Εθνικο Μετσοβιο Πολυτεχνειο

Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων

Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων

Προστασία Της Ιδιωτικότητας Στην Δημοσίευση Μη Σχεσιακών Δεδομένων

Διδακτορική Διατριβή

της

Όλγας Γκουντούνα

Διπλωματούχου Ηλεκτρολόγου Μηχανικού & Μηχανικού Υπολογιστών Ε.Μ.Π. (2007)

Αθήνα, Οκτώβριος 2015

Η παρούσα έρευνα έχει συγχρηματοδοτηθεί από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο - ΕΚΤ) και από εθνικούς πόρους μέσω του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» του Εθνικού Στρατηγικού Πλαισίου Αναφοράς (ΕΣΠΑ) - Ερευνητικό Χρηματοδοτούμενο Έργο: Ηράκλειτος ΙΙ. Επένδυση στην κοινωνία της γνώσης μέσω του Ευρωπαϊκού Κοινωνικού Ταμείου.

Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων Και Μηχανικων Υπολογιστων
Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων

Προστασία Της Ιδιωτικότητας Στην Δημοσίευση Μη Σχεσιακών Δεδομένων

Διδακτορική Διατριβή

της

Όλγας Γκουντούνα

Διπλωματούχου Ηλεκτρολόγου Μηχανικού &
Μηχανικού Υπολογιστών Ε.Μ.Π. (2007)

Συμβουλευτική Επιτροπή: Ι. Βασιλείου
Τ. Σελλής
Θ. Δαλαμάγκας

Εγκρίθηκε από την επταμελή εξεταστική επιτροπή την 29η Οκτωβρίου 2015.

Ι. Βασιλείου	Τ. Σελλής	Θ. Δαλαμάγκας
Καθ. ΕΜΠ	Καθ. ΕΜΠ	Ερευνητής Β ΙΠΣΥ/ΕΚ Αθηνά
Κ. Κοντογιάννης	Γ. Στάμου	Α.-Γ. Σταφυλοπάτης
Αναπλ. Καθ. ΕΜΠ	Επικ. Καθ. ΕΜΠ	Καθ. ΕΜΠ

Π. Βασιλειάδης
Αναπλ. Καθ. Πανεπιστήμιο Ιωαννίνων

...

Όλγα Γκουντούνα
Διδάκτωρ Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Πρόλογος

Η παρούσα διατριβή εκπληρώνει τις απαιτήσεις για την απόκτηση διπλώματος του Διδάκτορα της Σχολής Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, του Εθνικού Μετσόβιου Πολυτεχνείου. Η παρούσα δουλειά περιγράφει διάφορες μεθόδους ανωνυμοποίησης για την διασφάλιση της Προστασίας Ιδιωτικότητας σε Μη-Σχεσιακά Δεδομένα και πραγματοποιήθηκε στο Εργαστήριο Συστημάτων Βάσεων Γνώσεων και Δεδομένων του ΕΜΠ.

Με το πέρας αυτής της δύσκολης και ενδιαφέρουσας διαδρομής νιώθω την ανάγκη να ευχαριστήσω όλους όσους με βοήθησαν.

Θα ήθελα να ευχαριστήσω τον Επιβλέποντα Καθηγητή μου, κ. Ιωάννη Βασιλείου για την ευκαιρία που μου έδωσε να ξεκινήσω αυτό το ταξίδι στην επιστημονική έρευνα, για την υποστήριξη και την καθοδήγησή του όλα αυτά τα χρόνια. Χάρη στην εμπιστοσύνη που μου έδειξε μπόρεσα να ασχοληθώ με ενδιαφέρουσες ερευνητικές περιοχές. Επιπλέον, θα ήθελα να ευχαριστήσω τον Καθηγητή κ. Τιμολέοντα Σελλή, για τις πολύτιμες συμβουλές του που με βοήθησαν να βελτιώσω σημαντικά την ποιότητα της έρευνάς μου.

Θέλω να εκφράσω τις θερμές μου ευχαριστίες προς τον Δρ. Μανώλη Τερροβίτη, Μεταδιδακτορικό Ερευνητή του ΙΠΣΥ/ΕΚ «Αθηνά», για τη συνεργασία μας, για την καθοδήγηση και την ανεκτίμητη βοήθειά του, καθώς και για την συνεισφορά του στις εργασίες που περιγράφονται στην παρούσα διδακτορική διατριβή. Ευχαριστώ επίσης τον Δρ. Θοδωρή Δαλαμάγκα, Ερευνητή Β΄ του ΙΠΣΥ/ΕΚ «Αθηνά» για την βοήθεια και πρόσφατη ερευνητική συνεργασία.

Επιπλέον, θα ήθελα να ευχαριστήσω τα Μέλη της Επταμελούς Εξεταστικής Επιτροπής Καθ. ΕΜΠ Ι. Βασιλείου, Καθ. ΕΜΠ Τ. Σελλή, Ερευνητής Β΄ ΙΠΣΥ/ΕΚ «Αθηνά» Θ. Δαλαμάγκα, Αναπλ. Καθ. ΕΜΠ Κ. Κοντογιάννη, Επικ. Καθ. ΕΜΠ Γ. Στάμου, Καθ. ΕΜΠ Α.-Γ. Σταφυλοπάτη και Αναπλ. Καθ. Πανεπιστημίου Ιωαννίνων Π. Βασιλειάδη για τον χρόνο που αφιέρωσαν να μελετήσουν και να αξιολογήσουν την διδακτορική μου διατριβή. Ευχαριστώ ιδιαίτερα για τα σχόλια, τις ερωτήσεις και τις προτάσεις τους για θέματα μελλοντικής έρευνας, τόσο στην ενδιάμεση κρίση όσο και στην τελική υποστήριξη.

Ακόμη, θα ήθελα να ευχαριστήσω τα μέλη του Εργαστηρίου Συστημάτων Βάσεων Γνώσεων και Δεδομένων ΕΜΠ και του Ινστιτούτου Πληροφοριακών Συστημάτων του ΕΚ Αθηνά, για τις δημιουργικές στιγμές που περάσαμε, για την φιλία και για την συνεργασία τους.

Επιπρόσθετα, θα ήθελα να αναγνωρίσω την συνεισφορά της Κατερίνας Λεπενιώτη και του Σωτήρη Αγγελή, που εργάστηκαν σε θέματα που σχετίζονται με την παρούσα διατριβή στα πλαίσια των διπλωματικών τους εργασιών στο Εργαστήριο Συστημάτων Βάσεων Γνώσεων και Δεδομένων του ΕΜΠ.

Αφιερώνω την παρούσα διδακτορική διατριβή στην γιαγιά μου Όλγα. Η αγάπη και η στήριξή της μου έδωσαν την δύναμη να προχωρήσω προς την ερευνητική πορεία που ήθελα να ακολουθήσω. Η άσβεστη φιλομάθεια της ήταν για μένα φωτεινό παράδειγμα για την διαρκή αναζήτηση της γνώσης μέσα από την επιστημονική μελέτη και έρευνα.

Όλγα Γκουντούνα
Αθήνα, Οκτώβριος 2015

Η παρούσα έρευνα έχει συγχρηματοδοτηθεί από την Ευρωπαϊκή Ένωση (Ευρωπαϊκό Κοινωνικό Ταμείο - ΕΚΤ) και από εθνικούς πόρους μέσω του Επιχειρησιακού Προγράμματος «Εκπαίδευση και Δια Βίου Μάθηση» του Εθνικού Στρατηγικού Πλαισίου Αναφοράς (ΕΣΠΑ) - Ερευνητικό Χρηματοδοτούμενο Έργο: Ηράκλειτος ΙΙ . Επένδυση στην κοινωνία της γνώσης μέσω του Ευρωπαϊκού Κοινωνικού Ταμείου.

Ευρωπαϊκή Ένωση
Ευρωπαϊκό Κοινωνικό Ταμείο

ΕΠΙΧΕΙΡΗΣΙΑΚΟ ΠΡΟΓΡΑΜΜΑ
ΕΚΠΑΙΔΕΥΣΗ ΚΑΙ ΔΙΑ ΒΙΟΥ ΜΑΘΗΣΗ
επένδυση στην κοινωνία της γνώσης
ΥΠΟΥΡΓΕΙΟ ΠΑΙΔΕΙΑΣ ΚΑΙ ΘΡΗΣΚΕΥΜΑΤΩΝ
ΕΙΔΙΚΗ ΥΠΗΡΕΣΙΑ ΔΙΑΧΕΙΡΙΣΗΣ

ΕΣΠΑ 2007-2013
πρόγραμμα για την ανάπτυξη
ΕΥΡΩΠΑΪΚΟ ΚΟΙΝΩΝΙΚΟ ΤΑΜΕΙΟ

Με τη συγχρηματοδότηση της Ελλάδας και της Ευρωπαϊκής Ένωσης

Περιεχόμενα

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Περίληψη

Η παρούσα διδακτορική διατριβή πραγματεύεται ζητήματα προστασίας της ιδιωτικότητας σε δημοσιεύσεις μη σχεσιακών δεδομένων. Η σχετική έρευνα αφορά σε ανάπτυξη και υλοποίηση αλγορίθμων οι οποίοι τροποποιούν τα προς δημοσίευση δεδομένα κατά τέτοιο τρόπο ώστε να μην αποκαλύπτεται η ταυτότητα των εγγραφών και η ευαίσθητη πληροφορία των ατόμων.

Μεγάλο εύρος καθημερινών ανθρώπινων δραστηριοτήτων όπως οι ιατρικές εξετάσεις, οι αγορές μέσω πιστωτικών καρτών, οι ιστοσελίδες κοινωνικής δικτύωσης, η χρήση μηχανών αναζήτησης στο διαδίκτυο, κτλ. προκαλούν την καταγραφή πληροφορίας. Η δημοσίευση ή διανομή αυτή των δεδομένων θέτει σε κίνδυνο την ιδιωτικότητα των χρηστών, καθώς ακόμα και μετά την απαλοιφή μόνο των μοναδικών αναγνωριστικών (ΑΦΜ, ονοματεπώνυμο), συνδυασμοί άλλων γνωρισμάτων (π.χ. φύλο, ηλικία, ΤΚ) μπορεί να είναι σπάνιοι και να λειτουργήσουν ως ψευδο-αναγνωριστικά, αποκαλύπτοντας την ταυτότητα των εγγραφών. Σκοπός των μεθόδων ανωνυμοποίησης είναι να αποφευχθεί η ταυτοποίηση εγγραφών από κακόβουλους επιτιθέμενους, ώστε να μην μπορεί να αποκαλυφθεί ευαίσθητη πληροφορία ατόμων. Αυτό επιτυγχάνεται με μετασχηματισμό των δεδομένων ώστε να μην είναι δυνατή η παραβίαση κάποιας εγγύησης ιδιωτικότητας. Η διαδικασία αυτή επιφέρει κάποια απώλεια πληροφορίας στα τελικά δεδομένα διότι αποκρύπτει ή αλλοιώνει μέρος της πληροφορίας των ψευδο-αναγνωριστικών. Οι αλγόριθμοι ανωνυμοποίησης έχουν ως στόχο την εύρεση της χρυσής τομής ανάμεσα στην προστασία της ιδιωτικότητας και στην χρησιμότητα των δεδομένων. Στη σχετική βιβλιογραφία έχουν γίνει πολλά βήματα στην προστασία της ιδιωτικότητας σχεσιακών δεδομένων. Στα πλαίσια της παρούσας διατριβής επικεντρωνόμαστε στην ανωνυμοποίηση (α) δεδομένων με δενδρική δομή, όπως είναι τα XML δεδομένα αλλά και οι σχεσιακές βάσεις με πολλούς πίνακες που συνδέονται μεταξύ τους με ξένα κλειδιά, (β) δεδομένων με δομή γράφου με έμφαση στα διασυνδεδεμένα δεδομένα και ειδικά τα δεδομένα RDF λόγω της ευρείας διάδοσής τους στον Παγκόσμιο Ιστό, και (γ) αραιών πολυδιάστατων δεδομένων χωρίς δομή που αποτελούνται από σύνολα συνεχών αριθμητικών τιμών, όπως π.χ. τα οικονομικά δεδομένα από σύνολα πληρωμών ή αγορών από πιστωτικές κάρτες. Επίσης, μελετήθηκαν διαφορετικά μοντέλα επίθεσης, όπως (i) το σενάριο όπου συναθροιστική γνώση των αριθμητικών τιμών των γνωρισμάτων κάποιας εγγραφής δεδομένων (π.χ. άθροισμα, μέσος όρος) μπορεί να χρησιμοποιηθεί για να γίνει ταυτοποίησή της, και (ii) πιο πολύπλοκα σενάρια επίθεσης που περιλαμβάνουν την γνώση πολλαπλών συναρτήσεων που καθεμία ορίζεται πάνω σε οποιοδήποτε υποσύνολο των πραγματικών τιμών των γνωρισμάτων μιας εγγραφής.

Abstract

This doctoral thesis addresses issues of privacy preservation on the publications of non-relational datasets. The research focuses on proposing and implementing anonymization algorithms which transform the datasets to be published, so that the identity of each record and the personal information of individuals are not revealed.

A broad spectrum of human activities, such as medical examinations, credit card purchases, internet searching, webpage browsing, etc. produce massive amounts of data. Their publication or sharing is posing a threat to the privacy of users; even after the elimination of direct identifiers (SSN, full name) the combinations of other attributes (such as gender, age, or zip code) may be rare enough to act as quasi-identifiers, thus revealing the identity of records. Anonymization methods aim at preventing the identification of records by malicious attackers, so that the sensitive information concerning individuals will not be revealed. This can be achieved by transforming the data in order to ensure that a given privacy guarantee is not violated. This process causes an information loss of the released data as it alters or conceals part of the quasi-identifier values. The goal of anonymization algorithms is to achieve the best balance between the privacy preservation of individuals and the utility of the released data. Several steps have been made in the related literature on the privacy of relational data. The main focus of this thesis is the anonymization of (a) tree-structured data, such as XML data, as well as joined records from a relational database which includes many tables linked together via foreign keys, (b) graph-structured data such as linked and RDF data which are popular on the Web, and (c) unstructured sparse multidimensional data where every record is a bag of continuous numerical values, such as financial payment data or credit card purchases. Furthermore, alternative attack models were studied, such as (i) the scenario where the aggregate knowledge of the numerical data values (e.g. sum, average) can lead to the identification of a personal record, and (ii) more complex attack scenarios which include the knowledge of multiple functions, each being defined on any subset of the numerical attributes of a record.

Κεφάλαιο 1

Εισαγωγή

Η έννοια της προστασίας ιδιωτικότητας είναι ευρεία και περιλαμβάνει την ασφάλεια (security), που αφορά σε δικαιώματα και περιορισμούς πρόσβασης σε δεδομένα, και την ανωνυμία (anonymity) η οποία αναφέρεται στην προστασία ευαίσθητων πληροφοριών σε δημοσιεύσεις δεδομένων. Η δεύτερη αποτελεί το βασικό αντικείμενο έρευνας της παρούσας διατριβής.

Μεγάλο εύρος καθημερινών ανθρώπινων δραστηριοτήτων προκαλούν την καταγραφή προσωπικής πληροφορίας. Πολλοί οργανισμοί συλλέγουν δεδομένα από χρήστες όπως ιατρικοί φάκελοι ασθενών, καλάθια αγορών πιστωτικών καρτών, προτιμήσεις διαδικτυακής αναζήτησης, κτλ. τα οποία θα μπορούσαν να δημοσιευθούν ή να διανεμηθούν για ερευνητικούς σκοπούς. Η διανομή αυτή των δεδομένων προκαλεί ανησυχία γύρω από τους ενδεχόμενους κινδύνους της ιδιωτικότητας των χρηστών. Σε πολλές περιπτώσεις τα δεδομένα που συλλέγονται περιλαμβάνουν ευαίσθητη πληροφορία που δεν πρέπει να αποκαλυφθεί σε τρίτους. Παραδείγματα τέτοιων συλλογών είναι η AOL η οποία συνέλλεξε και δημοσίευσε ερωτήματα αναζήτησης των χρηστών της [16], ο Netflix που συλλέγει τις προτιμήσεις των πελατών του [61] και ο οργανισμός ασφάλισης Group Insurance Commission (GIS) της Μασαχουσέτης που συλλέγει δεδομένα ασφάλισης των δημοσίων υπαλλήλων της πολιτείας [66]. Η ανησυχία για τις πιθανές απειλές ιδιωτικότητας είναι απολύτως ορθή καθώς ελλιπής ανωνυμοποίηση δεδομένων προκάλεσε την αποκάλυψη ευαίσθητων ιατρικών πληροφοριών του κυβερνήτη της Μασαχουσέτης το 2002 [66]. Επιπλέον, ελλιπής ανωνυμοποίηση οδήγησε στην αποκάλυψη της ταυτότητας ενός χρήστη της AOL από τους New York Times [16] και τέλος στην πιθανή ταυτοποίηση εγγεγραμμένων χρηστών του Netflix βασιζόμενη σε δημοσιεύσεις ιστολογίων [61].

Εντούτοις, η αξιοποίηση τέτοιον συλλογών δεδομένων μπορεί να είναι χρήσιμη για έρευνα, επιδημιολογικές μελέτες, στατιστική ανάλυση, κ.α. Η διανομή των δεδομένων μπορεί να γίνει είτε δημόσια λ.χ. με ανάρτησή τους σε μια ιστοσελίδα, είτε περιορισμένα σε κάποιο τρίτο οργανισμό ή ερευνητική ομάδα. Σε κάθε περίπτωση, ο κάτοχος των δεδομένων θα πρέπει να μπορεί να εγγυηθεί την προστασία της ιδιωτικότητας των ατόμων των οποίων η προσωπική πληροφορία περιέχονται στα προς δημοσίευση δεδομένα.

Η απλή απαλοιφή στοιχείων όπως το ονοματεπώνυμο, το ΑΦΜ, ή ο αριθμός ταυτότητας τα οποία προσδιορίζουν μοναδικά ένα άτομο, δεν επαρκεί για την διατήρηση της ανωνυμίας του σε ένα σύνολο εγγραφών. Άλλα στοιχεία όπως το φύλο, η ηλικία, το επάγγελμα κ.α. μπορούν να

συνδυαστούν για να προδώσουν την ταυτότητα της εγγραφής ενός ατόμου. Τα στοιχεία αυτά καλούνται *ψευδο-αναγνωριστικά* και φαινομενικά δεν μπορούν να προσδιορίσουν μοναδικά ένα άτομο. Συνδυασμοί τιμών τέτοιων γνωρισμάτων όμως μπορεί να είναι μοναδικοί μέσα σε ένα σύνολο εγγραφών, καθιστώντας δυνατή την ταύτιση ατόμων με συγκεκριμένες εγγραφές στα δημοσιευμένα δεδομένα. Οι αλγόριθμοι ανωνυμοποίησης τροποποιούν τα δεδομένα κατά τέτοιο τρόπο ώστε να μην είναι δυνατή η παραβίαση της ιδιωτικότητας ατόμων από αυτά. Η διαδικασία αυτή συχνά αποκαλείται *ανακωδικοποίηση* και επιφέρει κάποια απώλεια πληροφορίας στα τελικά δεδομένα διότι αποκρύπτει ή αλλοιώνει μέρος της πληροφορίας των ψευδο-αναγνωριστικών. Οι αλγόριθμοι ανωνυμοποίησης συχνά έχουν ως στόχο να βρουν την χρυσή τομή ανάμεσα στην προστασία και την χρησιμότητα των δεδομένων: την κατάλληλη ανακωδικοποίηση που εγγυάται την ανωνυμία των εγγραφών με την μικρότερη δυνατή απώλεια πληροφορίας των στοιχείων τους.

1.1 Προβλήματα και προκλήσεις

1.1.1 Δεδομένα με Δενδρική Δομή

Στον πυρήνα της διατριβής βρίσκονται τα δεδομένα με δενδρική δομή, όπως οι εγγραφές XML. Τέτοια δεδομένα υπάρχουν σε πληθώρα εφαρμογών καθώς είναι πιο εκφραστικά και ευέλικτα από τα παραδοσιακά σχεσιακά δεδομένα. Παρόλη τη διάδοση των ημιδομημένων δεδομένων σε πολλές σύγχρονες εφαρμογές δεν έχει γίνει έρευνα για την προστασία της ιδιωτικότητας σε αυτά. Οι μέθοδοι ανωνυμοποίησης που έχουν αναπτυχθεί για τα σχεσιακά δεδομένα δεν μπορούν να ανταποκριθούν στις πολλές διαστάσεις των ημιδομημένων, ενώ σε πολλές περιπτώσεις είναι σημασιολογικά αδύνατη η εφαρμογή τους. Η ιδιαιτερότητα είναι ότι τόσο οι τιμές των γνωρισμάτων όσο και η δομή κάθε εγγραφής μπορεί να χρησιμοποιηθεί για την αναγνώριση και ταυτοποίηση ατόμων. Στα πλαίσια της παρούσας έρευνας επιχειρείται να προταθεί ένα νέο πλαίσιο ανωνυμίας που θα επεκτείνει έννοιες οι οποίες προτάθηκαν για την ιδιωτικότητας σχεσιακών δεδομένων (όπως η k-ανωνυμία) σε ημιδομημένα δεδομένα.

Δομική πληροφορία μπορεί επίσης να προκύψει και από βάσεις δεδομένων με πολλούς πίνακες, οι οποίοι συνδέονται μεταξύ τους με εξαρτήσεις ξένων κλειδιών. Η πληροφορία κάθε ατόμου βρίσκεται αποθηκευμένη σε εγγραφές διαφόρων πινάκων της βάσης. Η ανωνυμοποίηση κάθε πίνακα ξεχωριστά με τις κλασικές μεθόδους για σχεσιακά δεδομένα δεν οδηγεί σε ασφαλή αποτελέσματα, γιατί δεν λαμβάνει υπόψη τις εξαρτήσεις των δεδομένων και την συνολική πληροφορία κάθε ατόμου. Η τελευταία μπορεί να υπολογιστεί από τις συνδέσεις των πινάκων πάνω στα ξένα κλειδιά, και το αποτέλεσμα μπορεί να αναπαρασταθεί ως μια «δενδρική» εγγραφή για κάθε άτομο της βάσης. Η μέθοδος που προτείνουμε μπορεί να προσαρμοστεί και σε αυτό το σενάριο για βάσεις δεδομένων, η διαδικασία της ανωνυμοποίησης μπορεί να γίνει με επεξεργασία των δενδρικών εγγραφών που προκύπτουν, ενώ τα τελικά ανώνυμα δεδομένα μπορούν να δημοσιευθούν σε ξεχωριστούς πίνακες, ακολουθώντας το αρχικό σχήμα της βάσης.

1.1.2 Δεδομένα με Δομή Γράφου

Στη συνέχεια εξετάζεται το πρόβλημα της προστασίας της ιδιωτικότητας σε δημοσιεύσεις δεδομένων με δομή γράφου. Στην υπάρχουσα βιβλιογραφία η έρευνα πάνω σε ανωνυμοποίηση γράφων εστιάζει σε κοινωνικά δίκτυα τα οποία μοντελοποιούνται ως ένα σύνολο όμοιων κόμβων και ακμών χωρίς ετικέτες. Αντίθετα, στην παρούσα διατριβή δόθηκε έμφαση στα διασυνδεδεμένα δεδομένα και ειδικά τα δεδομένα RDF λόγω της ευρείας διάδοσής τους στον Παγκόσμιο Ιστό. Η δομή των δεδομένων αυτών μπορεί επίσης να λειτουργήσει ως ψευδοαναγνωριστικό, αντίστοιχα με τα δενδρικά, αλλά ο βαθμός δυσκολίας αυξάνεται καθώς λαμβάνουμε υπόψη τις ετικέτες ακμών και κορυφών καθώς επίσης την κατεύθυνση των ακμών που δεν είναι απαραίτητα η ίδια κατά μήκος ενός μονοπατιού. Στα πλαίσια της διατριβής, μοντελοποιούνται τα σενάρια επίθεσης κατά της ιδιωτικότητας των RDF οντοτήτων και προτείνονται πράξεις ανακωδικοποίησης συμβατές με το μοντέλο των δεδομένων. Επίσης, επεκτείνεται η εγγύηση ιδιωτικότητας των δενδρικών δεδομένων για να καλύψει τις ιδιαιτερότητες της δομής των RDF γράφων και προτείνεται ένας νέος αλγόριθμος ανωνυμοποίησης που ικανοποιεί την εγγύηση περιορίζοντας την απώλεια πληροφορίας.

1.1.3 Αδόμητα Συνεχή Δεδομένα

Οι συλλογές εγγραφών όπου καθεμία είναι ένα αδόμητο σύνολο τιμών πρόκειται για ένα ακόμη ενδιαφέρον παράδειγμα δεδομένων που δεν υπακούν σε αυστηρό σχεσιακό σχήμα. Τέτοια δεδομένα παρουσιάζονται συχνά σε πολλές εφαρμογές: οι μετρήσεις αισθητήρων, οι καταγραφές ανθρώπινης παρατήρησης, οι δείκτες ιατρικών εξετάσεων όπως οι μετρήσεις σφυγμών και πίεσης του αίματος, τα οικονομικά δεδομένα όπως σύνολα πληρωμών ή αγορών από πιστωτικές κάρτες, είναι όλα παραδείγματα τέτοιων δεδομένων και η δημοσιοποίησή τους χωρίς την κατάλληλη επεξεργασία μπορεί να πλήξει την ιδιωτικότητα των χρηστών. Αντίθετα η εφαρμογή πολύ αυστηρών εγγυήσεων μπορεί να καταστήσει τα δεδομένα άχρηστα για ανάλυση και στατιστική μελέτη. Το πρόβλημα της ανωνυμοποίησης τέτοιων συλλογών έχει μελετηθεί στην βιβλιογραφία για τις περιπτώσεις κατηγορικών τιμών και οι προτεινόμενες λύσεις χρησιμοποιούν μια προκαθορισμένη ιεραρχία γενίκευσης τιμών. Στα πλαίσια της διατριβής, αναπτύχθηκε ένας νέος αλγόριθμος που χρησιμοποιεί δυναμικές ιεραρχίες για την ανωνυμοποίηση συλλογών δεδομένων όπου οι εγγραφές είναι σύνολα από συνεχείς τιμές.

1.1.4 Επιθέσεις Σύνθετης Γνώσης

Παράλληλα με την μελέτη για την προστασία διαφορετικών τύπων δεδομένων, μελετήθηκαν και διαφορετικά μοντέλα επίθεσης, όπως το σενάριο όπου συναθροιστική γνώση διαφόρων αριθμητικών τιμών των γνωρισμάτων κάποιας εγγραφής δεδομένων (λ.χ. το άθροισμα ή ο μέσος όρος) μπορεί να χρησιμοποιηθεί για να γίνει ταυτοποίησή της. Συχνά τα δεδομένα που δημοσιεύει μια εταιρία ή ένας οργανισμός περιέχουν τόσο λεπτομερείς τιμές ώστε να είναι απίθανο ένας κακόβουλος επιτιθέμενος να κατέχει ακριβή μερική γνώση μιας εγγραφής. Εντούτοις, θα μπορούσε να έχει πιο αφηρημένη ή συναθροιστική γνώση σχετικά με τα πεδία της εγγραφής. Τέτοια παραδείγματα προκύπτουν από πολλές εφαρμογές, ένα χαρακτηριστικό

παράδειγμα είναι τα φορολογικά δεδομένα. Κάθε εγγραφή περιέχει πολυάριθμα πεδία τα οποία καταγράφουν ένα μεγάλο εύρος οικονομικών δραστηριοτήτων των φορολογούμενων σε πολύ λεπτομερές επίπεδο. Μετά την δημοσίευση τέτοιων δεδομένων, αναμένουμε οι περισσότερες επιθέσεις να προέρχονται από επιτιθέμενους που αναγνωρίζουν τις εγγραφές βασιζόμενοι σε πιο γενική συναθροιστική γνώση, παραδείγματος χάριν το συνολικό εισόδημα, και όχι στις ακριβείς τιμές των επιμέρους πεδίων που είναι δυσκολότερο να τις γνωρίζει εκ των προτέρων, λ.χ. τα εισοδήματα από γεωργικές εργασίες. Η ανωνυμοποίηση τέτοιων δεδομένων χρησιμοποιώντας παραδοσιακές εγγυήσεις ιδιωτικότητας θα μπορούσε να εγγυηθεί την προστασία των εγγραφών, αλλά θα αλλοίωνε περισσότερο από ότι χρειάζεται τις τιμές τους. Στα πλαίσια της διδακτορικής διατριβής, επιχειρείται η μοντελοποίηση της συναθροιστικής γνώσης του επιτιθέμενου και τίθεται το πλαίσιο της ιδιωτικότητας για την προστασία από τέτοιες επιθέσεις.

Το παραπάνω πρόβλημα επεκτάθηκε σε πιο πολύπλοκα σενάρια επίθεσης τα οποία δεν περιλαμβάνουν την γνώση μόνο μίας συναθροιστικής συνάρτησης που ορίζεται πάνω στις τιμές όλων των γνωρισμάτων μιας εγγραφής, αλλά την γνώση πολλαπλών συναρτήσεων που καθεμία ορίζεται πάνω σε οποιοδήποτε υποσύνολο των πραγματικών τιμών των γνωρισμάτων μιας εγγραφής. Ένα τέτοιο παράδειγμα είναι όταν ο επιτιθέμενος γνωρίζει τα καθαρά κέρδη (έσοδα - απώλειες) από επενδύσεις κεφαλαίων, τον μέσο όρο πληρωμών σε πιστωτικές κάρτες ανα μήνα, και το συνολικό εισόδημα από τον μισθό και τα ενοίκια ενός φορολογούμενου. Η γνώση αυτή μοντελοποιείται ως ένα σύστημα εξισώσεων με μεταβλητές τα πεδία της εγγραφής. Προτείνονται διάφοροι εναλλακτικοί αλγόριθμοι οι οποίοι ανωνυμοποιούν τα δεδομένα απέναντι σε τέτοιες επιθέσεις και εισάγουν μικρότερη απώλεια πληροφορίας.

1.2 Συνεισφορά

Η συνεισφορά της διατριβής συνοψίζεται στα παρακάτω σημεία:

1. Μελέτη της προστασία της ιδιωτικότητας σε δημοσιεύσεις δεδομένων με δενδρική δομή και δεδομένων με δομή γράφου RDF.

2. Ορισμός των μοντέλων επίθεσης και ανίχνευση των απειλών κατά της ιδιωτικότητας για τα παραπάνω δεδομένα.

3. Ανάλυση του τρόπου με τον οποίο η δομή των εγγραφών μπορεί να λειτουργεί ως ψευδο-αναγνωριστική πληροφορία που μπορεί να οδηγήσει στην ταυτοποίησή τους.

4. Διατύπωση νέων εγγυήσεων ιδιωτικότητας, οι οποίες είναι προσαρμοσμένες στην ιδιαίτερη δομή των δεδομένων.

5. Μελέτη της ανωνυμοποίησης αδόμητων δεδομένων με συνεχή γνωρίσματα, χωρίς προκαθορισμένες ιεραρχίες γενίκευσης.

6. Μελέτη σύνθετων σεναρίων επίθεσης. Αντιμετώπιση επιτιθέμενων με σύνθετη συναθροιστική ή συναρτησιακή γνώση.

7. Ανάπτυξη και υλοποίηση αλγορίθμων ανωνυμοποίησης που επιλύουν τα παραπάνω προβλήματα, με μικρή απώλεια πληροφορίας.

8. Πειραματική αξιολόγηση των προτεινόμενων μεθόδων και σύγκριση των αποτελεσμάτων με κλασσικές εγγυήσεις ανωνυμίας, όπου είναι δυνατή η σύγκριση.

1.3 Δομή της διατριβής

Η υπόλοιπη έκθεση αναπτύσσεται ως εξής: στο Κεφάλαιο 2 δίνεται συνοπτικά το υπόβαθρο στο οποίο στηρίζεται η παρούσα δουλειά, καθώς και μία επισκόπηση των σχετικών εργασιών. Στο Κεφάλαιο 3 παρουσιάζεται η νέα μέθοδος που προτείνουμε για την ανωνυμοποίηση δεδομένων με δενδρική δομή. Το Κεφάλαιο 4 πραγματεύεται τις προκλήσεις γύρω από την ανωνυμοποίηση δεδομένων RDF, τα οποία έχουν την δομή γράφου. Στο Κεφάλαιο 5 παρουσιάζει μια μέθοδο ανωνυμοποίησης αδόμητων δεδομένων με συνεχή γνωρίσματα, χωρίς την χρήση προκαθορισμένων ιεραρχιών γενίκευσης. Στο Κεφάλαιο 6 παρουσιάζεται η μέθοδος που προτείνουμε για την αντιμετώπιση επιτιθέμενων με συναθροιστική γνώση. Η μέθοδος αυτή επεκτείνεται στο Κεφάλαιο 7 όπου μελετάμε σενάρια επιθέσεων με γνώση πολλαπλών συνεχών συναρτήσεων που καθεμία ορίζεται πάνω σε οποιοδήποτε υποσύνολο των πραγματικών τιμών των γνωρισμάτων μιας εγγραφής. Τέλος, στο Κεφάλαιο 8 συνοψίζουμε την παρουσίαση της εργασίας και καταγράφουμε τα επόμενα βήματα της διατριβής.

Κεφάλαιο 2

Σχετικές Εργασίες

2.1 Υπόβαθρο

Η προστασία της ιδιωτικότητας των σχεσιακών δεδομένων έχει βρεθεί στο επίκεντρο της έρευνας τα τελευταία χρόνια λόγω της ανάγκης πολλών οργανισμών να δημοσιεύουν και να μοιράζονται δεδομένα χωρίς να αποκαλύπτονται οι ταυτότητες ατομικών εγγραφών. Παρά την απαλοιφή των αναγνωριστικών (π.χ. ονοματεπώνυμα, ΑΦΜ) ένας κακόβουλος επιτιθέμενος μπορεί να καταφέρει να αντιστοιχίσει μια εγγραφή με ένα συγκεκριμένο άτομο χρησιμοποιώντας ένα συνδυασμό άλλων γνωρισμάτων όπως φύλο, ηλικία και ταχυδρομικός κώδικας. Για να αποφευχθεί ένα τέτοιο σενάριο τα προς δημοσίευση δεδομένα μετασχηματίζονται κατάλληλα ώστε να δημοσιευθούν ανώνυμα. Η διαδικασία αυτή καλείται *ανακωδικοποίηση* (recoding).

Ας υποθέσουμε ότι οι ιατρικοί φάκελοι ενός νοσοκομείου θα πρέπει να διανεμηθούν για τη διενέργεια επιδημιολογικών μελετών. Το σύνολο αυτών των δεδομένων περιέχει εγγραφές ασθενών με πεδία όπως <ΑΔΤ, Όνομα, Ηλικία, Φύλο, Ταχυδρομικός Κώδικας, Διεύθυνση Ηλεκτρονικού Ταχυδρομείου, Ασθένεια>. Κάθε εγγραφή ανήκει σε ένα άτομο και περιέχει ευαίσθητες πληροφορίες του. Τα δεδομένα θα πρέπει να διανεμηθούν για ερευνητικό σκοπό χωρίς να τίθεται σε κίνδυνο η ιδιωτικότητα των ασθενών. Μια πρώτη σκέψη είναι να αφαιρεθούν από τα δεδομένα τα στοιχεία που μπορούν να συνδέσουν άμεσα ένα άτομο με την εγγραφή του. Γνωρίσματα όπως ονοματεπώνυμο, ΑΔΤ, διεύθυνση ηλεκτρονικού ταχυδρομείου και άλλα καλούνται *μοναδικά αναγνωριστικά* (unique identifiers) και απαλείφονται από τα προς δημοσίευση δεδομένα. Εντούτοις, η προστασία της ιδιωτικότητας των ασθενών δεν διασφαλίζεται καθώς οι συνδυασμοί τιμών των υπολοίπων γνωρισμάτων μπορεί να συνδυαστούν με εξωτερική γνώση από άλλες πηγές όπως εκλογικοί ή τηλεφωνικοί κατάλογοι και να οδηγήσει στην ταυτοποίηση ατόμων. Γνωρίσματα όπως η ηλικία και το φύλο δεν προσδιορίζουν μοναδικά έναν άνθρωπο. Συνεπώς διατηρούνται στα δημοσιευμένα δεδομένα καθώς η γνώση τους είναι σημαντική για τις μελέτες των ερευνητών. Ορισμένοι συνδυασμοί των τιμών τους όμως μπορεί να εμφανίζονται εξαιρετικά σπάνια μέσα στο σύνολο των εγγραφών, ή ακόμα και να είναι μοναδικοί. Σε αυτή την περίπτωση υπάρχει πιθανότητα αναγνώρισης ατόμων. Τα γνωρίσματα αυτά καλούνται *Ψευδο-αναγνωριστικά* (quasi-identifiers QI) και χρησιμοποιούνται για την αποκάλυψη της ταυτότητας ατόμων όταν συνδυαστούν με προϋπάρχουσα γνώση

(background knowledge) του επιτιθέμενου από εξωτερικές πηγές.

2.2 Μοντέλα Επιθέσεων κατά της Ιδιωτικότητας

Η προϋπάρχουσα γνώση του επιτιθέμενου στη συνηθέστερη περίπτωση μπορεί να μοντελοποιηθεί ως ένα σύνολο *τιμών* των ψευδο-αναγνωριστικών γνωρισμάτων μιας εγγραφής στόχου (target). Η μοντελοποίηση αυτή συναντάται στις περισσότερες εργασίες της υπάρχουσας βιβλιογραφίας.

Ο επιτιθέμενος θα μπορούσε επιπλέον να γνωρίζει πληροφορίες για την *δομή* των δεδομένων, δηλαδή συσχετίσεις μεταξύ τιμών μιας εγγραφής όπως είναι οι δομικές σχέσεις μεταξύ κόμβων μιας εγγραφής XML ή οι λειτουργικές εξαρτήσεις μεταξύ εγγραφών από διαφορετικούς πίνακες που συνδέονται μέσω ξένων κλειδιών. Η μοντελοποίηση και αντιμετώπιση αυτού του τύπου των επιθέσεων είναι το βασικό αντικείμενο μελέτης στο Κεφάλαιο 3 της παρούσας διατριβής.

Ένα ενδιαφέρον σενάριο είναι όταν ο επιτιθέμενος έχει κάποια *συναθροιστική* γνώση πάνω στις τιμές των ψευδο-αναγνωριστικών, όπως το άθροισμα, τον μέσο όρο των τιμών τους κτλ. Το πρόβλημα αυτό δεν έχει μελετηθεί μέχρι στιγμής και μια πρώτη μέθοδος για την προστασία από τέτοιες επιθέσεις προτείνουμε στο Κεφάλαιο 6.

Τέλος, ο επιτιθέμενος θα μπορούσε να γνωρίζει ακόμα και τον *αλγόριθμο* με τον οποίο έχουν μετασχηματιστεί τα δεδομένα και μπορεί να χρησιμοποιήσει αυτή τη γνώση για να αποκλείσει ορισμένες περιπτώσεις και να αυξήσει την πιθανότητα ταυτοποίησης μιας εγγραφής. Η επίθεση αυτή βασίζεται στην αρχή της ελάχιστης απώλειας πληροφορίας στην οποία στοχεύουν οι περισσότεροι αλγόριθμοι ανωνυμοποίησης και αντιμετωπίζεται επιτρέποντας να υπάρχει ένας βαθμός τυχαιότητας στις επιλογές των κατάλληλων μετασχηματισμών των δεδομένων.

Βασιζόμενος στην προϋπάρχουσα γνώση ο επιτιθέμενος προσπαθεί να ανακαλύψεις επιπλέον πληροφορίες για την εγγραφή ενός ατόμου-στόχου. Ανάλογα με τον τύπο της πληροφορίας που προσπαθεί να ανακαλύψει, οι απειλές κατά της ιδιωτικότητας κατηγοριοποιούνται σε αποκάλυψη ταυτότητας, γνωρίσματος και παρουσίας στα δεδομένα.

2.2.1 Αποκάλυψη Ταυτότητας

Θεωρώντας δεδομένη την παρουσία της εγγραφής ενός ατόμου στα δημοσιευμένα δεδομένα, ο επιτιθέμενος επιχειρεί να την αναγνωρίσει χρησιμοποιώντας την γνώση που διαθέτει από εξωτερικές πηγές. Η πιο χαρακτηριστική περίπτωση είναι η ταυτοποίηση της ιατρικής εγγραφής του κυβερνήτη της Μασαχουσέτης από την Sweeney το 2002 [66]. Οι τιμές των ψευδο-αναγνωριστικών του κυβερνήτη ταυτοποιήθηκαν με τα στοιχεία του από δημόσιες πηγές, όπως οι εκλογικοί κατάλογοι. Αυτή η τακτική καλείται και επίθεση σύνδεσης (linking attack). Για την αντιμετώπιση τέτοιων επιθέσεων έχει προταθεί η *k-ανωνυμία* [63, 64], μια εγγύηση ιδιωτικότητας για την προστασία από τέτοιες επιθέσεις. Με βάση την k-ανωνυμία κάθε εγγραφή των δημοσιευμένων δεδομένων θα πρέπει να μην μπορεί να διακριθεί ανάμεσα σε τουλάχιστον k-1 άλλες, ως προς τις τιμές των ψευδο-αναγνωριστικών της. Οι ομάδες εγγραφών που έχουν πανομοιότυπες τιμές στο σύνολο των ψευδο-αναγνωριστικών γνωρισμάτων

τους ανήκουν στην ίδια *κλάση ισοδυναμίας* (equivalence class). Το πρόβλημα επίλυσης της βέλτιστης k-ανωνυμίας έχει αποδειχθεί ότι είναι NP-hard στο [53], όπου δόθηκε προσεγγιστικός αλγόριθμος με όριο $O(klogk)$. Οι [14] μείωσαν το όριο αυτό σε $O(k)$, ενώ οι [62] το περιόρισαν περαιτέρω σε $O(logk)$.

Σχήμα 2.1: Παράδειγμα επίθεσης σύνδεσης με σκοπό την αποκάλυψη της ταυτότητας ιατρικής εγγραφής.

2.2.2 Αποκάλυψη Γνωρίσματος

Η προστασία από την αποκάλυψη της ταυτότητας δεν εγγυάται και την μη αποκάλυψη κάποιας ευαίσθητης πληροφορίας για ένα άτομο. Συγκεκριμένα, αν k ή περισσότερες εγγραφές είναι πανομοιότυπες τόσο ως προς τα ψευδο-αναγνωριστικά όσο και ως προς τα ευαίσθητα γνωρίσματα, τότε η εγγύηση της k-ανωνυμίας ικανοποιείται. Ο επιτιθέμενος όμως δεν χρειάζεται πλέον να εντοπίσει ποιά ακριβώς είναι η εγγραφή του στόχου εφόσον η ευαίσθητη πληροφορία είναι η ίδια και έχει αποκαλυφθεί. Αυτός ο τύπος επιθέσεων καλείται *επίθεση ομοιογένειας* (homogeneity attack). Παρόμοια είναι και η *επίθεση ομοιότητας* (similarity attack) όπου όλες οι τιμές σε μια κλάση ισοδυναμίας είναι παρεμφερής ή έχουν κοινά χαρακτηριστικά. Αν π.χ. οι ασθενής που ανήκουν στην ίδια κλάση ισοδυναμίας σε μια ιατρική βάση δεδομένων έχουν τις ασθένειες 'γρίπη', 'βρογχίτιδα' και 'πνευμονία' τότε ο επιτιθέμενος δεν μπορεί να μάθει με ακρίβεια την ευαίσθητη πληροφορία του στόχου, ανακαλύπει όμως ότι έχει κάποια πάθηση του αναπνευστικού συστήματος.

Αντίστοιχα, αν ο επιτιθέμενος μπορεί με την γνώση του να αποκλείσει κάποιες από τις ευαίσθητες τιμές που εμφανίζονται στην ιδια κλάση ισοδυναμίας τότε αυξάνει την πιθανότητα να βρει την ευαίσθητη πληροφορία του στόχου και έχουμε επίθεση προϋπάρχουσας γνώσης (background-knowledge attack).

Η βασικότερη εγγύηση που επεκτείνει την k-ανωνυμία και αντιμετωπίζει την απειλή αποκάλυψης γνωρίσματος είναι η ℓ-διαφορετικότητα [52, 78, 31]. Για να ικανοποιείται η εγγύηση αυτά θα πρέπει να ισχύει ως επιπλέον περιορισμός ότι μέσα σε κάθε κλάση ισοδυναμίας θα πρέπει να εμφανίζονται τουλάχιστον ℓ καλώς αντιπροσωπευόμενες τιμές του ευαίσθητου γνωρίσματος. Αντίστοιχες εγγυήσεις παρέχουν και οι [51, 73, 74, 83]. Στο [75] αποδείχθηκε ότι η βέλτιστη ℓ-διαφορετικότητα είναι NP-Hard, ενώ παρέχεται ένα όριο προσέγγισης $O(\ell \times d)$, όπου d είναι το πλήθος των ψευδο-αναγνωριστικών, για την περίπτωση που χρησιμοποιείται

μόνον η τεχνική της απαλοιφής τιμών (βλ. παρ. 2.3.2). Στο [59], οι συγγραφείς προτείνουν αλγόριθμους k-ανωνυμίας και ℓ-διαφορετικότητας οι οποίοι ελαχιστοποιούν το πλήθος των προσβάσεων στα δεδομένα χρησιμοποιώντας την δομή περίληψης που κρατάει το σύστημα διαχείρισης της βάσης δεδομένων για την επιλεξιμότητα των ερωτημάτων. Ενώ στο [32] επεκτείνεται το [78] για αραιά πολυδιάστατα δεδομένα.

Τέλος, ένα άλλο είδος απειλής μπορεί να συμβεί όταν η κατανομή των τιμών του ευαίσθητου γνωρίσματος στο σύνολο των δεδομένων παρουσιάζει μεγάλη διαφορά από την κατανομή του μέσα σε κάποια κλάση ισοδυναμίας. Ας θεωρήσουμε ότι δημοσιεύονται ανώνυμα τα αποτελέσματα εξετάσεων για τον ιό HIV και στο σύνολο των ασθενών τα ποσοστά είναι 90% αρνητικό και 10% θετικό. Αν σε μια κλάση ισοδυναμίας τα αντίστοιχα ποσοστά είναι 50% και 50% αντιστοίχως, τότε τα άτομα των οποίων οι εγγραφές βρίσκονται σε αυτή την κλάση κινδυνεύουν να χαρακτηριστούν με αυξημένη πιθανότητα θετικά στον ιό. Παρόλο που ικανοποιούνται οι εγγυήσεις της k-ανωνυμία και της ℓ-διαφορετικότητα, η ιδιωτικότητα των ασθενών μιας κλάσης απειλείται, είτε όντως πάσχουν είτε όχι. Αυτός ο τύπος επίθεσης καλείται *επίθεση διαφοράς κατανομών* (skewness attack) και για να αντιμετωπιστεί προτάθηκε η εγγύση της t-εγγύτητας [49, 50] όπου απαιτείται η κατανομή κάθε κλάσης ισοδυναμίας να μην διαφέρει από την κατανομή του συνόλου των δεδομένων περισσότερο από μία παράμετρο t. Η εγγύηση της *β-ομοιότητας* [18] επεκτείνει την t-εγγύτητα και περιορίζει το κέρδος πληροφορίας μιας ευαίσθητης τιμής, που ορίζεται ως η διαφορά ανάμεσα στην αρχική της υποστήριξη και την αντίστοιχη υποστήριξη της τιμής στα τελικά δεδομένα.

2.2.3 Αποκάλυψη Παρουσίας

Μια άλλη μορφή επίθεσης μπορεί να συμβεί όταν ο επιτιθέμενος δεν γνωρίζει αν η εγγραφή ενός στόχου συμπεριλαμβάνεται στα δημοσιευμένα δεδομένα και προσπαθεί να το ανακαλύψει. Η εγγύηση χαρακτιριστικότερη που παρέχει προστασία από αυτές τις επιθέσεις είναι η δ-παρουσία [57]. Εγγυήσεις απέναντι σε τέτοιες επιθέσεις προστατεύουν και από την αποκάλυψη της ταυτότητας, οι περισσότερες εργασίες της βιβλιογραφίας όμως θεωρούν ότι ο επιτιθέμενος γνωρίζει την παρουσία της εγγραφής ενός στόχου στα ανωνυοποιημένα δεδομένα, και ότι προσπαθεί να την ανακαλύψει.

2.3 Τεχνικές Ανωνυμοποίησης

Οι μετασχηματισμοί που εφαρμόζονται για την ανακωδικοποίηση των δεδομένων προς δημοσίευση με σκοπό την ικανοποίηση μιας εγγύησης ιδιωτικότητας και έχουν προταθεί έως τώρα στη βιβλιογραφία είναι οι παρακάτω.

2.3.1 Γενίκευση.

Η αντικατάσταση μιας τιμής ενός γνωρίσματος από μια άλλη πιο γενική τιμή που την περιέχει καλείται *γενίκευση* (generalization) [64, 65]. Στην περίπτωση των αριθμητικών τιμών μπορεί να γίνει αντικατάσταση από ένα εύρος τιμών που περιέχει την αρχική, π.χ. η τιμή 24 να

αντικατασταθεί από την [20-30]. Στην περίπτωση κατηγορικών γνωρισμάτων, υποθέτουμε την ύπαρξη μιας *ιεραχίας γενίκευσης* που έχει δενδρική δομή. Για παράδειγμα ο τόπος διαμονής από 'Αθήνα' και 'Πειραιά' γενικεύεται σε 'Αττική', ενώ στη συνέχεια η 'Αττική' μπορεί να γενικευθεί περαιτέρω σε 'Στερεά Ελλάδα' κ.ο.κ. Η διαδικασία ανακωδικοποίησης μπορεί να κατηγοριοποιηθεί σε μονοδιάστατη όπου η αντικατάσταση της γενίκευσης εφαρμόζεται σε κάθε ένα γνώρισμα χωριστά [45] και σε πολυδιάστατη όπου γίνεται απεικόνιση του καρτεσιανού γινομένου των πολλαπλών γνωρισμάτων [46].

Η γενίκευση μιας τιμής μπορεί να εφαρμόζεται συνολικά σε όλες τις εμφανίσεις στα δεδομένα και καλείται *ολική ανακωδικοποίηση* (global recoding) [17, 45]. Εναλλακτικά, ορισμένες δουλειές τροποποιούν μόνο συγκεκριμένες εμφανίσεις της τιμής εφαρμόζοντας *τοπική ανακωδικοποίηση* (local recoding) [46, 48, 15, 80]. Τέλος, έχει μελετηθεί η περίπτωση όπου οι γενικεύσεις τιμών δεν βασίζονται σε μια προϋπάρχουσα ιεραρχία, αντίθετα αποφασίζονται σε πραγματικό χρόνο ανάλογα με τα δεδομένα και με την επιλογή γενίκευσης που ελαχιστοποιεί τις απώλειες πληροφορίας κάθε φορά [60].

2.3.2 Απαλοιφή.

Η αφαίρεση κάποιας πληροφορίας από τα δεδομένα καλείται *απαλοιφή* (suppression). Είναι δυνατόν να απαλειφθούν τιμές γνωρισμάτων ή και ολόκληρες εγγραφές αν είναι τόσο σπάνιες που προδίδουν την ταυτότητα ατόμων. Η απαλοιφή μπορεί να θεωρηθεί ως η γενίκευση μιας τιμής στην ρίζα της ιεραρχίας της, όπου σημαίνει «οποιαδήποτε τιμή». Για το λόγο αυτό συχνά στη βιβλιογραφία η απαλοιφή μιας τιμής συμβολίζεται με τον ειδικό χαρακτηρα «*» τον οποίο συναντάμε και ως ρίζα στις ιεραρχίες γενίκευσης.

2.3.3 Εισαγωγή Θορύβου.

Η εισαγωγή θορύβου στις τιμές των δεδομένων ή/και η προσθήκη συνθετικών εγγραφών έχουν χρησιμοποιηθεί στην βιβλιογραφία ώστε να παραπλανηθούν οι επιθέσεις σύνδεσης [67, 19]. Η εισαγωγή στατιστικού θορύβου, συνήθως Λαπλασιανού ή Γκαουσιανού, στα δεδομένα υιοθετήθηκε από την διαφορική ιδιωτικότητα (differential privacy) [27, 28, 76, 29, 77, 47] ώστε να διασφαλίσει την ιδιωτικότητα ακόμα και για την χειρότερη περίπτωση ισχυρής γνώσης του επιτιθέμενου, ενώ παράλληλα να διατηρούνται οι στατιστικές ιδιότητες των αρχικών δεδομένων. Πρόκειται για μια ιδιαιτέρως αυστηρή εγγύηση ιδιωτικότητας, η οποία πρακτικά αλλοιώνει σημαντικά τα αρχικά δεδομένα. Ο στόχος της είναι να περιορίσει την διαφορά της γνώσης που μπορεί να μάθει ο επιτιθέμενος από την παρουσία ή απουσία κάθε μίας εγγραφής, για κάθε πληροφορία που μπορεί να εξαχθεί από τα δεδομένα. Με άλλα λόγια, αν δύο σύνολα δεδομένων διαφέρουν κατά μόνο μια εγγραφή, τότε τα ερωτήματα για τον υπολογισμό μιας συνάρτησης f θα επιστρέφουν παρόμοια αποτελέσματα για τις δύο βάσεις. Δηλαδή περιορίζεται η συνεισφορά κάθε ξεχωριστής εγγραφής στα συνολικά δεδομένα. Η διαφορική ιδιωτικότητα δεν είναι μια εγγύηση που λύνει απόλυτα το πρόβλημα όπως έδειξαν διάφορες σχετικές μελέτες. Συγκεκριμένα, στο [44] δείχθηκε ότι η διαφορική ιδιωτικότητα δεν περιορίζει ικανοποιητικά την εξαγωγή συμπερασμάτων σχετικά με την συμμετοχή ενός ατόμου στα δεδομένα. Επίσης,

στο [24] δείχθηκε ότι, παρόλο που κάθε επιμέρους άτομο καλύπτεται από τον θόρυβο, ο ίδιος ο θόρυβος καλύπτεται από το σήμα πληροφορίας που προκύπτει από τον συνολικό πληθυσμό των εγγραφών. Συνεπώς, μπορεί να εκπαιδευτεί ένας Naïve Bayes classifier που να προβλέπει την ευαίσθητη πληροφορία των ατόμων με ικανοποιητική ακρίβεια. Στη συνέχεια, στο [25] συγκρίθηκαν πειραματικά διάφορα μοντέλα ιδιωτικότητας ως προς την προστασία ιδιωτικότητας καθώς και ως προς την χρησιμότητα των τελικών δεδομένων. Οι μετρήσεις αφορούν την εκ των υστέρων πεποίθηση του επιτιθέμενου, δηλαδή την πιθανότητα να αντιστοιχίσει μια τιμή ενός γνωρίσματος με ένα άτομο αφού πρώτα δει τα τελικά δεδομένα, και την ικανότητά του να εξάγει συμπεράσματα σχετικά με τις ευαίσθητες τιμές των δεδομένων. Δείχνουν ότι η διαφορά μεταξύ της διαφορικής ιδιωτικότητας και των διάφορων συντακτικών μεθόδων (k-ανωνυμία, ℓ-διαφορετικότητα, κτλ.) είναι λιγότερο δραματική σε σχέση με ότι πίστευαν. Ειδικά όταν η ορθότητα θεωρείται σημαντική, οι συντακτικές μέθοδοι θεωρούνται προτιμότερες, καθώς κάνουν καλύτερο συμβιβασμό μεταξύ της ιδιωτικότητας και της χρησιμότητας των δεδομένων. Όμοια οι [23] συγκρίνουν και αναλύουν και τις δύο προσεγγίσεις και καταλήγουν στο ότι η διαφορική ιδιωτικότητα είναι προτιμότερη για προστασία της ιδιωτικότητας σε εφαρμογές εξόρυξης γνώσης από δεδομένα, ενώ οι συντακτικές μέθοδοι είναι καταλληλότερες για την προστασία της ιδιωτικότητας σε δημοσιεύσεις δεδομένων.

2.3.4 Αποσυσχέτιση.

Η τεχνική αυτή αντίθετα με τη γενίκευση και την απαλοιφή, διατηρεί αναλλοίωτη την *ακρίβεια* των τιμών των δεδομένων. Αντίθετα διασπά τους σπάνιους συνδυασμούς τιμών έτσι ώστε να μην είναι σαφές αν ανήκουν στην ίδια εγγραφή. Η πιο αντιπροσωπευτική εργασία που έχει γίνει σε αυτή την κατεύθυνση είναι αυτή της *ανατομίας* [78]. Η ανατομία παρέχει εγγυήσεις ιδιωτικότητας ανάλογες με αυτές της k-ανωνυμίας και της ℓ-διαφορετικότητας, αλλά χωρίς γενικεύσεις ή απαλοιφές τιμών. Ομαδοποιεί τις εγγραφές σε κλάσεις ισοδυναμίας ως προς τα ψευδο-αναγνωριστικά, χωρίς να τα γενικεύει και να κάνει τις τιμές τους να φαίνονται πανομοιότυπες. Αντίθετα, αποσυνδέει τα ευαίσθητα γνωρίσματα από τα ψευδο-αναγνωριστικά, και τα δημοσιεύει σε χωριστούς πίνακες οργανωμένα αντίστοιχα ανά κλάσεις ισοδυναμίας. Στο [43] δείχθηκε ότι η ανατομία [78] είναι ευπαθής σε επιθέσεις deFinetti, που στοχεύουν στην μάθηση της συσχέτισης μεταξύ ευαίσθητων γνωρισμάτων και ψευδο-αναγνωριστικών χρησιμοποιώντας ένα Bayesian network. Εντούτοις, στο [24] δείχθηκε ότι οι επιθέσεις deFinetti είναι αποτελεσματικές μόνο για μικρές τιμές της παραμέτρου ℓ. Στο [70] γίνεται εφαρμογή της μεθόδου της αποσυσχέτισης σε εγγραφές που αποτελούν σύνολα τιμών, όπου κάθε εγγραφή μπορεί να σπάσει σε δύο ή περισσότερα τμήματα.

2.4 Σχετικές εργασίες

Το πρόβλημα της ανωνυμοποίησης ημιδομημένων δεδομένων το οποίο αποτελεί τον βασικό πυρήνα της διατριβής δεν έχει επιλυθεί από τους αλγόριθμους της υπάρχουσας βιβλιογραφίας. Εντούτοις, υπάρχουν κάποιες εργασίες οι οποίες έχουν κάποια πιο άμεση σχέση με το αντικείμενο αυτής της μελέτης, γι΄ αυτό και παρουσιάζονται στις επόμενες παραγράφους.

Έχουν γίνει αρκετές εργασίες για την προστασία της ιδιωτικότητας σε δεδομένα πολλών δαστάσεων. Έχουν γίνει αξιόλογες εργασίες σε βάσεις δεδομένων γράφων (graph databases) [20, 84, 79], αλλά σε αυτές τις περιπτώσεις εστιάζουν στην προστασία της ταυτότητας ενός κόμβου ή άλλων ιδιοτήτων όπως ο βαθμός, δηλαδή το πλήθος των ακμών ενός κόμβου στον γράφο. Υπάρχουν επίσης εργασίες για την προστασία της ιδιωτικότητας σε δεδομένα τροχιών [69, 54, 82, 58, 26]. Τέτοιες εργασίες δεν μπορούν να συγκριθούν άμεσα με τους αλγόριθμους που προτείνουμε καθώς αφορούν σε χωροχρονικά δεδομένα. Επιπλέον, έχουν γίνει εργασίες για την προστασία ιδιωτικότητας αραιών πολυδιάστατων δεδομένων για την προστασία από αποκάλυψη ευαίσθητων γνωρισμάτων [33] ή για διαφορική ιδιωτικότητα [21].

2.4.1 k^m-Ανωνυμία

Η k^m-ανωνυμία (k^m-anonymity) [68] είναι μια πιο χαλαρή εγγύηση ιδιωτικότητας που προτάθηκε για την ανωνυμοποίηση συνόλων από τιμές δεδομένων (set-valued data) τα οποία δεν υπακούουν σε κάποιο σχεσιακό σχήμα αλλά είναι σύνολα (itemsets) από μια δεξαμενή τιμών, όπως για παράδειγμα τα καλάθια των αγορών του σουπερμάρκετ. Η πιθανή γνώση του επιτιθέμενου περιορίζεται σε ένα υποσύνολο από τιμές της εγγραφής ενός στόχου και προσπαθεί μέσω αυτών να την εντοπίσει και να ανακαλύψει και τις υπόλοιπες. Η εγγύηση που προτείνεται απαιτεί να μην μπορεί ένας επιτιθέμενος που γνωρίζει μέχρι m τιμές μιας εγγραφής να την διαχωρίσει από τουλάχιστον $k-1$ άλλες. Η ιδέα αυτή βασίζεται στο ότι δεν είναι πάντα εύκολο να διαχωρίσει ποιές τιμές μπορεί να είναι ευαίσθητες και ποιές όχι, καθώς και ποιές είναι παρατηρήσιμες αντιστοίχως. Έτσι όλα τα γνωρίσματα μπορούν να θεωρηθούν ευαίσθητα και ψευδο-αναγνωριστικά, ανάλογα με την γνώση του επιτιθέμενου. Επίσης, το όριο στην γνώση του επιτιθέμενου σημαίνει ότι αν στην πραγματικότητα γνωρίζει όλες ή σχεδόν όλες τις τιμές μιας εγγραφής, τότε δεν απομένει πληροφορία να προστατευθεί από αυτόν. Οι παραδοχές αυτές είναι αρκετά ρεαλιστικές και γι΄ αυτόν το λόγο τις έχουμε διατηρήσει στο μοντέλο επίθεσης της παρούσας εργασίας, ενώ το μοντέλο δεδομένων είναι διαφορετικό. Στην ίδια κατεύθυνση οι [42] προσπάθησαν να αντιμετωπίσουν το πρόβλημα της ανωνυμοποίησης δεδομένων από σύνολα τιμών μέσω μιας προσέγγισης από πάνω-προσ-τα-κάτω εξειδίκευσης τιμών (top-down specialization) και τοπικής ανακωδικοποίησης. Αντί να αναπτύξουν ένα μοντέλο ιδιωτικότητας που προσαρμόζεται στην ειδική μορφή των δεδομένων, χρησιμοποίησαν την κλασσική k-ανωνυμία. Αυτό είχε ως αποτέλεσμα σημαντική απώλεια πληροφορίας στα δεδομένα, αν και υπήρξε εξισορρόπηση λόγω της επιλογή της τοπικής αποκωδικοποίησης που ευνοεί την διατήρηση μεγαλύτερης ακρίβειας στις τιμές. Σημειώνεται ότι καμία από τις παραπάνω εργασίες δεν λαμβάνει υπόψιν κάποια δομική πληροφορία για τις εγγραφές των χρηστών.

Οι εργασίες των [70, 81, 30] παρέχουν επίσης προστασία ιδιωτικότητας για αδόμητα δεδομένα όπου κάθε εγγραφή είναι ένα σύνολο τιμών. Στο [70] προτάθηκε η εφαρμογή αποσυσχέτισης σε δεδομένα συνόλων τιμών, όπου κάθε εγγραφή μπορεί να χωριστεί σε δύο ή περισσότερα υποσύνολα. Οι [81] και [30] ανωνυμοποιούν δεδομένα συναλλαγών. Οι μέθοδοι αυτές δεν λαμβάνουν υπόψη καμία δομική γνώση επίθεσης, καθώς διαχειρίζονται αδόμητα δεδομένα.

2.4.2 Πολυσχεσιακή k-Ανωνυμία

Η πολυσχεσιακή k-ανωνυμία (multi-relational k-anonymity) [56] είναι μια εγγύηση ιδιωτικότητας για δεδομένα αποθηκευμένα σε ξεχωριστούς πίνακες, οι οποίοι συνδέονται μεταξύ τους μέσω εξαρτήσεων ξένων κλειδιών. Οι εγγραφές κάθε ατόμου είναι αποθηκευμένες σε διαφορετικούς πίνακες μιας βάσης δεδομένων. Η k-ανωνυμοποίηση κάθε πίνακα ξεχωριστά δεν οδηγεί σε ασφαλή αποτελέσματα. Το σύνολο της πληροοφορίας κάθε ατόμου θα πρέπει να λαμβάνεται υπόψη κατά την ανωνυμοποίηση. Η πληροφορίες αυτές συλλέγονται με πολλαπλές συνδέσεις (joins) των πινάκων της βάσης πάνω στα ξένα κλειδιά. Με τον τρόπο αυτό σχηματίζονται δενδρικές δομές που περιέχουν κόμβους με τιμές γνωρισμάτων για κάθε χρήστη. Η διαδικασία της ανωνυμοποίησης γίνεται πάνω σε αυτές τις «δενδρικές εγγραφές», αλλά το αποτέλεσμα παρουσιάζεται σε ξεχωριστούς πίνακες σύμφωνα με το αρχικό σχήμα της βάσης των δεδομένων.

Το πλαίσιο ιδιωτικότητας που υιοθετείται στο [56] είναι αντίστοιχο της k-ανωνυμίας, έτσι κατά την ανωνυμοποίηση οι δενδρικές εγγραφές ομαδοποιούνται σε κλάσεις ισοδυναμίας μεγέθους τουλάχιστον k. Μέσα σε κάθε κλάση απαιτείται οι εγγραφές να είναι πανομοιότυπες ως προς τα ψευδο-αναγνωριστικά. Επειδή δεν έχει γίνει καμία προσπάθεια μοντελοποίησης και επεξεργασίας της δομικής πληροφορίας, κάθε φορά που απαλείφεται μια τιμή από εγγραφή που ανήκε σε ένα πίνακα της αρχικής βάσης θα πρέπει να απαλειφθούν και όλες οι τιμές εγγραφών πινάκων που συνδέονταν με αυτήν μέσω ξένων κλειδιών. Αυτό έχει ως αποτέλεσμα, κάθε φορά που απαλείφεται ένας κόμβος των «δενδρικών» εγγραφών να απαλείφεται και ολόκληρο το υποδένδρο κάτω από αυτόν, γεγονός που οδηγεί σε μεγάλη απώλεια πληροφορίας στα ανωνυμοποιημένα δεδομένα.

Στο Κεφάλαιο αυτό παρουσιάστηκε συνοπτικά η έρευνα που έχει γίνει στον τομέα της προστασίας της ιδιωτικότητας τα τελευταία χρόνια. Εκτενέστερη βιβλιογραφική επισκόπηση έχει γίνει στο [36].

Κεφάλαιο 3

Προστασία Ιδιωτικότητας Δεδομένων με Δενδρική Δομή

Σε αυτό το κεφάλαιο παρουσιάζονται οι μέθοδοί μας για την προστασία ιδιωτικότητας σε δημοσιεύσεις δεδομένων με δενδρική δομή, ως γενική περίπτωση των ημιδομημένων και XML δεδομένων. Αρχικά παρουσιάζεται το μοντέλο των δεδομένων και οι απειλές κατά τις ιδιωτικότητας από πιθανούς επιτιθέμενους που εκμεταλλεύονται και την γνώση της δομής των εγγραφών. Στη συνέχεια, προτείνεται ο βασικός αλγόριθμος ανωνυμοποίησης δεδομένων δενδρικής δομής και οι μετρικές κόστους που χρησιμοποιούνται για τον υπολογισμό της απώλειας πληροφορίας που προκαλεί η ανακωδικοποίηση και προτείνεται μια παραλλαγή του αλγορίθμου μας με σκοπό την μείωση του χρόνου εκτέλεσης της ανωνυμοποίησης. Τέλος, η μέθοδος αξιολογείται πειραματικά και συγκρίνεται με την πολυ-σχεσιακή k-ανωνυμία.

3.1 Κίνητρο και Συνεισφορά

Καθώς εταιρίες και οργανισμοί συλλέγουν πληροφορίες για τους χρήστες τους σε ολοένα και περισσότερο λεπτομερές επίπεδο, η ανησυχία γύρω από την προστασία της ιδιωτικότητας των χρηστών θέτει σημαντικές προκλήσεις στην κοινότητα διαχείρισης δεδομένων. Για αυτό τον λόγο έχουν προταθεί διάφορες τεχνικές ανωνυμοποίησης ώστε αν είναι δυνατή η επεξεργασία προσωπικών πληροφοριών χωρίς να τίθεται σε κίνδυνο η ιδιωτικότητα των χρηστών. Εντούτοις, ορισμένα πρακτικά θέματα όπως είναι οι εξαρτήσεις μεταξύ τιμών στα δεδομένα δεν είχαν μια ικανοποιητική λύση. Σε αυτό το κεφάλαιο εστιάζουμε στην ανωνυμοποίηση δεδομένων όπου οι εγγραφές έχουν δενδρική δομή λόγω σύνδεσης των τιμών τους με εξαρτήσεις.

Το σύνολο της πληροφορίας που αφορά σε ένα άτομο, σπάνια περιορίζεται σε μια μόνο πλειάδα τιμών στα σύγχρονα πληροφοριακά συστήματα. Συνήθως τα προσωπικά αυτά δεδομένα είναι διασκορπισμένα σε διάφορους πίνακες που συνδέονται μεταξύ τους με ξένα κλειδιά, ή είναι αποθηκευμένα σε πιο ευέλικτες μορφές όπως είναι οι εγγραφές XML. Οι κλασσικές μέθοδοι ανωνυμοποίησης ενός σχεσιακού πίνακα αποτυγχάνουν να ανωνυμοποιήσουν αποτελεσματικά τέτοιες εγγραφές με δενδρική δομή. Οι δομικές σχέσεις μεταξύ των τιμών διαφορετικών πεδίων διαφοροποιεί σημαντικά το πρόβλημα. Η μοναδική προσπάθεια επίλυσης αυτού

του προβλήματος στην υπάρχουσα βιβλιογραφία έγινε με την πολυ-σχεσιακή k-ανωνυμία [56]. Η δική μας προσέγγιση εξετάζει την πιο γενική περίπτωση των δεδομένων δενδρικής δομής και προτείνουμε μια μέθοδο που δεν βασίζεται μόνο στην γενίκευση των τιμών αλλά και στην απλοποίηση της δομής της εγγραφής.

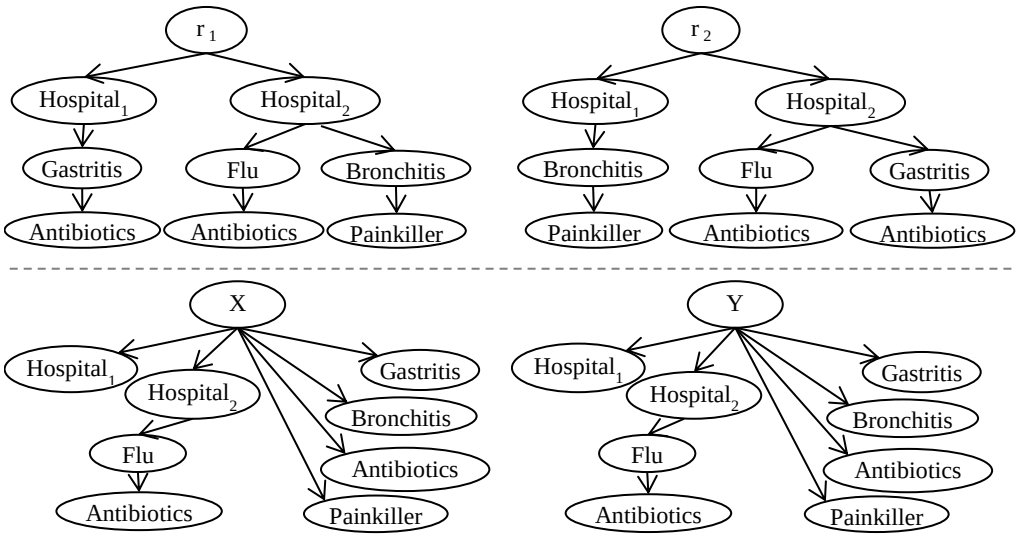
Ας θεωρήσουμε τις ιατρικές εγγραφές Σχήματος 3.1. Τα εικονιζόμενα δένδρα πάνω από την διαχωριστική γραμμή του σχήματος αναπαριστούν τις εγγραφές δυο ασθενών. Κάθε μονοπάτι περιγράφει ένα περιστατικό για το οποίο νοσηλεύθηκε ο ασθενής. Το πρώτο επίπεδο κόμβων κάτω από την ρίζα περιέχει την πληροφορία του νοσοκομείου στο οποίο διακομίστηκε ο ασθενής. Τα παιδιά των κόμβων-νοσοκομείων περιέχουν την διάγνωση της ασθένειας που έγινε σε ένα συγκεκριμένο νοσοκομείο. Στο τρίτο επίπεδο κάτω από την ρίζα φαίνεται η θεραπεία που έλαβαν για κάθε ασθένεια. Αν ένας κακόβουλος επιτιθέμενος γνωρίζει ότι ο ασθενής X έχει νοσηλευθεί στο νοσοκομείο «Hospital$_1$» και επίσης ότι έχει περάσει «Gastritis», δεν μπορεί να διακρίνει την εγγραφή του ασθενούς X από εκείνη του Y. Αντίθετα, αν ο επιτιθέμενος γνωρίζει επίσης ότι ο X νοσηλεύθηκε για «Gastritis» *στο* «Hospital$_1$», τότε μπορεί με βεβαιότητα να διεξάγει το συμπέρασμα ότι πρόκειται για την πάνω αριστερά εγγραφή του X. Για την πρόληψη τέτοιων επιθέσεων σύνδεσης μιας εγγραφής με ένα πραγματικό άτομο-στόχο του επιτιθέμενου, προτείνουμε μια μέθοδο προστασίας από την αποκάλυψη ταυτότητας. Οι λόγοι για τους οποίους εστιάζουμε στις επιθέσεις αποκάλυψης ταυτότητας είναι οι εξής:

(α′) Σε πολλές περιπτώσεις υπάρχουν περιορισμοί χρηστικότητας στα δεδομένα για κάποια πρακτική εφαρμογή. Τότε είναι πρακτικά αδύνατη η επιβολή πιο αυστηρών εγγυήσεων χωρίς την αχρήστευση των τελικών δεδομένων λόγω της αναπόφευκτα μεγάλης απώλειας σημαντικής πληροφορίας.

(β′) Είναι συχνά δύσκολος ο διαχωρισμός των τιμών σε ευαίσθητα γνωρίσματα και ψευδο-αναγνωριστικά. Στην πράξη κάθε τιμή που αφορά τον στόχο και μπορεί να τύχει να την μάθει ο επιτιθέμενος μπορεί δυνητικά να δράσει ως ψευδο-αναγνωριστικό.

(γ′) Το νομικό πλαίσιο περί προστασίας ιδιωτικότητας των περισσοτέρων χωρών εστιάζει στην μη-αποκάλυψη ταυτότητας. [4, 3, 5, 2, 1].

Σε αυτό το κεφάλαιο ορίζουμε την $k^{(m,n)}$-ανωνυμία, η οποία εγγυάται πως κανένας κακόβουλος επιτιθέμενος με γνώση έως m στοιχείων μιας εγγραφής και έως n *δομικών σχέσεων* μεταξύ των m στοιχείων δεν θα μπορεί να αντιστοιχίζει αυτή την γνώση σε λιγότερες από k εγγραφές στην ανωνυμοποιημένη συλλογή δεδομένων. Η διαδικασία ανωνυμοποίησης δεν αποκρύπτει μόνο την πληροφορία των τιμών των εγγραφών μέσω γενίκευσης, αλλά αποκρύπτει και δομική πληροφορία μέσω της απλοποίησης της δομής των εγγραφών. Η απλοποίηση αυτή συνίσταται στην αφαίρεση κόμβων από μονοπάτια μεγάλου μήκους και την δημιουργία μικρότερων μονοπατιών. Με αυτό τον τρόπο αποκρύπτονται ορισμένες σχέσεις μεταξύ των κόμβων που ανήκαν αρχικά στο ίδιο μονοπάτι. Επιστρέφοντας στο παράδειγμα του Σχήματος 3.1, μπορούμε να διασφαλίσουμε ότι ένας επιτιθέμενος που γνωρίζει ότι ο ασθενής νοσηλεύθηκε για «Gastritis» στο «Hospital$_1$», δεν θα μπορεί να διαχωρίσει τις δυο εγγραφές μεταξύ τους, με την τοποθέτηση του κόμβου «Gastritis» δίπλα στο «Hospital$_1$» ως αδελφό-κόμβο. Η τροπο-



Σχήμα 3.1: Δενδρικές εγγραφές μιας ιατρικής βάσης δεδομένων.

ποίηση αυτή φαίνεται στο κάτω μέρος του σχήματος. Εξετάζοντας τις δυο αυτές εγγραφές, ο επιτιθέμενος μπορεί να συμπεράνει ότι και οι δύο ασθενείς έχουν διακομιστεί στα νοσοκομεία «Hospital$_1$» και «Hospital$_2$», επίσης παρατηρεί ότι και οι δύο έχουν νοσήσει από «Gastritis». Η πληροφορία ότι ένας ασθενής νοσηλεύθηκε για γαστρίτιδα στο συγκεκριμένο νοσοκομείο έχει αποκρυφτεί, έτσι ο επιτιθέμενος δεν μπορεί πλέον να διαχωρίσει τις δύο εγγραφές με βάση την αρχική του γνώση για τον στόχο.

Η συνεισφορά του Κεφαλαίου 3 περιλαμβάνει τα ακόλουθα:

- Ορίζεται το πρόβλημα της ανωνυμοποίησης δεδομένων με δενδρική δομή και επεξηγείται λεπτομερώς πως η δομική πληροφορία μπορεί να δράσει ως ψευδο-αναγνωριστικό.

- Ορίζουμε την $k^{(m,n)}$-*ανωνυμία* μια νέα εγγύηση ιδιωτικότητας για δενδρικά δεδομένα και εξηγούμε την αποτελεσματικότητά της με ρεαλιστικά παραδείγματα.

- Εισάγουμε μια νέα πράξη για τον μετασχηματισμό των δεδομένων, την *δομική αποσυσχέτιση*, η οποία απλοποιεί την δομή των εγγραφών και παρέχει μεγαλύτερη ευελιξία στην διαδικασία της ανωνυμοποίησης.

- Ορίζουμε μια νέα μετρική για την αποτίμηση της απώλειας πληροφορίας η οποία λαμβάνει υπόψη τόσο τις δομικές απλοποιήσεις όσο και τις γενικεύσεις των τιμών.

- Προτείνουμε δύο νέους αλγόριθμους ανωνυμοποίησης. Ο πρώτος αλγόριθμος `ACS` εξερευνά από πάνω προς τα κάτω (top-down) τον χώρο των πιθανών λύσεων από τις πιο αυστηρές γενικεύσεις προς τις πιο λεπτομερείς τιμές. Για κάθε πιθανό συνδυασμό γενικεύσεων ψάχνει τις πιθανές δομικές απόσυσχετίσεις που θα τον οδηγήσουν σε μια λύση που ικανοποιεί την $k^{(m,n)}$-ανωνυμία. Λόγω του τεράστιου μεγέθους του χώρου των πιθανών λύσεων, ο `ACS` δεν κλιμακώνει καλά σε μεγάλα δεδομένα. Προτείνουμε

ένα ευρηστικό άπληστο αλγόριθμο (GCS), ο οποίος κλαδεύει δυναμικά τον χώρο λύσεων επιλέγοντας επιτόπου τις πιο υποσχόμενες υποψήφιες λύσεις σε κάθε βήμα.

