On the Anonymization of Sparse High-Dimensional Data

# Abstract

Proporre un nuovo metodo di anonimizzazione di dati sparsi ad alta dimensionalità, siccome le classiche tecniche come la k-anonymity e la l-diversity non si può usare. Si impiegherà una particolare rappresentazione che cattura la correlazione nei dati sottostanti e facilita la formazione di gruppi anonimizzati che basse perdite di informazione. Una tecnica di anonimizzazione basta su questa rappresentazione.

# Introduzione

Il problema della preservazione della privacy è un problema che è diventato molto caldo negli ultimi anni. Molti lavori sono formulati nel seguente contesto: diverse organizzazioni, come ospedali, pubblicano dati dettagliati sui loro pazienti per scopi di ricerca o statistici. Però alcune informazioni personali sensibili possono essere scoperte in questo processo, a causa dell’esistenza dei valori nei QID, come età, zip. Un attaccante può unire QID con conoscenza esterna e cercare di identificare record individuali.

Ma in questo caso la dimensionalità dei dati è piccola, e in altri cosai l’anonimizzazione dei dati personali non è sufficiente in alcune applicazioni, per esempio una grande compagnia che verde migliaia di prodotti differenti, e ha numerose transazioni giornaliere. Il grosso ammontare di dati transazionali possono contenere pattern dei clienti e trend che sono essenziali per il marketing e per scopi di pianificazione.

La compagnia potrebbe desiderare rendere i dati disponibili ad una terza parte la quale può processare i dati e estrarre pattern interessanti. Dal momento che lo scopo più probabile è quello di inferire le tendenze di acquisto, caratterizzate dalla correlazione tra i prodotti acquistati, e quindi i dati personali non sono rilevanti, vengono eliminate.

I diversi metodi classici come k-anonimity o l-diversity o Anatomy, non vanno bene perché si perde l’utilità dei dati oppure la perdita di alcuni pattern sono fatti indesiderabili.

Quindi viene proposta una tecnica di anonimizzazione la quale combina i vantaggi della generalizzazione e della permutazione e prova ad affrontare il problema dell’alta dimensionalità.

Per prima cosa 🡪 nuova rappresentazione dei dati che prende i vantaggi della sparsità di essi e preserva la correlazione. I dati sono organizzati come una **band matrix,** effettuando permutazioni di row e columns nella tabella originale tale che la maggior parte delle voci non zero sono vicine alla diagonale, il vantaggio è che le righe vicine hanno un’alta correlazione, infatti condividono un sacco di items comuni.

Dopodiché viene proposta una heuristica efficiente per creare gruppi di alta qualità, che ha solo bisogni di raggruppare insieme le transazioni vicine, quindi riducendo lo spazio di ricerca della soluzione. Quindi questa fase di formazione dei gruppi per QID simili costruisce gruppi anonimi che preservano la correlazione.

In ogni gruppo anonimizzato, gli items sensibili sono separati dai QID, e pubblicati in una tabella riassuntiva separata.

Il contributo specifico degli scrittori del paper è stato:

* Introdurre una nuova rappresentazione dei dati transazionali la quale prende i vantaggi dei dati sparsi e preserva la correlazione tra gli item e organizza le transazioni con QID simili vicini tra loro.
* Una euristica efficiente per creare gruppi anonimizzati con bassa perdita di informazione.
* Valutare sperimentalmente i risultati ottenuti con ds reali, e vedere le performance in data-utility e overhead computazionale.

# II – Background

Il nostro obiettivo è anonimizzare dati consistente di un insieme di transazioni . Ogni transazione contiene items da un insieme di Item . I dati vengono rappresentati come una matrice binaria A con n rows e d columns. Tra i dati ci sono anche alcuni dati sensibili.

*Definizione 1 (Sensitive Items)*, L’insieme di items che rappresentano una privacy threat se associata ad una certa transizione, costituiscono l’insieme degli items sensibili , il resto degli item in I come wine, cream,… sono non sensibili, nel senso che associati con un certo individuo non sono dannosi, un attaccante non può usarli per identificare la transazione di un individuo.

