di attributi per $c=2$ ha una probabilità decisamente più bassa rispetto agli altri casi.
  Un ragionamento simile può essere fatto per $H(C)$ e $H(C|B)$.

Per quanto riguarda il secondo esperimento, quello in cui la matrice di probabilità congiunta in input è uniforme, si evidenzia che:

- l'analisi condotta sull'intero database ha restituito dei valori ottenuti con l'approccio sperimentale coerenti con quelli ottenuto con l'approccio teorico.

  Inoltre come era prevedibile, mentre la mutua informazione teorica di tutte le possibili coppie di attributi risulta essere nulla, quella sperimentale non è esattamente pari a zero. Questo suo discostamento dal valore ottimale deve essere ricondotto all'incertezza che corrompe i dati.

- l'analisi condotta utilizzando il meccanismo di bootstrap ha evidenziato che, nonostante il minor numero di tuple utilizzate, i valori degli indicatori statistici ottenuti sono in linea con i corrispondenti valori ottenuti con l'analisi sull'intero insieme di tuple.
  L'analisi di sovrapposizione degli intervalli di confidenza tra l'entropia dell'attributo e l'entropia dell'attributo condizionata ai valori dell'altro attributo della coppia, ha prodotto i seguenti risultati:
  - Coppia A-B
    Come ci aspettavamo, dato che tutti gli attributi hanno la stessa probabilità, per la questa coppia si ha sempre sovrapposizione degli intervalli di confidenza di $H(A)$ e $H(A|B=b)$ con $b=1,2,3$ per qualsiasi λ.

    Anche gli intervalli di confidenza di $H(B)$ e $H(B|A=a)$ con $a=1,2,3$ si sovrappongono per qualsiasi λ.

  - Coppia A-C
    Anche per questa coppia le aspettative sono rispettate: c'è sempre sovrapposizione tra gli intervalli di confidenza tra $H(A)$ e $H(A|C=c)$ con $c=1,2,3$.

    Anche gli intervalli di sovrapposizione di $H(C)$ e $H(C|A)$ con $a=1,2,3$, sono sovrapposti in ogni possibile caso analizzato.

  - Coppia B-C
    Il comportamento osservato per le precedenti due coppie è confermato anche per questa coppia: si ha sovrapposizione degli intervalli di confidenza in ogni caso sia confrontando $H(B)$ e $H(B|C=c)$ con $c=1,2,3$, sia confrontando $H(C)$ e $H(C|B=b)$ con $b=1,2,3$.

I risultati ottenuti per questo esperimento sono in linea con quelli attesi. Essendo la matrice di probabilità congiunta in input uniforme, anche le matrici di probabilità congiunta relative alle possibili coppie sono uniformi, e di conseguenza le distribuzioni condizionate delle coppie di attributi non cambiano né al variare del valore assunto dagli attributi né al variare del parametro λ.

Questo è ragionevole, in quanto, se la distribuzione di un attributo non varia al variare del valore assunto dall'altro attributo della coppia si può affermare che i due attributi sono indipendenti, che è quello che si voleva dimostrare.

Per quanto riguarda il terzo esperimento, quello in cui la matrice di probabilità congiunta in input è sbilanciata, si nota che:

- l'analisi condotta sull'intero insieme di tuple del database ha evidenziato che i risultati ottenuti con l'approccio sperimentale sono coerenti con quelli ottenuti con l'approccio teorico.
- l'analisi condotta utilizzando il meccanismo di bootstrap ha mostrato che, nonostante il notevole minor numero di tuple considerate, i valori degli indicatori statistici ottenuti sono in linea con i corrispondenti valori dell'analisi sperimentale ottenuti considerando l'intero insieme di tuple

  L'analisi di sovrapposizione degli intervalli di confidenza tra l'entropia dell'attributo e l'entropia dell'attributo condizionata ai valori dell'altro attributo della coppia, ha prodotto i seguenti risultati:

  - Coppia A-B

    L'analisi di sovrapposizione degli intervalli di confidenza di $H(A)$ e $H(A|B=b)$ con $b=1,2,3$ ha evidenziato che sono sempre disgiunti per qualsiasi valore di λ.

    Anche gli intervalli di confidenza di $H(B)$ e $H(B|A=a)$ con $a=1,2,3$ risultano essere sempre disgiunti per qualsiasi λ considerato.

I risultati ottenuti per questa coppia si differenziano di molto rispetto alle seguenti due coppie di attributi. È infatti possibile notare come per questa coppia non si abbia mai sovrapposizione degli intervalli di confidenza tra le entropie considerate. Questo diverso comportamento, è dovuto alla diversa matrice di probabilità congiunta della coppia di attributi A-B rispetto a quelle delle altre due coppie, la quale è riportata di seguito.

P(A, B)		B		
		1	2	3
A	1	0,300	0,100	0
	2	0	0,300	0
	3	0	0	0,300

Tabella 5.44 Dati sintetici, probabilità congiunta sbilanciata: probabilità congiunta coppia attributi A-B

Come è possibile notare questa matrice presenta per metà delle combinazioni degli attributi un valore nullo.

La presenza di molti valori nulli ha la conseguenza di decrementare la precisione di calcolo e quindi i risultati possono essere diversi rispetto a quelli attesi.

- Coppia A-C

  L'analisi di sovrapposizione degli intervalli di confidenza tra H(A) e $H(A/C=c)$ con $c=1,2,3$ ha evidenziato che essi sono disgiunti solamente nel caso $H(A/2)$ con $\lambda=1$.

  L'analisi degli intervalli di confidenza di $H(C)$ e $H(C/A=a)$ con $a=1,2,3$ ha evidenziato che essi risultano essere disgiunti per $H(C/1)$ con $\lambda=1$.