- Διεξάγεται πειραματική μελέτη, καθώς και σύγκριση της μεθόδου με την πολυ-σχεσιακή k-ανωνυμία [56]. Η αποτίμηση δείχνει ότι η μέθοδός μας καταφέρνει να ανωνυμοποιήσει τα δεδομένα με μικρότερη απώλεια πληροφορίας. Τα πειραματικά αποτελέσματα δείχνουν ότι ο αλγόριθμος GCS κλιμακώνει πολύ καλύτερα με το μέγεθος των δεδομένων και βρίσκει λύση πολύ κοντά σε εκείνη του ACS στις περισσότερες περιπτώσεις που μελετήθηκαν.

3.2 Ορισμός του Προβλήματος

Εγγραφές οι οποίες αναφέρονται σε πρόσωπα και μπορούν να μοντελοποιηθούν ως δένδρα απαντώνται σε πολλές εφαρμογές. Ένα προφανές παράδειγμα είναι τα δεδομένα XML, αλλά και αντίστοιχες δενδρικές αναπαραστάσεις εγγραφών μπορούν να προκύψουν από βάσεις δεδομένων [56]. Η μορφή με την οποία είναι αποθηκευμένα τα αρχικά δεδομένα δεν υπαγορεύει τον τρόπο με τον οποίο θα επεξεργαστεί τα δεδομένα αυτά ο αλγόριθμος ανωνυμοποίησης. Το γεγονός ότι οι εγγραφές είναι αποθηκευμένες σε σχεσιακούς πίνακες δεν εγγυάται ότι η χρήση ενός αλγόριθμου που έχει σχεδιαστεί για σχεσιακά δεδομένα μπορεί να δώσει ένα ασφαλές αποτέλεσμα όταν εκτελεστεί για κάθε πίνακα ξεχωριστά. Η ανωνυμοποίηση που θα εφαρμοστεί θα πρέπει να λαμβάνει υπ' όψιν όλες τις πληροφορίες που ανήκουν σε κάθε χρήστη. Η δενδρική δομή των εγγραφών είναι ένας φυσικός τρόπος να μοντελοποιηθούν τα δεδομένα σε ένα μεγάλο εύρος εφαρμογών, ανεξαρτήτως της αρχικής αναπαράστασης των δεδομένων την οποία έχει επιλέξει ο κάτοχός τους.

Η μέθοδος ανωνυμοποίησης που προτείνουμε λαμβάνει υπ' όψιν και την δομική πληροφορία των δενδρικών εγγραφών κάθε ατόμου. Ένας πιθανός επιτιθέμενος μπορεί να γνωρίζει και τη δομική συσχέτιση των πληροφοριών που αφορούν ένα άτομο-στόχο. Για την αποτελεσματική αντιμετώπιση απειλών από επιτιθέμενους με τέτοιου είδους γνώση επεκτείνουμε την έννοια της k^m-ανωνυμίας [68] και ορίζουμε την $k^{(m,n)}$-ανωνυμία, η οποία παρέχει μια νέα εγγύηση ιδιωτικότητας που λαμβάνει υπ' όψιν την δομική γνώση του επιτιθέμενου.

pid	Patient
p1	Patient $_A$
p2	Patient $_B$
	…

hid	Hospital	pid
h1	Hospital $_1$	p1
h2	Hospital $_2$	p1
	…	

did	Disease	hid
d1	Gastritis	h1
d2	Bronchitis	h2
d3	Flu	h2
	…	

tid	Treatment	did
t1	Antibiotics	d2
t2	Pain killer	d3
	…	

Σχήμα 3.2: Εγγραφές μιας σχεσιακής ΒΔ από τις οποίες μπορεί να προκύψει η δενδρική εγγραφή του Σχήματος 3.3.

```
<Patient> Patient_A
  <Hospital> Hospital_1
    <Disease> Gastritis </Disease>
  </Hospital>
  <Hospital> Hospital_2
    <Disease> Flu
      <Treatment> Pain killer </Treatment>
    </Disease>
    <Disease> Bronchitis
      <Treatment> Antibiotics </Treatment>
    </Disease>
  </Hospital>
</Patient>
```

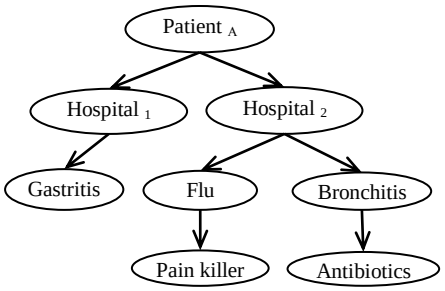
Πίνακας 3.1: Παράδειγμα εγγραφής XML από την οποία μπορεί να προκύψει η δενδρική εγγραφή του Σχήματος 3.3.

3.2.1 Μοντέλο Δεδομένων

Μοντελοποιούμε τα δεδομένα προς ανωνυμοποίηση D ως συλλογές εγγραφών με δενδρική δομή. Κάθε τέτοια εγγραφή $t \in D$ αντιστοιχεί σε ένα πρόσωπο. Η ρίζα της δενδρικής εγγραφής είναι ένας ψευδο-κωδικός που αντιπροσωπεύει το άτομο αυτό, ενώ όλοι οι υπόλοιποι κόμβοι παίρνουν τιμές από ένα πεδίο τιμών $\mathcal{I}$ και αντιστοιχούν στα στοιχεία του ατόμου. Θεωρούμε ότι στα δένδρα αυτά δεν υπάρχουν διπλότυπα κόμβων-αδελφών, έτσι έχουμε αταξινόμητα δένδρα γνωρισμάτων (unordered attribute trees). Όλες οι εγγραφές ακολουθούν ένα κοινό σχήμα που ορίζει την κλάση κάθε κόμβου, λ.χ. το όνομα της ετικέτας (tag) στην περίπτωση δεδομένων XML. Κάθε κλάση A αντιστοιχεί σε ένα γνώρισμα και έχει ένα πεδίο τιμών. Ένα μονοπάτι από τη ρίζα της εγγραφής ως οποιοδήποτε φύλλο δεν μπορεί να έχει περισσότερους από έναν κόμβους της ίδιας κλάσης. Επίσης, η σειρά με την οποία συναντάμε τις διαφορετικές κλάσεις στα μονοπάτια από τη ρίζα ως τα φύλλα είναι αντίστοιχη για όλες τις εγγραφές του συνόλου δεδομένων που επεξεργαζόμαστε. Αυτό σημαίνει ότι δεν γίνεται σε κάποιο μονοπάτι να έχουμε ένα κόμβο που αντιπροσωπεύει την τιμή μιας ασθένειας ως απόγονο ενός κόμβου που αντιπροσωπεύει την τιμή ενός νοσοκομείου και σε άλλο μονοπάτι ο κόμβος νοσοκομείου να είναι απόγονος του κόμβου της ασθένειας, είτε στην ίδια είτε σε άλλη εγγραφή της ίδιας συλλογής δεδομένων.

Το προτεινόμενο μοντέλο δεδομένων δεν περιορίζει τους τύπους των δεδομένων που μπορούμε να ανωνυμοποιήσουμε με την τεχνική που προτείνουμε παρακάτω, αντίθετα αντικατοπτρίζει τους περιορισμούς της γνώσης ενός πιθανού επιτιθέμενου. Τα αρχικά δεδομένα μπορεί παραδείγματος χάριν να ακολουθούν κάποια ταξινόμηση στη σειρά των κόμβων ή να επιτρέπουν την ύπαρξη διπλοτύπων. Αν ο επιτιθέμενος δεν γνωρίζει αυτήν την ταξινόμηση ή το πλήθος των διπλοτύπων, τότε η σειρά των κόμβων και τα διπλότυπα μπορούν να αγνοηθούν κατά την ανωνυμοποίηση.

Παράδειγμα 3.1. *Έστω η δενδρική εγγραφή του Σχήματος 3.3 όπου ο κόμβος με την*



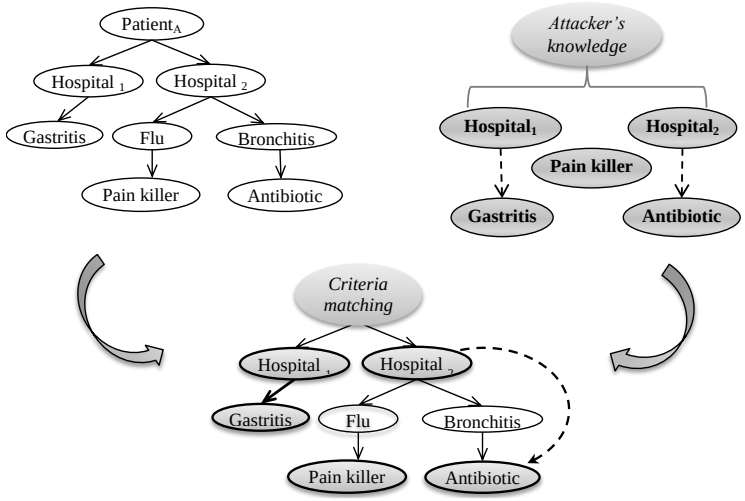
Σχήμα 3.3: Παράδειγμα δενδρικής εγγραφής ιατρικού ιστορικού ασθενούς.

ετικέτα "Gastritis" είναι απόγονος του κόμβου "Hospital$_1$". Αυτό σημαίνει πως ο ασθενής διακομίστηκε για γαστρίτιδα στο νοσοκομείο Hospital$_1$. Η δομή μπορεί να δηλώνεται άμεσα ή έμμεσα. Τα δεδομένα μπορεί αρχικά να ήταν αποθηκευμένα με ημιδομημένη μορφή όπως η εγγραφή XML του Πίνακα 3.1, ή θα μπορούσε να προκύψει από την συσχέτιση μεταξύ διαφορετικών εγγραφών που είναι αποθηκευμένοι σε σχεσιακούς πίνακες «Ασθένεια» και «Νοσοκομείο» και συνδέονται μεταξύ τους με ξένα κλειδιά, όπως απεικονίζεται στο Σχήμα 3.2.

3.2.2 Μοντέλο Επίθεσης

Στο σενάριο επίθεσης το οποίο αντιμετωπίζουμε θεωρούμε επιτιθέμενους οι οποίοι έχουν μερική γνώση της πληροφορίας που σχετίζεται με ένα άτομο-στόχο και θέλουν να την αξιοποιήσουν ώστε να ταυτοποιήσουν την εγγραφή του στα δεδομένα. Με αυτόν τον τρόπο ανακαλύπτουν την υπόλοιπη πληροφορία που υπάρχει για το άτομο. Εστιάζουμε σε επιθέσεις *αποκάλυψης ταυτότητας* για λόγους απλότητας αλλά και για να είναι εφαρμόσιμο στην πράξη όπου συχνά δεν είναι προφανής ο διαχωρισμός των γνωρισμάτων σε ευαίσθητα και μη. Θεωρούμε ότι οι επιτιθέμενοι μπορούν να έχουν μόνο θετική γνώση για τις τιμές των γνωρισμάτων καθώς και για την δομική σχέση τους μέσα σε μια εγγραφή. Δεν λαμβάνουμε υπ΄ όψιν επιτιθέμενους με αρνητική γνώση, όπως π.χ. ότι ένας ασθενής δεν έχει νοσήσει ποτέ από μια συγκεκριμένη πάθηση.

Η δομική γνώση που μπορεί να έχει ο επιτιθέμενος είναι η σχέση *προγόνου - απογόνου* μεταξύ δυο κόμβων πληροφορίας, δηλαδή γνωρίζει αν δυο κόμβοι ανήκουν στο ίδιο μονοπάτι. Συμβολίζουμε ότι ένας κόμβος b είναι απόγονος του κόμβου a ως $a \rightsquigarrow b$. Δεν μελετάται ξεχωριστά η ειδική περίπτωση της σχέσης *γονέα - παιδιού*, διότι αυτό συνεπάγεται αρνητική γνώση του επιτιθέμενου: θα έπρεπε να γνωρίζει ότι δυο κόμβοι είναι στο ίδιο μονοπάτι αλλά και ότι κανένας ενδιάμεσος κόμβος δεν υπάρχει ανάμεσά τους. Επιπλέον, η σειρά των κόμβων στο μονοπάτι είναι μια τυχαία επιλογή του κατόχου των δεδομένων: θα μπορούσε να επιλέξει να τοποθετήσει τα νοσοκομεία κάτω από τις ασθένειες η το αντίστροφο. Τέλος, θεωρούμε ότι η δομική γνώση που μπορεί να έχει ο επιτιθέμενος αφορά σε κόμβους των οποίων τις τιμές γνωρίζει. Δεν είναι δυνατόν να γνωρίζει δομικές σχέσεις μεταξύ κόμβων που του είναι άγνωστοι, λ.χ. ότι ένας κόμβος x ενός γνωρίσματος άγνωστης τιμής έχει ως απόγονο έναν άγνωστο κόμβο y, είτε θα γνωρίζει και τις τιμές των κόμβων είτε δεν θα γνωρίζει τίποτα για την δομική τους σχέση. Συνεπώς, αν ένας επιτιθέμενος γνωρίζει m τιμές μιας



Σχήμα 3.4: Σενάριο επίθεσης με στόχο την εγγραφή του Σχήματος 3.3.

εγγραφής $\{v_1, \ldots, v_m\}$, η δομική του γνώση θα αποτελείται από ένα σύνολο n ζευγών τιμών από το $\{v_1, \ldots, v_m\} \times \{v_1, \ldots, v_m\}$. Κάθε τέτοιο ζεύγος τιμών αντιστοιχεί σε κόμβους που συνδέονται με την σχέση προγόνου-απογόνου, δηλαδή εμφανίζονται σε κοινό μονοπάτι.

Ο επιτιθέμενος προσπαθεί να εκμεταλλευτεί τη γνώση που έχει πάνω σε τιμές κόμβων και στις δομικές σχέσεις μεταξύ τους για να αναζητήσει μέσα στο σύνολο των δεδομένων την εγγραφή του ατόμου-στόχου. Αν οι εγγραφές που ανταποκρίνονται στα κριτήριά του είναι λίγες ή αν είναι μία μοναδική τότε γίνεται παραβίαση της ιδιωτικότητας.

Παράδειγμα 3.2. *Ας θεωρήσουμε το σενάριο επίθεσης του Σχήματος 3.4 όπου ένας επιτιθέμενος προσπαθεί ναι ταιριάξει την γνώση του στην εγγραφή του Σχήματος 3.3. Γνωρίζει ότι ο ασθενής - στόχος έχει νοσηλευθεί για «γαστρίτιδα» στο νοσοκομείο «Hospital$_1$» και ότι του χορηγήθηκαν «αντιβιοτικά» στο Νοσοκομείο «Hospital$_2$». Αυτή η πληροφορία αναπαρίσταται ως δυο μονοπάτια "Hospital$_1$ ⇝ Gastritis" και "Hospital$_2$ ⇝ Antibiotic" τα οποία υποδηλώνουν την ύπαρξη αυτών των σχέσεων προγόνου-απογόνου στην εγγραφή του στόχου. Ο επιτιθέμενος γνωρίζει επίσης ότι ο ασθενής έχει λάβει «παυσίπονα» (Pain killers), αλλά δεν γνωρίζει σε ποιό νοσοκομείο και για ποιά ασθένεια. Έτσι ο κόμβος «παυσίπονο» εμφανίζεται χωρίς να είναι συνδεδεμένος με κάποιον άλλο στην αναπαράσταση της γνώσης του επιτιθέμενου. Ο κόμβος αυτός θα ρέπει να εμφανίζεται στη δενδρική εγγραφή του στόχου, αλλά η θέση του είναι άγνωστη. Απορρίπτοντας τις εγγραφές που δεν περιέχουν κάποιους από αυτούς τους κόμβους η τις σχέσεις μεταξύ τους, ο επιτιθέμενος μπορεί να συλλέξει τις εγγραφές που είναι πιθανόν να ανήκουν στο άτομο-στόχο.*

3.2.3 Εγγύηση Ιδιωτικότητας

Προτείνουμε τη $k^{(m,n)}$-ανωνυμία, μια νέα εγγύηση ιδιωτικότητας που προστατεύει την ταυτότητα ατόμων τα οποία σχετίζονται με δενδρικές εγγραφές από κακόβουλους επιτιθέμενους με τις ικανότητες που περιγράφηκαν στην Παράγραφο 3.2.2. Επεκτείνουμε την k^m-ανωνυμία [68]

ώστε να διαχειριστούμε την επιπλέον δομική γνώση του επιτιθέμενου. Η k^m-ανωνυμία εγγυάται ότι αν ένας επιτιθέμενος γνωρίζει έως m αντικείμενα από μια εγγραφή (σύνολο τιμών), δεν θα μπορεί να αναγνωρίσει λιγότερες από k εγγραφές στα δημοσιευμένα δεδομένα. Ορίζουμε την $k^{(m,n)}$-ανωνυμία ως εξής:

Ορισμός 3.1. ($k^{(m,n)}$-ανωνυμία)*: Μια δενδρική βάση δεδομένων D θεωρείται $k^{(m,n)}$-ανώνυμη αν οποιοσδήποτε επιτιθέμενος ο οποίος γνωρίζει το πολύ m τιμές γνωρισμάτων και το πολύ n δομικές σχέσεις (προγόνου-απογόνου) μεταξύ τους, δεν μπορεί χρησιμοποιώντας αυτή τη γνώση του να αναγνωρίσει λιγότερες από k δενδρικές εγγραφές στην D.*

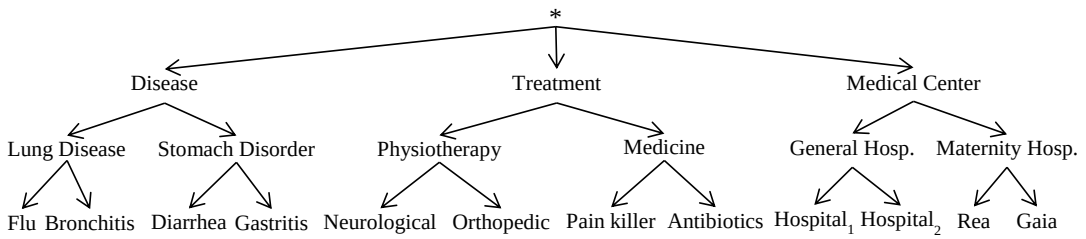
Ένα σημαντικό χαρακτηριστικό της $k^{(m,n)}$-ανωνυμίας είναι η υπόθεση ότι οποιοσδήποτε κόμβος ή δομική σχέση μπορεί να δράσει ως ψευδο-αναγνωριστικό. Αυτή η παραδοχή είναι τελείως διαφορετική από την κλασσική k-ανωνυμία όπου θεωρείται γνωστό εκ των προτέρων ποιά μέρη των εγγραφών είναι ψευδο-αναγνωριστικά. Όταν όλα τα πεδία μπορούν να δρουν ως ψευδο-αναγνωριστικά, η k-ανωνυμία θα δημιουργήσει ομάδες από k πανομοιότυπες εγγραφές. Συνεπώς, η k-ανωνυμία εισάγει μεγαλύτερη απώλεια πληροφορίας στα δεδομένα ενώ παρέχει μικρό όφελος από την πλευρά της ιδιωτικότητας: οι εγγραφές ανωνυμοποιούνται για να προληφθεί η αναγνώρισή τους από επιτιθέμενους που ήδη τις γνωρίζουν, δηλαδή από επιτιθέμενους που γνωρίζουν όλες τις τιμές τους.

Έχοντας ως κίνητρο αυτήν την παρατήρηση, στην $k^{(m,n)}$-ανωνυμία υποθέτουμε ότι ενώ οι επιτιθέμενοι μπορεί να γνωρίζουν *οποιοδήποτε* μέρος μιας δενδρικής εγγραφής, είναι απίθανο να γνωρίζουν ολόκληρη την εγγραφή. Επιπλέον, όταν ένας επιτιθέμενος γνωρίζει ολόκληρη την εγγραφή του στόχου, δεν προσφέρει κάτι η απόκρυψη της ταυτότητάς της στα δεδομένα εφόσον δεν απομένει τίποτα άλλο για να αποκαλυφθεί. Συνεπώς, η πρόληψη των επιθέσεων αποκάλυψης ταυτότητας από επιτιθέμενους με μερική γνώση των εγγραφών έχει μεγαλύτερη χρησιμότητα στην πράξη. Η ικανότητα παραμετροποίησης της $k^{(m,n)}$-ανωνυμίας επιτρέπει στον κάτοχο των δεδομένων να την προσαρμόσει στο επίπεδο προστασίας της ιδιωτικότητας που ταιριάζει καλύτερα στις ανάγκες των δεδομένων. Με αυτόν τον τρόπο, η $k^{(m,n)}$-ανωνυμία περιορίζει τις απώλειες πληροφορίας σε σχέση με την αυστηρότερη kανωνυμία, ενώ κλιμακώνει καλύτερα για δεδομένα πολλών διαστάσεων.

3.2.4 Πράξεις Ανακωδικοποίησης

Μια δενδρική βάση δεδομένων D η οποία δεν είναι $k^{(m,n)}$-ανώνυμη μπορεί να μετατραπεί σε μια ανακωδικοποιημένη μορφή D' η οποία θα ικανοποιεί τον ορισμό της $k^{(m,n)}$-ανωνυμίας. Οι μετασχηματισμοί των εγγραφών περιλαμβάνουν γενικεύσεις των τιμών των κόμβων, καθώς επίσης και δομικούς μετασχηματισμούς στα μονοπάτια των δένδρων. Η μέθοδός μας προβλέπει την αντικατάσταση των σπάνιων τιμών με κοινές πιο γενικές τιμές καθώς επίσης την απόκρυψη σχέσεων προγόνου-απογόνου όταν υπάρχει απειλή κατά της ιδιωτικότητας.

Γενίκευση. Υποθέτουμε την ύπαρξη μιας ιεραρχίας γενίκευσης τιμών (ΙΓΤ) για κάθε γνώρισμα (κλάση). Οι τιμές που ανήκουν στο πεδίο τιμών ενός γνωρίσματος A μπορούν να αντικατασταθούν από μια τιμή του επόμενου πιο γενικού επιπέδου που τις περιλαμβάνει. Αυτή μπορεί να γενικευθεί περαιτέρω στο επόμενο πιο γενικό επίπεδο κ.ο.κ. Η ιεραρχία γενίκευσης

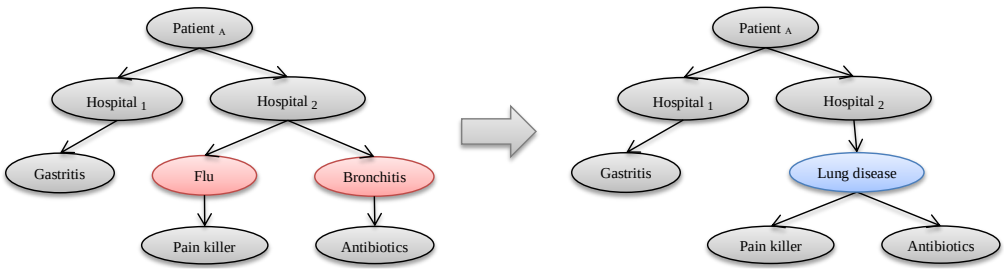


Σχήμα 3.5: Ιεραρχία Γενίκευσης Τιμών.

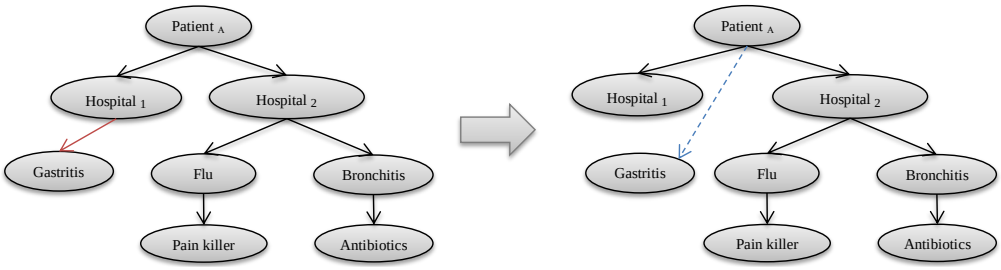
μπορεί να αναπαρασταθεί γραφικά υπό τη μορφή ενός δένδρου ιεραρχίας γενίκευσης όπως φαίνεται στο σχήμα 3.5. Όλες οι ιεραρχίες των κλάσεων βρίσκονται κάτω από μια κοινή ρίζα με την τιμή "*", η οποία σημαίνει «οποιαδήποτε» τιμή και ισοδυναμεί με την απαλοιφή της τιμής. Όταν μια *γενίκευση* $\{a_i \rightarrow a^*\}$ εφαρμόζεται σε μια δενδρική εγγραφή t, τότε όλοι οι κόμβοι της t που φέρουν την τιμή $a_i \in A$ υφίστανται αντικατάσταση της τιμής τους με την γενικευμένη τιμή a^*. Επιπλέον, η ίδια γενίκευση γίνεται σε όλες τις εγγραφές που περιέχουν την τιμή a_i, πραγματοποιούμε δηλαδή ολική ανακωδικοποίηση (global recoding).

Συγχώνευση. Όπως αναφέραμε στην Υποενότητα 3.2.1, οι αδελφοί κόμβοι (οι κόμβοι που έχουν κοινό γονέα) δεν μπορούν να έχουν ίδια τιμή μεταξύ τους. Αρχικά, αυτή η ιδιότητα ισχύει σε όλες της εγγραφές του συνόλου των δεδομένων. Μετά από κάποιες γενικεύσεις τιμών όμως μπορούν να προκύψουν διπλότυπα. Σε αυτές τις περιπτώσεις οι κόμβοι συγχωνεύονται σε έναν, ο οποίος διατηρεί την κοινή τους τιμή και το σύνολο των παιδιών τους. Παραδείγματος χάριν, στο Σχήμα 3.6 οι τιμές των κόμβων «Flu» και «Bronchitis» γενικεύονται σε «Lung disease». Για να διατηρηθούν οι ιδιότητες του μοντέλου των δεδομένων μας *συγχωνεύουμε* αυτούς τους κόμβους σε έναν που φέρει την κοινή τους τιμή και συνενώνουμε τα υποδένδρα τους, δηλαδή όλα τα μονοπάτια που βρίσκονταν κάτω από τους κόμβους «Flu» και «Bronchitis» στην αρχική εγγραφή τώρα είναι κάτω από το «Lung disease». Κατά την συνένωση των μονοπατιών εφαρμόζουμε επιπλέον συγχωνεύσεις αν τυχόν προκύψουν διπλότυπα.

Δομική Αποσυσχέτιση. Η δομική γνώση που μπορεί να έχει ο επιτιθέμενος για μια εγγραφή στόχο δεν μπορεί να αντιμετωπιστεί με την κλασική τεχνική της γενίκευσης τιμών. Αν ο επιτιθέμενος μπορεί γνωρίζει ότι δυο διαφορετικές τιμές a και b βρίσκονται στο



Σχήμα 3.6: Συγχώνευση των κόμβων που αντιστοιχούν στις ασθένειες «Γρίπη» και «Βρογχίτιδα» μετά την γενίκευσή τους σε «Πνευμονικό νόσημα».

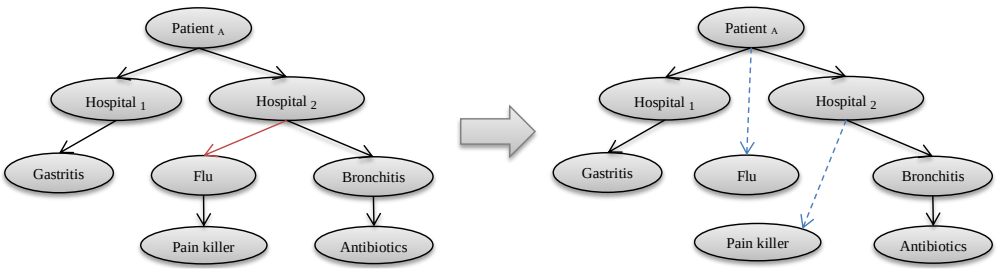


Σχήμα 3.7: Δομική αποσυσχέτιση των κόμβων ασθένειας «Gastritis» (γαστρίτιδα) και νοσοκομείου «Hospital$_1$», αποκρύπτοντας τη μεταξύ τους συσχέτιση.

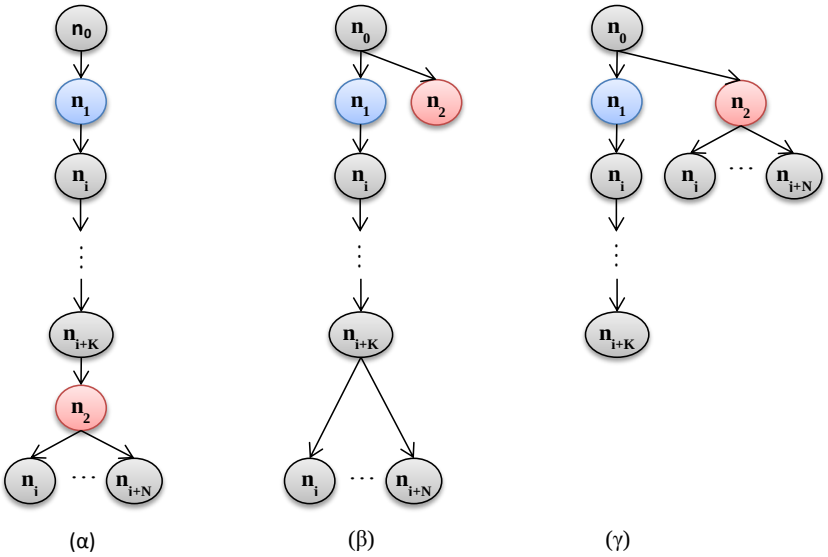
ίδιο μονοπάτι, και η σχέση των δύο αυτών τιμών είναι σπάνια ή μοναδική, ακόμη και αν η συχνότητα εμφάνισής τους ως τιμές (σε διαφορετικά μονοπάτια) είναι συχνή, τότε για να αποφύγουμε την παραβίαση της ιδιωτικότητας θα πρέπει να αποκρύψουμε αυτές τις σχέσεις προγόνου-απογόνου. Αυτή η *δομική αποσυσχέτιση* γίνεται με την απαλοιφή του μονοπατιού από το a στο b και οι δύο αυτοί κόμβοι εμφανίζονται σαν αδέλφια κάτω από τον γονέα του a. Όταν τα a και b εμφανίζονται σε πολλές εγγραφές χωρίς να είναι στο ίδιο μονοπάτι, αποκρύπτεται αυτή η σπάνια σχέση προγόνου - απογόνου $a \rightsquigarrow b$ στις λίγες εγγραφές που εμφανίζεται ένα τέτοιο μονοπάτι. Σε αυτήν την περίπτωση δεν είναι απαραίτητη ούτε χρήσιμη η γενίκευση τιμών, παρά μόνον ο δομικός μετασχηματισμός κάποιων μονοπατιών.

Όπως φαίνεται στο παράδειγμα του Σχήματος 3.7, η αρχική εγγραφή του ασθενούς προδίδει την πληροφορία ότι έχει νοσηλευθεί στο νοσοκομείο «Hospital$_1$» για «Γαστρίτιδα». Αν αυτήν την πληροφορία την γνωρίζει ο επιτιθέμενος, θα ψάξει για όλες τις εγγραφές οι οποίες περιέχουν ένα μονοπάτι με τις δύο αυτές τιμές. Αν θεωρήσουμε ότι το μονοπάτι αυτό είναι σπάνιο, παρόλο που οι τιμές του δεν είναι, τότε θα πρέπει να γίνει η δομική αποσυσχέτιση των δύο κόμβων, όπως φαίνεται δεξιά στο ίδιο σχήμα. Ενώ υπάρχει η πληροφορία ότι ασθενής έχει νοσήσει από γαστρίτιδα, ωστόσο δεν είναι σαφές αν έχει νοσηλευθεί για αυτήν στον «Hospital$_1$» ή στο «Hospital$_2$».

Αν ο απόγονος b που μετακινείται είχε στη δενδρική εγγραφή ως παιδιά τους κόμβους $c_1, \ldots, c_n$, τότε αυτοί οι κόμβοι γίνονται παιδιά του πρώην γονέα του b. Ο λόγος είναι για να διατηρηθούν όσο το δυνατόν περισσότερες σχέσεις της αρχικής εγγραφής μετά την δομική



Σχήμα 3.8: Δομική αποσυσχέτιση του κόμβου ασθένειας «Flu» (γρίπη) από το νοσοκομείο «Hospital$_2$», αποκρύπτοντας τη μεταξύ τους συσχέτιση.



Σχήμα 3.9: (α) Αρχική κατάσταση μονοπατιού. (β) Ορθός τρόπος δομικής αποσυσχέτισης του $n_1 \rightsquigarrow n_2$. (γ) Λανθασμένος τρόπος δομικής αποσυσχέτισης του $n_1 \rightsquigarrow n_2$.

αποσυσχέτιση του $a \rightsquigarrow b$. Ένα τέτοιο παράδειγμα φαίνεται στο Σχήμα 3.8. Λόγω της δομικής αποσυσχέτισης των κόμβων «Γρίπη» και «Hospital$_2$», ο κόμβος «Παυσίπονο» (Pain killer) υιοθετείται από τον πλησιέστερο πρόγονο «Hospital$_2$».

Συνοψίζοντας τα παραπάνω ορίζουμε την λειτουργία της δομικής αποσυσχέτισης μιας σχέσης προγόνου-απογόνου:

Ορισμός 3.2. (Δομική Αποσυσχέτιση)*: Έστω ένα μονοπάτι* $r \rightarrow \cdots p_a \rightarrow a \rightarrow \cdots p_b \rightarrow b \rightarrow c \rightarrow \cdots \rightarrow l$*. Η δομική αποσυσχέτιση της σχέσης* $a \rightsquigarrow b$ *στο μονοπάτι θα προκαλέσει την δημιουργία δύο νέων μονοπατιών* $r \rightarrow \cdots p_a \rightarrow a \rightarrow \cdots p_b \rightarrow c \rightarrow \cdots \rightarrow l$ *και* $r \rightarrow \cdots p_a \rightarrow b$*, τα οποία έχουν κοινό πρόθεμα* $r \rightarrow \cdots p_a$*.*

Αντίστοιχα με την ολική ανακωδικοποίηση που εφαρμόζουμε στην περίπτωση της γενίκευσης τιμών, και στην περίπτωση της δομικής αποσυσχέτισης όταν αποκρύπτουμε μια σχέση μεταξύ προγόνου a και απογόνου b, η μετατροπή αυτή εφαρμόζεται σε όλες τις εγγραφές που περιέχουν μονοπάτια $a \rightsquigarrow b$.

Αξίζει να σημειωθεί ότι η δομική αποσυσχέτιση μεταξύ προγόνου- απογόνου θα μπορούσε να οριστεί με διαφορετικό τρόπο: Αντί να μετακινηθεί μόνο ο κόμβος b, το υποδένδρο των απογόνων του θα μπορούσε να ακολουθήσει, παραμένοντας κάτω από το b. Οι δύο εναλλακτικοί τρόποι απεικονίζονται γραφικά στο Σχήμα 3.9. Αριστερά φαίνεται η μορφή ενός μονοπατιού στα αρχικά δεδομένα, όπου θεωρούμε ότι η σχέση $n_1 \rightsquigarrow n_2$ είναι σπάνια και πρέπει να αποκρυφθεί κατά την ανωνυμοποίηση.

Στο μονοπάτι από τον n_1 ως τον n_2 παρεμβάλλονται K το πλήθος κόμβοι, ενώ στο υποδένδρο κάτω από τον n_2 ανήκουν N το πλήθος κόμβοι της εγγραφής. Επιλέγοντας να υλοποιήσουμε την δομική αποσυσχέτιση όπως ορίσαμε παραπάνω προκύπτουν τα 2 μονοπάτια κάτω από το n_0 όπως φαίνεται στο Σχήμα 3.9(β). Οι δομικές σχέσεις προγόνου-απογόνου

που αποκρύφθηκαν είναι $K + N + 1$: οι N σχέσεις του n_2 με όλους τους απογόνους του και οι $K + 1$ σχέσεις με όλους τους προγόνους του, συμπεριλαμβανομένου και του n_1. Αν ακολουθήσουμε τον εναλλακτικό τρόπο, όπως φαίνεται στο Σχήμα 3.9(γ), τότε οι δομικές σχέσεις που αποκρύπτονται στα τελικά δεδομένα είναι $(K+1)\times(N+1) = K\times N+K+N+1$. Συνεπώς, με τον εναλλακτικό τρόπο χάνεται επιπλέον πληροφορία για $K \times N$ δομικές σχέσεις, είναι οι σχέσεις προγόνου-απογόνου μεταξύ των προγόνων του n_2 στο μονοπάτι και των απογόνων του στο υποδένδρο του. Δεδομένου ότι αυτή η επιπλέον πληροφορία χάνεται χωρίς να συντρέχουν λόγοι παραβίασης ιδιωτικότητας, και θέλοντας να διατηρήσουμε όσο το δυνατόν καλύτερη την ποιότητα των ανωνυμοποιημένων δεδομένων, επιλέγουμε την πρώτη εναλλακτική στην οποία χάνονται οι λιγότερες δομικές σχέσεις.

3.2.5 Μετρικές Αποτίμησης Απώλειας Πληροφορίας

Η ανακωδικοποίηση κατά την διαδικασία της ανωνυμοποίησης προκαλεί απώλειες πληροφορίας στα προς δημοσίευση δεδομένα. Η αποτίμηση αυτής της απώλειας είναι σημαντική δεδομένου ότι όσο λιγότερη πληροφορία χάνεται τόσο καλύτερη είναι η ποιότητα των ανωνυμοποιημένων δεδομένων για μελέτες όπως η στατιστική ανάλυση. Τόσο οι γενικεύσεις των τιμών των γνωρισμάτων όσο και οι δομικές αποσυσχετίσεις μεταξύ κόμβων αποκρύπτουν πληροφορίες των δενδρικών εγγραφών. Το κρίσιμο σημείο στο συγκεκριμένο πρόβλημα είναι να διατυπώσουμε μετρικές στις οποίες θα είναι συγκρίσιμες αυτές οι δύο μορφές απώλειας πληροφορίας: τόσο των τιμών όσο και της δομής των εγγραφών. Επιθυμούμε να μετρήσουμε την μείωση της εκφραστικότητας μιας ανωνυμοποιημένης δενδρικής εγγραφής συγκριτικά με την αντίστοιχη εγγραφή των αρχικών δεδομένων.

Μετρική Αντιστρόφου Πεδίου Μονοπατιών (RPD)

Στην περίπτωση των δενδρικών εγγραφών η μείωση της εκφραστικότητας των δεδομένων μετά τους μετασχηματισμούς ανωνυμοποίησης εκτιμάται ως προς το πλήθος των πιθανών μονοπατιών που μπορούν να εκφράσουν τα μετασχηματισμένα δεδομένα. Η επιλογή αυτή γίνεται διότι τα δένδρα είναι σύνολα μονοπατιών όπως αντίστοιχα οι σχεσιακές εγγραφές είναι σύνολα τιμών. Ορίζουμε την έννοια του *πεδίου μονοπατιού* για να ποσοτικοποιήσουμε την χρησιμότητα της πληροφορίας που φέρει σε σχέση με τα αρχικά δεδομένα.

Έστω το μονοπάτι $p = a_1 \to b_1 \to c_1$, όπου οι τιμές a_1, b_1, c_1 είναι αρχικές μη-γενικευμένες τιμές με πεδία τιμών κλάσεων τα $\mathcal{A}, \mathcal{B}, \mathcal{C}$ αντιστοίχως. Το αρχικό *πεδίο μονοπατιού* $\mathcal{I}_p$ ορίζεται ως $\mathcal{I}_p = \mathcal{A} \to \mathcal{B} \to \mathcal{C}$ και έχει εύρος τιμών $|\mathcal{I}_p| = |\mathcal{A}| \times |\mathcal{B}| \times |\mathcal{C}|$. Ας υποθέσουμε ότι κατά την ανωνυμοποίηση γενικεύεται η τιμή a_1 σε A_1 και ότι το $C(A_i)$ είναι το πλήθος των διαφορετικών τιμών που υπάρχουν στο ίδιο επίπεδο της ιεραρχίας γενίκευσης με την τιμή A_i, στην ίδια κλάση. Τότε το μονοπάτι p μετασχηματίζεται σε ένα *γενικευμένο μονοπάτι* $p_g = A_1 \to b_1 \to c_1$. Το εύρος τιμών του πεδίου $\mathcal{I}_{p_g}$ του νέου μονοπατιού p_g είναι $|\mathcal{I}_{p_g}| = |C(A_1)| \times |\mathcal{B}| \times |\mathcal{C}|$. Η μετρική Αντίστροφου Πεδίου Μονοπατιού RPD ορίζεται ως εξής:

$$RPD(p_g) = \frac{1}{\mathcal{I}_{p_g}} = \frac{1}{d(A_1) \times |C(A_1)| \times d(b_1) \times |\mathcal{B}| \times d(c_1) \times |\mathcal{C}|}$$

όπου $d()$ είναι η συνάρτηση που επιστρέφει το *βάθος* ενός κόμβου, δηλαδή την απόστασή του από την ρίζα της εγγραφής. Η λογική πίσω από τον παράγοντα $d()$ είναι ότι οι κόμβοι που είναι πιο κοντά στην ρίζα είναι πιο σημαντικοί. Ο ισχυρισμός αυτός επαληθεύτηκε πειραματικά. Η μετρική RPD ενός τυχαίου μονοπατιού $p = u_1 \to \cdots \to u_n$ ορίζεται ως:

$$RPD(p) = \frac{1}{\mathcal{I}_p} = \frac{1}{d(u_1) \times |C(u_1)| \times \cdots \times d(u_n) \times |C(u_n)|} \tag{3.1}$$

Η μετρική RPD μιας δενδρικής εγγραφής t ορίζεται ως ο μέσος όρος των τιμών RPD για κάθε ξεχωριστό μέγιστο μονοπάτι από την ρίζα της εγγραφής προς κάθε φύλλο p:

$$RPD(t) = \frac{1}{lvs_t} \sum_{p \in t} RPD(p) \tag{3.2}$$

όπου lvs_t είναι το πλήθος των φύλλων της εγγραφής t.

Από τους παραπάνω ορισμούς είναι προφανές ότι κάθε γενίκευση τιμής θα μειώνει τον παρονομαστή κάποιων μονοπατιών, επομένως θα αυξάνει την τιμή RPD των μονοπατιών αυτών και κατ΄ επέκταση θα αυξάνει τον μέσο όρο του RPD των μονοπατιών της εγγραφής.

Πόρισμα 3.1. *Η γενίκευση τιμών πάντα αυξάνει την τιμή της μετρικής απώλειας πληροφορίας $RPD(t)$.*

Σημειώνουμε ότι δεν χρειάζεται να ληφθούν άμεσα υπόψη οι δομικοί μετασχηματισμοί, διότι έμμεσα επηρεάζουν την τιμή της μετρικής $RPD(t)$. Παραδείγματος χάριν, ας υποθέσουμε ότι στο μονοπάτι $p = a_1 \to b_1 \to c_1$ η σχέση $b_1 \rightsquigarrow c_1$ πρέπει να αποκρυφθεί για λόγους ιδιωτικότητας. Αυτό έχει ως αποτέλεσμα την δημιουργία δύο μονοπατιών $p_1 = a_1 \to b_1$ και $p_2 = a_1 \to c_1$ με κοινό πρόθεμα το a_1. Μπορούμε εύκολα να παρατηρήσουμε ότι το RPD καθενός από αυτά τα μονοπάτια είναι πάντα μεγαλύτερο από το RPD του αρχικού μονοπατιού, δηλαδή ισχύει ότι $RPD(p) < RPD(p_1)$ και $RPD(p) < RPD(p_2)$. Συνεπώς το $RPD(p)$ θα είναι επίσης μικρότερο του μέσου όρου RPD των p_1 και p_2.

Πόρισμα 3.2. *Η δομική αποσυσχέτιση πάντα αυξάνει την τιμή της μετρικής απώλειας πληροφορίας $RPD(t)$.*

Παράδειγμα 3.3. *Έστω η ιεραρχία γενίκευσης τιμών του Σχήματος 3.5. Η RPD των μετασχηματισμένων εγγραφών των Σχημάτων 3.6 και 3.8 είναι αντίστοιχα:*

$RPD(r_{3.6}) = \frac{1}{3} \cdot (\frac{1}{1 \cdot 12 \cdot 2 \cdot 12} + 2\frac{1}{1 \cdot 12 \cdot 2 \cdot 6 \cdot 3 \cdot 12}) = 0.00128$

$RPD(r_{3.8}) = \frac{1}{4} \cdot (2\frac{1}{1 \cdot 12 \cdot 2 \cdot 12 \cdot} + \frac{1}{1 \cdot 12} + \frac{1}{1 \cdot 12 \cdot 2 \cdot 12 \cdot 3 \cdot 12}) = 0.0225$

Η μετρική RPD για ένα σύνολο δεδομένων D ορίζεται ως ο μέσος όρος των τιμών RPD όλων των δενδρικών του εγγραφών t:

$$RPD(D) = \frac{1}{|D|} \sum_{t \in D} RPD(t) \tag{3.3}$$

όπου $|D|$ είναι το συνολικό πλήθος των δενδρικών εγγραφών του D.

Απώλεια Εξόρυξης Πολλαπλών Επιπέδων (ML^2)

Η αποτίμηση της απώλειας πληροφορίας λόγω ανωνυμοποίησης είναι ένα πολύπλοκο έργο διότι τα τελικά δεδομένα μπορεί να χρησιμοποιηθούν για διαφορετικούς τρόπους ανάλυσης. Αυτό είναι ένα πρόβλημα που απαντάται σε κάθε τύπο ανωνυμοποίησης και για το οποίο δεν υπάρχει μια κοινά αποδεκτή βέλτιστη απάντηση. Για να έχουμε μια πληρέστερη εικόνα του βαθμού στον οποίο επηρεάζεται η ποιότητα των δεδομένων από την $k^{(m,n)}$-ανωνυμοποίηση, χρησιμοποιούμε πολλαπλές μετρικές που αποτιμούν διαφορετικές πλευρές της χρηστικότητας της πληροφορίας.

Σε πολλές εφαρμογές εξόρυξης δεδομένων (data mining) είναι σημαντικό να διατηρείται η πληροφορία των αντικειμένων που έχουν τις μεγαλύτερες συχνότητες εμφάνισης στα δεδομένα. Η μετρική Απωλειών Εξόρυξης Πολλαπλών Επιπέδων (multiple level mining loss metric - ML^2) [70, 34] εκφράζει την απώλεια πληροφορίας στην ανίχνευση συχνών συνόλων τιμών από διάφορα επίπεδα γενίκευσης, όταν γίνεται εξόρυξη πληροφορίας από τα ανωνυμοποιημένα δεδομένα αντί των αρχικών δεδομένων. Όταν μια συλλογή δεδομένων D ανωνυμοποιείται και παράγεται η νέα ανώνυμη συλλογή D', η μετρική απωλειών εξόρυξης πολλαπλών επιπέδων υπολογίζεται ως εξής:

$$ML^2 = 1 - \frac{\sum_{i=0}^{h} FI_{D'}^i}{\sum_{i=0}^{h} FI_D^i}$$

όπου: FI_D^i είναι το πλήθος των συχνών συνόλων τιμών (frequent itemsets) στα αρχικά δεδομένα, προβεβλημένα στο i-στό επίπεδο της Ιεραρχίας Γενίκευσης, και $FI_{D'}^i$ είναι το πλήθος των συχνών συνόλων τιμών στα ανωνυμοποιημένα δεδομένα, με τιμές στο i-στό επίπεδο της ιεραρχίας γενίκευσης.

Η εξόρυξη συχνών συνόλων τιμών από τα δεδομένα σε διαφορετικά επίπεδα της ιεραρχίας γενίκευσης επιτρέπει την ανίχνευση κανόνων συσχέτισης και συχνών συνόλων που μπορεί να μην εμφανίζονται από την εξόρυξη στα αρχικά μόνο δεδομένα [40, 39]. Ο λόγος είναι ότι τα αρχικά δεδομένα περιέχουν πιο λεπτομερείς τιμές και κατά συνέπεια μικρότερες συχνότητες εμφάνισης σε σχέση με τα γενικευμένα δεδομένα.

Στην περίπτωση των δενδρικών δεδομένων, όπου η δομική πληροφορία ενδιαφέρει εξίσου με τις τιμές των γνωρισμάτων, η εξόρυξη δεδομένων αναζητά να εντοπίσει τα πιο *συχνά υποδένδρα* που εμφανίζονται στο σύνολο των εγγραφών. Η ML^2 μπορεί να προσαρμοστεί για συχνά υποδένδρα ως εξής:

$$ML^2 = 1 - \frac{\sum_{i=0}^{h} FT_i(D')}{\sum_{i=0}^{h} FT_i(D)} \tag{3.4}$$

όπου $FT_i()$ είναι το πλήθος των συχνών υποδένδρων, στο i-στο επίπεδο γενίκευσης. Υπολογίζουμε τα $FT_i(D)$ και $FT_i(D')$ με τον εξής τρόπο: Προβάλουμε τα αρχικά δεδομένα σε όλα τα επίπεδα της ιεραρχίας γενίκευσης (οριζόντιες τομές της ιεραρχίας που περιέχουν τιμές του ίδιου επιπέδου γενίκευσης) και βρίσκουμε τα συχνά υποδένδρα για κάθε επίπεδο γενίκευσης i. Ο συνολικός αριθμός αυτών των υποδένδρων είναι ο παρονομαστής $\sum_{i=0}^{h} FT_i(D)$. Ακολουθούμε την ίδια διαδικασία για τα ανωνυμοποιημένα δεδομένα D', αλλά σε αυτή την περίπτωση

δεν μπορεί να γίνει η προβολή μιας ήδη γενικευμένης τιμής σε πιο εξειδικευμένα (χαμηλότερα) επίπεδα της ιεραρχίας γενίκευσης από ότι το τρέχον επίπεδό της.

Ουσιαστικά η μετρική αυτή εντοπίζει το ποσοστό των συχνών υποδένδρων τα οποία δεν διατηρήθηκαν στα τελικά δεδομένα λόγω των γενικεύσεων και των δομικών αποσυσχετίσεων που έγιναν κατά τη διαδικασία της ανωνυμοποίησης.

Παράδειγμα 3.4. *Ας υποθέσουμε ότι το "r →Hospital$_2$→Flu" είναι ένα συχνό υποδένδρο στα αρχικά δεδομένα. Τότε τα υποδένδρα "r →General Hospital→Lung Disease" και "r →Hospital→Disease" θα είναι επίσης συχνά υποδένδρα στα δυο επόμενα επίπεδα γενίκευσης, με βάση την ιεραρχία του Σχήματος 3.5. Αυτά τα τρία υποδένδρα θα συμβάλουν στον παρονομαστή της μετρικής ML^2. Έστω ότι η ανωνυμοποίηση καταλήγει σε μια τομή που περιέχει τις τιμές "Hospital$_2$" και "Lung Disease". Σε αυτήν την περίπτωση, το συχνό υποδένδρο "r →Hospital$_2$→Flu" δεν θα εμφανίζεται στα ανωνυμοποιημένα δεδομένα. Αντίθετα τα δυο άλλα συχνά υποδένδρα "r →General Hospital→Lung Disease" και "r →Hospital→Disease" θα μπορούν να εξωρυχθούν καθώς οι τιμές τους είναι πάνω από την τομή γενίκευσης. Έτσι ο αριθμός 2 θα προστεθεί στον αριθμητή του ML^2. Αν το μονοπάτι "Hospital$_2$⇝Lung Disease" είχε υποστεί δομική αποσυσχέτιση, τότε κανένα από τα τρία αρχικά συχνά υποδένδρα δεν θα μπορούσε να εξορυχθεί από το αποτέλεσμα της ανωνυμοποίησης.*

Διαφορική Απώλεια Εξόρυξης Πολλαπλών Επιπέδων (dML^2)

Η μετρική ML^2 δεν μπορεί να περιγράψει ποιοτικά το βαθμό αλλοίωσης των συχνών υποδένδρων, παρά μόνο το ποσοστό αυτών που αλλοιώθηκαν σε σχέση με τα αρχικά συχνά υποδένδρα. Είναι δυνατό να επεκταθεί ο ορισμός αυτής της μετρικής ώστε να αποτιμηθεί ο βαθμός απώλειας πληροφορίας στο σύνολο των συχνών υποδένδρων.

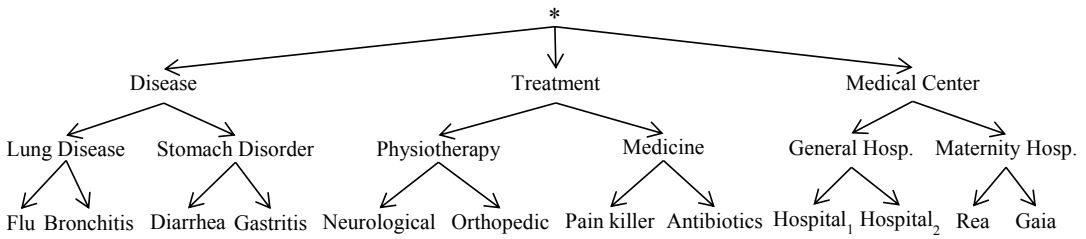
Η Διαφορική Μετρική Απωλειών Εξόρυξης Πολλαπλών Επιπέδων (differential multiple level mining loss metric - dML^2) [70] μπορεί να οριστεί για στο σενάριο των δενδρικών δεδομένων ως εξής:

$$dML^2 = \frac{\sum_{i=0}^{h} \sum_{ft \in \mathcal{FT}_i(D)} dtree(ft|D \rightarrow D')}{\sum_{i=0}^{h} FT_i(D)} \tag{3.5}$$

όπου $\mathcal{FT}_i(D)$ είναι το σύνολο των συχνών υποδένδρων ft στα αρχικά δεδομένα, προβεβλημένα στο i-στό επίπεδο της Ιεραρχίας Γενίκευσης, η συνάρτηση $dtree(ft|D \rightarrow D')$ υπολογίζει τη «διαφορά» μεταξύ ενός υποδένδρου ft που υπήρχε στα αρχικά δεδομένα και της ανώνυμης εκδοχής του στο σύνολο των ανωνυμοποιημένων δεδομένων D'. Η διαφορά αυτή υπολογίζεται τόσο ως προς τις τιμές που έχουν προβληθεί σε διαφορετικά επίπεδα γενίκευσης, όσο και ως προς τις δομικές αλλοιώσεις των μονοπατιών λόγω δομικής αποσυσχέτισης.

Ορίζω τη συνάρτηση $dtree(ft|D \rightarrow D')$ ως το μέσο όρο των διαφορών τιμών κόμβων και δομής, μεταξύ αρχικού και ανωνυμοποιημένου υποδένδρου ft:

$$dtree(ft \mid D \rightarrow D') = 0.5\frac{\sum_{v \in ft} dlevel(v, v')}{N(ft)} + 0.5\frac{BR(ft)}{R(ft)}$$



Σχήμα 3.10: Οριζόντια τομή στην ιεραρχία γενίκευσης τιμών.

όπου $N(ft)$ είναι ο συνολικός αριθμός των κόμβων που ανήκουν στο υποδένδρο ft, $R(ft)$ είναι ο συνολικός αριθμός των σχέσεων προγόνου-απογόνου μεταξύ κόμβων που ανήκουν στο υποδένδρο ft, και $BR(ft)$ είναι ο αριθμός των σχέσεων προγόνου-απογόνου που αποσυσχετίστηκαν κατά την ανωνυμοποίηση του ft. Η συνάρτηση $dlevel(v, v')$ επιστρέφει την διαφορά επιπέδου στην Ιεραρχία Γενίκευσης μεταξύ της αρχικής τιμής του κόμβου v και της τελικής του τιμής v' στα ανωνυμοποιημένα δεδομένα διά του ύψους της ιεραρχίας γενίκευσης.

Παράδειγμα 3.5. *Επιστρέφοντας στα συχνά υποδένδρα του Παραδείγματος 3.4, θα υπολογίσουμε την τιμή της dtree για το συχνό υποδένδρο "r → Hospital$_2$ → Lung Disease", δοσμένου του αρχικού υποδένδρου "r → Hospital$_2$ → Flu", ως* $0.5 \cdot \frac{1/3}{2} + 0.5 \cdot 0 = 0.083$*, διότι το ύψος της ιεραρχίας γενίκευσης είναι 3 και η διαφορά μεταξύ των επιπέδων γενίκευσης των τιμών "Flu" και "Lung Disease" είναι 1. Αν επιπλέον η σχέση "Hospital$_2$ ⇝ Lung Disease" είχε υποστεί δομική αποσυσχέτιση, τότε η τιμή της dtree θα ήταν* $0.5 \cdot \frac{1/3}{2} + 0.5 \cdot \frac{1}{1} = 0.583$.

3.3 Αλγόριθμος Ανωνυμοποίησης

Στην ενότητα αυτή παρουσιάζονται οι κύριες δομές που χρησιμοποιούνται στον προτεινόμενο αλγόριθμο ανωνυμοποίησης, εξηγείται το πως αξιοποιούνται οι τεχνικές μετασχηματισμού δενδρικών δεδομένων που αναλύθηκαν στις προηγούμενες ενότητες και αναλύονται τα βασικά βήματα του αλγορίθμου.

Σε ότι αφορά τις γενικεύσεις τιμών, ο χώρος των υποψήφιων λύσεων, δηλαδή των μετασχηματισμών των δεδομένων, αντιστοιχεί στο σύνολο των πιθανών οριζόντιων τομών που μπορούν να γίνουν στο δένδρο της ιεραρχίας γενίκευσης. Κάθε τέτοια τομή ορίζει ένα μοναδικό σύνολο κανόνων γενίκευσης. Παραδείγματος χάριν, η οριζόντια τομή του Σχήματος 3.10 περιέχει τις τιμές (Lung Disease, Diarrhea, Gastritis, Neurological, Orthopedic, Medicine, Hospital$_1$, Hospital$_2$, Rea, Gaia). Η τομή αυτή υποδηλώνει τους κανόνες γενίκευσης {Flu, Bronchitis}→Lung Disease και {Pain killer, Antibiotics}→ Medicine.

Το προς επίλυση πρόβλημα που μελετάμε μπορεί να διατυπωθεί ως εξής: *Δοσμένου ενός συνόλου δεδομένων D, των παραμέτρων της $k^{(m,n)}$-ανωνυμίας και μιας ιεραρχίας γενίκευσης τιμών H, επιθυμούμε να μετασχηματίσουμε το D μέσω γενικεύσεων και δομικών αποσυσχετίσεων σε ένα νέο σύνολο D' για το οποίο ικανοποιείται η $k^{(m,n)}$-ανωνυμία και η απώλεια πληροφορίας έχει περιοριστεί στο ελάχιστο δυνατόν.* Η λύση του προβλήματος είναι ένα ζεύγος (C, SD) μιας *τομής* στην ιεραρχία γενίκευσης C και ενός συνόλου κανόνων *δομικής*

αποσυσχέτισης SD. Από την εφαρμογή των (C, SD) στο σύνολο δεδομένων D προκύπτει ένα $k^{(m,n)}$-ανώνυμο σύνολο δεδομένων D'. Λόγω του βαθμού δυσκολίας του προβλήματος, το οποίο εξηγούμε ότι είναι NP-Hard, προτείνουμε ένα ευρηστικό αλγόριθμο που βρίσκει μια τοπικά βέλτιστη αλλά όχι πάντα ολικά βέλτιστη λύση. Για τον υπολογισμό της απώλειας πληροφορίας των υποψηφίων λύσεων σε κάθε βήμα του αλγορίθμου χρησιμοποιούμε ως συνάρτηση αποτίμησης την RPD_a, μια παραλλαγή της μετρικής RPD, που ορίζουμε παρακάτω.

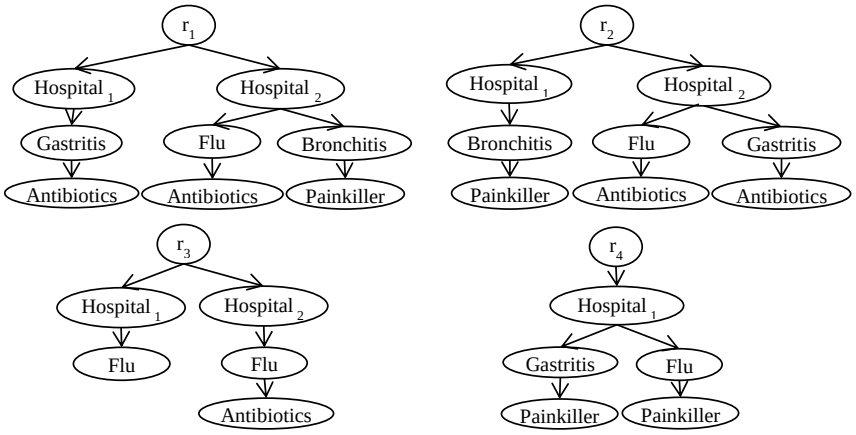
Για κάθε τομή της ιεραρχίας γενίκευσης υπάρχουν πολλαπλές διαφορετικές λύσεις που προκύπτουν από κανόνες δομικών αποσυσχετίσεων. Ο πλήρης χώρος των λύσεων περιλαμβάνει το σύνολο των κανόνων γενίκευσης που υποδηλώνονται από τις τομές της ιεραρχίας επί τους διαφορετικούς κανόνες δομικών αποσυσχετίσεων. Η εύρεση της βέλτιστης λύσης είναι NP-Hard. Αυτό προκύπτει από το γεγονός ότι η εύρεση λύσης της κλασσικής k-ανωνυμίας για ένα σχεσιακό πίνακα, η οποία έχει δειχθεί ότι είναι NP-Hard [53], μπορεί να αναχθεί στην ειδική περίπτωση της $k^{(m,n)}$-ανωνυμίας για δενδρικές εγγραφές. Ας υποθέσουμε ότι ο σχεσιακός πίνακας R απεικονίζεται ως μια συλλογή D από δενδρικές εγγραφές που έχουν όλες την ίδια δομή: μια ρίζα ανά εγγραφή και όλα τα πεδία του πίνακα ως άμεσα παιδιά της ρίζας. Η εύρεση της βέλτιστης $k^{(m,n)}$-ανωνυμίας για το D, με m, n ίσες με το πλήθος των διαστάσεων του R επιλύει επίσης το πρόβλημα της εύρεσης βέλτιστης λύσης για την k-ανωνυμία του R. Δεδομένου ότι το τελευταίο είναι ένα NP-Hard πρόβλημα, ομοίως είναι και το πρόβλημα της βέλτιστης $k^{(m,n)}$-ανωνυμοποίησης.

Λόγω της πολυπλοκότητας του προβλήματος, αποφεύγουμε την εξαντλητική αναζήτηση ολόκληρου του χώρου των λύσεων. Αντί αυτού προτείνουμε έναν αλγόριθμο που εξερευνά τους πιο υποσχόμενους υποχώρους. Η αναζήτηση γίνεται από πάνω προς τα κάτω (top-down) θεωρώντας αρχικά όλες τις τιμές γενικευμένες σε '*'. Ο αλγόριθμος διασχίζει την ιεραρχία γενίκευσης από πάνω προς τα κάτω εξειδικεύοντας ένα κόμβο σε κάθε βήμα. Κάθε εξειδίκευση δημιουργεί μια νέα οριζόντια τομή στην ιεραρχία. Αν το σύνολο των κανόνων γενίκευσης που υποδηλώνονται από μια τομή δεν δύναται να εγγυηθεί την $k^{(m,n)}$-ανωνυμία των δεδομένων, τότε εξερευνούμε τις πιθανές δομικές αποσυσχετίσεις που μπορούν να γίνουν στα δένδρα όταν οι τιμές τους είναι γενικευμένες σύμφωνα με αυτήν την τομή.

Ο έλεγχος αν ένα ζεύγος τομής γενικεύσεων και δομικών αποσυσχετίσεων (C, SD) μπορεί να ικανοποιήσουν την $k^{(m,n)}$-ανωνυμία αν εφαρμοστούν στο σύνολο δεδομένων D δεν είναι ένα απλοϊκή έργο. Για να γίνει ο έλεγχος αποτελεσματικά δημιουργήσαμε μια δενδρική δομή στη μνήμη την οποία καλούμε *δένδρο σύνοψης* το οποίο περιγράφουμε στην επόμενη ενότητα.

3.3.1 Δένδρο Σύνοψης

Χρησιμοποιούμε μια συμπιεσμένη δενδρική δομή για να κρατάμε το σύνολο των αρχικών δεδομένων στη μνήμη, την οποία αποκαλούμε *Δένδρο Σύνοψης*. Θεωρούμε ότι κάθε δενδρική εγγραφή προσδιορίζεται μοναδικά από ένα κωδικό αριθμό, ο οποίος συμβολίζει το άτομο του οποίου τα στοιχεία υπάρχουν στην εγγραφή. Οι ρίζες όλων των δενδρικών εγγραφών αντιστοιχίζονται σε ένα μοναδικό κόμβο, τη ρίζα του δένδρου σύνοψης. Όλα τα παιδιά της ρίζας κάθε εγγραφής προστίθενται ως παιδιά της ρίζας της σύνοψης. Κόμβοι οι οποίοι φέρουν α-



Σχήμα 3.11: Παράδειγμα βάσης δενδρικών δεδομένων.

κριβώς την ίδια τιμή συγχωνεύονται. Επειδή έχουμε κάνει την παραδοχή ότι δεν υπάρχουν διπλότυπα στα δεδομένα, οι κόμβοι αυτοί προέρχονται από διαφορετικές εγγραφές. Συνδέουμε κάθε κόμβο της σύνοψης με μια ανεστραμμένη λίστα η οποία περιέχει τους κωδικούς αριθμούς των δένδρων στα οποία ανήκει. Η ίδια διαδικασία εφαρμόζεται αναδρομικά και στα παιδιά των κόμβων αυτών για όλες τις εγγραφές, μέχρι και το επίπεδο των φύλλων. Αδελφοί κόμβοι με κοινή τιμή πάντα συγχωνεύονται και στην ανεστραμμένη λίστα του συγχωνευμένου κόμβου προστίθενται όλοι οι κωδικοί των δενδρικών εγγραφών στις οποίες εμφανίζονται.

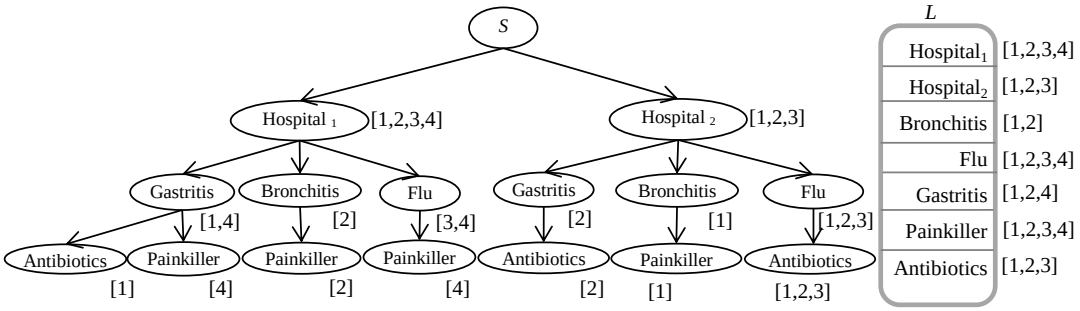
Το δένδρο σύνοψης διευκολύνει τον έλεγχο για την ικανοποίηση της $k^{(m,n)}$-ανωνυμίας από ένα σύνολο δεδομένων, διατηρώντας όχι μόνο την υποστήριξη συνδυασμών τιμών από το σύνολο $\mathcal{I}$, αλλά επίσης την υποστήριξη των μονοπατιών που τις περιέχουν. Ο όρος *υποστήριξη (support)* αναφέρεται στο πλήθος των εγγραφών που περιέχουν το μονοπάτι.

Ορισμός 3.3. *Υποστήριξη (support) ενός συνδυασμού γνώσης τιμών ή/και μονοπατιών είναι το πλήθος των εγγραφών του D στις οποίες εμφανίζεται ο συνδυασμός.*

Το δένδρο σύνοψης είναι μια παραλλαγή του trie tree, παρόμοια με το FP-tree [41] και αποτελείται από δυο βασικά μέρη:

Μια δενδρική δομή S, η οποία δημιουργείται με την υπέρθεση όλων των εγγραφών του συνόλου D. Η ρίζα κάθε εγγραφής απεικονίζεται στον κόμβο ρίζα r_s του δένδρου σύνοψης. Όλα τα μονοπάτια που εμφανίζονται σε μια εγγραφή τοποθετούνται στη σύνοψη ξεκινώντας από τον κόμβο r_s. Κάθε κόμβος n της σύνοψης έχει δύο στοιχεία: (α) μια ετικέτα με την κοινή τιμή των κόμβων των εγγραφών που απεικονίζονται πάνω του και (β) μια ταξινομημένη λίστα των κωδικών των εγγραφών που περιέχουν το ακριβές μονοπάτι από την ρίζα ως τον συγκεκριμένο κόμβο $r_s \rightarrow \cdots \rightarrow n$.

Ένα βοηθητικό πίνακα L ο οποίος περιέχει ένα στοιχείο για κάθε τιμή i από το πεδίο τιμών των δεδομένων $\mathcal{I}$. Κάθε στοιχείο του πίνακα περιέχει τρία αντικείμενα: (α) μια ετικέτα με την τιμή i η οποία προσδιορίζει το στοιχείο, (β) μια λίστα κωδικών με τα ids όλων των εγγραφών που περιέχουν το i τουλάχιστον μια φορά, και (γ) ένα σύνδεσμο (link) προς κάθε κόμβο του δένδρου S που φέρει την ίδια ετικέτα i. Οι σύνδεσμοι αυτοί απεικονίζονται



Σχήμα 3.12: Δένδρο Σύνοψης του παραδείγματος του Σχήματος 3.11.

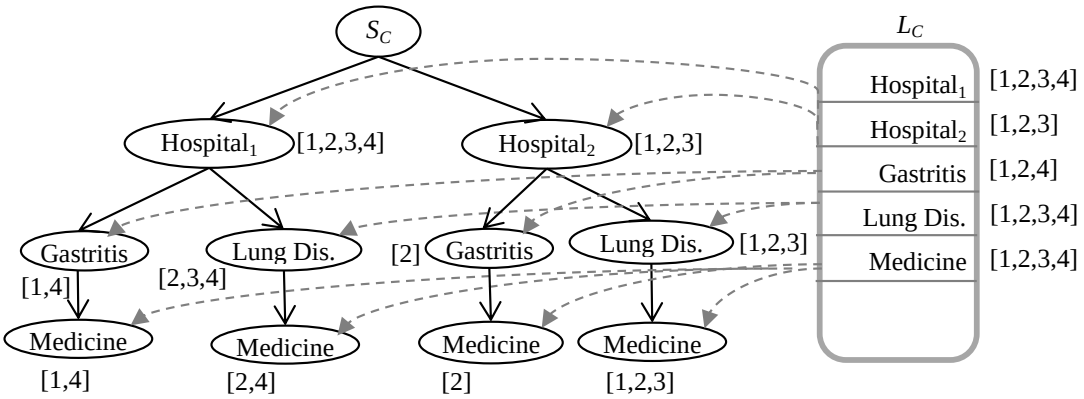
με διακεκομμένα βέλη στο Σχήμα 3.13 και θα αναφέρονται ως *πλάγιοι σύνδεσμοι (sidelinks)* στο υπόλοιπο κείμενο του κεφαλαίου. Ο πίνακας δεν είναι ταξινομημένος, η σειρά των στοιχείων δεν έχει σημασία και εξαρτάται από την σειρά εισαγωγής τους κατά την δημιουργία της σύνοψης. Ο πίνακας L αφενός παρέχει την δυνατότητα για την οριζόντια πρόσβαση στο δένδρο S και αφετέρου διευκολύνει τον έλεγχο για την ικανοποίηση της k^m-ανωνυμίας, που είναι προαπαιτούμενο για την ικανοποίηση της $k^{(m,n)}$-ανωνυμίας, χωρίς να είναι απαραίτητη η διάσχιση του δένδρου S . Σημειώνεται ότι η διατήρηση των λιστών των κωδικών (ids) των εγγραφών που σχετίζονται με κάθε στοιχείο i του πίνακα L είναι πλεονασμός. Η λίστα αυτή θα μπορούσε να ανακτηθεί από την συγχώνευση των λιστών όλων των κόμβων του δένδρου S που φέρουν την ετικέτα i. Εντούτοις, επιλέγουμε να διατηρούμε τις πλεονάζουσες λίστες του L για να βελτιώσουμε την απόδοση του αλγόριθμου, διότι ο έλεγχος της υποστήριξης συνδυασμών τιμών είναι πολύ συχνός κατά την ανωνυμοποίηση.

Παράδειγμα 3.6. *Έστω το σύνολο δενδρικών δεδομένων του Σχήματος 3.11 το οποίο περιέχει τέσσερεις εγγραφές. Το αντίστοιχο δένδρο σύνοψης που προκύπτει απεικονίζεται στο σχήμα 3.12. Ο κόμβος «Hospital$_2$» υπάρχει στις εγγραφές 1, 2 και 3 ως παιδί της ρίζας τους. Έτσι, στο δένδρο σύνοψης υπάρχει μόνο ένα παιδί της ρίζας με τιμή «Hospital$_2$» με λίστα κωδικών την [1, 2, 3] όπως φαίνεται στο σχήμα.*

3.3.2 Δένδρο Προβολής

Το δένδρο σύνοψης που ορίσαμε εμπεριέχει ολόκληρη την πληροφορία των αρχικών δεδομένων σε συμπιεσμένη μορφή. Η αρχική βάση δεδομένων μπορεί να ανακατασκευαστεί πλήρως από το δένδρο σύνοψης. Επιπλέον, το δένδρο επαρκεί για τον υπολογισμό της υποστήριξης των συνδυασμών αρχικών τιμών και μονοπατιών στα δεδομένα. Κατά την διαδικασία της ανωνυμοποίησης χρειάζεται ένα δένδρο σύνοψης για κάθε προβολή του συνόλου των δεδομένων D σε μια τομή γενίκευσης C. Η χρονοβόρα διαδικασία της προβολής όλων των εγγραφών του D στην τομή C μπορεί να παρακαμφθεί, καθώς μπορούμε να προβάλουμε απευθείας το δένδρο σύνοψης στην τομή C και να προκύψει το ίδιο αποτέλεσμα. Έτσι μπορούμε να δημιουργήσουμε με ισοδύναμο τρόπο το αντίστοιχο *προβεβλημένο δένδρο σύνοψης* S_C το ποίο για συντομία καλούμε *δένδρο προβολής*. Η διαδικασία της προβολής περιλαμβάνει τα παρακάτω:

- Ένα νέο στοιχείο προστίθεται στον πίνακα L για κάθε γενικευμένη τιμή gi που εμ-



Σχήμα 3.13: Δένδρο προβολής στην τομή $c_i = \{A, b_1, b_2\}$.

φανίζεται στην τομή C. Το νέο στοιχείο διαθέτει μια λίστα κωδικών εγγραφών που προκύπτει από το αποτέλεσμα της *ένωση* όλων των λιστών κωδικών από κάθε στοιχείο i που η ετικέτα του γενικεύεται σε gi. Δημιουργούνται οι πλάγιοι σύνδεσμοι του gi, ως το σύνολο των πλαγίων συνδέσμων που υπάρχουν σε κάθε στοιχείο i που η ετικέτα του γενικεύεται σε gi.

- Η ετικέτα i κάθε κόμβου του δένδρου S αντικαθίσταται από την γενίκευσή της gi που καθορίζεται από την τομή C.
- Αδέλφια κόμβοι που φέρουν την ίδια ετικέτα gi συγχωνεύονται. Ο νέος συγχωνευμένος κόμβος έχει την ίδια ετικέτα gi, ενώ η λίστα κωδικών του είναι η *ένωση* όλων των λιστών κωδικών των αρχικών κόμβων. Οι πλεονάζοντες πλάγιοι σύνδεσμοι από το στοιχείο gi του πίνακα L προς τον ίδιο συγχωνευμένο κόμβο του δένδρου απαλείφονται.

Παράδειγμα 3.7. *Θεωρούμε το δένδρο σύνοψης του Σχήματος 3.12 και την τομή γενίκευσης C=\{Lung disease, Gastritis, Diarrhea, Neurological, Orthopedic, Medicine, Hospital$_1$, Hospital$_2$, Rea, Gaia\} η οποία υποδηλώνει τους κανόνες γενίκευσης \{Flu, Bronchitis\}→Lung disease και \{Antibiotics, Painkiller\}→Medicine. Το αντίστοιχο δένδρο προβολής S_C που προκύπτει από την προβολή της C στη σύνοψη και η λίστα πλαγίων συνδέσμων L_C απεικονίζονται στο Σχήμα 3.13. Οι κόμβοι «Flu» και «Bronchitis» ήταν αδέλφια κάτω από το «Hospital$_1$» και συγχωνεύθηκαν μετά από την γενίκευσή τους σε «Lung disease» στο δένδρο προβολής. Στο νέο κόμβο που προέκυψε η λίστα των κωδικών εγγραφών [2, 3, 4] είναι η ένωση των αντίστοιχων λιστών των κόμβων «Flu» και «Bronchitis» που βρίσκονταν κάτω από το «Hospital$_1$» στη σύνοψη.*

Το RPD ως προσεγγιστική μετρική. Οι μετρικές της απώλειας πληροφορίας που ορίστηκαν στην Ενότητα 3.2.5 χρησιμοποιούνται για την αποτίμηση της ποιότητας των τελικών δεδομένων και υπολογίζονται πάνω στις ανωνυμοποιημένες εγγραφές. Το συνολικό RPD είναι ο μέσος όρος του RPD κάθε εγγραφής, τα ML^2 και dML^2 απαιτούν εξόρυξη δεδομένων από το ανωνυμοποιημένο σύνολο D'. Ο υπολογισμός τους είναι υπολογιστικά ακριβός και η χρήση τους για την αξιολόγηση κάθε υποψήφιας λύσης δεν είναι πρακτική.