*Definition 2 (Quasi-Identifier Items)* , insieme di items in I che un attaccante può guadagnare conoscenza al fine di identificare una transazione individuale costituita dall’insieme dei QI. Potenzialmente ogni non sensitive items è un QI, quindi

Denotiamo una transazione che contiene items da S come sensibile e una che ne contiene solo da Q come non sensibile.

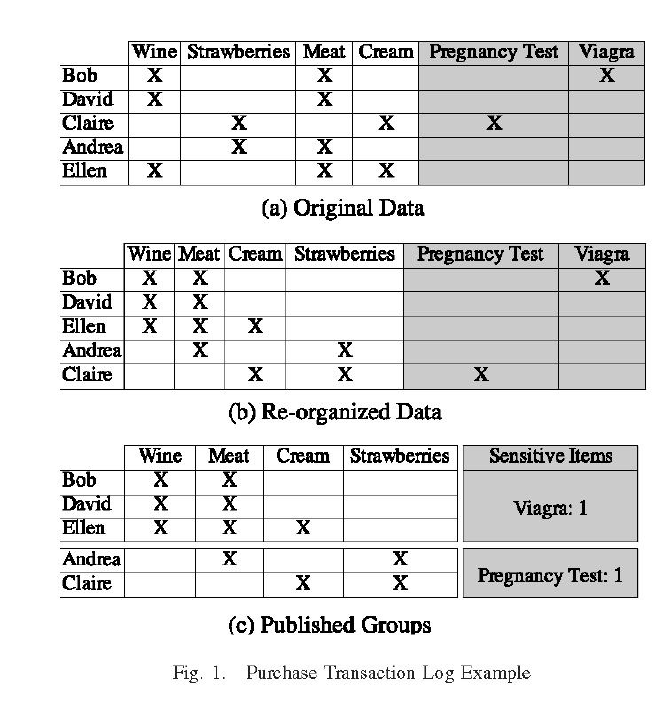
Ogni transazione può essere re identificata sulla base di un numero piccolo di items. Per una data transizione , definiamo *relevent quasi-identifier* come l’intersezione di Q con gli items in t, cioè gli item presenti nella transazione t.

Nei dati transazionali la minaccia della privacy viene definita come l’associazione di una transazione individuale ad un item sensibile. Ma la differenza tra i database relazionali è che lì tutti i record hanno lo stesso numero di valori sensibili (di solito 1), nel nostro caso invece alcune transazioni possono non contenere affatto nessun item sensibile e quindi essere innocue oppure contenerne più di uno.

*Definizione 3 (Privacy)* Una privacy-preserving transformation di un insieme di transazioni T hanno un grado di privacy p se la probabilità di associare una qualunque transizione con un particolare dato sensibile non è maggiore di

L’associazione di un individuo con un item in Q non rappresenta privacy breach.

Noi facciamo rispettare i requisiti delle privacy partizionando l’insieme T in insieme disgiunti di transazioni, al quale ci riferiamo come *anonymized groups.* Per ogni gruppo G, noi pubblichiamo esattamente i QID, insieme ad un summary delle frequenze degli items sensibili contenuti in G.



Nel nostro esempio il secondo gruppo contiene due transizioni, corrispondente ad Andrea e Claire, e una occurenze dell’attributo sensibile *Pregnancy Test.* La probabilità di associare ogni transizione in G a quell’items è ½. In generale, sia il numero delle occorrenze degli items sensibili in un gruppo G.

Quindi il gruppo G offre il seguente grado di privacy :

Quindi dell’intero partizionamento P di T è:

I dati transazionali sono principalmente usati per derivare certi pattern.

Tornando all’esempio possiamo vedere che ci sono due tipi di pattern 🡪 quelli che involvono solo gli items Q e quelli che involvono almeno un item in S.