In questo caso, al contrario di quanto successo per la precedente coppia, nonostante la matrice di probabilità congiunta in input presenti numerosi valori nulli, la matrice di probabilità congiunta relativa della coppia risulta esserne priva.

P(A, C)		C		
		1	2	3
A	1	0,100	0,200	0,100
	2	0,100	0,100	0,100
	3	0,100	0,100	0,100

Tabella 5.45 Dati sintetici, probabilità congiunta sbilanciata: probabilità congiunta coppia A-C

L'assenza di valori nulli da questa matrice, fa sì che il calcolo delle entropie necessarie alla valutazione degli intervalli di sovrapposizione sia decisamente più preciso rispetto al caso della coppia precedente e quindi anche la valutazione di sovrapposizione degli intervalli di confidenza sia più precisa.

- Coppia B-C

  L'analisi di sovrapposizione degli intervalli di confidenza di $H(B)$ e $H(B|C=c)$ con $c=1,2,3$, ha evidenziato che essi sono disgiunti per $H(B|2)$ con $\lambda=1$.

  L'analisi di sovrapposizione degli intervalli di confidenza di $H(C)$ e $H(C|B=b)$, con $b=1,2,3$, ha mostrato che essi sono disgiunti per $H(C|2)$ per $\lambda=1$.

  Come per la coppia di attributi precedenti, anche in questo caso, nonostante la matrice di probabilità congiunta in input presentasse numero valori nulli, la matrice di probabilità congiunta relativa alla coppia di attributi B-C non ne

presenta nemmeno uno, come è possibile vedere dalla seguente tabella.

P(B, C)		C		
		1	2	3
B	1	0,100	0,100	0,100
	2	0,100	0,200	0,100
	3	0,100	0,100	0,100

Tabella 5.46 Dati sintetici, probabilità congiunta sbilanciata: probabilità congiunta coppia B-C

L'assenza di valori nulli permette una migliore stima degli indicatori statistici e di conseguenza una migliore valutazione di sovrapposizione degli intervalli di confidenza delle entropie degli attributi.

A valle di quanto esposto sugli esperimenti fatti finora, le conclusioni che possiamo trarre considerando i risultati ottenuti sul database di dati sintetici, sono che gli indicatori statistici utilizzati per l'analisi sono consistenti e possono quindi essere applicati a database di dati reali, senza il timore che i risultati ottenuti siano errati.

5.2 Analisi database dati medici

5.2.1 Informazioni preliminari

Dopo aver condotto l'analisi sul database di dati sintetici, abbiamo testato la tecnica di database exploration proposta su un database reale di dati medici. Il database è composto da 135 attributi, che assumono valori numerici, alfanumerici o stringhe, e comprende 13235 tuple.

La nostra analisi si concentra su un sottogruppo di attributi numerici e discreti, che sono:

- Reparto;
- Sesso del paziente;
- Giornate di degenza del paziente;
- Età assistito;
- Codice MDC (Major Diagnostic Category).

L'attributo 'Reparto' rappresenta il reparto in cui è stato ricoverato il paziente. Può assumere 48 diversi valori.

L'attributo 'Sesso', rappresenta il sesso del paziente, e può assumere due valori:

- Femmina (F): rappresentato con 1;
- Mascio (M): rappresentato con 2.

L'attributo 'Giornate Degenza', rappresenta il numero di giorni in cui il paziente è stato ricoverato. Per l'analisi non sono stati utilizzati i singoli valori dei giorni, ma sono stati suddivisi in intervalli come segue:

- Da 0 a 10: intervallo 1;
- Da 11 a 20: intervallo 2;
- Da 21 a 30: intervallo 3;
- Da 31 a 40: intervallo 4;
- Da 41 a 50: intervallo 5;
- Da 51 a 60: intervallo 6;
- Da 61 a 70: intervallo 7;
- Da 71 a 80: intervallo 8;
- Da 81 a 90: intervallo 9;
- Da 91 in poi: intervallo 10.

L'attributo 'Età assistito', rappresenta l'età del paziente. Anche in questo caso si è provveduto a sostituire il valore dell'età con la sua fascia d'età di appartenenza:

- Da 0 a 10: fascia età 1;
- Da 11 a 20: fascia età 2;
- Da 21 a 30: fascia età 3;
- Da 31 a 40: fascia età 4;
- Da 41 a 50: fascia età 5;
- Da 51 a 60: fascia età 6;
- Da 61 a 70: fascia età 7;
- Da 71 a 80: fascia età 8;
- Da 81 a 90: fascia età 9;
- Da 91 in poi: fascia età 10.

L'attributo codice MDC sono i gruppi di diagnosi che formano la struttura del sistema di classificazione DRG[5], e può assumere 18 valori diversi.

5.2.2 Descrizione della soluzione basata su two-sample chi square test

Il two sample chi-square test è un test d'ipotesi statistico utilizzato per determinare se c'è una associazione rilevante tra due variabili. Il test viene applicato a attributi o variabili categorici di un singolo campione e rappresenta una delle soluzioni per la tecnica di database exploration suggerita in [1] e descritta in Sezione 3.7.

Più in dettaglio, supponiamo che l'attributo *A* assuma *r* valori, e l'attributo *B* assuma *c* valori. L'ipotesi nulla può essere formulata dicendo che conoscere il valore dell'attributo *A*, non aiuta a predire il valore dell'attributo *B*. Quindi gli attributi sono indipendenti.