Αλγόριθμος 1 ValueCheck()

Require: C,L,m

Ensure: **true**, **false** {**true** if D_C is k^m-anonymous, else **false**}

```
1: for all cm combinations of size m in C do
2:    Intersect the lists from L for every item of cm
3:    if the intersection size is between k and 0 then
4:       return false
5: return true
```

Αντίθετα, ο αλγόριθμος που προτείνουμε χρησιμοποιεί μια υπολογιστικά φθηνή μετρική, η οποία βασίζεται στο RPD αλλά υπολογίζεται πάνω στο προβεβλημένο δένδρο σύνοψης S_C. Για καλύτερη προσέγγιση, μετά από πειραματικές δοκιμές, λαμβάνουμε υπόψη την υποστήριξη (support) κάθε κόμβου καθώς επίσης και το συνολικό πλήθος των κόμβων (nds_{S_C}) που υπάρχουν στο δένδρο S_C. Το προσεγγιστικό RPD_a για ένα μονοπάτι p από την ρίζα του δένδρου προβολής S_C προς ένα φύλλο υπολογίζεται από την ακόλουθη συνάρτηση:

$$RPD_a(p) = \frac{(sup(u_1) + \cdots + sup(u_n))}{d(u_1) \times |C(u_1)| \times \cdots \times d(u_n) \times |C(u_n)|} \tag{3.6}$$

όπου $sup(u_i)$ είναι η υποστήριξη του κόμβου u_i. Ο τύπος της RPD_a για το συνολικό δένδρο προβολής S_C δίνεται από την Εξίσωση 3.7:

$$RPD_a(S_C) = \frac{1}{nds_{S_C}} \times \frac{1}{lvs_{S_C}} \sum_{p \in S_C} RPD_a(p) \tag{3.7}$$

όπου nds_{S_C} είναι ο συνολικός αριθμός κόμβων του S_C.

3.3.3 Έλεγχος Υποψηφίων λύσεων

Το δένδρο προβεβλημένης σύνοψης S_C και ο πίνακας πλάγιων συνδέσμων L ενός συνόλου δεδομένων D, μπορούν να χρησιμοποιηθούν για να επαληθεύσουμε γρήγορα αν μια υποψήφια λύση (C, SD), η οποία αποτελείται από μια τομή γενίκευσης C και ένα σύνολο κανόνων δομικής αποσυσχέτισης SD, επαρκούν για την ικανοποίηση της $k^{(m,n)}$-ανωνυμίας όταν εφαρμόζονται στο D. Η διαδικασία αυτή εκτελείται σε δύο φάσεις: τον *έλεγχο γενίκευσης τιμών* και τον *έλεγχο δομικών σχέσεων*. Η πρώτη φάση εξετάζει αν όλοι οι πιθανοί συνδυασμοί m τιμών από το D_C εμφανίζονται σε τουλάχιστον k εγγραφές. Ως D_C συμβολίζουμε τα δεδομένα του D προβεβλημένα στους κανόνες γενίκευσης της τομής C. Η δεύτερη φάση εξετάζει αν υπάρχουν τουλάχιστον k εγγραφές του D, οι οποίες περιέχουν συνδυασμούς τους μεγέθους m, ενώ επίσης λαμβάνουν υπόψη οποιοδήποτε συνδυασμό n δομικών σχέσεων μεταξύ τους. Ο ψευδοκώδικας των συναρτήσεων `ValueCheck` και `StructureCheck` παρουσιάζονται στους Αλγόριθμους 1 και 2 αντιστοίχως.

Η συνάρτηση `ValueCheck` εξετάζει αν υπάρχει συνδυασμός τιμών, μεγέθους μικρότερου ή ίσου του m, ο οποίος δεν εμφανίζεται σε τουλάχιστον k εγγραφές του D_C. Σε αυτήν την

Αλγόριθμος 2 StructureCheck()

Require: S_c, L, (cmn, cnr) {cmn items, and cnr relations between them}
Ensure: $true, false$ {**true** if D_C contains (cmn,cnr) k times, else **false**}
1: $clist$ = all ids {first intersection will initialize it}
2: **for all** $\{an \rightsquigarrow dn\}$ relations of cnr **do**
3: $tlist = \emptyset$
4: **for all** nodes dn in the tree **do**
5: **if** the path from dn to the $root$ contains an **then**
6: $tlist = dn.list \cup tlist$ {all trees that adhere to $an \rightsquigarrow dn$}
7: remove $\{an \rightsquigarrow dn\}$ from cnr
8: $clist = clist \cap tlist$;
9: **if** $clist$ has 0 items **then**
10: return **true**
11: **else if** $clist$ has less than k items **then**
12: return **false**
13: return **true**

περίπτωση το C, SD δεν μπορεί να θεωρείται έγκυρη υποψήφια λύση καθώς η τομή C αδυνατεί να εξασφαλίσει την απαραίτητη ελάχιστη υποστήριξη ($support \geq k$) για κάθε συνδυασμό τιμών. Αν μια λύση απορριφθεί κατά τον έλεγχο γενίκευσης τιμών, δεν χρειάζεται να προχωρήσουμε στην δημιουργία του δένδρου S_C, η οποία είναι ακριβή. Η φάση ελέγχου γενίκευσης μπορεί να γίνει χρησιμοποιώντας μόνο τον πίνακα L. Μπορούμε εύκολα να συμπεράνουμε ότι η `ἄλυεῆεςκ` εγγυάται την ακόλουθη Ιδιότητα:

Ιδιότητα 3.1. *Η συνάρτηση* `ValueCheck` *επιστρέφει* `true` *(αληθής) αν και μόνον αν το σύνολο δεδομένων D_C είναι k^m-ανώνυμο, δηλαδή κάθε πιθανός συνδυασμός m τιμών εμφανίζεται σε τουλάχιστον k εγγραφές.*

Αν ο έλεγχος γενίκευσης τιμών είναι επιτυχής, το δένδρο προβολής S_C δημιουργείται ώστε να χρησιμοποιηθεί από την `StructureCheck` για τον έλεγχο αν το σύνολο D_C υποστηρίζει τουλάχιστον k φορές ένα συνδυασμό τιμών cmn με cnr σχέσεις μεταξύ τους. Η συνάρτηση `StructureCheck` λαμβάνει ως είσοδο το δένδρο προβολής S_C, δηλαδή την σύνοψη προβεβλημένη στην τομή C, τον πίνακα πλαγίων συνδέσμων L, ένα συνδυασμό τιμών cmn, και ένα σύνολο δομικών σχέσεων cnr που ισχύει μεταξύ των cmn τιμών. Στις γραμμές 4-6 ο αλγόριθμος συλλέγει την λίστα ($tlist$) των εγγραφών που υποστηρίζουν την σχέση $an \rightsquigarrow dn$ στο D_C και στη συνέχεια υπολογίζει την *τομή* της λίστας αυτής με την λίστα ($clist$) των εγγραφών που υποστηρίζουν τις υπόλοιπες σχέσεις του cnr στην Γραμμή 8. Αν σε κάποιο σημείο η $clist$ περιέχει πάνω από 0 και λιγότερες από k εγγραφές τότε ο αλγόριθμος τερματίζει καθώς η λύση παραβιάζει την $k^{(m,n)}$-ανωνυμία.

Ιδιότητα 3.2. *Η συνάρτηση* `StructureCheck` *επιστρέφει* `true` *(αληθής) αν και μόνον αν οι cmn τιμές με τις cnr σχεσεις μεταξύ τους, εμφανίζονται σε τουλάχιστον k εγγραφές του D_C.*

Αλγόριθμος 3 FixStructure()

Require: S_C, L, C
Ensure: SD {disassociation rules}
1: $SD = \emptyset$
2: **for all** cmn combinations of m items from C **do**
3: **for** $i = 1 \dots n$ **do**
4: **for all** $cnr_i \backslash SD$ combinations of size i of the items of cmn **do**
5: **while** not $StructureCheck(S_C, L, cmn, cnr_i)$ **do**
6: select the relation $a \rightsquigarrow b$ from cnr_i with the least positive support
7: **for all** paths that contain $a \rightsquigarrow b$ **do**
8: move the children of b to become its siblings
9: move b to become a sibling of a
10: $SD = SD \cup r$ {add r to existing disassociation rules}
11: return SD

Οι συναρτήσεις `ValueCheck` και `StructureCheck` δεν είναι συμμετρικές. Η `ValueCheck` ελέγχει αν μια τομή C παρέχει k^m-ανωνυμία στο D_C. Για να το επιτύχει πρέπει να ελέγξει όλους τους συνδυασμούς m τιμών που εμφανίζονται στην C. Αντίθετα, η `StructureCheck` ελέγχει μόνο ένα συνδυασμό τιμών cmn και ένα σύνολο δομικών σχέσεων μεταξύ τους cnr. Ο λόγος για αυτή την επιλογή θα γίνει φανερός με την περιγραφή του αλγορίθμου στην επόμενη ενότητα.

3.3.4 Αλγόριθμος Πλήρους Αναζήτησης Τομών (ACS)

Ο αλγόριθμος που προτείνουμε εξετάζει όλους τους πιθανούς μετασχηματισμούς που ανωνυμοποιούν τα δεδομένα και ελέγχει αν ικανοποιούν την εγγύηση της $k^{(m,n)}$-ανωνυμίας. Από αυτούς επιλέγει τη λύση η οποία εισάγει τη μικρότερη απώλεια πληροφορίας στα δεδομένα. Ο αλγόριθμος που αναπτύξαμε είναι top-down και εξερευνά τον χώρο των λύσεων ξεκινώντας από την κατάσταση στην οποία όλες οι τιμές είναι γενικευμένες στην ρίζα της ιεραρχίας ($C = \{*\}$) και καμία δομική αποσυσχέτιση δεν έχει πραγματοποιηθεί ($SD = \emptyset$). Στη συνέχεια προχωρά ελέγχοντας λιγότερο γενικές τομές και τις αντίστοιχες δομικές αποσυσχετίσεις ώστε να ικανοποιείται η $k^{(m,n)}$-ανωνυμία.

Ο πλήρης χώρος των λύσεων του προβλήματος αποτελείται από όλες τις πιθανές τομές γενίκευσης και όλους τους πιθανούς κανόνες δομικής αποσυσχέτισης για καθεμία από αυτές. Ο εξαντλητικός έλεγχος όλων των περιπτώσεων δεν είναι πρακτικός ακόμη και για δεδομένα μικρού μεγέθους. Εναλλακτικά, προτείνουμε έναν ευρηστικό αλγόριθμο, τον οποίο ονομάζουμε `AllCutSearch (ACS)`. Ο `ACS` εξετάζει τους κανόνες γενίκευσης και δομικής αποσυσχέτισης με ασύμμετρο τρόπο. Ελέγχει εξαντλητικά κάθε πιθανή τομή γενίκευσης, αλλά αποφασίζει ποιές δομικές αποσυσχετίσεις να εφαρμόσει με άπληστο τρόπο. Επιλέξαμε αυτή την στρατηγική λόγω του αντίστοιχου ασύμμετρου υπολογιστικού κόστους που απαιτούν οι έλεγχοι εξέτασης όλων των γενικεύσεων και όλων των δομικών αποσυσχετίσεων. Το δεύτερο εί-

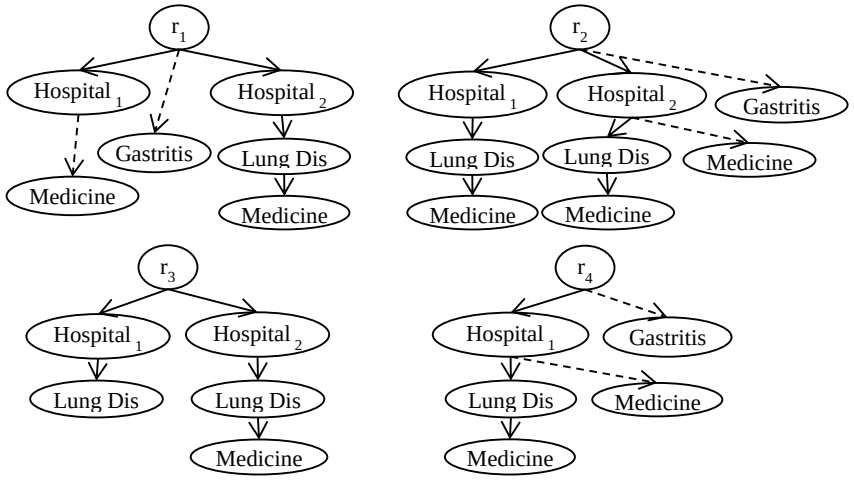
Αλγόριθμος 4 AllCutSearch (ACS)

Require: D, $DGHierarchy$, k, n, m
Ensure: (C, SD) $\{(C, SD)$ renders D $k^{(m,n)}$-anonymous$\}$
1: Create synopsis tree S from D
2: Create inverted list L $\{L$ is created for generalized terms too$\}$
3: stack $STK = \emptyset$
4: $bestCost = \infty$; {Minimum Loss}
5: $root = \star$
6: mark $root$ as closed
7: STK.push($root$)
8: **while** STK not empty **do**
9: $cCut$=STK.pop() {current cut $cCut$}
10: **for all** children C of $cCut$ **do**
11: **if** C not closed **then**
12: mark C as closed
13: STK.push(C)
14: **if** $ValueCheck(cCut, S, L)$ **then**
15: Create S_{cCut} {we project S to $cCut$}
16: $cSD = FixStructure(S_{cCut}, L, cCut)$
17: $cCost = cost(cCut, cSD)$ {estimated cost of current solution}
18: **if** $cCost < bestCost$ **then**
19: $bestCost = cCost$
20: $(C, SD) = (cCUT, cSD)$
21: **return** (C, SD);

ναι σημαντικά ακριβότερο του πρώτου για τις περισσότερες ρεαλιστικές περιπτώσεις συνόλων δεδομένων.

Για να γίνει αντιληπτό με ποιά σειρά ο ACS εξετάζει τις υποψήφιες τομές γενίκευσης, ας παρατηρήσουμε τον γράφο υποψηφίων τομών του Σχήματος 3.14. Οι κόμβοι του γράφου απεικονίζουν όλες τις υποψήφιες τομές γενίκευσης που προκύπτουν από την ιεραρχία του Σχήματος 3.5. Οι ακμές του γράφου δείχνουν με ποιόν τρόπο μια τομή C μπορεί να εξειδικευθεί προς μια άλλη C', γενικεύοντας μόνο μια τιμή από την C. Ο ACS ξεκινά τον έλεγχο από την πιο γενική τομή $(*)$ και επισκέπτεται όλους τους κόμβους-τομές του γράφου ακριβώς μία φορά. Ο ψευδοκώδικας του ACS παρουσιάζεται στον Αλγόριθμο 4.

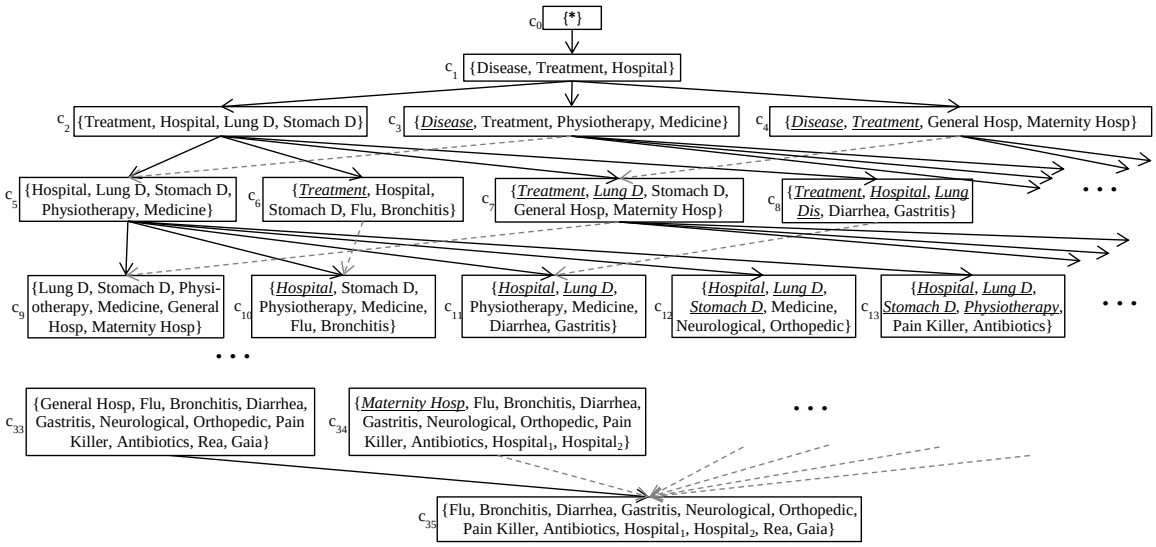
Ο αλγόριθμος χρησιμοποιεί μια στοίβα STK για να κρατάει όλους τους γειτονικούς κόμβους του γράφου υποψηφίων τομών που δεν έχουν ελεγχθεί ακόμα. Σε κάθε βήμα αφαιρείται ο πρώτος κόμβος από την στοίβα και ο αλγόριθμος εξετάζει αν η αντίστοιχη τομή γενίκευσης $cCut$ μπορει να ικανοποιήσει την $k^{(m,n)}$-ανωνυμία. Αρχικά εξετάζεται αν ισχύει η απλή k^m-ανωνυμία, καλώντας την συνάρτηση ValueCheck. Αν ο έλεγχος της ValueCheck αποτύχει, τότε η $k^{(m,n)}$-ανωνυμία δεν μπορεί να επιτευχθεί για αυτή την τομή, ούτε για *οποιαδήποτε τομή πιο εξειδικευμένη από την τρέχουσα*. Έτσι ο αλγόριθμος συνεχίζει τον έλεγχο με το επόμενο



Σχήμα 3.13: $3^{(2,1)}$-ανωνυμοποίση των δεδομένων του Σχήματος 3.11.

στοιχείο της STK, δηλαδή τον επόμενο αδελφό κόμβο του $cCut$. Αντίθετα, αν ο έλεγχος της `ValueCheck` είναι επιτυχής, τότε είναι βέβαιο ότι η τρέχουσα τομή μπορεί να ικανοποιήσει την $k^{(m,n)}$-ανωνυμία, εφαρμόζοντας όσες δομικές αποσυσχετίσεις χρειαστεί. Σε αυτό το σημείο ο αλγόριθμος δημιουργεί το δένδρο S_C και καλεί την συνάρτηση `FixStructure`, η οποία εφαρμόζει τις απαιτούμενες δομικές αποσυσχετίσεις cSD και ο ψευδοκώδικάς της φαίνεται στον Αλγόριθμο 3. Αν το κόστος σε απώλεια πληροφορίας (RPD_a) της υποψήφιας λύσης $(cCut, cSD)$ είναι μικρότερο του κόστους της τρέχουσας βέλτιστης λύσης που έχει βρεθεί ως τώρα, τότε το $(cCut, cSD)$ αποθηκεύεται ως η τρέχουσα βέλτιστη λύση. Ο αλγόριθμος τότε εισάγει όλα τα παιδιά (εξειδικεύσεις) της τομής $cCut$ στην στοίβα STK.Όταν δεν έχουν απομείνει άλλοι κόμβοι στη στοίβα STK ο αλγόριθμος τερματίζει και επιστρέφει την τρέχουσα λύση (C, SD) ως την καλύτερη λύση. Η απώλεια πληροφορίας του αποτελέσματος κάθε λύσης υπολογίζεται χρησιμοποιώντας την μετρική RPD που εισάγαμε στην Ενότητα 3.2.5.

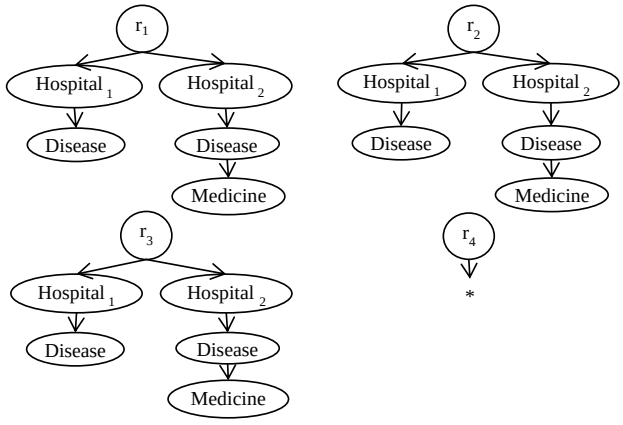
Παράδειγμα 3.8. *Έστω η ιεραρχία γενίκευσης τιμών του Σχήματος 3.5. Ο* `ACS` *πρώτα θα γενικεύσει όλες τις τιμές στην τομή {*}, όπως φαίνεται στον κόμβο c_0 του Σχήματος 3.14. Στη συνέχεια ο* `ACS` *προχωρά στην επόμενη τομή c_1 = {Disease, Treatment, Hospital}. Στο επόμενο βήμα υπάρχουν τρεις πιθανές τιμές που μπορούν να εξειδικευτούν και οι εξειδικεύσεις αυτές αντιστοιχούν στα τρία παιδιά του c_1. Μετά τον έλεγχο για την $k^{(m,n)}$-ανωνυμία, αυτοί οι κόμβοι ταξινομούνται κατά σειρά κόστους (απώλειας πληροφορίας) από το χαμηλότερο προς το υψηλότερο. Έστω ότι η σειρά που προέκυψε είναι c_2, c_3, c_4 όπως φαίνεται στο Σχήμα 3.14. Η τομή c_2 εξειδικεύεται πρώτη. Η εξειδίκευσή της δημιουργεί άλλες τέσσερεις υποψήφιες τομές οι οποίες ελέγχονται και ταξινομούνται κατά αυξανόμενο κόστος. Αν έστω μία από αυτές ικανοποιεί την $k^{(m,n)}$-ανωνυμία και έχει μικρότερο κόστος από την c_2, τότε ο* `ACS` *προχωρεί στην εξειδίκευσή της, και ούτω καθεξής. Διαφορετικά,* επιστρέφουμε *(rollback) στην τομή c_3, η οποία έχει μόνο τρεις νέες πιθανές εξειδικεύσεις καθώς η τιμή «Disease» είναι τώρα κλειστή. Αν καμία από τις τομές c_2, c_3 και c_4 δεν μπορεί να ικανοποιήσει την $k^{(m,n)}$-ανωνυμία, τότε ο* `ACS` *θα επιστρέψει στην τομή c_0 και θα τερματίσει.*



Σχήμα 3.14: Πλήρης και ελάχιστος γράφος απαρίθμησης υποψηφίων τομών.

Ο ψευδοκώδικας της συνάρτησης `FixStructure` παρουσιάζεται στον Αλγόριθμο 3. Για κάθε συνδυασμό τιμών cmn από την τρέχουσα τομή γενίκευσης, η `FixStructure` εξετάζει όλους τους πιθανούς συνδυασμούς cnr μεγέθους από 1 έως n σχέσεις και υπολογίζει την υποστήριξη του συνδυασμού των cmn τιμών με τις cnr σχέσεις χρησιμοποιώντας το δένδρο προβεβλημένης σύνοψης S_C και τους πλάγιους συνδέσμους του πίνακα L. Αν ένας συνδυασμός cnr σχέσεων προκαλεί την παραβίαση της $k^{(m,n)}$-ανωνυμίας, η συνάρτηση `FixStructure` θα αποσυσχετίσει δομικά κάποιες σχέσεις από το σύνολο cnr, ξεκινώντας από την λιγότερο συχνή, και θα προσθέσει τον αντίστοιχο κανόνα γενίκευσης στο σύνολο SD, έως ότου ο συνδυασμός cmn τιμών και $cnr \setminus SD$ σχέσεων να εμφανίζονται σε τουλάχιστον k εγγραφές. Δεδομένου ότι όλοι οι πιθανοί συνδυασμοί m τιμών έχουν ελεγχθεί και ότι όλοι οι συνδυασμοί των μεταξύ τους σχέσεων μεγέθους n έχουν ελεγχθεί, η τελική λύση (C, SD) θα εγγυάται την $k^{(m,n)}$-ανωνυμία για το σύνολο D.

Αξίζει να σημειωθεί ότι στην υλοποίηση του αλγόριθμου δεν κρατάμε τον γράφο υποψηφίων τομών στη μνήμη, ούτε δηλώνουμε άμεσα ποιες εξειδικεύσεις είναι επιτρεπτές από κάθε κόμβο. Αντίθετα, σημειώνουμε ποιές τιμές είναι κλειστές, διότι είναι λιγότερες και η αναπαράστασή τους είναι ευκολότερη. Όταν μια τιμή εξειδικευθεί, δηλαδή όταν κάνουμε έλεγχο σε τομές που περιέχουν τα παιδιά της από το δένδρο ιεραρχίας γενίκευσης, τότε η τιμή δηλώνεται ως κλειστή στις επόμενες τομές που την περιέχουν. Μια τομή θεωρείται κλειστή όταν κάθε τιμή τις είναι είτε κλειστή είτε φύλλο της ιεραρχίας γενίκευσης και δεν μπορεί να εξειδικευθεί περεταίρω. Στο Σχήμα 3.14, τα βέλη με συνεχή γραμμή αναπαριστούν τις επιτρεπόμενες εξειδικεύσεις τομών. Αντίθετα, τα βέλη που αναπαριστάνονται με διακεκομμένη γραμμή θα οδηγούσαν σε διπλό έλεγχο της ίδιας τομής και για αυτό τον λόγο δεν ακολουθούνται. Οι τιμές που είναι υπογραμμισμένες στους κόμβους του δένδρου θεωρούνται *κλειστές* και δεν εξειδικεύονται περεταίρω διότι θα οδηγούσαν σε διπλούς ελέγχους τομών, όπως φαίνεται από τα διακεκομμένα βέλη.



Σχήμα 3.15: ΠολυΣχεσιακή 3-ανωνυμία των δεδομένων του Σχήματος 3.11.

Παράδειγμα 3.9. *Οι δενδρικές εγγραφές του Σχήματος 3.13 αποτελούν την* $3^{(2,1)}$*-ανώνυμη εκδοχή του αρχικού συνόλου δεδομένων του Σχήματος 3.11. Η λύση αποτελείται από την τομή γενίκευσης* $C = \{Lung\ Disease,\ Gastritis,\ Diarrhea,\ Medicine,\ Hospital_1,\ Hospital_2,\ Rea,\ Gaia\}$ *και τους κανόνες δομικής αποσυσχέτισης* $SD = \{Gastritis \rightsquigarrow Medicine\}$*. Η πολυΣχεσιακή 3-ανωνυμία [56] θα είχε προκαλέσει τρία από τα τέσσερα δένδρα να γίνουν πανομοιότυπα. Καθένα θα διατηρούσε μόνο δύο μονοπάτια:* $r \rightarrow Hospital_2 \rightarrow Flu \rightarrow Pain\ killer$ *και* $r \rightarrow Hospital_1 \rightarrow Disease$*. Οι υπόλοιποι κόμβοι θα είχαν απαλοιφθεί. Επιπλέον θα είχε διαγράψει εντελώς την τελευταία δεδνρική εγγραφή, όπως φαίνεται στο Σχήμα 3.15.*

3.3.5 Αλγόριθμος Άπληστης Αναζήτησης Τομών (GCS)

Ο αλγόριθμος *AllCutSearch* (ACS) αποφεύγει την εξαντλητική αναζήτηση ολόκληρου του χώρου των λύσεων, αλλά παραμένει σχετικά ακριβός αν το μέγεθος των δεδομένων ή το πεδίο των τιμών είναι μεγάλα και η ιεραρχία γενίκευσης έχει πολλά επίπεδα. Για την ανωνυμοποίηση μεγαλύτερων και πιο εκφραστικών δεδομένων, προτείνουμε τον αλγόριθμο *GreedyCutSearch* (GCS), ο οποίος εκτελεί μια (partial best-first) διάσχιση του γράφου των υποψήφιων τομών γενίκευσης. Εκμεταλλευόμενοι την επιτόπου ταξινόμηση των υποψηφίων λύσεων ως προς το κόστος, δηλαδή την απώλεια πληροφορίας που εισάγουν, μπορούμε να αποφύγουμε τον πλήρη έλεγχο σε όλο το χώρο λύσεων επιτυγχάνοντας μια καλή προσέγγιση του αποτελέσματος. Η λογική είναι ότι στα πρώτα μονοπάτια του δένδρου απαρίθμησης υποψηφίων τομών (βλ. Σχήμα 3.14) είναι πιο πιθανό να βρίσκεται η βέλτιστη ή μια σχεδόν βέλτιστη λύση. Ο GCS κάνει τις ίδιες λειτουργίες με τον ACS, αλλά αντί για τον έλεγχο όλων των παιδιών της τρέχουσας τομής γενίκευσης $cCut$, εξετάζει μόνο τα g παιδιά με το χαμηλότερο κόστος σε απώλεια πληροφορίας. Η αντίστοιχη STK του GCS είναι μια ουρά προτεραιότητας όπου οι τομές με την χαμηλότερη RPD_a εξετάζονται πρώτες. Σε κάθε βήμα του αλγορίθμου, ο GCS εξάγει από την ουρά όλες τις τομές που είναι αδέλφια της πρώτης τομής στην STK, όπως φαίνεται στην Γραμμή 9 του Αλγορίθμου 4, αλλά δεν εισάγει αμέσως κανένα από τα παιδιά τους στην ουρά STL. Πρώτα εξετάζει κάθε τομή, τις ταξινομεί κατά αυξανόμενο κόστος και εισάγει μόνο

τα παιδιά των g καλύτερων τομών, δηλαδή αυτών με το χαμηλότερο κόστος. Με αυτόν τον τρόπο ο αλγόριθμος επιλέγει με άπληστο τρόπο τα πιο υποσχόμενα μονοπάτια του γράφου υποψηφίων τομών και μπορεί να περιορίσει αισθητά των χώρο αναζήτησης των λύσεων και να μειώσει το υπολογιστικό κόστος. Τα πειραματικά αποτελέσματα της Ενότητας 3.4 δείχνουν ότι ακόμα και για μικρές τιμές της παραμέτρου g, οι λύσεις που επιλέγονται είναι σχεδόν ισοδύναμες, από πλευράς ποιότητας των αποτελεσμάτων, με εκείνες του `ACS`.

Παράδειγμα 3.10. *Επιστρέφοντας στο παράδειγμα του Σχήματος 3.14 και υποθέτοντας ότι $g = 2$, ο αλγόριθμος `GCS` θα γενικεύσει πρώτα όλες τις τιμές στην τομή c_0 και θα συνεχίσει εξετάζοντας την c_1. Στο επόμενο βήμα, ο `GCS` θα ελέγξει τις τρεις νέες υποψήφιες τομές για την ικανοποίηση της $k^{(m,n)}$-ανωνυμίας και θα τις ταξινομήσει από την μικρότερη προς την μεγαλύτερη τιμή της RPD_a. Έστω ότι η σειρά είναι c_2, c_3, c_4. Ο `GCS` θα προσθέσει τις τομές c_2 και c_3 στην ουρά προτεραιότητας, αλλά όχι την c_4 επειδή έχει μεγαλύτερο κόστος από τις άλλες δυο και $g = 2$. Η εξειδίκευση των τομών c_2 και c_3 δημιουργεί 4+3=7 νέες τομές, καθώς η τιμή «Disease» θεωρείται τώρα κλειστή για την c_3. Ο `GCS` θα προσθέσει στην ουρά προτεραιοτήτων μόνο τις 2 τομές με το μικρότερο κόστος RPD_a, κ.ο.κ. Όταν η ουρά προτεραιοτήτων αδειάσει, ο `GCS` τερματίζει.*

3.3.6 Ανάλυση Πολυπλοκότητας

Η συνάρτηση `ValueCheck` υπολογίζει όλους τους συνδυασμούς m τιμών από το σύνολο τιμών μιας τομής γενίκευσης μεγέθους $|C|$. Το πλήθος των συνδυασμών είναι $\binom{|C|}{m} = \frac{|C|!}{m!(|C|-m)!} = O(\frac{|C|^m}{m!})$. Η μέγιστη τιμή που μπορεί να πάρει το $|C|$ είναι το μέγεθος του αρχικού πεδίου τιμών, $\mathcal{I}$. Οι m τιμές ενός συνδυασμού θα αναζητηθούν στον πίνακα L σε χρόνο $m \cdot log|L| = O(m \cdot log\mathcal{I})$, δεδομένου ότι το μέγεθός του είναι $O(\mathcal{I})$, Συνεπώς, η `ValueCheck` είναι $O(\frac{\mathcal{I}^m}{(m-1)!} \cdot log\mathcal{I})$.

Η συνάρτηση `StructureCheck` ελέγχει το πολύ n ζεύγη προγόνου-απογόνου. Πρώτα αναζητά τους n απογόνους από τον πίνακα L σε χρόνο $O(n \cdot log\mathcal{I})$. Στη συνέχεια ακλουθεί τους πλάγιους συνδέσμους προς κάθε ομώνυμο κόμβο του δένδρου προβολής και διασχίζει τα μονοπάτια τους προς την ρίζα του δένδρου. Στην χειρότερη περίπτωση θα μπορούσε να διασχίσει ολόκληρο το δένδρο προβολής. Το μέγιστο πλήθος κόμβων του δένδρου προβολής είναι το μέγεθος του αρχικού δένδρου σύνοψης: $|S_C|_{max} = |S| = |D| \cdot avgTr$, όπου $avgTr$ είναι το μέσο μέγεθος δενδρικής εγγραφής και $|D|$ είναι το πλήθος των εγγραφών. Αυτό συμβαίνει μόνο στο χειρότερο σενάριο όπου όλες οι δενδρικές εγγραφές έχουν διαφορετικές τιμές και δεν υπάρχουν επικαλύψεις μονοπατιών στη σύνοψη (Στην πράξη δεν συμβαίνει στα περισσότερα ρεαλιστικά σενάρια δεδομένων). Έτσι η πολυπλοκότητα της `StructureCheck` είναι $O(n \cdot log\mathcal{I} + n \cdot |D| \cdot avgTr)$. Σημειώνουμε ότι στην πράξη μόνο ένα μικρό υποσύνολο των κόμβων του δένδρου προβολής θα ελεγχθούν. Επιπλέον, το μέγεθος του δένδρου προβολής θα είναι αρκετά μικρότερο από το μέγεθος της σύνοψης, λόγω των απαραίτητων συγχωνεύσεων των κόμβων που προκύπτουν από τις γενικεύσεις των τιμών τους.

Η συνάρτηση `FixStructure` υπολογίζει όλους τους $\binom{|C|}{m} = O(\frac{|C|^m}{m!}) = O(\frac{\mathcal{I}^m}{m!})$ συνδυασμούς m τιμών από μια τομή μεγέθους $|C|$. Για κάθε τέτοιο συνδυασμό υπάρχουν

$\binom{m}{2} = O(m^2)$ πιθανά ζεύγη σχέσεων. Το πλήθος όλων των συνδυασμών από 1 έως n ζεύγη (για $n \leq \binom{m}{2}$) είναι: $\sum_{i=1}^{n}\binom{\binom{m}{2}}{i} = O(2^{m^2})$. Για κάθε τέτοιο συνδυασμό, καλείται η συνάρτηση `StructureCheck`. Αν επιστρέψει false, τότε η `FixStructure` προσπαθεί να επιλύσει το πρόβλημα. Η πολυπλοκότητα της `FixStructure` είναι: $O(\frac{|C|^m}{m!} \cdot [\sum_{i=1}^{n}\binom{\binom{m}{2}}{i})] \cdot [n \cdot log\mathcal{I} + n \cdot |D| \cdot avgTr]) = O(\frac{\mathcal{I}^m}{m!} \cdot 2^{m^2} \cdot n \cdot (log\mathcal{I} + |D| \cdot avgTr))$.

Το πλήθος των επαναλήψεων του αλγορίθμου `ACS` είναι στην χειρότερη περίπτωση ο μέγιστος αριθμός πιθανών τομών γενίκευσης πάνω στο δένδρο της ιεραρχίας ($totCuts$), ο οποίος μπορεί να εκφραστεί συναρτήσει του πεδίου τιμών των δεδομένων $\mathcal{I}$ και της παραμέτρου εξάπλωσης (fanout) f της ιεραρχίας γενίκευσης τιμών, δηλαδή το πλήθος παιδιών ανά κόμβο-τιμή της ιεραρχίας.
$totCuts = \sum_{i_1=0}^{f}\{\binom{f}{i_1} \cdot \sum_{i_2=0}^{f(f-i_1)}[\binom{f(f-i_1)}{i_1} \ldots \sum_{i_h=0}^{f(f\ldots(f(f-i_1)-i_2)\ldots-i_{h-1})}\binom{f(f\ldots(f(f-i_1)-i_2)\ldots-i_{h-1})}{i_h}]\}$
$+1 = O(2^{f^{(h-1)}}) = O(2^{\mathcal{I}})$, διότι $\mathcal{I} \approx f^h$.

Ο αλγόριθμος `GCS` εξετάζει μόνο τις g πιο υποσχόμενες τομές κάθε επιπέδου, διασχίζοντας την ιεραρχία από την ρίζα προς τα φύλλα. Το πλήθος των τομών που εξετάζονται σε κάθε βήμα είναι $g \cdot |Cut|$. Σε ένα ενδιάμεσο βήμα i, το μέγεθος τομής είναι $|Cut| = (if - i + 1)$. Ο αλγόριθμος τερματίζει μόλις φτάσει στο επίπεδο των φύλλων της ιεραρχίας γενίκευσης. Η τελευταία τομή πριν το επίπεδο των φύλλων έχει μέγεθος $f^h - f + 1$, διότι μόνο μια τιμή της δεν έχει γενικευθεί ακόμη. Τότε, το τελευταίο βήμα $i_{max} = O(f^{h-1})$. Το πλήθος των τομών που εξετάζονται από τον αλγόριθμο `GCS` είναι $totalCuts = 1 + f + g \cdot \sum_{i=2}^{f^{h-1}}(if - i + 1) = O(gf^{2h}) = O(g\mathcal{I}^2)$.

Στην πράξη, η συμπεριφορά ως προς τα m και I διαφέρει σημαντικά από την χειρότερη περίπτωση. Η πειραματική ανάλυση που θα παρουσιάσουμε στην επόμενη ενότητα έδειξε ότι στην πράξη ο χρόνος εκτέλεσης δεν αυξάνει εκθετικά με την παράμετρο m. Στην πραγματικότητα, μειώνεται έως μια τιμή της m και μετά αυξάνεται σχεδόν γραμμικά. Ο λόγος είναι ότι η παράμετρος m επηρεάζει την διαδικασία της ανωνυμοποίησης με δυο αντικρουόμενους τρόπους. Καθώς η m αυξάνεται, κάθε έλεγχος υποψήφιας λύσης γίνεται περισσότερο ακριβός υπολογιστικά, αλλά ταυτόχρονα περιορίζεται ο χώρος των πιθανών λύσεων και έτσι γλυτώνουμε επαναλήψεις. Ο αλγόριθμος εξετάζει λιγότερες και πιο απλές λύσεις (με λιγότερες πιο γενικές τιμές) καθώς το m μεγαλώνει. Το μέγεθος του πεδίου τιμών $\mathcal{I}$ δεν επηρεάζει την απόδοση σημαντικά στην πράξη. Ακόμη και αν το $\mathcal{I}$ αυξηθεί, ο αλγόριθμος δεν θα επηρεαστεί σημαντικά αν η αναζήτηση δεν φτάσει στις πιο εξειδικευμένες τομές στο κάτω μέρος του γράφου υποψηφίων τομών. Τέλος, ενώ η παράμετρος k δεν επηρεάζει άμεσα τους αλγόριθμους, εντούτοις καθορίζει πόσες υποψήφιες λύσεις θα ικανοποιούν την εγγύηση και έτσι μεγάλες τιμές της k περιορίζουν το πλήθος των υποψηφίων λύσεων που ια εξετάσουν οι αλγόριθμοι.

3.3.7 Επέκταση για την l-διαφορετικότητα

Η εγγύηση που προτείνουμε, όπως και η κλασσική k-ανωνυμία, δεν μπορεί να αντιμετωπίσει τις επιθέσεις ομοιότητας. Αν k εγγραφές στα τελικά δεδομένα γίνουν πανομοιότυπες μετά την ανωνυμοποίηση, και κάποιος επιτιθέμενος ταιριάξει την γνώση του με αυτές, τότε θα ανακαλύψει ολόκληρη την πληροφορία του στόχου. Σε αυτήν την περίπτωση δεν χρειάζεται

να ταυτοποιήσει την μοναδική εγγραφή του στόχου, εφόσον όλες αυτές οι k εγγραφές είναι πανομοιότυπες τόσο ως προς τις τιμές όσο και ως προς τις δομικές σχέσεις που περιέχουν. Δεν έχει σημασία ποια από αυτές ανήκει στον στόχο, ο επιτιθέμενος μπορεί να ανακαλύψει ολόκληρη την ευαίσθητη προσωπική του πληροφορία. Αυτό το πρόβλημα αντιμετωπίζεται με την επιβολή περιορισμών διαφορετικότητας πάνω στις πιο ευαίσθητες τιμές των δεδομένων, ανάμεσα σε εγγραφές που είναι όμοιες [78, 33].

Η μέθοδός μας μπορεί να επεκταθεί ώστε να ικανοποιεί την εγγύηση της l-διαφορετικότητας, υποθέτοντας ότι ορισμένα γνωρίσματα θεωρούνται ευαίσθητα και δεν δρουν ως ψευδο-αναγνωριστικά. Σε αυτήν την περίπτωση, θα προσθέταμε έναν ακόμα περιορισμό στις συναρτήσεις *ValueCheck* και *StructureCheck* που θα απαιτούσε ότι οι ομάδες δενδρικών εγγραφών που υποστηρίζουν ένα $k^{(m,n)}$-ανώνυμο συνδυασμό, θα πρέπει επιπλέον να είναι l-διαφορετικές, δηλαδή να περιέχουν l καλά-αντιπροσωπευμένες τιμές των ευαίσθητων γνωρισμάτων [52].

3.3.8 Αντιμετώπιση αρνητικής γνώσης

Το μοντέλο επίθεσης που μελετήσαμε δεν λαμβάνει υπόψη την πιθανότητα ένας επιτιθέμενος να διαθέτει αρνητική γνώση, δηλαδή να γνωρίζει ότι ένας κόμβος «δεν εμφανίζεται» στην δενδρική εγγραφή του στόχου. Όταν τα δεδομένα είναι αραιά, όπως είναι τα δενδρικά σύνολα δεδομένων, η αρνητική γνώση είναι λιγότερο σημαντική και επικίνδυνη σε σχέση με την θετική γνώση. υπάρχει ένα μικρό σύνολο τιμών και σχέσεων που περιέχονται σε μια εγγραφή, ενώ είναι πολυάριθμες οι τιμές και οι πιθανές σχέσεις ου δεν σχετίζονται με την εγγραφή. Εντούτοις, υπάρχουν περιπτώσεις που κάποιος επιτιθέμενος μπορεί να αποκτήσει αρνητική γνώση και να την χρησιμοποιήσει σε επιθέσεις πάνω στα ανωνυμοποιημένα δεδομένα. Θα μπορούσε για παράδειγμα να γνωρίζει ότι ο ασθενής δεν νοσηλεύθηκε ποτέ σε ένα νοσοκομείο που βρίσκεται σε διαφορετική πόλη.

Οι αλγόριθμοι ανωνυμοποίησης που προτείνουμε δουλεύουν με την παραδοχή ότι οι επιτιθέμενη δεν έχουν πρόσβαση σε τέτοια γνώση. Αν έπρεπε να προληφθούν συστηματικά όλες οι πιθανές επιθέσεις αρνητικής γνώσης, δηλαδή να λαμβάνουμε υπόψη κάθε αρνητική τιμή ή σχέση που δεν εμφανίζεται σε μια εγγραφή, τότε η προσαρμογή των αλγορίθμων μας για την εφαρμογή της κλασικής k-ανωνυμίας θα ήταν η πιο σωστή επιλογή, αν και θα έριχνε σημαντικά την ποιότητα των τελικών δεδομένων. Όμως, δεδομένου ότι η αρνητική γνώση δεν είναι το ίδιο επικίνδυνη και εύκολη στο να αποκτηθεί με την θετική γνώση, θα έχει μεγαλύτερο πρακτικό ενδιαφέρον να την λάβουμε *μερικώς* υπόψη. Παραδείγματος χάριν, θα μπορούσαμε να εξετάσουμε μόνο τις αρνήσεις νοσοκομείων, διότι ο επιτιθέμενος ίσως γνωρίζει ότι ο ασθενής στόχος δεν νοσηλεύθηκε σε νοσοκομείο που βρίσκεται σε διαφορετική πόλη, αλλά να αγνοήσουμε τις αρνήσεις των θεραπευτικών φαρμάκων, διότι είναι πιο δύσκολο ένας επιτιθέμενος να μπορεί να επαληθεύσει ότι ένας ασθενής δεν έλαβε ποτέ ένα συγκεκριμένο είδος θεραπείας.

Στις περιπτώσεις όπου η τιμή ενός κόμβου είναι σημαντική ή η πιθανότητα να γνωρίζει την άρνησή της ένας επιτιθέμενος είναι αυξημένη, μπορούμε να προσαρμόσουμε το μοντέλο δεδομένων μας, προσθέτοντας επιπλέον κόμβους. Η μη ύπαρξη μιας τιμής «a» θα μπορούσε να μοντελοποιηθεί προσθέτοντας ένα κόμβο «$-a$» σε κάθε εγγραφή όπου η «a» δεν εμφανίζεται.

Πλήθος εγγραφών $\|D\|$	100k,250k,1M	Γνωρίσματα	6
Μ. όρος κόμβων ανά εγγραφή	12	Μέγεθος Πεδίου Τιμών $\|\mathcal{I}\|$	276
Βάθος δένδρου Ιεραρχίας d	4	Παιδιά ανά κόμβο Ιεραρχίας f	5

Πίνακας 3.2: Περιγραφή των Δεδομένων

Ο κόμβος αυτός θα είναι απευθείας παιδί της ρίζας, καθώς καμία δομική σχέση δεν έχει νόημα σε αυτή την περίπτωση. Αντίστοιχα, μπορούμε να μοντελοποιήσουμε την αρνητική δομική γνώση ότι ένας ασθενής δεν νοσηλεύθηκε ποτέ για μια ασθένεια «b» στο νοσοκομείο «a», προσθέτοντας ένα κόμβο «$-b$» ως παιδί του «a».

Η παραπάνω προσαρμογή επιτρέπει στους αλγορίθμους μας την εφαρμογή της $k^{(m,n)}$-ανωνυμίας, λαμβάνοντας υπόψη θετική και αρνητική γνώση. Με άλλα λόγια, θα εγγυούμαστε την προστασία της ταυτότητας των εγγραφών από επιτιθέμενους που γνωρίζουν m τιμές οι οποίες εμφανίζονται ή δεν εμφανίζονται στην εγγραφή-στόχο, και n σχέσεις που ισχύουν ή δεν ισχύουν ανάμεσα στις τιμές, λαμβάνοντας υπόψη ένα υποσύνολο της πιθανής αρνητικής γνώσης. Με αυτόν τον τρόπο, αντιμετωπίζομε επιθέσεις από τις πιο πιθανές και επικίνδυνες περιπτώσεις αρνητικής γνώσης, χωρίς να εισάγουμε σημαντική επιπρόσθετη απώλεια πληροφορίας στα δεδομένα. Μετά την ανωνυμοποίηση, οι αρνητικοί κόμβοι μπορούν να διαγραφούν από τις τελικές εγγραφές, καθώς περιέχουν πλεονάζουσα πληροφορία.

3.4 Πειραματική Μελέτη

Σε αυτή την ενότητα παρουσιάζονται τα πειραματικά αποτελέσματα για την αποτίμηση της μεθόδου. Όλες οι υλοποιήσεις έγιναν σε γλώσσα προγραμματισμού C++ και η διεξαγωγή των πειραμάτων έγινε σε υπολογιστή Intel Core i7 CPU, με 6GB RAM, σε λειτουργικό σύστημα Ubuntu Linux.

3.4.1 Υλοποίηση.

Υλοποιήσαμε και συγκρίναμε 4 αλγορίθμους, συμπεριλαμβανομένων των `ACS` και `GCS` που περιγράφηκαν στην Ενότητα 3.3. Υλοποίησα και ένα τρίτο αλγόριθμο ο οποίος δεν κάνει καμία δομική αποσυσχέτιση στα δεδομένα. Αντίθετα, απορρίπτει κάθε υποψήφια λύση η οποία θα χρειαζόταν δομικές αποσυσχετίσεις προκειμένου να ικανοποιήσει την $k^{(m,n)}$-ανωνυμία. Ονομάζουμε αυτό τον αλγόριθμο `OnlyCutSearch (OCS)` και τον χρησιμοποιούμε ως σημείο αναφοράς για να γίνει καλύτερα κατανοητή η συνεισφορά της δομικής αποσυσχέτισης στην διαδικασία της ανωνυμοποίησης και στην ποιότητα των τελικών δεδομένων.

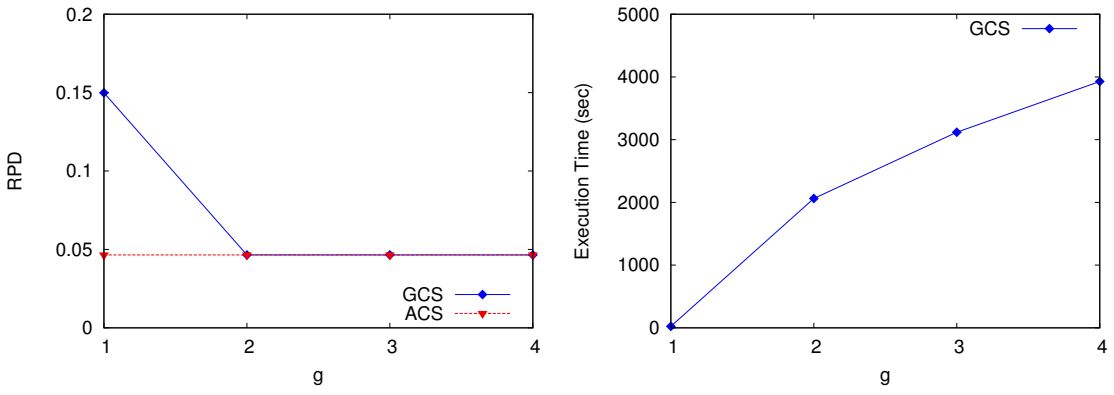
Επίσης, υλοποιήσαμε τον αλγόριθμο που σχετίζεται περισσότερο με το πρόβλημα που μελετάμε, τον `MiRaCle` [56], έναν αλγόριθμο με γενικεύσεις τοπικής ανακωδικοποίησης ο οποίος συσταδοποιεί τις εγγραφές σε k-ανώνυμες ομάδες. Ο αλγόριθμος `MiRaCle` χρησιμοποιεί μόνο γενίκευση και απαλοιφή για τον μετασχηματισμό των δεδομένων.

3.4.2 Πειραματικά Δεδομένα

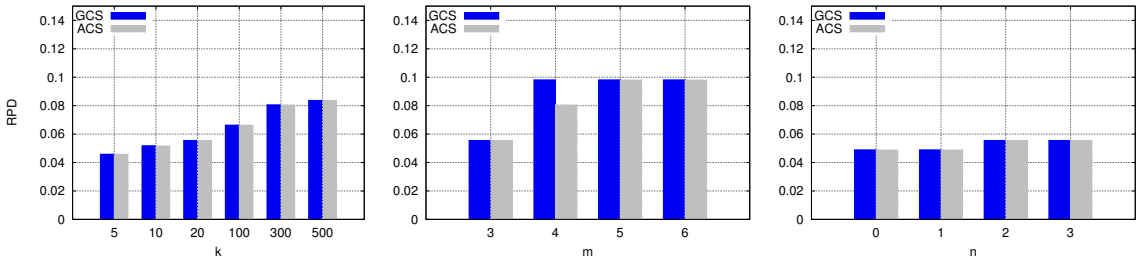
Για την πειραματική αποτίμηση της προτεινόμενης μεθόδου χρησιμοποιήθηκαν δεδομένα από τοTPC-H [8], που είναι ένα τυπικό παράδειγμα μιας βάσης δεδομένων με σχεσιακούς πίνακες για τους πελάτες, τα προϊόντα, τις παραγγελίες και τους προμηθευτές, όλοι συνδεδεμένοι μεταξύ τους μέσω ξένων κλειδιών. Εκτελώντας κατάλληλα sql ερωτήματα με συνδέσμους (joins) πάνω στα ξένα κλειδιά, συλλέξαμε όλη την πληροφορία των πελατών σε μια δενδρική εγγραφή ανά πελάτη. Τα δένδρα που προέκυψαν εκφράζουν την εξής πληροφορία: κάθε πελάτης (ρίζα του δένδρου εγγραφής) έχει πραγματοποιήσει έναν αριθμό παραγγελιών σε κάποιες ημερομηνίες, καθεμία περιέχει ένα διαφορετικό πλήθος προϊόντων. Για λόγους απλότητας της πειραματικής διαδικασίας κρατήθηκαν μόνο τα πεδία: χώρα πελάτη, ημερομηνία παραγγελίας, τιμή παραγγελίας, ποσότητα προϊόντος, κατασκευαστής και μάρκα προϊόντος (brand name) από τους σχεσιακούς πίνακες. Διατηρήθηκαν οι δομικές σχέσεις μεταξύ των τιμών όπως προκύπτουν από το σχήμα της βάσεις και τις συσχετίσεις λόγω ξένων κλειδιών. Χρησιμοποιώντας την γεννήτρια δεδομένων του TPC-H, κατασκευάσαμε τα δεδομένα που περιγράφονται στον Πίνακα 3.2. Πρώτα δημιουργήσαμε ένα σύνολο δεδομένων από 1,000,000 δενδρικές εγγραφές πελατών και στην συνέχεια με δειγματοληψία προέκυψαν δυο τυχαία υποσύνολα 250,000 και 100,000 εγγραφών, όπου το πρώτο είναι υπερσύνολο του δεύτερου. Περιορίσαμε την εξάπλωση (fanout) των εγγραφών, επιτρέποντας έως δύο παραγγελίες ανά πελάτη και έως τρία είδη προϊόντων ανά παραγγελία, ώστε ο λόγος του μεγέθους κάθε εγγραφής προς το συνολικό μέγεθος των δεδομένων να είναι μικρός. Τέλος δημιουργήσαμε μια συνδετική Ιεραρχία Γενίκευσης Τιμών για το σύνολο των γνωρισμάτων των δεδομένων, με μέση εξάπλωση 5 παιδιά ανά κόμβο.

3.4.3 Παράμετροι

Μελετήθηκε η συμπεριφορά των αλγορίθμων ως προς την μεταβολή των εξής παραμέτρων: α) η k παράμετρος που ελέγχει την αυστηρότητα της εγγύησης μας, β) η m παράμετρος που ποσοτικοποιεί την γνώση του επιτιθέμενου, γ) η n παράμετρος της δομικής πληροφο-

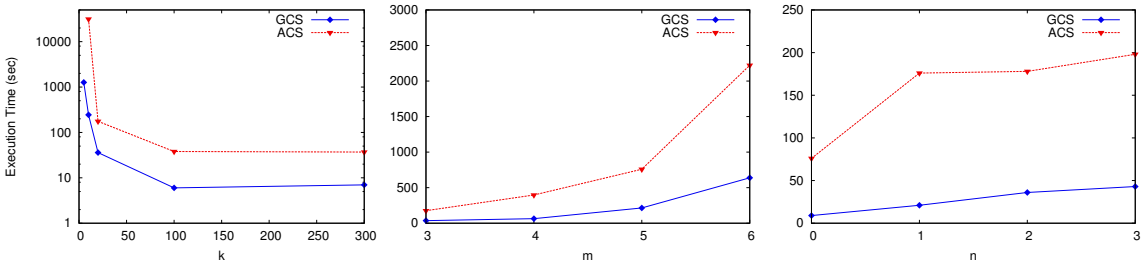


Σχήμα 3.16: Συμπεριφορά του GCS ως προς την παράμετρο g.



(α) (β) (γ)

Σχήμα 3.17: Σύκριση της απώλειας πληροφορίας (RPD) μεταξύ των ACS και GCS ως προς τις παραμέτρους k, m και n.



(α) (β) (γ)

Σχήμα 3.18: Σύκριση του χρόνου εκτέλεσης μεταξύ των ACS και GCS ως προς τις παραμέτρους k, m και n.

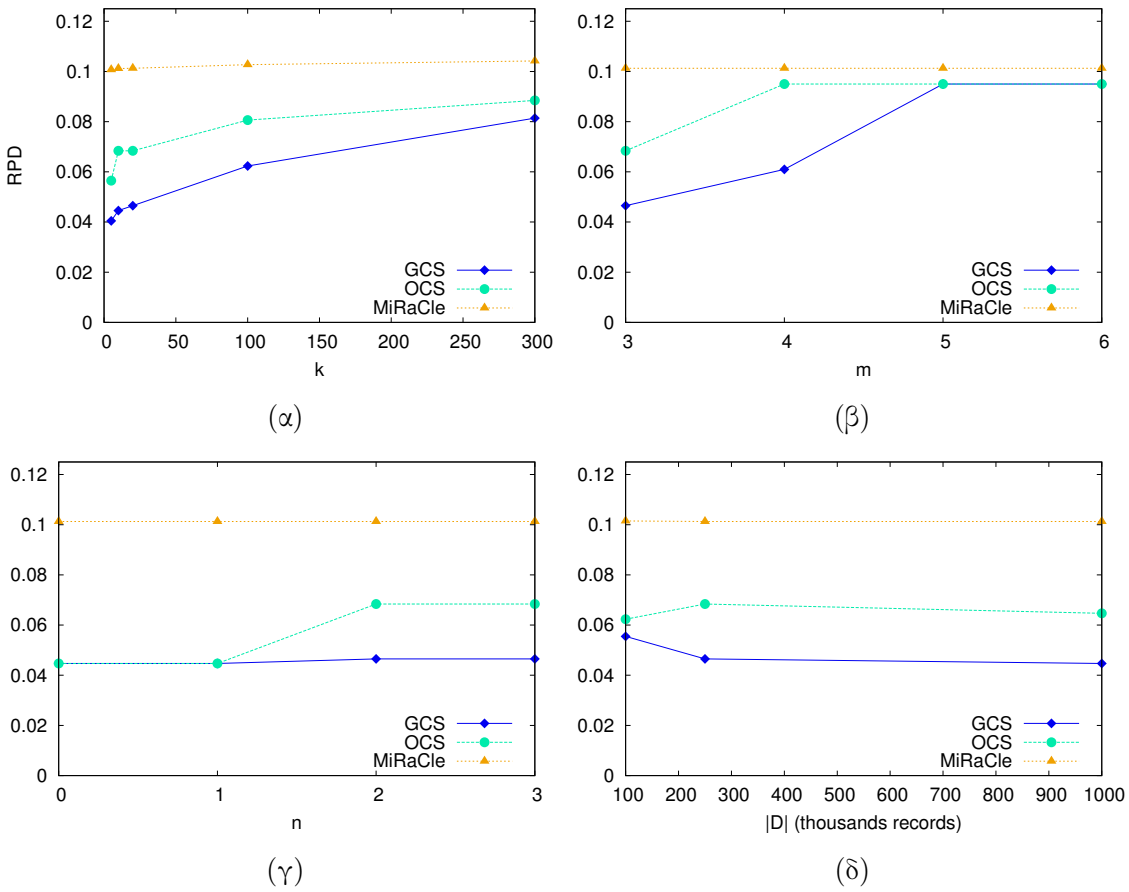
ρίας που μπορεί να χρησιμοποιηθεί ως ψευδο-αναγνωριστικό, δ) το μέγεθος των δεδομένων (σε πλήθος δενδρικών εγγραφών) $|D|$ και ε) η παράμετρος g του αλγόριθμου GCS. Σε κάθε πείραμα μεταβάλουμε μία από τις παραπάνω παραμέτρους κρατώντας σταθερές τις υπόλοιπες. Οι προκαθορισμένες τιμές που επιλέξαμε για τα πειράματα είναι $k = 20$, $m = 3$, $n = 2$, $|D| = 250,000$ και $g = 2$. Μετά από δοκιμές καταλήξαμε στις βέλτιστες για τα δεδομένα μας τιμές των παραμέτρων *climit* και *threshold* του αλγορίθμου MiRaCle, οι οποίες είναι 150 και 0.1 αντιστοίχως.

3.4.4 Μετρικές Αποτίμησης

Για την αξιολόγηση της μεθόδου μας μετρήθηκε ο χρόνος εκτέλεσης των πειραμάτων καθώς και οι μετρικές RPD, ML^2 και dML^2 για την αποτίμηση της ποιότητας των αποτελεσμάτων.

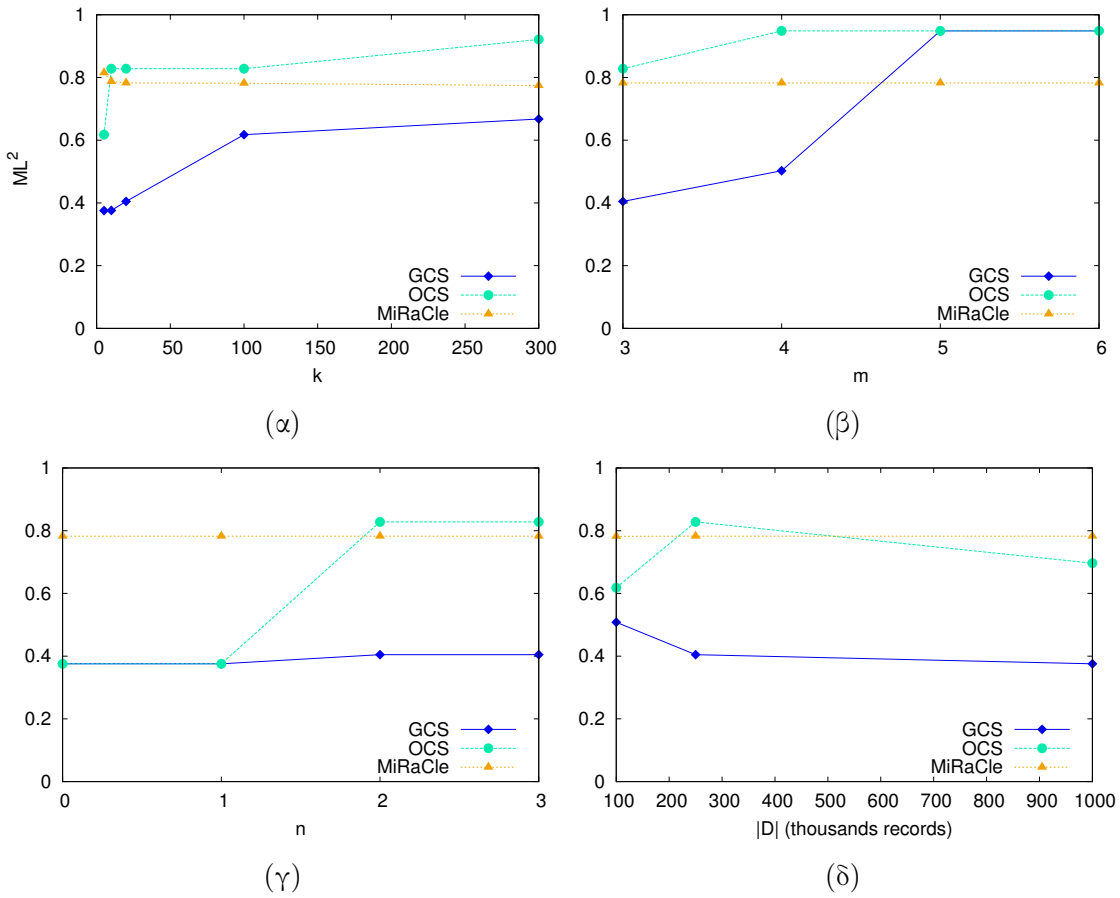
3.4.5 Σύγκριση των ACS και GCS

Η πρώτη σειρά πειραμάτων έχει ως στόχο την αποτίμηση της διαφοράς στην απόδοση μεταξύ του ACS και του άπληστου ευρηστικού αλγόριθμου GCS. Στο Σχήμα 3.16 φαίνεται η συμπεριφορά του GCS καθώς η παράμετρος g αυξάνει από 1 σε 4. Ήδη για $g = 2$ η ποιότητα των αποτελεσμάτων του GCS (κρινόμενη από την μετρική απώλειας πληροφορίας RPD) συγκλίνει



Σχήμα 3.19: Σύκριση της απώλειας πληροφορίας (RPD) μεταξύ των `ACS`, `GCS` και `MiRaCle` ως προς τις παραμέτρους k, m, n και $|D|$.

προς αυτή του `ACS`. Για μεγαλύτερες τιμές του g ο χρόνος εκτέλεσης αυξάνεται (αλλά υπογραμμικά), ενώ η απώλεια πληροφορίας παραμένει σχεδόν σταθερή και αντίστοιχη με εκείνη του `ACS`. Αυτή η σύγκλιση για $g = 2$ επιβεβαιώνεται και από τα αποτελέσματα που παρουσιάζονται στο Σχήμα 3.17. Η απόδοση ως προς την απώλεια πληροφορίας είναι παρόμοια για όλες τις τιμές των παραμέτρων k, m και n, ενώ σε πολλές περιπτώσεις οι δυο αλγόριθμοι καταλήγουν στην ίδια λύση. Εντούτοις, υπάρχει τεράστια απόκλιση στο υπολογιστικό κόστος των δύο αλγορίθμων. Όπως φαίνεται στο Σχήμα 3.18 ο `GCS` είναι ταχύτερος κατά τουλάχιστον μια τάξη μεγέθους στις περισσότερες περιπτώσεις. Για τα πειράματα των γραφικών παραστάσεων που παρουσιάζονται στα Σχήματα 3.17 και 3.18 χρησιμοποιήθηκε το μικρότερο σύνολο δεδομένων μεγέθους 100000 εγγραφών, καθώς το υπολογιστικό κόστος του `ACS` αυξάνει σημαντικά για μεγαλύτερα δεδομένα. Τα αποτελέσματα δείχνουν ότι ο `GCS` παρέχει σχεδόν τη ίδιας ποιότητας ανώνυμα δεδομένα με αυτά του `ACS` με μόλις ένα μέρος του υπολογιστικού του κόστους. Λόγω αυτής της διαπίστωσης στην υπόλοιπη ενότητα εστιάζουμε την πειραματική ανάλυση στον αλγόριθμο `GCS`.

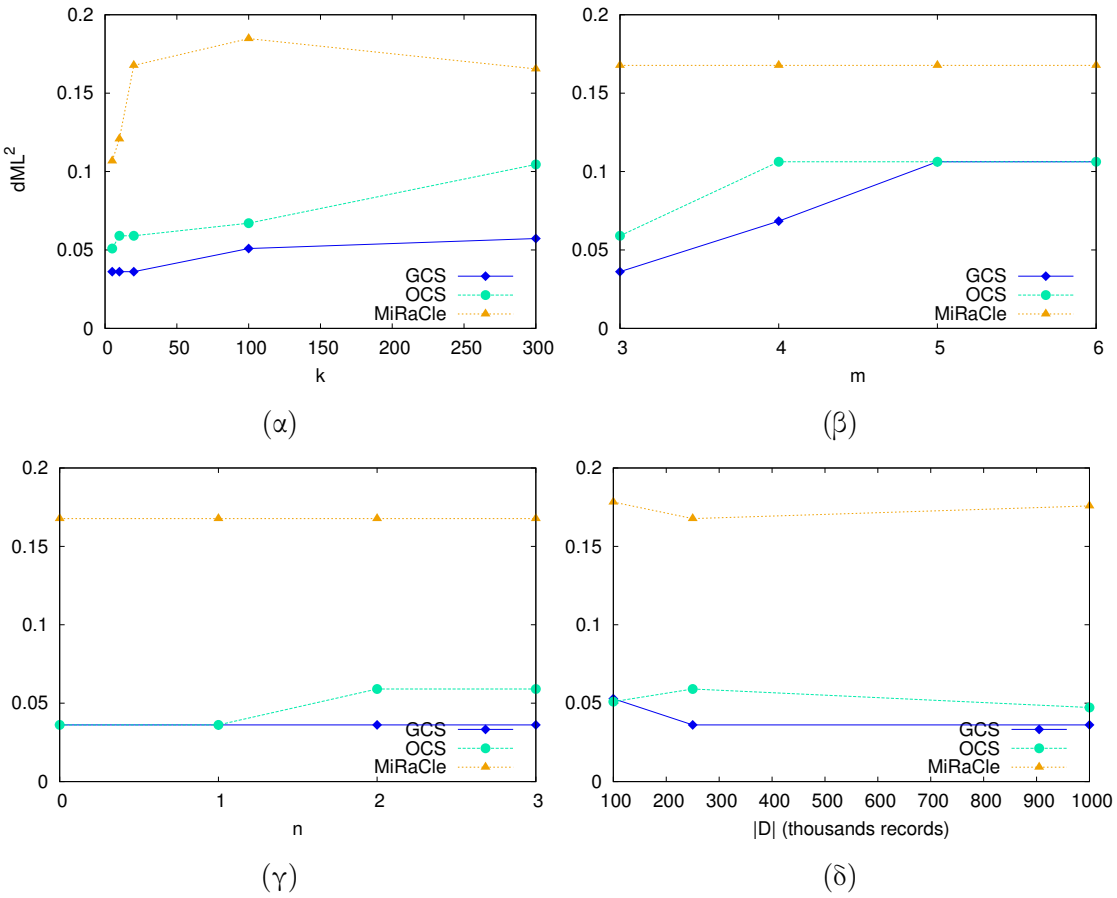


Σχήμα 3.20: Σύκριση της μετρικής ML^2 μεταξύ των ACS, GCS και MiRaCle ως προς τις παραμέτρους k, m, n και $|D|$.

3.4.6 Ποιότητα των Ανώνυμων Αποτελεσμάτων

Στα Σχήματα 3.19, 3.20 και 3.21 αξιολογείται η απόδοση του GCS σε σχέση με την ποιότητα της ανωνυμοποίησης. Συγκρίνεται πειραματικά τόσο με τον αλγόριθμο MiRaCle όσο και με τον OCS που απορρίπτει κάθε υποψήφια λύση που θα χρειαζόταν δομικές αλλαγές. Τα αποτελέσματα δείχνουν ότι ο GCS καταφέρνει να διατηρήσει περισσότερη χρήσιμη πληροφορία στα δεδομένα σε σχέση με αλγόριθμους οι οποίοι δεν διαθέτουν την επιλογή της δομικής αποσυσχέτισης που προτείναμε.

Στο Σχήμα 3.19 παρουσιάζεται η σύγκριση των τριών αλγορίθμων ως προς την μετρική RPD. Παρά το γεγονός ότι ο MiRaCle χρησιμοποιεί γενίκευση με μερική ανακωδικοποίηση, αδυνατεί να ανταγωνιστεί τους προτεινόμενους αλγόριθμους. Εισάγει μεγαλύτερο κόστος RPD σε κάθε σενάριο που μελετήθηκε ενώ για $m < 5$ το RPD του MiRaCle είναι διπλάσιο από το RPD του GCS. Η κατώτερη απόδοση του MiRaCle στην ποιότητα των αποτελεσμάτων οφείλεται σε δύο βασικούς παράγοντες: α) η εγγύηση ιδιωτικότητας που παρέχουν οι GCS και OCS είναι πιο ευέλικτη από την κλασσική k-ανωνυμία και β) το γεγονός ότι ο MiRaCle δεν διαθέτει την πράξη των δομικών αποσυσχετίσεων τον αναγκάζει να κάνει πολλές απαλοιφές κόμβων. Σε κάθε απαλοιφή κόμβου διαγράφεται ολόκληρο το υποδένδρο που σχετιζόταν μαζί

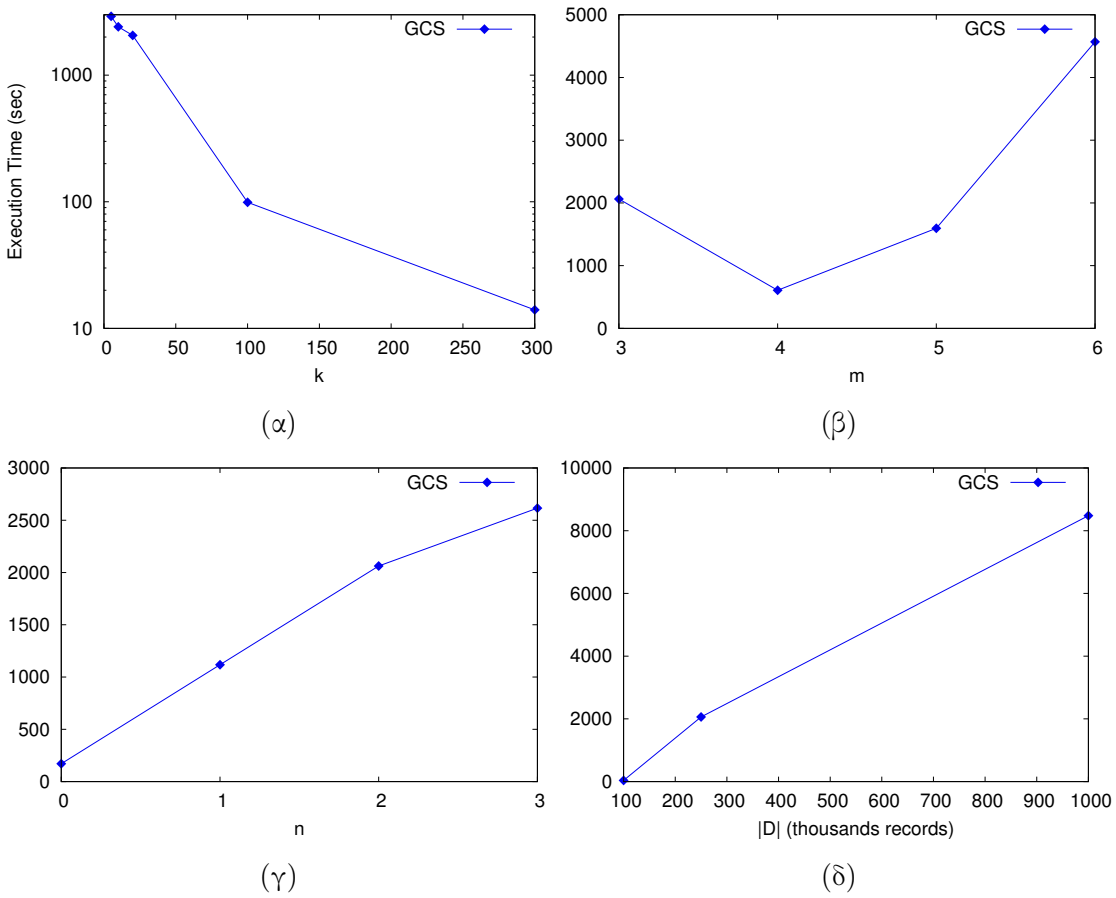


Σχήμα 3.21: Σύκριση της μετρικής dML^2 μεταξύ των ACS, GCS και MiRaCle ως προς τις παραμέτρους k, m, n και $|D|$.

του και έτσι χάνεται σημαντική πληροφορία από τα δεδομένα. Όπως ήταν αναμενόμενο, ο ενΟ^Σ εισάγει μεγαλύτερη απώλεια πληροφορίας από τον GCS για κάθε τιμή των παραμέτρων. Η διαφορά απόδοσης ανάμεσα στον GCS και τον OCS γίνεται σημαντικότερη στις πιο χαλαρές εγγυήσεις: Καθώς το k μειώνεται από 300 σε 5 η διαφορά ανάμεσα στους δυο αλγορίθμους αυξάνει από 8.7% σε 39.5%. Επίσης, μια μείωση του m από 6 σε 4 επίσης αυξάνει την απώλεια πληροφορίας του OCS σε έως και 54%υψηλότερη από την αντίστοιχη του GCS.

Το βασικό πλεονέκτημα του GCS γίνεται εμφανές όταν θεωρούμε ότι ο επιτιθέμενος έχει δομική γνώση. Όταν η τιμή της παραμέτρου n είναι 0, ο επιτιθέμενος δεν γνωρίζει καμία δομική σχέση μεταξύ των τιμών. Συνεπώς και οι δύο αλγόριθμοι GCS και OCS παράγουν το ίδιο ανώνυμο αποτέλεσμα και ξεπερνούν σε ποιότητα τονMiRaCle κατά 56%. Καθώς το n αυξάνεται η απόδοση του GCS καταφέρνει να παραμένει σχετικά σταθερή, ενώ ο OCS εισάγει 53% περισσότερη απώλεια πληροφορίας από ότι για $n = 0$. Η απώλεια RPD του OCS γίνεται 1.5 φορές υψηλότερη από εκείνη του GCS για $n = \{2, 3\}$. Ο MiRaCle παραμένει σταθερός, αλλά έχει ήδη χειροτερέψει σημαντικά την ποιότητα των δεδομένων.

Καθώς το μέγεθος των δεδομένων $|D|$ αυξάνει από 100,000 σε 1,000,000 εγγραφές, ο GCS καταφέρνει να μειώσει την απώλεια πληροφορίας, εκμεταλλευόμενος την ευελιξία των



Σχήμα 3.22: Χρόνος Εκτέλεσης του GCS ως προς τις παραμέτρους k, m, n και $|D|$.

πράξεων μετασχηματισμού των δεδομένων που διαθέτει. Αντίθετα ο OCS ρίχνει την ελαφρώς την ποιότητα. Η απώλεια πληροφορίας του MiRaCle είναι σχετικά σταθερή και κατά μέσο όρο διπλάσια από εκείνη του GCS.

Στα Σχήματα 3.20 και 3.21 παρουσιάζονται τα πειραματικά αποτελέσματα για τις μετρικές ML^2 και dML^2 αντιστοίχως. Για τον υπολογισμό τους πραγματοποιήθηκε εξόρυξη των συχνών υποδένδρων τόσο στα ανώνυμα όσο και στα αρχικά δεδομένα, προβεβλημένα σε όλα τα επίπεδα της Ιεραρχίας Γενίκευσης Τιμών. Το κατώφλι συχνότητας που χρησιμοποιήθηκε είναι 1%. Με άλλα λόγια συλλέξαμε τα συχνά υποδένδρα που εμφανίζονται σε τουλάχιστον 2,500 εγγραφές.

Οι τιμές των αποτελεσμάτων των μετρικών ML^2 και dML^2 ακολουθούν όμοια συμπεριφορά με εκείνη του RPD. Στην περίπτωση του ML^2, ο MiRaCle καταφέρνει να ξεπεράσει σε κάποιες περιπτώσεις τους OCS και GCS. Εντούτοις, αυτό συμβαίνει όταν τα ανώνυμα αποτελέσματα είναι χαμηλής ποιότητας για όλους και η μετρική ML^2 είναι κοντά στο 0.8, που σημαίνει ότι το 80% των συχνών υποδένδρων έχουν χαθεί. Αυτό συμβαίνει για παράδειγμα όταν $m \geq 5$ όπου το μέγεθος της γνώσης του επιτιθέμενου είναι το μισό του μέσου μεγέθους εγγραφής. Αντίθετα, για $m < 5$, ο GCS επιτυγχάνει αποτελέσματα καλύτερης ποιότητας, εισάγοντας κατά 40% με 50% λιγότερη απώλεια πληροφορίας σε σχέση με τον MiRaCle. Παραδείγματος χάριν,

στην περίπτωση του συνόλου δεδομένων των 250,000 εγγραφών, εξορύχτηκαν 10845 συχνά υποδένδρα στα αρχικά δεδομένα, προβεβλημένα σε όλα τα επίπεδα γενίκευσης. Ενώ ο OCS διατηρεί μόλις 1864 από αυτά, ο MiRaCle διατηρεί 2358 υποδένδρα στα τελικά δεδομένα. Και οι δυο ξεπεράστηκαν από τον GCS ο οποίος κατάφερε να διατηρήσει 6457 συχνά υποδένδρα για τις προεπιλεγμένες τιμές των παραμέτρων ανωνυμοποίησης ($k = 20$, $m = 3$, $n = 2$).