Per formare le categorie, grazie alle permutazioni basate sul metodo che adotteremo, le informazioni derivate dai dati anonimizzati deve essere identica ai dati originali. Per esempio, noi possiamo derivare dall’esempio sopra che metà delle persone che comprano *strawberry*  hanno comprato anche *cream.*

D’altra parte, quando i dati sensibili sono coinvolti i dati sensibili , le informazioni derivate dai dati anonimizzati è solo una stima di quella reale.

Nel nostro esempio, nella figura 1.c possiamo inferire con il 50% di probabilità chi compra sia cream che strawberry comprerà anche *pregnancy test.* E nei dati originali lo possiamo inferire con il 100% di probabilità.

I pattern possono essere espressi come query nella forma:

Il processo di stima dei resultati delle query per ogni gruppo anonimizzato G, viene referred come *data recostruction.*

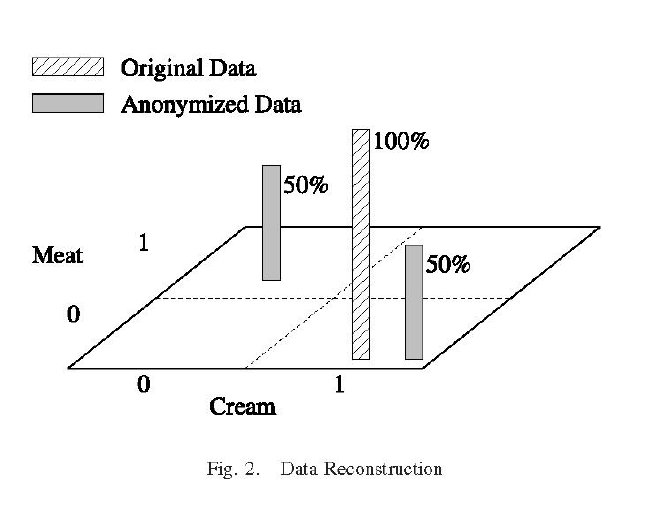
Denotiamo il numero di occorrenze dell’item *s* in *G* con *a,* e il numero di transazioni che matchano la selezione dei QID (vedi query) con *b.* Quindi il risultato stimato della query, assumendo ogni permutazione di elementi sensibili a ciascuna combinazione QID in G sono equiprobabili, equivale:

Notiamo che, se tutte le transazioni in G hanno identici QID, quindi in entrambi i casi b = |G| o b=0 e l’errore di ricostruzione è 0.

Idealmente, per minimizzare l’errore di ricostruzione, abbiamo bisogno di minimizzare |G| - b, quindi includere in ogni gruppo di transazioni avere simili, e quando possibili identici, QID.

Un modo significativo di modellare tali domande che involvono gli item sensibili è usare le funzioni di distribuzione di probabilità (pdf) di un item s sullo spazio definito dal numero di r items in Q.

Nell’esempio precedente assumiamo che gli analisti hanno il desiderio di trovare la correlazione tra *pregnancy test* e i QID *cream e meat.* Nella figura 2 rappresenta questo scenario: ogni cella corrisponde alla combinazione degli items QID, per l’istanza (1,0) corrisponde alla transazione la quale contiene *cream* ma non *meat*, mentre (1,1) comprende entrambe.

Dai dati originali possiamo inferire che tutti i clienti che comprano *cream* ma non *meat* hanno anche comprato *pregnancy test*. Mentre nei dati anonimizzati possiamo solo dire che metà di quei clienti hanno comprato un *pregnancy test.*

Se la query da valutare include *r* QID, il numero totale di celle è , corrispondente a tutte le combinazioni possibili di quando un item è presente oppure no (vedi r bit).

L’attuale valore pdf di un dato sensibile s in una cella C è:

T indica tutte le transazioni.

La stima delle pdf viene calcolata similarmente, eccezione del numeratore che consiste in sommato su tutti i gruppi che intersecano la cella C.