[5] DRG (Diagnosis Related Group): il sistema DRG identifica un numero piuttosto elevato di classi finali di ricovero, definite in modo da risultare significative sotto il profilo clinico ed omogenee dal punto di vista delle risorse assorbite e quindi dei costi di produzione dell'assistenza ospedaliera (iso-risorse)

H_0: attributo *A* e attributo *B* sono indipendenti

H_a: attributo *A* e attributo *B* sono dipendenti

L'ipotesi alternativa è quindi che conoscendo il valore dell'attributo A, questo aiuti a predire il valore dell'attributo B.

Se viene accettata l'ipotesi alternativa, i due attributi sono in relazione tra loro.

Il core di questa soluzione è il two-sample chi square test, che è un test di ipotesi in grado di verificare se due set di dati discreti sono stati generati dalla stessa distribuzione (ipotesi nulla) o da differenti distribuzioni (ipotesi alternative). Il test è caratterizzato da un livello di significatività.

Il piano di analisi deve prevedere i seguenti elementi:

- Livello di significatività, è la probabilità di commettere un errore (Type I error). Solitamente viene usato il valore α=0.01
- Metodo di test, usare il chi square test per verificare l'indipendenza dei due attributi.

Il formalismo statistico del chi square test è il seguente:

- **Gradi di libertà** (Degrees of freedom DF), vengono calcolati con la formula

  $DF=(r-1) * (c-1)$

  dove *r* è il numero di valori possibili assunti dal primo attributo, e *c* è il numero di valori possibili assunti dal secondo attributo
- **Frequenze attese**. Il conto delle frequenze attese viene calcolato separatamente per ogni valore di ogni variabile categorica ad ogni valore dell'altra variabile categorica. La formula che viene utilizzata è la seguente:

  $E_{r,c} = (n_r * n_c) / n$

  dove $E_{r,c}$ è il conto delle frequenze attese per il valore *r* dell'attributo *A* e il valore *c* dell'attributo *B*, n_r è il numero totale di osservazioni del campione del valore dell'attributo *A*, n_c è il numero totale di osservazioni del campione del valore c dell'attributo *B*.
- **Test statistico**. Il test utilizzato è

$$X^2 = \sum \left(\frac{\left(O_{r,c} - E_{r,c}\right)^2}{E_{r,c}} \right)$$

dove $O_{r,c}$ è la frequenza osservata dei valori *r* e *c*, e $E_{r,c}$ è la frequenza attesa dei valori *r* e *c*.

- **P-value**. Il p-value è il più basso livello di significatività per cui i dati osservati portano a rifiutare H_0. Ricordiamo che più è basso il livello di significatività e più risulta difficile rifiutare H_0 (cioè ci vogliono prove più evidenti).

Di seguito verranno illustrati i risultati ottenuti sul database dei dati biomedici.

La presentazione dei risultati è strutturata come segue: vengono prima mostrati i risultati ottenuti sull'intero dataset per ogni coppia possibili di attributi considerati.

Dopodiché per le coppie di attributi considerate rilevanti, vengono presentati i risultati ottenuti con l'analisi di sovrapposizione degli intervalli di confidenza e con il two-sample chi square test.

Alla fine della presentazione dei risultati verranno effettuate delle considerazioni sui risultati ottenuti.

5.2.3 Risultati ottenuti

Come abbiamo fatto per il database di dati sintetici, anche in questo caso siamo andati a calcolare gli indicatori statistici sia sull'intero dataset di tuple, sia applicando il bootstrap. Nella seguente tabella sono riportati i valori ottenuti nella prima delle due analisi.

Attributo X	Attributo Y	H(X)	H(Y)	H(X,Y)	H(Y,X)	H(X|Y)	H(Y|X)	I(X; Y)
Reparto	Sesso	3,117574	0,999463	4,0988	4,0988	3,099337	0,981227	0,018236
Reparto	Giornate Deg.	3,117574	1,006853	4,014917	4,014917	3,008065	0,897343	0,109509
Reparto	Età assistito	3,117574	2,371607	5,204885	5,204885	2,833278	2,087312	0,284295
Sesso	Giornate Deg.	0,999463	1,006853	2,00271	2,00271	0,995858	1,003247	0,003605
Sesso	Età assistito	0,999463	2,371607	3,366209	3,366209	0,994603	2,366746	0,004861
Sesso	Cmdc	0,999463	2,844134	3,840085	3,840085	0,995951	2,840621	0,003513
Giornate Deg.	Età assistito	1,006853	2,371607	3,299638	3,299638	0,928031	2,292786	0,078821
Giornate Deg.	Cmdc	1,006853	2,844134	3,720456	3,720456	0,876322	2,713603	0,130531
Età assistito	Cmdc	2,371607	2,844134	4,995313	4,995313	2,151179	2,623706	0,220428

Tabella 5.47 Database dati medici: valori indicatori statistici ottenuti considerando tutte le tuple

Quello che ci interessa maggiormente è però il confronto tra l'analisi degli intervalli di sovrapposizione tra l'entropia del singolo attributo e l'entropia del singolo attributo condizionata ai valori assunti dagli altri attributi e il two-sample chi square test. Verrà fatta ora un'analisi coppia per coppia in modo tale da evidenziare il comportamento di ciascuna di esse.