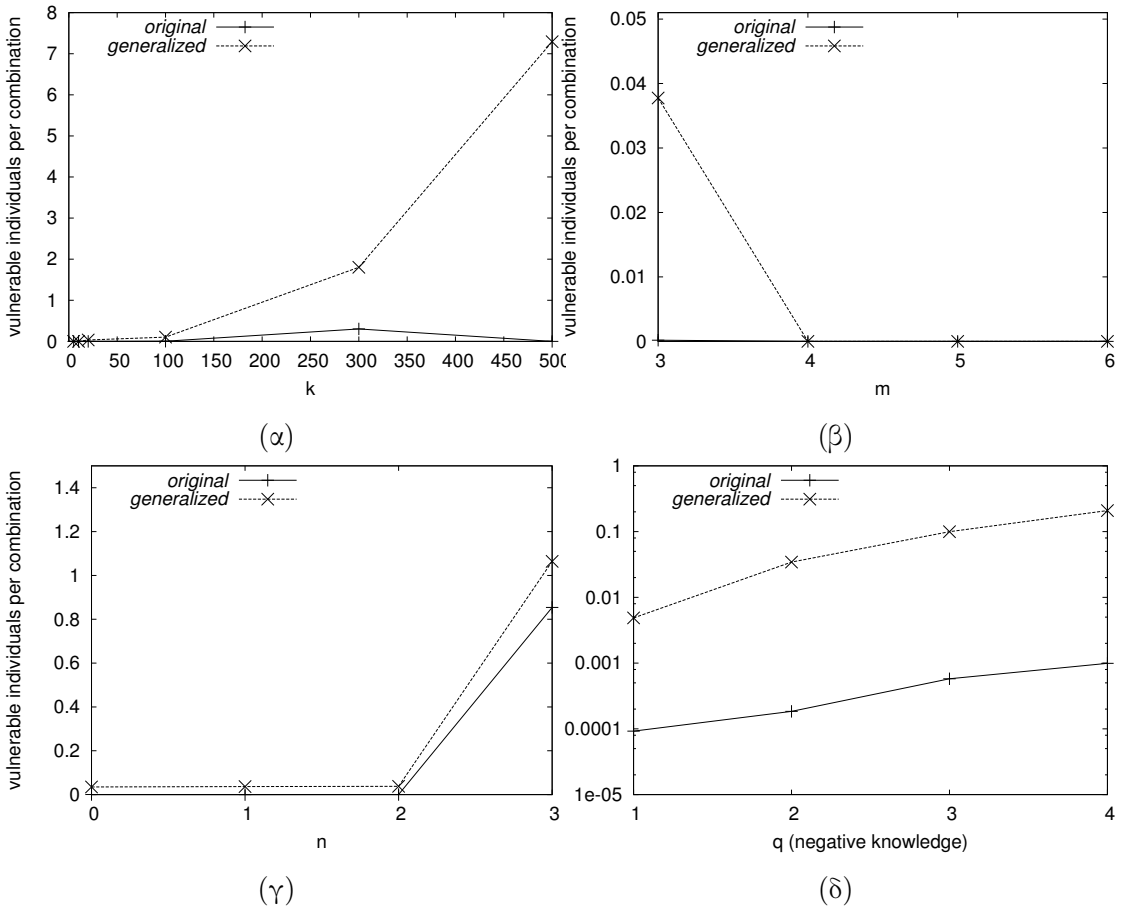
Τα αποτελέσματα του dML^2 υποδεικνύουν ότι ακόμα και όταν τα ακριβή αρχικά υποδένδρα δεν μπορούν να εξορυχτούν από τα ανώνυμα δεδομένα, η διαφορά ανάμεσα στα συχνά υποδένδρα που εξορύχτηκαν και στα αρχικά είναι αρκετά μικρή. Πράγματι, ενώ το ML^2 είναι κοντά στο 80% για τον MiRaCle και υψηλά για τους άλλους δύο αλγόριθμους, οι αντίστοιχες τιμές του dML^2 είναι κάτω του 19%. Οι γραφικές παραστάσεις του Σχήματος 3.21 αναδεικνύουν την υπεροχή της μεθόδου μας. Τόσο ο GCS όσο και ο OCS διατηρούν σημαντικά μεγαλύτερη πληροφορία και είναι νικητές έναντι του MiRaCle, ο οποίος εφαρμόζει αυστηρές γενικεύσεις και απαλοιφές που οδηγούν σε απώλειες ολόκληρων υποδένδρων. Για μια ακόμα φορά ο GCS υπερτερεί καθώς είναι πιο ανθεκτικός απέναντι στην αύξηση του n, ενώ η απώλεια πληροφορίας του OCS αυξάνεται σημαντικά καθώς το n μεγαλώνει.

3.4.7 Χρόνος Εκτέλεσης

Το υπολογιστικό κόστος του αλγόριθμου GCS παρουσιάζεται στις γραφικές παραστάσεις του Σχήματος 3.22. Στο πρώτη γραφική φαίνεται η συμπεριφορά του αλγορίθμου ως προς την παράμετρο k. Υψηλές τιμές του k επιτρέπουν στον GCS να κλαδεύει σημαντικό μέρος του χώρου των λύσεων και να μειώνει αισθητά τον χρόνο εκτέλεσης των πειραμάτων. Το υπολογιστικό κόστος πέφτει κατά 97% καθώς το k αυξάνεται από 5 σε 100 ενώ μειώνεται κατά ακόμα 85.8% όταν το k παίρνει την τιμή 300. Η παράμετρος m επηρεάζει τον χρόνο με δυο διαφορετικούς και αντίθετους τρόπους όπως εξηγήθηκε στην Ενότητα 3.3.6. Το γεγονός αυτό προκαλεί ένα τοπικό ελάχιστο για $m = 4$, όπως φαίνεται στην γραφική παράσταση του Σχήματος 3.22(β). Όταν το m αυξάνει από 3 σε 4, ο χρόνος εκτέλεσης μειώνεται κατά -70.5%, ενώ πέραν της τιμής 4 αυξάνεται αισθητά. Η παράμετρος n δεν περιορίζει τον χώρο αναζήτησης υποψηφίων λύσεων του αλγορίθμου, αλλά ορίζει το πλήθος των δομικών σχέσεων που μπορεί να γνωρίζει ένας επιτιθέμενος. Συνεπώς, το υπολογιστικό κόστος αυξάνεται με την άνοδο του n, όπως φαίνεται στα Σχήμα 3.22(γ). Ο λόγος για την αύξηση του χρόνου εκτέλεσης είναι ότι όσο μεγαλύτερο είναι το n, τόσο περισσότεροι συνδυασμοί από ζεύγη τιμών πρέπει να ελεγχθούν ως πιθανή γνώση του επιτιθέμενου. Στην γραφική παράσταση του Σχήματος 3.22(δ) παρατηρούμε ότι το υπολογιστικό κόστος του αλγόριθμου GCS κλιμακώνει γραμμικά με το μέγεθος των δεδομένων $|D|$.

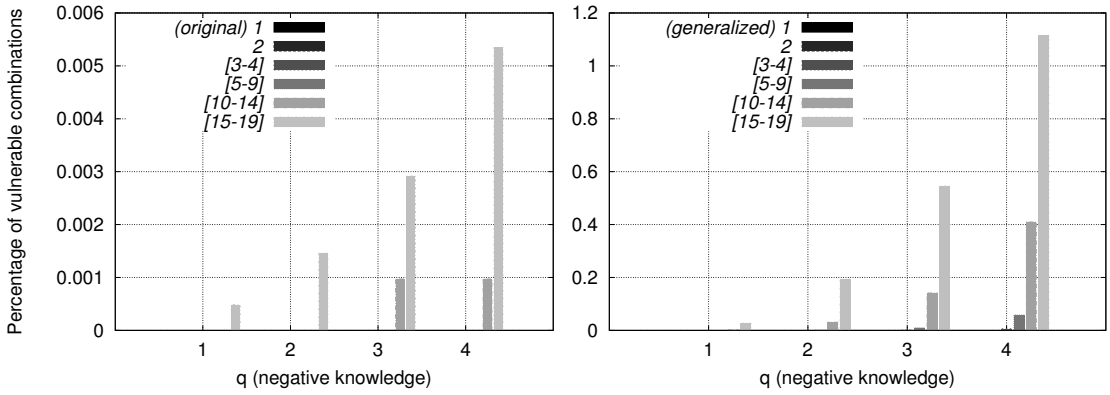
3.4.8 Αρνητική Γνώση

Για να στηρίζουμε την υπόθεση ότι η αρνητική γνώση δεν συνιστά αξιόλογο κίνδυνο στα περισσότερα ρεαλιστικά σενάρια επίθεσης, προσομοιώνουμε επιθέσεις με συνδυασμούς θετικής και αρνητικής γνώσης πάνω στα δεδομένα που έχει ανωνυμοποιήσει η μέθοδός μας. Διεξάγουμε το ακόλουθο πείραμα: για κάθε συνδυασμό γνώσης m τιμών και n δομικών σχέσεων



Σχήμα 3.23: Ποσοστό των ευάλωτων ατόμων ανά συνδυασμό γνώσης που περιέχει και αρνητικές τιμές, ως προς τις παραμέτρους k, m, n και την ποσότητα αρνητικής γνώσης q.

μεταξύ τους, επιλέγουμε τυχαία από 1 έως 4 επιπλέον τιμές, τις οποίες θεωρούμε ως αρνητική γνώση του επιτιθέμενου. Ας θεωρήσουμε για παράδειγμα έναν επιτιθέμενο με θετική γνώση $\{a, b\}$, επιλέγουμε τυχαία μια επιπλέον τιμή x από το πεδίο των αρχικών τιμών των δεδομένων και υποθέτουμε ότι η γνώση του επιτιθέμενου είναι $\{a, b,$ όχι $x\}$, δηλαδή γνωρίζει ότι στην εγγραφή του στόχου θα πρέπει να εμφανίζονται οι τιμές a και b, αλλά όχι η τιμή x. Λαμβάνουμε υπόψη επιτιθέμενους οι οποίοι γνωρίζουν την άρνηση τιμών από το αρχικό πεδίο τιμών των δεδομένων (είναι οι τιμές που βρίσκονται στα φύλλα της Ιεραρχίας Γενίκευσης Τιμών) και επιτιθέμενους που γνωρίζουν την άρνηση γενικευμένων τιμών από το πρώτο επίπεδο πάνω από τα φύλλα της Ιεραρχίας Γενίκευσης. Τα αποτελέσματα που αφορούν στην πρώτη κατηγορία επιτιθέμενων τα ονομάζουμε «original» ενώ όσα αφορούν στην δεύτερη κατηγορία επιτιθέμενων εμφανίζονται με την ετικέτα «generalized» στις γραφικές παραστάσεις των Σχημάτων 3.23-3.24. Στην περίπτωση των επιτιθέμενων που γνωρίζουν αρνήσεις τιμών από τα αρχικά δεδομένα, οι επιθέσεις αντιμετωπίζονται με ευκολία: όταν οι αρνητικές τιμές δεν εμφανίζονται στο αποτέλεσμα γιατί έχουν γενικευθεί, δεν είναι πλέον χρήσιμες στον επιτιθέμενο. Αν ο επιτιθέμενος γνωρίζει «όχι Γρίπη» και η «Γρίπη» έχει γενικευθεί σε «Πνευμονικό Νόσημα» στα ανώνυμα δεδομένα, τότε ο επιτιθέμενος δεν μπορεί να αποκλείσει τις εγγραφές



(α) (β)

Σχήμα 3.24: Αντίκτυπο της αρνητικής γνώσης q: Ποσοστό των ευαίσθητων συνδυασμών ανάλογα με το πλήθος εγγραφών που επιστρέφονται (1-19), για $k = 20$.

που έχουν την τιμή «Πνευμονικό Νόσημα» καθώς μπορεί να αναφέρονται σε κάποια άλλη ασθένεια εκτός της γρίπης όπως «Πνευμονία» ή «Βρογχίτιδα». Επιλέγουμε τις αρνητικές τιμές με τέτοιο τρόπο ώστε η συνολική γνώση του επιτιθέμενου, θετική και αρνητική, να ταιριάζει με τουλάχιστον μία αρχική εγγραφή στα δεδομένα. Δεν θεωρούμε την περίπτωση της γνώσης αρνήσεων πιο γενικευμένων τιμών σε υψηλότερα επίπεδα της ιεραρχίας, καθώς θα είναι απίθανο να βρεθεί συνδυασμός με τόσο γενική αρνητική γνώση που να ταιριάζει σε τουλάχιστον 1 εγγραφή στα δεδομένα μας.

Στα πειράματα του Σχήματος 3.23 μετρήθηκε το πλήθος των ατόμων των οποίων ιδιωτικότητα μπορούσε να παραβιαστεί, δηλαδή το πλήθος των δενδρικών εγγραφών που περιέχουν συνδυασμούς γνώσης με υποστήριξη (support) μικρότερη του k στα ανωνυμοποιημένα δεδομένα, όταν λαμβάνουμε υπόψη την αρνητική γνώση. Το προεπιλεγμένο μέγεθος αρνητικής γνώσης που θέσαμε στα πειράματα είναι $q = 2$, δηλαδή θεωρούμε επιτιθέμενους που μπορεί να γνωρίζουν την ταυτόχρονη άρνηση 2 τιμών « (όχι x) και (όχι y) ». Στις γραφικές παραστάσεις απεικονίζεται ο μέσος όρος των ατόμων των οποίων η ιδιωτικότητα παραβιάζεται, ανά αρχικό συνδυασμό θετικής γνώσης.

Στο Σχήμα 3.23(α) παρατηρούμε πως αλλάζει ο αριθμός αυτός για διαφορετικά k. Καθώς η παράμετρος αυξάνει, ο μέσος όρος των ευάλωτων ατόμων ανά συνδυασμό γνώσης γίνεται επίσης μεγαλύτερος, καθώς περισσότερες προσωπικές εγγραφές μπορούν να βρεθούν σε συνδυασμούς γνώσης με support (έστω και λίγο) μικρότερο του k, λόγω της παρουσίας αρνητικής γνώσης που δεν είχαμε λάβει υπόψη κατά την ανωνυμοποίηση. Όταν θεωρούμε την γνώση της άρνησης αρχικών μη-γενικευμένων τιμών (original), στην χειρότερη περίπτωση για $k = 300$ απειλούνται μόλις 0.3 άτομα ανά συνδυασμό κατά μέσο όρο, που σημαίνει οτι οι ευάλωτοι συνδυασμοί ήταν ελάχιστοι, μιας και δεν υπήρχαν συνδυασμοί τόσο σπάνιοι ώστε να οδηγούν σε μοναδική αναγνώριση κάποιας εγγραφής. Μια ενδιαφέρουσα παρατήρηση είναι οτι για $k = 500$ δεν υπήρχε κανένας ευάλωτος συνδυασμός με support μικρότερο του k από τους «original» επιτιθέμενους. Ο λόγος είναι ότι όταν οι τιμές της αρνητικής γνώσης έχουν γενικευθεί στα ανωνυμοποιημένα δεδομένα, δεν έχουν πια καμία χρησιμότητα για τον

επιτιθέμενο, όπως εξηγήθηκε παραπάνω.

Η αύξηση της παραμέτρου m μεγαλώνει την πιθανότητα περισσότερες τιμές να γενικευθούν σε υψηλότερα επίπεδα της Ιεραρχίας Γενίκευσης στα ανωνυμοποιημένα δεδομένα. Αν η γνώση του επιτιθέμενου περιλαμβάνει την άρνηση αρχικών τιμών, δεν μπορεί να τις χρησιμοποιήσει για την αναγνώριση εγγραφών. Συνεπώς για μεγαλύτερα m συμβαίνουν πολύ λιγότερες παραβιάσεις τις ιδιωτικότητας, όπως φαίνεται στο Σχήμα 3.23(β). Παρατηρούμε ότι για $m > 4$ ο μέσος αριθμός ατόμων για τα οποία η $k^{(m,n)}$-ανωνυμία παραβιάζεται είναι μηδέν.

Η αύξηση της παραμέτρου n έχει περισσότερες πιθανότητες να προκαλέσει δομικές αποσυσχετίσεις στα τελικά δεδομένα παρά να επηρεάσει το επίπεδο γενίκευσης των τιμών τους. Για μεγαλύτερες τιμές του n, ένας συνδυασμός γνώσης που αποτελείται από m τιμές συν n δομικές σχέσεις συν q επιπλέον αρνητικές τιμές είναι πιθανότερο να είναι σπάνιος από έναν αντίστοιχο συνδυασμό με μικρότερο n. Όπως φαίνεται στο Σχήμα 3.23(γ), για $n < 2$ ο μέσος συνδυασμός ευάλωτων ατόμων ανά συνδυασμό γνώσης επιτιθέμενου είναι κάτω από 0.038, ενώ για $n = 3$ ανεβαίνει σε 0.85 για την άρνηση αρχικών τιμών και σε 1.06 για την αντίστοιχη άρνηση γενικευμένων τιμών.

Στο Σχήμα 3.23(δ) παρατηρούμε πως κλιμακώνει ο μέσος όρος των ευάλωτων ατόμων καθώς μεταβάλλεται η ποσότητα αρνητικής γνώσης q από 1 έως 4 τιμές, κρατώντας σταθερές τις προκαθορισμένες τιμές των παραμέτρων $k = 20$, $m = 3$ και $n = 2$. Παρατηρούμε ότι ακόμα και όταν θεωρούμε την αρνητική γνώση 4 αρχικών τιμών (το 1/3 του μέσου μεγέθους εγγραφής), απειλούνται λιγότερα από 0.001 άτομα ανά συνδυασμό γνώσης. Ενώ όταν θεωρούμε την άρνηση 4 γενικευμένων τιμών ο αριθμός αυξάνει σε μόλις 0.21 άτομα ανά συνδυασμό επίθεσης.

Στις γραφικές παραστάσεις του Σχήματος 3.24 εξετάζουμε τον *βαθμό επικινδυνότητας* στον οποίο εκτίθενται τα ευάλωτα άτομα που απειλούνται. Απεικονίζεται η κατανομή της υποστήριξης (support) των προβληματικών συνδυασμών, δηλαδή των συνδυασμών support μικρότερο από k. Ο οριζόντιος άξονας των x απεικονίζει την ποσότητα αρνητικής γνώσης που κατείχε ο επιτιθέμενος σε κάθε πείραμα. Ο κατακόρυφος άξονας των y καταμετρά το ποσοστό των συνολικών συνδυασμών που πέφτουν σε κάθε «κουβά». Οι κουβάδες περιέχουν συνδυασμούς με support: 1 (μοναδική αναγνώριση - ταυτοποίηση), 2, [3-5], [5-10], [10-15] και [15-20], δεδομένου ότι $k = 20$. Σημειώνεται ότι δεν συμπεριλαμβάνονται στην καταμέτρηση οι συνδυασμοί με υποστήριξη support=20, καθώς αυτή είναι η ελάχιστη ασφαλής υποστήριξη για ένα συνδυασμό γνώσης. Ουσιαστικά, ένας κουβάς με ετικέτα [α-β] αντιστοιχεί στο εύρος τιμών υποστήριξης [α, β).

Στο Σχήμα 3.24(α), θεωρούμε ότι ο επιτιθέμενος γνωρίζει συνδυασμούς που περιλαμβάνουν την άρνηση τιμών από τα αρχικά δεδομένα. Παρατηρούμε ότι αδυνατεί να περιορίσει την υποστήριξη των συνδυασμών σε λιγότερο από 10 με 14 εγγραφές (για $k = 20$). Ακόμη και για αρνητική γνώση $q = 4$ τιμών, αυτό συμβαίνει μόλις στο 0.001% των συνδυασμών. Όταν λαμβάνουμε υπόψη την άρνηση γενικευμένων τιμών, ο επιτιθέμενος καταφέρνει να βρει συνδυασμούς που εμφανίζονται μόνο σε [3-5] εγγραφές, αλλά αυτό συμβαίνει μόνον για $q = 4$ και μόλις στο 0.0058% των συνδυασμών, όπως φαίνεται στο Σχήμα 3.24(β). Και στις δύο περιπτώσεις, η πλειοψηφία των επιθέσεων αφορούν συνδυασμούς των οποίων η υποστήριξη

πέφτει ελάχιστα κάτω από το όριο k, δηλαδή στον κουβά [15-20]. Τονίζουμε ότι κανένα σενάριο επίθεσης σε αυτό το πείραμα δεν οδήγησε σε μοναδική αναγνώριση εγγραφής.

Με βάση τις παραπάνω μετρήσεις καταλήγουμε στο συμπέρασμα ότι τα πειραματικά αποτελέσματα στηρίζουν την αρχική μας υπόθεση ότι η αρνητική γνώση αδυνατεί να δώσει σε ένα κακόβουλο επιτιθέμενο ισχυρή δύναμη αναγνώρισης και ταυτοποίησης ανωνυμοποιημένων εγγραφών στην περίπτωση των αραιών πολυδιάστατων δεδομένων.

3.5 Συμπεράσματα

Σε αυτό το κεφάλαιο μελετήθηκε το πρόβλημα της ανωνυμοποίησης δεδομένων με δενδρική δομή λαμβάνοντας υπόψη την δομική γνώση του επιτιθέμενου. Εστιάσαμε σε ένα πολύ διαδεδομένο τύπο δεδομένων, ο οποίος περιλαμβάνει είτε δεδομένα με δενδρική δομή που εκφράζεται άμεσα μέσω του συντακτικού της γλώσσας, όπως λ.χ. οι εγγραφές XML, είτε δεδομένα των οποίων η δομή προκύπτει έμμεσα, όπως η περίπτωση όπου το σύνολο της προσωπική πληροφορίας ενός ατόμου είναι διασκορπισμένη σε εγγραφές διαφόρων πινάκων, οι οποίοι συνδέονται μεταξύ τους με ξένα κλειδιά σε μια βάση δεδομένων.

Προτάθηκε η εγγύηση της $k^{(m,n)}$-ανωνυμίας, μιας νέας εγγύησης ιδιωτικότητας για την αντιμετώπιση επιθέσεων που περιλαμβάνουν την μερική γνώση τόσο των τιμών των γνωρισμάτων όσο και των δομικών σχέσεων μεταξύ τους. Παρουσιάστηκε ένας νέος αλγόριθμος ανωνυμοποίησης `ACS` και μια άπληστη παραλλαγή του, ο `GCS`, οι οποίοι δημιουργούν $k^{(m,n)}$-ανώνυμα σύνολα δεδομένων εφαρμόζοντας γενικεύσεις των τιμών των γνωρισμάτων και ένα νέο δομικό μετασχηματισμό τον οποίο ονομάζουμε δομική αποσυσχέτιση. Η πειραματική αξιολόγηση της μεθόδου έδειξε ότι ο προτεινόμενος άπληστος αλγόριθμος επιτυγχάνει καλύτερη κλιμάκωση σε μεγάλα δεδομένα και ταυτόχρονα ξεπερνά τις μεθόδους που βασίζονται μόνο σε γενικεύσεις και απαλοιφές τιμών, ως προς την ποιότητα και την χρησιμότητα των τελικών ανωνυμοποιημένων δεδομένων. Σημαντικό ρόλο για την επιλογή των καλύτερων μετασχηματισμών ανωνυμοποίησης έπαιξε και ο ορισμός της νέας μετρικής $RPDa$ για την συνδυαστική αποτίμηση της απώλειας πληροφορίας τόσο των τιμών όσο και της δομής των δεδομένων.

Τα αποτελέσματα αυτής της ερευνητικής εργασίας δημοσιεύθηκαν στο διεθνές επιστημονικό περιοδικό TKDE [38]. Θεωρούμε αυτή την εργασία ως ένα πρώτο βήμα προς την ανωνυμοποίηση δεδομένων με πιο πολύπλοκη εκφραστική δομή, όπως είναι οι γράφοι δεδομένων RDF. Τέτοια δεδομένα είναι ευρέως διαδεδομένα στον Ιστό και αποτελούν το αντικείμενο μελέτης του επόμενου κεφαλαίου. Άλλες πιθανές κατευθύνσεις για μελλοντική έρευνα είναι η επέκταση της μεθόδου για πιο εκφραστικά μοντέλα επίθεσης (λ.χ. δομική γνώση ανεξάρτητη των γνωστών τιμών) και η διατύπωση αυστηρότερων εγγυήσεων που καλύπτουν την πιθανή γνώση δομικών περιορισμών και άλλων σημασιολογικών σχέσεων των δεδομένων, παρέχοντας εγγυήσεις απέναντι σε διάφορους τύπους αποκάλυψης πληροφορίας, όπως η αποκάλυψη ταυτότητας, γνωρίσματος ή ύπαρξης.

Κεφάλαιο 4

Προστασία Ιδιωτικότητας Δεδομένων Με Δομή Γράφου

Σε αυτό το κεφάλαιο παρουσιάζονται οι μέθοδοί μας για την προστασία ιδιωτικότητας σε δημοσιεύσεις δεδομένων με δομή γράφου, ως γενική περίπτωση των διασυνδεδεμένων και RDF δεδομένων. Αρχικά παρουσιάζεται το μοντέλο των δεδομένων και οι απειλές κατά τις ιδιωτικότητας από πιθανούς επιτιθέμενους που εκμεταλλεύονται και την γνώση της δομής των εγγραφών. Στη συνέχεια, προτείνεται ο βασικός αλγόριθμος ανωνυμοποίησης δεδομένων δενδρικής δομής και οι μετρικές κόστους που χρησιμοποιούνται για τον υπολογισμό της απώλειας πληροφορίας που προκαλεί η ανακωδικοποίηση. Τέλος, παρουσιάζουμε κάποιες παραλλαγές του αλγορίθμου μας με σκοπό την μείωση του χρόνου εκτέλεσης της ανωνυμοποίησης.

4.1 Κίνητρο και Συνεισφορά

Το πλαίσιο περιγραφής πόρων (Resource Description Framework - RDF) είναι ένα μοντέλο ανταλλαγής δεδομένων στον Ιστό. Παρέχει ποικίλους συντακτικούς συμβολισμούς και είναι ευρέως διαδεδομένο ως μια γενική μέθοδος για εννοιολογική μοντελοποίηση πληροφοριών από πηγές στον παγκόσμιο ιστό (web resources). Το RDF χρησιμοποιεί μοναδικά αναγνωριστικά (Uniform Resource Identifiers - URIs) για να προσδιορίσει οντότητες και τις συσχετίσεις μεταξύ τους. Τα URIs λειτουργούν και ως σύνδεσμοι, παρόμοια με τα URLs στο διαδίκτυο. Αυτή η διασυνδεδεμένη δομή δημιουργεί ένα κατευθυνόμενο γράφο με ετικέτες, όπου οι κορυφές αναπαριστούν οντότητες, κλάσεις ή τιμές και οι ακμές αναπαριστούν τις μεταξύ τους σχέσεις. Αυτή η αναπαράσταση των RDF δεδομένων ως γράφων βρίσκει εφαρμογή στην οπτικοποίηση πληροφορίας. Η χρήση ενός τέτοιου απλού μοντέλου επιτρέπει την σύνθεση δομημένων και ημιδομημένων δεδομένων και τη διανομή τους σε διάφορες εφαρμογές.

Η αξιοποίηση του μοιράσματος δεδομένων στον Ιστό έχει πολλαπλά οφέλη για τους ερευνητές, τις εταιρίες και το ευρύ κοινό. Εντούτοις, θέτει σε κίνδυνο την ιδιωτικότητα αυτών των πληροφοριών και των ατόμων με τα οποία συνδέονται αυτές οι πληροφορίες. Σε αυτό το Κεφάλαιο θα μελετήσουμε το πρόβλημα της ανωνυμοποίησης των RDF δεδομένων μοντελοποιώντας τα ως ένα κατευθυνόμενο γράφο με ετικέτες στις ακμές και στις κορυφές του.

Εξετάζουμε πως τόσο η πληροφορία των ετικετών όσο και η δομή του γράφου μπορεί να οδηγήσουν ένα κακόβουλο επιτιθέμενο στην παραβίαση της ιδιωτικότητας ατόμων των οποίων η πληροφορία βρίσκεται στα δημοσιευμένα δεδομένα.

Συγκεκριμένα, θα δείξουμε με ποιόν τρόπο ο δημοσιευμένος γράφος μπορεί να εκμεταλλευθεί από τον κακόβουλο επιτιθέμενο ώστε να αναγνωρίσει την ταυτότητα οντοτήτων που συνδέονται με πραγματικά πρόσωπα, τις οποίες καλούμε *προσωπικές οντότητες*, και ορίζουμε μια κατάλληλη εγγύηση ιδιωτικότητας ώστε να προληφθούν τέτοιες απειλές. Εισάγουμε πράξεις ανακωδικοποίησης που μπορούν να εφαρμοστούν στα RDF δεδομένα πριν τη δημοσίευσή τους ώστε να μην είναι δυνατή η παραβίαση των εγγυήσεων ιδιωτικότητας που ορίσαμε. Τέλος, προτείνουμε ένα ευρηστικό αλγόριθμο ανωνυμοποίησης χρησιμοποιεί κατάλληλα τις πράξεις ανακωδικοποίησης ώστε να ικανοποιήσει την εγγύηση ιδιωτικότητας με την μικρότερη δυνατή απώλεια πληροφορίας.

Συνοψίζοντας, η συνεισφορά του Κεφαλαίου 4 περιλαμβάνει τα παρακάτω:

- Ορίζουμε το πρόβλημα της ανωνυμοποίησης δεδομένων με δομή RDF γράφου και εξηγούμε λεπτομερώς πως η δομή δρα ως ψευδο-αναγνωριστικό.
- Επεκτείνουμε την $k^{(m,n)}$-ανωνυμία σε RDF δεδομένα και δείχνουμε την αποτελεσματικότητά της σε ρεαλιστικά σενάρια επίθεσης.
- Προτείνουμε πέντε πράξεις μετασχηματισμού των RDF οντοτήτων, ιδιοτήτων και κλάσεων ώστε να παρέχουμε μεγαλύτερη ευελιξία στη διαδικασία ανωνυμοποίησης.
- Προτείνουμε ένα νέο αλγόριθμο καθώς και μια νέα μετρική απώλειας πληροφορίας η οποία λαμβάνει υπόψη τόσο την αλλοίωση της δομής όση και της πληροφορίας των ετικετών σε κόμβους και ακμές.

4.2 Ορισμός του Προβλήματος

Στις ενότητες 4.2.1-4.2.5 επεξηγούνται οι βασικές έννοιες και δίνεται ο φορμαλισμός του προβλήματος.

4.2.1 Μοντέλο Δεδομένων

Τα δεδομένα που μελετήσαμε έχουν τη δομή ενός κατευθυνόμενου γράφου με ετικέτες στις κορυφές και στις ακμές, όπως είναι τα διασυνδεδεμένα και τα RDF δεδομένα. Μοντελοποιούμε τα δεδομένα ως εξής:

Ορισμός 4.4. *Ένα σύνολο δεδομένων RDF είναι ένας γράφος $G(V, E, \mathcal{L})$ όπου*

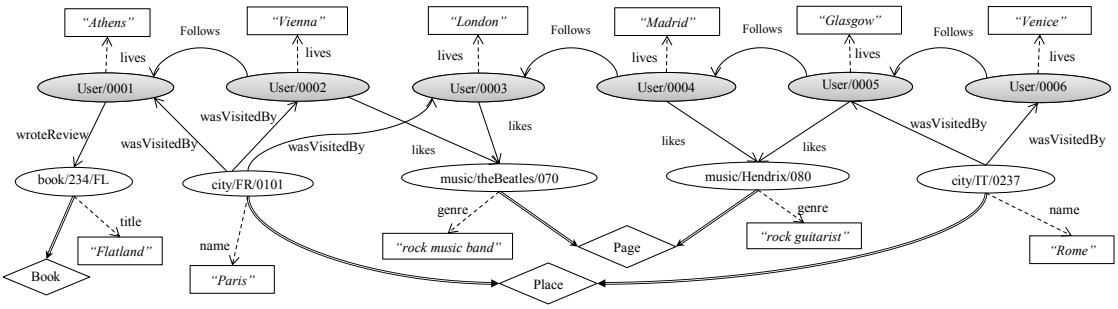
- *V είναι ένα σύνολο κορυφών, οι οποίοι αποκαλούνται και κόμβοι του γράφου. Αποτελείται από την ένωση τριών ξένων μεταξύ τους υποσυνόλων $V_E \cup V_C \cup V_V$. Το V_E είναι το σύνολο των κορυφών που αντιστοιχούν σε RDF* οντότητες (entities)*, το V_C είναι το σύνολο των κορυφών που αντιστοιχούν σε* κλάσεις (classes) *οντοτήτων και το V_V*

αποτελεί το σύνολο των κορυφών που αντιστοιχούν σε κυριολεκτικές τιμές (literals), *δηλαδή συμβολοσειρές χαρακτήρων.*

- $E \subseteq V \times V$ *είναι πεπερασμένο σύνολο ακμών, όπου καθεμία αντιστοιχεί σε ένα διατεταγμένο ζεύγος κορυφών* (v_1, v_2)*, όπου* $v_1, v_2 \in V$*. Οι ακμές αυτές αντιστοιχούν σε* ιδιότητες *των RDF δεδομένων. Μια ακμή* (v_1, v_2) *ανάμεσα σε δυο κόμβους* v_1 *και* v_2 *μπορεί να υπάρχει μόνον όταν:*
  - *και οι δύο* $v_1, v_2 \in V_E$*, είτε*
  - $v_1 \in V_E$ *και* $v_2 \in V_V$*, είτε*
  - $v_1 \in V_E$ *και* $v_2 \in V_C$*, είτε*
  - $v_1 \in V_C$ *και* $v_2 \in V_C$*.*
- $\mathcal{L} : \{V \cup E\} \rightarrow L$ *είναι μια συνάρτηση απόδοσης ετικετών η οποία αναθέτει σε κάθε κορυφή* $v \in V$ *και σε κάθε ακμή* $p \in E$ *μια ετικέτα από ένα αλφάβητο* L*. Το αλφάβητο* L *αποτελείται από την ένωση των ξένων μεταξύ τους υποσυνόλων* $L_E \cup L_C \cup L_V \cup L_R \cup L_A \cup \{type, subclass\}$*, όπου:*
  - L_E *είναι το σύνολο ετικετών οντοτήτων.* $\mathcal{L}(v) \in L_E$ *αν και μόνο αν* $v \in V_E$*.*
  - L_C *είναι το σύνολο ετικετών κλάσεων.*$\mathcal{L}(v) \in L_C$ *αν και μόνο αν* $v \in V_C$*.*
  - L_V *είναι το σύνολο ετικετών τιμών.*$\mathcal{L}(v) \in L_V$ *αν και μόνο αν* $v \in V_V$*.*
  - L_R *είναι το σύνολο ετικετών ιδιοτήτων μεταξύ οντοτήτων.* $\mathcal{L}(p) \in L_R, p = (v_1, v_2)$ *αν και μόνο αν* $v_1, v_2 \in V_E$*.*
  - L_A *είναι το σύνολο ετικετών ιδιοτήτων από οντότητα σε τιμή.* $\mathcal{L}(p) \in L_R, p = (v_1, v_2)$ *αν και μόνο αν* $v_1 \in V_E$ *και* $v_2 \in V_V$*.*
  - *Η ετικέτα ιδιοτήτων "type" είναι μια προκαθορισμένη ετικέτα που υποδηλώνει ότι μια οντότητα είναι στιγμιότυπο μιας συγκεκριμένης κλάσης. Δηλαδή* $\mathcal{L}((v_1, v_2)) =$ "$type$" *αν και μόνο αν* $v_1 \in V_E$ *ανδ* $v_2 \in V_C$*,*
  - *Η ετικέτα ιδιοτήτων "subclass" είναι μια προκαθορισμένη ετικέτα που υποδηλώνει ότι μια κλάση είναι υποκλάση μιας άλλης και προϋποθέτει μια ιεραρχία γενίκευσης [13]. Ισχύει ότι* $\mathcal{L}((v_1, v_2)) =$ "$subclass$" *αν και μόνο αν* $v_1, v_2 \in V_C$*.*

Οι ακμές $e \in V_E$ αντιπροσωπεύουν τις RDF οντότητες, οποίες χαρακτηρίζονται από μοναδικά αναγνωριστικά (IDs), τα URIs [12].Τα URIs των οντοτήτων αποτελούν το σύνολο L_E. Αντίστοιχα τα ονόματα των RDF κλάσεων είναι και αυτά URIs και ορίζουν το σύνολο LC. Τα ονόματα των RDF ιδιοτήτων ορίζουν το $L_R \cup L_A$, ενώ το σύνολο των τιμών αποτελεί το L_V.

Θεωρούμε επίσης μια ιεραρχία ιδιοτήτων [13] η οποία ορίζει επίπεδα γενίκευσης ακμών. Εντούτοις, αυτή η πληροφορία δεν αναπαρίσταται στον γράφο δεδομένων. Αντίθετα, μπορούμε να εκμεταλλευτούμε αυτή την πληροφορία για τον μετασχηματισμό των δεδομένων κατά την διαδικασία ανωνυμοποίησής τους. Δεχόμαστε ότι η ιεραρχία αυτή είναι μπορεί να είναι γνωστή στον επιτιθέμενο, είτε διότι αποτελεί γενική γνώση είτε γιατί είναι διαθέσιμη ξεχωριστά.



Σχήμα 4.1: Παράδειγμα γράφου RDF δεδομένων.

Ένα υποσύνολο των οντοτήτων $V_{E_p} \subseteq V_E$ θεωρούνται *προσωπικές οντότητες*. Αποτελούν στιγμιότυπα της ίδιας κλάσης $c_p \in V_C$ η οποία αντιστοιχεί σε πρόσωπα. Αυτέ οι οντότητες, των οποίων την ιδιωτικότητα ενδιαφερόμαστε να προστατεύσουμε, συνήθως αντιστοιχούν σε άτομα των οποίων οι πληροφορίες είναι αποθηκευμένες σε κάποιο σύνολο δεδομένων RDF. Τα μοναδικά αναγνωριστικά (URIs) των προσωπικών οντοτήτων θεωρούμε ότι δεν είναι γνωστά πριν δημοσιεύσουμε το ανωνυμοποιημένο σύνολο δεδομένων. Στοχεύουμε στην πρόληψη επιθέσεων που αποσκοπούν στην αντιστοίχιση των αναγνωριστικώνURIs προσωπικών οντοτήτων σε αληθινά άτομα στον πραγματικό κόσμο.

Τέλος, το σύνολο των κορυφών V χωρίζεται επίσης σε δυο ξένα μεταξύ τους υποσύνολα τις *εξωτερικές* V_X και τις *εσωτερικές* V_I κορυφές, όπου $V_X \cup V_I = V$. Το σύνολο των εσωτερικών κορυφών αποτελείται από νέους κόμβους οι οποίοι θα είναι διαθέσιμο μόνο στο τελικό σύνολο δεδομένων. Αντίθετα, οι *εξωτερικές* οντότητες θεωρούνται διαθέσιμες και συνεπώς γνωστές στο ευρύ κοινό πριν δημοσιεύσουμε τον ανώνυμο γράφο. Αυτό συμβαίνει όταν μέρος του γράφου περιέχει αναφορές (URIs) σε οντότητες που προϋπήρχαν διαθέσιμες στο διαδίκτυο και συμμετείχαν σε άλλα σύνολα δεδομένων. Συνεπώς, οι τιμές (literals) και η κλάση τους δεν μπορούν να μεταβληθούν. Θεωρούνται επίσης εξωτερικά και ανήκουν στο V_X. Για λόγους απλότητας θεωρούμε ότι όλες τις μη-προσωπικές ακμές είναι *εξωτερικές*. Οι προσωπικές οντότητες, οι τιμές και η κλάση τους θεωρούνται *εσωτερικές* και μπορούν να τροποποιηθούν πριν γίνει η δημοσίευση των δεδομένων. Το ίδιο ισχύει για οντότητες, ιδιότητες και κλάσεις που δημιουργούνται κατά την ανωνυμοποίηση, όπως θα εξηγήσουμε παρακάτω.

Ακμές που συνδέουν δυο εξωτερικές κορυφές και ιδιότητες από εξωτερική οντότητα σε τιμή θεωρούνται *εξωτερικές ιδιότητες* $E_X \subseteq V_X \times V_X$. Δεν μπορεί να γίνει διαγραφή τους ή τροποποίηση της ετικέτας τους. Οι υπόλοιπες ακμές που συνδέουν δυο εσωτερικές κορυφές ή μια εσωτερική μα μια εξωτερική κορυφή ή αντιστοιχούν σε ιδιότητες από εξωτερική οντότητα σε τιμή, θεωρούνται *εσωτερικές ιδιότητες* $E_I \subseteq (V_I \times V) \cup (V \times V_I)$. Αυτές μπορούν να διαγραφούν ή να τροποποιηθούν.

Παράδειγμα 4.11. *Στο Σχήμα 4.1 απεικονίζεται ένα μικρό παράδειγμα γράφου RDF για τα δεδομένα ενός κοινωνικού δικτύου. Οι κορυφές που ανήκουν στο V_E και αντιστοιχούν σε οντότητες αναπαριστώνται με οβάλ σχήμα, οι κλάσεις του V_C έχουν σχήμα ρόμβου και οι*

κόμβοι με τιμές του V_V έχουν παραλληλόγραμμο σχήμα. Οι προσωπικές οντότητες συμβολίζονται ως οβάλ με γκρι φόντο. Για την απλότητα του σχήματος και για οικονομία χώρου, η κλάση των προσωπικών οντοτήτων c_p έχει παραληφθεί. Οι διακεκομμένες ακμές συμβολίζουν ιδιότητες από οντότητα σε τιμή, ενώ οι συνεχείς ακμές αναπαριστούν ιδιότητες μεταξύ οντοτήτων. Τέλος, οι διπλές ακμές αναπαριστούν την ιδιότητα "type" και συνδέουν οντότητες με την κλάση τους. Παραδείγματος χάριν, η οβάλ κορυφή με ετικέτα "book/237/FL" είναι μια RDF οντότητα της κλάσης Βιβλίο ("Book") όπως φαίνεται από την διπλή ακμή που ξεκινά από την οντότητα αυτή. Η διακεκομμένη ακμή με ετικέτα 'title' είναι ιδιότητα που συνδέει την οντότητα με την τιμή "Flatland" η οποία είναι ο τίτλος του βιβλίου. Επίσης, η οντότητα αυτή συνδέεται με την προσωπική οντότητα "User/0001" μέσω της ιδιότητας 'ωροτεΡειεω', που δηλώνει ότι ο χρήστης "User/0001" έγραψε κριτική για το βιβλίο αυτό. Η τελευταία ιδιότητα συνδέει δυο οντότητες και συνεπώς απεικονίζεται ως συνεχής ακμή. Για απλότητα, η κλάση "Users" των προσωπικών οντοτήτων δεν απεικονίζεται στο σχήμα.

Τα RDF δεδομένα συχνά αποθηκεύονται με την μορφή *τριπλετών*. Κάθε τριπλέτα ακολουθεί την βασική δομή «Υποκείμενο Κατηγόρημα Αντικείμενο» ακολουθεί μια από τις παρακάτω μορφές:

(i) Οντότητα Ιδιότητα Τιμή

(ii) Οντότητα Ιδιότητα Οντότητα

(iii) Οντότητα *τύπος* Κλάση

(iv) Κλάση *υποκλάση* Κλάση

(v) Ιδιότητα *υποϊδιότητα* Ιδιότητα

όπου κάθε «Ιδιότητα» που ανήκει σε τριπλέτα της πρώτης μορφής είναι μια ετικέτα που παίρνει τιμές από το σύνολο L_A και αντιστοιχεί σε μια ακμή από οντότητα σε τιμή. Οι «Ιδιότητες» τριπλετών της δεύτερης μορφής είναι ετικέτες από το σύνολο L_R και αντιστοιχούν σε ακμές μεταξύ οντοτήτων. Κάθε «Οντότητα» σε καθεμία από τις τριπλέτες των τριών πρώτων μορφών είναι μια ετικέτα από το σύνολο L_E, οι πραγματικές «Τιμές» είναι ετικέτες του L_V και οι «Κλάσεις» είναι ετικέτες που προέρχονται από το L_C. Οι ιδιότητες τύπος και υποκλάση δηλώνονται με τις δεσμευμένες λέξεις type και subclass αντίστοιχα.

Οι τύποι των RDF τριπλετών μπορούν να εκφραστούν χρησιμοποιώντας το συμβολισμό του Ορισμού 4.4 ως εξής:

(i) $\mathcal{L}(v_E)$ $\mathcal{L}((v_E, v_L))$ $\mathcal{L}(v_L)$

(ii) $\mathcal{L}(v_{E1})$ $\mathcal{L}((v_{E1}, v_{E2}))$ $\mathcal{L}(v_{E2})$

(iii) $\mathcal{L}(v_E)$ $type$ $\mathcal{L}(v_C)$

(iv) $\mathcal{L}(v_{C1})$ $subclass$ $\mathcal{L}(v_{C2})$

(v) $\mathcal{L}(e_i)$ $subproperty$ $\mathcal{L}(e_j)$

Ισχύει ότι οι ετικέτες $\mathcal{L}(v_E), \mathcal{L}(v_{E1}), \mathcal{L}(v_{E2}) \in L_E$, $\mathcal{L}(v_L) \in L_V$ και $\mathcal{L}(v_C), \mathcal{L}(v_{C1}), \mathcal{L}(v_{C2}) \in L_C$ αντιστοιχούν σε κορυφές, ενώ οι ετικέτες $\mathcal{L}(v_E, v_L) \in L_A$ ανδ $\mathcal{L}(v_{E1}, v_{E2}) \in L_R$ καθώς και οι προκαθορισμένες "type" και "subclass" αντιστοιχούν σε ακμές του γράφου.

Παράδειγμα 4.12. *Ο γράφος του Σχήματος 4.1 μπορεί να εκφραστεί ως ένα σύνολο τριπλετών. Για παράδειγμα, "book/234/FL* type *Book", "book/234/FL title Flatland" και "User/0001 wroteReview book/234/FL" είναι τρεις τριπλέτες που δείχνουν την κλάση, ένα γνώρισμα και μια ιδιότητα μεταξύ οντοτήτων που σχετίζονται με την οντότητα "book/234/FL".*

Γράφος Εγγραφής μιας Οντότητας

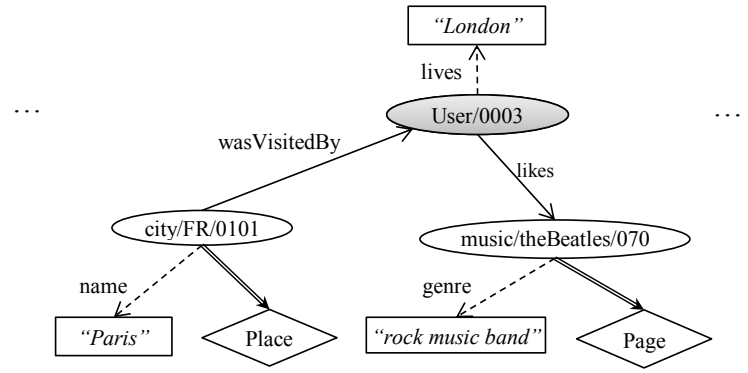
Μια εγγραφή ενός κλασσικού σχεσιακού πίνακα είναι μια πλειάδα γνωρισμάτων που αφορούν στην πληροφορία ενός ατόμου, του κατόχου της εγγραφής. Όταν τα δεδομένα δεν έχουν την αυστηρή δομή πίνακα, αλλά διασυνδέονται μεταξύ τους σε δομή γράφου G είναι δύσκολο να οριστεί κατάλληλα η έννοια της εγγραφής ενός ατόμου σε αυτά. Επιθυμούμε να συλλέξουμε όλη την πληροφορία που αφορά προσωπικά ένα άτομο και σχετίζεται με την προσωπική του οντότητα $e \in V_p$ στα δεδομένα. Έτσι, ξεχωρίζουμε έναν υπογράφο του G τον οποίο καλούμε *γράφο εγγραφής* G_e, ή απλούστερα *εγγραφή* της οντότητας e.

Ορισμός 4.5. *Ορίζουμε ως προσωπικό συσχετισμό $G_{\Pi\Sigma}$ του e όλες τις ακμές και τις κορυφές που ανήκουν στο μέγιστο συνεκτικό υπογράφο του G που προκύπτει διατρέχοντας όλα τα μονοπάτια που ξεκινούν από το e, αγνοώντας την κατεύθυνση των ακμών και σταματώντας όταν καταλήξουμε σε τιμή $l \in V_V$, είτε σε κλάση $c \in V_C$ είτε σε άλλη προσωπική οντότητα $e'_p \in V_p$.*

Τα μονοπάτια που καταλήγουν σε τιμή, δεν έχουν επέκταση. Ουσιαστικά ο παραπάνω ορισμός αποκλείει τα τμήματα του γράφου με τα οποία ο e συνδέεται έμμεσα, μέσω άλλων οντοτήτων. Παραδείγματος χάριν, είναι φίλος με κάποιον που έχει κόκκινο αυτοκίνητο. Το συγκεκριμένο όχημα δεν συνδέεται προσωπικά με τον e. Επίσης αποκλείονται τμήματα του γράφου με τα οποία ο e δεν έχει ουσιαστική σχέση. Για παράδειγμα, αν ο e συνδέεται με την σχέση «έγραψε κριτική» προς μια οντότητα τύπου «Βιβλίο», δεν σημαίνει ότι έχει σχέση και με όλες τις υπόλοιπες οντότητες τύπου «Βιβλίο» στα δεδομένα.

Ορισμός 4.6. *Γράφος εγγραφής G_e μιας προσωπικής οντότητας e ορίζεται ως ο προσωπικός του συσχετισμός στο γράφο δεδομένων G, αφαιρώντας κάθε άλλη προσωπική οντότητα, $G_e = G_{\Pi\Sigma} \setminus \{e'_p \in V_p : e'_p \neq e\}$.*

Παράδειγμα 4.13. *Επιστρέφοντας στον γράφο του Σχήματος 4.1, η εγγραφή της προσωπικής οντότητας "User/0003" απεικονίζεται στο Σχήμα 4.2. Τα μη-κατευθυνόμενα μονοπάτια που ξεκινούν από αυτήν την οντότητα, τερματίζουν είτε σε τιμές ("London", "Paris", "rock music band") είτε σε κλάσεις (Place, Page).*



Σχήμα 4.2: Γράφος Εγγραφής G_p της προσωπικής οντότητας "User/0003" από το παράδειγμα του Σχήματος 4.1.

4.2.2 Μοντέλο Επίθεσης

Θεωρούμε επιτιθέμενους οι οποίοι έχουν *μερική* γνώση του γράφου εγγραφής μιας προσωπικής οντότητας $e_p \in G$ η οποία αντιστοιχεί σε ένα πραγματικό άτομο - στόχο. Παραδείγματος χάριν μπορεί να γνωρίζει ότι ο στόχος «έχει ένα κόκκινο αυτοκίνητο». Η γνώση αυτή αντιστοιχεί σε έναν υπογράφο με δύο μονοπάτια:
$e_p \xrightarrow{has} e \xrightarrow{color} «red»$ και
$e_p \xrightarrow{has} e \xrightarrow{type} c_{car}$
όπου $e \in V_E$, είναι μια οντότητα τύπου αυτοκίνητο (car) και χρώματος κόκκινου (red), $c_{car} \in V_{E_C}$ είναι η κλάση των αυτοκινήτων, ενώ $e_p \in V_{E_p}$ είναι προσωπική οντότητα.

Ο επιτιθέμενος επιθυμεί να χρησιμοποιήσει αυτή τη γνώση για να ταυτοποιήσει αυτή την οντότητα του RDF γράφου. Συγκεκριμένα υποθέτουμε ότι ο επιτιθέμενος γνωρίζει ένα *συνεκτικό υπογράφο* του G, ο οποίος περιέχει την προσωπική οντότητα e_p καθώς και ένα υποσύνολο των άλλων οντοτήτων, ιδιοτήτων και τιμών με τις αντίστοιχες ετικέτες τους. Ο επιτιθέμενος διασχίζει τον δημοσιευμένο γράφο δεδομένων και απορρίπτει τις προσωπικές οντότητες των οποίων ο γράφος εγγραφής δεν ταιριάζει στη γνώση του.

Ο λόγος που υποθέτουμε ένα όριο στη γνώση του επιτιθέμενου, ότι δηλαδή ότι ο επιτιθέμενος έχει μερική και όχι ολική γνώση του γράφου εγγραφής, είναι γιατί αν ένας επιτιθέμενος γνωρίζει ήδη όλη την πληροφορία που σχετίζεται με ένα άτομο-στόχο στο σύνολο δεδομένων, δεν απομένει τίποτα για να ανακαλύψει. Επομένως, η προστασία από έναν τέτοιο επιτιθέμενο δεν θα είχε νόημα.

Δεν θεωρούμε την περίπτωση γνώσης των ετικετών οντοτήτων (URIs οντοτήτων), διότι είναι μοναδικά και ο επιτιθέμενος θα μπορούσε να τα χρησιμοποιήσει για να κερδίσει πλήρη πρόσβαση σε όλη την πληροφορία της οντότητας (ιδιότητες, τιμές και κλάση). Συνεπώς, η γνώση ενός συγκεκριμένου URI μιας οντότητας είναι μια ειδική περίπτωση του σεναρίου επίθεσης που μελετάμε και μπορεί να μοντελοποιηθεί ως η γνώση *όλων* των ιδιοτήτων, *όλων* των τιμών και της κλάσης αυτής της οντότητας.

Επιπλέον, ο υπογράφος που αποτελεί την γνώση του επιτιθέμενου πρέπει να είναι *συνεκτικός*, έτσι ώστε κάθε κορυφή και κάθε ακμή να βρίσκεται τουλάχιστον σε ένα μονοπάτι που

την συνδέει με την προσωπική οντότητα-στόχο. Διαφορετικά, ο επιτιθέμενος θα γνωρίζει ένα μέρος πληροφορίας όπως «κάποιο αυτοκίνητο είναι κόκκινο», αλλά δεν θα γνώριζε πως αυτό σχετίζεται με την οντότητα-στόχο. Δεν θα μπορούσε να συμπεράνει αν ο στόχος είναι ο ιδιοκτήτης ενός κόκκινου αυτοκινήτου, ή αν είχε ενοικιάσει ένα κόκκινο αυτοκίνητο, ή αν απλώς είχα γράψει μια κριτικά για ένα κόκκινο αυτοκίνητο στο διαδίκτυο. Αυτού του τύπου τα κενά πληροφορίας δεν είναι ρεαλιστικά σε πραγματικά σενάρια επίθεσης. Επίσης, δεν δίνουν επιπλέον δυνατότητα στον επιτιθέμενο να αναγνωρίσει τον στόχο αφού δεν γνωρίζει αν και πως μπορεί να σχετίζεται με μια τέτοια πληροφορία. Συνεπώς, δεν τα λαμβάνουμε υπόψη.

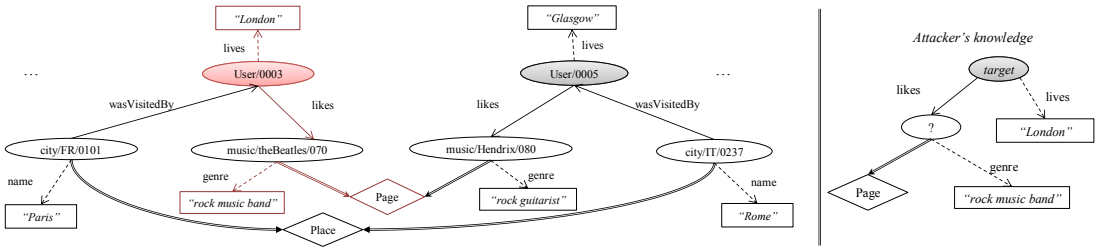
Είναι σημαντικό να σημειωθεί ότι ο κακόβουλος επιτιθέμενος δεν χρειάζεται να έχει πρότερη γνώση για την ακριβή δομή του γράφου, δηλαδή πως τα δεδομένα έχουν αποθηκευθεί. Γνωρίζει κάποια πληροφορία σχετική με τον στόχο, όπως για παράδειγμα ότι «Ο Κώστας έχει ένα κόκκινο αυτοκίνητο». Δεν γνωρίζει αν στο γράφο αυτό απεικονίζεται ως «$e_p \xrightarrow{owns} e$» ή ως «$e \xrightarrow{bought\_by} e_p$», όπου e_p η οντότητα του Κώστα και e η οντότητα του κόκκινου αυτοκινήτου. Αυτή είναι μια σχεδιαστική επιλογή που γίνεται από τον κάτοχο των δεδομένων. Η δομή της πληροφορίας «Ο Κώστας έχει ένα κόκκινο αυτοκίνητο» την οποία γνωρίζει ο επιτιθέμενος γίνεται φανερή όταν αυτός μελετήσει τον δημοσιευμένο γράφο δεδομένων.

Σύμφωνα με το παραπάνω μοντέλο επίθεσης, ο γράφος γνώσης του επιτιθέμενου αντιστοιχεί σε ένα σύνολο μονοπατιών, τα οποία μπορεί να είναι επικαλυπτόμενα, και κάθε ένα από αυτά περιέχει την προσωπική οντότητα-στόχο e_p. Τα μονοπάτια μπορεί να είναι της μορφής «$e_p \xrightarrow{p_1} e_1 \xrightarrow{p_2} e_2 \xleftarrow{p_3} e_3 ... \xrightarrow{p_n} e_n$», όπου οι ακμές με ετικέτες $p_i, (i = 1, 2, 3, ...)$ είναι ιδιότητες και οι κορυφές χωρίς γνωστή ετικέτα $e_j, (j = 1, 2, 3, ...)$ είναι οντότητες. Αυτά τα μονοπάτια αντιστοιχούν στη γνώση των ετικετών ιδιοτήτων χωρίς την γνώση των URIs των οντοτήτων, όπως περιγράψαμε πιο πάνω. Ένας επιτιθέμενος μπορεί να γνωρίζει μονοπάτια που καταλήγουν σε μια τιμή «$... \xrightarrow{p_L} l$». Σε αυτή την περίπτωση γνωρίζει τόσο την ετικέτα της από-οντότητα-προσ-τιμή ιδιότητας p_L, όσο και την τιμή l. Αν γνώριζε την τιμή χωρίς την ιδιότητα p_L, τότε ο γράφος γνώσης επιτιθέμενου δεν θα ήταν συνεκτικός. Αν γνώριζε την ετικέτα της ιδιότητας p_L χωρίς την τιμή l, δεν θα ήταν γράφος. Αντίστοιχα, το ίδιο ισχύει για μονοπάτια που καταλήγουν σε κορυφή που αντιστοιχεί σε κλάση «$... \xrightarrow{type} c_i$». Ο επιτιθέμενος γνωρίζει τόσο την ετικέτα της ιδιότητας «τύπος» όσο και την ετικέτα της κλάσης.

Σημειώνεται ότι οι ακμές ενός μονοπατιού δεν χρειάζεται να έχουν την ίδια κατεύθυνση. Η κατεύθυνση δεν επηρεάζει την γνώση του επιτιθέμενου ούτε την μέθοδο ανωνυμοποίησης που προτείνουμε. Όπως εξηγήσαμε παραπάνω, πρόκειται για μια σχεδιαστική επιλογή για την αναπαράσταση της σχέσης μεταξύ οντοτήτων. Αν παραδείγματος χάριν έστω ένα άτομο e_{pers} το οποίο δουλεύει σε μια εταιρία e_{com}. Ο κάτοχος των δεδομένων μπορεί να αναπαραστήσει αυτή την σχέση ως «$e_{pers} \xrightarrow{works} e_{com}$» είτε ως «$e_{pers} \xleftarrow{hasEmployee} e_{com}$». Ο επιτιθέμενος μπορεί να εξάγει το σχήμα των δεδομένων παρατηρώντας τον γράφο.

Συνοψίζοντας, θεωρούμε ότι η γνώση του επιτιθέμενου σχετικά με μια οντότητα-στόχο e_p είναι ένας συνεκτικός υπογράφος G_{Attack} του G που αποτελείται από:

- μια προσωπική οντότητα e_p (στόχος) άγνωστης ετικέτας,
- το πολύ m κορυφές που περιλαμβάνουν:



Σχήμα 4.3: Σενάριο επίθεσης για $m = 4$ και $n = 2$. Οι κόκκινες ακμές και κορυφές υποδεικνύουν το ταίριασμα με τη γνώση του επιτιθέμενου.

  - m_e οντότητες άγνωστης ετικέτας,
  - m_c κλάσεις με τις ετικέτες τους από το σύνολο L_C,
  - m_l τιμές με τις ετικέτες τους από το σύνολο L_V,

  όπου $m = m_e + m_c + m_l$ και $m_e, m_c, m_l \in [0, m]$.

- ένα σύνολο ακμών-ιδιοτήτων με γνωστές ετικέτες από το σύνολο $L_R \cup L_A$. Αυτές οι ακμές συνενώνουν τις παραπάνω κορυφές σχηματίζοντας ένα σύνολο μονοπατιών.

- Κάθε μονοπάτι περιλαμβάνει το e_p. Με άλλα λόγια, κάθε κορυφή συνδέεται με το e_p μέσω κάποιου μονοπατιού του G_{Attack}.

- Η μέγιστη απόσταση μεταξύ του e_p και οποιουδήποτε άλλου κόμβου σε μονοπάτι του G_{Attack} είναι n.

Στο παραπάνω μοντέλο, δεν θεωρούμε ότι ο επιτιθέμενος μπορεί να έχει αρνητική γνώση, δηλαδή να γνωρίζει ότι μια ιδιότητα, κλάση ή τιμή *δεν σχετίζεται* με την προσωπική οντότητα του στόχου. Ο λόγος πίσω από αυτή την επιλογή είναι ότι σε αραιά πολυδιάστατα δεδομένα, όπως είναι οι RDF γράφοι, αυτή η γνώση είναι δύσκολο να αποκτηθεί και σπάνια οδηγεί σε ταυτοποίηση καθώς στατιστικά θα υπάρχουν πολλές οντότητες που δεν θα σχετίζονται με μια συγκεκριμένη πληροφορία αν ο γράφος είναι αραιός. Συνεπώς, οι συνδυασμοί που περιέχουν αρνητική γνώση αναμένεται να μην είναι σπάνιοι και να μην μπορούν να οδηγήσουν σε παραβίαση ιδιωτικότητας.

Παράδειγμα 4.14. *Στο Σχήμα 4.3 φαίνεται ένα ρεαλιστικό παράδειγμα επίθεσης σύμφωνα με το παραπάνω μοντέλο. Ο επιτιθέμενος γνωρίζει ότι ο τόπος διαμονής του στόχου είναι το Λονδίνο και ότι έχει δηλώσει την προτίμησή του σε μια σελίδα κοινωνικού δικτύου που ανήκει στο είδος ροκ μουσικής. Εξετάζοντας το σχήμα του γράφου G του Σχήματος 4.1, προκύπτει ότι αυτή η πληροφορία μεταφράζεται σε μια οντότητα-στόχο που θα πρέπει να είναι συνδεδεμένη μέσω της ιδιότητας "likes" σε μια οντότητα τύπου "Page", η οποία έχει την τιμή "rockmusicband" για την ιδιότητα "genre". Ο στόχος θα πρέπει να συνδέεται επίσης μέσω της ιδιότητας "lives" με την τιμή "London". Αυτή η γνώση αποτελείται από τρία μονοπάτια:*

- $target \xrightarrow{likes} e_x \xrightarrow{type} Page$

- $target \xrightarrow{likes} e_x \xrightarrow{genre}$ *"rock music band"*
- $target \xrightarrow{lives}$ *"London"*

όπου το target συμβολίζει μια προσωπική οντότητα-στόχο και το e_x συμβολίζει μια οποιαδήποτε οντότητα άγνωστης ετικέτας. Έτσι προκύπτει ο γράφος επίθεσης G_{Attack} ο οποίος είναι ένας συνεκτικός υπογράφος του G του Σχήματος 4.1 και φαίνεται στην δεξιά πλευρά του σχήματος. Στην αριστερή πλευρά του σχήματος φαίνεται ένα τμήμα του γράφου G του Σχήματος 4.1. Οι κόκκινοι κόμβοι και ακμές υποδεικνύουν το ταίριασμα με τη γνώση του επιτιθέμενου με την εγγραφή του χρήστη User/0003 του κοινωνικού δικτύου.

4.2.3 Εγγύηση Ιδιωτικότητας

Προτείνουμε μια νέα εγγύηση ιδιωτικότητας η οποία προλαμβάνει την ταυτοποίηση ατόμων οι οποίοι σχετίζονται με προσωπικές οντότητες ενός δοσμένου συνόλου $V_{E_p} \in G$ στα δεδομένα, από επιτιθέμενους με γνώση της μορφής που περιγράφηκε και μοντελοποιήθηκε παραπάνω. Επεκτείνουμε την εγγύηση της $k^{(m,n)}$-ανωνυμίας [38] δεδομένων δενδρικής δομής, ώστε να μπορέσουμε να την εφαρμόσουμε για να αντιμετωπίσουμε το πρόβλημα της προστασίας ιδιωτικότητας σε διασυνδεδεμένα και RDF δεδομένα με δομή γράφου.

Ορισμός 4.7. *($k^{(m,n)}$-ανωνυμία γράφων) Ένα RDF σύνολο δεδομένων σε μορφή γράφου G θεωρείται$k^{(m,n)}$-ανώνυμο ως προς ένα σύνολο από προσωπικές οντότητες $V_{E_p} \subset V_E$ αν και μόνο αν οποιοσδήποτε επιτιθέμενος που έχει ως προτερη γνώση ένα συνεκτικό υπογράφο του G με μια άγνωστη προσωπική οντότητα e και m κορυφές συνολικά, και αποτελεί ένα σύνολο μονοπατιών, όπου το κάθε μονοπάτι ξεκινά από την e και έχει το πολύ n ακμές, όπως περιγράφηκε στην ενότητα 4.2.2, δεν θα μπορεί χρησιμοποιώντας αυτή τη γνώση να ταυτοποιήσει λιγότερες από k προσωπικές οντότητες του V_{E_p} στα δεδομένα.*

Η $k^{(m,n)}$-ανωνυμία γράφων καθώς και η δενδρική $k^{(m,n)}$-ανωνυμία [38] για δεδομένα δενδρικής δομής, η οποία με την σειρά της ήταν μια επέκταση της k^m-ανωνυμίας για αδόμητα δεδομένα συνόλων τιμών [68], προκύπτουν όλες από την χαλάρωση της κλασικής εγγύησης της k-ανωνυμίας [63, 66]. Η ειδοποιός διαφορά ανάμεσα σε αυτές τις χαλαρώσεις και της αυθεντικής εγγύησης είναι η ρεαλιστική παραδοχή ότι όλα τα γνωρίσματα, και στην περίπτωσή μας όλες οι ιδιότητες, τιμές και αναγνωριστικά κλάσεων που σχετίζονται με μια οντότητα-στόχο, είναι εξίσου πιθανά να λειτουργήσουν ως ψευδο-αναγνωριστικά. Η κλασσική k-ανωνυμία διαχωρίζει τα γνωρίσματα των εγγραφών σε δύο μη-επικαλυπτόμενες κατηγορίες: ευαίσθητα και ψευδο-αναγνωριστικά. Υποθέτει ότι ο επιτιθέμενος γνωρίζει το σύνολο των ψευδο-αναγνωριστικών μιας εγγραφής και μπορεί να ανακαλύψει την ευαίσθητη τιμή. Αντίθετα, στο μοντέλο μας οποιοδήποτε κομμάτι του γράφου εγγραφής μπορεί να δράσει ως ψευδο-αναγνωριστικό αν το γνωρίζει ο επιτιθέμενος, ενώ η υπόλοιπη πληροφορία της εγγραφής είναι αυτή που προσπαθεί να ανακαλύψει.

4.2.4 Πράξεις Ανακωδικοποίησης

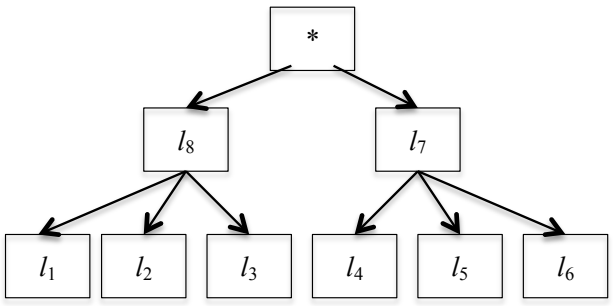
Όταν μια εγγραφή προσωπικής οντότητας αναγνωρίζεται μοναδικά από ένα πιθανό σενάριο επίθεσης, ή όταν ένας επιτιθέμενος μπορεί να ταιριάξει στη γνώση του λιγότερες από k προσωπικές οντότητες, τότε έχουμε *παραβίαση της εγγύησης ιδιωτικότητας*. Ένας RDF γράφος G που δεν ικανοποιεί την εγγύηση της $k^{(m,n)}$-ανωνυμίας γράφων μπορεί να μετασχηματιστεί κατάλληλα σε έναν ανώνυμο γράφο G^* ο οποίος ικανοποιεί την εγγύηση. Για να επιτευχθεί αυτό, ορίζουμε τις ακόλουθες πράξεις ανακωδικοποίησης των δεδομένων οι οποίες μπορούν να εφαρμοστούν στον γράφο κατά την διαδικασία ανωνυμοποίησής του.

Γενίκευση Τιμής

Υποθέτουμε την ύπαρξη μιας Ιεραρχίας Γενίκευσης Τιμών (ΙΓΤ) η οποία περιέχει κάθε ετικέτα του συνόλου L_V στα φύλλα. Κάθε τιμή του συνόλου μπορεί να απεικονιστεί σε μια πιο γενική τιμή που την περιέχει. Για παράδειγμα ο αριθμός '12' μπορεί να απεικονιστεί στο εύρος '[10, 20]' ή μια συμβολοσειρά «γάτα» μπορεί να απεικονιστεί στα «αιλουροειδή». Οι γενικευμένες τιμές μπορούν να απεικονιστούν περεταίρω σε πιο γενικές τιμές, όπως φαίνεται στο Σχήμα 4.4. Οι τιμές που ανήκουν στην ίδια ιδιότητα p δημιουργούν ιεραρχίες παρόμοιες με τα δένδρα ταξινομίας. Όλες οι ιεραρχίες τιμών από διαφορετικές ιδιότητες ανήκουν σε μια κοινή ρίζα που συμβολίζεται με '*', και σημαίνει *οποιαδήποτε* τιμή. Η γενίκευση μιας τιμής στη ρίζα ισοδυναμεί με απαλοιφή της τιμής, δηλαδή με πλήρη απόκρυψή της στα δημοσιευμένα δεδομένα.

Η αντικατάσταση μιας ή περισσοτέρων ετικετών τιμών $l_1, ..., l_n \in L_V$ από τον κοινό τους πρόγονο στην ιεραρχία l_{gen} ονομάζεται *πράξη γενίκευσης τιμών* και συμβολίζεται ως $\{l_1, ..., l_n\} \rightarrow l_{gen}$.

Στο μοντέλο δεδομένων που ορίσαμε, μόνο οι τιμές που συνδέονται άμεσα σε εσωτερικές και προσωπικές οντότητες μπορούν να γενικευθούν. Ως άμεση σύνδεση εννοούμε την σύνδεσή τους με την εσωτερική οντότητα μέσω μιας ιδιότητας, δηλαδή μιας ακμής. Ο λόγος που οι τιμές που σχετίζονται άμεσα με εξωτερικές οντότητες δεν μπορούν να γενικευθούν είναι ότι είναι ήδη διαθέσιμες δημόσια και γνωστές σε όσους γνωρίζουν τα URIs των εξωτερικών οντοτήτων. Επομένως μπορούν να ρωτήσουν όλα τα στοιχεία τους και τις τιμές που συνδέονται με αυτά. Συνεπώς, μπορεί να μην έχουμε δικαίωμα να τις αλλάξουμε ή η αλλαγή τους να προ-



Σχήμα 4.4: Ιεραρχία Γενίκευσης Τιμών.

καλεί ασυνέπειες σε σχέση με εφαρμογές που έχουν ήδη ερωτήσει πληροφορίες για αυτές τις οντότητες πριν την διαδικασία ανωνυμοποίησης. Εντούτοις, μπορούμε να δημιουργήσουμε μια νέα εσωτερική οντότητα με γενικευμένες τιμές, όπως θα εξηγήσουμε σε επόμενη παράγραφο.

Γενίκευση Κλάσης

Αξιοποιούμε την προκαθορισμένη ιδιότητα *υποκλάση* (subclass) για να δημιουργήσουμε μια ιεραρχία πάνω στις ετικέτες του συνόλου L_C. Με αυτόν τον τρόπο ορίζεται μια ιεραρχία των τύπων των οντοτήτων που βρίσκονται στον γράφο δεδομένων. Παραδείγματος χάριν, η κλάση «αυτοκίνητο» μπορεί να είναι υποκλάση μιας πιο γενικής κλάσης «όχημα». Αυτό σημαίνει ότι μια οντότητα e_i τύπου «αυτοκίνητο» θεωρείται επίσης και «όχημα». Η αντικατάσταση μιας κλάσης c ως τύπος μιας εσωτερικής οντότητας e, από μια γενική κλάση c_{gen}, τέτοια ώστε να ισχύει η σχέση $c \xrightarrow{subclass} c_{gen}$, είναι μια *πράξη γενίκευσης κλάσης* και συμβολίζεται ως $\{c\} \rightarrow c_{gen}$.

Γενίκευση Ιδιότητας

Το μοντέλο RDF δίνει την δυνατότητα δημιουργίας ιεραρχιών ιδιοτήτων. Χρησιμοποιούμε την προκαθορισμένη ιδιότητα *υποϊδιότητα* (subproperty) για να δημιουργήσουμε μια ιεραρχία γενίκευσης των ετικετών του συνόλου $L_R \cup L_A$. Παραδείγματος χάριν, οι ιδιότητες «υπάλληλος» και «διευθυντής» μπορεί να συνδέουν μια προσωπική οντότητα με την εταιρία ή τον οργανισμό όπου εργάζεται. Οι ετικέτες δείχνουν την θέση του στην εταιρία και μπορούν να θεωρηθούν ως υποϊδιότητες της πιο γενικής ιδιότητας «εργαζόμενος». Όταν μια εσωτερική οντότητα e_1 συνδέεται μέσω μιας σπανίας ιδιότητας p_1 με μια άλλη οντότητα e_2, τότε μπορούμε να αντικαταστήσουμε την ετικέτα p_1 με μια πιο γενική p_{gen} αν ισχύει ότι $p_1 \xrightarrow{subproperty} p_{gen}$. Η αντικατάσταση μιας ή περισσοτέρων ιδιοτήτων $p_1, ..., p_n$ από τον κοινό τους πρόγονο στην ιεραρχία ιδιοτήτων p_{gen} είναι μια *πράξη γενίκευσης ιδιοτήτων* και συμβολίζεται ως $\{p_1, ..., p_n\} \rightarrow p_{gen}$.

Θεωρούμε τις ιεραρχίες τιμών, κλάσεων και ιδιοτήτων να είναι είτε δημόσια διαθέσιμες ξεχωριστά, είτε να αποτελούν γενική γνώση στην οποία έχει πρόσβαση ο επιτιθέμενος. Έτσι, δεν τις απεικονίζουμε στο σχήμα του γράφου δεδομένων.

Δημιουργία Οντοτήτων και Κλάσεων

Οι εξωτερικές οντότητες θεωρούνται διαθέσιμες δημόσια και δεν μπορούν να τροποποιηθούν. Ας υποθέσουμε ότι μια εξωτερική οντότητα $e_x \in V_X$ συνδέεται άμεσα με μια σπάνια τιμή l. Αν υπάρχει σπάνιο μονοπάτι που να συνδέει μια προσωπική οντότητα e_p με το l, τότε θα μπορούσε αυτό να είναι μέρος της γνώσης ενός επιτιθέμενου ο οποίος θα καταφέρει να ανακαλύψει την ταυτότητα του e_p. Για να αντιμετωπιστούν τέτοιες απειλές μπορεί να δημιουργηθεί μια νέα οντότητα $e_i \in V_I$ η οποία θα ανήκει στον ίδιο τύπο κλάσης, θα έχει τις ίδιες ιδιότητες και τις ίδιες τιμές με την οντότητα e_x, εκτός από την τιμή l. Η τελευταία μπορεί να αντικατασταθεί από μια πιο γενική τιμή l_{gen} η οποία δεν θα είναι σπάνια.

Η ίδια διαδικασία μπορεί να ακολουθηθεί και για μια από-οντότητα-προσ-τιμή ιδιότητα με σπάνια ετικέτα p που ανήκει στην εξωτερική οντότητα e_x. Δηλαδή ένα από τα δύο άκρα

της ακμής p είναι η κορυφή e_x. Μια νέα εσωτερική οντότητα e_i δημιουργείται και το p αντικαθίσταται από την πιο γενική ετικέτα p_{gen}, όπου ισχύει ότι $p \xrightarrow{subproperty} p_{gen}$.

Επιπροσθέτως, ο «τύπος» δηλαδή η κλάση μιας εξωτερικής οντότητας e_x θα μπορούσε επίσης να συνιστά αναγνωριστική πληροφορία για μια προσωπική οντότητα e_p με την οποία σχετίζεται. Δεν μπορούμε να αλλάξουμε την ετικέτα της κλάσης c μιας εξωτερικής οντότητας $e_x \in V_X$. Ούτε και μπορούμε να αφαιρέσουμε ή να αλλάξουμε την ιδιότητα «τύπος» (type) της e_x ώστε να δείχνει προς μια πιο γενική υπερκλάση c_{gen}. Αντίθετα μπορούμε να δημιουργήσουμε μια νέα εσωτερική οντότητα e_{gen} τύπου c_{gen}, όπου $c \xrightarrow{subclass} c_{gen}$, δηλαδή η κλάση c είναι υποκλάση της c_{gen}. Η νέα οντότητα θα έχει τις ίδιες ιδιότητες και τιμές όπως η e_x, ή θα μπορεί να υποστεί και γενικεύσεις τιμών και ιδιοτήτων αν χρειαστεί. Σημειώνεται πως αν η υπερκλάση c_{gen} δεν προϋπάρχει στα δεδομένα , μπορούμε να την δημιουργήσουμε ως μια νέα εσωτερική κλάση $c_{gen} \in V_I$ και να ορίσουμε την θέση της στην ιεραρχία γενίκευσης κλάσεων χρησιμοποιώντας την ιδιότητα «υποκλάση» (subclass).

Οι ετικέτες των καινούργιων εσωτερικών κλάσεων, ιδιοτήτων και τιμών μπορούν να γενικευθούν περεταίρω αν χρειαστεί. Δεν χρειάζεται να αντικατασταθούν από άλλες νέες κλάσεις εφόσον δεν ήταν διαθέσιμες πριν την ανωνυμοποίηση των δεδομένων.