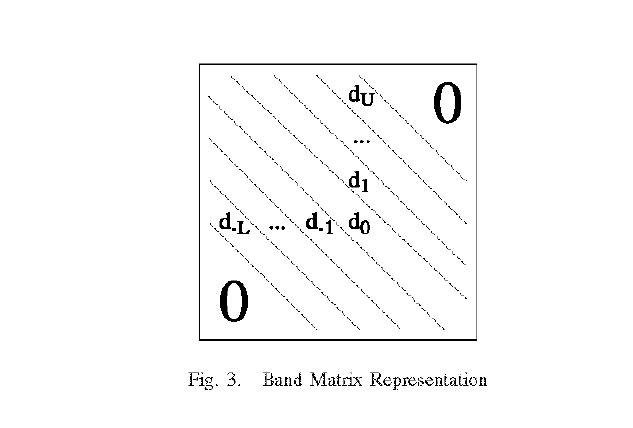
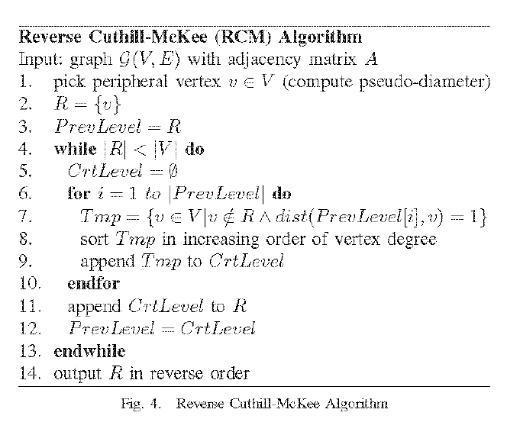
Quindi valutiamo l’utilità dei dati anonimizzati come la distanza tra la vera e le pdf stimate su tutte le celle, misurata dal , come una metrica per valutare il totale delle informazioni perse dopo avere eseguito l’anonimizzazione dei dati:

Se sono identiche Act e Est 🡪 la KL\_Divergence = 0.

Avendo ora dato chiarito i problemi riguardo alla privacy e all’utilità, ora definiamo **Problem Statement.** Dato un insieme di transazioni T contenenti items from I, e un sottoinsieme di items sensibili, determinare una partizione P di T all’interno dei gruppi anonimizzati con grado di privaci almeno p, tale che l’errore di ricostruzione, misurato da KL\_Divergence sia minimizzato.

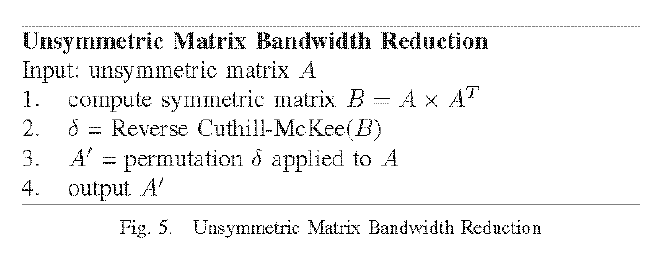
# III – Reduction to band matrix representation

Per minimizzare l’errore di ricostruzione, è necessario raggruppare insieme transazione che simili rilevati QID. Organizziamo la matrice A, dei dati, come una *band matrix,* tale che le righe consecutive condividono un grande numero comune di items. E’ un buon modo per rappresentare i dati sparsi in diversi contesti.

Una *band matrix,* ha come forma generale la seguente, dove tutti gli elementi sono 0, ad eccezione per la diagonale principale , un numero di diagonali superiori U e L diagonali più bassi . Dove U rappresenta *upper bandwidth* della matrice e L *lower bandwidth.* Il nostro obiettivo è minimizzare la *total bandwidth* . Trovare una matrice ottima band con un minimo B è NP-complete.

**The Reverse Cuthill-McKee Algorithm.** Una matrice generale può essere trasformata in una band matrix attraverso la permutazione di rows and columns. La migliore euristica e la più promettente è la Reserve Cuthill-McKee algorithm (RCM) una variante dell’algoritmo Cuthill-McKee. RCM lavora per matrici quadrate e simmetriche.