- **Coppia Reparto-Sesso**

  Per quanto riguarda la coppia *Reparto-Sesso* di particolare interesse risulta essere $H(X|Y=y)$, dove X rappresenta l'attributo *Reparto*, Y rappresenta l'attributo *Sesso* e y sono tutti i possibili valori assunti da Y. Considerando l'intero dataset prima, e la tecnica di bootstrap poi, per $H(X|Y=y)$, abbiamo ottenuto i seguenti risultati:

	H(X|1)	H(X|2)
Intero dataset	2,9992634	3,205023
Bootstrap	2,972777	3,146842

Tabella 5.48 Database dati medici, coppia Reparto-Sesso: valori entropia H(X|Y=y) ottenuti dalla duplice analisi

L'analisi di rilevanza condotta con i due test ha restituito i seguenti risultati:

	Sovrapposizione intervalli			Two-sample Chi square test
	λ=1	**λ=2**	**λ=3**	**HT**
H(X|1)	No	Sì	Sì	Non rilevante
H(X|2)	Sì	Sì	Sì	Non rilevante

Tabella 5.49 Database dati medici, coppia Reparto-Sesso: comparazione risultati ottenuti con i due test per H(X) – H(X|Y=y)

Dalla tabella precedente risulta per l'analisi di sovrapposizione degli intervalli di confidenza, che condizionando l'entropia dell'attributo *Reparto* con il valore *1*, corrispondente al genere femminile, dell'attributo *Sesso*, si provoca la scoperta di nuove informazioni utili.

Quindi la possibile query risultante è:

(Q) "Potrebbe essere interessante esplorare la differente distribuzione dei due *sessi* nei *reparti*"

- **Coppia Reparto-Giornate Degenza**

Per la seconda coppia, quella formata da *Reparto-Giornate Degenza*, di particolare interesse risulta essere *H(X|Y=y)*, dove *X* rappresenta l'attributo *Reparto*, *Y* rappresenta l'attributo *Giornate Degenza* e *y* sono tutti i possibili valori assunti da *Y*. Considerando l'intero dataset prima, e la tecnica di bootstrap poi, per *H(X|Y=y)*, abbiamo ottenuto i seguenti risultati:

	H(X|1)	H(X|2)	H(X|3)	H(X|4)	H(X|5)
Intero dataset	3,010881	3,131473	3,04387	2,374659	2,672032
Bootstrap	2,979957	2,988456	2,698965	2,243768	2,150982

Tabella 5.50 Database dati medici, coppia Reparto-Giornate Degenza: valori entropia H(X|Y=y) ottenuti dalla duplice analisi

Come si può vedere dalla tabella precedente i risultati ottenuti con i due diversi approcci sono coerenti tra loro.

L'analisi di rilevanza condotta con i due test ha restituito i seguenti risultati:

	Sovrapposizione intervalli			**Two-sample Chi square test**
	λ=1	**λ=2**	**λ=3**	**HT**
H(X|1)	No	Sì	Sì	Non rilevante
H(X|2)	Sì	Sì	Sì	Rilevante
H(X|3)	Sì	Sì	Sì	Rilevante
H(X|4)	No	No	Sì	Rilevante
H(X|5)	No	No	Sì	Rilevante

Tabella 5.51 Database dati medici, coppia Reparto-Giornate Degenza: comparazione risultati ottenuti con i due test per H(X) –H(X|Y=y)

I test si comportano in maniera differente per i primi tre valori dell'attributo Y, mentre per i restanti due valori hanno comportamenti identici. Questo potrebbe essere dovuto alla differente granularità di valutazione dei due test. La valutazione di sovrapposizione degli intervalli di confidenza ha una granularità più fine, che permette di stabilire di quanto gli intervalli di confidenza si sovrappongono, mentre il two-sample chi square test ha una granularità di valutazione meno precisa, in quanto indica solamente se i due intervalli sono sovrapposti o no.

Una possibile query potrebbe essere:

(Q) “Potrebbe essere interessante esplorare come sono distribuiti i pazienti, in base *alla durata della degenza*, nei *reparti*”.

- **Coppia Reparto-Età assistito**

  Per quanto riguarda la coppia *Reparto-Età assistito* di particolare interesse risulta essere *H(X|Y=y)*, dove *X* rappresenta l’attributo *Reparto*, *Y* rappresenta l’attributo *Età assistito* e *y* sono tutti i possibili valori assunti da *Y*. Considerando l’intero dataset prima, e la tecnica di bootstrap poi, per *H(X|Y=y)*, abbiamo ottenuto i seguenti risultati:

	H(X|2)	**H(X|3)**	**H(X|4)**	**H(X|5)**	**H(X|6)**	**H(X|7)**	**H(X|8)**	**H(X|9)**
Intero dataset	2,16184	3,11208	3,11358	2,35553	2,46585	2,71028	3,18102	3,41034
Bootstrap	0,74694	2,77596	2,73545	2,08406	2,31521	2,66623	2,95571	3,01035

Tabella 5.52 Database dati medici, coppia Reparto-Età assistito: valori entropia H(X|Y=y) ottenuti dalla duplice analisi

Come si può vedere dalla tabella precedente i risultati ottenuti con i due diversi approcci sono coerenti tra loro. L’unico caso in cui le due entropie assumono valori molto differenti sono *H(X|2)*. Questo è dovuto alla bassa percentuale con cui *Età assistito* assume il valore *2*.

L’analisi di rilevanza condotta con i due test ha restituito i seguenti risultati:

	Sovrapposizione intervalli			Two-sample Chi square test
	λ=1	**λ=2**	**λ=3**	**HT**
H(X|2)	No	No	Sì	Rilevante
H(X|3)	Sì	Sì	Sì	Rilevante
H(X|4)	No	Sì	Sì	Rilevante
H(X|5)	No	No	No	Rilevante
H(X|6)	No	No	No	Rilevante
H(X|7)	No	No	Sì	Rilevante
H(X|8)	Sì	Sì	Sì	Rilevante
H(X|9)	Sì	Sì	Sì	Rilevante

Tabella 5.53 Database dati medici, coppia Reparto-Età assistito: comparazione risultati ottenuti con i due test per H(X) –H(X|Y=y)

Per questa coppia di attributi il two-sample chi square test presenta lo stesso comportamento per qualsiasi valore assunto da *Y*, indica l'attributo *Età assistito* come *'rilevante'* per l'attributo *Reparto*. Per i valori di *y=3,8,9*, secondo l'analisi di sovrapposizione degli intervalli di confidenza, l'attributo *Età assistito* è invece *'non rilevante'* per l'attributo *Reparto*, in quanto gli intervalli di confidenza sono sempre sovrapposti.