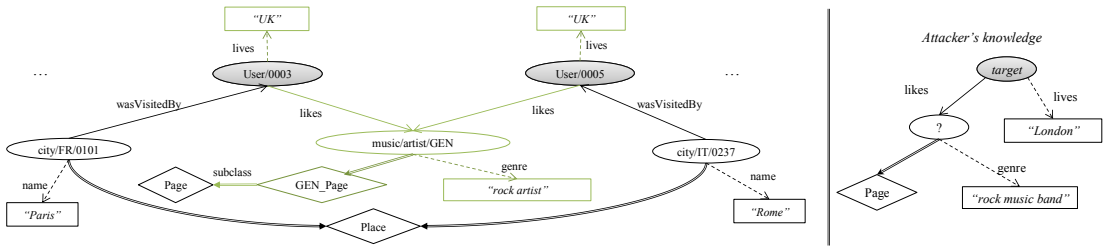
Αποσυσχέτιση Οντοτήτων

Ας υποθέσουμε ότι μια προσωπική οντότητα e_p συνδέεται μέσω μιας σπάνιας ιδιότητας p με μια άλλη οντότητα $e \in V_E$. Αν το πρόβλημα δεν μπορεί να επιλυθεί μέσω γενίκευσης ιδιοτήτων, μπορούμε να απαλείψουμε την ιδιότητα, κρύβοντας έτσι την σχέση μεταξύ των δύο οντοτήτων. Αντίστοιχα, μπορεί να γίνει απαλοιφή μιας σπάνιας ιδιότητας μεταξύ δυο εσωτερικών οντοτήτων ή μεταξύ μιας εσωτερικής και μιας εξωτερικής οντότητας. Ορίζουμε αυτή την πράξη ως *αποσυσχέτιση οντοτήτων*.

Παρόμοια με τις ιδιότητες μεταξύ οντοτήτων, μπορούμε να διαγράψουμε ιδιότητες μεταξύ μιας οντότητας και μιας τιμή αν η ιδιότητα είναι σπάνια και το πρόβλημα δεν μπορεί να επιλυθεί με γενίκευση της ετικέτας της. Σε αυτή την περίπτωση, η αντίστοιχη τιμή διαγράφεται επίσης από τα δεδομένα.

Σημειώνεται ότι αποσυσχετίζουμε επίσης μια προσωπική οντότητα e_p από μια εξωτερική οντότητα e_x, όταν έχουμε δημιουργήσει μια νέα εσωτερική οντότητα e_i για να λειτουργήσει ως γενίκευση της e_x αναφορικά με μια γενίκευση τιμής, ιδιότητας ή/και της κλάσης της e_x. Η σύνδεση της e_p με την καινούργια οντότητα e_i δεν επαρκεί, η οντότητα e_p πρέπει επίσης να αποσυσχετιστεί από την e_x με την απαλοιφή της ιδιότητας που τις συνδέει.

Όταν επιθυμούμε να αποσυσχετίσουμε μια εσωτερική οντότητα από οποιαδήποτε άλλη οντότητα, αρκεί να απαλείψουμε την ιδιότητα που τις συνδέει, δηλαδή να αφαιρέσουμε την ακμή που συνδέει τις κορυφές που αντιστοιχούν στις δύο οντότητες. Εντούτοις, όταν η σπάνια ιδιότητα είναι μια ακμή μεταξύ δυο εξωτερικών οντοτήτων, δεν μπορούμε να την γενικεύσουμε ούτε να την απαλείψουμε. Ας θεωρήσουμε το μονοπάτι «$e_p \xrightarrow{p_1} e_1 ... \xrightarrow{p_j} e_j \xrightarrow{p_{j+1}} e_{j+1} ... \xrightarrow{p_{x1}} e_{x1} \xrightarrow{p_{x2}} e_{x2}$» όπου η e_p είναι μια προσωπική οντότητα, τα $e_{x1} kai e_{x2}$είναι εξωτερικές οντότητες και οι μεταξύ τους ιδιότητα p_{x2} συνιστά πιθανή πληροφορία αναγνώρισης για την e_p. Ας



Σχήμα 4.5: Γενίκευση των οντοτήτων Music/theBeatles/070 και Music/Hendrix/080.

υποθέσουμε ότι το e_j είναι η τελευταία *εσωτερική* οντότητας στο μονοπάτι. Επιλέγουμε να αποσυσχετίσουμε την e_j από την εξωτερική οντότητα e_{j+1} αφαιρώντας την ιδιότητα p_{j+1}. Για να επιτύχουμε μικρότερη απώλεια πληροφορίας, μπορούμε να δημιουργήσουμε νέες εσωτερικές οντότητες οι οποίες θα αντικαταστήσουν το υπόλοιπο μονοπάτι «$e'_{j+1}... \xrightarrow{p_{x1}} e'_{x1}$», χωρίς την τελευταία ιδιότητα p_{x2}. Συνδέουμε την οντότητα e_j στην e'_{j+1} μέσω μιας ιδιότητας με την ίδια ετικέτα όπως η p_{j+1}. Οι νέες οντότητες έχουν τις ίδιες τιμές, ιδιότητες και κλάσεις όπως οι αυθεντικές εξωτερικές οντότητες, εκτός από την ιδιότητα p_{x2}.

Αν υπάρχει κατάλληλη γενίκευση p_{g2} της ιδιότητας p_{x2}, μπορούμε να συνδέσουμε την νέα εξωτερική οντότητα e'_{x1} με την εξωτερική οντότητα e_{x2} μέσω της p_{g2}. Το νέο μονοπάτι που σχηματίζεται είναι το «$e_p \xrightarrow{p_1} e_1... \xrightarrow{p_j} e_j \xrightarrow{p'_{j+1}} e'_{j+1}... \xrightarrow{p'_{x1}} e'_{x1} \xrightarrow{p_{g2}} e_{x2}$».

Παράδειγμα 4.15. *Επιστρέφοντας στο σενάριο επίθεσης του Σχήματος 4.3, μπορούμε να παρατηρήσουμε ότι ο γράφος γνώσης του επιτιθέμενου ταιριάζει μόνο στην προσωπική οντότητα "User/0003". Μπορούμε να γενικεύσουμε τις τιμές του τόπου διαμονής έτσι ώστε το Λονδίνο και η Γλασκόβη να αντικατασταθούν από το "UK" (Ηνωμένο Βασίλειο). Η πράξη γενίκευσης συμβολίζεται ως* $\{London, Glasgow\} \rightarrow UK$. *Έτσι το δεξί μονοπάτι στον γράφο γνώσης του επιτιθέμενου δεν μπορεί να χρησιμοποιηθεί ώστε να αποκλειστεί ο χρήστης "User/0005", αλλά το αριστερό υποδένδρο παραμένει μοναδικό για τον "User/0003" στον γράφο. Μπορούμε να δημιουργήσουμε μια νέα εσωτερική οντότητα "Music/artist/GEN" η οποία είναι «τύπου» "GEN-Page", όπου είναι υπερκλάση του Page, δηλαδή* $\{Page\} \xrightarrow{subclass} GEN\text{-}Page$. *Έχει επίσης την ιδιότητα "genre" προς μια τιμή "rock artist". Οι ετικέτα αυτές αποτελεί γενίκευση των ετικετών των τιμών "rock music band" και "rock guitarist". Μπορούμε τώρα να αποσυσχετίσουμε τον χρήστη "User/0003" από την οντότητα "Music/theBeatles/070" διαγράφοντας την ιδιότητα "likes". Προσθέτουμε μια νέα ιδιότητα με ετικέτα "likes" η οποία συνδέει τον "User/0003" με την νέα οντότητα "Music/artist/GEN". Αποσυσχετίζοντας επίσης τον χρήστη "User/0005" από την οντότητα "Music/Hendrix/080" και συνδέοντάς τον με την "Music/Music/artist/GEN", οι δυο προσωπικές οντότητες "User/0003" και "User/0005" γίνονται πανομοιότυπες ως προς την γνώση του επιτιθέμενου. Το παράδειγμα αυτό απεικονίζεται στο Σχήμα 4.5.*

4.2.5 Μετρικές Αποτίμησης Απώλειας Πληροφορίας

Οι πράξεις ανακωδικοποίησης αλλοιώνουν την ποιότητα των δεδομένων καθώς οδηγούν σε απώλεια πληροφορίας. Η αποτίμηση αυτής της απώλειας πρέπει να λαμβάνει υπόψη τόσο τις γενικεύσεις τιμών και ετικετών όσο και τις δομικές αλλοιώσεις του γράφου δεδομένων. Ο υπολογισμός της απώλειας πληροφορίας γίνεται μελετώντας πόσο έχουν επηρεαστεί οι γράφοι εγγραφών κάθε προσωπικής οντότητας από τους μετασχηματισμούς που γίνονται κατά την ανωνυμοποίηση. Αυτό το κόστος της ανωνυμοποίησης σε κάθε εγγραφή είναι το άθροισμα του επιμέρους κόστους καθενός από τα μακρύτερα μονοπάτια, δηλαδή τα μονοπάτια που ξεκινούν από την προσωπική οντότητα και καταλήγουν σε τιμή ή σε κλάση. Λαμβάνουμε επίσης υπόψη τις *διαπροσωπικές ιδιότητες* οι οποίες οδηγούν σε μια άλλη προσωπική οντότητα, όπως η ακμή που δηλώνει μια σχέση «φιλίας» σε ένα κοινωνικό δίκτυο.

Κάθε μονοπάτι περιέχει πληροφορία ετικετών και δομής. Ας θεωρήσουμε το μονοπάτι «$e_p \xrightarrow{p_1} l_1$», όπου το l_1 είναι μια τιμή από την ιεραρχία τιμών του Σχήματος 4.4 και ισχύει ότι $\{l_1, l_2, l_3\} \rightarrow l_8$. Ας υποθέσουμε ότι η p_1 είναι μια ετικέτα ιδιότητας η οποία επίσης ανήκει σε μια ιεραρχία ιδιοτήτων έτσι ώστε να ισχύει ότι $\{p_1, p_2, p_3, p_4\} \rightarrow p_9$ και $\{p_5, p_6, p_7, p_8\} \rightarrow p_{10}$. Υπάρχουν τρεις πιθανές γενικεύσεις του αρχικού μονοπατιού: (α) $e_p \xrightarrow{p_8} l_1$, (β) $e_p \xrightarrow{p_1} l_9$ και (γ) $e_p \xrightarrow{p_8} l_9$. Το αρχικό μονοπάτι φέρει λεπτομερή πληροφορία, τα μονοπάτια (α) και (β) φέρουν πιο γενική πληροφορία, ενώ το μονοπάτι (γ) περιέχει την λιγότερη πληροφορία από όλα επειδή οι τιμές του είναι πιο γενικές. Η τιμή l_1 είναι μία από τις έξι πιθανές τιμές του αρχικού πεδίου τιμών των δεδομένων, ενώ η τιμή l_8 είναι μόλις μία από τις δύο πιθανές τιμές του επιπέδου γενίκευσης 1.

Ορισμός 4.8. *Ορίζουμε ως* πεδίο τιμών επιπέδου γενίκευσης *μιας ετικέτας τιμής v και το συμβολίζουμε ως $o(v)$, το πλήθος των τιμών που βρίσκονται στο ίδιο επίπεδο με την v στο δένδρο ιεραρχίας γενίκευσης της ετικέτας. Το μέγεθος του $o(v)$ είναι μια ένδειξη του πόσο γενική είναι η ετικέτα v και πόση χρησιμότητα προσδίδει η παρουσία της στο μονοπάτι των δεδομένων όπου εμφανίζεται. Αντίστοιχα ορίζεται το πεδίο τιμών επιπέδου γενίκευσης $o(p)$ για μια ετικέτα ιδιότητας p που ανήκει σε μια δοσμένη ιεραρχία ιδιοτήτων και το πεδίο τιμών επιπέδου γενίκευσης $o(c)$ για μια ετικέτα κλάσης c η οποία ανήκει σε μια ιεραρχία κλάσεων.*

Η χρησιμότητα της πληροφορίας ενός μονοπατιού που περιέχει από διάφορες ιδιότητες με ετικέτες $p_1, p_2 ... p_n$, και που καταλήγει σε μια τιμή με ετικέτα v ή σε μια κλάση με ετικέτα c θα είναι ανάλογη του μεγέθους $o(p_1)...o(p_n) \cdot o(v)$ ή του $o(p_1)...o(p_n) \cdot o(c)$ αντιστοίχως. Η χρησιμότητα της πληροφορίας μιας διαπροσωπικής ιδιότητας με ετικέτα p θα είναι ανάλογη του $o(p)$. Διαισθητικά, όσο μεγαλύτερος ο βαθμός λεπτομέρειας της πληροφορίας ενός μονοπατιού, τόσο μεγαλύτερη η χρησιμότητά της και τόσο μικρότερη η απώλεια πληροφορίας που έχει υποστεί. Ισοδύναμα, η απώλεια πληροφορίας θα πρέπει να είναι αντιστρόφως ανάλογη των προαναφερθέντων ποσοτήτων.

Επιπλέον, όσο μεγαλύτερο είναι το μήκος ενός μονοπατιού, τόσο καλύτερα έχει διατηρήσει την αρχική του δομή. Η αποσυσχέτιση οντοτήτων μπορεί μόνο να μειώσει το μήκος των μονοπατιών Επομένως, το μήκος του μονοπατιού είναι ένας ακόμα δείκτης της χρησιμότητάς του. Στις παραπάνω περιπτώσεις το μήκος ενός μονοπατιού που καταλήγει σε μια τιμή v

είναι το βάθος του κόμβου που αντιστοιχεί στην τιμή $d(v)$. Ομοίως, για ένα μονοπάτι που καταλήγει σε μια κλάση c, το μήκος του ισούται με το βάθος $d(c)$, ενώ για τις διαπροσωπικές ιδιότητες το μήκος είναι ίσο με 1. Τα αντίστοιχα κόστη των μονοπατιών αυτών σε απώλεια πληροφορίας θα είναι $1/o(p_1)...o(p_n) \cdot o(v) \cdot d(v)$ για την τιμή, $1/o(p_1)...o(p_n) \cdot o(c) \cdot d(c)$ για την κλάση και $1/o(p)$ για την διαπροσωπική οντότητα.

Θεωρώντας για παράδειγμα το μονοπάτι «$e_p \xrightarrow{p_1} l_1$», το κόστος σε απώλει πληροφορίας για την γενίκευση (α) είναι $1/(2 \cdot 3 \cdot 1) = 0.17$, ενώ η (β) δίνει κόστος $1/(4 \cdot 2 \cdot 1) = 0.125$ και η περίπτωση (γ) έχει κόστος $1/(2 \cdot 2 \cdot 1) = 0.25$.

Η απώλεια πληροφορίας του γράφου εγγραφής G_e μιας προσωπικής οντότητας e είναι ο μέσος όρος του επιμέρους κόστους όλων των μονοπατιών που καταλήγουν σε τιμές, των μονοπατιών που καταλήγουν σε κλάσεις και των διαπροσωπικών του ιδιοτήτων:

$$IL(e) = \frac{1}{lvs}\left(\sum_{v:literal} \frac{1}{o(p_1)...o(p_n) \cdot o(v) \cdot d(v)} + \sum_{c:class} \frac{1}{o(p_1)...o(p_n) \cdot o(c) \cdot d(c)} + \sum_{p:interpers.} \frac{1}{o(p)}\right)$$

όπου lvs είναι το πλήθος των ακμών που αντιστοιχούν σε τιμή, κλάση ή άλλη προσωπική οντότητα στην αρχική εγγραφή, δηλαδή το πλήθος των μονοπατιών.

Η συνολική απώλεια πληροφορίας των δεδομένων υπολογίζεται ως ο μέσος όρος του κόστους $IL(e)$ όλων των εγγραφών προσωπικών οντοτήτων $e \in V_p$ των δεδομένων. Το αποτέλεσμα είναι ένας δείκτης της αλλοίωσης των δεδομένων που επιφέρει η διαδικασία της ανωνυμοποίησης.

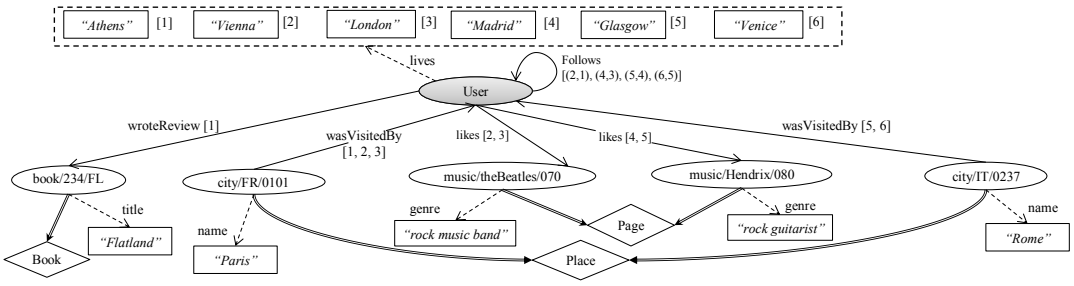
4.3 Αλγόριθμος Ανωνυμοποίησης

Προτείνουμε ένα αλγόριθμο τοπικής ανακωδικοποίησης για την ανωνυμοπίηση RDF γράφων, έτσι ώστε να ικανοποιείται η εγγύηση της $k^{(m,n)}$-ανωνυμίας γράφων, ενώ η απώλεια πληροφορίας θα διατηρείται σε χαμηλά επίπεδα.

4.3.1 Γράφος Σύνοψης

Ορίζουμε μια δομή για να κρατάμε συμπιεσμένο τον γράφο δεδομένων, την οποία καλούμε *Γράφο Σύνοψης*. Δημιουργείται με την απεικόνιση όλων των προσωπικών οντοτήτων την μία πάνω στην άλλη, δημιουργώντας έτσι ένα μοναδικό κόμβο που συμβολίζει όλα τα άτομα και τον οποίο καλούμε *ρίζα*. Σε κάθε προσωπική οντότητα ανατίθεται ένας ψευδο-κωδικός αριθμός ID. Οι ακμές που αντιπροσωπεύουν ιδιότητες μεταξύ οντοτήτων διατηρούν μια λίστα από IDs. Οι ακμές που συνδέουν προσωπικές οντότητες με μια άλλη οντότητα e απεικονίζονται σε μία νέα ακμή από την ρίζα προς την e. Οι κωδικοί (IDs) αυτών των προσωπικών οντοτήτων διατηρούνται σε μια λίστα που ανήκει σε αυτή τη νέα ακμή.

Οι ιδιότητες από οντότητα προς τιμή δεν διατηρούν λίστα από IDs. Αντίθετα επιλέγουμε να διατηρήσουμε αυτή την πληροφορία στις κορυφές που αντιστοιχούν στις τιμές, εφόσον ένας επιτιθέμενος μπορεί να γνωρίζει ένα ζεύγος «ιδιότητασ-τιμής» και όχι μόνο το ένα από τα δύο.



Σχήμα 4.6: Γράφος Σύνοψης για το παράδειγμα του Σχήματος 4.1.

Οι διαπροσωπικές ιδιότητες που φέρουν την ίδια ετικέτα, όπως λ.χ. η «φιλία» στα κοινωνικά δίκτυα, απεικονίζονται σε μια ακμή από και προς την ρίζα. Σε αυτή την ακμή διατηρούμε ζεύγη από IDs που αντιστοιχούν στις προσωπικές οντότητες που συνδέονται μέσω αυτού του τύπου την ιδιότητα.

Τέλος, οι τιμές που ανήκουν στην ίδια ιδιότητα μπορούν να ομαδοποιηθούν σε ένα κόμβο που περιέχει ένα πίνακα με τις τιμές αυτές. Κάθε τιμή του πίνακα διατηρεί και την λίστα με τα IDs των προσωπικών οντοτήτων με τις οποίες σχετίζεται.

Ο γράφος σύνοψης για το παράδειγμα των RDF δεδομένων του Σχήματος 4.1 απεικονίζεται στο Σχήμα 4.6.

4.3.2 Αλγόριθμος

Προτείνουμε ένα αλγόριθμο τοπικής ανακωδικοποίησης που ελέγχει όλους τους πιθανούς συνδυασμούς μονοπατιών που μπορεί να αποτελούν γνώση ενός κακόβουλου επιτιθέμενου, με βάση το μοντέλο επίθεσης που περιγράφηκε στην Ενότητα 4.2.2 και παράγει ένα $k^{(m,n)}$-ανώνυμο RDF γράφο.

Δοσμένων των παραμέτρων m και n που περιορίζουν την γνώση του επιτιθέμενου, υπάρχει μεγάλο πλήθος πιθανών συνδυασμών από μονοπάτια με διαφορετικά μήκη, ακμές και κορυφές. Παραδείγματος χάριν, ας υποθέσουμε ότι ο επιτιθέμενος μπορεί να γνωρίζει το πολύ $m = 2$ κορυφές και η μέγιστη απόστασή τους από την οντότητα-στόχο e_P είναι $n = 2$. Η πιθανές περιπτώσεις για την δομή της γνώσης του επιτιθέμενου είναι:

(i) 1 μονοπάτι με μια κορυφή που συνδέεται άμεσα με τον στόχο με μια ακμή $e_P \xrightarrow{p_1} e_1$,

(ii) 2 μονοπάτια που το καθένα ξεκινά από τον e_P και περιέχει μια ακμή και μια ακόμα κορυφή $e_P \xrightarrow{p_1} e_1$ και $e_P \xrightarrow{p_2} e_2$,

(iii) 1 ακυκλικό μονοπάτι που αποτελείται από τον e_P δυο ακμές και δυο επιπλέον κορυφές (σε σειρά) $e_P \xrightarrow{p_1} e_1 \xrightarrow{p_2} e_2$,

(iv) 1 κυκλικό μονοπατι που αποτελείται από τα μέρη $e_P \xrightarrow{p_1} e_1 \xrightarrow{p_2} e_2$ και $e_P \xrightarrow{p_3} e_2$.

Αρχικά θεωρούμε τους απλούστερους συνδυασμούς γνώσης που περιέχουν 1 ακμή και 1 κορυφή πέρα από τον στόχο και ελέγχουμε αν είναι σπάνιοι. Εν συνεχεία, γίνεται έλεγχος

Αλγόριθμος 5 *$k^{(m,n)}$-Ανωνυμοποίηση*

Require: G: original RDF graph
Ensure: G^*: $k^{(m,n)}$-anonymous RDF graph.
1: Construct synopsis graph S_G from G.
2: **for** $m_i = 1$ to m **do**
3: **for** $n_j = 1$ to n **do**
4: **for** every combination cmb of possible knowledge of m_i nodes and n_j maximum path length **do**
5: $S_G = checkCombination(cmb, S_G)$;
6: reconstruct G^* from S_G
7: **return** G^*.

για ολοένα πιο πολύπλοκους συνδυασμούς γνώσης προσθέτοντας μια ακμή κάθε φορά έως το μέγιστο όριο m ακμών και n κορυφών ανά μονοπάτι, όπως φαίνεται στον Αλγόριθμο 5.

Δοσμένου ενός συνόλου μονοπατιών $cmb = \{P_1, P_2, ..., P_i\}$ τα οποία συνιστούν πιθανό συνδυασμό γνώσης του επιτιθέμενου, μπορούμε να ελέγξουμε αν είναι ευάλωτος υπολογίζοντας αν υπάρχουν λιγότερες από k προσωπικές οντότητες που σχετίζονται με αυτά τα μονοπάτια. Ο έλεγχος αυτός πραγματοποιείται συνδυάζοντας τις λίστες των IDs ιδιοτήτων και τιμών του γράφου σύνοψης.

Η συνάρτηση *checkCombination*() καλείται στην γραμμή 5. Ο ψευδοκώδικάς της παρουσιάζεται στον Αλγόριθμο 6. Για κάθε μονοπάτι $P_i \in cmb$ βρίσκουμε όλα τα μονοπάτια στο γράφο σύνοψης που ταιριάζουν σε αυτό. Εν συνεχεία, υπολογίζουμε την *ένωση συνόλων* όλων των λιστών των IDs αυτών των μονοπατιών. Με αυτόν τον τρόπο βρίσκουμε όλες τις προσωπικές οντότητες των οποίων η εγγραφή περιέχει τουλάχιστον ένα μονοπάτι που ταιριάζουν σε τουλάχιστον ένα από τα $P_i \in cmb$. Παραδείγματος χάριν, αν θεωρήσουμε το μονοπάτι $P_i =$ «User $\xrightarrow{likes}$ e» το οποίο ταιριάζει σε δυο μονοπάτια στο γράφο σύνοψης: « User$\xrightarrow{likes[2,3]}$ Music/theBeatles/070 » και « User$\xrightarrow{likes[4,5]}$Music/Hendrix/080 », όπου οι αριθμοί μέσα στις αγκύλες αντιστοιχούν στις λίστες των IDs. Αυτό σημαίνει ότι οι προσωπικές οντότητες $\{2,3\} \cup \{3,4\} = \{2,3,4,5\}$ ταιριάζουν σε αυτό το μονοπάτι γνώσης.

Οι γραμμές 6-13 του Αλγορίθμου 6 υπολογίζουν την ένωση των λιστών των IDs, ανάλογα με τον τύπο του μονοπατιού. Ξεκινώντας από την προσωπική οντότητα ρίζα, αν στο μονοπάτι ακολουθεί μια ιδιότητα μεταξύ οντοτήτων τότε χρησιμοποιούμε τη λίστα IDs αυτής της ιδιότητας, όπως φαίνεται στις γραμμές 6-7. Αν το μονοπάτι αποτελείται από μια ιδιότητα προς τιμή και μια τιμή, τότε χρησιμοποιούμε την λίστα IDs της τιμής, όπως φαίνεται στις γραμμές 8-9. Τέλος, στις γραμμές 10-13 προσθέτουμε όλα τα IDs που εμφανίζονται στη λίστα ζευγών IDs, αν το μονοπάτι περιέχει διαπροσωπική ιδιότητα.

Έχοντας υπολογίσει τις λίστες των IDs κάθε μονοπατιού στη γραμμή 14 υπολογίζουμε την *τομή* αυτών των συνόλων, η οποία περιέχει μόνο εκείνες τις προσωπικές οντότητες που ο γράφος εγγραφής τους ταιριάζει σε *όλα* τα μονοπάτια του συνδυασμού γνώσης cmb. Αν η τελική λίστα από IDs που υπολογίσαμε δεν είναι άδεια και περιέχει λιγότερα από k IDs,

Αλγόριθμος 6 checkCombination

Require: S_G: RDF synopsis graph, *cmb*: knowledge combination
Ensure: S'_G: generalized RDF synopsis graph.

```
1: cList = all Ids {initialization}
2: S'_G = S_G
3: for every path i in cmb do
4:    pList = ∅
5:    for all paths in S_G that match i do
6:       if path starts with an inter-entity property p then
7:          pList = pList ∪ p.Ids
8:       else if path starts with an entity-to-attribute property, with the literal v then
9:          pList = pList ∪ v.Ids
10:      else if path starts with an interpersonal property p then
11:         for every p.IdPair in p's list of Id pairs do
12:            pList = plist ∪ {p.IdPair.first}
13:            pList = plist ∪ {p.IdPair.second}
14:      cList = cList ∩ pList
15: if 0 < |clist| < k then
16:    S'_G = fixCombination(cmb, S_G) {privacy breach}
17: return S'_G
```

τότε αυτός ο συνδυασμός είναι *ευάλωτος* και η γνώση του από έναν επιτιθέμενο μπορεί να οδηγήσει σε παραβίαση της εγγύησης ιδιωτικότητας που θέσαμε. Διαφορετικά, αν υπάρχουν 0 ή $\geq k$ IDs, είναι *ασφαλής* και δεν χρειάζεται τροποποιήσεις. Κάθε φορά που εντοπίζεται ένας ευάλωτος συνδυασμός, καλείται η συνάρτηση $fixCombination()$ στην γραμμή 16 ώστε να γίνουν οι κατάλληλες γενικεύσεις ή/και απαλοιφές στοιχείων του γράφου και να μπορεί να δημοσιευθεί με ασφάλεια.

Ο ψευδοκώδικας της $fixCombination()$ παρουσιάζεται στον Αλγόριθμο 7. Εξετάζουμε όλες τις πιθανές πράξεις ανακωδικοποίησης που παρουσιάζονται στην Ενότητα 4.2.4, ανάλογα με τον τύπο του μονοπατιού. Οι γραμμές 3-8 εξετάζουν την γενίκευση μιας τιμής όπως περιγράφηκε στην Ενότητα 4.2.4. Σημειώνεται ότι η *απαλοιφή* τιμής ισοδυναμεί με γενίκευση στην ρίζα της ιεραρχίας γενίκευσης τιμών "*". Υπολογίζεται το κόστος απώλειας πληροφορίας αυτής της επιλογής στη γραμμή XX. Η εύρεση μιας «κατάλληλης» γενίκευσης που λύνει το πρόβλημα είναι μια γενίκευση που θα κάνει το μονοπάτι αρκετά συχνό, με εμφάνιση (support) σε τουλάχιστον k εγγραφές προσωπικών οντοτήτων. Αν δεν μπορεί να βρεθεί κατάλληλη γενίκευση, τότε η τιμή απαλείφεται.

Στις γραμμές 9-19 εξετάζεται η περίπτωση όπου το μονοπάτι τερματίζει σε κορυφή κλάσης c. Σε αυτό το σενάριο χρειάζονται η γενίκευση κλάσης, η δημιουργία οντότητας και η αποσυσχέτιση οντοτήτων. Αν υπάρχει κατάλληλη γενίκευση κλάσης που να μπορεί να αυξήσει το support του συνδυασμού, τότε δημιουργούνται μια νέα οντότητα e_g και μια νέα υπερκλάση C_g, έτσι ώστε $e_g \xrightarrow{type} C_g$. Η νέα κλάση είναι υπερκλάση της c. Η τελευταία οντότητα e του

μονοπατιού, που είναι τύπου κλάσης c, αποσυσχετίζεται από το μονοπάτι και όλες οι ιδιότητες και οι τιμές της αντιγράφονται στην e_g. Η ίδια λογική εφαρμόζεται στις γραμμές 20-28, όπου εξετάζουμε ένα μονοπάτι, μήκους μεγαλύτερου του ενός, που καταλήγει σε κόμβο τιμής.

Οι γραμμές 29-33 αφορούν στην περίπτωση των διαπροσωπικών ιδιοτήτων. Πρώτα εξετάζεται η γενίκευση της ιδιότητας (βλ. Ενότητα 4.2.4) και στη συνέχεια η αποσυσχέτιση των δυο προσωπικών οντοτήτων (βλ. Ενότητα 4.2.4) με την απαλοιφή της ιδιότητας.

Εν τέλει, ο αλγόριθμος επιλέγει εκείνες τις πράξεις ανακωδικοποίησης που καθιστούν τον συνδυασμό ασφαλή, δηλαδή εξασφαλίζουν ότι θα εμφανίζεται σε 0 ή σε $\geq k$ εγγραφές προσωπικών οντοτήτων, με την λιγότερη δυνατή απώλεια πληροφορίας (γραμμές 30-31).

Ιδιότητα 4.3. *Ο Αλγόριθμος 5 θα παράγει πάντα ένα $k^{(m,n)}$-ανώνυμο Γράφο.*

Απόδειξη. Όπως φαίνεται στον ψευδοκώδικα του Αλγορίθμου 5, ξεκινάμε να ελέγχουμε όλους τους πιθανούς συνδυασμούς που αποτελούνται από 1 κορυφή και προσθετικά ελέγχουμε ολοένα και μεγαλύτερους συνδυασμούς γνώσης, έως m ακμές. Σε κάθε ευάλωτο συνδυασμό γνώσης, ακόμα και αν δεν υπάρχουν γενικεύσεις που να λύνουν το πρόβλημα, η απαλοιφή ενός κόμβου από ένα μονοπάτι (π.χ. με την απαλοιφή μιας ιδιότητας από οντότητα προς τιμή) πάντα θα οδηγεί σε ασφαλή λύση. Ο λόγος είναι ότι το μέγεθος του συνδυασμού γίνεται $m-1$ ακμές, το οποίο είχε εξεταστεί στο προηγούμενο βήμα του αλγορίθμου. Συνεπώς, η μέθοδός μας πάντα θα καταλήγει σε γράφο που ικανοποιεί την $k^{(m,n)}$-ανωνυμία. □

4.4 Συμπεράσματα

Το κεφάλαιο αυτό εστιάζει στην προστασία της ιδιωτικότητας δεδομένων με δομή γράφου και δίνει έμφαση στις συλλογές RDF δεδομένων, λόγω της διάδοσής τους στον Ιστό. Μοντελοποιήθηκαν τα σενάρια επίθεσης απέναντι σε RDF οντότητες που αντιστοιχούν σε πραγματικά άτομα και προτάθηκαν πράξεις ανακωδικοποίησης συμβατές με το μοντέλο των δεδομένων. Επεκτάθηκε η εγγύηση ιδιωτικότητας για δενδρικά δεδομένα που είχε προταθεί στο προηγούμενο κεφάλαιο, ώστε να καλύψει τις ιδιαιτερότητες της δομής των RDF γράφων και αναπτύχθηκε ένας νέος αλγόριθμος ανωνυμοποίησης που ικανοποιεί την εγγύηση περιορίζοντας την απώλεια πληροφορίας.

Από όσο γνωρίζουμε, αυτή είναι η πρώτη εργασία που προτείνει μια ολιστική λύση για την ανωνυμοποίηση RDF γράφων, λαμβάνοντας υπόψη τις τιμές και τις ετικέτες ακμών και κορυφών καθώς και την δομή των RDF εγγραφών, δηλαδή των υπογράφων που περιέχουν πληροφορία που αφορά σε μια προσωπική οντότητα. Η έρευνα αυτή είναι σε εξέλιξη και πραγματοποιούνται πειράματα σε πραγματικά δεδομένα, τα αποτελέσματα των οποίων θα υποβληθούν προς δημοσίευση σε διεθνές περιοδικό.

Αλγόριθμος 7 fixCombination

Require: S_G: RDF synopsis graph, $sPaths$: vulnerable combination of paths in S_G
Ensure: S'_G: generalized RDF synopsis graph.

for every path i in cmb **do**
  if path starts with an entity-to-attribute property, with the literal v **then**
    if an appropriate generalization V of v exists **then**
      cost_i = cost of Generalizing $\{v\} \rightarrow V$
    else
      cost_i = cost of Suppressing v
  else if path starts with an inter-entity property **then**
    if path ends in a class vertex c **then**
      if There exists a solution by generalizing c_g to C_g **then**
        if C_g does not exist **then**
          Create a new class C_g
          Declare that $c \xrightarrow{subclass} C_g$.
        Create a new entity e_g of class C_g.
        Copy all properties and literals of the last external entity e_x to e_g.
        Disconnect path from e_x entity and connect it to e_g via the same property.
        cost_i = cost of Replacing e_x by e_g
      else
        Let p_l be the property edge connecting the last internal entity of the path to an external entity.
        cost_i = cost of Suppressing p_l
    else if path ends in a literal vertex v **then**
      if an appropriate generalization V of v exists **then**
        cost_i = cost of Generalizing $\{v\} \rightarrow V$
      else
        cost_i = cost of Suppressing v
  else if path starts with an interpersonal property p_f **then**
    if an appropriate generalization P_f of p_f exists **then**
      cost_i = cost of Generalizing $\{p_f\} \rightarrow P_f$
    else
      cost_i = cost of Suppressing p_f
minCost = $\min\{\text{cost}_i\}$, $\forall$path $i \in cmb$.
Apply the operation that introduces the minimum information loss to S'_G.
return S'_G

Κεφάλαιο 5

Προστασία Ιδιωτικότητας Αδόμητων Δεδομένων με Συνεχή Γνωρίσματα

Σε αυτό το κεφάλαιο μελετάμε την ανωνυμοποίηση δεδομένων των οποίων οι εγγραφές είναι σύνολα τιμών από ένα συνεχές πεδίο ορισμού, λχ. αριθμητικά δεδομένα. Προτείνουμε μια μέθοδο για την προστασία τους από επιθέσεις αποκάλυψης ταυτότητας. Η βασική συνεισφορά της μεθόδου μας είναι ότι αντικαθιστούμε την χρήση της προκαθορισμένης ιεραρχίας γενίκευσης τιμών με την δημιουργία δυναμικών ιεραρχιών γενίκευσης που αποφασίζονται από τον αλγόριθμο κατά την ανωνυμοποίηση. Το όφελος αυτής της προσέγγισης είναι διπλό: α) μας επιτρέπει να γενικεύουμε τιμές χωρίς να απαιτείται η βοήθεια ενός ειδικού για την δημιουργία μιας προκαθορισμένης ιεραρχίας, και β) μας δίνει την δυνατότητα να μειώσουμε την απώλεια πληροφορίας περιορίζοντας τα εύρη των γενικεύσεων. Τέλος, δείχνουμε πειραματικά τα οφέλη της μεθόδου μας ως προς την ποιότητα των ανωνυμοποιημένων δεδομένων και τα συγκρίνουμε με την state-of-the-art μέθοδο [68].

5.1 Κίνητρο και Συνεισφορά

Δεδομένα που αποτελούνται από σύνολα αριθμητικών τιμών παρουσιάζονται συχνά σε πολλές εφαρμογές. Οι μετρήσεις αισθητήρων, οι καταγραφές ανθρώπινης παρατήρησης, οι δείκτες ιατρικών εξετάσεων όπως οι μετρήσεις σφυγμών και πίεσης του αίματος, τα οικονομικά δεδομένα όπως σύνολα πληρωμών ή αγορών από πιστωτικές κάρτες, είναι όλα παραδείγματα τέτοιων δεδομένων.

Ας θεωρήσουμε το παράδειγμα του Πίνακα 5.1, όπου παρουσιάζονται σύνολα πληρωμών από διαφορετικούς χρήστες μιας υπηρεσίας, λ.χ. επαναφορτίσεις μιας χρεωστικής κάρτας. Αν τα δεδομένα δημοσιευθούν ως έχουν, με απλή απαλοιφή των ονομάτων, ένας κακόβουλος επιτιθέμενος που έχει μερική γνώση μιας εγγραφής θα μπορούσε να ταυτοποιήσει μια εγγραφή στο σύνολο δεδομένων. Παραδείγματος χάριν, η Αλίκη μπορεί να γνωρίζει ότι ο Μανώλης έχει πραγματοποιήσει μια πληρωμή αξίας 11,000 και άλλη που το ποσό της κυμαίνεται από

Όνομα	*Πληρωμές*
Μανώλης	{11000, 11000, 20000, 40000, 40000}
Αιμιλία	{11000, 30500, 40000}
Νίκος	{11000, 11000, 40000, 40000}
Θεοδώρα	{11000}
Θανάσης	{20000}

Πίνακας 5.1: Αρχικά δεδομένα πληρωμών.

18,000 έως 22,000. Ακόμη και αν όλα τα μοναδικά αναγνωριστικά (ονοματεπώνυμο, ΑΦΜ, κτλ.) αφαιρεθούν από τον δημοσιευμένο πίνακα, η Αλίκη μπορεί με βεβαιότητα να συμπεράνει ότι η εγγραφή του Μανώλη είναι η πρώτη εγγραφή του πίνακα.

Σε αυτό το κεφάλαιο εστιάζουμε επίσης στην πρόληψη από επιθέσεις αποκάλυψης ταυτότητας, δηλαδή στην προστασία από επιτιθέμενους οι οποίοι προσπαθούν να συσχετίσουν μια εγγραφή στα δεδομένα με ένα πραγματικό άτομο στόχο. Για την προστασία της ιδιωτικότητας των εγγραφών εφαρμόζουμε την εγγύηση της k^m-ανωνυμίας [68]. Η k^m-ανωνυμία εγγυάται ότι κάθε κακόβουλος επιτιθέμενος που γνωρίζει έως m αντικείμενα από το σύνολο αντικειμένων που αποτελούν την εγγραφή, δεν θα μπορεί να αντιστοιχίσει αυτή τη γνώση του σε λιγότερες από k προσωπικές εγγραφές στα δεδομένα. Η εγγύηση αυτή αποτελεί μια χαλάρωση της κλασσικής k-ανωνυμίας [63].

Παράδειγμα 5.16. *Ο Πίνακας 5.2 αποτελεί την 2^2-ανωνυμοποίηση των δεδομένων του Πίνακα 5.1. Οποιοσδήποτε επιτιθέμενος με μερική γνώση έως 2 πληρωμών ενός ατόμου, δεν θα μπορεί να αναγνωρίσει λιγότερες από 2 εγγραφές στα δημοσιευμένα δεδομένα. Για να εξασφαλίσουμε αυτή την εγγύηση ιδιωτικότητας, χρησιμοποιήσαμε την προκαθορισμένη ιεραρχία γενίκευσης τιμών του Σχήματος 5.1. Όλες οι τιμές του παραδείγματος έπρεπε να γενικευθούν διότι οι τιμές {20,000} και {30,500} ήταν σπάνιες στα αρχικά δεδομένα. Εντούτοις, το ίδιο επίπεδο ιδιωτικότητας μπορεί να επιτευχθεί με διαφορετικό τρόπο όπως φαίνεται στον Πίνακα 5.3, όπου οι τιμές {20,000} και {30,500} έχουν γενικευθεί στο εύρος [20,000-30,500]. Όπως μπορούμε να παρατηρήσουμε λιγότερες τιμές έχουν γενικευθεί και η απώλεια πληροφορίας είναι σημαντικά περιορισμένη. Αξίζει να σημειωθεί ότι ενώ η κλασική k-ανωνυμία λύνει επίσης το πρόβλημα της αποκάλυψης ταυτότητας, προκαλεί μεγαλύτερη απώλεια πληροφορίας στα δεδομένα όπως έχει δειχθεί στο [68]. Στο παράδειγμα που μελετάμε, η 2-ανώνυμη εκδοχή*

Id	*Πληρωμές*
1	(10000-20000], (10000-20000], (30000-40000], (30000-40000], (30000-40000]
2	(10000-20000], (30000-40000], (30000-40000]
3	(10000-20000], (10000-20000], (30000-40000], (30000-40000]
4	(10000-20000]
5	(10000-20000]

Πίνακας 5.2: 2^2-Ανώνυμα δεδομένα χρησιμοποιώντας Ιεραρχία Γενίκευσης.

Id	*Πληρωμές*
1	11000, 11000, [20000-30500], 40000, 40000
2	11000, [20000-30500], 40000
3	11000, 11000, 40000, 40000
4	11000
5	[20000-30500]

Πίνακας 5.3: 2^2-Ανώνυμα δεδομένα με χρήση δυναμικής Ιεραρχίας Γενίκευσης.

του Πίνακα 5.1 παρουσιάζεται στον Πίνακα 5.4. Η περισσότερη πληροφορία έχει χαθεί καθώς οι εγγραφές χωρίζονται σε ομάδες μεγέθους τουλάχιστον 2 *και μέσα σε κάθε ομάδα επιβάλλεται να γίνουν πανομοιότυπες μεταξύ τους. Οι αστερίσκοι (*) δεν δημοσιεύονται, συμβολίζουν τις απαλοιφές των τιμών που έχουν πραγματοποιηθεί.*

Η βασική καινοτομία της μεθόδου μας είναι ότι δεν χρησιμοποιούμε μια προκαθορισμένη ιεραρχία γενίκευσης, δηλαδή δεν είναι εκ των προτέρων γνωστό πως θα ομαδοποιηθούν οι αρχικές τιμές και θα απεικονιστούν σε γενικότερες (εύρη τιμών) που τις περιέχουν. Αντίθετα, ο αλγόριθμος ανωνυμοποίησης εξερευνά δυναμικά τους διαφορετικούς γενίκευσης του αρχικού πεδίου τιμών. Ομαδοποιεί τιμές που είναι κοντά και τις γενικεύει στο μιρότερο δυνατό εύρος. Τα βασικότερα οφέλη της μεθόδου μας είναι ότι (α) δεν χρειαζόμαστε μια σαφώς ορισμένη ιεραρχία, η δημιουργία της οποίας αποτελεί ένα ακόμη πρόβλημα για τον κάτοχο των δεδομένων και (β) εξερευνώντας ένα μεγαλύτερο χώρο πιθανών λύσεων, δηλαδή διαφορετικών πιθανών συνδυασμών από κανόνες γενίκευσης, μας δίνεται η δυνατότητα να περιορίσουμε τις απώλειες πληροφορίας που οφείλονται στην ανωνυμοποίηση.

Το μοντέλο των δεδομένων που μελετάμε σε αυτή την εργασία αυτή διαφέρει από το αντίστοιχο της k^m-ανωνυμίας [68, 71, 72] διότι (α) εστιάζει σε συνεχείς τιμές, και όχι κατηγορικές όπως στις προηγούμενες μεθόδους, (β) επιτρέπει την ύπαρξη διπλοτύπων, δηλαδή οι εγγραφές είναι bags και όχι sets, και (γ) δεν προϋποθέτει την ύπαρξη μιας προκαθορισμένης ιεραρχίας γενίκευσης.

Η βασική συνεισφορά του Κεφαλαίου 5 περιλαμβάνει τα ακόλουθα σημεία:

- Επεκτείνουμε το πρόβλημα της ανωνυμοποίησης δεδομένων από σύνολα τιμών [68] σε συλλογές συνόλων συνεχών δεδομένων που επιτρέπουν την ύπαρξη διπλοτύπων,

Id	*Πληρωμές*
1	(10000-20000], *, *, 40000, *
2	(10000-20000], *, 40000
3	(10000-20000], *, 40000, *
4	(10000-20000]
5	(10000-20000]

Πίνακας 5.4: 2-Ανώνυμα δεδομένα πληρωμών.

- Παρουσιάζουμε τις βασικές διαφορές και προκλήσεις της εφαρμογής της εγγύησης της k^m-ανωνυμίας σε αυτό το σενάριο δεδομένων,
- Προτείνουμε έναν αλγόριθμο k^m-ανωνυμοποίησης συνεχών δεδομένων που διατηρεί καλύτερη ποιότητα στα τελικά δεδομένα και δεν χρησιμοποιεί προκαθορισμένες ιεραρχίες γενίκευσης τιμών,
- Αξιολογούμε την προτεινόμενη μέθοδο με πραγματικά δεδομένα και συγκρίνουμε τα πειραματικά αποτελέσματα με τον apriori [68], έναν αλγόριθμο k^m-ανωνυμοποίησης που χρησιμοποιεί προκαθορισμένες ιεραρχίες σε σύνολα κατηγορικών τιμών.

5.2 Ορισμός του Προβλήματος

5.2.1 Μοντέλο Δεδομένων

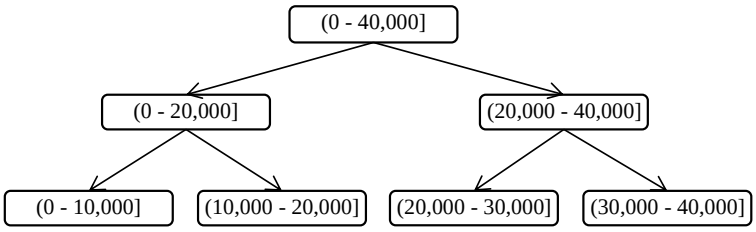
Θεωρούμε ένα σύνολο δεδομένων D, όπου κάθε εγγραφή t είναι μια συλλογή τιμών v από ένα συνεχές πεδίο ορισμού $\mathcal{I}$. Κάθε εγγραφή είναι προσωπική, δηλαδή αντιστοιχεί σε μια διαφορετική πραγματική οντότητα (άτομο). Οι εγγραφές μπορεί να περιέχουν την ίδια τιμή περισσότερες από μία φορά.

5.2.2 Μοντέλο Επίθεσης

Θεωρούμε επιτιθέμενους που κατέχουν μερική γνώση μιας εγγραφής. Μπορούν να γνωρίζουν έως m τιμές που σχετίζονται με ένα πραγματικό άτομο, και επιθυμούν να αναγνωρίζουν ολόκληρη την εγγραφή στα δημοσιευμένα δεδομένα. Δεν διακρίνουμε εκ των προτέρων τις τιμές σε ευαίσθητα στοιχεία και σε ψευδο-αναγνωριστικά στοιχεία. Κάθε τιμή θεωρείται εν δυνάμει ψευδο-αναγνωριστικό, ενώ όλες οι τιμές είναι εξίσου ευαίσθητες. Ο ορισμός της εγγύησης της k^m-ανωνυμίας [68] είναι ο παρακάτω:

Ορισμός 5.9. *(k^m-ανωνυμία [68]) Ένα σύνολο δεδομένων D θεωρείται k^m-ανώνυμο αν οποιοσδήποτε επιτιθέμενος που γνωρίζει έως m τιμές μιας εγγραφής $t \in D$, δεν μπορεί να χρησιμοποιήσει αυτή τη γνώση του για να αναγνωρίσει λιγότερες από k εγγραφές του D.*

Η k^m-ανωνυμία απαιτεί κάθε εγγραφή του συνόλου δεδομένων να μην μπορεί να διαχωριστεί από τουλάχιστον $k-1$ άλλες, ως προς κάθε πιθανό συνδυασμό m τιμών του. Αυτό



Σχήμα 5.1: Ιεραρχία Γενίκευσης Τιμών για τα δεδομένα του Πίνακα 5.1.

σημαίνει ότι οποιοσδήποτε επιτιθέμενος που γνωρίζει έως m τιμές που σχετίζονται με ένα άτομο, θα βρίσκει πάντα τουλάχιστον k εγγραφές στα δημοσιευμένα δεδομένα που να ταιριάζουν στη γνώση του. Αντίθετα με την παραδοσιακή k-ανωνυμία, δεν απαιτείται από τις εγγραφές να είναι πανομοιότυπες. Στην περίπτωση των αραιών πολυδιάστατων δεομένων, κάτι τέτοιο θα εισήγαγε μεγάλες απώλειες πληροφορίας. Επιπλέον, θα είχε μικρή αξία καθώς θα προστάτευε από επιτιθέμενους οι οποίοι ήδη γνωρίζουν ολόκληρη την εγγραφή.

5.2.3 Πράξεις Ανακωδικοποίησης

Ένα σύνολο δεδομένων D το οποίο δεν είναι k^m-ανώνυμο, μπορεί να μετασχηματιστεί σε ένα k^m-ανώνυμο σύνολο $D^\star$, ανακωδικοποιώντας τις τιμές του έτσι ώστε το $D^\star$ να ικανοποιεί την εγγύηση της k^m-ανωνυμίας. Για να επιτευχθεί αυτό, γενικεύονται μόνο οι τιμές εκείνες που είναι απαραίτητες ώστε κάθε συνδυασμός μεγέθους m να εμφανίζεται σε τουλάχιστον k εγγραφές, όπως φαίνεται στον Πίνακα 5.3. Οι πράξεις ανακωδικοποίησης που εφαρμόζονται είναι κανόνες γενίκευσης της μορφής $v \rightarrow [a, b]$, που απεικονίζουν μια τιμή v των αρχικών δεδομένων σε ένα εύρος τιμών που την περιέχει. Η μέθοδός μας χρησιμοποιεί ολική ανακωδικοποίηση, δηλαδή όταν μια τιμή v γενικεύεται σε $[a, b]$, τότε όλες οι εμφανίσεις της v στα δεδομένα αντικαθίστανται από το εύρος $[a, b]$.

Μπορεί να υπάρχουν πολλές διαφορετικές πιθανές ανωνυμοποιήσεις των δεδομένων που ικανοποιούν την εγγύηση της k^m-ανωνυμίας, δεδομένου του ορίου γνώσης του επιτιθέμενου m και της παραμέτρου ιδιωτικότητας k, όπως φαίνεται από τους Πίνακες 5.3 και 5.2. Το χειρότερο σενάριο από πλευράς ποιότητας του αποτελέσματος είναι να γενικευθούν όλες οι τιμές των εγγραφών στο μέγιστο εύρος $\mathcal{I}$. Μια τέτοια λύση θα ικανοποιούσε την εγγύηση, αλλά θα εισήγαγε την μέγιστη απώλεια πληροφορίας και τα ανωνυμοποιημένα δεδομένα δεν θα είχαν πρακτικά καμία χρησιμότητα.

Το πρόβλημα της εύρεσης της βέλτιστης k^m-ανωνυμοποίησης είναι η εύρεση ενός συνόλου κανόνων γενίκευσης που να ικανοποιούν την εγγύηση της k^m-ανωνυμίας ενώ προκαλεί την μικρότερη απώλεια πληροφορίας στα δεδομένα.

5.3 Αλγόριθμος Ανωνυμοποίησης

5.3.1 Χώρος Λύσεων

Ο χώρος των λύσεων του προβλήματος αποτελείται από όλους τους πιθανούς συνδυασμούς κανόνων γενίκευσης των τιμών. Αυτοί περιλαμβάνουν όλες τις πιθανές αντικαταστάσεις οποιασδήποτε τιμής v από ένα εύρος που την περιλαμβάνει. το εύρος γενίκευσης μπορεί να είναι οποιοδήποτε υποσύνολο του $\mathcal{I}$. Οι *αποδεκτές* λύσεις είναι εκείνες που δεν παραβιάζουν την εγγύηση της k^m-ανωνυμίας. Το πρόβλημα της βέλτιστης πολυδιάστατης k-ανωνυμίας έχει αποδειχθεί πως είναι NP-hard [53]. Το μοντέλο δεδομένων του προβλήματός μας θα μπορούσε να αναπαρασταθεί ως ένας αραιός πολυδιάστατος πίνακας, ενώ ο χώρος των λύσεων είναι πολύ μεγαλύτερος από εκείνον της κλασικής k-ανωνυμίας. Οι λόγοι για αυτό είναι ότι αφενός η k^m-ανωνυμία δεν απαιτεί την δημιουργία αυστηρών κλάσεων ισοδυναμίας μέσα στις οποίες οι

εγγραφές είναι πανομοιότυπες, και αφετέρου ότι δεν χρησιμοποιούμε προκαθορισμένη ιεραρχία γενίκευσης, συνεπώς το σύνολο των πιθανών γενικεύσεων είναι σημαντικά μεγαλύτερο. Λόγω της πολυπλοκότητας του προβλήματος της βέλτιστης ανωνυμοποίησης, αναπτύξαμε έναν ευρηστικό αλγόριθμο. Εκμεταλλευόμαστε το αξίωμα *a priori* και εφαρμόζουμε γενικεύσεις ολικής ανακωδικοποίησης στις πιο σπάνιες τιμές σε κάθε βήμα του αλγορίθμου.

5.3.2 Δυναμικό Δένδρο Καταμέτρησης (Dynamic Count Tree)

Σύμφωνα με την αρχή a priori, δοθέντος ενός κατωφλίου συχνοτήτων k, οποιοδήποτε σύνολο αντικειμένων μεγέθους n δεν μπορεί να έχει μεγαλύτερη συχνότητα από k αν οποιοδήποτε υποσύνολό του είναι σπάνιο. Ισοδύναμα, αν ένα σύνολο αντικειμένων μεγέθους n έχει συχνότητα μικρότερη του k, τότε όλα τα υπερσύνολα μεγέθους $n+1, n+2$, κτλ. που το περιέχουν, είναι επίσης σπάνια.

Αξιοποιώντας αυτή την ιδιότητα, ο προτεινόμενος αλγόριθμος χρησιμοποιεί μια δενδρική δομή παρόμοια με το FP-tree [41], την οποία καλούμε *δυναμικό δένδρο καταμέτρησης*. Κάθε κόμβος αντιστοιχεί σε μια τιμή, είτε αρχική είτε ένα γενικευμένο εύρος τιμών. Οι κόμβοι του πρώτου επιπέδου κάτω από τη ρίζα του δένδρου διατηρούν την *υποστήριξη* (support) των τιμών, δηλαδή το πλήθος των εγγραφών που περιέχουν κάθε τιμή. Κάθε μονοπάτι από την ρίζα προς ένα κόμβο με βάθος i, αντιστοιχεί σε ένα συνδυασμό τιμών με μέγεθος i. Κάθε κόμβος n_i που βρίσκεται σε κάποιο ενδιάμεσο επίπεδο i του δένδρου, διατηρεί την *υποστήριξη* του συνδυασμού των τιμών που εμφανίζονται στο μονοπάτι από την ρίζα ως τον κόμβο n_i. Ο όρος υποστήριξη διαφέρει από την συχνότητα στο ότι δεν λαμβάνει υπόψη αν σε μια εγγραφή ένας συνδυασμός τιμών εμφανίζεται πολλαπλές φορές. Αντίστοιχα με τον Ορισμό 3.3 για μονοπάτια δενδρικών εγγραφών, ο όρος «υποστήριξη» για το σενάριο δεδομένων που μελετάμε, μπορεί να διατυπωθεί ως εξής:

Ορισμός 5.10. *Υποστήριξη (support) ενός συνδυασμού τιμών σε ένα σύνολο δεδομένων είναι το πλήθος των εγγραφών στις οποίες εμφανίζεται ο συνδυασμός.*

Οι κόμβοι που είναι αδέλφια είναι ταξινομημένοι κατά φθίνουσα σειρά της υποστήριξής τους, δηλαδή οι πιο συχνοί συνδυασμοί εμφανίζονται πρώτοι. Στο πρώτο βήμα κτισίματος του δένδρου καταμέτρησης, προστίθεται στο πρώτο επίπεδο κάτω από την ρίζα ένας κόμβος για κάθε διαφορετική τιμή που εμφανίζεται στα δεδομένα, όπως φαίνεται στο Σχήμα 5.2(α). Στο επόμενο βήμα, εισάγονται στο δένδρο οι κόμβοι του δεύτερου επιπέδου. Αυτοί αντιστοιχούν στους συνδυασμούς τιμών μεγέθους 2. Οι συνδυασμοί αυτοί επίσης ταξινομούνται κατά φθίνουσα υποστήριξη. Αν μια τιμή v_1 του κόμβου n_1 είναι πιο συχνή από την v_2 του n_2, περιμένουμε να βρούμε τον συνδυασμό $\{v_1, v_2\}$ στο μονοπάτι $n_1 \rightarrow n_2$. Σε κάθε βήμα i του αλγορίθμου, ένα νέο επίπεδο κόμβων εισάγεται στο δένδρο. Συνδυασμοί με κοινά προθέματα μοιράζονται κοινά υπο-μονοπάτια στο δένδρο. Παραδείγματος χάριν οι συνδυασμοί {5, 10, 2}και {5, 10, 1} θα μοιράζονται το μονοπάτι $5 \rightarrow 10$ στο δένδρο. Σημειώνουμε ότι επιτρέπεται να εμφανίζονται στο ίδιο μονοπάτι διαφορετικοί κόμβοι που φέρουν την ίδια τιμή, λόγω της ύπαρξης διπλοτύπων τιμών στις εγγραφές.

Αλγόριθμος 8 UpdateDCTree Ενημέρωση του Δυναμικού Δένδρου Καταμέτρησης

Require: D {Original Dataset}, T_{i-1} {tree of size $i-1$}, G {current generalizations}
Ensure: T_i is the count tree of height i.

```
1: for every record t ∈ D do
2:   for every value v ∈ t do
3:     if ∃ generalization range g ∈ G, such that v ∈ g then
4:       replace v with g.
5:   for every combination cmb_i of i values in t do
6:     find path p_{i-1} that contains (i-1)-subset of cmb_i (prefix)
7:     if the i^th value exists as a leaf then
8:       increase its support by 1.
9:     else
10:      add the remaining i^th value as a leaf under p_{i-1}
11: return D*
```

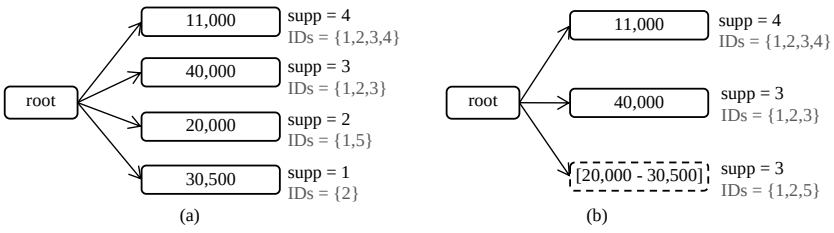
Ο στόχος είναι κάθε συνδυασμός τιμών μεγέθους m να αποκτήσει υποστήριξη τουλάχιστον k. Για να επιτευχθεί αυτό, ακολουθώντας την ιδιότητα a priori, εξετάζουμε συνδυασμούς με προοδευτικά αυξανόμενο μέγεθος $i = 1, 2, ..., m$. Σε κάθε βήμα i, εξασφαλίζουμε ότι όλοι οι συνδυασμοί τιμών μεγέθους i υποστηρίζονται από τουλάχιστον k εγγραφές, πριν προχωρήσουμε στο επόμενο βήμα $i+1$.

5.3.3 Μετρική Απώλειας Πληροφορίας

Η αποτίμηση της απώλειας πληροφορίας και η αντίστοιχη μείωση της ποιότητας των δεδομένων λόγω των γενικεύσεων γίνεται συχνά στην βιβλιογραφία χρησιμοποιώντας την μετρική Κανονικοποιημένης Ποινής Βεβαιότητας (Normalized Certainty Penalty - NCP) [80]. Έστω μια τιμή v από το αρχικό πεδίο τιμών $\mathcal{I}$ των δεδομένων. Η κανονικοποιημένη ποινή βεβαιότητας για την γενίκευσή της στο εύρος $[g_{min}, g_{max}]$ δίνεται από την εξίσωση:

$$NCP(v) = \begin{cases} 0, & v \text{ is not generalized} \\ |g_{max} - g_{min}|/|\mathcal{I}|, & otherwise \end{cases}$$

Η συνολική απώλεια πληροφορίας ενός ανωνυμοποιημένου συνόλου δεδομένων D^* καλείται



Σχήμα 5.2: (α) Δένδρο Καταμέτρησης T_1 για τα δεδομένα του Πίνακα 5.1. (β) T_1 μετά την απαραίτητη γενίκευση 30,500→[20,000-30,500].

Γενικευμένη Ποινή Βεβαιότητας (GCP) και είναι ο μέσος όρος της NCP όλων των τιμών του:

$$GCP(D^*) = \frac{\sum_{t_i \in D^*} \{\sum_{v_{i,j} \in t_i} NCP(v_{i,j})\}}{\sum_{t_i \in D^*} |t_i|}$$

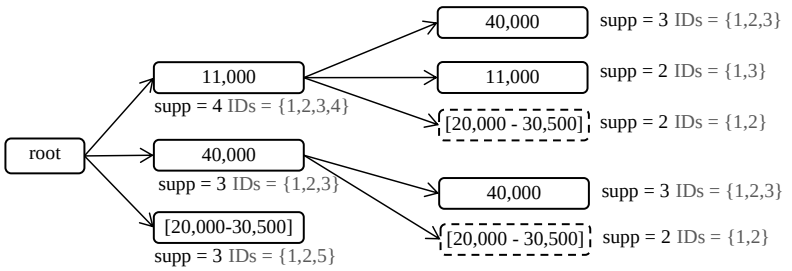
όπου $v_{i,j}$ είναι η j-στή τιμή της i-στής εγγραφής και $|t_i|$ είναι το μέγεθος (ως πλήθος τιμών) της i-στής εγγραφής.

5.3.4 Αλγόριθμος

Προτείνουμε έναν ευρηστικό αλγόριθμο γενίκευσης με ολική ανακωδικοποίηση. Όπως φαίνεται από τον ψευδοκώδικα του Αλγορίθμου 9, η μέθοδός μας αποτελείται από m βασικά βήματα. Σε κάθε βήμα $i = 1, ..., m$, ο αλγόριθμος τον οποίο καλούμε `ACD` (Anonymization of Continuous Data). ελέγχει για παραβιάσεις της ιδιωτικότητας από συνδυασμούς μεγέθους i. Για τον έλεγχο κάθε πιθανού συνδυασμού i τιμών, χρησιμοποιούμε το δυναμικό δένδρο καταμέτρησης που δημιουργείται από τον Αλγόριθμο 8.

Κάθε μονοπάτι από την ρίζα ως κάποιο φύλλο αντιστοιχεί σε ένα συνδυασμό τιμών του οποίου η υποστήριξη ισούται με την υποστήριξη που κρατάει αποθηκευμένη ο κόμβος-φύλλο. Αν ένα φύλλο έχει υποστήριξη μικρότερη από k, τότε ο συνδυασμός αυτός είναι σπάνιος και θεωρείται ευάλωτος σε επιθέσεις. Για την προστασία ατόμων των οποίων οι εγγραφές περιέχουν αυτό το συνδυασμό, μία ή περισσότερες τιμές του θα πρέπει να γενικευθούν. Ο στόχος είναι να αυξηθεί η υποστήριξη του μονοπατιού. Ο μόνος τρόπος για να επιτευχθεί αυτό είναι να γενικευθεί μια τιμή αρκετά, ώστε το εύρος γενίκευσης να περιλαμβάνει τιμές κάποιων αδελφών κόμβων από γειτονικά μονοπάτια. Έτσι θα μπορέσει να συγχωνευθεί ο κόμβος της γενικευμένης τιμής με κάποια από τα αδέλφια του και οι υποστηρίξεις τους θα συνδυαστούν. Αν τα αδέλφια περιέχονταν σε διαφορετικές εγγραφές, η συνολική υποστήριξη του συγχωνευμένου κόμβου θα είναι υψηλότερη από τις υποστηρίξεις των αρχικών κόμβων. Η τιμή του συγχωνευμένου κόμβου θα είναι το ελάχιστο εύρος τιμών που περιέχει τις τιμές των αρχικών κόμβων.

Η χρήση ολικής ανακωδικοποίησης σημαίνει ότι από την στιγμή που ο αλγόριθμος αποφασίζει να εφαρμόσει έναν κανόνα γενίκευσης $v \rightarrow [v_{min}, v_{max}]$ για μια τιμή v, τότε κάθε άλλη τιμή v', τέτοια ώστε $v' \in [v_{min}, v_{max}]$, θα γενικευθεί επίσης στο ίδιο εύρος τιμών στο δυναμικό δένδρο καταμέτρησης, όπως φαίνεται στις γραμμές 10-14του Αλγορίθμου 9. Αυτό



Σχήμα 5.3: Δένδρο Καταμέτρησης T_2 για τα δεδομένα του Πίνακα 5.1.

Αλγόριθμος 9 ACD k^m-Ανωνυμοποίηση για Συνεχή Δεδομένα

Require: D {Original Dataset}, m {maximum size of attacker's knowledge}, k {privacy parameter}, d {NCP threshold}

Ensure: $D^\star$ is a k^m-anonymous Dataset.

1: sort tuples' values with reference to their support.
2: $G = \emptyset$
3: $T_0 = null$
4: **for** i = 1, 2, ..., m **do**
5: $T_i = UpdateDCTree(D, T_{i-1}, G)$
6: **for** every leaf node f in T_i **do**
7: **if** $support(f) < k$ **then**
8: $G_f = findGeneralizations(T_i, f, k, d)$
9: add generalization rules: $G = G \cup G_f$.
10: parse T_i in a breadth-first traversal
11: **if** there exist sibling nodes with values $v_1, ..., v_n \in g$, where $g \in G_f$ **then**
12: replace values $v_1, ..., v_n$ with g
13: merge them into a single node n
14: update n's support
15: **return** $D^\star$

προκαλεί τα αδέλφια κόμβους που φέρουν την ίδια γενικευμένη ετικέτα να συγχωνεύονται μαζί. Τέτοιες συγχωνεύσεις συμβαίνουν σε κόμβους που μπορεί να βρίσκονται σε διάφορα τα επίπεδα του δένδρου, μειώνοντας σημαντικά το μέγεθός του στη μνήμη. Κάθε γενίκευση που έχει αποφασιστεί σε προηγούμενα βήματα 1,...,i-1, διατηρείται σε ένα σύνολο κανόνων γενίκευσης G (γραμμή 9) έτσι ώστε να ληφθούν υπόψη κατά το κτίσιμο του επόμενου επιπέδου i του δυναμικού δένδρου καταμέτρησης.

Η διαδικασία που ακολουθεί ο αλγόριθμος για την εύρεση των κατάλληλων κανόνων γενίκευσης που θα προκαλέσουν λιγότερη απώλεια πληροφορίας στα δεδομένα, περιγράφεται στον ψευδοκώδικα του Αλγορίθμου 10. Όταν ένα φύλλο έχει υποστήριξη μικρότερη από k, τα αδέλφια του είναι οι πρώτοι επιλαχόντες κόμβοι για πιθανή συγχώνευση. Ο λόγος είναι διότι μοιράζονται κοινό πρόθεμα, το μονοπάτι από την ρίζα ως τον κοινό τους γονέα. Το πρόθεμα αυτό είναι ένας συνδυασμός τιμών μεγέθους i-1 και η υποστήριξή του έχει εξασφαλιστεί ότι είναι μεγαλύτερη ή ίση του k από το προηγούμενο βήμα του αλγορίθμου `ACD`. Συνεπώς αρκεί να γενικευθούν μόνο οι τιμές των κατάλληλων φύλλων.

Η συνάρτηση $range(v_1, v_2)$στην γραμμή 11 επιστρέφει το εύρος μεταξύ δυο τιμών. Αν $v_1 < v_2$ τότε $range(v_1, v_2) = [v_1, v_2]$, διαφορετικά $range(v_1, v_2) = [v_2, v_1]$. Αν η συνδυασμένη υποστήριξη μεταξύ δυο μονοπατιών είναι $\geq k$ τότε αποτελεί μια υποψήφια λύση για αυτόν τον προβληματικό συνδυασμό τιμών. Για κάθε υποψήφια λύση υπολογίζεται η μετρική NCP της απώλειας πληροφορίας που θα προκαλούσε και επιλέγεται εκείνη η λύση που επιφέρει την μικρότερη αλλοίωση στα δεδομένα. Ορίζεται από τον χρήστη ένα μέγιστο όριο απώλειας ανά γενίκευση d. Αν η υποψήφια λύση με την μικρότερη απώλεια πληροφορίας δίνει $NCP < d$,

Αλγόριθμος 10 findGeneralizations Εύρεση Κατάλληλης Γενίκευσης

Require: T_i {Count Tree}, f {leaf of a vulnerable itemset path},
k {privacy parameter}, d {NCP threshold}

Ensure: generalized path of f will have a support $\geq k$.

1: $n = f$
2: $S = \emptyset$
3: $G_f = \emptyset$ {Generalization rules}
4: **for** every s_j sibling of node n **do**
5: $S = S \cup \{s_j\}$ {merge candidates}
6: **for** every node $s_j \in S$ **do**
7: **if** the combined support of s_j and n is $\geq k$ **then**
8: $NCP_j = NCP(\{v_n, v_{s_j} \rightarrow range(v_n, v_{s_j})\})$
9: **if** n is not a leaf **then**
10: **for** every node nc in the path from n to leaf f **do**
11: $NCP_j = NCP_j + NCP(\{v_{nc}, v_{sc_j} \rightarrow range(v_{nc}, v_{sc_j})\})$ {node sc_j is descendant of s_j, and it is at the same level as nc.}
12: find $s_j \in S$ such that NCP_j is minimum
13: **if** $NCP_j < d$ **then**
14: $g = range(v_n, v_{s_j})$
15: $G_f = G_f \cup g$
16: **for** every node nc in the path from n to leaf f **do**
17: $g = range(v_{nc}, v_{sc_j})$ {sc_j is descendant of s_j, and at the same level as nc.}
18: $G_f = G_f \cup g$
19: **else**
20: let node n be f's parent
21: goto 2
22: **return** G_f

εφαρμόζουμε αυτή την γενίκευση στα δεδομένα. Διαφορετικά, διατρέχουμε το προβληματικό μονοπάτι προς τα πάνω ως την ρίζα. Στο επόμενο επίπεδο, αναζητούνται γενικεύσεις και συγχωνεύσεις μεταξύ των φύλλων και των γονέων τους, και ούτω καθεξής, όπως φαίνεται στην γραμμή 20 του Αλγορίθμου 10.

Σημειώνεται ότι στην χειρότερη περίπτωση όλες οι τιμές θα γενικευθούν στο μέγιστο δυνατό εύρος, δηλαδή το αρχικό πεδίο τιμών $\mathcal{I}$. Συνεπώς ο αλγόριθμος `ACD` θα βρίσκει πάντα μια k^m-ανώνυμη λύση του προβλήματος για οποιοδήποτε σύνολο δεδομένων.

Παράδειγμα 5.17. *Θεωρείστε τα αρχικά δεδομένα πληρωμών του Πίνακα 5.1. Έστω οι τιμές των παραμέτρων $k = 2$ και $m = 2$. Το Σχήμα 5.2(a) απεικονίζει το δένδρο καταμέτρησης T_1, ύψους 1. Η τιμή 11,000 εμφανίζεται στης εγγραφές 1, 2, 3 και 4, συνεπώς έχει υποστήριξη 4, ενώ η τιμή 30,500 έχει υποστήριξη 1 διότι εμφανίζεται μόνον στην εγγραφή 2. Δεδομένου ότι έχει υποστήριξη μικρότερη του $k = 2$, θα πρέπει να γενικευθεί. Το καλύτερο*

εύρος γενίκευσης είναι το [20,000-30,500] καθώς επηρεάζει λιγότερες τιμές στο σύνολο των δεδομένων και δίνει μικρότερη NCP σε σχέση με τια άλλες επιλογές. Η γενίκευση αυτή εφαρμόζεται τόσο στον κόμβο 30,500 όσο και στον 20,000 που εμπίπτουν στο εύρος τιμών της γενίκευσης. Οι δύο κόμβοι συγχωνεύονται και η υποστήριξη του νέου κόμβου είναι 3'k, όπως φαίνεται στο Σχήμα 5.2(β). Στο επόμενο βήμα προστίθενται οι συνδυασμοί τιμών μεγέθους 2. Το δένδρο T_2 απεικονίζεται στο Σχήμα 5.3 όπου όλα τα φύλλα έχουν υποστήριξη τουλάχιστον k. Η έξοδος του αλγορίθμου είναι ο 2^2-ανώνυμος Πίνακας 5.3.

5.4 Πειραματική Μελέτη

Σε αυτή την ενότητα παρουσιάζονται τα πειραματικά αποτελέσματα για την αποτίμηση της μεθόδου. Η υλοποίηση έγινε σε γλώσσα προγραμματισμού C++ και η διεξαγωγή των πειραμάτων έγινε σε υπολογιστή Intel Core 2 Duo 2.53GHz, με 4GB RAM, σε λειτουργικό σύστημα Mac OS. Χρησιμοποιήθηκαν πραγματικά δεδομένα από το UCI repository [9].

5.4.1 Αλγόριθμοι

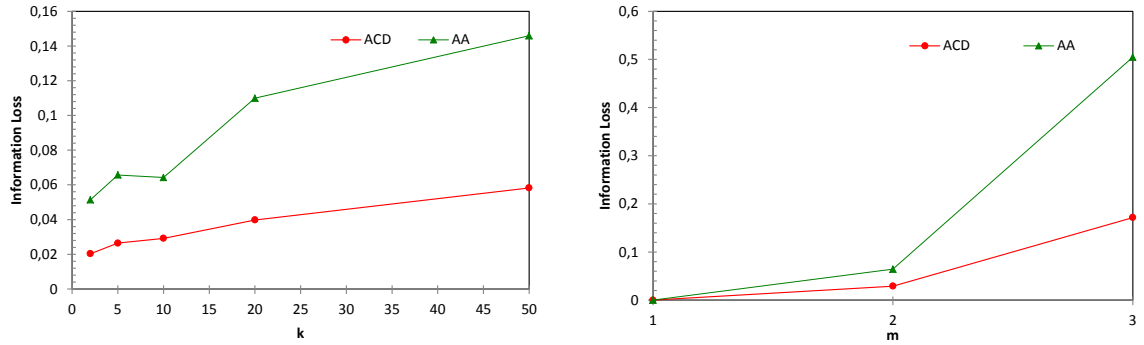
Συγκρίνουμε τον προτεινόμενο αλγόριθμο με τον αλγόριθμο Apriori (`AA`) από το [68]. Ο `AA` είναι ο state-of-the-art αλγόριθμος για την k^m-ανωνυμοποίηση δεδομένων με χρήση γενικεύσεων. Χρησιμοποιεί μια προκαθορισμένη ιεραρχία και αξιοποιεί την αρχή a priori: πρώτα δημιουργεί ένα k^1-ανώνυμο σύνολο, μετά ένα k^2-ανώνυμο, έως το k^m-ανώνυμο. Χρειάστηκε μια μικρή τροποποίησή του ώστε να δέχεται διπλότυπα τιμών στις εγγραφές. Η υλοποίηση του `AA` έγινε στην ίδια πλατφόρμα όπως ο βασικός μας αλγόριθμος `ACD`.

5.4.2 Πειραματικά Δεδομένα

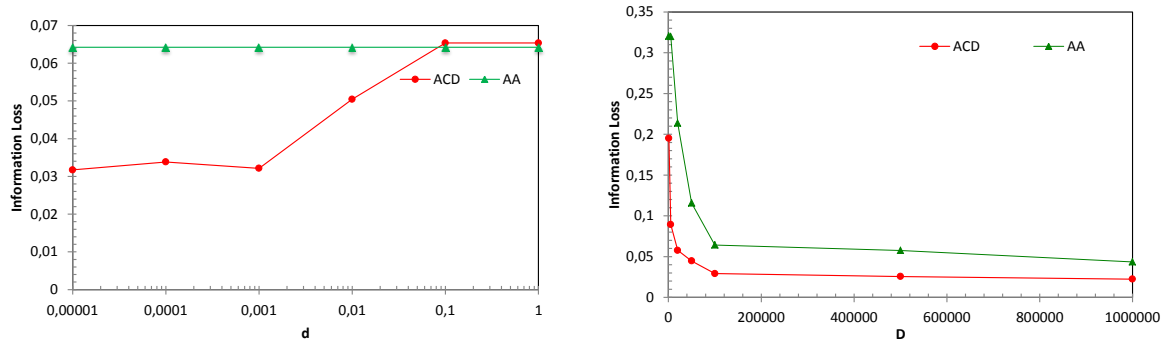
Χρησιμοποιήσαμε το σύνολο δεδομένων US Census 1990 Data Set [10] από την αποθήκη δεδομένων UCI data mining repository. Επιλέξαμε 8 αριθμητικά πεδία που αντιστοιχούν σε διαφορετικούς τύπους εισοδημάτων. Θεωρήσαμε τις μηδενικές τιμές ως κενές και τις αφαιρέσαμε από τις εγγραφές. Το συνολικό εύρος των τιμών που εμφανίζονται στα δεδομένα είναι [0-197297]. Το σύνολο δεδομένων περιείχε περίπου 2.5M εγγραφές, αλλά μετά την αφαίρεση όσων είχαν μηδενικές τιμές σε όλα τα επιλεγμένα πεδία έμειναν περίπου 1 εκατομμύριο εγγραφές. Το μέσο μέγεθος εγγραφής είναι 2.27.

5.4.3 Παράμετροι

Μελετάμε την συμπεριφορά του προτεινόμενου αλγορίθμου ως προς τις ακόλουθες παραμέτρους: (α) την παράμετρο ανωνυμίας k, (β) το όριο της γνώσης του επιτιθέμενου m, (γ) το κατώφλι της NCP d, και (δ) το μέγεθος των δεδομένων $|D|$. Σε κάθε πείραμα μεταβάλλεται μια από αυτές τις παραμέτρους ενώ οι υπόλοιπες κρατούνται σταθερές. Η προεπιλεγμένες τιμές των παραμέτρων που επιλέξαμε είναι $k = 10$, $m = 2$, $d = 0.001$ και $|D| = 100000$. Για να είναι δίκαιη η σύγκριση με τον αλγόριθμο `AA`, δημιουργήσαμε μια πολύ λεπτομερή ιεραρχία.