Data una matrice sparsa A, viene costruito un grafo G=(V,E) dove V contiene un vertice per ogni riga della matrice, e c’è un un arco dal vertice al vertice per ogni elemento della matrice non zero . Se A è simmetrica allora G è unidirezionale. RCM è basato sull’osservazione che una permutazione di una riga in A corrisponde ad una re-labeling dei vertici in G. Data una re-labeling (permutazione) di V, la bandwith di G (quindi la matirce A con le righe permutate in accordo a è :

Per determinare una permutazione che riduce B, RCM esegue una breadth-first (BFS) trasversale partendo inizialmente scegliendo un nodo root. Tutti i nodi (le rows) alla stessa distanza dal root nella traversa costituisce a *level set*. Ad ogni step, i vertici nello stesso livello che condividono lo stesso parent sono ordinati crescentemente in accordo con il grado del vertice. Reversando l’ordine ottenuto, troviamo la permutazione che devono essere applicate alle righe della matrice. La selezione del nodo radice è essenziale per l’efficienza della trasformazione. Di solito viene determinato trovando il pseudo-diametro del grafo.

RCM è correntemente implementato nelle librerie di routine delle matrici, come in MATLAB. La complessità computazionale di RCM è dove D è il massimo grado di ogni vertice nella lista delle adiacenze.

**Bandwith Reduction for Unsymmetric Matrices**

RCM funziona solo per matrici simmetriche, recenti lavori hanno cercato investigato su diversi approcci per ridurre la bandwith di una matrice non simmetrica. Due metodi possono essere usati per ottenere questo data una matrice A, si applica RCM ad una delle seguenti matrici simmetriche:

E si ottiene la permutazione da applicare ad A.

Il metodo 1 è applicabile quando la matrice A è quasi simmetrica, è meno espensivo, ma il risultato ottenuto non è di buona qualità, specialmente se la matrice iniziale è lontana da essere simmetrica.

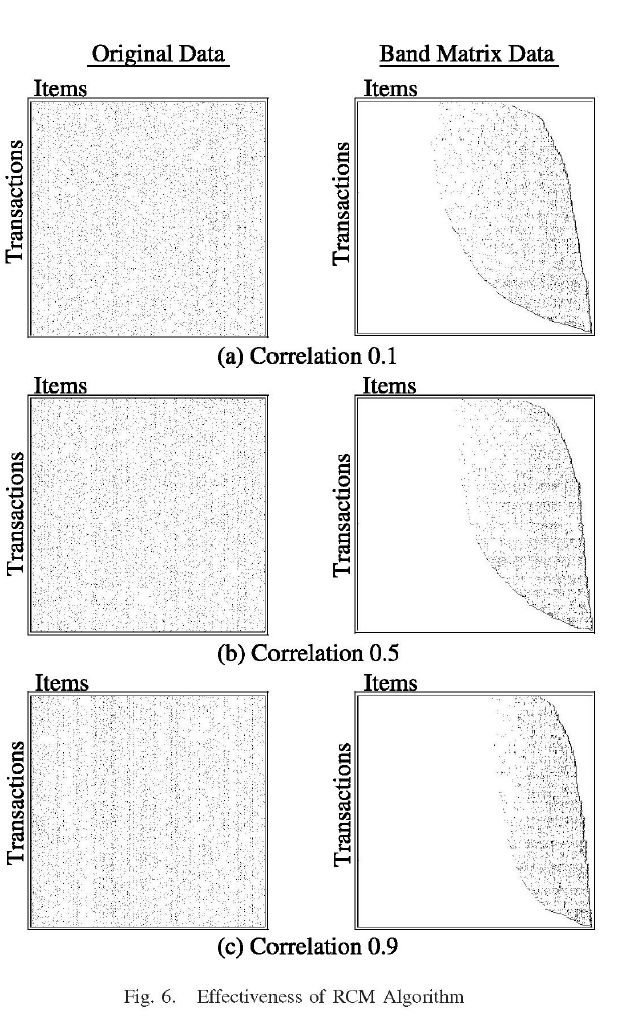
Il metodo 2 può essere applicata ad ogni matrice arbitraria, la moltiplicazione introduce dell’overhead computazionale ma il risultato ottenuto è di buona qualità. Essendo comunque interessati alle entries non zero, l’overhead non e sostanziale.