- **Coppia Sesso-Giornate Degenza**

Per questa coppia di particolare interesse risulta essere *H(X|Y=y)*, dove *X* rappresenta l'attributo *Sesso*, *Y* rappresenta l'attributo *Giornate Degenza* e *y* sono tutti i possibili valori assunti da *Y*. Considerando l'intero dataset prima, e la tecnica di bootstrap poi, per *H(X|Y=y)*, abbiamo ottenuto i seguenti risultati:

	H(X|1)	**H(X|2)**	**H(X|3)**	**H(X|4)**	**H(X|5)**
Intero dataset	0,9999997	0,981865	1	0,918296	0,999411
Bootstrap	0,9998417	0,966451	0,965573	0,832598	0,904086

Tabella 5.54 Database dati medici, coppia Sesso-Giornate Degenza: valori entropia H(X|Y=y) ottenuti dalla duplice analisi

L'analisi di rilevanza condotta con i due test ha restituito i seguenti risultati:

	Sovrapposizione intervalli			Two-sample Chi square test
	λ=1	**λ=2**	**λ=3**	**HT**
H(X|1)	Sì	Sì	Sì	Non rilevante
H(X|2)	Sì	Sì	Sì	Non rilevante
H(X|3)	Sì	Sì	Sì	Non rilevante
H(X|4)	No	Sì	Sì	Non rilevante
H(X|5)	Sì	Sì	Sì	Non rilevante

Tabella 5.55 Database dati medici, coppia Sesso-Giornate Degenza: comparazione risultati ottenuti con i due test per H(X) –H(X|Y=y)

Per questa coppia di attributi i due test hanno un comportamento analogo, tranne per il valore di *Giornate Degenza* uguale a *4*. Infatti per questo valore l'analisi di sovrapposizione degli intervalli di confidenza rivela che, per *λ=1*, l'attributo *Giornate Degenza* è *'rilevante'* per l'attributo *Sesso*.

- **Coppia Sesso-Età assistito**

  Per questa coppia di particolare interesse risulta essere *H(X|Y=y)*, dove *X* rappresenta l'attributo *Sesso*, *Y* rappresenta l'attributo *Età assistito* e *y* sono tutti i possibili valori assunti da *Y*. Considerando l'intero dataset prima, e la tecnica di bootstrap poi, per *H(X|Y=y)*, abbiamo ottenuto i seguenti risultati:

	H(X|2)	**H(X|3)**	**H(X|4)**	**H(X|5)**	**H(X|6)**	**H(X|7)**	**H(X|8)**	**H(X|9)**
Intero dataset	0,99538	0,98704	0,99549	0,99345	0,99648	0,99832	0,93983	0,99569
Bootstrap	0,89579	0,94448	0,97993	0,98474	0,99576	0,99605	0,93835	0,99523

Tabella 5.56 Database dati medici, coppia Sesso-Età assistito: valori entropia H(X|Y=y) ottenuti dalla duplice analisi

L'analisi di rilevanza condotta con i due test ha restituito i seguenti risultati:

	Sovrapposizione intervalli			Two-sample Chi square test
	λ=1	**λ=2**	**λ=3**	**HT**
H(X|2)	Sì	Sì	Sì	Non rilevante
H(X|3)	Sì	Sì	Sì	Non rilevante
H(X|4)	Sì	Sì	Sì	Non rilevante
H(X|5)	Sì	Sì	Sì	Non rilevante
H(X|6)	Sì	Sì	Sì	Non rilevante
H(X|7)	Sì	Sì	Sì	Non rilevante
H(X|8)	Sì	Sì	Sì	Non rilevante
H(X|9)	Sì	Sì	Sì	Non rilevante

Tabella 5.57 Database dati medici, coppia Sesso-Età assistito: comparazione risultati ottenuti con i due test per H(X) –H(X|Y=y)

In questo caso i due test hanno per tutti i possibili valori di *Y* lo stesso comportamento, dando per ogni possibilità *'Non rilevante'*. Questo sta a significare che condizionando l'entropia dell'attributo *Sesso* con i valori dell'attributo *Età assistito*, non si ha alcun vantaggio.

- **Coppia Sesso-Cmdc**

Per quanto riguarda la coppia *Sesso-Cmdc* entrambi i test hanno rivelato che l'attributo *Cmdc* risulta *'non rilevante'* per l'attributo *Sesso*. Questo indica che non vi è alcun vantaggio nell'andare ad esplorare l'uno o l'altro cercando delle relazioni con il secondo attributo.

- **Coppia Giornate Degenza-Età assistito**

Per la coppia *Giornate Degenza-Età assistito* entrambe le entropie condizionate al valore assunto dall'altro attributo della coppia risultano essere interessanti. Per quanto riguarda i valori ottenuti per *H(X|Y=y)*, dove *X* rappresenta l'attributo *Giornate Degenza*, *Y* rappresenta

l'attributo *Età assistito* e *y* sono tutti i possibili valori di *Y*, essi sono riassunti nella seguente tabella:

	H(X|2)	**H(X|3)**	**H(X|4)**	**H(X|5)**	**H(X|6)**	**H(X|7)**	**H(X|8)**	**H(X|9)**
Intero dataset	0,52936	0,51238	0,65662	1,24887	0,87406	0,79348	1,31553	1,10097
Bootstrap	0,07219	0,40594	0,57587	1,27255	0,90039	0,78786	1,36262	1,10548

Tabella 5.58 Database dati medici, coppia Giornate Degenza-Età assistito: valori entropia H(X|Y=y) ottenuti dalla duplice analisi

Come si può vedere dalla tabella precedente i risultati ottenuti con i due diversi approcci sono coerenti tra loro. L'unico caso in cui i valori ottenuti sono veramente discordanti è per *y=2*.