Σχήμα 5.4: Σύκριση της απώλειας πληροφορίας μεταξύ των ACD και AA ως προς τις παραμέτρους k και m.



Σχήμα 5.5: Σύκριση της απώλειας πληροφορίας μεταξύ των ACD και AA ως προς τις παραμέτρους d και $|D|$.

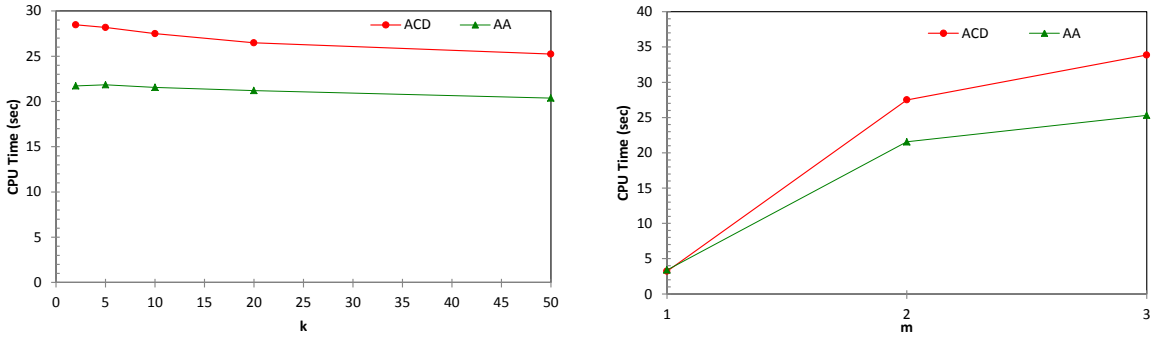
Χωρίσαμε το αρχικό πεδίο τιμών [0-197297] σε εύρη μεγέθους 100 και στη συνέχεια δημιουργήσαμε ένα δένδρο ιεραρχίας με εξάπλωση 2 παιδιά ανά κόμβο, που χρησιμοποιείται από τον αλγόριθμο AA.

5.4.4 Μετρικές Αποτίμησης

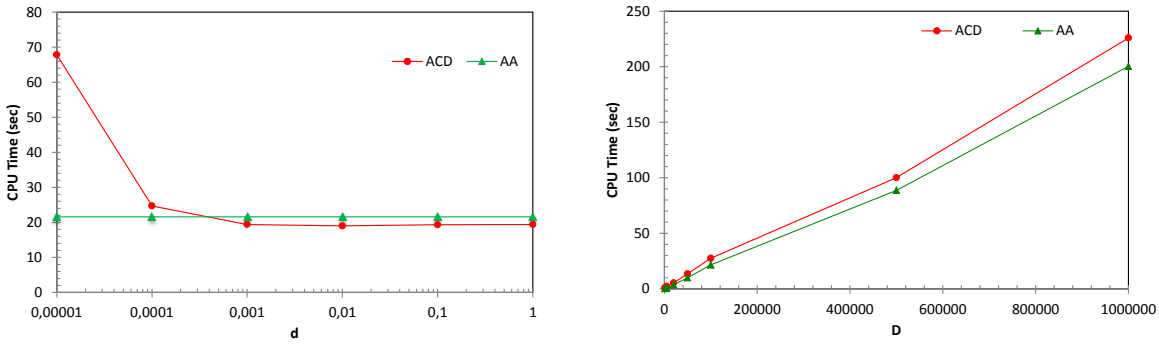
Για την αξιολόγηση της μεθόδου μας μετρήθηκε ο χρόνος εκτέλεσης των πειραμάτων σε δευτερόλεπτα καθώς και η απώλειας πληροφορίας με την μετρική GCP, για την αποτίμηση της ποιότητας των αποτελεσμάτων.

5.4.5 Ποιότητα Αποτελεσμάτων

Στο Σχήμα 5.4 οι δύο αλγόριθμοι συγκρίνονται ως προς την απώλεια πληροφορίας που προκαλούν στα δεδομένα. Καθώς η τιμή της παραμέτρου k μεγαλώνει, η μετρική GCP και των δύο αλγορίθμων αυξάνεται υπογραμμικά, αλλά ο ACD προκαλεί απώλειες ίσες με το $1/3$ εκείνων του AA. Καθώς το μέγιστο όριο γνώσης m του επιτιθέμενου μεγαλώνει, η τιμή της GCP αυξάνεται υπεργραμμικά και για τους δύο αλγόριθμους. Εντούτοις, κλιμακώνει καλύτερα για τον ACD. Το κόστος του AA γίνεται τριπλάσιο του κόστους του ACD καθώς η m αυξάνεται.



Σχήμα 5.6: Χρόνος Εκτέλεσης των ACD και AA ως προς τις παραμέτρους k και m.



Σχήμα 5.7: Χρόνος Εκτέλεσης των ACD και AA ως προς τις παραμέτρους d και $|D|$.

Η επιρροή των αλλαγών του κατωφλίου απωλειών d φαίνεται στην γραφική παράσταση του Σχήματος 5.5(α). Ο AA δεν επηρεάζεται από το d, συνεπώς μεταβάλλεται μόνο η τιμή της GCP για τον ACD. Απεικονίζεται η τιμή της GCP του AA για τις προεπιλεγμένες τιμές των παραμέτρων ($k = 10$, $m = 2$) ως γραμμή αναφοράς. Όταν το d είναι μικρό, ο ACD διατηρεί σημαντικά περισσότερη χρηστικότητα στα τελικά δεδομένα. Ακόμα και όταν το d είναι κοντά στην τιμή 1 (δηλαδή την μέγιστη τιμή που μπορεί να λάβει η μετρική NCP) ο αλγόριθμός μας παράγει αντίστοιχης ποιότητας αποτελέσματα με τον AA.

Στην δεύτερη γραφική παράσταση του Σχήματος 5.5. μεταβάλλουμε το πλήθος των εγγραφών των δεδομένων $|D|$. Για την διεξαγωγή αυτού του πειράματος, δημιουργήσαμε επτά υποσύνολα των δεδομένων. Τα μεγέθη τους είναι 500,000, 100,000, 50,000, 25,000, 10,000, 5,000 και 1,000 εγγραφές. Καθένα από αυτά προέκυψε από τυχαία δειγματοληψία του αμέσως επόμενου σε μέγεθος υποσυνόλου. Η απώλεια πληροφορίας και των δύο αλγορίθμων μειώνεται με το μέγεθος των δεδομένων, με τον ACD να ξεπερνά σε ποιότητα δεδομένων τον αλγόριθμο AA σε κάθε σύνολο δεδομένων.

5.4.6 Χρόνος Εκτέλεσης

Το υπολογιστικό κόστος του αλγόριθμου παρουσιάζεται στα Σχήματα 5.6 και 5.7. Ο χρόνος εκτέλεσης είναι μεγαλύτερος για μικρές τιμές της παραμέτρου k, και μειώνεται μονότονα καθώς η k αυξάνεται. Ο αλγόριθμος AA είναι ταχύτερος από τον ACD. Εντούτοις η διαφορά των χρόνων

τους είναι περιορισμένη, περίπου 25%, και δεν επηρεάζεται από την τιμή της k.

Ο χρόνος εκτέλεσης αυξάνεται υπογραμμικά ως προς το μέγεθος της γνώσης του επιτιθέμενου m και για τους δυο αλγορίθμους. Κάτι τέτοιο είναι αναμενόμενο, καθώς για μεγαλύτερες τιμές του m, θα πρέπει να εξεταστούν περισσότεροι συνδυασμοί μεγαλύτερου πλήθους τιμών και θα πρέπει να δημιουργηθούν περισσότερα επίπεδα του δένδρου καταμέτρησης.

Στην γραφική παράσταση του Σχήματος 5.7(α) βλέπουμε πως επηρεάζεται το υπολογιστικό κόστος από το κατώφλι d. Ενώ ο αλγόριθμος `ACD` είναι αργός για πολύ μικρά d, προσεγγίζει και καταφέρνει να ξεπεράσει τον `AA` για $d = 0.0001$ και πάνω.

Τέλος, η ικανότητα κλιμάκωσης του αλγόριθμου παρουσιάζεται στην δεύτερη γραφική του Σχήματος 5.7. Η καμπύλη του χρόνου αυξάνεται σχεδόν γραμμικά με το μέγεθος των δεδομένων $|D|$ και στους δυο αλγόριθμους.

Συνοψίζοντας, ο αλγόριθμος `ACD` καταφέρνει να περιορίσει σημαντικά τις απώλειες πληροφορίας στα δεδομένα. Τα πειραματικά αποτελέσματα έδειξαν ότι η μετρική GCP των δεδομένων που ανωνυμοποιήθηκαν από τον `ACD` είναι η μισή έως το ένα τρίτο εκείνων που ανωνυμοποιήθηκαν από τον `AA`. Το κέρδος αυτό συνοδεύεται από ένα αυξημένο υπολογιστικό κόστος, όμως η επιβάρυνση αυτή είναι περιορισμένη από 20% έως 40% στις περισσότερες περιπτώσεις.

5.5 Συμπεράσματα

Σε αυτό το κεφάλαιο μελετήθηκε το πρόβλημα της k^m-ανωνυμοποίησης δεδομένων με συνεχή γνωρίσματα, χωρίς την χρήση μιας προκαθορισμένης ιεραρχίας γενίκευσης. Προτάθηκε ο αλγόριθμος `ACD`, ένας άπληστος ευρηστικός αλγόριθμος ολικής ανακωδικοποίησης που στοχεύει στην μείωση της απώλειας πληροφορίας. Επιλέγει με άπληστο τρόπο το καλύτερο εύρος γενίκευσης σε κάθε βήμα, διασφαλίζοντας ότι όλοι οι συνδυασμοί τιμών, μεγέθους έως m, θα εμφανίζονται σε τουλάχιστον k εγγραφές των δεδομένων, ικανοποιώντας έτσι την εγγύηση της k^m-ανωνυμίας. Η προτεινόμενη μέθοδος αξιολογήθηκε χρησιμοποιώντας πραγματικά δεδομένα και ο αλγόριθμος συγκρίθηκε με τον `AA` [68] ο οποίος χρησιμοποιεί προκαθορισμένες ιεραρχίες για την k^m-ανωνυμοποίηση. Τα πειραματικά αποτελέσματα δείχνουν ότι ο `ACD` διατηρεί σημαντικά μεγαλύτερη χρησιμότητα και καλύτερη ποιότητα στα τελικά δεδομένα, με αντίτιμο μια μικρή αύξηση του υπολογιστικού κόστους.

Τα αποτελέσματα αυτής της ερευνητικής εργασίας δημοσιεύθηκαν στο [37]. Πιθανές μελλοντικές προεκτάσεις αυτής της εργασίας είναι η ενσωμάτωση αυστηρότερων κριτηρίων στην εγγύηση ιδιωτικότητας για την πρόληψη των επιθέσεων αποκάλυψης γνωρίσματος, αντίστοιχα με τις εγγυήσεις της l-διαφορετικότητας και της t-εγγύτητας.

Κεφάλαιο 6

Επιθέσεις Συναθροιστικής Γνώσης μιας Συνάρτησης

Στο κεφάλαιο αυτό παρουσιάζουμε τη μέθοδο που προτείνουμε για την ανωνυμοποίηση δεδομένων με σκοπό την προστασία τους από επιτιθέμενους με συναθροιστική γνώση. Αρχικά ορίζεται το πρόβλημα και παρουσιάζεται ο τύπος των δεδομένων και η πιθανή γνώση επιτιθέμενου η οποία τα καθιστά ευάλωτα σε μια συναθροιστική επίθεση. Προτείνεται ένας αλγόριθμος ο οποίος ανωνυμοποιεί τα δεδομένα απέναντι σε τέτοιες επιθέσεις και εισάγει μικρή απώλεια πληροφορίας. Η πειραματική αξιολόγηση της μεθόδου πάνω σε πραγματικά δεδομένα δείχνει ότι μπορεί να διασφαλίσει καλύτερη ποιότητα ανωνυμοποιημένων δεδομένων σε σχέση με την κλασσική k-ανωνυμία.

6.1 Κίνητρο και Συνεισφορά

Το κοινό χαρακτηριστικό των σεναρίων επίθεσης που μελετήθηκαν στα προηγούμενα κεφάλαια είναι η ακριβής γνώση του επιτιθέμενου για ένα υποσύνολο της εγγραφής. Όμως τα δεδομένα που δημοσιεύει μια εταιρία ή ένας οργανισμός συχνά περιέχουν τόσο λεπτομερείς τιμές ώστε να είναι απίθανο ένας κακόβουλος επιτιθέμενος να κατέχει ακριβή μερική γνώση μιας εγγραφής, δηλαδή ένα υποσύνολο από τις ακριβείς τιμές των γνωρισμάτων της. Εντούτοις, θα μπορούσε να έχει πιο αφηρημένη ή συναθροιστική γνώση των πεδίων μιας εγγραφής. Τέτοια παραδείγματα προκύπτουν από πολλά είδη δεδομένων στην πράξη, όπως είναι τα φορολογικά δεδομένα. Κάθε εγγραφή περιέχει πολυάριθμα πεδία τα οποία καταγράφουν ένα μεγάλο εύρος οικονομικών δραστηριοτήτων των φορολογούμενων σε πολύ λεπτομερές επίπεδο. Μετά την δημοσίευση τέτοιων δεδομένων, αναμένουμε οι περισσότερες επιθέσεις να προέρχονται από επιτιθέμενους που αναγνωρίζουν τις εγγραφές βασιζόμενοι σε πιο γενική συναθροιστική γνώση, παραδείγματος χάριν το συνολικό εισόδημα, και όχι στις ακριβείς τιμές των επιμέρους πεδίων που είναι δυσκολότερο να τις γνωρίζει εκ των προτέρων, λ.χ. τα ετήσια εισοδήματα από γεωργικές εργασίες.

Το ίδιο σενάριο επίθεσης μπορεί να προκύψει σε πολλές εφαρμογές. Κατά την δημοσίευση δεδομένων κίνησης, ο επιτιθέμενος μπορεί να γνωρίζει την συνολική διάρκεια ενός ταξιδιού,

ΑΦΜ	Μισθός	Ενοίκια	Τόκοι καταθέσεων	Συνολικό εισόδημα
012345001	20K	10K	20K	50K
012345002	10K	20K	15K	45K
012345003	40K	30K	12K	82K
012345004	30K	40K	11K	81K
012345005	20K	20K	22K	62K
012345006	10K	20K	32K	62K

Πίνακας 6.1: Πίνακας φορολογικών δεδομένων.

αλλά είναι λιγότερο πιθανό να ξέρει την ακριβή χρονική στιγμή άφιξης σε κάθε ενδιάμεσο προορισμό ή την διάρκεια κάθε στάσης. Επίσης, κατά την δημοσίευση ιατρικών δεδομένων, ένας επιτιθέμενος μπορεί να γνωρίζει μια προηγούμενη διάγνωση του ασθενούς, που αντιστοιχεί σε ένα συνδυασμό από δείκτες ιατρικών μετρήσεων, αλλά είναι απίθανο να γνωρίζει κάθε τιμή των ιατρικών του εξετάσεων. Η ανωνυμοποίηση τέτοιων δεδομένων χρησιμοποιώντας παραδοσιακές εγγυήσεις ιδιωτικότητας θα μπορούσε να εγγυηθεί την προστασία των εγγραφών, αλλά θα αλλοίωνε περισσότερο από ότι χρειάζεται τις τιμές τους. Στην πραγματικότητα χρειάζεται να δημιουργηθούν κλάσεις ισοδυναμίας μόνο ως προς την πιο γενική συναθροιστική γνώση του επιτιθέμενου.

Ενώ η έρευνα σχετικά με την προστασία της ιδιωτικότητας συχνά εστιάζει στον ορισμό αυστηρότερων εγγυήσεων ιδιωτικότητας που θεωρούν ολοένα πιο εξειδικευμένες επιθέσεις, είναι εξίσου σημαντικό να υπάρχουν εγγυήσεις και μέθοδοι ανωνυμοποίησης για κάθε ενδιάμεση περίπτωση. Η εφαρμογή μιας αυστηρότερης από όσο χρειάζεται εγγύησης θα προκαλέσει άσκοπη αύξηση στην απώλεια πληροφορίας και θα χειροτερεύσει την ποιότητα των δεδομένων.

Παράδειγμα 6.18. *Ας θεωρήσουμε το παράδειγμα του Πίνακα 6.1 όπου παρουσιάζονται τα φορολογικά στοιχεία κάποιων προσώπων. Ένα ρεαλιστικό σενάριο επίθεσης είναι όταν ο επιτιθέμενος γνωρίζει μόνο προσεγγιστικά την κλάση συνολικού εισοδήματος του στόχου, αλλά όχι πιο λεπτομερή πληροφορία για τις επιμέρους πηγές εισοδημάτων. Συνεπώς, σε τέτοιες περιπτώσεις δεν είναι η ακριβείς τιμές των γνωρισμάτων που δρουν ως ψευδο-αναγνωριστικά, αλλά η συναθροιστική πληροφορία που υπολογίζεται πάνω σε αυτές.*

Ας υποθέσουμε ότι ένας κακόβουλος επιτιθέμενος γνωρίζει ότι το συνολικό ετήσιο εισόδημα του στόχου κυμαίνεται μεταξύ 48 και 52 χιλιάδες. Αν τα δεδομένα του Πίνακα 6.1 δημοσιευτούν ως έχουν, αφαιρώντας μόνο τα μοναδικά αναγνωριστικά (ΑΦΜ), τότε ο επιτι-

Id	Μισθός	Ενοίκια	Τόκοι καταθέσεων	Συνολικό εισόδημα
1	[10-20]	[10-20]	[10-20]	[30-60]
2	[10-20]	[10-20]	[10-20]	[30-60]
3	[30-40]	[30-40]	[10-20]	[70-100]
4	[30-40]	[30-40]	[10-20]	[70-100]
5	[10-20]	[10-20]	[20-32]	[40-72]
6	[10-20]	[10-20]	[20-32]	[40-72]

Πίνακας 6.2: Κλασική 2-ανωνυμοποίηση του Πίνακα 6.1.

Id	Μισθός	Ενοίκια	Τόκοι καταθέσεων	Συνολικό εισόδημα
1	20	10	[15-20]	[45-50]
2	10	20	[15-20]	[45-50]
3	40	30	[11-12]	[81-82]
4	30	40	[11-12]	[81-82]
5	20	20	22	62
6	10	20	32	62

Πίνακας 6.3: 2-Ανωνυμοποίηση του Πίνακα 6.1 για προστασία από επιθέσεις με συναθροιστική γνώση (άθροισμα).

θέμενος μπορεί να υπολογίσει το συνολικό εισόδημα κάθε εγγραφής και να συμπεράνει ότι η πρώτη εγγραφή ανήκει στον στόχο. Ο 2-ανώνυμος Πίνακας 6.2 αποτελεί την κλασσική k-ανωνυμοποίηση του Πίνακα 6.1. Η τελευταία στήλη (συνολικό εισόδημα) δεν χρειάζεται να δημοσιευθεί, προκύπτει άμεσα ως το άθροισμα των υπολοίπων πεδίων. Οποιοσδήποτε επιτιθέμενος με συναθροιστική γνώση δεν θα μπορεί να ταυτοποιήσει λιγότερες από 2 εγγραφές. Εντούτοις, το ίδιο επίπεδο ιδιωτικότητας διασφαλίζεται και από τον Πίνακα 6.3 όπου οι ομάδες εγγραφών {1, 2}, {3, 4}, καθώς και {5, 6} αποτελούν κλάσεις ισοδυναμίας ως προς την συναθροιστική γνώση του επιτιθέμενου. Παρατηρούμε ότι δεν έχουν γενικευθεί όλες οι τιμές και έτσι έχει περιοριστεί η απώλεια πληροφορίας. Και στις δύο περιπτώσεις, ο επιτιθέμενος δεν μπορεί να διακρίνει την πρώτη εγγραφή-στόχο από την δεύτερη. Όμως ο Πίνακας 6.3 περιέχει πολύ πιο λεπτομερή πληροφορία για αυτές χωρίς να παραβιάζεται η ιδιωτικότητα ως προς την γνώση του επιτιθέμενου.

Σε αυτό το κεφάλαιο προτείνουμε μια παραλλαγή της k-ανωνυμίας για την πρόληψη επιθέσεων που βασίζονται σε συναθροιστική γνώση και οδηγούν σε αποκάλυψη ταυτότητας. Η βασική ιδέα της προτεινόμενης μεθόδου είναι η ομαδοποίηση των εγγραφών σε κλάσεις ισοδυναμίας που έχουν παρόμοιες τιμές της συναθροιστικής συνάρτησης που γνωρίζει ο επιτιθέμενος. Για να επιτευχθεί αυτό εφαρμόζουμε τοπικά γενικεύσεις τιμών, ανεξάρτητες μέσα σε κάθε ομάδα εγγραφών. Συγκριτικά με την k-ανωνυμία, ακόμα και για αλγόριθμους τοπικής ανακωδικοποίησης, επιτυγχάνουμε καλύτερη ποιότητα αποτελεσμάτων καθώς οι κλάσεις ισοδυναμίας που δημιουργούμε δεν απαιτούν οι εγγραφές να είναι πανομοιότυπες ως προς όλα τα γνωρίσματα.

Η συνεισφορά του Κεφαλαίου 6 συνοψίζεται στα ακόλουθα σημεία:

- Ορίζουμε το πρόβλημα της ανωνυμοποίησης δεδομένων ως προς την συναθροιστική γνώση επίθεσης,
- Διατυπώνουμε την *k^f-ανωνυμία* για την εγγύηση της προστασίας ιδιωτικότητας απέναντι σε επιθέσεις συναθροιστικής γνώσης,
- Προτείνουμε έναν αλγόριθμο ανωνυμοποίησης με στόχο την διατήρηση καλύτερης ποιότητας των τελικών ανώνυμων δεδομένων,

- Αξιολογούμε την μέθοδο με πραγματικά δεδομένα και συγκρίνουμε τα πειραματικά αποτελέσματα με τον Mondrian έναν αλγόριθμο τοπικής ανακωδικοποίησης για πολυδιάστατη k-ανωνυμία.

6.2 Ορισμός του Προβλήματος

6.2.1 Μοντέλο Δεδομένων

Θεωρούμε δεδομένα με αριθμητικές τιμές πάνω στις οποίες μπορεί να οριστεί μια συναθροιστική συνάρτηση, όπως άθροισμα, μέσος όρος, κτλ. Το αποτέλεσμα αυτής της συνάρτησης όταν οριστεί πάνω στις τιμές μιας εγγραφής αποτελεί πιθανή γνώση του επιτιθέμενου, την οποία μπορεί να χρησιμοποιήσει για να αναγνωρίσει μια εγγραφή. Κάθε εγγραφή αντιστοιχεί σε ένα πραγματικό άτομο. Οι εγγραφές μπορεί να ανήκουν σε σχεσιακούς πίνακες ή να έχουν κάποια πιο χαλαρή δομή όπως είναι τα ημιδομημένα δεδομένα, XML κ.α. Στο κεφάλαιο αυτό εξετάζουμε για λόγους απλότητας την περίπτωση του ενός σχεσιακού πίνακα D, όπως ο Πίνακας 6.4, με γνωρίσματα τα $QI_1, QI_2, ..., QI_n$ τα οποία παίρνουν αριθμητικές τιμές από ένα κοινό πεδίο $\mathcal{I}$.

6.2.2 Μοντέλο Επίθεσης

Θεωρούμε επιτιθέμενους των οποίων η γνώση περιορίζεται στην τιμή μιας συναθροιστικής συνάρτησης που υπολογίζεται πάνω στα πεδία μιας εγγραφής στόχου t. Η πιθανή γνώση του επιτιθέμενου μοντελοποιείται ως μια συναθροιστική συνάρτηση η οποία ορίζεται πάνω στις τιμές των ψευδο-αναγνωριστικών $f(QI_1, QI_2, ..., QI_n) : \mathcal{I}^n \to \mathbb{R}$. Το πεδίο τιμών της f είναι το σύνολο της πιθανής γνώσης που μπορεί να γνωρίζει ο επιτιθέμενος για κάποια από τις $|D|$ εγγραφές. Κατά τον υπολογισμό της $f(QI_1, QI_2, ..., QI_n)$ σε όλες τις εγγραφές, όταν κάποια τιμή f_i είναι μοναδική ή σπάνια τότε υπάρχει κίνδυνος παραβίασης της ιδιωτικότητας. Οι τιμές των πεδίων του πίνακα D πρέπει να μετασχηματιστούν κατά τέτοιο τρόπο ώστε τουλάχιστον k εγγραφές να φαίνονται ότι ικανοποιούν τη συνθήκη $f(qi_1, qi_2, ..., qi_n) = f_i$.

6.2.3 Εγγύηση Ιδιωτικότητας

Παρέχουμε μια νέα εγγύηση ιδιωτικότητας ώστε να αντιμετωπιστούν επιθέσεις που περιορίζονται στην γνώση μιας συναθροιστικής συνάρτησης των γνωρισμάτων μιας εγγραφής:

QI_1	QI_2	...	QI_n	S	$f(QI_1, ..., QI_n)$
qi_{11}	qi_{12}	...	qi_{1n}	s_1	f_1
qi_{21}	qi_{22}	...	qi_{2n}	s_2	f_2
qi_{31}	qi_{32}	...	qi_{3n}	s_3	f_3
$\vdots$	$\vdots$		$\vdots$	$\vdots$	$\vdots$
qi_{m1}	qi_{m2}	...	qi_{mn}	s_m	f_m

Πίνακας 6.4: Μοντελοποίηση δεδομένων και γνώσης επιτιθέμενου.

Ορισμός 6.11. *(εγγύηση συναθροιστικής ιδιωτικότητας) Ένα σύνολο δεδομένων D θεωρείται k^f-ανώνυμο αν οποιοσδήποτε επιτιθέμενος που γνωρίζει την τιμή της μιας συναθροιστικής συνάρτησης f ορισμένης πάνω στις τιμές μιας εγγραφής $t \in D$, δεν θα μπορεί να αντιστοιχίσει αυτή την γνώση σε λιγότερες από k εγγραφές του D.*

Η εγγύηση αυτή αποτελεί μια επέκταση της κλασσικής k-ανωνυμίας [63, 66] προσαρμοσμένης στο παραπάνω σενάριο επίθεσης. Η χρήση του *συνολικού εισοδήματος* ($f = sum$) για να ανακαλύψουμε τα λεπτομερή οικονομικά στοιχεία ενός φορολογούμενου, ή η χρήση του μέσου όρου ($f = average$) για να ανακαλύψουμε την επιμέρους βαθμολογία ενός μαθητή είναι ρεαλιστικά παραδείγματα αυτού του είδους επίθεσης.

Συχνά, ο επιτιθέμενος δεν γνωρίζει ούτε την ακριβή συναθροιστική τιμή με 100% ακρίβεια, αλλά μια προσέγγισή της. Παραδείγματος χάριν, μπορεί να μην γνωρίζει ότι το ακριβές *συνολικό εισόδημα* ενός φορολογούμενου είναι 53,415.22 με ακρίβεια λεπτού. Αντίθετα, μπορεί να γνωρίζει ότι το εισόδημα είναι της τάξης των 50,000 με 55,000. Η παραδοχή αυτή μας επιτρέπει να προτείνουμε ακόμα μια πιο ευέλικτη παραλλαγή της k^f-ανωνυμίας που διατηρεί καλύτερη ποιότητα δεδομένων, όπως θα φανεί στην ενότητα της πειραματικής αξιολόγησης.

Ορισμός 6.12. *(χαλαρή εγγύηση συναθροιστικής ιδιωτικότητας) Ένα σύνολο δεδομένων D θεωρείται $k^{(f,d)}$-ανώνυμο αν οποιοσδήποτε επιτιθέμενος που γνωρίζει την τιμή της μιας συναθροιστικής συνάρτησης f ορισμένης πάνω στις τιμές μιας εγγραφής $t \in D$, υπάρχει πραγματικό θετικό $d < 1$ τέτοιο ώστε τουλάχιστον k εγγραφές του D έχουν τιμές για την συναθροιστική συνάρτηση που περιέχει το εύρος $f \cdot (1 \pm d)$.*

Αν ένα σύνολο δεδομένων D δεν είναι k^f-ανώνυμο, μπορεί να μετασχηματιστεί σε ένα $D^\star$, έτσι ώστε το $D^\star$ να ικανοποιεί την εγγύηση της *k^f-ανωνυμίας*. Οι μετασχηματισμοί των εγγραφών περιλαμβάνουν τις γενικεύσεις εκείνων των τιμών που απαιτούνται ώστε να δημιουργηθούν ομάδες από τουλάχιστον k εγγραφές με πανομοιότυπες τιμές για την συνάρτηση f ή πανομοιότυπα εύρη τιμών της f, όπως φαίνεται στον Πίνακα 7.3. Όπως έχει οριστεί και στα προηγούμενα κεφάλαια, η γενίκευση είναι ένα σύνολο κανόνων της μορφής $v \to [a, b]$, όπου μια αρχική τιμή v αντικαθίσταται από ένα εύρος τιμών που την περιέχει.

Υπάρχουν πολλές πιθανές ανωνυμοποιήσεις ενός συνόλου δεδομένων που να ικανοποιούν την k^f-ανωνυμία για μια δοσμένη συνάρτηση f, όπως προκύπτει και από τους Πίνακες 7.3 και 7.2. Παραδείγματος χάριν, η ανωνυμοποίηση όλων των τιμών στο μέγιστο εύρος του πεδίου τιμών $\mathcal{I}$ είναι μια υποψήφια λύση, αλλά θα προσέθετε την μέγιστη απώλεια πληροφορίας καθιστώντας άχρηστα τα τελικά δεδομένα.

Το πρόβλημα της k^f-ανωνυμοποίησης ενός συνόλου δεδομένων είναι ένα πρόβλημα βελτιστοποίησης, όπου αναζητείται το σύνολο των κανόνων γενίκευσης που ικανοποιούν την k^f-ανωνυμία και ταυτόχρονα διατηρούν όσο το δυνατόν περισσότερη χρηστική πληροφορία στα δεδομένα.

Αλγόριθμος 11 aggrAnon $\texttt{AA}(D, f, k)$

Require: D {Original Dataset}, f {aggregate function},
k {privacy parameter}
Ensure: $D^\star$ {k^f-anonymous Dataset.}

for all tuples $t < qi_1, qi_2, ..., qi_n > \in D$ **do**
  estimate $f(qi_1, qi_2, ..., qi_n)$
sort tuples with reference to their f values.
form groups of size $\geq k$ and $\leq 2k-1$
for every group EC **do**
  if all tuples have the same f value **then**
    add EC to $D^\star$.
  else
    $Q = \{QI_1, QI_2, ..., QI_n\}$ //Q contains all attributes
    $j = n - 1$
    while Q not empty **do**
      estimate f for all combinations of j attributes
      Let C_j be the combination with most similar f
          for all tuples
      generalize the remaining attribute $QI_j = Q \backslash C_j$
          to a common range $[v_{min}, v_{max}]$
      remove QI_j from Q
      $j = j - 1$
      estimate f for all tuples in EC
      if all tuples have the same f value **then**
        break
    add EC to $D^\star$
return $D^\star$

6.3 Αλγόριθμος Ανωνυμοποίησης

6.3.1 Χώρος Λύσεων

Ο χώρος των λύσεων αποτελείται από το σύνολο όλων των πιθανών κανόνων γενίκευσης, όπως και στην κλασσική k-ανωνυμία. Εντούτοις, το σύνολο των αποδεκτών λύσεων που ικανοποιούν την εγγύησή μας είναι σημαντικά μεγαλύτερο. Το πρόβλημα της βέλτιστης πολυδιάστατης k-ανωνυμίας έχει αποδειχθεί πως είναι NP-hard [53]. Στην χειρότερη περίπτωση, όταν η συνάρτηση λαμβάνει διαφορετική τιμή για κάθε συνδυασμό τιμών των γνωρισμάτων, το πρόβλημα είναι το ίδιο. Λόγω της πολυπλοκότητας του προβλήματος της βέλτιστης ανωνυμοποίησης, αναπτύξαμε έναν ευρηστικό αλγόριθμο. Ομαδοποιούμε τις εγγραφές σε κλάσεις ισοδυναμίας και εφαρμόζουμε γενικεύσει *τοπικής ανακωδικοποίησης* στις τιμές σε κάθε κλάση ισοδυναμίας ξεχωριστά.

6.3.2 Αλγόριθμος

Ο αλγόριθμος ανωνυμοποίησης που αναπτύξαμε λειτουργεί σε δύο φάσεις, όπως φαίνεται από τον ψευδοκώδικα του Αλγόριθμου 11. Η πρώτη φάση χωρίζει τις εγγραφές σε ομάδες (γραμμές 1-4). Οι κλάσεις ισοδυναμίας σχηματίζονται ως προς την τιμή της συνάρτησης f. Αρχικά, όλες οι εγγραφές ταξινομούνται ως προς την τιμή της $f(qi_2, qi_2, ..., qi_n)$. Στη συνέχεια, χωρίζονται σε κλάσεις ισοδυναμίας, όπου η καθεμία έχει μέγεθος $k \leq |EC| \leq 2k-1$. Ο περιορισμός του μεγέθους των κλάσεων γίνεται για να αποφευχθεί η υπερβολική γενίκευση των τιμών.

Κατά την δεύτερη φάση θεωρούμε κάθε κλάση ισοδυναμίας ξεχωριστά, και εφαρμόζουμε γενικεύσεις στις τιμές των εγγραφών της (γραμμές 6-16). Αν όλες οι εγγραφές μιας κλάσης ισοδυναμίας EC ήδη λαμβάνουν την ίδια τιμή για την συνάρτηση f, τότε η κλάση EC προστίθεται απευθείας στο ανώνυμο αποτέλεσμα $D^\star$. Διαφορετικά, αναζητούμε το ελάχιστο σύνολο γενικεύσεων ώστε να γίνουν όλες οι εγγραφές της EC αδιαχώριστες ως προς την συνάρτηση f. Έστω ότι ο αριθμός των διαστάσεων του πίνακα των δεδομένων είναι n, δηλαδή το πλήθος των γνωρισμάτων μιας εγγραφής. Υπολογίζουμε την τιμή της f για όλους τους συνδυασμούς των $n-1$ γνωρισμάτων, δηλαδή όλων πλην *ενός* γνωρίσματος κάθε φορά. Έστω $\{Q = QI_1, .., QI_{j-1}, QI_{j+1}, ..., QI_n\}$ είναι το σύνολο των $(n-1)$ γνωρισμάτων για τα οποία η τιμή της f είναι περισσότερο όμοια ανάμεσα στις εγγραφές της EC. Τότε, γενικεύουμε το γνώρισμα QI_j στο εύρος τιμών $[v_{min}, v_{max}]$, όπου v_{min} και v_{max} είναι η ελάχιστη και η μέγιστη τιμή που εμφανίζονται για το γνώρισμα QI_j μέσα στις εγγραφές της κλάσης EC. Τότε αφαιρούμε το QI_j από το σύνολο Q και συνεχίζουμε την αναζήτηση με τα υπόλοιπα γνωρίσματα από το Q. Υπολογίζουμε την τιμή της f για όλους τους συνδυασμούς $n-2$ γνωρισμάτων από το Q κα επαναλαμβάνουμε την παραπάνω διαδικασία έως ότου όλες οι εγγραφές της EC να λαμβάνουν την ίδια τιμή ή εύρος τιμών για την συνάρτηση f.

Η επιλογή του γνωρίσματος που θα γενικευθεί σε κάθε βήμα γίνεται με τον υπολογισμό της $f(t)$ για κάθε εγγραφή $t \in EC$, εξαιρώντας ένα γνώρισμα κάθε φορά, όπως περιγράφηκε παραπάνω. Η τιμή της f όταν εξαιρείται από τον υπολογισμό το i-στό γνώρισμα σημειώνεται ως f_i. Λαμβάνουμε υπόψη όλες τις διαφορές $f_i(t) - f_i(t')$, για κάθε πιθανό ζεύγος εγγραφών $t, t' \in EC$ $(t \neq t')$ και υπολογίζουμε τον μέσο όρο τους $avg(f_i(t) - f_i(t'))$. Επιλέγουμε εκείνο το γνώρισμα j που αντιστοιχεί στον ελάχιστο $avg(f_j(t) - f_j(t'))$. Για λόγους απλότητας θεωρούμε ότι όλα τα γνωρίσματα είναι εξίσου σημαντικά και εξίσου ευαίσθητα. Η προσαρμογή της μεθόδου λαμβάνοντας υπόψη διαφορετικά βάρη για κάθε γνώρισμα, είναι προφανής.

Ο Αλγόριθμος 11 μπορεί να τροποποιηθεί ώστε να ικανοποιεί την χαλαρότερη εγγύηση της $k^{(f,d)}$-ανωνυμίας, προσαρμόζοντας την συνθήκη τερματισμού στην γραμμή 18 ώστε να απαιτείται όλες οι εγγραφές να λαμβάνουν τιμές της συναθροιστικής συνάρτησης στο εύρος $f \cdot (1 \pm d)$.

Ορθότητα

Μπορεί να αποδειχθεί ότι ο Αλγόριθμος 11 πάντα επιστρέφει ένα k^f–ανώνυμο σύνολο δεδομένων, δείχνοντας ότι η έξοδός του δεν μπορεί να περιέχει καμία εγγραφή t η οποία να μοιράζεται την ίδια τιμή της f, την $f(t)$ με λιγότερες από $k-1$ άλλες εγγραφές. Αυτό μπορεί

να συμβεί αν είτε (α) η κλάση ισοδυναμίας της έχει μέγεθος μικρότερο του k, είτε (β) η κλάση ισοδυναμίας της δεν είναι επαρκώς ανωνυμοποιημένη. Ο αλγόριθμος από την αρχή χωρίζει τις εγγραφές σε ομάδες μεγέθους $\geq k$, στην γραμμή 4, και κάθε τέτοια ομάδα αποτελεί μι α κλάση ισοδυναμίας που ανωνυμοποιείται ξεχωριστά. Επομένως, το (α) δεν ισχύει ποτέ, διότι είναι αδύνατο για μια πλειάδα α βρεθεί σε κλάση ισοδυναμίας με μέγεθος μικρότερο από k εγγραφές. Η μόνη προϋπόθεση για αυτό είναι να επιλεχθεί το k έτσι ώστε ισχύει $k < |D|$, κάτι που ισχύει πάντα στην πράξη. Ο αλγόριθμος γενικεύει προοδευτικά κάθε γνώρισμα μέσα στην κλάση ισοδυναμίας σε ένα κοινό εύρος για όλες τις εγγραφές της κλάσης, έως να γίνει ίδια η τιμή της $f(t)$ για κάθε εγγραφή στην κλάση. Δεδομένου ότι οι κανόνες γενίκευσης επιτρέπουν πάντα την γενίκευση όλων των τιμών ενός γνωρίσματος σε ένα κοινό εύρος και $f(t_1) = f(t_2)$ για $t_1 = t_2$, τότε υπάρχει πάντα η προφανής λύση της γενίκευσης όλων των γνωρισμάτων μέσα στην κλάση ισοδυναμίας στο ίδιο εύρος τιμών, δημιουργώντας έτσι πανομοιότυπες εγγραφές. Συνεπώς, δεν μπορεί να υπάρξει καμία εγγραφή μη-διαχωρίσιμη από λιγότερες από $k-1$ άλλες ως προς την τιμή της συνάρτησης f.

Σημειώνεται ότι στην χειρότερη περίπτωση όλες οι κλάσεις ισοδυναμίας θα γίνουν εντελώς k-ανώνυμες. Αυτό συμβαίνει αν όλα τα γνωρίσματα γενικευθούν στο εύρος $[min, max]$ της ελάχιστης και μέγιστης τιμής που εμφανίζονται μέσα σε κάθε κλάση ισοδυναμίας. Συνεπώς, το άνω όριο της απώλειας πληροφορίας που εισάγει ο προτεινόμενος αλγόριθμος είναι το κόστος που θα εισήγαγε ένας αλγόριθμος k-ανωνυμίας που εφαρμόζει γενικεύσεις με τοπική ανακωδικοποίηση.

6.4 Πειραματική Μελέτη

Σε αυτή την ενότητα παρουσιάζονται τα πειραματικά αποτελέσματα για την αποτίμηση της μεθόδου. Όλες οι υλοποιήσεις έγιναν σε γλώσσα προγραμματισμού C++ και η διεξαγωγή των πειραμάτων έγινε σε υπολογιστή Intel Core i7 CPU 3.2GHz, με 6GB RAM, σε λειτουργικό σύστημα Ubuntu Linux.

6.4.1 Αλγόριθμοι

Συγκρίνουμε τον αλγόριθμο `aggrAnon` με τον `Mondrian` [46] έναν αλγόριθμο πολυδιάστατης k-ανωνυμίας που εφαρμόζει γενικεύσεις τοπικής ανακωδικοποίησης. Χρησιμοποιήθηκε η υλοποίηση από το [11], γραμμένη σε γλώσσα προγραμματισμού java. Τα πειραματικά αποτελέσματα δείχνουν ότι ο `aggrAnon` διατηρεί καλύτερη ποιότητα στα τελικά δεδομένα από ότι ο `Mondrian`, όταν θεωρούμε επιτιθέμενους των οποίων η γνώση περιορίζεται στην τιμή μιας συναθροιστικής συνάρτησης που ορίζεται στα γνωρίσματα των εγγραφών. Η συναθροιστική συνάρτηση που χρησιμοποιήθηκε στα πειράματά είναι το άθροισμα (sum). Δεδομένου ότι οι υλοποιήσεις είναι διαφορετικές δεν είναι δίκαιο να συγκρίνουμε τους δύο αλγόριθμους ως προς το υπολογιστικό κόστος. Εντούτοις, αναφέρουμε ότι ο `aggrAnon` είναι κατά μέσο όρο ταχύτερος κατά τουλάχιστον μια τάξη μεγέθους σε όλα τα πειράματα που διεξήχθησαν.

6.4.2 Πειραματικά Δεδομένα

Χρησιμοποιήσαμε το σύνολο δεδομένων IPUMS από την αποθήκη δεδομένων UCI data mining repository [9]. Επιλέχθηκαν 7 αριθμητικά γνωρίσματα που αντιστοιχούν σε διαφορετικούς τύπους εισοδημάτων. Τα χαρακτηριστικά του συνόλου δεδομένων φαίνονται στον Πίνακα 6.5.

6.4.3 Παράμετροι

Μελετάμε την συμπεριφορά του προτεινόμενου αλγορίθμου ως προς τις ακόλουθες παραμέτρους: (α) την παράμετρο ανωνυμίας k, (β) την παράμετρο d που ποσοτικοποιεί τον βαθμό ακρίβειας της γνώσης του επιτιθέμενου, (γ) το μέγεθος των δεδομένων $|D|$ και (δ) το πλήθος των διαστάσεων dim. Σε κάθε πείραμα μεταβάλλεται μια από αυτές τις παραμέτρους ενώ οι υπόλοιπες κρατούνται σταθερές. Η προεπιλεγμένες τιμές των παραμέτρων που επιλέξαμε είναι $k = 10$, $m = 2$, $d = 0\%$, $|D| = 233584$ και $dim = 7$.

6.4.4 Μετρικές Αποτίμησης

Για την αξιολόγηση της μεθόδου μας μετρήθηκε ο χρόνος εκτέλεσης των πειραμάτων σε δευτερόλεπτα καθώς και η απώλειας πληροφορίας με την μετρική GCP (βλ. Ενότητα 5.3.3), για την αποτίμηση της ποιότητας των αποτελεσμάτων.

6.4.5 Ποιότητα αποτελεσμάτων

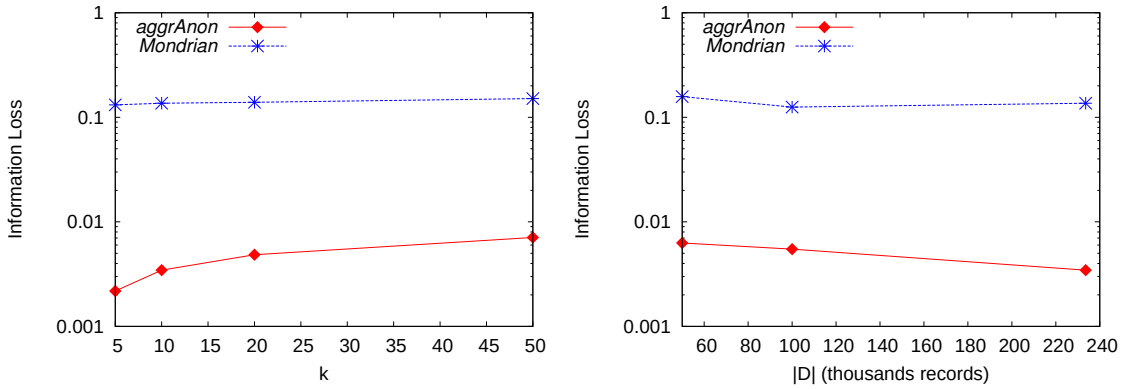
Στο Σχήμα 6.1 συγκρίνουμε την συμπεριφορά των δυο αλγορίθμων ως προς την απώλεια πληροφορίας που προκαλείται από τις γενικεύσεις των τιμών. Η κλίμακα του κατακόρυφου άξονα είναι λογαριθμική. Καθώς η παράμετρος k μεγαλώνει, η GCP και των δυο αλγορίθμων αυξάνεται υπογραμμικά. Όμως ο `aggrAnon` (ακόμα και για $d = 0$) διατηρεί σημαντικά περισσότερη χρησιμότητα στα δεδομένα από τον `Mondrian` για κάθε τιμή της k, όπως ήταν αναμενόμενο. Η απώλεια πληροφορίας που εισάγει ο `aggrAnon` είναι τουλάχιστον μια τάξη μεγέθους μικρότερη από τον `Mondrian` σε όλα τα πειραματικά αποτελέσματα.

Στην δεύτερη γραφική παράσταση του Σχήματος 6.1 μεταβάλλεται το μέγεθος των δεδομένων $|D|$. Για την διεξαγωγή αυτού του πειράματος δημιουργήθηκαν υποσύνολα με τυχαία δειγματοληψία, μεγέθους 100,000 και 50,000 εγγραφών αντίστοιχα. Το δεύτερο είναι υποσύνολο του πρώτου. Παρατηρούμε ότι η απώλεια πληροφορίας του `aggrAnon` φθίνει μονότονα με το πλήθος των εγγραφών $|D|$ και σημαντικά μικρότερη του `Mondrian` κατά τουλάχιστον μια τάξη μεγέθους.

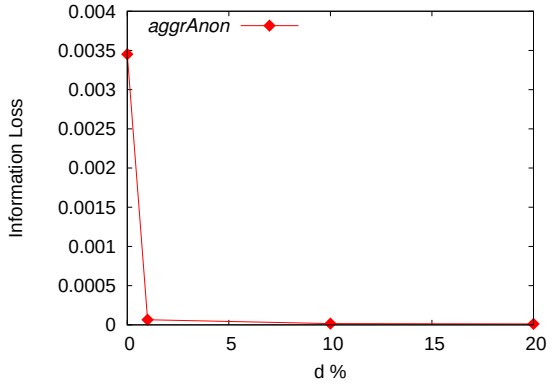
Στο Σχήμα 6.2 καθώς το ποσοστό της αβεβαιότητας του επιτιθέμενου d αυξάνεται, η απώλεια πληροφορίας μειώνεται δραματικά. Ακόμα και για $d = 1\%$ η απώλεια πληροφορίας

Δεδομένα	Εγγραφές	Γνωρίσματα	Εύρος πεδίου τιμών
ipums	233,584	7	1010000

Πίνακας 6.5: Περιγραφή των Δεδομένων



Σχήμα 6.1: Σύκριση της απώλειας πληροφορίας των **aggrAnon** και **Mondrian** ως προς τις παραμέτρους k και $|D|$.



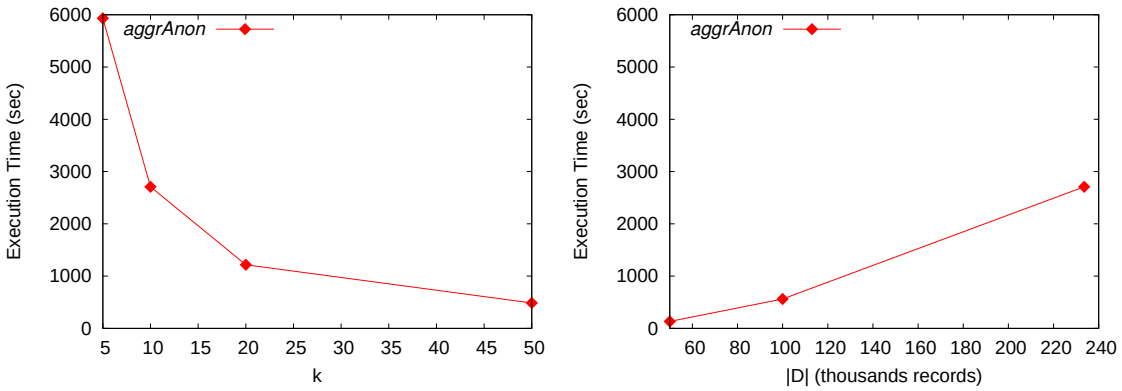
Σχήμα 6.2: Απώλεια πληροφορίας του **aggrAnon** ως προς την παράμετρο αβεβαιότητας του επιτιθέμενου d.

γίνεται 35 φορές μικρότερη από ότι για $d = 0$. Δεν έγινε σύγκριση με τον **Mondrian** καθώς δεν διαθέτει αντίστοιχη παράμετρο για χαλάρωση της εγγύησης.

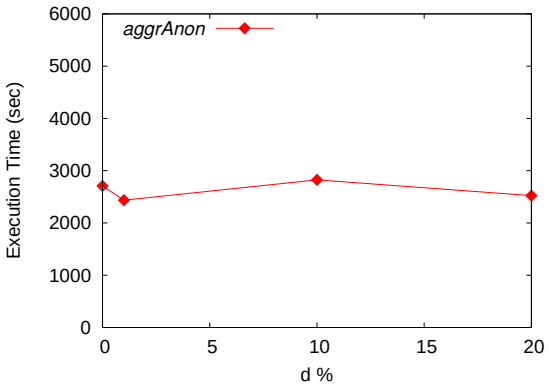
6.4.6 Χρόνος Εκτέλεσης

Τα Σχήματα 6.3 και 6.4 απεικονίζουν το υπολογιστικό κόστος του προτεινόμενου αλγόριθμου. Ο χρόνος εκτέλεσης είναι μεγαλύτερος για μικρές τιμές της παραμέτρου k, αλλά φθίνει υπεργραμμικά καθώς το k αυξάνεται. Η συμπεριφορά αυτή οφείλεται στο γεγονός ότι τα δεδομένα είναι χωρισμένα σε λιγότερες κλάσεις ισοδυναμίας μεγαλύτερου μεγέθους. Κάθε τέτοια κλάση EC ανωνυμοποιείται ξεχωριστά. Συνεπώς, απαιτούνται λιγότερα βήματα από τον αλγόριθμο.

Η κλιμάκωση του αλγορίθμου ως προς το μέγεθος της εισόδου φαίνεται στη δεύτερη γραφική παράσταση του Σχήματος 6.3. Ο χρόνος εκτέλεσης αυξάνεται γραμμικά με το πλήθος των εγγραφών $|D|$, για σταθερό k. Ο λόγος που συμβαίνει αυτό είναι διότι περισσότερες εγγραφές θα δημιουργήσουν περισσότερες κλάσεις ισοδυναμίας ίδιου μεγέθους, που η καθεμία εξετάζεται ξεχωριστά από τις υπόλοιπες. Η συμπεριφορά αυτή οφείλεται στο γεγονός ότι ο



Σχήμα 6.3: Χρόνος εκτέλεσης ως τις παραμέτρους k και $|D|$.



Σχήμα 6.4: Χρόνος εκτέλεσης ως προς την παράμετρο αβεβαιότητας του επιτιθέμενου d.

αλγόριθμος εξετάζει τις κλάσεις ισοδυναμίας σειριακά. Είναι προφανές ότι η απλή παραλληλοποίηση του αλγορίθμου μετά τη δημιουργία των κλάσεων ισοδυναμίας, θα μείωνε σημαντικά το υπολογιστικό κόστος.

Τέλος, στο Σχήμα 6.4 παρατηρούμε ότι ο χρόνος εκτέλεσης δεν επηρεάζεται σημαντικά από την μεταβολή της παραμέτρου d που ποσοτικοποιεί τον βαθμό ακρίβειας της γνώσης του επιτιθέμενου.

6.5 Συμπεράσματα

Σε αυτό το κεφάλαιο μελετήθηκε το πρόβλημα της ανωνυμοποίησης δεδομένων ως προς την συναθροιστική γνώση ενός κακόβουλου επιτιθέμενου. Για την αντιμετώπιση τέτοιων επιθέσεων ορίστηκε μια παραλλαγή της κλασσικής εγγύησης της k-ανωνυμίας, η k^f-ανωνυμία. Προτάθηκε ο `aggrAnon`, ένας αλγόριθμος ανωνυμοποίησης που περιορίζει την απώλεια πληροφορίας των δεδομένων, επιλέγοντας άπληστα μια λύση που ικανοποιεί την προτεινόμενη εγγύηση. Τα πειραματικά αποτελέσματα πάνω σε πραγματικά δεδομένα έδειξαν ότι ο `aggrAnon` διατηρεί σημαντικά μεγαλύτερη χρησιμότητα και καλύτερη ποιότητα στα τελικά δεδομένα σε σύγκριση

με τον `Mondrian` [46], έναν αλγόριθμο k-ανωνυμοποίησης με τοπική ανακωδικοποίηση.

Τα αποτελέσματα αυτής της ερευνητικής εργασίας δημοσιεύθηκαν στο [35]. Από ότι γνωρίζουμε, αυτή είναι η πρώτη εργασία η οποία μελετά σενάρια επίθεσης με συναθροιστική γνώση. Άμεση μελλοντική προέκταση αυτής της εργασίας είναι η μελέτη πιο πολύπλοκων σεναρίων επίθεσης που περιλαμβάνουν την γνώση πολλαπλών συναρτήσεων που καθεμία ορίζεται πάνω σε οποιοδήποτε υποσύνολο των πραγματικών τιμών των γνωρισμάτων μιας εγγραφής. Η αντιμετώπιση τέτοιων επιθέσεων αποτελεί τον βασικό στόχο του επόμενου κεφαλαίου. Άλλες πιθανές κατευθύνσεις για μελλοντική έρευνα είναι η επέκταση της μεθόδου σε αυστηρότερες εγγυήσεις ιδιωτικότητας για την πρόληψη των επιθέσεων αποκάλυψης γνωρίσματος, αντίστοιχα με τις εγγυήσεις της l-διαφορετικότητας και της t-εγγύτητας.

Κεφάλαιο 7

Επιθέσεις Συναθροιστικής Γνώσης Πολλαπλών Συναρτήσεων

Στο κεφάλαιο αυτό παρουσιάζουμε τη μέθοδο που προτείνουμε για την ανωνυμοποίηση δεδομένων με σκοπό την προστασία τους από επιτιθέμενους με γνώση διάφορων συναρτήσεων που ορίζονται πάνω στα δεδομένα. Επεκτείνουμε την εργασία του προηγούμενου κεφαλαίου για α καλύψουμε τα σενάρια που ο επιτιθέμενος μπορεί να γνωρίζει περισσότερες από μια συναρτήσεις, ενώ το είδος αυτής της γνώσης δεν περιορίζεται σε απλές συναθροιστικές (πρόσθεση, μέσος όρος), αλλά σε περισσότερους τύπους συναρτήσεων που μπορεί να οριστούν πάνω στις αριθμητικές τιμές των δεδομένων. Αρχικά ορίζεται το πρόβλημα και παρουσιάζεται ο τύπος των δεδομένων και μοντελοποιείται η πιθανή γνώση επιτιθέμενου ως ένα σύστημα εξισώσεων με μεταβλητές τα πεδία της εγγραφής. Προτείνονται διάφοροι εναλλακτικοί αλγόριθμοι οι οποίοι ανωνυμοποιούν τα δεδομένα απέναντι σε τέτοιες επιθέσεις και εισάγουν μικρότερη απώλεια πληροφορίας. Τέλος, οι αλγόριθμοι αυτοί συγκρίνονται και αξιολογούνται πειραματικά πάνω σε πραγματικά σύνολα δεδομένων. Η πειραματική αξιολόγηση της προτεινόμενης μεθόδου δείχνει ότι η μπορεί να διασφαλίσει καλύτερη ποιότητα ανωνυμοποιημένων δεδομένων σε σχέση με την κλασσική k-ανωνυμία.

7.1 Κίνητρο και Συνεισφορά

Πολλοί οργανισμοί, εταιρίες και δημόσιοι φορείς συλλέγουν καθημερινά μεγάλο όγκο δεδομένων των χρηστών τους σε ολοένα και πιο λεπτομερές επίπεδο. Η δημοσίευση τέτοιων δεδομένων είναι χρήσιμη για ερευνητικούς και εμπορικούς σκοπούς, αλλά ανησυχία γύρω από την ιδιωτικότητα των ατόμων. Η πρόσφατη βιβλιογραφία στον τομέα της προστασίας της ιδιωτικότητας εστιάζει σε αυστηρότερες εγγυήσεις, υποθέτοντας επιτιθέμενους με αυξανόμενα ισχυρή γνώση [27, 29, 76, 22]. Εντούτοις, η απόκτηση λεπτομερούς γνώσης των στοιχείων ενός ατόμου είναι δύσκολη και λιγότερο πιθανή στην πράξη. Στην πράξη, είναι συχνότερο για έναν επιτιθέμενο να έχει πρόσβαση σε πιο αφηρημένη πληροφορία όπως είναι το άθροισμα, η

Όνομα	*Μισθός*	*Εισόδημα Κεφ.*	*Απώλειες Κεφ.*	*Συν. Εισόδημα*	*Κέρδη Κεφ.*
Λένα	10	20	10	30	10
Γεωργία	15	15	15	30	0
Ανθή	30	40	20	70	20
Στέλλα	50	30	10	80	20

Πίνακας 7.1: Παράδειγμα Φορολογικών δεδομένων

διαφορά ή κάποια άλλη συνάρτηση που ορίζεται πάνω στα γνωρίσματα των εγγραφών, αντί για τις ακριβείς τιμές τους.

Παράδειγμα 7.19. *Ας θεωρήσουμε το παράδειγμα του Πίνακα 7.1, όπου παρουσιάζονται οι οικονομικές εγγραφές τεσσάρων φορολογούμενων. Οι δύο τελευταίες στήλες δεν χρειάζεται να δημοσιευθούν, προκύπτουν άμεσα από τις τιμές των υπολοίπων πεδίων* Μισθός, Εισοδήματα κεφαλαίων *και* Απώλειες κεφαλαίων. *Ένα ρεαλιστικό σενάριο είναι όταν κάποιος κακόβουλος επιτιθέμενος μπορεί να γνωρίζει κατά προσέγγιση το άθροισμα ή τις διαφορές κάποιων πηγών εισοδημάτων του στόχου, αλλά όχι την ακριβή τιμή κάθε επιμέρους εισοδήματος. Έτσι, δεν πρόκειται για τις ακριβείς τιμές των γνωρισμάτων που δρουν ως ψευδο-αναγνωριστικά, αλλά για τις τιμές συναρτήσεων που ορίζονται πάνω σε αυτά.*

Ας υποθέσουμε ότι ένας επιτιθέμενος γνωρίζει ότι το συνολικό εισόδημα της Λένας είναι 30 χιλιάδες και ότι το καθαρό κέρδος που είχε από κεφάλαια είναι μηδενικό, δηλαδή τα εισοδήματα από κεφάλαια είναι ίσα με τις απώλειες. Αν δημοσιευθεί ο Πίνακας 7.1, ακόμη και μετά την αφαίρεση μοναδικών αναγνωριστικών όπως είναι τα ονοματεπώνυμα, ο επιτιθέμενος μπορεί να υπολογίσει το συνολικό εισόδημα καθώς και την διαφορά μεταξύ των δυο οικονομικών μεγεθών που αφορούν κεφάλαια, για κάθε εγγραφή και να ανακαλύψει ότι η πρώτη ανήκει στην Λένα, παρόλο που δεν γνωρίζει τις ακριβείς τιμές για κάθε εισόδημα. Ο 2-ανώνυμος Πίνακας 7.2 αποτελεί την ανωνυμοποίηση του Πίνακα 7.1. Οι δύο τελευταίες στήλες δεν χρειάζεται να δημοσιευθούν, υπολογίζονται άμεσα από τις τιμές των πεδίων κάθε εγγραφής. Οποιοσδήποτε επιτιθέμενος γνωρίζει τις τιμές των δυο συναρτήσεων f_1 =(μισθός + εισόδημα κεφαλαίων) και f_2 =(εισόδημα κεφαλαίων - απώλειες κεφαλαίων) για την εγγραφή ενός στόχου, δεν θα μπορεί να τις ταιριάξει με λιγότερες από 2 εγγραφές. Αντίστοιχο επίπεδο ιδιωτικότητας μπορεί να επιτευχθεί από τον Πίνακα 7.3 όπου οι εγγραφές {1,2}, και οι {3,4}, έχουν πανομοιότυπες τιμές στις συναρτήσεις που γνωρίζει ο επιτιθέμενος. Παρατηρούμε ότι δεν χρειάζεται η γενίκευση όλων των τιμών και έτσι επιτυγχάνεται μικρότερη απώλεια πληροφορίας. Και στις δυο περιπτώσεις ο επιτιθέμενος δεν μπορεί να διαχωρίσει την εγγραφή της Λένας από την εγγραφή της Γεωργίας. Εντούτοις, ο Πίνακας 7.3 περιέχει πολύ πιο λεπτομερή πληροφορία χωρίς να παραβιάζεται η ιδιωτικότητα των χρηστών.

Στόχος αυτού του κεφαλαίου είναι η διατύπωση μιας εγγύησης αντίστοιχης με την k-ανωνυμία για την πρόληψη επιθέσεων αποκάλυψης ταυτότητας. Προτείνεται μια προσέγγιση με γενικεύσεις τοπικής ανακωδικοποίησης που διατηρεί περισσότερη χρήσιμη πληροφορία στα τελικά δεδομένα γενικεύοντας τις τιμές τους μόνο όσο χρειάζεται ώστε η γνώση του επιτιθέμενου να μην αντιστοιχεί σε μοναδικές ή σπάνιες εγγραφές.

Η βασική ιδέα της μεθόδου είναι η δημιουργία ομάδων από εγγραφές που θα έχουν τις ίδιες τιμές ως προς τις συναρτήσεις που θεωρούνται γνώση του επιτιθέμενου, χωρία απαραίτητα να είναι πανομοιότυπες μεταξύ τους. Για να επιτευχθεί αυτό γίνεται συσταδοποίηση των εγγραφών με βάση μια μετρική ομοιότητας που λαμβάνει υπόψη τις τιμές των συναρτήσεων που υπολογίζονται από τις τιμές των γνωρισμάτων των εγγραφών. Έτσι δημιουργούνται ομάδες εγγραφών μεγέθους τουλάχιστον k και εφαρμόζονται γενικεύσεις τοπικής ανακωδικοποίησης μέσα σε κάθε ομάδα ανεξάρτητα από τις υπόλοιπες.

Η συνεισφορά του Κεφαλαίου 7 συνοψίζεται στα ακόλουθα σημεία:

- Ορίζουμε το πρόβλημα της ανωνυμοποίησης δεδομένων ως προς την σύνθετη γνώση πολλών συναρτήσεων,
- Διατυπώνουμε την k_f^m-*ανωνυμία* για την εγγύηση της προστασίας ιδιωτικότητας απέναντι σε τέτοιες επιθέσεις,
- Προτείνουμε πέντε αλγόριθμους ανωνυμοποίησης με στόχο την διατήρηση καλύτερης ποιότητας των τελικών ανώνυμων δεδομένων,
- Αξιολογούμε την μέθοδο με πραγματικά δεδομένα και συγκρίνουμε τα πειραματικά αποτελέσματα με εκείνα της k-ανωνυμίας.

7.2 Ορισμός του Προβλήματος

7.2.1 Μοντέλο Δεδομένων

Έστω D ένας σχεσιακός πίνακας αποτελούμενος από n συνεχή γνωρίσματα $\{A_0, A_1, ..., A_{n-1}\}$. Κάθε γραμμή του πίνακα αντιστοιχεί σε μια εγγραφή $t \in D$ ενός ατόμου και μπορεί να αναπαρασταθεί ως ένα διάνυσμα $\mathbf{X} = [x_0\ x_1\ x_2\ ...\ x_{n-1}]$ με n διαστάσεις που λαμβάνουν τιμές από ένα συνεχές πεδίο τιμών $\mathcal{I} \subseteq \mathbb{R}$.

7.2.2 Μοντέλο Επίθεσης

Θεωρούμε επιτιθέμενους οι οποίοι δεν γνωρίζουν τις ακριβείς τιμές των γνωρισμάτων που ανήκουν στην εγγραφή του στόχου. Μπορούν αντιθέτως να γνωρίζουν τις τιμές από ένα σύνολο m συνεχών συναρτήσεων $F = \{f_0(\mathbf{X}),\ f_1(\mathbf{X}),\ ...\ f_{m-1}(\mathbf{X})\}$. Τα ορίσματα των συναρτήσεων αντιστοιχούν στα γνωρίσματα του πίνακα των δεδομένων και έχουν πεδίο ορισμού το $\mathcal{I}$.

Id	*Μισθός*	*Εισόδημα Κεφ.*	*Απώλειες Κεφ.*	*Συν. Εισόδημα*	*Κέρδη Κεφ.*
1	[10-15]	[15-20]	[10-15]	[25-35]	[0-10]
2	[10-15]	[15-20]	[10-15]	[25-35]	[0-10]
3	[30-50]	[30-40]	[10-20]	[60-90]	[10-30]
4	[30-50]	[30-40]	[10-20]	[60-90]	[10-30]

Πίνακας 7.2: 2-Ανώνυμος πίνακας φορολογικών δεδομένων

Οι συναρτήσεις μπορεί να ορίζονται πάνω σε όλα η σε υποσύνολο των γνωρισμάτων. Τα σύνολα γνωρισμάτων που αποτελούν τα ορίσματα διαφορετικών συναρτήσεων μπορεί να επικαλύπτονται. Παραδείγματος χάριν, ο επιτιθέμενος μπορεί να γνωρίζει τις τιμές των συναρτήσεων $f_0(\mathbf{X}) = \sum_{i=1}^{n} x_i : \mathcal{I}^n \rightarrow \mathbb{R}$, $f_1(\mathbf{X}) = (x_1 + x_2)/x_3 : \mathcal{I}^3 \rightarrow \mathbb{R}$, $f_2(\mathbf{X}) = {x_n}^3 - x_1 : \mathcal{I}^2 \rightarrow \mathbb{R}$, κτλ.

Η γνώση του επιτιθέμενου μοντελοποιείται ως ένα σύνολο εξισώσεων $\{f_0(\mathbf{X}) = c_0$, $f_1(\mathbf{X}) = c_1$, ... $f_{m-1}(\mathbf{X}) = c_{m-1}\}$, όπου τα $c_0, c_1, ...c_{m-1}$ είναι σταθερές. Ο επιτιθέμενος μπορεί να υπολογίσει τις τιμές των συναρτήσεων για κάθε εγγραφή του πίνακα D και να απορρίψει όσες δεν ταιριάζουν στην γνώση του, ώστε να ανακαλύψει ποιά ανήκει στον στόχο.

7.2.3 Εγγύηση Ιδιωτικότητας

Παρέχουμε μια νέα εγγύηση ιδιωτικότητας επεκτείνοντας την k-ανωνυμία [63, 66] ώστε να αντιμετωπιστούν επιθέσεις που βασίζονται στην γνώση συναρτήσεων πάνω στις τιμές των γνωρισμάτων των εγγραφών.

Ορισμός 7.13. *(k_f^m-ανωνυμία) Οποιοσδήποτε κακόβουλος επιτιθέμενος γνωρίζει τις τιμές από ένα σύνολο m συναρτήσεων $f_0(\mathbf{X}), f_1(\mathbf{X}), ...f_{m-1}(\mathbf{X})$ που υπολογίζονται πάνω στις τιμές των γνωρισμάτων της εγγραφή ενός στόχου, δεν θα μπορεί να ταιριάξει αυτή την γνώση σε λιγότερες από k εγγραφές στα δημοσιευμένα δεδομένα.*

Ένα σύνολο δεδομένων D που δεν ικανοποιεί την k_f^m-ανωνυμία μπορεί να μετασχηματιστεί σε ένα k_f^m-ανώνυμο σύνολο δεδομένων D^* γενικεύοντας τις τιμές των γνωρισμάτων του έτσι ώστε κάθε εγγραφή να μην μπορεί να διαχωριστεί από τουλάχιστον k-1 άλλες, ως προς τις τιμές των συναρτήσεων που υπολογίζονται πάνω στα γνωρίσματά της.

Η κλασσική k-ανωνυμία θα αντιμετώπιζε αυτές τις επιθέσεις δημιουργώντας κλάσεις ισοδυναμίας μεγέθους τουλάχιστον k εγγραφών. Οι εγγραφές κάθε κλάσης θα είναι πανομοιότυπες μεταξύ τους ως προς όλα τους τα γνωρίσματα. Δοσμένης μια κλάσης ισοδυναμίας EC, ας υποθέσουμε ότι η ελάχιστη και η μέγιστη τιμή ενός γνωρίσματος A_i που εμφανίζονται μέσα στις εγγραφές EC, είναι v_{min} και v_{max} αντιστοίχως. Ένας αλγόριθμος k-ανωνυμοποίησης θα αντικαθιστούσε τις τιμές του A_i σε όλες τις εγγραφές EC με το ίδιο εύρος γενίκευσης, το οποίο δεν θα μπορούσε να είναι μικρότερο από το $[v_{min}, v_{max}]$.

Η προτεινόμενη μέθοδος μπορεί να k_f^m-ανωνυμοποιήσει το D εφαρμόζοντας τις ελάχιστες γενικεύσεις που απαιτούνται ώστε οι εγγραφές μέσα σε μια κλάση ισοδυναμίας να έχουν πανομοιότυπες τις τιμές των συναρτήσεων, αντί για τα γνωρίσματα. Με το τρόπο αυτό περιορίζεται

Id	*Μισθός*	*Εισόδημα Κεφ.*	*Απώλειες Κεφ.*	*Συν. Εισόδημα*	*Κέρδη Κεφ.*
1	10	20	[10-20]	30	[0-10]
2	15	15	[5-15]	30	[0-10]
3	[30-40]	40	20	[70-80]	20
4	[40-50]	30	10	[70-80]	20

Πίνακας 7.3: 2_f^2-Ανώνυμος πίνακας φορολογικών δεδομένωνε

σημαντικά η απώλεια πληροφορίας των ανωνυμοποιημένων δεδομένων, όπως εξηγείται στην Ενότητα 7.3 και δείχνεται πειραματικά στην Ενότητα 7.4.

Σημειώνεται ότι η περίπτωση ο επιτιθέμενος να γνωρίζει την ακριβή τιμή ενός γνωρίσματος μπορεί να ενσωματωθεί στην προτεινόμενη μέθοδο και δεν χρειάζεται να εξεταστεί χωριστά. Αυτού του τύπου η γνώση μοντελοποιείται με την συνάρτηση ταυτότητας ενός γνωρίσματος, δηλαδή την συνάρτηση $f(\mathbf{X}) = x_i$.

Στην ακραία περίπτωση που ο επιτιθέμενος γνωρίζει τις τιμές των συναρτήσεων ταυτότητας για κάθε γνώρισμα, δηλαδή το σύνολο των τιμών των γνωρισμάτων $\{f_i(\mathbf{X}) = x_i, \forall i = 0, ..., n-1\}$, η μέθοδός μας παρέχει την ίδια εγγύηση ιδιωτικότητας όπως η κλασσική k-ανωνυμία. Αυτό είναι το χειρότερο σενάριο όπου η γνώση του επιτιθέμενου είναι τόσο ισχυρή που δεν μας επιτρέπει να μειώσουμε την απώλεια πληροφορίας κάτω από τα επίπεδα που εισάγει η k-ανωνυμία. Εντούτοις, υποθέτοντας επιτιθέμενους που ήδη γνωρίζουν ολόκληρη την εγγραφή του στόχου έχει μικρή πρακτική σημασία, καθώς δεν απομένει τίποτα περισσότερο για να ανακαλύψουν.

7.3 Αλγόριθμος Ανωνυμοποίησης

Η γνώση που έχει ο επιτιθέμενος για την εγγραφή του στόχου αποτελείται από το σύστημα εξισώσεων S=$\{f_0(X) = c_0, f_1(X) = c_1, ..., f_{m-1}(X) = c_{m-1}\}$, όπου c_i ($\forall i$=0,...$m-1$) είναι οι τιμές των συναρτήσεων που γνωρίζει. Αν το σύστημα είναι $m \times m$, δηλαδή αν έχει μοναδική λύση, τότε μια τέτοια επίθεση θα μπορεί να υπολογίσει τις ακριβείς τιμές των γνωρισμάτων του στόχου. Σε αυτή την περίπτωση τα δεδομένα θα πρέπει να ανωνυμοποιηθούν με βάση την κλασσική εγγύηση της k-ανωνυμίας.

Επιπλέον, υποθέτουμε ότι το σύστημα των εξισώσεων εκφράζεται σε απλοποιημένη μορφή, έτσι ώστε οι εξισώσεις του δεν μπορούν να χρησιμοποιηθούν για να εξάγουμε απλούστερες εξισώσεις. Παραδείγματος χάριν, αν ο επιτιθέμενος γνωρίζει $\{x_1 + x_2 + x_3 = 10$ και $x_2 = 1\}$. Το σύστημα αυτό είναι ισοδύναμο με το $\{x_1 + x_3 = 9$ και $x_2 = 1\}$. Η τελευταία μορφή είναι αυτή που χρησιμοποιείται ως είσοδος στον αλγόριθμο.

Προτείνουμε ένα ευρηστικό αλγόριθμο για την ανωνυμοποίηση συνόλων δεδομένων παρουσία της γνώσης πολλαπλών συναρτήσεων. Ο επιτιθέμενος μπορεί να γνωρίζει έως m δοσμένες συναρτήσεις, αλλά όχι τις ακριβείς τιμές όλων των πεδίων της εγγραφής του στόχου. Ο αλγόριθμος δέχεται ως είσοδο το σύνολο δεδομένων D και ένα σύνολο συναρτήσεων F που μοντελοποιούν την πιθανή γνώση του επιτιθέμενου. Τα ορίσματα μιας συνάρτησης $f_i \in F$ είναι ένα υποσύνολο των γνωρισμάτων του D. Η έξοδος του αλγόριθμου είναι το k_f^m-ανώνυμο σύνολο D^*.

Η προτεινόμενη μέθοδος αποτελείται από δυο κύριες φάσεις: Την φάση *συσταδοποίησης* και την φάση *γενίκευσης*. Η πρώτη μοιράζει τις εγγραφές του συνόλου D σε ομάδες μεγέθους τουλάχιστον k και η δεύτερη εφαρμόζει τοπικές γενικεύσεις ανεξάρτητα μέσα σε κάθε ομάδα, ώστε να ικανοποιείται η εγγύηση της k_f^m-ανωνυμίας. Ο ψευδοκώδικας της μεθόδου φαίνεται στον Αλγόριθμο 12.

Αρχικά, στις γραμμές 2-5 ο αλγόριθμος ελέγχει αν υπάρχει υποσύστημα εξισώσεων $d \times d$

Αλγόριθμος 12 k_f^m-Anonymize

Require: D: Dataset, F: Set of functions
Ensure: $D^\star$ {Anonymized dataset}
1: $D^* = \emptyset$ {Initialization}
2: **if** there exist $d \times d$ subsystem of equations ($d < m$) that can have unique solution **then**
3: k-anonymize D w.r.t. the d attributes of the system.
4: Remove the d functions from F.
5: $QI = X \setminus X_d$, where X_d is the set of the d attributes.
6: $SetPriorities(F)$.
7: $G = Group(D, F)$.
8: **for all** groups of records $g \in G$ **do**
9: **for** every pair of functions $f_i \neq f_j$ such that $priority[f_i] = priority[f_j]$ **do**
10: $InClassPriority(EC, f_i, f_j)$.
11: re-order functions in set F from higher to lower priority.
12: **for** every function $f_i \in F$ **do**
13: Let f_{min}, f_{max} be the minimum and maximum values of f_i that appear in g.
14: $g = Generalize(g, f_i, true, f_{max})$.
15: $g = Generalize(g, f_i, false, f_{min})$.
16: $D^* = D^* \cup \{g\}$.
17: Publish D^*.

(όπου $d < m$) το οποίο έχει μοναδική λύση και k-ανωνυμοποιεί τα αντίστοιχα d γνωρίσματα των δεδομένων πριν προχωρήσει. Ο αλγόριθμος αναθέτει προτεραιότητες σε κάθε συνάρτηση του F καλώντας την $SetPriorities(F)$ στην γραμμή 6. Οι προτεραιότητες αυτές υπαγορεύουν την σειρά με την οποία θα εξετάσουμε τις συναρτήσεις κατά την φάση γενίκευσης, όπως περιγράφεται στις Ενότητες 7.3.1 και 7.3.3.

Εν συνεχεία, ο αλγόριθμος προχωρά στην φάση *συσταδοποίησης* όπου καλείται η συνάρτηση $Group(D, F)$ για την δημιουργία ομάδων από τουλάχιστον k εγγραφές. Τέλος, στην φάση *γενίκευσης* (γραμμές 8 έως 16) αυτές οι ομάδες θα μετασχηματιστούν σε *κλάσεις ισοδυναμίας* ως προς τις τιμές των συναρτήσεων του F.

7.3.1 Προτεραιότητες Συναρτήσεων

Κατά την αναζήτηση κατάλληλων γενικεύσεων ώστε να έχουν όλες οι εγγραφές μιας ομάδας την ίδια τιμή στις συναρτήσεις, εξετάζουμε κάθε συνάρτηση χωριστά. Η σειρά με την οποία εξετάζουμε τις συναρτήσεις επηρεάζει τα εύρη των γενικεύσεων των γνωρισμάτων. Για αυτό τον λόγο πρώτα ταξινομούμε τις συναρτήσεις του συνόλου F κατά τέτοιο τρόπο ώστε οι γενικεύσεις που εφαρμόζονται σε ένα γνώρισμα λόγω μιας συνάρτησης να επηρεάζουν όσο το δυνατόν λιγότερο τις επόμενες συναρτήσεις. Η συνάρτηση $SetPriorities(F)$ αναθέτει προτεραιότητες στις συναρτήσεις σύμφωνα με τρία βασικά κριτήρια:

(α′) το πλήθος των «ελεύθερων» γνωρισμάτων μιας συνάρτησης, τα οποία δεν εμφανίζονται

Αλγόριθμος 13 SetPriorities

Require: $F = \{f_0, f_1, ..., f_{m-1}\}$: a set of functions

Ensure: set F is ordered by priority of functions

1: Let $fx_i = |args(f_i)|$ {number of attributes in $f_i()$.}
2: Let $fr_i = |args(f_i) - \{\cup_{\forall j \neq i} args(f_j)\}|$ {number of free attributes in $f_i()$.}
3: Let $ff_i = |\{f_j : args(f_j) \cap args(f_i) \neq \emptyset,\ \forall f_j \neq f_i\}|$ {number of other functions that share attributes with $f_i()$.}
4: Order the functions by decreasing number of their fr_i.
5: Assign higher priorities to those with smaller fr_i.
6: **if** there is a tie ($fr_i == fr_j$) **then**
7: Order them by decreasing number of their fx_i.
8: Assign higher priorities to those with smaller fx_i.
9: **if** there is a tie ($xs_i == xs_j$) **then**
10: Order them by increasing number of ff_i.
11: Assign higher priorities to those with greater ff_i.
12: **if** there is a tie ($ff_i == ff_j$) **then**
13: Assign them equal priorities.
14: return F.