Nella figura 5 è presente la reduction matrix bandwith per una matrice non simmetrica.

**Valutazione del RCM**

Abbiamo usato dati sintetici per testare l’algoritmo RCM, controllando la correlazione di essi. Si è usato *IBM Quest Market-Basket Sythetci Data Generato* per ottenere un dataset e utilizzato MatView per visualizzare la matrice sparsa.

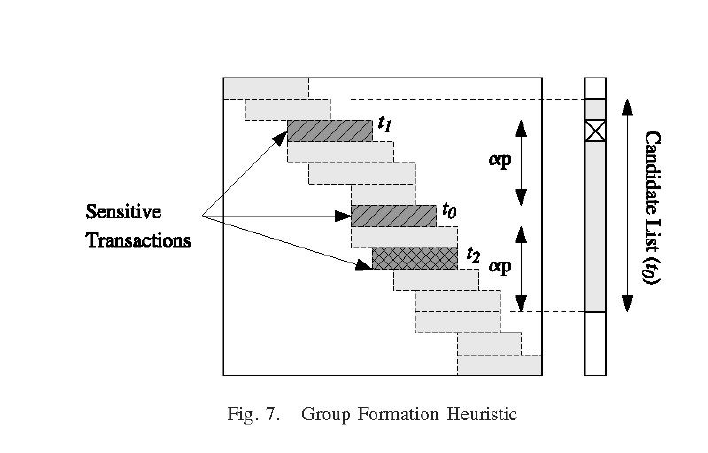
Consideriamo una matrice di 1000 x 1000 (1000 transazioni e 1000 items) con una media di 20 item per transazione, e considereremo tre gradi di correlazione 0.1, 0.5 e 0.9.

Per le matrici originali, noi possiamo osservare che variando il grado di correlazione: quando la correlazione è alta i dati tendono a clusterizzarsi in linee verticali, corrispondenti agli item che appaiono più frequentemente nelle transizioni. Nella parte di destra, noi possiamo osservaro come RCM cattura la correlazione dei dati, organizzando le transazioni con items simili più vicini. Più è alta la correlazione più è grande efficienza della riduzione della banda. Quindi, raggruppando insieme le transazioni vicine in gruppi anonimizzati preserva la correlazione tra gli items e quindi aumenta l’utilità dei dati.

Per il problema in studio, il numero delle transazioni n è di solito più grande nel numero di items d. In questo caso per ottenere una matrice n x n quadrata si aggiungono colonne con 0 (“fake” items). L’algoritmo non cambia e non aumenta la complessità.

# IV - Anonymized Group Formation

Dopo aver trasformato i dati in accordo con RCM, lo step successivo è quello di creare gruppi di transizioni. Per avere a pieno il requisito della privacy, ogni transizione sensibile ha bisogno di essere raggruppata con transazioni non sensibili, o con altri item sensibili. Noi proponiamo *CAHD (Correlation aware Anonymization of High-dimensional Data),* è una euristica greedy che capitalizza (si concentra) sulla correlazione dei dati, e raggruppa insieme transizioni che sono vicine nella rappresentazione band matrix.

Consideriamo l’esempio nella Fig 7., le transazioni non sensibili sono in grigietto, e le sensibili sono scure.

CAHD scansiona l’insieme delle transizioni T nell’ordine delle righe, trova la prima transizione sensibile nella sequenza e cerca di formare un gruppo anonimizzato per essa.

Definiamo due transizioni in conflitto se hanno almeno un dato sensibile in comune.

Nel nostro esempio e sono in conflitto (vedi che hanno gli stessi segni nell’immagine). non è in conflitto con nessuna delle due precedenti. Assumiamo di voler anonimizzare con un grado di privacy p. In questo caso, abbiamo bisogno di raggrupparla con almeno p -1 differenti transizioni.

Scegliamo di adottare l’euristica “one-occurence-per-group” che permette una solo occorrenza di ogni items sensibile in ogni gruppo.