L'analisi di rilevanza condotta con i due test ha restituito i seguenti risultati:

	Sovrapposizione intervalli			**Two-sample Chi square test**
	λ=1	**λ=2**	**λ=3**	**HT**
H(X|2)	No	No	No	Non rilevante
H(X|3)	No	No	Sì	Rilevante
H(X|4)	No	No	No	Non rilevante
H(X|5)	No	Sì	Sì	Rilevante
H(X|6)	No	Sì	Sì	Rilevante
H(X|7)	No	No	Sì	Non rilevante
H(X|8)	No	Sì	Sì	Rilevante
H(X|9)	Sì	Sì	Sì	Non rilevante

Tabella 5.59 Database dati medici, coppia Giornate Degenza-Età assistito: comparazione risultati ottenuti con i due test per H(X) –H(X|Y=y)

Come è possibile osservare dalla tabella precedente l'analisi di sovrapposizione degli intervalli restituisce che i due attributi sono *'rilevanti'* per tutti i valori di *Età assistito*, fatta eccezione per il valore *9* per il quale gli attributi risultano indipendenti. Il two-sample chi square test, invece, indica che per *y=2,4,7,9* gli attributi sono indipendenti. Questa distinzione è dovuta alla diversa granularità con cui i due test

valutano la dipendenza o meno di un attributo. Essendo l'approccio che valuta la sovrapposizione degli intervalli di confidenza parametrizzabile attraverso il valore assunto da λ, esso permette di riuscire a capire meglio l'andamento delle entropie. Invece il two-sample chi square test, restituisce solo *'rilevante'* o *'non rilevante'*, senza la possibilità di andare a esplorare meglio i dati.

Una possibile query potrebbe essere:

(Q) "Potrebbe essere interessante scoprire la *durata della degenza* dei pazienti in base alla loro *età*".

Passando ad analizzare i valori ottenuti per *H(Y|X=x)*, dove *Y* rappresenta l'attributo *Età assistito*, *X* rappresenta l'attributo *Giornate Degenza* e *x* sono tutti i possibili valori di *X*, essi sono riassunti nella seguente tabella:

	H(Y|1)	**H(Y|2)**	**H(Y|3)**	**H(Y|4)**	**H(Y|5)**
Intero dataset	2,3430303	2,396235	1,669781	1,538862	1,382028
Bootstrap	2,3386376	2,341527	1,630407	1,387099	1,291793

Tabella 5.60 Database dati medici, coppia Giornate Degenza-Età assistito: valori entropia H(Y|X=x) ottenuti dalla duplice analisi

Come si può vedere dalla tabella precedente i risultati ottenuti con i due diversi metodi sono coerenti tra loro, data la differenza ridotta tra i due valori.

L'analisi di rilevanza condotta con i due test ha restituito i seguenti risultati:

	Sovrapposizione intervalli			Two-sample Chi square test
	λ=1	**λ=2**	**λ=3**	**HT**
H(Y|1)	Sì	Sì	Sì	Rilevante
H(Y|2)	Sì	Sì	Sì	Rilevante
H(Y|3)	No	No	No	Rilevante
H(Y|4)	No	No	No	Rilevante
H(Y|5)	No	No	No	Rilevante

Tabella 5.61 Database dati medici, coppia Giornate Degenza-Età assistito: comparazione risultati ottenuti con i due test per H(Y) –H(Y|X=x)

In questo caso i due approcci si comportano in maniera completamente diversa per i valori dell'attributo *Giornate Degenza* pari a *1* e *2*, mentre hanno un comportamento analogo per i restanti valori. Consultando i valori delle entropie ottenute su tutto il dataset e con il bootstrap, si può notare che la differenza sia di *H(X|1)* sia *H(X|2)* tra i due approcci, non è elevata. Quindi anche in questo caso la differenza nei risultati ottenuti dai due test può essere ricondotta alla diversa granularità di valutazione.

Potrebbe risultare interessante interrogare il sistema attraverso la seguente query:

(Q) "Potrebbe essere interessante verificare la distribuzione *età* dei pazienti in base alla *durata della loro*".

- **Coppia Giornate Degenza-Cmdc**

Per quanto riguarda la coppia di attributi *Giornate Degenza-Cmdc* di particolare interesse risulta essere *H(Y|X=x)*, dove *Y* rappresenta l'attributo *Cmdc*, *X* rappresenta l'attributo *Giornate Degenza* e x sono tutti i possibili valori assunti da *X*. Considerando l'intero dataset prima, e la tecnica di bootstrap poi, per *H(Y|X=x)*, abbiamo ottenuto i seguenti risultati:

	H(Y|1)	H(Y|2)	H(Y|3)	H(Y|4)	H(Y|5)
Intero dataset	2,7294301	2,896622	2,350209	2,222486	1,92156
Bootstrap	2,7400389	2,750176	2,238457	2,083345	1,675202

Tabella 5.62 Database dati medici, coppia Giornate Degenza-Cmdc: valori entropia H(Y|X=x) ottenuti dalla duplice analisi

Come si può vedere dalla tabella precedente i risultati ottenuti con i due diversi metodi sono coerenti tra loro.