ως ορίσματα άλλων συναρτήσεων $fr_i = |args(f_i) - \{\cup_{\forall j \neq i} args(f_j)\}|$,

(β′) το πλήθος των ορισμάτων μιας συνάρτησης $fx_i = |args(f_i)|$,

(γ′) το πλήθος των άλλων συναρτήσεων οι οποίες μοιράζονται κοινά ορίσματα με μια συνάρτηση $ff_i = |F_{share_i}|$, ωηερε $F_{share_i} = \{f_j : args(f_j) \cap args(f_i) \neq \emptyset,\ \forall f_j \neq f_i\}$.

Διαισθητικά, όσο περισσότερα «*ελεύθερα*» γνωρίσματα έχει μια συνάρτηση ως ορίσματα, τόσο ευκολότερο είναι να βρεθούν γενικεύσεις γνωρισμάτων που δεν επηρεάζουν τις τιμές των άλλων συναρτήσεων. Για αυτό τον λόγο το κριτήριο (α) θεωρείται πιο σημαντικό για την ανάθεση προτεραιοτήτων. Οι συναρτήσεις με μεγαλύτερο fr_i θα εξεταστούν τελευταίες, διότι υπάρχουν περισσότερες πιθανότητες να υπάρχουν διαθέσιμα ορίσματα που μπορούν να γενικευθούν για να ικανοποιηθεί η εγγύηση.

Όταν υπάρχει ισοπαλία μεταξύ δυο ή περισσοτέρων συναρτήσεων ως προς το κριτήριο (α), δίνεται μεγαλύτερη προτεραιότητα σε εκείνες με τα λιγότερα συνολικά γνωρίσματα, σύμφωνα με το κριτήριο (β). Αν μία συνάρτηση έχει μόνο ένα γνώρισμα στη λίστα ορισμάτων της, τότε θα πρέπει να εξεταστεί πρώτη. Αν αντίθετα έχει περισσότερα γνωρίσματα, υπάρχουν περισσότερες πιθανές γενικεύσεις αυτών των γνωρισμάτων ώστε να αποκτήσουν όλες οι εγγραφές της ομάδας την ίδια τιμή.

Τέλος, αν δυο ή περισσότερες συναρτήσεις έχουν ισοπαλία ως προς τα δυο πρώτα κριτήρια, λαμβάνεται υπόψη το κριτήριο (γ): θεωρείται προτιμότερη η εξέταση συναρτήσεων που δεν θα επηρεάσουν σημαντικά τις τιμές των επόμενων, δηλαδή συναρτήσεων που μοιράζονται ορίσματα με λιγότερες άλλες συναρτήσεις. Έτσι σε αυτές δίνεται μεγαλύτερη προτεραιότητα. Ο ψευδοκώδικας για την παραπάνω διαδικασία φαίνεται στον Αλγόριθμο 13.

Αλγόριθμος 14 InClassPriority

Require: g group of records, $\{f_i(X), f_j(X)\}$: functions with the same priority
Ensure: set priority of functions for this group

1: $\Delta f_i = 0$.
2: $\Delta f_j = 0$.
3: **for** every pair of tuples $t_p, t_q \in g$ **do**
4: $\quad \Delta f_i = \Delta f_i + |f_i(X)_{t=t_p} - f_i(X)_{t=t_q}|$.
5: $\quad \Delta f_j = \Delta f_j + |f_j(X)_{t=t_p} - f_i(X)_{t=t_q}|$.
6: **if** $\Delta f_i \leq \Delta f_j$ **then**
7: $\quad$ set $priority[f_i] > priority[f_j]$.
8: **else**
9: $\quad$ set $priority[f_i] < priority[f_j]$.
10: return.

Αν υπάρξει ισοπαλία και στα τρία κριτήρια, τότε το ζήτημα των προτεραιοτήτων λύνεται μέσα σε κάθε κλάση ισοδυναμίας ανεξάρτητα. Η συνάρτηση $InClassPriority(EC, f_i, f_j)$ που καλείται στην γραμμή 10 του βασικού αλγορίθμου της μεθόδου, αποφασίζει αν η συνάρτηση f_i θα εξεταστεί πριν από την f_j κατά την φάση γενίκευσης της ομάδας EC (κλάση ισοδυναμίας). Όπως φαίνεται από τον ψευδοκώδικα του Αλγορίθμου 14. υπολογίζονται οι τιμές αυτών των συναρτήσεων για κάθε τιμή της EC και δίνεται μεγαλύτερη προτεραιότητα στην συνάρτηση που οι τιμές τις είναι πιο κοντά μεταξύ τους. Η απόσταση υπολογίζεται στις γραμμές 4-5 με βάση το άθροισμα των απολύτων τιμών των μεταξύ τους διαφορών.

7.3.2 Συσταδοποίηση Εγγραφών

Μας ενδιαφέρει η δημιουργία κλάσεων ισοδυναμίας μεγέθους τουλάχιστον k εγγραφών, οι οποίες στο τέλος θα πρέπει να έχουν τις ίδιες τιμές στις συναρτήσεις που αναπαριστούν την γνώση του επιτιθέμενου. Στόχος μας είναι να περιοριστεί η απώλεια πληροφορίας λόγω γενικεύσεων όσο το δυνατόν περισσότερο. Επομένως, η συσταδοποίηση των εγγραφών πρέπει να γίνει κατά τέτοιο τρόπο ώστε οι εγγραφές μέσα σε μια συστάδα να έχουν όσο το δυνατόν πιο όμοιες τιμές στις συναρτήσεις. Για το σκοπό αυτό, χρησιμοποιούμε έναν αλγόριθμο συσταδοποίησης, ο οποίο ς υπολογίζει τις τιμές των συναρτήσεων για κάθε εγγραφή και τοποθετεί στην ίδια ομάδα εγγραφές με παρόμοιες τιμές συναρτήσεων. Η μετρική ομοιότητας που χρησιμοποιούμε είναι η απόσταση Manhattan, αλλά θα μπορούσαν να ενσωματωθούν άλλες μετρικές όπως η ευκλείδεια απόσταση.

Η συνάρτηση $Group(D, F)$ καλείται από τον κύριο αλγόριθμο στην γραμμή 7, για την δημιουργία του συνόλου G από ομάδες τουλάχιστον k εγγραφών. Ο Αλγόριθμος 15 περιγράφει αυτή την διαδικασία. Ο αλγόριθμος διαβάζει τις εγγραφές μία φορά και κάθε χρονική στιγμή διατηρεί το πολύ cl συστάδες (ομάδες εγγραφών) στην μνήμη, στο σύνολο $currGroups$. Αρχικά, το σύνολο αυτό είναι κενό. Η πρώτη εγγραφή διαβάζεται και τοποθετείται σε μια νέα συστάδα μεγέθους 1. Κάθε νέα εγγραφή $t \in D$ διαβάζεται και δημιουργεί μια νέα συστάδα

$g = \{t\}$. Υπολογίζεται η *απόσταση* της νέας συστάδας με καθεμία από τις συστάδες που υπάρχουν στη μνήμη στο σύνολο $currGroups$. Η συνάρτηση υπολογισμού της απόστασης δίνεται από τον Αλγόριθμο 16. Έστω $bestG$ είναι η συστάδα με την μικρότερη απόσταση από την g. Αν η απόσταση μεταξύ της g και της $bestG$ είναι κάτω από ένα κατώφλι th, ή αν ο αριθμός των διαθέσιμων συστάδων στην μνήμη ξεπερνά το μέγιστο επιτρεπτό όριο cl, τότε συγχωνεύονται οι συστάδες g και $bestG$ σε μια νέα που περιέχει το σύνολο των εγγραφών τους. Διαφορετικά, η g παραμένει μια συστάδα μεγέθους 1 και προστίθεται στο σύνολο των διαθέσιμων συστάδων στη μνήμη $currGroups$. Κάθε φορά που μια συστάδα γεμίζει με k εγγραφές, την προσθέτουμε στο αποτέλεσμα G και την διαγράφουμε από το σύνολο των διαθέσιμων συστάδων.

Στο τέλος αυτής της διαδικασίας μπορεί να έχουν απομείνει ορισμένες μισογεμάτες συστάδες, δηλαδή ομάδες μικρότερες των k εγγραφών. Αυτές συγχωνεύονται κατάλληλα ώστε να έχουν μεγέθη τουλάχιστον k. Η επιλογή των ομάδων που θα συγχωνευτούν μεταξύ τους γίνεται εξετάζοντας τις αποστάσεις τους, όπως υπολογίζονται από τον Αλγόριθμο 16.

Η συνάρτηση $Distance(g_1, g_2)$ δέχεται ως είσοδο δυο ομάδες εγγραφών $g_1 \neq \emptyset$, $g_2 \neq \emptyset$ και υπολογίζει την μεταξύ τους απόσταση ως προς το σύνολο των τιμών των συναρτήσεων $\{f_0, f_1, ..., f_{m-1}\}$ κάθε ζεύγους εγγραφών t_i, t_j, τέτοιες ώστε $t_i \in g_1$ και $t_j \in g_2$. Επιστρέφει την μέση απόσταση manhattan για κάθε ζεύγος εγγραφών.

Ο αλγόριθμος συσταδοποίησης είναι παραλλαγή του [60], η οποία προσαρμόστηκε έτσι ώστε η συσταδοποίηση να γίνεται με βάση τις τιμές των συναρτήσεων από το σύνολο F, αντί για τις τιμές των γνωρισμάτων των εγγραφών. Οποιοσδήποτε άλλος αλγόριθμος συσταδοποίησης θα μπορούσε να χρησιμοποιηθεί, αρκεί το ελάχιστο μέγεθος κάθε συστάδας να είναι εγγυημένο ότι δεν θα είναι μικρότερο από k και η μετρική ομοιότητας να προσαρμοστεί για τις τιμές των $\{f_0, f_1, ..., f_{m-1}\}$.

7.3.3 Γενίκευση Γνωρισμάτων

Έχοντας ορίσει τις προτεραιότητες των συναρτήσεων και έχοντας δημιουργήσει τις ομάδες των εγγραφών που θα γίνουν κλάσεις ισοδυναμίας, ο αλγόριθμος περνά στην φάση της γενίκευσης των γνωρισμάτων, όσο χρειάζεται ώστε οι εγγραφές να μην αναγνωρίζονται από τις τιμές των συναρτήσεων που γνωρίζει ο επιτιθέμενος. Στις γραμμές 14 και 15 του Αλγορίθμου 12, καλείται η συνάρτηση $Generalize(g, f_i, upper, limit)$ για κάθε ομάδα εγγραφών και κάθε συνάρτηση. Οι αλγόριθμοι που προτείνουμε για την φάση γενίκευσης χωρίζονται σε τρεις κατηγορίες: (α) αλγόριθμοι που χρησιμοποιούν αριθμητική ανάλυση. (β) αλγόριθμοι που χρησιμοποιούν τις αντίστροφες συναρτήσεις για να φτιάξουν κάθε συνάρτηση χωριστά, και (γ) συνδυασμός όλων των αντίστροφων συναρτήσεων και όλων των γνωρισμάτων.

Η βασική ιδέα είναι ότι είτε εκτιμάται με χρήση αριθμητικής ανάλυσης, είτε υπολογίζεται με βάση τις αντίστροφες συναρτήσεις, το πόσο πρέπει να γενικευτεί η τιμή ενός γνωρίσματος v_{rj} μιας εγγραφής $t_r \in g$ έτσι ώστε μια δοσμένη συνάρτηση f_i να πάρει ένα συγκεκριμένο εύρος τιμών για την εγγραφή. Το επιθυμητό εύρος τιμών ορίζεται από την μικρότερη τιμή (f_{min}) και την μεγαλύτερη τιμή (f_{max}) που εμφανίζει η f_i όταν υπολογίζεται πάνω στις αρχικές τιμές

Αλγόριθμος 15 Group

Require: D: dataset, F: a set of functions, th: threshold, cl: limit of clusters currently in memory.

Ensure: G: set of ECs, i.e. non-overlapping groups of records from D which have similar values of the functions in F.

1: $currGroups = \emptyset, G = \emptyset$.
2: **for all** tuples $t_j \in D$ **do**
3: **for all** functions $f_i() \in F$ **do**
4: calculate $f_{ij} = f_i(X)_{t=t_j}$.
5: Form new group $g_j = \{t_j\}$.
6: $minCost = \infty$.
7: **for all** groups $g \in currGroups$ **do**
8: $Cost = Distance(g_j, g)$.
9: **if** $Cost < minCost$ **then**
10: $minCost = Cost$.
11: $bestG = g$.
12: **if** $(minCost < th)$ OR $(|currGroups| \geq cl)$ **then**
13: $bestG = bestG \cup \{t\}$.
14: **if** $|bestG| == k$ **then**
15: $currGroups = currGroups \setminus \{bestG\}$.
16: $G = G \cup \{bestG\}$.
17: **else**
18: $currGroups = currGroups \cup \{g_j\}$.
19: **if** $|currGroups| > 0$ **then**
20: {Merge remaining clusters:}
21: **for all** groups $g_i \in currGroups$ **do**
22: $minCost = \infty$.
23: **for all** groups $g_j \in currGroups$ **do**
24: $Cost = Distance(g_i, g_j)$.
25: **if** $Cost < minCost$ **then**
26: $minCost = Cost$.
27: $bestG = g_j$.
28: $g_i = g_i \cup bestG$.
29: **if** $|g_i| \geq k$ **then**
30: $currGroups = currGroups \setminus \{g_i\}$.
31: $G = G \cup \{g_i\}$.
32: return G.

των εγγραφών της ομάδας g, δηλαδή της κλάσης ισοδυναμίας στην οποία ανήκει η t_r.

Σε ορισμένες περιπτώσεις η v_{rj} γενικεύεται μόλις σε ένα μέρος του απαιτούμενου εύρους που εκτιμήθηκε ή υπολογίστηκε. Έτσι, πλησιάζει τα ζητούμενα όρια της f_i, αλλά δεν τα

Αλγόριθμος 16 Distance

Require: g_p, g_q: groups of records

Ensure: d: average Manhattan distance between each pair of records from the two groups.

1: $MD_{pq} = 0$.
2: **for all** pairs of tuples $t_i \in g_p$, $t_j \in p_q$ **do**
3: **for all** functions $f_r \in F$ **do**
4: $md_{ijr} = |f_r(X)_{t=t_i} - f_r(X)_{t=t_j}|$
5: $MD_{pq} = MD_{pq} + md_{ijr}$
6: $d = MD_{pq}/(|F| \cdot |g_p| \cdot |p_q|)$ {average}
7: $d = d/|\mathcal{I}|$ {normalization to 1}
8: return d.

έχει φτάσει ακόμα. Εν συνεχεία, γίνεται ο υπολογισμός για την γενίκευση του επόμενου γνωρίσματος, και ούτω καθεξής. Η διαδικασία σταματά όταν έχουν προσεγγιστεί τα όρια f_{min} και f_{max}, δηλαδή όταν υπάρχει συνδυασμός τιμών$\{v_{r0}, v_{r1}, ...v_{rn-1}\}$ μέσα από τα αντίστοιχα εύρη γενίκευσης των γνωρισμάτων $x_0, x_1, ...x_{n-1}$ της εγγραφής t_r, έτσι ώστε η τιμή της $f_i(v_{r0}, v_{r1}, ...v_{rn-1})$ νε είναι ίση με f_{min} (αντίστοιχα για την f_{max}).

Για τον υπολογισμό του εύρους τιμών μιας πραγματικής συνάρτησης $[f_{min}, f_{max}]$ βασιστήκαμε στα εύρη τιμών των ορισμάτων της και χρησιμοποιήσαμε ανάλυση διαστημάτων [55]. Παραδείγματος χάριν, αν $x_1 = [min_1, max_1]$, $x_2 = [min_2, max_2]$ και $f_1(X) = x_1 + x_2$, τότε το εύρος τιμών της f_1 είναι $[min_1 + min_2, max_1 + max_2]$. Ενώ για την $f_2(X) = x_1 - x_2$, το αντίστοιχο εύρος είναι $[min_2 - max_1, max_2 - min_1]$, Για τις δυνάμεις με περιττό εκθέτη, το εύρος της $f_3(X) = x_1^{2\kappa+1}$ είναι $[min_1^{2\kappa+1}, max_1^{2\kappa+1}]$, $\kappa \in \mathbb{N}$, ενώ για άρτιο εκθέτη, το εύρος τιμών της $f_4(X) = x_2^{2\kappa}$ είναι $[0, max_2^{2\kappa}]$ αν $min_2 < 0$ και $[min_2^{2\kappa}, max_2^{2\kappa}]$ διαφορετικά.

Χρήση Αριθμητικής Ανάλυσης

Newton. Ο πρώτος προτεινόμενος αλγόριθμος χρησιμοποιεί αριθμητική ανάλυση για να μοιράσει τις γενικεύσεις σε όλα τα γνωρίσματα και να επιτύχει μικρή απώλεια πληροφορίας. Ο ψευδοκώδικάς του παρουσιάζεται στον Αλγόριθμο 17. Συγκεκριμένα χρησιμοποιεί την μέθοδο Newton, όπου δοσμένης μιας αρχικής εκτίμησης x_0, μια καλύτερη εκτίμηση για την ρίζα μιας συνάρτησης $g(x)$ είναι η $x_1 = x_0 - g(x_0)/g'(x_0)$. Στην δική μας περίπτωση, για να γίνει η συνάρτηση $f(X)$ ίση με μια ζητούμενη τιμή $limit$, θέτουμε $g(X) = f(X) - limit$. Η καλύτερη εκτίμηση για ένα γνώρισμα X_j είναι:

$$x_{j1} = x_{j0} - \frac{f(x_{j0}) - limit}{\partial f(x_{j0})/\partial x_j} \tag{7.1}$$

Όταν καλείται η συνάρτηση $Generalize\text{-}Newton()$, δίνεται ως είσοδος το άνω ή κάτω όριο της f_i, $limit$. Αυτή είναι η τιμή της f_i που επιθυμούμε να φτάσουμε με τις γενικεύσεις των γνωρισμάτων. Για κάθε εγγραφή t_r σε μια ομάδα $g_,$ ξεκινούμε από τις αρχικές τιμές των γνωρισμάτων $\{v_{r0}, v_{r1}, ...v_{rn-1}\}$. Αν κάποια γνωρίσματα είναι ήδη γενικευμένα, λ.χ. από μια άλλη συνάρτηση υψηλότερης προτεραιότητας από την f_i, τότε χρησιμοποιούμε ως αρχική

Αλγόριθμος 17 Generalize-Newton

Require: g: group of records, $f_i()$: function, *upper*: **true** if we consider the maximum value of the function in the group, **false** otherwise, *limit*: either min or max value of f_i in g.

Ensure: g^*: all records are indistinguishable w.r.t $f_i(X)$.

1: $g^* = \emptyset$
2: Let $\mathcal{I}_{min}, \mathcal{I}_{max}$ be the limits of the attribute domain $\mathcal{I}$.
3: Let x_j be the j^{th} attribute of D.
4: Let $range_{rj} = [min_{rj}, max_{rj}]$ be the value range of the j^{th} attribute at the r^{th} record. Initially $min_{rj} = max_{rj}$ = initial value of x_{rj}.
5: Let $v_{rj} = \frac{min_{rj}+max_{rj}}{2}$. {If not generalized, v_{rj} is the initial value.}
6: **for all** records $t_r \in g$ **do**
7: $\quad \Delta f_i = limit - f_i(X)|_{x_j=v_{rj}, \forall j}$
8: $\quad$ **while** $|\Delta f_i| > threshold$ **do**
9: $\quad\quad$ **for all** attributes $X_j \in args(f_i)$ **do**
10: $\quad\quad\quad y_r = f_i(X)|_{x_j=v_{jr}}$
11: $\quad\quad\quad part_r = \frac{\partial f}{\partial x_j}|_{x_j=v_{jr}}$
12: $\quad\quad\quad \Delta x = -\frac{y_r - limit}{part_r}$
13: $\quad\quad\quad v_{new} = v_{jr} + \Delta x$
14: $\quad\quad\quad$ **if** $v_{new} < min_{rj}$ **then**
15: $\quad\quad\quad\quad v_{new} = min\{v_{new}, \mathcal{I}_{min}\}$
16: $\quad\quad\quad\quad range_{rj} = [v_{new}, max_{rj}]$
17: $\quad\quad\quad$ **else if** $v_{new} > max_{rj}$ **then**
18: $\quad\quad\quad\quad v_{new} = max\{v_{new}, \mathcal{I}_{max}\}$
19: $\quad\quad\quad\quad range_{rj} = [min_{rj}, v_{new}]$
20: $\quad\quad\quad v_{rj} = v_{new}$
21: $\quad\quad\quad \Delta f_i = limit - f_i(X)|_{x_j=v_{new}}$
22: $\quad g^* = g^* \cup \{t_r\}$ {Add anonymized record t_r to g^*.}
23: Return g^*.

εκτίμηση τον μέσο του εύρους τους (γραμμή 5).

Για κάθε γνώρισμα x_j το οποίο είναι στη λίστα ορισμάτων της συνάρτησης f_i, γίνεται εκτίμηση του εύρους στο οποίο πρέπει να γενικευθεί ώστε η f_i να πάρει την τιμή $limit$. Στις γραμμές 10-11 υπολογίζεται το y_r ως η τρέχουσα τιμή της f_i και το $part_r$ ως η μερική παράγωγος $(\partial f_i/\partial x_j)$ της f_i ως προς το x_j, θεωρώντας τις τιμές των γνωρισμάτων $\{v_{r0}, v_{r1}, ...v_{rn-1}\}$. Στη συνέχεια, η νέα εκτιμώμενη τιμή v_{new} του γνωρίσματος x_j υπολογίζεται με βάση την εξίσωση 7.1.

Το εύρος γενίκευσης $range_{rj} = [min_j, max_j]$ του γνωρίσματος x_j της εγγραφής t_r ενημερώνεται στις γραμμές 14-19. Αν η νέα εκτιμώμενη τιμή v_{new} είναι μικρότερη του min_j, τότε το εύρος γενίκευσης θα επεκταθεί προς τα κάτω: $[v_{new}, max_j]$, Διαφορετικά αν η v_{new} είναι

μεγαλύτερη του max_j, τότε το εύρος θα επεκταθεί προς τα πάνω: $[min_j, v_{new}]$. Εντούτοις, τα όρια του αρχικού πεδίου τιμών των γνωρισμάτων $[\mathcal{I}_{min}, \mathcal{I}_{max}]$ δεν παραβιάζονται κα κανένα εύρος γενίκευσης δεν μπορεί να τα υπερβεί.

Τέλος, η τρέχουσα τιμή v_{rj} του γνωρίσματος x_j ενημερώνεται σε v_{new}. Η διαφορά Δf_i ανάμεσα στο επιθυμητό όριο $limit$ και στην τρέχουσα τιμή της f_i για $x_j = v_{new}$ υπολογίζεται στη γραμμή 21. Αν υπερβαίνει ένα προκαθορισμένο κατώφλι $threshold$ (γραμμή 8), τότε η διαδικασία επαναλαμβάνεται για το επόμενο γνώρισμα που ανήκει στην λίστα ορισμάτων της f_i, και ούτω καθεξής. Όταν το $|\Delta f_i|$ είναι κάτω του κατωφλίου, η γενικευμένη εγγραφή προστίθεται στην ανώνυμη κλάση ισοδυναμίας g^* και η διαδικασία της ανωνυμοποίησης προχωρά στην επόμενη εγγραφή της ομάδας g. Όταν ολοκληρωθεί η διαδικασία, όλες οι εγγραφές της g έχουν γενικευθεί και έχουν προστεθεί στην g^* που επιστρέφεται και προστίθεται στο ανώνυμο σύνολο δεδομένων D^* (Αλγόριθμος 12, γραμμή 17).

Secant. Όταν οι μερικές παράγωγοι των συναρτήσεων δε είναι γνωστές, ή ο υπολογισμός τους είναι ακριβός, η μέθοδος Newton μπορεί να παραλλαχθεί παρέχοντας μια προσέγγιση για τις μερικές παραγώγους. Εξ ορισμού ισχύει ότι:

$$\frac{\partial f_i(X)}{\partial x_j}|_{x_j=v} = \lim_{dx \to 0} \frac{f_i(x_0, ..., v + dx, ...x_{n-1}) - f_i(x_0, ...v, ...x_{n-1})}{dx} \tag{7.2}$$

Συνεπώς, αν επιλεγεί ένα σημείο v_0 αρκετά κοντά στην τιμή v_{rj} ενός γνωρίσματος x_j, έτσι ώστε $|v_{rj} - v_0| < \epsilon$ για αρκετά μικρό $\epsilon > 0$, τότε η μερική παράγωγος $\partial f_i(X)/\partial x_j$ στο σημείο $x_j = v_{rj}$ μπορεί να προσεγγιστεί από το κλάσμα: $\frac{f_i(x_0,...,v_0,...x_{n-1}) - f_i(x_0,...v_{rj},...x_{n-1})}{v_0 - v_{rj}}$. Αυτή η προσέγγιση της μερικής παραγώγου στην μέθοδο Newton είναι γνωστή ως μέθοδος Secant. Η μόνη πληροφορία που χρειαζόμαστε για τις συναρτήσεις είναι τα σημεία που αλλάζει η μονοτονία της συνάρτησης f_i για το γνώρισα x_j, δηλαδή τις ρίζες της $\partial f_i/\partial x_j$, ώστε να μην επιλεγούν τα v_{rj} και v_0 εκατέρωθεν ενός σημείου μηδενισμού. Ο ψευδοκώδικας του $Generalize\text{-}Secant()$ παρουσιάζεται στον Αλγόριθμο 18.

Έστω v_n είναι η μικρότερη τιμή για την οποία αλλάζει η μονοτονία της f_i ως προς το x_j και για την οποία ισχύει $v_n > v_{rj}$. Ονομάζουμε το σημείο αυτό το «επόμενο» (next) σημείο αλλαγής της μονοτονίας. Επιλέγουμε ένα τυχαίο σημείο v_0 μεταξύ των v_{rj} και v_n, στη γραμμή 11. Η τιμή v_0 μπορεί να αναπαρασταθεί ως $v_{rj} + \alpha \cdot (v_n - v_{rj})$, όπου το α είναι επαρκώς μικρή θετική πραγματική παράμετρος που επιλέγεται από τον χρήστη, έτσι ώστε $0 < \alpha \ll 1$. Στην συνέχεια, στις γραμμές 12 και 13 υπολογίζονται τα y_r και y_0, δηλαδή οι τιμές της συνάρτησης $f_i(X)$ για $x_j = v_{rj}$ και για $x_j = v_0$ αντιστοίχως, ενώ οι τιμές των υπολοίπων γνωρισμάτων δεν μεταβάλλονται. Συνεπώς, η μερική παράγωγος $\partial f_i/\partial x_j$ προσεγγίζεται από τον λόγο των διαφορών $(y_0 - y_r)/(v_0 - v_{rj})$.

Η νέα εκτίμηση του x_j υπολογίζεται στις γραμμές 14-15. Έστω $\Delta f_i = (limit - y_r)$ η διαφορά μεταξύ της επιθυμητής τιμής $limit$ για την συνάρτηση f_i και της τρέχουσας τιμής y_r. Έστω Δx η (θετική η αρνητική) τιμή που πρέπει να προσθέσουμε στην τρέχουσα τιμή του γνωρίσματος v_{jr} ώστε η f_i να προσεγγίσει την $limit$. Αντικαθιστώντας την μερική παράγωγο στην εξίσωση 7.1 έχουμε:

Αλγόριθμος 18 Generalize-Secant

Require: g: group of records, $f_i()$: function, *upper*: **true** if we consider the maximum value of the function in the group, **false** otherwise, *limit*: either *min* or *max* value of f_i in g.

Ensure: g^*: all records are indistinguishable w.r.t $f_i(X)$.

1: $g^* = \emptyset$
2: Let $\mathcal{I}_{min}, \mathcal{I}_{max}$ be the limits of the attribute domain $\mathcal{I}$.
3: Let x_j be the j^{th} attribute of D.
4: Let $range_{rj} = [min_{rj}, max_{rj}]$ be the value range of the j^{th} attribute at the r^{th} record. Initially $min_{rj} = max_{rj}$ = initial value of x_{rj}.
5: Let $v_{rj} = \frac{min_{rj}+max_{rj}}{2}$. {If not generalized, v_{rj} is the initial value.}
6: **for all** records $t_r \in g$ **do**
7: $\quad \Delta f_i = limit - f_i(X)|_{x_j=v_{rj}, \forall j}$
8: $\quad$ **while** $|\Delta f_i| > threshold$ **do**
9: $\quad\quad$ **for all** attributes $X_j \in args(f_i)$ **do**
10: $\quad\quad\quad$ Let v_n be the next point of change in monotonicity.
11: $\quad\quad\quad$ Pick a $v_0 = v_{rj} + \alpha \cdot (v_n - v_{rj})$, where $0 < \alpha \ll 1$. $\quad \{ v_{rj} < v_0 < v_n \}$
12: $\quad\quad\quad$ $y_r = f_i(X)|_{x_j=v_{jr}}$
13: $\quad\quad\quad$ $y_0 = f_i(X)|_{x_j=v_0}$
14: $\quad\quad\quad$ $\Delta x = w \cdot \Delta f_i \cdot \frac{(v_0 - v_{rj})}{y_0 - y_r} \qquad \{w \in (0, 1]\}$
15: $\quad\quad\quad$ $v_{new} = v_{jr} + \Delta x$
16: $\quad\quad\quad$ **if** $v_{new} < min_{rj}$ **then**
17: $\quad\quad\quad\quad$ $v_{new} = min\{v_{new}, \mathcal{I}_{min}\}$
18: $\quad\quad\quad\quad$ $range_{rj} = [v_{new}, max_{rj}]$
19: $\quad\quad\quad$ **else if** $v_{new} > max_{rj}$ **then**
20: $\quad\quad\quad\quad$ $v_{new} = max\{v_{new}, \mathcal{I}_{max}\}$
21: $\quad\quad\quad\quad$ $range_{rj} = [min_{rj}, v_{new}]$
22: $\quad\quad\quad$ $v_{rj} = v_{new}$
23: $\quad\quad\quad$ $\Delta f_i = limit - f_i(X)|_{x_j=v_{new}}$
24: $\quad g^* = g^* \cup \{t_r\}$ {Add anonymized record t_r to g^*.}
25: Return g^*.

$$v_n = v_{rj} - \frac{(y_r - limit)}{\partial f_i / \partial x_j} \approx v_{rj} - \Delta f \cdot \frac{v_0 - v_{rj}}{y_0 - y_r}$$

$$\Rightarrow \Delta x = v_n - v_{rj} \approx -\Delta f \cdot \frac{v_0 - v_{rj}}{y_0 - y_r}$$

Για να γίνουν ισότιμα γενικεύσεις από όλα τα γνωρίσματα, αντί να υπερ-γενικευθεί ένα από αυτά σε κάθε εγγραφή, το Δx πολλαπλασιάζεται με ένα θετικό βάρος $w < 1$. Ο ρόλος του w εξηγείται στην ενότητα πειραματικής ανάλυσης. Η νέα εκτίμηση v_{new} για το γνώρισμα x_j στην εγγραφή t_r είναι η τιμή $v_{rj} + w \cdot \Delta x$. Το εύρος γενίκευσης αυτού του γνωρίσματος

Αλγόριθμος 19 Generalize-revF-w

Require: g: group of records, $f_i()$: function, *upper*: **true** if we consider the maximum value of the function in the group, **false** otherwise, *limit*: either *min* or *max* value of f_i in g.

Ensure: g^*: all records are indistinguishable w.r.t $f_i(X)$.

1: $g^* = \emptyset$
2: Let $\mathcal{I}_{min}, \mathcal{I}_{max}$ be the limits of the attribute domain $\mathcal{I}$.
3: Let x_j be the j^{th} attribute of D.
4: Let $range_{rj} = [min_{rj}, max_{rj}]$ be the value range of the j^{th} attribute at the r^{th} record. Initially $min_{rj} = max_{rj}$ = initial value of x_{rj}.
5: Let $v_{rj} = \frac{min_{rj}+max_{rj}}{2}$. {If not generalized, v_{rj} is the initial value.}
6: **for all** records $t_r \in g$ **do**
7: $\quad \Delta f_i = limit - f_i(X)|_{x_j=v_{rj},\forall j}$
8: $\quad$ **while** $|\Delta f_i| > threshold$ **do**
9: $\quad\quad$ **for all** attributes $X_j \in args(f_i)$ **do**
10: $\quad\quad\quad \Delta x = reverseF_i(limit, t_r, j) - v_{rj}$
11: $\quad\quad\quad Sum = \sum_{\sigma=1}^{n} \left|\frac{\partial f_i}{\partial x_\sigma}\right|_{x_\sigma=v_{r\sigma}}$
12: $\quad\quad\quad w_j = \frac{1}{Sum} \cdot \left|\frac{\partial f_i}{\partial x_{rj}}\right|_{x_j=x_{rj}}$
13: $\quad\quad\quad v_{new} = v_{jr} + w_j \cdot \Delta x$
14: $\quad\quad\quad$ **if** $v_{new} < min_{rj}$ **then**
15: $\quad\quad\quad\quad v_{new} = min\{v_{new}, \mathcal{I}_{min}\}$
16: $\quad\quad\quad\quad range_{rj} = [v_{new}, max_{rj}]$
17: $\quad\quad\quad$ **else if** $v_{new} > max_{rj}$ **then**
18: $\quad\quad\quad\quad v_{new} = max\{v_{new}, \mathcal{I}_{max}\}$
19: $\quad\quad\quad\quad range_{rj} = [min_{rj}, v_{new}]$
20: $\quad\quad\quad v_{rj} = v_{new}$
21: $\quad\quad\quad \Delta f_i = limit - f_i(X)|_{x_j=v_{new}}$
22: $\quad g^* = g^* \cup \{t_r\}$ {Add anonymized record t_r to g^*.}
23: Return g^*.

της t_r ενημερώνεται στις γραμμές 16-21. Σημειώνεται ότι αν το γνώρισμα x_j της t_r δεν ήταν ήδη γενικευμένο, ως όρια του εύρους του θεωρούνται τα $min_j = max_j = v_{rj}$ δηλαδή ίσα με την αρχική τιμή του x_j στην t_r.

Χρήση των Αντίστροφων Συντρήσεων

revF-w. Ο Αλγόριθμος 19 χειρίζεται συναρτήσεις που μπορούν να αντιστραφούν για όλα τα γνωρίσματα στο διάστημα που ορίζεται από το αρχικό πεδίο τιμών $\mathcal{I}$. Υλοποιήθηκε η συνάρτηση $reverseF_i(V, X, j)$ που επιστρέφει την τιμή ενός γνωρίσματος x_j που κάνει την συνάρτηση f_i να λάβει την τιμή V, δοσμένων των τιμών των υπολοίπων γνωρισμάτων που ανήκουν στο σύνολο $args(f_i) \subseteq X$. Όταν υπάρχουν περισσότερες τιμές του x_j που ικανοποιούν

αυτό το κριτήρια (λ.χ. σε πολυωνυμικές συναρτήσεις), τότε η συνάρτηση $reverseF_i(V, X, j)$ επιστρέφει εκείνη την λύση που είναι πιο κοντά στην v_{rj}, την τρέχουσα τιμή του x_j. Παραδείγματος χάριν, αν $f_i(X) = (x_1)^2$, $limit = 16$ και η τρέχουσα τιμή του x_1 είναι -2, τότε η $reverseF_1$ θα επιστρέψει -4, Αν αντίθετα η τρέχουσα τιμή ήταν θετική, τότε η $reverseF_1$ θα επέστρεφε +4. Ο λόγος πίσω από αυτή την επιλογή είναι γιατί προσπαθούμε να βρούμε το μικρότερο εύρος γενίκευσης των γνωρισμάτων που κάνουν το εύρος τιμών της f_i να περιέχει την τιμή $limit$.

Σημειώνεται ότι κάθε συνάρτηση πρέπει να αντιστραφεί για κάθε γνώρισμα ξεχωριστά, κρατώντας σταθερά τα υπόλοιπα. Αν το πρώτο γνώρισμα που εξετάζεται γενικευθεί στο εύρος που ορίζεται από την αρχική του τιμή και την τιμή που επιστρέφει η $reverseF_i$, τότε τα υπόλοιπα ορίσματα της f_i δεν θα χρειαζόταν να εξεταστούν καθώς θα είχαμε ήδη φτάσει στην επιθυμητή τιμή $limit$. Εντούτοις, αυτή μπορεί να μην είναι η βέλτιστη επιλογή γενίκευσης, καθώς άλλα γνωρίσματα μπορεί να χρειάζονται μικρότερα εύρη γενίκευσης για να κάνουν την f_i να λάβει την τιμή $limit$. Για τον λόγο αυτό, το βήμα $\Delta x = reverseF_i(V, X, j) - v_{ij}$ που θα προσθέταμε στην τρέχουσα τιμή v_{ij} (γραμμή 10) πολλαπλασιάζεται μένα βάρος $w_j \in [0, 1]$ πριν το προσθέσουμε στην v_{ij}, στη γραμμή 13 του αλγορίθμου. Με αυτό τον τρόπο, κάθε όρισμα της f_i θα γενικευθεί σε ένα μέρος του εύρους που θα χρειαζόταν για να φτάσει η f_i στο όριο $limit$, αν γενικευόταν μόνο αυτό το γνώρισμα.

Το βάρος w_j ορίζεται ως ο λόγος του $|\partial f_i/\partial x_j|$ για $x_j = v_{rj}$ προς το συνολικό άθροισμα όλων των μερικών παραγώγων $|\partial f_i/\partial x_\sigma|, \forall x_\sigma \in args(f_i)$, στο $x_\sigma = v_{r\sigma}$ της εγγραφής t_r. Το σκεπτικό πίσω από αυτή την επιλογή είναι ότι όλα τα γνωρίσματα θα πρέπει να γενικευθούν ανάλογα με την απόλυτη τιμή της μερικής τους παραγώγου, η οποία εκφράζει τον ρυθμό μεταβολής της συνάρτησης f_i ως προς το γνώρισμα x_j στο σημείο $x_j = v_{rj}$. Κανονικοποιούμε αυτό το μέγεθος στο [0, 1] διαιρώντας με το άθροισμα των απολύτων τιμών των μερικών παραγώγων όλων των ορισμάτων της f_i.

revF-minDx. Εναλλακτικά, μπορεί να υπολογιστεί πρώτα η $reverseF_i$ για κάθε ένα από τα ορίσματα της συνάρτησης f_i και στην συνέχεια να αποφασιστεί να γενικευθεί μόνο εκείνο το όρισμα που απαιτεί το μικρότερο εύρος γενίκευσης. Αυτή η μέθοδος περιγράφεται από τον ψευδοκώδικα του Αλγορίθμου 20, όπου πρώτα υπολογίζεται το απαιτούμενο βήμα Δx_j όπως ορίζεται από την $reverseF_i$ στην γραμμή 11, για κάθε γνώρισμα. Αν η νέα τιμή $v_{rj} + \Delta x_j$ θα ξεπερνούσε τα όρια του αρχικού πεδίου τιμών $\mathcal{I}$, τότε μικραίνουμε το βήμα Δx_j όσο χρειάζεται στις γραμμές 12-15. Κρατάμε το τρέχον βέλτιστο γνώρισμα x_b το οποίο χρειάζεται το μικρότερο εύρος γενίκευσης δηλαδή δίνει το ελάχιστο $|\Delta x_b|$. Μετά την εξέταση όλων των ορισμάτων της f_i, στις γραμμές 19-23 ενημερώνεται το εύρος γενίκευσης του «καλύτερου» γνωρίσματος x_b. Ονομάζουμε αυτόν τον αλγόριθμο `revF-minDx`.

revF-onepass. Σημειώνεται ότι αν το $v_{rb} + \Delta x_b$ ήταν η τιμή που επέστρεψε η $reverseF_i$ και βρισκόταν εντός των $[\mathcal{I}_{min}, \mathcal{I}_{max}]$, τότε η τιμή $limit$ της f_i θα έχει επιτευχθεί από την πρώτη επανάληψη. Διαφορετικά, αν το $|\Delta x_b|$ μειωθεί στις γραμμές 12-15, ο Αλγόριθμος 20 θα χρειαστεί περισσότερες επαναλήψεις για να τερματίσει. Για να αποφευχθεί αυτό και να μειωθεί ο χρόνος εκτέλεσης, υλοποιήθηκε μια παραλλαγή του `revF-minDx`, στην οποία προτιμάται η επιλογή του γνωρίσματος εκείνου που χρειάζεται το μικρότερο $|\Delta x_b|$ ανάμεσα σε εκείνα που

Αλγόριθμος 20 Generalize-revF-minDx

Require: g: group of records, $f_i()$: function, *upper*: **true** if we consider the maximum value of the function in the group, **false** otherwise, *limit*: either *min* or *max* value of f_i in g.

Ensure: g^*: all records are indistinguishable w.r.t $f_i(X)$.

1: $g^* = \emptyset$
2: Let $\mathcal{I}_{min}, \mathcal{I}_{max}$ be the limits of the attribute domain $\mathcal{I}$.
3: Let x_j be the j^{th} attribute of D.
4: Let $range_{rj} = [min_{rj}, max_{rj}]$ be the value range of the j^{th} attribute at the r^{th} record. Initially $min_{rj} = max_{rj}$ = initial value of x_{rj}.
5: Let $v_{rj} = \frac{min_{rj}+max_{rj}}{2}$. {If not generalized, v_{rj} is the initial value.}
6: **for all** records $t_r \in g$ **do**
7: $\quad \Delta f_i = limit - f_i(X)|_{x_j=v_{rj},\forall j}$
8: $\quad$ **while** $|\Delta f_i| > threshold$ **do**
9: $\quad\quad mindx = \mathcal{I}_{max} - \mathcal{I}_{min}$ {initialization}
10: $\quad\quad$ **for all** attributes $X_j \in args(f_i)$ **do**
11: $\quad\quad\quad \Delta x_j = reverseF_i(limit, t_r, j) - v_{rj}$
12: $\quad\quad\quad$ **if** $v_{rj} + \Delta x_j < \mathcal{I}_{min}$ **then**
13: $\quad\quad\quad\quad \Delta x_j = \mathcal{I}_{min} - v_{rj}$
14: $\quad\quad\quad$ **else if** $v_{rj} + \Delta x_j > \mathcal{I}_{max}$ **then**
15: $\quad\quad\quad\quad \Delta x_j = \mathcal{I}_{max} - v_{rj}$
16: $\quad\quad\quad$ **if** $|Deltax_j| < mindx$ **and** $|\Delta x_j| \neq 0$ **then**
17: $\quad\quad\quad\quad b = j$ $\quad \{x_b$ is the attribute with min$|\Delta x_j|\}$
18: $\quad\quad\quad\quad mindx = |\Delta x_j|$
19: $\quad\quad v_{new} = v_{rb} + \Delta x_b$
20: $\quad\quad$ **if** $v_{new} < min_{rb}$ **then**
21: $\quad\quad\quad range_{rb} = [v_{new}, max_{rb}]$
22: $\quad\quad$ **else if** $v_{new} > max_{rb}$ **then**
23: $\quad\quad\quad range_{rb} = [min_{rb}, v_{new}]$
24: $\quad\quad v_{rb} = v_{new}$
25: $\quad\quad \Delta f_i = limit - f_i(X)|_{x_b=v_{new}}$
26: $\quad g^* = g^* \cup \{t_r\}$ {Add anonymized record t_r to g^*.}
27: Return g^*.

$\delta\epsilon\nu$ μειώθηκαν στις γραμμές 12-15, αν μια τέτοια λύση υπάρχει. Διαφορετικά, όταν όλες οι υποψήφιες γενικεύσεις ξεπερνούν τα όρια του $\mathcal{I}$, γενικεύει το ίδιο γνώρισμα που θα επέλεγε και ο `revF-minDx`. Ονομάζουμε αυτήν την παραλλαγή του αλγορίθμου ως `revF-op`.

revF-groupBounds. Μια άλλη παραλλαγή του αλγορίθμου είναι η αντικατάσταση των ορίων $\mathcal{I}_{min}, \mathcal{I}_{max}$ στις συνθήκες των γραμμών 12-15 του Αλγορίθμου 20, με την ελάχιστη τιμή min_j και μέγιστη τιμή max_j αντίστοιχα που εμφανίζονται στην ομάδα εγγραφών g για το γνώρισμα j. Στην χειρότερη περίπτωση, κάθε γνώρισμα x_j γενικευθεί στο εύρος

$[min_j, max_j]$. Αυτή η τροποποίηση θυμίζει την k-ανωνυμία (μόνο για το χειρότερο σενάριο), αλλά δεν εγγυάται ότι η μέγιστη απώλεια πληροφορίας είναι η ίδια με την κλασσική εγγύηση. Ο λόγος είναι ότι η συσταδοποίηση των εγγραφών στην πρώτη φάση της μεθόδου βασίστηκε στις τιμές των συναρτήσεων και όχι στις τιμές των γνωρισμάτων. Έτσι οι ομαδοποίηση δεν είναι βελτιστοποιημένη για την k-ανωνυμία. Εντούτοις, δεν περιμένουμε η μέγιστη απώλεια πληροφορίας να είναι σημαντικά μεγαλύτερη στο χειρότερο σενάριο, ανάλογα με τις τιμές των συναρτ6ήσεων και των γνωρισμάτων. Αυτή η παραλλαγή καλείται `revF-gB`. Μπορεί επίσης να συνδυαστεί με την επιλογή onepass που περιγράφηκε στην προηγούμενη παράγραφο για την δημιουργία του εναλλακτικού αλγορίθμου `revF-op-gB`.

Συνδυάζοντας τα γνωρίσματα και τις συναρτήσεις

combXF. Αναπτύξαμε έναν επιπλέον αλγόριθμο βασισμένο σε μια διαφορετική προσέγγιση. Ο `combXF` υπολογίζει πρώτα την $reverseF_i$ για όλα τα ορίσματα όλων των συναρτήσεων του προβλήματος, για μία εγγραφή κάθε φορά, και στην συνέχεια αποφασίζει πως θα γενικεύσει τα γνωρίσματα.

Δημιουργούμε τον $2m \times n$ πίνακα Δ, του οποίου οι $2m$ γραμμές αντιστοιχούν στις μέγιστες και ελάχιστες επιθυμητές τιμές ($limit$) για καθεμία από τις συναρτήσεις του συνόλου $\{f_0, f_1, ..., f_{m-1}\}$. Οι n στήλες αντιστοιχούν σε καθένα από τα γνωρίσματα των δεδομένων $\{x_0, x_1, ..., x_{n-1}\}$. Συμβολίζουμε το στοιχείο της i-στης γραμμής και j-στης στήλης του πίνακα Δ ως $\delta_{i,j}$.

Η τιμή του στοιχείου $\delta_{2i,j}$ στην $2i$-στη γραμμή και j-στη στήλη είναι το βήμα Δx_j που θα χρειαζόταν πα προσθέσουμε στο άνω ή κάτω όριο του τρέχοντος εύρους τιμών του γνωρίσματος x_j, έτσι ώστε να λάβει η συνάρτηση f_i την *μέγιστη επιθυμητή τιμή*, αν $x_j \in args(f_i)$. Το στοιχείο $\delta_{2i+1,j}$ αντιστοιχεί στο βήμα που χρειάζεται για να επιτύχουμε την *ελάχιστη επιθυμητή τιμή* της f_i. Σημειώνεται ότι οι συναρτήσεις είναι ταξινομημένες από την μέγιστη προς την ελάχιστη προτεραιότητα. Σε αυτήν την περίπτωση ο δείκτης i της f_i δείχνει την θέση της σε αυτή την ταξινόμηση.

Αν η τιμή του βήματος $\delta_{2i,j}$ (ή του $\delta_{2i+1,j}$ αντίστοιχα) είναι θετική, τότε αυτή είναι η τιμή που πρέπει να προσθέσουμε στην μέγιστη τιμή του τρέχοντος εύρους γενίκευσης του γνωρίσματος x_j, για την εγγραφή που εξετάζουμε. Διαφορετικά, αν είναι αρνητική την προσθέτουμε στο ελάχιστο όριο του εύρους του x_j, ουσιαστικά επεκτείνοντάς το προς τα κάτω. Αν ένα γνώρισμα x_j μιας εγγραφής δεν ανήκει στην λίστα των ορισμάτων της συνάρτησης f_i, τότε υποχρεωτικά τα $\delta_{2i,j}$ και $\delta_{2i+1,j}$ τίθενται ως μηδέν, το οποίο σημαίνει καμία επιπλέον γενίκευση. Υπενθυμίζεται ότι όταν η τιμή ενός γνωρίσματος x_j μιας εγγραφής t_r δεν έχει γενικευθεί ακόμα, τα όρια του αρχικού «εύρος» γενίκευσής του θεωρούνται ίσα με την αρχική του τιμή $min_j = max_j = v_{rj}$.

Η δημιουργία του πίνακα Δ παρουσιάζεται στον ψευδοκώδικα του Αλγόριθμου 21. Η διαδικασία διατρέχει την λίστα όλων των συναρτήσεων, ταξινομημένη κατά σειρά προτεραιότητας στην γραμμή 2, Για κάθε γνώρισμα x_j το οποίο ανήκει στην λίστα των ορισμάτων της f_i, υπολογίζεται, με κλήση της $reverseF_i$, η τιμή $temp$ που θα έπρεπε να λάβει το x_j ώστε η

Αλγόριθμος 21 CalculateΔ

Require: t_r: record, F: set of functions.

Ensure: $\Delta_{2m\times n}$: matrix of the steps needed to reach upper/lower limits of functions.

1: let $[min_{rj}, max_{rj}]$ be the current generalization range of the j^{th} attribute of record t_r.
2: **for all** functions $f_i(X) \in F$ {`ordered by priority`} **do**
3: **for all** attributes $X_j \in args(f_i)$ **do**
4: $temp = reverseF_i(f_i^{max}, t_r, j)$ {for the maximum value of f_i}
5: **if** $temp \in [min_{rj}, max_{rj}]$ **then**
6: set all $\delta_{2i,\tau} = 0, \forall \tau = 1, ..., n$. {There is a value combination from the current attribute ranges that makes f_i equal to f_i^{max}.}
7: goto 16.
8: **else if** $temp \in (max_{rj}, \mathcal{I}_{max}]$ **then**
9: $\delta_{2i,j} = temp - max_{rj}$
10: **else if** $temp \in [\mathcal{I}_{min}, min_{rj})$ **then**
11: $\delta_{2i,j} = temp - min_{rj}$
12: **else if** $temp < \mathcal{I}_{min}$ **then**
13: $\delta_{2i,j} = \mathcal{I}_{min} - min_{rj}$
14: **else if** $temp > \mathcal{I}_{max}$ **then**
15: $\delta_{2i,j} = \mathcal{I}_{max} - max_{rj}$
16: $temp = reverseF_i(f_i^{min}, t_r, j)$ {for the minimum value of f_i}
17: **if** $temp \in [min_{rj}, max_{rj}]$ **then**
18: set all $\delta_{2i+1,\tau} = 0, \forall \tau = 1, ..., n$. {There is a value combination from the current attribute ranges that makes f_i equal to f_i^{min}.}
19: j++. goto 4.
20: **else if** $temp \in (max_{rj}, \mathcal{I}_{max}]$ **then**
21: $\delta_{2i+1,j} = temp - max_{rj}$
22: **else if** $temp \in [\mathcal{I}_{min}, min_{rj})$ **then**
23: $\delta_{2i+1,j} = temp - min_{rj}$
24: **else if** $temp < \mathcal{I}_{min}$ **then**
25: $\delta_{2i+1,j} = \mathcal{I}_{min} - min_{rj}$
26: **else if** $temp > \mathcal{I}_{max}$ **then**
27: $\delta_{2i+1,j} = \mathcal{I}_{max} - max_{rj}$
28: return Δ

f_i να φτάσει σε ένα επιθυμητό άνω όριο f_i^{max} στην γραμμή 4 του ψευδοκώδικα. Αν η τιμή αυτή περιέχεται ήδη στο τρέχον εύρος γενίκευσης $[min_{rj}, max_{rj}]$ του γνωρίσματος, τότε δεν χρειάζεται καμία επιπλέον γενίκευση διότι υπάρχει ήδη συνδυασμός τιμών μέσα στα εύρη των γνωρισμάτων $\{x_0, x_1, ..., x_{n-1}\}$ που δίνει στην f_i την τιμή f_i^{max}. Όλα τα στοιχεία αυτής της γραμμής του πίνακα τίθενται ίσα με μηδέν στην γραμμή 6, για να δηλωθεί ότι δεν χρειάζονται επιπλέον γενικεύσεις για την συγκεκριμένη εγγραφή όσον αφορά στο όριο f_i^{max} της συνάρτησης f_i. Η διαδικασία προχωρά στην επόμενη *ελάχιστη επιθυμητή τιμή* της f_i στην γραμμή 16.

Διαφορετικά, στις γραμμές 8-13 εξετάζεται αν η τιμή $\delta_{2i,j}$ είναι πάνω από max_{rj} ή κάτω από min_{rj}. Στην πρώτη περίπτωση το στοιχείο $\delta_{2i,j}$ ενημερώνεται λαμβάνοντας ως τιμή την διαφορά ανάμεσα στην τιμή $temp$ που μόλις υπολογίστηκε μείον το max_{rj}, έτσι ώστε $\delta_{2i,j} > 0$. Στην δεύτερη περίπτωση, το στοιχείο $\delta_{2i,j}$ ενημερώνεται λαμβάνοντας ως τιμή την διαφορά ανάμεσα στην τιμή $temp$ που μόλις υπολογίστηκε μείον το min_{rj}, έτσι ώστε $\delta_{2i,j} < 0$.Τέλος, στις γραμμές 12-15 ελέγχεται αν η νέα τιμή $temp$ ξεπερνά τα επιτρεπτά όρια του πεδίου των γνωρισμάτων $[\mathcal{I}_{min}, \mathcal{I}_{max}]$, οπότε το βήμα $\delta_{2i,j}$ προσαρμόζεται αντίστοιχα.

Με τον ίδιο τρόπο, η διαδικασία προχωρά στις γραμμές 16-27 και υπολογίζει την τιμή του j-στου στοιχείου της $(2i+1)$-γραμμής του πίνακα Δ, ως προς την ελάχιστη επιθυμητή τιμή f_i^{min} της συνάρτησης f_i.

Οι τιμές του πίνακα είναι *προτεινόμενες* γενικεύσεις. Σε κάθε γραμμή του πίνακα αρκεί να γίνει μία ή λίγες από αυτές για φτάσει η αντίστοιχη συνάρτηση την αντίστοιχη επιθυμητή τιμή που αντιστοιχεί στην γραμμή του πίνακα. Δύο βασικά κριτήρια λαμβάνονται υπόψη για την απόφαση αν πρέπει ή όχι το εύρος γενίκευσης ενός γνωρίσματος να μεγαλώσει κατά το βήμα $\delta_{\mu,j}$, $\mu = \{1, ..., 2m\}$:

(i) αν η αύξηση του άνω (ή κάτω αντίστοιχα) ορίου του x_j κατά $\delta_{\mu,j}$ καλύπτει επίσης άλλες προτεινόμενες γενικεύσεις $\delta_{\lambda,j}$ του ίδιου γνωρίσματος για άλλες επιθυμητές τιμές των συναρτήσεων (δηλαδή στην ίδια στήλη του πίνακα)

(ii) αν είναι το ελάχιστο $|\delta_{\mu,j}|$ ανάμεσα σε όλα τα προτεινόμενα μη-μηδενικά βήματα των γνωρισμάτων για αυτή την γραμμή του πίνακα.

Το πρώτο είναι σημαντικότερο του δεύτερου, καθώς μας γλυτώνει από περιττές γενικεύσεις που θα πραγματοποιούνταν για άλλες συναρτήσεις.

Παράδειγμα 7.20. *Έστω ότι με την παραπάνω διαδικασία έχει δημιουργηθεί ο παρακάτω πίνακας για μια εγγραφή των δεδομένων με δυο γνωρίσματα μιας συνάρτησης f_0:*

$$\Delta = \begin{bmatrix} 10 & 5 \\ 7 & -10 \end{bmatrix}$$

Το στοιχείο $\delta_{0,0}$ έχει την τιμή 10 ενώ το $\delta_{0,1}$ είναι ίσο με 5. Αυτό σημαίνει ότι με την αύξηση του άνω ορίου του εύρους γενίκευσης του γνωρίσματος x_0 κατά +10, επιτυγχάνουμε την μέγιστη επιθυμητή τιμή της f_0. Το ίδιο θα μπορούσε να επιτευχθεί με την αύξηση του άνω ορίου του εύρους γενίκευσης του γνωρίσματος x_1 κατά μόλις +5. Παρόλο που η δεύτερη επιλογή εισάγει μικρότερη επιπρόσθετη απώλεια πληροφορίας (λόγω μικρότερης αύξησης στο εύρος γενίκευσης), η πρώτη επιλογή είναι προτιμότερη όταν δούμε την συνολική εικόνα του πίνακα Δ. Για να επιτύχουμε την ελάχιστη επιθυμητή τιμή της f_0, θα πρέπει είτε να προσθέσουμε +7 στο άνω όριο του εύρους γενίκευσης του γνωρίσματος x_0 ($\delta_{1,0} = 7$), είτε να αφαιρέσουμε 10 από το κάτω όριο του εύρους γενίκευσης του γνωρίσματος x_1 ($\delta_{1,1} = -10$). Συνεπώς, προσθέτοντας +10 στο άνω όριο του εύρους γενίκευσης του γνωρίσματος x_0 από το πρώτο βήμα, μπορούμε να φτιάξουμε και την μέγιστη και την ελάχιστη τιμή της f_0. Διαφορετικά, το σύνολο των επιπρόσθετων γενικεύσεων θα ξεπερνούσε την απώλεια πληροφορίας αυτής της μοναδικής γενίκευσης.

Αλγόριθμος 22 Generalize-combXF

Require: g: group of records, F: set of functions, ordered from higher to lower priority
Ensure: g^*: all records are indistinguishable w.r.t $f_i(X)$.

1: $g^* = \emptyset$
2: Let $range_{rj} = [min_{rj}, max_{rj}]$ be the value range of the j^{th} attribute at the r^{th} record.
3: Let $v_{rj} = \frac{min_{rj}+max_{rj}}{2}$. {If not generalized, v_{rj} is the initial value.}
4: **for all** records $t_r \in g$ **do**
5: $Calculate\Delta(t_r, F)$
6: **for all** lines $\mu = 1$ to $2{\cdot}m$ of array Δ **do**
7: $i = \mu \div 2$. {function index}
8: **if** $(\mu \mod 2) == 0$ **then**
9: $limit = f_i^{max}$
10: **else**
11: $limit = f_i^{min}$
12: **while** $\left|limit - f_i(X)|_{x_j=v_{rj},\forall j}\right| > threshold$ **do**
13: $bestDx = \mathcal{I}_{max} - \mathcal{I}_{min}$ {initializations}
14: $maxContrib = 0$
15: **for all** columns j of array Δ **do**
16: **for all** lines λ of array Δ, with $\lambda > \mu$ **do**
17: **if** $|\delta_{\mu,j} - \delta_{\lambda,j}| > 0$ **and** $\delta_{\mu,j} \cdot \delta_{\lambda,j} > 0$ **then**
18: $contrib[j]$ += $min\{|\delta_{\mu,j}|, |\delta_{\lambda,j}|\}$.
19: **if** $\{(contrib[j] > maxContrib)$
or $(contrib[j] == maxContrib$ **and** $|\delta_{\mu,j}| < bestDx)\}$ **then**
20: $b = j$
21: maxContrib = contrib[j]
22: $bestDx = |\delta_{\mu,j}|$
23: **if** $\delta_{\mu,b} < 0$ **then**
24: $range_{rb} = [min_{rb} + \delta_{\mu,b}, max_{rb}]$
25: **else if** $\delta_{\mu,b} > 0$ **then**
26: $range_{rb} = [min_{rb}, max_{rb} + \delta_{\mu,b}]$
27: $v_{rb} = v_{rb} + \delta_{\mu,b}/2$
28: $Calculate\Delta(t_r, F)$
29: $g^* = g^* \cup \{t_r\}$ {Add anonymized record t_r to g^*.}
30: Return g^*.

Ο ψευδοκώδικας του `combXF` παρουσιάζεται στον Αλγόριθμο 22. Ο αλγόριθμος λαμβάνει ως είσοδο μια ομάδα εγγραφών g και ένα σύνολο συναρτήσεων F, ταξινομημένο από την υψηλότερη ως την χαμηλότερη προτεραιότητα. Θεωρούμε ότι η αρίθμηση των συναρτήσεων έχει ενημερωθεί αντίστοιχα, δηλαδή η f_i έχει μεγαλύτερη προτεραιότητα από την f_{i+1}. Για κάθε εγγραφή της ομάδας g, αρχικοποιείται ο πίνακας Δ, λαμβάνοντας υπόψη τις συναρτήσεις και τις αρχικές τιμές των γνωρισμάτων, στην γραμμή 5. Εξετάζουμε τις γραμμές του πίνακα

Δ σειριακά από πάνω προς τα κάτω, και σε καθεμία αναζητούμε το στοιχείο που έχει την μεγαλύτερη *συνεισφορά* ($maxContrib$) σε εύρη γενίκευσης άλλων τιμών συναρτήσεων, για το ίδιο γνώρισμα.

Σημειώνεται ότι δοσμένης της γραμμή μ του πίνακα, η συνάρτηση στην οποία αντιστοιχεί είναι η f_i, όπου $i = \mu \div 2$. Ο τελεστής «$\div$» συμβολίζει το πηλίκο της ευκλείδειας διαίρεσης. Το υπόλοιπο της διαίρεσης $imod2$ μας πληροφορεί αν η γραμμή αντιστοιχεί σε μέγιστη επιθυμητή τιμή (f_i^{max}) αν είναι 0, ή σε ελάχιστη (f_i^{min}) αν είναι 1. Οι πληροφορίες αυτές υπολογίζονται στις γραμμές 7-11 του αλγόριθμου.

Ο βρόχος while στις γραμμές 12-27 εκτελεί γενικεύσεις επαναληπτικά ώστε να επιτύχει μια δοσμένη τιμή για την συνάρτηση f_i. Σε κάθε επανάληψη, η «καλύτερη» γενίκευση επιλέγεται ανάμεσα σε όλες τις υποψήφιες γενικεύσεις μιας γραμμής του πίνακα. Ο βασικός στόχος είναι να περιοριστεί η απώλεια πληροφορίας. Για αυτό τον λόγο προτιμούμε τις γενικεύσεις γνωρισμάτων οι οποίες «καλύπτουν» άλλα υποψήφια εύρη γενίκευσης του ίδιου γνωρίσματος για άλλες τιμές συναρτήσεων (επόμενες γραμμές του πίνακα Δ) που θα χρειαστεί να εξετάσουμε σε επόμενα βήματα. Αυτά τα επικαλυπτόμενα εύρη γενίκευσης αντιστοιχούν σε στοιχεία του πίνακα Δ τα οποία ανήκουν στην ίδια στήλη. Για να υπάρχει επικάλυψη μεταξύ τους, τα αντίστοιχα στοιχεία του πίνακα πρέπει να έχουν κοινό πρόσημο.

Δεδομένου ότι σε κάθε βήμα του αλγορίθμου κάνουμε μια συνάρτηση να λάβει μια συγκεκριμένη επιθυμητή τιμή, η οποία αντιστοιχεί σε μια γραμμή μ του πίνακα Δ, οι γραμμές πάνω από την μ έχουν φτιαχτεί και δεν χρειάζεται να ληφθούν υπόψη στα επόμενα βήματα. Έτσι, διατρέχουμε τις γραμμές του πίνακα από πάνω προς τα κάτω και σε κάθε βήμα εξετάζουμε όλα τα γνωρίσματα (γραμμή 15) και όλες τις γραμμές που είναι κάτω από την τρέχουσα για να δούμε την συνεισφορά γενίκευσης των στοιχείων της μ (γραμμή 16).

Όταν ένα στοιχείο του πίνακα $\delta_{\lambda,j}$, με $\lambda > \mu$, είναι ομόσημο με το στοιχείο $\delta_{\lambda,j}$, δηλαδή όταν $\delta_{\mu,j} \cdot \delta_{\lambda,j} > 0$, τότε οι υποψήφιες γενικεύσεις που τους αντιστοιχούν, για το γνώρισμα x_j, επικαλύπτονται. Στην γραμμή 18, η μη-μηδενική επικάλυψή τους $min\{|\delta_{\mu,j}|,|\delta_{\lambda,j}|\}$, προστίθεται στην συνολική *συνεισφορά* $contrib[j]$ της προτεινόμενης γενίκευσης $\delta_{\mu,j}$ για το γνώρισμα x_j. Η συνολική συνεισφορά που προκύπτει από επικαλύψεις με τις επόμενες γραμμές ($\forall \lambda > \mu$) ανά γνώρισμα x_j αποθηκεύεται στον μονοδιάστατο πίνακα $contrib[]$. Στις γραμμές 19-22 κρατάμε το τρέχον γνώρισμα με την μέγιστη συνεισφορά γενικεύσεων. Σε περίπτωση ισοπαλίας των συνεισφορών δυο γνωρισμάτων, επιλέγεται εκείνο με το μικρότερο επιπρόσθετο εύρος γενίκευσης, δηλαδή το μικρότερο $|\delta_{\mu,j}|$.

Παράδειγμα 7.21. *Ας υποθέσουμε ότι σε κάποιο ενδιάμεσο βήμα του αλγόριθμου, το τρέχον εύρος γενίκευσης του γνωρίσματος x_j είναι $[5, 10]$, και επίσης ότι $\delta_{\lambda,j} = 4$ και $\delta_{\mu,j} = 2$, $\mu > \lambda$. Η πρώτη προτεινόμενη γενίκευση θα αυξήσει το εύρος του διαστήματος του x_j σε $[5, 14]$, ενώ η δεύτερη σε $[5, 12]$. Η επικάλυψη των δυο προτεινόμενων (επιπρόσθετων) γενικεύσεων είναι $min\{|\delta_{\mu,j}|, |\delta_{\lambda,j}|\} = 2$. Σημειώνεται ότι για την επιλογή μιας νέας γενίκευσης δεν λαμβάνουμε υπόψη τις γενικεύσεις που είχε υποστεί το γνώρισμα x_j σε προηγούμενα βήματα (δηλαδή το εύρος του διαστήματος $[5, 10]$), μόνο τις επιπλέον γενικεύσεις του τρέχοντος βήματος. Αν στον πινάκα υπάρχει επόμενη γραμμή ν με στοιχείο $\delta_{\mu,j} = -2$, τότε αυτό δεν επικαλύπτεται με το $\delta_{\mu,j}$, καθώς θα μείωνε το κάτω όριο του διαστήματος του x_j σε $[3, 10]$.*

Εν τέλει, το επιλεγμένο γνώρισμα x_b γενικεύεται στις γραμμές 23-26. Αν ισχύει $\delta_{\mu,b} > 0$, προστίθεται στο άνω όριο του τρέχοντος εύρους του x_b, διαφορετικά προστίθεται στο κάτω όριο. Το διάστημα γίνεται $[min_{rb}, max_{rb} + \delta_{\mu,b}]$ στην πρώτη περίπτωση, ή $[min_{rb} + \delta_{\mu,b}, max_{rb}]$ στην δεύτερη. Σε κάθε περίπτωση, η νέα μέση τιμή v'_{rb} γίνεται $\frac{min_{rb}+max_{rb}+\delta_{\mu,b}}{2} = \frac{min_{rb}+max_{rb}}{2} + \frac{\delta_{\mu,b}}{2}$, ίση με την προηγούμενη τιμή v_{rb} αυξημένη κατά $\delta_{\mu,b}/2$, όπως φαίνεται στην γραμμή 27. Ο πίνακας Δ επαναϋπολογίζεται στην γραμμή 28, λαμβάνοντας υπόψη το νέο εύρος γενίκευσης του γνωρίσματος x_b.

Όταν η συνάρτηση f_i φτάσει σε μια επιθυμητή τιμή, ο βρόχος while τερματίζει και ο αλγόριθμος προχωρά στην επόμενη γραμμή του πίνακα Δ. Διαφορετικά, επαναλαμβάνεται η ίδια διαδικασία για την επιλογή της γενίκευσης ενός άλλου γνωρίσματος από την γραμμή μ του Δ. Σημειώνεται ότι κάτι τέτοιο συμβαίνει μόνον όταν η γενίκευση που επιλέχτηκε $\delta_{\mu,b}$ ήταν μειωμένη για να μην υπερβεί τα όρια $[\mathcal{I}_{min}, \mathcal{I}_{max}]$ από τις γραμμές 12-15 ή 24-27 του Αλγόριθμου 21.

Έχοντας εξετάσει όλες τις γραμμές του πίνακα Δ και έχοντας φτάσει τα μέγιστα και ελάχιστα επιθυμητά όρια όλων των συναρτήσεων, η ανωνυμοποιημένα εγγραφή t_r προστίθεται στην κλάση ισοδυναμίας g^* στην γραμμή 29 του αλγορίθμου. Όταν όλες οι εγγραφές έχουν k_f^m-ανωνυμοποιηθεί, η κλάση g^* επιστρέφεται ως αποτέλεσμα.

7.4 Πειραματική Μελέτη

Σε αυτή την ενότητα παρουσιάζονται τα πειραματικά αποτελέσματα για την αποτίμηση της μεθόδου. Όλες οι υλοποιήσεις έγιναν σε γλώσσα προγραμματισμού C++ και η διεξαγωγή των πειραμάτων έγινε σε υπολογιστή Intel Core i7 CPU 3.2GHz, με 6GB RAM, σε λειτουργικό σύστημα Ubuntu Linux.

7.4.1 Πειραματικά Δεδομένα

Χρησιμοποιήσαμε δυο πραγματικά σύνολα δεδομένων από την αποθήκη δεδομένων UCI data mining repository [9]. Το πρώτο είναι το IPUMS Census dataset [7], στο οποίο αναφερόμαστε ως `IPUMS`. Επιλέχθηκαν 5 αριθμητικά γνωρίσματα που αντιστοιχούν σε διαφορετικούς τύπους εισοδημάτων. Συνενώθηκαν οι εγγραφές των συνόλων από τα 97, 98 και 99 LA σύνολα δεδομένων ώστε να δημιουργηθεί ένα μεγαλύτερο 233,584 εγγραφών. Το πεδίο τιμών των γνωρισμάτων του είναι $\mathcal{I}_i$ = [-99999, 999999]. Μεταφέρθηκε στο [1, 1010000] ώστε να αποφευχθούν διαιρέσεις με τον μηδέν στις συναρτήσεις. Το δεύτερο σύνολο δεδομένων είναι το Individual household electric power consumption dataset [6], το οποίο περιέχει διάφορες μετρήσεις κατανάλωσης ηλεκτρικού ρεύματος από διάφορους καταναλωτές και στο οποίο αναφερόμαστε ως `ENERGY`. Μετά την απαλοιφή των πεδίων ημερομηνίας, ώρας και ηλεκτρικής τάσης (η οποία είναι σταθερή στις περισσότερες μετρήσεις), επιλέχθηκαν 5 γνωρίσματα που αντιστοιχούν σε μετρήσεις ηλεκτρικής κατανάλωσης. Το σύνολο των δεδομένων περιέχει 2,075,259 εγγραφές, Το συνολικό πεδίο των τιμών του είναι [0, 80] και οι τιμές εκφράζονται με ακρίβεια τριών δεκαδικών ψηφίων. Έτσι, το εύρος μεταφέρθηκε στο [0.001, 80.001] ώστε να αποφευχθούν οι διαιρέσεις με τον μηδέν.

7.4.2 Συναρτήσεις

Διεξήχθησαν δύο σειρές πειραμάτων. Στην πρώτη σειρά πειραμάτων χρησιμοποιήθηκαν πέντε τύποι συναρτήσεων: η συνάρτηση ταυτότητας, το άθροισμα όλων των γνωρισμάτων, το άθροισμα με βάρη, το άθροισμα γνωρισμάτων υψωμένα σε δυνάμεις και διαίρεση/πολλαπλασιασμός. Συγκεκριμένα, μελετήθηκε η ειδική περίπτωση των συναρτήσεων ταυτότητας και των πέντε γνωρισμάτων, ώστε να επιτευχθεί η εγγύηση της κλασσικής k-ανωνυμίας. Ο συμβολισμός που χρησιμοποιείται στα σχήματα για τους αντίστοιχους συνδυασμούς γνώσης του επιτιθέμενου και οι αντίστοιχες μαθηματικές του εκφράσεις είναι οι ακόλουθες:

- k-anonymity: $f_i(X) = x_i,\ \forall i \in \{1,2,3,4,5\}$
- addition: $f_6(X) = x_1 + x_2 + x_3 + x_4 + x_5$
- weights: $f_7(X) = 0.1x_1 + x_2 + 10x_3 + 100x_4 + 1000x_5$
- powers: $f_8(X) = x_1 + {x_2}^2 + {x_3}^3 + {x_4}^4 + {x_5}^5$
- division: $f_9(X) = (x_1 \cdot x_3)/(x_2 \cdot x_4 \cdot x_5)$

Στην δεύτερη σειρά πειραμάτων αυξάνουμε σταδιακά την γνώση του επιτιθέμενου από 1 σε 4 πολυωνυμικές συναρτήσεις. Κάθε συνδυασμός γνώσης είναι υπερσύνολο των προηγούμενων:

- 1 pol. func : $f_1(X) = {x_1}^2 + 5x_2$
- 2 pol. funcs: $f_1(X) = {x_1}^2 + 5x_2,\ f_2(X) = {x_3}^3 - x_2$
- 3 pol. funcs: $f_1(X) = {x_1}^2 + 5x_2,\ f_2(X) = {x_3}^3 - x_2,\ f_3(X) = {x_4}^3 + 100x_5$
- 4 pol. funcs: $f_1(X) = {x_1}^2 + 5x_2,\ f_2(X) = {x_3}^3 - x_2,\ f_3(X) = {x_4}^3 + 100x_5,$ $f_4(X) = {x_5}^2 - 2x_1$

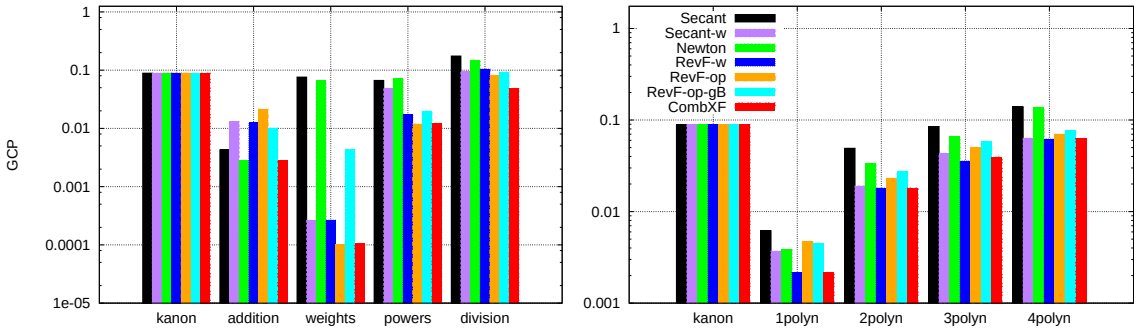
Οι δύο σειρές πειραμάτων εξυπηρετούν διαφορετικούς σκοπούς αξιολόγησης: Η πρώτη μελετά την συμπεριφορά της προτεινόμενης μεθόδου πάνω σε ένα εύρος διαφορετικών τύπων συναρτήσεων, ενώ στη δεύτερη θεωρούμε μόνο πολυωνυμική γνώση του επιτιθέμενου και μεταβάλλουμε το πλήθος των συναρτήσεων δηλαδή το μέγεθος γνώσης του επιτιθέμενου.

7.4.3 Αλγόριθμοι

Υλοποιήθηκαν όλοι οι προτεινόμενοι αλγόριθμοι όπως περιγράφηκαν στην Ενότητα 7.3.3 και εμπίπτουν σε τρεις βασικές κατηγορίες:

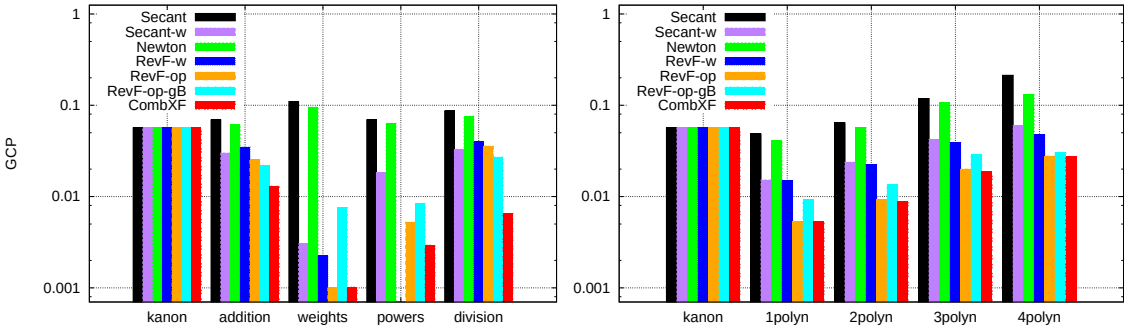
(i) Χρήση αριθμητικής ανάλυσης: Υλοποιήθηκαν οι `Newton` και `Secant`. Επιπλέον υλοποιήθηκε ο `Secant-w`, μια παραλλαγή του `Secant` που πολλαπλασιάζει κάθε βήμα γενίκευσης Δx με το ίδιο βάρος w όπως ο αλγόριθμος `revF-w`.

(ii) Χρήση της αντίστροφης συνάρτησης: Υλοποιήθηκαν οι αλγόριθμοι `revF-w`, `revF-minDx`, `revF-op`, `revF-gB` και `revF-op-gB`.



(α) 1^η σειρά πειραμάτων (β) 2^η σειρά πειραμάτων

Σχήμα 7.1: Συγκριτική μελέτη: GCP των δεδομένων `IPUMS`, με k=20.