Mostreremo brevemente, che se esiste una soluzione per anonimizzare con grado di privacy p, allora l’euristica troverà sempre la soluzione.

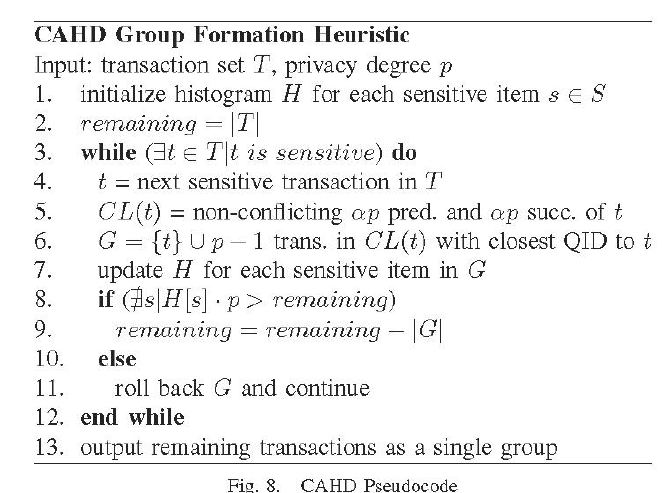
CAHD funziona come segue:

Dati una transizione sensibile una lista di candidate viene formata (CL), con transizioni che la precedono o rispettivamente seguono e non son in conflitto con o con ogni altra (quelle che sono in conflitto sono skippate durante la costruzione di CL). , è un parametro di sistema il quale restringe il range di ricerca. Intuitivamente più è grande, migliori sono le chance di includere in CL transizioni che items simili, ma incrementa il tempo di esecuzione. D’altronde anche con valori piccoli si possono ottenere buoni risultati grazie all’organizzazione della band matrix.

Notiamo che è escluso da CL() e il suo predecessore (che è non sensibile, quindi non in conflitto con viene incluso.

Quindi di fronte alle transizioni in CL(), le p-1 che hanno il più grande numero di items QID in comune con sono scelti per formare il gruppo anonimizzato. Intuizione è che più le transizioni hanno in comune gli stessi QID più è piccolo l’errore di ricostruzione.

Tutte le transizioni selezionare sono rimosse da T e il processo continua con le successive transizione sensibile.



Dato che usiamo una euristica greedy, noi dobbiamo assicurarci che il nostro metodo trovi una soluzione, cioè, ogni volta che si forma un gruppo non deve rimanere un insieme di transizioni che non possono essere anonimizzate (per esempio, se tutte le transizioni rimanenti condividono lo stesso item sensibile). Per questa ragione, manteniamo un istogramma con il numero delle restanti occorrenze per ogni dato sensibile. L’istogramma viene init quando i dati sono letti (line 1), e viene aggiornato ogni volta che un nuovo gruppo viene formato (line 7). Per convalidare un gruppo, controlliamo (line 8) se gli insiemi che rimangono delle transizioni soddisfano il requisito della privacy. Se così non fosse il gruppo non viene validato, e la formazione di un nuovo gruppo parte dalla transizione sensibile successiva.

L’algoritmo si ferma quando non ci sono più transizioni sensibili senza gruppo, o quando nessun nuovo gruppo può essere formato.

Se rimangono transizioni non raggruppate, vengono pubblicate come un gruppo singolo, ed è garantito, grazie al nostro controllo di convalida del gruppo, che questo gruppo soddisfa il grado di privacy p. Inoltre se tutte le transizioni che rimangono sono non sensibili, non c’è nessuna perdita di informazione se sono pubblicate come singolo gruppo, siccome i QID sono pubblicati direttamente.

Per implementare efficientemente la costruzione di CL ( cioè trovare transizioni non in conflitto), noi possiamo impiegare una linked-list data representation, dove ogni transizione (list entry) punta al suo predecessore e successore con un particolare sensitive item. Lo spazio richiesto è dove per transizione, e la complessità computazione dell’algoritmo è .