L'analisi di rilevanza condotta con i due test ha restituito i seguenti risultati:

	Sovrapposizione intervalli			**Two-sample Chi square test**
	λ=1	**λ=2**	**λ=3**	**HT**
H(Y|1)	No	Sì	Sì	Rilevante
H(Y|2)	Sì	Sì	Sì	Rilevante
H(Y|3)	No	No	Sì	Rilevante
H(Y|4)	No	No	Sì	Rilevante
H(Y|5)	No	No	No	Rilevante

Tabella 5.63 Database dati medici, coppia Giornate Degenza-Cmdc: comparazione risultati ottenuti con i due test per H(Y) –H(Y|X=x)

Per questa coppia l'unico caso in cui i due test hanno un valore differente è per *x=2*, per il quale l'analisi di sovrapposizione degli intervalli di confidenza indica l'attributo *Giornate Degenza* come *'non rilevante'* per l'attributo *Cmdc*, mentre il two-sample chi square test indica che è *'rilevante'*. Nei restanti casi essi si comportano nello stesso modo, indicando l'attributo *Giornate Degenza* come *'rilevante'* per l'attributo *Cmdc*.

Una possibile query con cui interrogare il sistema potrebbe essere:

(Q) "Potrebbe essere interessante esplorare la distribuzione del *codice MDC* in base al *periodo di degenza* dei pazienti".

- **Coppia Età assistito-Cmdc**

  Per quanto riguarda la coppia di attributi *Età assistito-Cmdc,* di particolare interesse risulta essere *H(Y|X=x)*, dove *Y* rappresenta l'attributo *Cmdc*, *X* rappresenta l'attributo *Età assistito* e x sono tutti i possibili valori assunti da *X*. Considerando l'intero dataset prima, e la tecnica di bootstrap poi, per *H(Y|X=x)*, abbiamo ottenuto i seguenti risultati:

	H(Y|2)	**H(Y|3)**	**H(Y|4)**	**H(Y|5)**	**H(Y|6)**	**H(Y|7)**	**H(Y|8)**	**H(Y|9)**
Intero dataset	1,47484	2,63779	2,59961	2,23124	2,4113	2,85542	2,82150	2,71117
Bootstrap	0,81326	2,44900	2,42229	2,06312	2,3231	2,85041	2,58629	2,58275

Tabella 5.64 Database dati medici, coppia Età assistito-Cmdc: valori entropia H(Y|X=x) ottenuti dalla duplice analisi

Come si può vedere dalla tabella precedente i risultati ottenuti con il bootstrap sono coerenti con quelli ottenuti sull'intero dataset. Anche in questo l'unico caso in cui i valori sono veramente differenti è *H(Y|2)*.

L'analisi di rilevanza condotta con i due test ha restituito i seguenti risultati:

	Sovrapposizione intervalli			**Two-sample Chi square test**
	λ=1	**λ=2**	**λ=3**	**HT**
H(Y|2)	No	No	Sì	Rilevante
H(Y|3)	No	Sì	Sì	Rilevante
H(Y|4)	No	Sì	Sì	Rilevante
H(Y|5)	No	No	Sì	Rilevante
H(Y|6)	No	No	No	Rilevante
H(Y|7)	Sì	Sì	Sì	Rilevante
H(Y|8)	Sì	Sì	Sì	Rilevante
H(Y|9)	No	Sì	Sì	Rilevante

Tabella 5.65 Database dati medici, coppia Età assistito-Cmdc: comparazione risultati ottenuti con i due test per H(Y) –H(Y|X=x)

I due test si comportano in ugual maniera in tutti casi, fatti eccezione per *y= 7,8.* Per questi due valori il two-sample chi square test indica che l'attributo *Età assistito* è *'rilevante'* per l'attributo *Cmdc*, mentre l'analisi di sovrapposizione degli intervalli di confidenza indica che *non* è *rilevante*.

Una possibile query per questa coppia potrebbe essere:

(Q) "Potrebbe essere interessante esplorare la distribuzione del *codice MDC* in base all'*età* dei pazienti".

5.2.4 Osservazioni finali sui risultati

Alla fine dell'elaborazione è possibile osservare che le due tecniche confrontate, la valutazione di sovrapposizione degli intervalli di confidenza e il two-sample chi square test, presentano nella maggior parte dei casi lo stesso comportamento. Tuttavia in alcuni casi i risultati ritornati dalle due tecniche sono differenti, a volte anche di molto. Questo può essere riconducibile alla diversa granularità di valutazione dei due test.

La valutazione di sovrapposizione degli intervalli di confidenza risulta avere una granularità più fine, in quanto varia al variare del parametro λ. Grazie a questo è possibile anche valutare quanto gli intervalli di confidenza sono sovrapposti tra di loro.

Il two-sample chi square test ha una granularità meno fine rispetto alla valutazione di sovrapposizione degli intervalli di confidenza. Esso è infatti in grado di restituire solamente se i due attributi sono o non sono indipendenti.

L'utilizzo della tecnica di bootstrap permette inoltre una notevole diminuzione delle tuple considerate per l'analisi. Nonostante il numero inferiore di tuple considerate, i valori degli indicatori statisti ottenuti utilizzando il bootstrap e il campionamento casuale con ripetizione,

possono essere considerati validi e coerenti con quelli ottenuti sull'intero dataset.

Per quanto riguarda le coppie di attributi analizzati, è possibile osservare che sono poche le coppie per le quali il condizionamento dell'entropia di uno dei due attributi con i valori dell'altro attributo della coppia, non porti la scoperta di informazioni utili.

Per la maggior parte delle coppie, invece, è possibile osservare come il condizionamento dell'entropia di un attributo con uno o più valori dell'altro attributo della coppia, provoca una variazione della distribuzione dell'entropia tale che utili informazioni (che prima non erano esplicite) possono essere estratte dal database e mostrate all'utente.

Capitolo 6

Conclusioni

In questo lavoro di tesi è stata presentata una nuova tecnica per l'esplorazione di basi di dati di grandi dimensioni basata su entropia. Il calcolo dell'entropia degli attributi viene fatto considerando un meccanismo di bootstrap e di campionamento, che permettono di analizzare database di grandi dimensioni andando a selezionare un sottoinsieme di tuple di cardinalità molto inferiore rispetto al numero totale delle tuple del database. Terminate le iterazioni del meccanismo di bootstrap vengono calcolati gli intervalli di confidenza delle entropie considerate e confrontati per identificare gli attributi e/o i valori rilevanti.

Per poter applicare la tecnica proposta a basi di dati di grandi dimensioni si è dovuto prima effettuare una vasta campagna di esperimenti basata su database di dati sintetici. Questo ci ha permesso di valutare il comportamento degli indicatori statistici utilizzati, ed, in particolar modo, il comportamento di entropia e entropia condizionata.

Una volta provata la solidità degli indicatori statistici utilizzati, si è proceduto all'applicazione della tecnica basata su entropia su un database reale di dati biomedici. Per poter applicare la tecnica a questa base di dati, si è dovuta prima effettuare una fase di preparazione dei dati, andando ad identificare eventuali valori mancanti.

Terminata questa fase di pre-processing dei dati, abbiamo applicato la tecnica basata su entropia al database di dati biomedici, la quale ci ha rivelato che la tecnica proposta si comporta in maniera decisamente più precisa rispetto al two-sample chi square test, migliorando fortemente la database exploration. Infatti essa permette di individuare in maniera più precisa e affidabile le informazioni rilevanti e non esplicite in una base di dati, selezionando gli attributi che possono essere rilevanti per altri attributi e individuando per gli

attributi considerati rilevanti anche i valori di essi che portano alla scoperta di queste nuove informazioni

6.1 Sviluppi futuri

La tecnica di esplorazione dei dati basata su entropia proposta in questa tesi può essere vista come un punto di partenza per migliorare l'esplorazione delle basi di dati di grandi dimensioni, per la scoperta di informazioni utili e non esplicite.

La tecnica proposta in questo lavoro potrebbe essere integrata al metodo di data exploration basato sui test di ipotesi presentato in [1].

Un'estensione di questo lavoro è l'esplorazione intelligente delle coppie di attributi della base di dati e dei loro valori, su cui applicare la tecnica proposta al fine di evitare l'analisi di tutte le combinazioni tra le coppie di attributi.

Infine, si può pensare di considerare indicatori statistici differenti rispetto all'entropia, andando poi a considerare questi indicatori all'interno del meccanismo di bootstrap proposto.

Sarebbe di estremo interesse, inoltre, la costruzione di un prototipo per il testing della tecnica proposta in condizioni scenari/applicazioni reali come ad esempio i dati provenienti da reti di sensori/attuatori per monitoraggio, i dati provenienti da social media e social networks, i dati provenienti dal traffico web.

Bibliografia

[1] Marcello Buoncristiano, Giansalvatore Mecca, Elisa Quintarelli, Manuel Roveri, Donatello Santoro, and Letizia Tanca. *Exploratory Computing: What is there for the Database Researcher?*

[2] Marcello Buoncristiano, Giansalvatore Mecca, Elisa Quintarelli, Manuel Roveri, Donatello Santoro, Letizia Tanca. *Database Challenges for Exploratory Computing*

[3] B. Kavšek, Nada Lavrač, and V. Jovanoski. *Apriori-sd: adapting association rule learning to subgroup discovery*.
In Proc. 5th Int. Symp. On Intelligent Data Analysis, pages 230 – 241. Springer-Verlag, 2003.

[4] Nada Lavrač, Branko Kavšek, Peter Flach, and Ljupčo Todorovski. *Subgroup Discovery with CN2-SD.*
Journal of Machine Learning Research 5 (2004) pp.153-188

[5] M. Atzmueller and F. Puppe. *Sd-map: a fast algorithm for exhaustive subgroup discovery*.
In Proc. PKDD 2006, volume 4213 of LNAI, pages 6 – 17. Springer-Verlag, 2006.

[6] Franciso Herrera, Cristóbal José Carmona, Pedro González, María José del Jesus. *An overview on subgroup discovery: foundations and applications*
Springer-Verlag London Limited 2010 pp.495-525

[7] P. Hanrahan. *Analytic database technologies for a new kind of user: the data enthusiast*. In SIGMOD, pages 577–578, 2012.

[8] K. Morton, M. Balazinska, D. Grossman, and J. D. Mackinlay. *Support the data enthusiast: Challenges for next-generation data-analysis systems*. PVLDB, 7(6):453–456, 2014.

[9] W. Klösgen. Explora: *A multipattern and multistrategy discovery assistant. In Advancesin Knowledge Discovery and Data Mining*, pages 249–271. 1996.

[10] J. W. Tukey. *Exploratory data analysis*. Addison-Wesley, Reading, MA, 1977.

[11] N. D. Blas, M. Mazuran, P. Paolini, E. Quintarelli, and L. Tanca. *Exploratory computing: a challenge for visual interaction*. In AVI, pages 361–362, 2014.

[12] E. L. Lehmann and J. P. Romano. *Testing statistical hypotheses*. Springer Science & Business Media, 2006.

[13] Mathworks. *Information Theory Toolbox*