(α) 1^η σειρά πειραμάτων (β) 2^η σειρά πειραμάτων

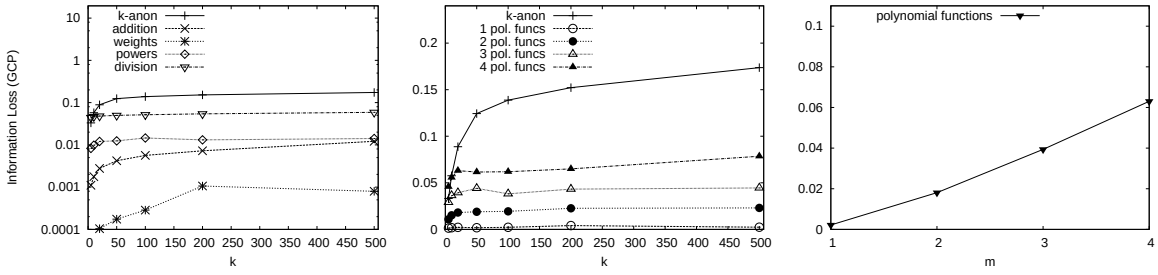
Σχήμα 7.2: Συγκριτική μελέτη: GCP των δεδομένων `ENERGY`, με k=20.

(iii) Συνδυάζοντας όλες τις συναρτήσεις και τα γνωρίσματα: Τέλος, υλοποιήθηκε ο `combXF` όπως περιγράφηκε στην Ενότητα 7.3.3.

Σημειώνεται ότι σε όλα τα πειράματα ο αλγόριθμος `revF-op` επιτυγχάνει είτε τα ίδια, είτε καλύτερα αποτελέσματα από εκείνα του `revF-minDx`, ενώ ακολουθούσε παρόμοια συμπεριφορά ως προς την μεταβολή της παραμέτρου k και ως προς το είδος και την ποσότητα της γνώσης του επιτιθέμενου. Το ίδιο ισχύει για τον αλγόριθμο `revF-op-gB` σε σύγκριση με τον `revF-gB`. Αυτό σημαίνει ότι η επιλογή «one-pass» επιτυγχάνει μικρότερη απώλεια πληροφορίας, όπου είναι εφικτό. Έτσι στις γραφικές παραστάσεις παρουσιάζονται μόνο τα αποτελέσματα των αλγόριθμων `revF-op` και `revF-op-gB`, ενώ παραλείπονται εκείνα των αλγορίθμων `revF-minDx` και `revF-gB`.

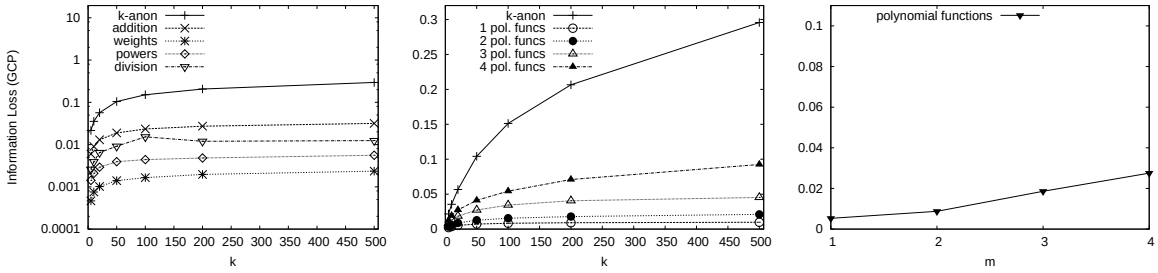
7.4.4 Παράμετροι

Μελετάμε την συμπεριφορά του προτεινόμενου αλγορίθμου ως προς την παράμετρο ανωνυμίας $k = \{5, 10, 20, 50, 100, 200, 500\}$, και το πλήθος των συναρτήσεων $m = \{1, 2, 3, 4\}$ που αντιπροσωπεύει το μέγεθος γνώσης του επιτιθεμένου στην δεύτερη σειρά πειραμάτων. Η προεπιλεγμένες τιμές των παραμέτρων που επιλέξαμε είναι $k = 20$ και $m = 1$, ενώ μετά από



(α) (β) (γ)

Σχήμα 7.3: Απώλεια πληροφορίας του `combXF` για τα δεδομένα `IPUMS`: επίδραση του k (α) για την 1^η και (β) 2^η σειρά πειραμάτων, (γ) επίδραση του m.



(α) (β) (γ)

Σχήμα 7.4: Απώλεια πληροφορίας του `combXF` για τα δεδομένα `ENERGY`: επίδραση του k (α) για την 1^η και (β) 2^η σειρά πειραμάτων, (γ) επίδραση του m.

κάποιες δοκιμές καταλήξαμε στις βέλτιστες για τα δεδομένα μας τιμές των παραμέτρων για την συσταδοποίηση των εγγραφών: $climit = 500$ και $threshold = 0.1$.

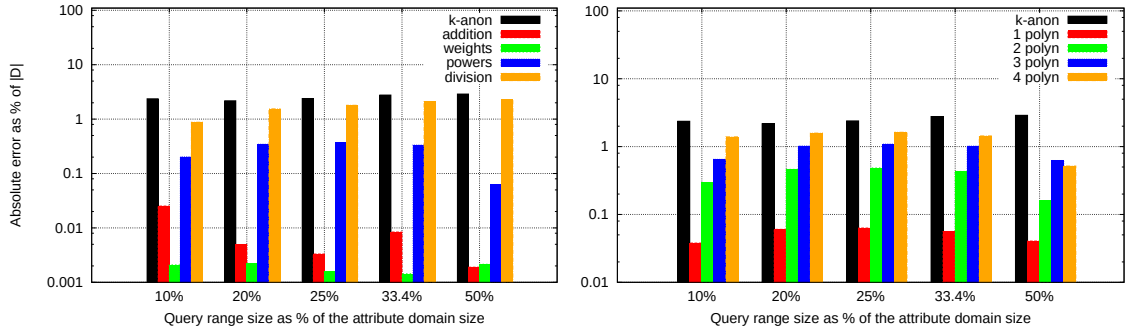
7.4.5 Μετρικές Αποτίμησης

Για την αξιολόγηση της μεθόδου μας μετρήθηκε ο χρόνος εκτέλεσης των πειραμάτων σε δευτερόλεπτα καθώς και η απώλειας πληροφορίας με την μετρική GCP (βλ. Ενότητα 5.3.3) η οποία είναι εξορισμού κανονικοποιημένη στο εύρος $[0, 1]$, για την αποτίμηση της ποιότητας των αποτελεσμάτων. Επίσης, μετρήθηκε το μέσο απόλυτο σφάλμα για ερωτήματα εύρους τιμών πάνω στα ανωνυμοποιημένα δεδομένα.

7.4.6 Ποιότητα Αποτελεσμάτων

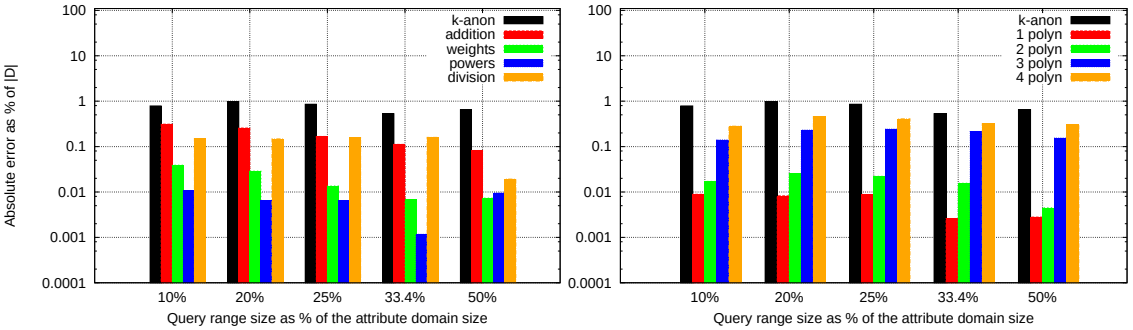
Σύγκριση των Αλγορίθμων

Στα Σχήματα 7.1 και 7.2 παρουσιάζεται η συγκριτική μελέτη όλων των αλγορίθμων ως προς την απώλεια πληροφορίας και των δύο σειρών πειραμάτων, για $k = 20$. Όλες οι τιμές της μετρικής GCP ήταν κάτω του 0.175 για τα δεδομένα `IPUMS`, σε κάθε πείραμα που διεξήχθη. Η κλασσική 20-ανωνυμοποίηση αυτού του συνόλου είχε απώλειες πληροφορίας $GCP = 0.09$.



(α) 1^η σειρά πειραμάτων (β) 2^η σειρά πειραμάτων

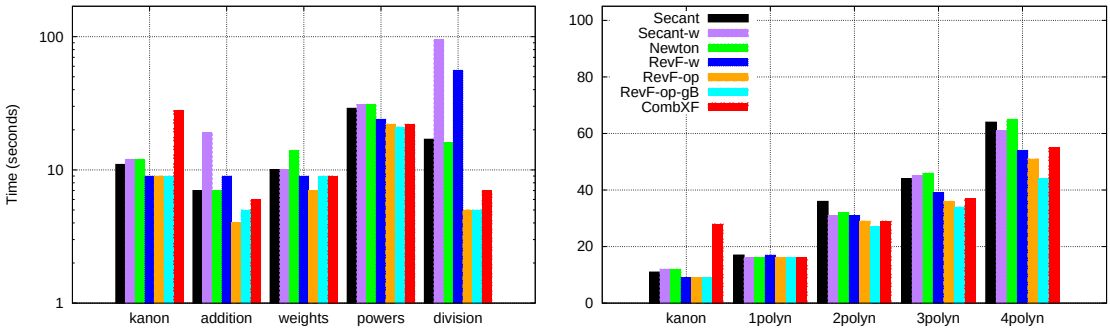
Σχήμα 7.5: Απόλυτο Σφάλμα των Ερωτημάτων Εύρους πάνω στα δεδομένα `IPUMS` ανωνυμοποιημένα από τον `CombXF`.



(α) 1^η σειρά πειραμάτων (β) 2^η σειρά πειραμάτων

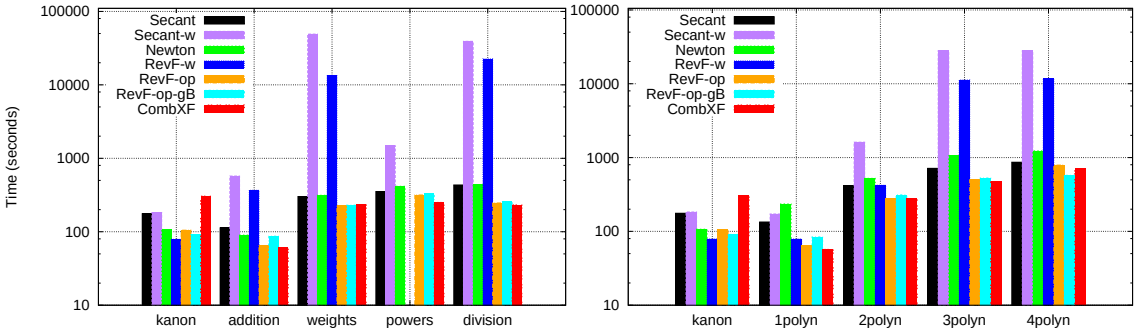
Σχήμα 7.6: Απόλυτο Σφάλμα των Ερωτημάτων Εύρους πάνω στα δεδομένα `ENERGY` ανωνυμοποιημένα από τον `CombXF`.

IPUMS. Όπως φαίνεται στο Σχήμα 7.1(α), η συνάρτηση f_7 (weights) ως γνώση επίθεσης επιτρέπει στους αλγόριθμους να επιτύχουν τα καλύτερα αποτελέσματα, με την GCP να κυμαίνεται από 0.000104 για τους αλγόριθμους `combXF` και `revF-op` έως και 0.076 για τον `Secant`. Η πρόσθεση των γνωρισμάτων χωρίς βάρη (f_6) επιβάλλει απώλεια πληροφορίας από 0.00277 για τους `combXF` και `Newton` έως 0.02121 για τον `revF-op`. Η τιμή της GCP για την συνάρτηση f_8 (powers) λαμβάνει τιμές από 0.012 για τους `combXF` και `revF-op` έως 0.07 για τους αλγορίθμους `Secant` και `Newton`. Τέλος, η συνάρτηση f_9 (*διισιον*) είναι χειρότερη από την k-ανωνυμία για ορισμένους αλγόριθμους, κυρίως όσους χρησιμοποιούν αριθμητική ανάλυση για να υπολογίσουν τα εύρη γενίκευσης των γνωρισμάτων. Ο λόγος για αυτό είναι ότι αυτοί οι αλγόριθμοι χρειάστηκαν να k-ανωνυμοποιήσουν αρκετές κλάσεις ώστε να ικανοποιηθεί η εγγύηση, δηλαδή να τις γενικεύσουν σε τέτοιο βαθμό ώστε οι εγγραφές να γίνουν πανομοιότυπες όπως στην κλασσική k-ανωνυμία. Εντούτοις, η ομαδοποίηση των εγγραφών βασίζεται στις τιμές των συναρτήσεων και δεν είναι βελτιστοποιημένη για την k-ανωνυμία. Έτσι, ορισμένα γνωρίσματα υπερ-γενικεύονται, κάτι που οδηγεί σε μεγαλύτερη απώλεια πληροφορίας από ότι χρειάζεται. Η μετρική GCP στην περίπτωση της f_9 (*διισιον*) κυμαίνεται από



(α) 1^η σειρά πειραμάτων (β) 2^η σειρά πειραμάτων

Σχήμα 7.7: Συγκριτική μελέτη: Χρόνοι εκτέλεσης για τα δεδομένα `IPUMS`, με k=20.



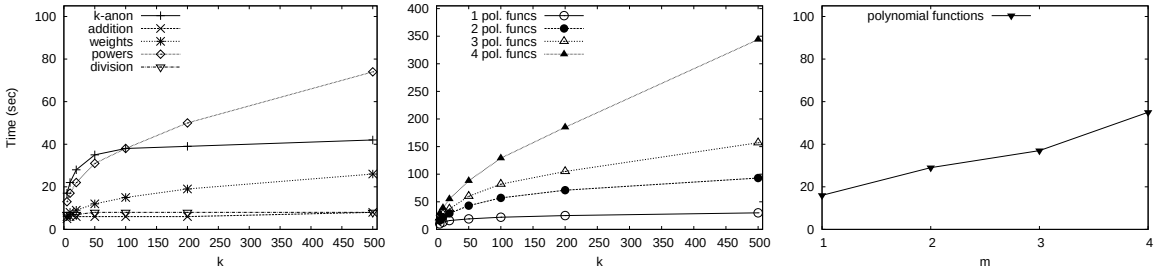
(α) 1^η σειρά πειραμάτων (β) 2^η σειρά πειραμάτων

Σχήμα 7.8: Συγκριτική μελέτη: Χρόνοι εκτέλεσης για τα δεδομένα `ENERGY`, με k=20.

0.04809 για τον `combXF` έως 0.1749 για τον `Secant`.

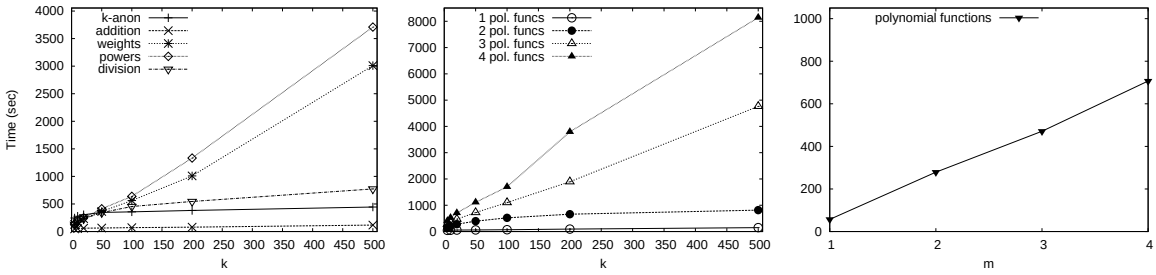
Στο Σχήμα 7.1(β) παρουσιάζονται τα αποτελέσματα της δεύτερης σειράς πειραμάτων για τα δεδομένα `IPUMS`. Όλοι οι προτεινόμενοι αλγόριθμοι επιδεικνύουν παρόμοια συμπεριφορά καθώς αυξάνεται το πλήθος των πολυωνυμικών συναρτήσεων που μοντελοποιούν την γνώση του επιτιθέμενου. Η απώλεια πληροφορίας αυξάνεται καθώς η γνώση του επιτιθέμενου γίνεται πιο ισχυρή.

Οι αλγόριθμοι που χρησιμοποιούν αριθμητική ανάλυση `Newton` και `Secant` δεν αποδίδουν καλά και ρίχνουν την ποιότητα των δεδομένων περισσότερο από την k-ανωνυμία για $m = 4$ συναρτήσεις. Η παραλλαγή `Secant-w` όμως καταφέρνει να ξεπεράσει και τους δύο και επιτυγχάνει μικρότερες απώλειες από την k-ανωνυμία και παρόμοιες με τον `revF-w`. Αυτοί οι αλγόριθμοι επιχειρούν να μοιράσουν τα εύρη γενίκευσης σε όλα τα γνωρίσματα των εγγραφών. Οι `Secant-w` και `revF-w` αποδίδουν καλύτερα διότι χρησιμοποιούν βάρη (w) τα οποία είναι ανάλογα του ρυθμού μεταβολής της συνάρτησης f_i ως προς το γνώρισμα x_j. Συνεπώς, γενικεύονται περισσότερο εκείνα τα γνωρίσματα τα οποία έχουν μεγαλύτερο αντίκτυπο στην τιμή της f_i και είναι πιθανότερο να επιτύχουν την ζητούμενη τιμή της f_i με μικρότερα εύρη γενίκευσης. Αντίθετα, οι αλγόριθμοι `revF-op`, `revF-op-gB` και `combXF` στοχεύουν στην γενίκευση ενός ή λίγων μόνο γνωρισμάτων με το μικρότερο κατά το δυνατόν εύρος γενίκευσης.



(α) (β) (γ)

Σχήμα 7.9: Χρόνος εκτέλεσης του `combXF` για τα δεδομένα `IPUMS`: επίδραση του k για την (α) 1^η και (β) 2^η σειρά πειραμάτων, (γ) επίδραση του m.



(α) (β) (γ)

Σχήμα 7.10: Χρόνος εκτέλεσης του `combXF` για τα δεδομένα `ENERGY`: επίδραση του k για την (α) 1^η και (β) 2^η σειρά πειραμάτων, (γ) επίδραση του m.

Οι δύο πρώτοι έχουν παρόμοια αποτελέσματα. Σε ορισμένες περιπτώσεις, ο `revF-op-gB` μπορεί να επιτύχει καλύτερα αποτελέσματα αποφεύγοντας τις γενικεύσεις που ξεπερνούν τα όρια της ελάχιστης και μέγιστης τιμής ενός γνωρίσματος μέσα σε μια κλάση ισοδυναμίας. Αυτό παρατηρείται στο πείραμα της συνάρτησης addition από την πρώτη σειρά πειραμάτων και στην 1poly στην δεύτερη σειρά του Σχήματος 7.1. Σε άλλες περιπτώσεις, η συνθήκη αυτή μπορεί να είναι περιοριστική και να μην επιτρέπει στον `revF-op-gB` να εξερευνήσει λύσεις όπου κάνοντας μια μόνο γενίκευση που ξεπερνά αυτά τα όρια για ένα γνώρισμα, μας γλιτώνει από την εφαρμογή πολλών μικρότερων γενικεύσεων. Αυτό συμβαίνει στα πειράματα των συναρτήσεων weights, powers, division και 2poly-4poly. Ο αλγόριθμος `combXF` αποδείχθηκε καλύτερος όλων στην διατήρηση της ποιότητας των αποτελεσμάτων και για τις δύο σειρές πειραμάτων. Για $m = 4$ πολυωνυμικές συναρτήσεις κατάφερε να επιτύχει 30% μικρότερο GCP από την k-ανωνυμία, ενώ για $m = 1$ συνάρτηση είναι 97.5% καλύτερος.

ENERGY. Η απώλεια πληροφορίας των ανωνυμοποιημένων δεδομένων `ENERGY` από όλους τους αλγόριθμους με $k = 20$ απεικονίζεται στο Σχήμα 7.2, Η κλασσική 20-ανωνυμία προκαλεί απώλειες με $GCP = 0.057$. Οι μέθοδοι με χρήση αριθμητικής ανάλυσης `Newton` και `Secant` δημιουργούν χειρότερα αποτελέσματα από την k-ανονψμιτψ στα περισσότερα πειράματα. Εντούτοις, όλοι οι υπόλοιποι αλγόριθμοι καταφέρνουν να διατηρήσουν καλύτερη ποιότητα στα τελικά δεδομένα. Ειδικά ο `combXF` επιτυγχάνει τις μικρότερες τιμές της μετρικής GCP

ξεπερνώντας όλους τους υπόλοιπους.

Τα αποτελέσματα της πρώτης σειράς πειραμάτων φαίνονται στο Σχήμα 7.2(α). Για την συνάρτηση πρόσθεσης με διαφορετικά βάρη (f_7) επιτυγχάνεται το μικρότερο $GCP = 0.001$ από τους αλγόριθμους `combXF` και `revF-op`, το οποίο είναι 98.2% καλύτερο από την k-ανωνυμία. Ο αλγόριθμος `revF-gB` προκαλεί επταπλάσια απώλεια από τον `combXF`, ενώ ο `Secant` φτάνει ως 0.11, μια τάξη μεγέθους περισσότερο από τον `combXF`. Η απλή πρόσθεση (f_6) έχει χειρότερα αποτελέσματα, εντούτοις ο `combXF` επιτυγχάνει απώλειες με GCP =0.01297 το οποίο είναι 77.2% μικρότερο σε σχέση με την k-ανωνυμία.

Στο Σχήμα 7.2(β) δίνουμε τα αποτελέσματα της συγκριτικής μελέτης για την δεύτερη σειρά πειραμάτων πάν στα δεδομένα `ENERGY`. Τόσο ο αλγόριθμος `combXF` όσο και ο `revF-op` αποδίδουν καλά, με τον πρώτο να είναι οριακά καλύτερος του δεύτερου. Ο `Secant` και ο `Newton` αποτυγχάνουν να ξεπεράσουν τη k-ανωνυμία. Παρόλα αυτά, ο `Secant-w` καταφέρνει να διατηρήσει ίδια επίπεδα χρησιμότητας των δεδομένων όπως ο `revF-w`. Ο αλγόριθμος `revF-op-gB` είναι ελαφρώς χειρότερος από τον `revF-op` καθώς επιτρέπει λιγότερη ευελιξία στην επιλογή των γενικεύσεων. Το μέγιστο όφελος που μπορεί να επιτευχθεί με την μέθοδό μας για $m = 1$ είναι μέσω του αλγορίθμου `combXF` ο οποίος δίνει GCP =0.0053, μία τάξη μεγέθους μικρότερο από την k-ανωνυμία. Για $m = 4$ πολυωνυμικές συναρτήσεις, μπορούμε να πετύχουμε μείωση 51.8% σε σχέση με την κλασσική εγγύηση.

Ο αλγόριθμος `combXF` που προτείνουμε ξεπέρασε όλους τους υπόλοιπους και μείωσε σημαντικά την απώλεια πληροφορίας σε σχέση με την κλασσική k-ανωνυμία σε όλα τα πειράματα και για όλα τα σύνολα δεδομένων που μελετήσαμε. Σημειώνεται ότι η ποιότητα της k-ανωνυμοποίησης εξαρτάται μόνο από την ποιότητα του αλγόριθμου συσταδοποίησης των εγγραφών που χρησιμοποιούμε. Διαφορετικοί αλγόριθμοι μπορεί να επιτύχουν καλύτερα αποτελέσματα για την k-ανωνυμία αν κάνουν καλύτερη ομαδοποίηση των εγγραφών σε κλάσεις ισοδυναμίας. Εντούτοις, μπορεί η μέθοδος συσταδοποίησής τους να ενσωματωθεί στους αλγόριθμους που προτείνουμε για την k_f^m-ανωνυμοποίηση, δηλαδή να εφαρμοστεί η ίδια συσταδοποίηση ως προς τις τιμές των συναρτήσεων, και να περιμένουμε αντίστοιχη βελτίωση στα αποτελέσματα της μεθόδου μας.

Αξιολόγηση του combXF

Στα υπόλοιπα πειράματα εστιάζουμε στον αλγόριθμο `combXF` ο οποίος κατάφερε να ξεπεράσει όλους τους υπόλοιπους διατηρώντας σημαντικά καλύτερη ποιότητα αποτελεσμάτων.

IPUMS. Το Σχήμα 7.3(α) δείχνει την συμπεριφορά του αλγορίθμου ως προς την μεταβολή της παραμέτρου k για την πρώτη σειρά πειραμάτων πάνω στα δεδομένα `IPUMS`. Με την αύξηση του k η απώλεια πληροφορίας μεγαλώνει υπογραμμικά, αλλά η κλασσική k-ανωνυμία διατηρεί λιγότερη ποιότητα σε σχέση με την μέθοδό μας για όλες τις συναρτήσεις. Για $k = 500$ η GCP της k-ανωνυμίας ξεπερνά το 0.17, ενώ για κάθε άλλη συνάρτηση είναι κάτω του 0.0785. Η διατήρηση πληροφορίας που επιτυγχάνεται για την συνάρτηση της πρόσθεσης με βάρη (f_7) είναι αξιοσημείωτη, καθώς για $k = 5$ δίνει GCP ίσο με $4.597 \cdot 10^{-05}$, το οποίο είναι 99.8% χαμηλότερο από την k-ανωνυμία. Για $k = 500$, φτάνει μόλις μέχρι την τιμή 0.000797, η οποία

είναι 99.5% χαμηλότερη από την k-ανωνυμία. Η πρόσθεση χωρίς βάρη (f_6) έχει απώλειες που κυμαίνονται μεταξύ 0.0011 για $k = 5$ και 0.0121 για $k = 500$. Η συνάρτηση f_8 έχει απώλειες μεταξύ 0.00827 και 0.01413 που είναι κατά μια τάξη μεγέθους χαμηλότερες από την κλασσική εγγύηση. Η διαίρεση (f_9) δίνει μικρότερη ποιότητα, αλλά παραμένει καλύτερη από την k-ανωνυμία, με GCP το οποίο ξεκινά από 0.04434 και φτάνει έως 0.05893 για $k = 500$, το οποίο είναι 66% καλύτερο από την k-ανωνυμία. Στο Σχήμα 7.3(β) παρατηρούμε παρόμοια συμπεριφορά ως προς την παράμετρο k. Η τιμή της μετρικής GCP αυξάνεται υπογραμμικά με το k, αλλά όχι τόσο σημαντικά όσο για την k-ανωνυμία. Για μεγαλύτερες τιμές του k το κέρδος στην ποιότητα των δεδομένων είναι μεγαλύτερο σε σύγκριση με την κλασσική εγγύηση, ακόμα και όταν θεωρούμε την γνώση 4 πολυωνυμικών συναρτήσεων οι οποίες συνολικά περιέχουν και τα 5 γνωρίσματα των δεδομένων. Η τιμή της GCP για $m = 4$ κυμαίνεται μεταξύ 0.0458 για $k = 5$ έως 0.0785 για $k = 500$, το οποίο είναι κατά 54.8% μικρότερο από την GCP της 500-ανωνυμίας. Το Σχήμα 7.3(γ) δείχνει μια σχεδόν γραμμική αύξηση της απώλειας πληροφορίας ως προς το πλήθος των συναρτήσεων που γνωρίζει ο επιτιθέμενος m. Για $k = 20$, η τιμή της GCP αυξάνεται από 0.00216 έως 0.063 καθώς μεγαλώνει το m. Το αντίστοιχο κόστος της 20-ανωνυμίας είναι 0.09, το οποίο είναι κατά προσέγγιση 1.4 φορές το κόστος της μεθόδου μας για $m = 4$ συναρτήσεις.

ENERGY. Στο Σχήμα 7.4 παρουσιάζεται η απώλεια πληροφορίας που εισάγει ο αλγόριθμος `combXF` για το σύνολο δεδομένων `ENERGY`. Σε αυτά τα πειράματα το κέρδος της προτεινόμενης μεθόδου από την κλασσική ανωνυμία γίνεται ακόμη πιο σημαντικό με την αύξηση του k. Το Σχήμα 7.4(α) δείχνει ότι το GCP όλων των συναρτήσεων αυξάνεται υπογραμμικά με το k. Η απώλεια πληροφορίας της k-ανωνυμίας κυμαίνεται μεταξύ 0.0217 και 0.2957 καθώς αυξάνεται το k. Ο κατακόρυφος άξονας είναι σε λογαριθμική κλίμακα, καθώς οι απώλειες των περισσότερων συναρτήσεων είναι αισθητά μικρότερες και δεν θα φαίνονταν σωστά στην απλή κλίμακα. Η συνάρτηση f_7 (weights) με $k = 5$ επιτρέπει 97.8% λιγότερη απώλεια πληροφορίας από την 5-ανωνυμία και έως 99.2% λιγότερο για $k = 500$. Οι υπόλοιπες συναρτήσεις χρειάζονται μεγαλύτερη αλλοίωση των δεδομένων, αλλά παραμένουν καλύτερες από την k-ανωνυμία κατά τουλάχιστον μια τάξη μεγέθους. Τα αποτελέσματα της δεύτερης σειράς πειραμάτων φαίνονται στο Σχήμα 7.4(β). Για $k = 5$ και $m = 1$ η μέθοδός μας είναι κατά 90.2% καλύτερη, ενώ για $m = 4$ είναι 43.2% καλύτερη από την 5-ανωνυμία. Καθώς η παράμετρος k αυξάνει, το κέρδος πληροφορίας του `combXF` είναι σημαντικό. Για $k = 500$ και $m = 4$, η προτεινόμενη μέθοδος επιτυγχάνει 68.7% λιγότερη απώλεια πληροφορίας σε σχέση με την 500-ανωνυμία, ενώ για $m = 1$ το αντίστοιχο κέρδος είναι 96.7%. Τέλος, στο Σχήμα 7.4(γ) δείχνει ότι για τα δεδομένα `ENERGY`, όχι μόνο η μετρική GCP είναι κατά μέσο όρο μικρότερη, αλλά επίσης αυξάνεται με μικρότερο ρυθμό καθώς αυξάνεται το πλήθος των συναρτήσεων που συνιστούν την γνώση του επιτιθέμενου. Για $k = 20$, η τιμή της GCP αυξάνεται από 0.0053 ως 0.0275 καθώς το m μεταβάλλεται από 1 σε 4 συναρτήσεις. Το αντίστοιχο κόστος της 20-ανωνυμίας είναι 0.057, το οποίο είναι διπλάσιο από την αντίστοιχη απώλεια για $m = 4$ συναρτήσεις.

Σφάλμα Ερωτημάτων Εύρους

Αξιολογήθηκε επίσης η ποιότητα της ανωνυμοποίησης μετρώντας το σφάλμα ερωτημάτων εύρους. Συγκεκριμένα, θεωρούμε ερωτήματα εύρους της ακόλουθης μορφής:

`SELECT count(*) FROM D WHERE (D.x`$_i \leq$ `L`$_1$`) AND (D.x`$_i >$ `L`$_2$`)`

Οι γενικεύσεις των τιμών των δεδομένων που εφαρμόστηκαν κατά την ανωνυμοποίηση εισάγουν σφάλματα στα αποτελέσματα της μέτρησης αυτών των ερωτημάτων εύρους. Εκτελέσαμε αυτά τα ερωτήματα τόσο στα αρχικά όσο και στα ανωνυμοποιημένα δεδομένα. Έστω μια αρχική τιμή v η οποία έχει γενικευθεί στο εύρος $[a, b]$, και έστω ότι το εύρος του ερωτήματος είναι $[L_1, L_2]$. Μια εγγραφή που περιέχει το $[a, b]$ καταμετράται ως $+1$ στο αποτέλεσμα του ερωτήματος αν $L_1 \leq a < b \leq L_2$. Η εγγραφή δεν προσμετρείται στο αποτέλεσμα αν $a > L_2$ ή $b < L_1$. Σε κάθε άλλη περίπτωση, υπάρχει μερική επικάλυψη όπου μόνο ένα μέρος p του διαστήματος $[a, b]$ περιλαμβάνεται στο εύρος του ερωτήματος. Τότε προσμετρείται $+\gamma$ στο αποτέλεσμα, όπου $\gamma = |p|/(b - a)$. Παραδείγματος χάριν, αν $a \leq L_1 \leq b \leq L_2$, τότε προστίθεται $\gamma = (L_1 - a)/(b - a)$ στο αποτέλεσμα του ερωτήματος.

Τα εύρη των ερωτημάτων κυμαίνονται από 0.1 έως 0.5 του πεδίου τιμών κάθε γνωρίσματος $\mathcal{I}_i$. Τα ερωτήματα εύρους εκτελέστηκαν για κάθε γνώρισμα χωριστά. Παραδείγματος χάριν, για τα εύρη μεγέθους $0.1|\mathcal{I}_i|$ ενός γνωρίσματος x_i, τα ερωτήματα που εκτελέστηκαν αφορούσαν τα διαστήματα: $[0, 0.1|\mathcal{I}_i|]$ $[0.1|\mathcal{I}_i|, 0.2|\mathcal{I}_i|]$, ... $[0.9|\mathcal{I}_i|, |\mathcal{I}_i|]$. Υπολογίστηκε ο μέσος όρος των αποτελεσμάτων, ανά μέγεθος εύρους ερωτήματος. Στην συνέχεια υπολογίστηκε ο μέσος όρος για όλα τα εύρη ενός γνωρίσματος. Στα Σχήματα 7.5 και 7.6 παρουσιάζεται το απόλυτο σφάλμα σε πλήθος εγγραφών, εκφρασμένο ως ποσοστό % του μεγέθους των δεδομένων $|D|$. Τα αποτελέσματα παρουσιάζονται σε λογαριθμική κλίμακα.

IPUMS. Όπως φαίνεται από το Σχήμα 7.5, η μέθοδός μας ξεπερνά την k-ανωνυμία, καθώς επιτυγχάνει μικρότερα σφάλματα. Ο μέσος όρος σφάλματος της 20-ανωνυμίας είναι 2.3% με 3% του συνόλου των εγγραφών, για όλα τα εύρη ερωτημάτων. Το όφελος της μεθόδου είναι μεγαλύτερο για τις συναρτήσεις f_7 (weights) και f_6 (addition) περίπου δυο τάξεις μεγέθους, αλλά μικρότερο για την f_9 (division) που κυμαίνεται από 0.88% έως $2.28\% \cdot |D|$. Για $m = 4$ πολυωνυμικές συναρτήσεις, το μέσο σφάλμα είναι περίπου τέσσερεις φορές μικρότερο από την 20-ανωνυμία, ενώ για $m = 1$ είναι οκτώ φορές μικρότερο αντιστοίχως.

ENERGY. Το μέγεθος του συνόλου δεδομένων `ENERGY` είναι μεγαλύτερο $|D|$=2,075,259 εγγραφές. Καθώς φαίνεται στο Σχήμα 7.6(α) το απόλυτο σφάλμα είναι κάτω του 1% του μεγέθους των δεδομένων σε όλα τα ερωτήματα, για κάθε εύρος και για κάθε σενάριο γνώσης του επιτιθέμενου. Το σφάλμα για την συνάρτηση f_8 (powers) είναι από 1 έως 2 τάξεις μεγέθους μικρότερο από το σφάλμα της k-ανωνυμίας. Η πρόσθεση των γνωρισμάτων (f_6) και η διαίρεση (f_9) εισάγουν υψηλότερα σφάλματα, εντούτοις παραμένουν κατά 60.4% καλύτερα από την κλασσική 20-ανωνυμία. Από το Σχήμα 7.6(β) προκύπτει ότι το σφάλμα αυξάνει καθώς το πλήθος των συναρτήσεων επίθεσης αυξάνονται. Για $m = 1$, το σφάλμα είναι τουλάχιστον δυο τάξεις μεγέθους μικρότερο, ενώ για $m = 2$ είναι 97.4% καλύτερο από την k-ανωνυμία αντιστοίχως. Ακόμη και για $m = 4$, το σφάλμα είναι μειωμένο κατά 39.7% - 64.2% σε σχέση με την k-ανωνυμία.

7.4.7 Χρόνος Εκτέλεσης

Σύγκριση των Αλγορίθμων

Το υπολογιστικό κόστος δεν αποτελεί το σημαντικότερο σημείο αξιολόγησης για την ανωνυμοποίηση, καθώς η διαδικασία αυτή συνήθως εκτελείται μία φορά πριν τη δημοσίευση των δεδομένων. Εντούτοις είναι σημαντικό οι χρόνοι εκτέλεσης να είναι ρεαλιστικοί και να μην είναι απαγορευτικοί για πραγματικά δεδομένα.

IPUMS. Για $k = 20$ όλα τα πειράματα που εκτελέστηκαν πάνω στο σύνολο δεδομένων `IPUMS` ολοκληρώθηκαν σε λιγότερο από 95 δευτερόλεπτα. Το υπολογιστικό κόστος για την πρώτη σειρά πειραμάτων φαίνεται στο Σχήμα 7.7(α). Στα περισσότερα πειράματα ο αλγόριθμος `combXF` όχι μόνο παράγει τα καλύτερα ανώνυμα αποτελέσματα, αλλά το πραγματοποιεί σε σύντομο χρόνο εκτέλεσης. Η συνάρτηση f_8 που χρειάζεται τον υπολογισμό δυνάμεων από 2ης έως και 5ης τάξης προκαλεί μεγαλύτερο υπολογιστικό κόστος, διότι οι υπολογισμοί αυτοί χρειάζονται περισσότερο χρόνο. Στο Σχήμα 7.7(β) ο χρόνος εκτέλεσης αυξάνεται γραμμικά για όλους τους αλγόριθμους, καθώς προστίθενται περισσότερες συναρτήσεις στην γνώση του επιτιθέμενου, όπως είναι αναμενόμενο. Ο `combXF` παραμένει μια από τις ταχύτερες λύσεις, ενώ οι `revF-op` και `revF-op-gB` τον ξεπερνούν οριακά σε κάποιες περιπτώσεις.

ENERGY. Τα δεδομένα `ENERGY` έχουν δεκαπλάσιο πλήθος εγγραφών και η ανωνυμοποίησή τους χρειάζεται περισσότερο χρόνο. Όπως φαίνεται από τα Σχήματα 7.8(α) και (β), οι αλγόριθμοι `Secant-w` και `revF-w` είναι υπολογιστικά ακριβοί και σε ορισμένα πειράματα χρειάστηκαν ώρες για αν τερματίσουν. Όλοι οι υπόλοιποι αλγόριθμοι τερματίζουν μέσα σε 7.25 λεπτά, για $k = 20$. Γενικά για συναρτήσεις που δεν απαιτούν την ύψωση των γνωρισμάτων σε δυνάμεις με μεγάλους εκθέτες, οι αλγόριθμοι είτε έχουν παρόμοια απόδοση είτε είναι ταχύτεροι από την k-ανωνυμία. Όταν απαιτούνται πολύπλοκοι υπολογισμοί σε κάθε βήμα, ο χρόνος εκτέλεσης είναι μεγαλύτερος. Εντούτοις, η επιβάρυνση σε υπολογιστικό κόστος αντισταθμίζεται από το κέρδος στην βελτιωμένη ποιότητα των τελικών δεδομένων.

Παρατηρούμε από το σύνολο των πειραμάτων ότι οι αλγόριθμοι που προσπαθούν να διανείμουν τα εύρη των γενικεύσεων ανάμεσα σε όλα τα γνωρίσματα που είναι ορίσματα μιας συνάρτησης, όπως είναι οι `Newton`, `Secant`, `Secant-w` και `revF-w`, χρειάζονται περισσότερο χρόνο καθώς απαιτούν περισσότερες επαναλήψεις σε σύγκριση με τους αλγόριθμους που προσπαθούν να επιλέξουν την καλύτερη γενίκευση ενός γνωρίσματος στο ελάχιστο δυνατό εύρος, όπως είναι οι `revF-op`, `revF-op-gB` και `combXF`. Το μειονέκτημα του `combXF` είναι ότι απαιτεί τον υπολογισμό του πίνακα Δ. Αυτό το επιπρόσθετο υπολογιστικό κόστος, το οποίο είναι ασήμαντο στις περισσότερες περιπτώσεις, αντισταθμίζεται από την αξιόλογη μείωση της απώλειας πληροφορίας στα ανωνυμοποιημένα δεδομένα.

Υπολογιστικό κόστος του combXF

IPUMS. Στα Σχήματα 7.9 (α) και (β) παρατηρούμε ότι ο αλγόριθμος `combXF` κλιμακώνει υπογραμμικά με την παράμετρο k, για το σύνολο δεδομένων `IPUMS`. Όλα τα πειράματα ολοκληρώθηκαν σε λιγότερο από 6 λεπτά. Η συνάρτηση f_8 (powers) δίνει τους χειρότερους χρόνους ανάμεσα στην πρώτη σειρά των πειραμάτων, για τους λόγους που εξηγήθηκαν παραπάνω. Στο

Σχήμα 7.9(γ) φαίνεται ότι το υπολογιστικό κόστος μεγαλώνει σχεδόν γραμμικά με το πλήθος των συναρτήσεων m.

ENERGY. Τα Σχήματα 7.10 (α) και (β) δείχνουν ότι ο `combXF` κλιμακώνει σχεδόν γραμμικά με την παράμετρο k για τις συναρτήσεις weights και powers, ενώ κλιμακώνει υπογραμμικά για όλες τις υπόλοιπες, για τα δεδομένα `ENERGY`. Στο Σχήμα 7.10 (γ) παρατηρούμε ότι ο χρόνος εκτέλεσης αυξάνεται γραμμικά καθώς προσθέτουμε περισσότερες συναρτήσεις στην γνώση του επιτιθέμενου.

7.5 Συμπεράσματα

Το κεφάλαιο αυτό είχε ως βασικό αντικείμενο μελέτης την ανωνυμοποίηση αριθμητικών δεδομένων από επιτιθέμενους οι οποίοι δεν γνωρίζουν τις ακριβείς τιμές των γνωρισμάτων, ενώ έχουν αποκτήσει την γνώση τιμών από μία ή περισσότερες συνεχείς συναρτήσεις που ορίζονται πάνω στα γνωρίσματα των δεδομένων. Προτάθηκε η εγγύηση της k_f^m-ανωνυμίας για την πρόληψη της αποκάλυψης ταυτότητας από επιτιθέμενους που κατέχουν την προαναφερθείσα γνώση. Αναπτύχθηκαν πέντε νέοι αλγόριθμοι ανωνυμοποίησης για την ικανοποίηση της προτεινόμενης εγγύησης με μικρή απώλεια πληροφορίας. Η πειραματική αξιολόγηση της προτεινόμενης μεθόδου πάνω σε σύνολα πραγματικών δεδομένων έδειξε ότι μπορεί να διασφαλίσει καλύτερη ποιότητα των ανωνυμοποιημένων δεδομένων σε σχέση με την κλασσική k-ανωνυμία. Η σύγκριση των διαφορετικών προτεινόμενων αλγόριθμων μεταξύ τους αναδεικνύει ως καλύτερη επιλογή τον αλγόριθμο `combXF`, ο οποίος συνδυάζει ένα σχετικά περιορισμένο υπολογιστικό κόστος με την σημαντικότερη μείωση της απώλειας πληροφορίας από όλες τις προτεινόμενες εναλλακτικές.

Πιθανές κατευθύνσεις για μελλοντική έρευνα είναι η επέκταση της μεθόδου για την ικανοποίηση αυστηρότερων κριτηρίων ιδιωτικότητας, όπως οι εγγυήσεις της l-διαφορετικότητας και της t-εγγύτητας για αποτελεσματικότερη πρόληψη από επιθέσεις αποκάλυψης γνωρίσματος.

Κεφάλαιο 8

Σύνοψη και Μελλοντικές Επεκτάσεις

Η παρούσα διδακτορική διατριβή επικεντρώνεται σε προβλήματα ανωνυμοποίησης δεδομένων τα οποία δεν υπακούν στο κλασικό σχεσιακό σχήμα. Οι εγγραφές στα σύνολα δεδομένων που μελετάμε μπορεί να έχουν δομή δένδρων, γράφων ή να είναι αδόμητα σύνολα τιμών. Τέτοια δεδομένα συναντώνται σε πληθώρα εφαρμογών και είναι ιδιαίτερα διαδεδομένα στο διαδίκτυο. Στο υπόλοιπο αυτού του κεφαλαίου συνοψίζονται με λεπτομέρεια οι συνεισφορές της διατριβής και προτείνονται ορισμένα ενδιαφέροντα θέματα για μελλοντική έρευνα.

8.1 Σύνοψη

Αρχικά μελετήθηκε το πρόβλημα της προστασίας της ιδιωτικότητας σε δημοσιεύσεις δεδομένων με δενδρική δομή, όπως είναι τα XML δεδομένα αλλά και οι σχεσιακές βάσεις με πολλούς πίνακες που συνδέονται μεταξύ τους με ξένα κλειδιά. Έγινε ανάλυση των πιθανών μοντέλων επίθεσης και διατύπωση εγγυήσεων ιδιωτικότητας για την πρόληψη από επιθέσεις αποκάλυψης ταυτότητας σε δενδρικά δεδομένα. Συγκεκριμένα διατυπώθηκε μια παραλλαγή της εγγύησης της κ-ανωνυμίας, η $k^{(m,n)}$-ανωνυμία για δενδρικά και XML δεδομένα. Η ιδιαιτερότητα σε αυτό το πρόβλημα είναι ότι η ίδια η δομή των εγγραφών μπορεί να προδίδει πληροφορίες, λ.χ. τη συσχέτιση κάποιων γνωρισμάτων. Συνεπώς, υπάρχει ο κίνδυνος να λειτουργήσει και η ίδια η δομή ως ψευδο-αναγνωριστικό. Υπάρχοντες αλγόριθμοι σχεσιακών δεδομένων δεν μπορούν να εφαρμοστούν σε αυτό το σενάριο. Βασική συνεισφορά αυτής της εργασίας είναι: α) η ανάλυση του τρόπου με τον οποίο η ίδια η δομή των εγγραφών μπορεί να λειτουργεί ως ψευδο-αναγνωριστικό, β) η παραδοχή ότι οποιοδήποτε τμήμα της δενδρικής εγγραφής είναι εξίσου ευαίσθητο αλλά και εξίσου πιθανό να λειτουργήσει ως ψευδο-αναγνωριστικό, γ) η εισαγωγή μιας νέας πράξης ανακωδικοποίησης, της δομικής αποσυσχέτισης, η οποία αποκρύπτει την δομική σχέση μεταξύ δυο τιμών και δ) ο ορισμός της νέας μετρικής RPD για την ποσοτικοποίηση της απώλειας πληροφορίας τόσο ως προς τις γενικεύσεις τιμών των γνωρισμάτων όσο και ως προς τις δομικές απλοποιήσεις των εγγραφών. Ο προτεινόμενος αλγόριθμος ανωνυμοποίησης ACS και η άπληστη παραλλαγή του GCS που υλοποιήθηκαν στα πλαίσια αυτής της

διατριβής εστιάζουν στα ιδιαίτερα χαρακτηριστικά του τύπου αυτών των δεδομένων. Η πειραματική ανάλυση σε πραγματικά δεδομένα έδειξε ότι ο προτεινόμενος αλγόριθμος ανωνυμοποιεί αποτελεσματικά τα σύνολα δενδρικών δεδομένων εισάγοντας μικρότερη απώλεια πληροφορίας σε σχέση με εναλλακτικές που δεν χρησιμοποιούν δομικές αποσυσχετίσεις.

Στη συνέχεια μελετήθηκε το πρόβλημα της προστασίας της ιδιωτικότητας σε δημοσιεύσεις δεδομένων με δομή γράφου. Η δομή των δεδομένων αυτών μπορεί επίσης να λειτουργήσει ως ψευδο-αναγνωριστικό, αντίστοιχα με τα δενδρικά, αλλά ο βαθμός δυσκολίας αυξάνεται καθώς λαμβάνουμε υπόψη τις ετικέτες ακμών και κορυφών καθώς επίσης την κατεύθυνση των ακμών που δεν είναι απαραίτητα η ίδια κατά μήκος ενός μονοπατιού. Δόθηκε έμφαση στα διασυνδεδεμένα δεδομένα και ειδικά τα δεδομένα RDF λόγω της ευρείας διάδοσής τους στον Παγκόσμιο Ιστό. Στα πλαίσια της διατριβής, μοντελοποιήθηκαν τα σενάρια επίθεσης κατά της ιδιωτικότητας δεδομένων σε μορφή RDF και προτάθηκαν νέες πράξεις ανακωδικοποίησης συμβατές με το μοντέλο των δεδομένων. Επεκτάθηκε η εγγύηση ιδιωτικότητας των δενδρικών δεδομένων για να καλύψει τις ιδιαιτερότητες της δομής των RDF γράφων και προτάθηκε ένας νέος αλγόριθμος ανωνυμοποίησης που ικανοποιεί την εγγύηση περιορίζοντας την απώλεια πληροφορίας.

Οι συλλογές εγγραφών όπου καθεμία είναι ένα αδόμητο σύνολο τιμών πρόκειται για ένα ακόμη ενδιαφέρον παράδειγμα δεδομένων που δεν υπακούν σε αυστηρό σχεσιακό σχήμα. Τέτοια δεδομένα παρουσιάζονται συχνά σε πολλές εφαρμογές: οι μετρήσεις αισθητήρων, οι καταγραφές ανθρώπινης παρατήρησης, οι δείκτες ιατρικών εξετάσεων όπως οι μετρήσεις σφυγμών και πίεσης του αίματος, τα οικονομικά δεδομένα όπως σύνολα πληρωμών ή αγορών από πιστωτικές κάρτες, είναι όλα παραδείγματα τέτοιων δεδομένων και η δημοσιοποίησή τους χωρίς την κατάλληλη επεξεργασία μπορεί να πλήξει την ιδιωτικότητα των χρηστών. Αντίθετα η εφαρμογή πολύ αυστηρών εγγυήσεων μπορεί να καταστήσει τα δεδομένα άχρηστα για ανάλυση και στατιστική μελέτη. Το πρόβλημα της ανωνυμοποίησης τέτοιων συλλογών έχει μελετηθεί στην βιβλιογραφία για τις περιπτώσεις κατηγορικών τιμών και οι προτεινόμενες λύσεις χρησιμοποιούν μια προκαθορισμένη ιεραρχία γενίκευσης τιμών. Στα πλαίσια της διατριβής, αναπτύχθηκε ένας νέος αλγόριθμος ολικής ανακωδικοποίησης, ο οποίος δεν χρησιμοποιεί προκαθορισμένες ιεραρχίες για την ανωνυμοποίηση συλλογών δεδομένων όπου οι εγγραφές είναι σύνολα από συνεχείς τιμές. Η πειραματική ανάλυση έδειξε ότι ο προτεινόμενος αλγόριθμος ACD καταφέρνει να ξεπεράσει τις υπάρχουσες μεθόδους αλλοιώνοντας λιγότερο τις τιμές των δεδομένων.

Παράλληλα με την μελέτη για την προστασία διαφορετικών τύπων μη-σχεσιακών δεδομένων, μελετήθηκαν και διαφορετικά μοντέλα επίθεσης, όπως το σενάριο όπου συναθροιστική γνώση διαφόρων αριθμητικών τιμών των γνωρισμάτων κάποιας εγγραφής δεδομένων, όπως το άθροισμα ή ο μέσος όρος, μπορεί να χρησιμοποιηθεί για να γίνει ταυτοποίησή της. Συχνά τα δεδομένα που δημοσιεύει μια εταιρία ή ένας οργανισμός περιέχουν τόσο λεπτομερείς τιμές ώστε να είναι απίθανο ένας κακόβουλος επιτιθέμενος να κατέχει ακριβή μερική γνώση μιας εγγραφής. Εντούτοις, θα μπορούσε να έχει πιο αφηρημένη ή συναθροιστική γνώση σχετικά με τα πεδία της εγγραφής. Τέτοια παραδείγματα προκύπτουν από πολλές εφαρμογές, ένα χαρακτηριστικό παράδειγμα είναι τα φορολογικά δεδομένα. Κάθε εγγραφή περιέχει πολυάριθμα

πεδία τα οποία καταγράφουν ένα μεγάλο εύρος οικονομικών δραστηριοτήτων των φορολογούμενων σε πολύ λεπτομερές επίπεδο. Μετά την δημοσίευση τέτοιων δεδομένων, αναμένουμε οι περισσότερες επιθέσεις να προέρχονται από επιτιθέμενους που αναγνωρίζουν τις εγγραφές βασιζόμενοι σε πιο γενική συναθροιστική γνώση. Η ανωνυμοποίηση τέτοιων δεδομένων χρησιμοποιώντας παραδοσιακές εγγυήσεις ιδιωτικότητας θα μπορούσε να εγγυηθεί την προστασία των εγγραφών, αλλά θα αλλοίωνε περισσότερο από ότι χρειάζεται τις τιμές τους. Στην πραγματικότητα χρειάζεται να δημιουργηθούν κλάσεις ισοδυναμίας μόνο ως προς την πιο γενική συναθροιστική γνώση του επιτιθέμενου. Στα πλαίσια της διδακτορικής διατριβής, μελετήθηκε το πρόβλημα, μοντελοποιήθηκε η συναθροιστική γνώση του επιτιθέμενου και προτάθηκε ένας αποδοτικός αλγόριθμος, ο οποίος ελαττώνει σημαντικά την απώλεια πληροφορίας που προκύπτει από τον μετασχηματισμό των ψευδο-αναγνωριστικών γνωρισμάτων, σε σχέση με την κλασσική k-ανωνυμία.

Ως επέκταση του παραπάνω προβλήματος, μελετήθηκαν πιο πολύπλοκα σενάρια επίθεσης τα οποία δεν περιλαμβάνουν την γνώση μόνο μίας συναθροιστικής συνάρτησης που ορίζεται πάνω στις τιμές όλων των γνωρισμάτων μιας εγγραφής, αλλά και την γνώση πολλαπλών συναρτήσεων που καθεμία ορίζεται πάνω σε οποιοδήποτε υποσύνολο των πραγματικών τιμών των γνωρισμάτων μιας εγγραφής. Ένα τέτοιο παράδειγμα είναι όταν ο επιτιθέμενος γνωρίζει τα καθαρά κέρδη (έσοδα - απώλειες) από επενδύσεις κεφαλαίων, τον μέσο όρο πληρωμών σε πιστωτικές κάρτες το μήνα, και το συνολικό εισόδημα από μισθό και ενοίκια ενός φορολογούμενου. Η γνώση αυτή μοντελοποιείται ως ένα σύστημα εξισώσεων με μεταβλητές τα πεδία της εγγραφής. Προτείνονται διάφοροι εναλλακτικοί αλγόριθμοι οι οποίοι ανωνυμοποιούν τα δεδομένα απέναντι σε τέτοιες επιθέσεις και εισάγουν μικρότερη απώλεια πληροφορίας. Τέλος, οι αλγόριθμοι αυτοί συγκρίνονται και αξιολογούνται πειραματικά πάνω σε πραγματικά σύνολα δεδομένων. Η πειραματική αξιολόγηση της προτεινόμενης μεθόδου δείχνει ότι μπορεί να διασφαλίσει καλύτερη ποιότητα των ανωνυμοποιημένων δεδομένων σε σχέση με την κλασσική k-ανωνυμία. Η σύγκριση των διαφορετικών προτεινόμενων αλγόριθμων μεταξύ τους αναδεικνύει ως καλύτερη επιλογή τον αλγόριθμο combXF, ο οποίος συνδυάζει περιορισμένο υπολογιστικό κόστος με σημαντική μείωση της απώλειας πληροφορίας.

Όλες οι προηγούμενες εργασίες κατατέθηκαν για δημοσίευση σε διεθνή περιοδικά και συνέδρια (τρεις ήδη δημοσιευμένες και δύο υπό υποβολή). Επιπλέον, μέσω των παραπάνω εργασιών, προέκυψαν περαιτέρω ερευνητικά προβλήματα, τα περιγράφονται στην επόμενη παράγραφο και θα μπορούσαν να αποτελέσουν αντικείμενο μελλοντικής δουλειάς.

8.2 Μελλοντικές Εργασίες

Οι επεκτάσεις της υπάρχουσας εργασίας αφορούν τις εξής κατευθύνσεις:

1. Η επέκταση των εγγυήσεων της ανωνυμίας δενδρικών, διασυνδεδεμένων και αδόμητων δεδομένων και για προστασία από αποκάλυψη ευαίσθητου γνωρίσματος (attribute disclosure) πέραν της αποκάλυψης ταυτότητας (identity disclosure), κατά αναλογία με τις εγγυήσεις της l-διαφορετικότητας και της t-εγγύτητας, προσφέροντας παράλληλα αποδοτικούς αλγόριθμους για την εφαρμογή τους.

2. Η προσαρμογή των αλγορίθμων προστασίας ιδιωτικότητας δενδρικών δεδομένων που προτείναμε και σε πολυ-σχεσιακές βάσεις δεδομένων με συναρτησιακές εξαρτήσεις, δίνοντας ιδιαίτερη προσοχή στην υλοποίηση δομικών αποσυσχετίσεων παρουσία ενός σχήματος με ξένα κλειδιά, πιθανώς με τη χρήση επιπλέον βοηθητικών πινάκων, ώστε να τηρούνται οι περιορισμοί της βάσης.

3. Η μελέτη του προβλήματος της ανωνυμοποίησης δεδομένων των οποίων τα γνωρίσματα δεν είναι ισότιμα ως προς το βαθμό παρατηρισιμότητας ή/και ευαισθησίας. Έτσι ορισμένα ψευδο-αναγνωριστικά μπορούν να είναι πιο εύκολα παρατηρίσιμα από άλλα ή αντίστοιχα ορισμένες τιμές γνωρισμάτων ή δομικές πληροφορίες να χρειάζονται περισσότερο αυστηρή προστασία από άλλες.

Βιβλιογραφία

[1] Australian Privacy Act. `www.austlii.edu.au/au/legis/cth/consol_act/pa1988108`, χ.χ.

[2] Canadian Privacy Act. `laws-lois.justice.gc.ca/eng/acts/P-21/`, χ.χ.

[3] Data Protection Act 1998, UK. `www.legislation.gov.uk/ukpga/1998/29/contents`, χ.χ.

[4] GR Law. `www.dpa.gr/portal/page?_pageid=33,43560&_dad=portal`, χ.χ.

[5] HIPAA act, US. `http://health.state.tn.us/hipaa/`, χ.χ.

[6] Individual household electric power consumption dataset. https://archive.ics.uci.edu/ml/datasets/Individual +household+electric+power+consumption, χ.χ.

[7] Ipums census dataset. https://archive.ics.uci.edu/ml/datasets/IPUMS+Census+Database, χ.χ.

[8] TPC-H Homepage. http://www.tpc.org/tpch/, χ.χ.

[9] Uci repository. http://archive.ics.uci.edu/ml/datasets.html, χ.χ.

[10] Uci repository us census data 1990 data set. http://archive.ics.uci.edu/ml/datasets/US+Census+Data+%281990%29, χ.χ.

[11] Utd anonymization toolbox. http://cs.utdallas.edu/dspl/cgi-bin/toolbox/, χ.χ.

[12] W3C RDF 1.1 Concepts. `www.w3.org/TR/rdf11-concepts/`, χ.χ.

[13] W3C RDF 1.1 Schema. `www.w3.org/TR/rdf-schema/`, χ.χ.

[14] G. Aggarwal, T. Feder, K. Kenthapadi, R. Motwani, R. Panigrahy, D. Thomas και A. Zhu. Approximation Algorithms for k-Anonymity. *J. of Privacy Technology*, 2005.

[15] Gagan Aggarwal, Tomas Feder, Krishnaram Kenthapadi, Samir Khuller, Rina Panigrahy, Dilys Thomas και An Zhu. Achieving Anonymity via Clustering. Στο *PODS*, σελίδες 153–162, 2006.

[16] M. Barbaro και T. Zeller. A face is exposed for AOL searcher no. 4417749. New York Times, 2006.

[17] Roberto J. Bayardo και Rakesh Agrawal. Data Privacy through Optimal k-Anonymization. Στο *ICDE*, σελίδες 217–228, 2005.

[18] Jianneng Cao και Panagiotis Karras. Publishing microdata with a robust privacy guarantee. *PVLDB*, 5(11):1388–1399, 2012.

[19] R. Chaytor και K. Wang. Small-domain randomization: Same privacy more utility. Στο *VLDB*, 2010.

[20] James Cheng, Ada Wai chee Fu και Jia Liu. K-isomorphism: privacy preserving network publication against structural attacks. Στο *SIGMOD*, 2010.

[21] Rui Chen, Noman Mohammed, Benjamin C. M. Fung, Bipin C. Desai και Li Xiong. Publishing set-valued data via differential privacy. *PVLDB*, 4(11):1087–1098, 2011.

[22] Rui Chen, Mohammed Noman, Benjamin C.M. Fung, Bipin C. Desai και Li Xiong. Publishing Set-Valued Data via Differential Privacy. *PVLDB*, 2011.

[23] Chris Clifton και Tamir Tassa. On syntactic anonymity and differential privacy. Στο *PRIVDB*, 2013.

[24] Graham Cormode. Personal privacy vs population privacy: learning to attack anonymization. Στο *SIGKDD*, σελίδες 1253–1261, 2011.

[25] Graham Cormode, Cecilia Procopiuc, Entong Shen, Divesh Srivastava και Ting Yu. Empirical privacy and empirical utility of anonymized data. Στο *PRIVDB*, σελίδες 77–82, 2013.

[26] Josep Domingo-Ferrer, Michal Sramka και Rolando Trujillo-Rasua. Privacy-preserving publication of trajectories using microaggregation. Στο *SPRINGL*, σελίδες 26–33, 2010.

[27] C. Dwork, F. McSherry, K. Nissim και A. Smith. Calibrating noise to sensitivity in private data analysis. *Theory of Cryptography*, 2006.

[28] C. Dwork, M. Naor, O. Reingold, G.N. Rothblum και S. Vadhan. On the complexity of differentially private data release: Efficient algorithms and hardness results. Στο *STOC*, σελίδες 381–390, 2009.

[29] Cynthia Dwork. Differential privacy. Στο *ICALP (2)*, σελίδες 1–12, 2006.

[30] Gabriel Ghinita, Panos Kalnis και Yufei Tao. Anonymous publication of sensitive transactional data. *TKDE*, 23(2):161–174, 2011.

[31] Gabriel Ghinita, Panagiotis Karras, Panos Kalnis και Nikos Mamoulis. Fast Data Anonymization with Low Information Loss. Στο *VLDB*, 2007.

[32] Gabriel Ghinita, Yufei Tao και Panos Kalnis. On the anonymization of sparse high-dimensional data. Στο *In Proceedings of the IEEE 24th International Conference on Data Engineering (ICDE)*, 2008.

[33] Gabriel Ghinita, Yufei Tao και Panos Kalnis. On the Anonymization of Sparse High-Dimensional Data. Στο *ICDE*, 2008.

[34] Aris Gkoulalas-Divanis και Grigorios Loukides. Utility-guided clustering-based transaction data anonymization. *TDP*, 5(1):223–251, 2012.

[35] O. Gkountouna, K. Lepenioti και M. Terrovitis. Privacy against aggregate knowledge attacks. Στο *ICDEW*, σελίδες 99–103, 2013.

[36] Olga Gkountouna. A Survey on Privacy Preservation Methods. *NTUA, Technical Report*, 2011.

[37] Olga Gkountouna, Sotiris Angeli, Athanasios Zigomitros, Manolis Terrovitis και Yannis Vassiliou. k^m-anonymity for continuous data using dynamic hierarchies. Στο *PSD*, σελίδες 156–169. Springer, 2014.

[38] Olga Gkountouna και Manolis Terrovitis. Anonymizing collections of tree-structured data. *TKDE*, 27(8):2034–2048, 2015.

[39] Jiawei Han και Yongjian Fu. Discovery of multiple-level association rules from large databases. Στο *VLDB*, 1995.

[40] Jiawei Han και Yongjian Fu. Mining multiple-level association rules in large databases. *TKDE*, 1999.

[41] Jiawei Han, Jian Pei και Yiwen Yin. Mining frequent patterns without candidate generation. Στο *SIGMOD*, σελίδες 1–12, 2000.

[42] Yeye He και Jeffrey F. Naughton. Anonymization of set-valued data via top-down, local generalization. *PVLDB*, 2009.

[43] Daniel Kifer. Attacks on privacy and deFinetti's theorem. Στο *SIGMOD*, σελίδες 127–138, 2009.

[44] Daniel Kifer και Ashwin Machanavajjhala. No free lunch in data privacy. Στο *SIGMOD*, σελίδες 193–204, 2011.

[45] Kristen LeFevre, David J. DeWitt και Raghu Ramakrishnan. Incognito: Efficient Full-domain k-Anonymity. Στο *SIGMOD*, 2005.

[46] Kristen LeFevre, David J. DeWitt και Raghu Ramakrishnan. Mondrian Multidimensional k-Anonymity. Στο *ICDE*, 2006.

[47] Chao Li και Gerome Miklau. An adaptive mechanism for accurate query answering under differential privacy. *PVLDB*, 5(6), 2012.

[48] Jiuyong Li, Raymond Chi Wing Wong, Ada Wai Chee Fu και Jian Pei. Anonymization by local recoding in data with attribute hierarchical taxonomies. *TKDE*, 2008.

[49] Ninghui Li, Tiancheng Li και Suresh Venkatasubramanian. t-Closeness: Privacy Beyond k-Anonymity and l-Diversity. Στο *ICDE*, σελίδες 106–115, 2007.

[50] Ninghui Li, Tiancheng Li και Suresh Venkatasubramanian. Closeness: A new privacy measure for data publishing. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 2009.

[51] Junqiang Liu και Ke Wang. On optimal anonymization for l^+-diversity. Στο *ICDE*, σελίδες 213–224, 2010.

[52] Ashwin Machanavajjhala, Johannes Gehrke, Daniel Kifer και Muthuramakrishnan Venkitasubramaniam. l-Diversity: Privacy Beyond k-Anonymity. Στο *ICDE*, 2006.

[53] Adam Meyerson και Ryan Williams. On the Complexity of Optimal k-anonymity. Στο *PODS*, σελίδες 223–228, 2004.

[54] Anna Monreale, Roberto Trasarti, Dino Pedreschi, Chiara Renso και Vania Bogorny. C-safety: a framework for the anonymization of semantic trajectories. *TDP*, 4(2):73–101, 2011.

[55] Ramon E. Moore. *Interval Analysis (Automatic Computation S.)*. Prentice Hall, 1967.

[56] M.E. Nergiz, C. Clifton και A.E. Nergiz. Multirelational k-anonymity. Στο *ICDE*, 2007.

[57] Mehmet Ercan Nergiz, Maurizio Atzori και Chris Clifton. Hiding the presence of individuals from shared databases. Στο *SIGMOD Conference*, σελίδες 665–676, 2007.

[58] Mehmet Ercan Nergiz, Maurizio Atzori, Yücel Saygin και Baris Güç. Towards trajectory anonymization: a generalization-based approach. *TDP*, 2(1):47–75, 2009.

[59] Mehmet.Ercan Nergiz, Acar Tamersoy και Yucel Saygin. Instant anonymization. *TODS*, 36(2):1–33, 2011.

[60] M.Ercan Nergiz και Chris Clifton. Thoughts on k-anonymization. *DKE*, 2007.

[61] Netflix Prize FAQ. http://www.netflixprize.com/faq, 2009.

[62] Hyoungmin Park και Kyuseok Shim. Approximate algorithms for k-anonymity. Στο *SIGMOD*, σελίδες 67–78, 2007.

[63] P. Samarati. Protecting Respondents' Identities in Microdata Release. *TKDE*, 13(6):1010–1027, 2001.

[64] Pierangela Samarati και Latanya Sweeney. Generalizing Data to Provide Anonymity when Disclosing Information (abstract). Στο *PODS (see also Technical Report SRI-CSL-98-04)*, 1998.

[65] Latanya Sweeney. Achieving k-anonymity privacy protection using generalization and suppression. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(5):571–588, 2002.

[66] Latanya Sweeney. k-Anonymity: A Model for Protecting Privacy. *IJUFKS*, 2002.

[67] Yufei Tao, Xiaokui Xiao, Jiexing Li και Donghui Zhang. On anti-corruption privacy preserving publication. Στο *ICDE*, 2008.

[68] M. Terrovitis, N. Mamoulis και P. Kalnis. Privacy-preserving Anonymization of Set-valued Data. *PVLDB*, 1(1):115–125, 2008.

[69] Manolis Terrovitis και Nikos Mamoulis. Privacy Preservation in the Publication of Trajectories. Στο *MDM*, 2008.

[70] Manolis Terrovitis, Nikos Mamoulis και Panos Kalnis. Local and global recoding methods for anonymizing set-valued data. *VLDBJ*, 2010.

[71] Manolis Terrovitis, Nikos Mamoulis και Panos Kalnis. Local and global recoding methods for anonymizing set-valued data. *The VLDB Journal*, 20(1):83–106, 2011.

[72] Manolis Terrovitis, Nikos Mamoulis, John Liagouris και Spiros Skiadopoulos. Privacy preservation by disassociation. *Proceedings of the VLDB Endowment*, 5(10):944–955, 2012.

[73] R. C. W. Wong, J. Li, A. W. C. Fu και K. Wang. (α, k)-anonymity: An enhanced k-anonymity privacy-preserving data publishing. Στο *SIGKDD*, 2006.

[74] Raymond Chi Wing Wong, Jiuyong Li, Ada Wai Chee Fu και Ke Wang. (alpha, k)-anonymity: an enhanced k-anonymity model for privacy preserving data publishing. Στο *KDD*, σελίδες 754–759, 2006.

[75] K. Yi X. Xiao και Y. Tao. The hardness and approximation algorithms for l-diversity. Στο *Proceedings of the 13th International Conference on Extending Database Technology (EDBT), Lausanne, Switzerland*, 2010.

[76] X. Xiao, G. Wang και J. Gehrke. Differential privacy via wavelet transforms. *TKDE*, σελίδες 1200–1214, 2010.

[77] Xiaokui Xiao, Gabriel Bender, Michael Hay και Johannes Gehrke. ireduct: differential privacy with reduced relative errors. Στο *SIGMOD*, 2011.

[78] Xiaokui Xiao και Yufei Tao. Anatomy: simple and effective privacy preservation. Στο *VLDB*, 2006.

[79] Mingqiang Xue, Panagiotis Karras, Chedy Raïssi, Panos Kalnis και Hung Keng Pung. Delineating social network data anonymization via random edge perturbation. Στο *CIKM*, σελίδες 475–484, 2012.

[80] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi και A. Fu. Utility-Based Anonymization Using Local Recoding. Στο *KDD*, σελίδες 785–790, 2006.

[81] Yabo Xu, Ke Wang, Ada Wai Chee Fu και Philip S. Yu. Anonymizing transaction databases for publication. Στο *KDD*, σελίδες 767–775, 2008.

[82] Roman Yarovoy, Francesco Bonchi, Laks VS Lakshmanan και Wendy Hui Wang. Anonymizing moving objects: how to hide a mob in a crowd? Στο *EDBT*, 2009.

[83] Qing Zhang, Nick Koudas, Divesh Srivastava και Ting Yu. Aggregate Query Answering on Anonymized Tables. Στο *ICDE*, 2007.

[84] Lei Zou, Lei Chen και M. Tamer Özsu. K-automorphism: A general framework for privacy preserving network publication. *PVLDB*, 2(1):946–957, 2009.

Παράρτημα Α΄

Μεταφράσεις Ξένων Όρων

Μετάφραση	Αγγλικός όρος
ανεστραμμένη λίστα	inverted list
ανωνυμία	anonymity
απαλοιφή	suppression
αποκάλυψη ταυτότητας	identity disclosure
αποκάλυψη γνωρίσματος	attribute disclosure
αποσυσχέτιση	disassociation
γενίκευση	generalization
γνώρισμα	attribute
δάσος	forest
δένδρο	tree
εξειδίκευση	specialization
εξόρυξη δεδομένων	data mining
ιδιωτικότητα	privacy
κλάση	class
κλάση ισοδυναμίας	equivalence class
κόμβος	node
μοναδικό αναγνωριστικό	unique identifier
μονοπάτι	path
ολική ανακωδικοποίηση	global recoding
οντότητα	entity
πρότερη γνώση	background knowledge
στοίβα	stack
συγχώνευση	merge
συναθροιστική συνάρτηση	aggregate function
σύνδεσμος	link
τοπική ανακωδικοποίηση	local recoding
υποστήριξη	support
ψευδο-αναγνωριστικό	quasi-identifier

k-ανωνυμία	k-anonymity
l-διαφορετικότητα	l-diversity
t-εγγύτητα	t-closeness
δ-παρουσία	δ-presence

Παράρτημα Β΄

Βιογραφικό Σημείωμα

Στοιχεία Επικοινωνίας

Εργαστήριο Συστημάτων Βάσεων Γνώσεων και Δεδομένων
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Εθνικό Μετσόβιο Πολυτεχνείο
Ηρώων Πολυτεχνείου 9. Ζωγράφου
157 80 Αθήνα, Ελλάδα

Τηλέφωνο: (+30) 210 772 1402
Fax: (+30) 210 772 1442
Ηλεκτρονικό ταχυδρομείο (e-mail): `olga@dblab.ece.ntua.gr`
Προσωπική Ιστοσελίδα: `http://www.dblab.ece.ntua.gr/~olga`

Σπουδές και Επαγγελματικές Άδειες

- **Εθνικό Μετσόβιο Πολυτεχνείο**, Ελλάδα (2008–2015)
  Διδακτορικό στη Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών.
  Περιοχή έρευνας: Προστασία της Ιδιωτικότητας σε Δημοσιεύσεις Δεδομένων
  Επιβλέπων: καθ. Ιωάννης Βασιλείου

- **Εθνικό Μετσόβιο Πολυτεχνείο**, Ελλάδα (2001–2007)
  Δίπλωμα Ηλεκτρολόγου Μηχανικού και Μηχανικού Υπολογιστών.
  Διπλωματική εργασία: Σύνθεση Προσχεδίων Εκτέλεσης Ερωτημάτων Διαρκείας
  Επιβλέπων: καθ. Τιμολέων Σελλής

- Άδεια άσκησης επαγγέλματος Ηλεκτρολόγου Μηχανικού και Μηχανικού Υπολογιστών από Τεχνικό Επιμελητήριο Ελλάδας (ΤΕΕ). (2008)

- Επάρκεια διδασκαλίας της Αγγλικής Γλώσσας από το Υπουργείο Παιδείας. (2004)

- Απολυτήριο Λυκείου από το $2^ο$ Ενιαίο Λύκειο Ηλιούπολης (Βαθμός 19.3)

Ερευνητικά Ενδιαφέροντα

- Προστασία Ιδιωτικότητας και Ανωνυμοποίηση
- Εξόρυξη Γνώσης από Δεδομένα
- Διασυνδεδεμένα Δεδομένα
- Επεξεργασία στο Νέφος
- Ρεύματα Δεδομένων

Διακρίσεις και Υποτροφίες

- **Ε.Σ.Π.Α, Υπουργείο Παιδείας - Ευρωπαϊκή Ένωση** (2010-2013)
  Υποτροφία υποστήριξης διδακτορικής έρευνας (Ηράκλειτος ΙΙ)
- **ΤΕΕ** Υποψηφιότητα βράβευσης διπλωματικής εργασίας (2008)

Ακαδημαϊκή Εμπειρία

- **Εθνικό Μετσόβιο Πολυτεχνείο**, Ελλάδα (2007-2010)
  Βοηθός Διδασκαλίας
  - Βάσεις Δεδομένων (Χειμερινό 2011-12)
  - Βάσεις Δεδομένων (Χειμερινό 2010-11)
  - Βάσεις Δεδομένων (Χειμερινό 2009-10)
  - Προγραμματιστικές Τεχνικές (Θερινό 2010-11)
  - Προγραμματιστικές Τεχνικές (Θερινό 2009-10)
  - Εισαγωγή στον Προγραμματισμό (Χειμερινό 2008-09)

  Συνεπιβλέπουσα Διπλωματικών Εργασιών
  - Κατερίνα Λεπενιώτη, *«Προστασία Ιδιωτικότητας από Επιτιθέμενους με Συναθροιστική Γνώση»* (2013)
  - Σωτήρης Καρράς και Φήβη Πανοπούλου, *«Εργαλείο Ανωνυμοποίησης»* (2014)
  - Σωτήρης Αγγελή, *«Ανωνυμοποίηση Συλλογών Δεδομένων Με Συνεχή Γνωρίσματα»* (2014)

Ερευνητικά Έργα

- **ΙΠΣΥ, Ε. Κ. Αθηνά**, Ελλάδα
  - Βοηθός Έρευνας σε Ανωνυμοποίηση Πολυδιάστατων Δεδομένων (Έργο ΜΕΔΑ).

Εξωτερική Κριτής

- Σε διεθνή Επιστημονικά Συνέδρια (2009-2015): CIKM, SIGMOD, PADM, ADBIS, EDBT, KDD, MDM, MEDI, DATA,και Περιοδικά: TKDE, TKDD, IS, VLDB.

Τεχνικές Ικανότητες

- **Προγραμματισμός**: Pascal, Fortran, Perl, C, Java, C++, SQL, SparQL, HTML, PHP, XML, Javascript.
- **Λειτουργικά Συστήματα**: Windows, Linux, Mac OS
- **Άλλα**: MySQL, Berkeley DB, MS SQL Server, Visual Studio .NET, Eclipse, LaTeX, Map-Reduce framework.

Ξένες Γλώσσες

- **Αγγλικά** Certificate of Proficiency - *University of Cambridge*
- **Γαλλικά** DELF - *Institut Français de Grèce (ifa)*

Δημοσιεύσεις

1. Olga Gkountouna and Manolis Terrovitis, *Anonymizing Collections of Tree-Structured Data*, IEEE Transactions on Knowledge and Data Engineering (TKDE), 27(8): 2034-2048, August 2015.
2. Olga Gkountouna Manolis Terrovitis and Yannis Vassiliou, *Preventing Identity Disclosure by Attackers with Knowledge of Multiple Functions*, Technical Report, NTUA, 2015.
3. Olga Gkountouna, Sotiris Angeli, Athanasios Zigomitros, Manolis Terrovitis, Yannis Vassiliou, *k^m-Anonymity for Continuous Data Using Dynamic Hierarchies*, Privacy in Statistical Databases 2014: 156-169.
4. Olga Gkountouna and Manolis Terrovitis, *Anonymization in the Presence of Structural Knowledge*, Technical Report, NTUA, 2013.
5. Olga Gkountouna, Katerina Lepenioti, Manolis Terrovitis, *Privacy against Aggregate Knowledge Attacks*, in the Proceedings of the 1st International workshop on Privacy-Preserving Data Publication and analysis (PrivDB 2013), in conjunction with the 29th IEEE International Conference on Data Engineering (ICDEW), p. 99-103, 2013.
6. Όλγα Γκουντούνα, *Μοντέλα Επιθέσεων και Πλαίσιο Ιδιωτικότητας Ημιδομημένων Δεδομένων*, Τεχνική Ανανφορά, ΕΜΠ, 2012.

7. O. Gkountouna, *A Survey on Privacy Preservation Methods*, Technical Report, Knowledge and Database Systems Laboratory, NTUA, 2011.

8. Όλγα Γκουντούνα, *Σύνθεση Προσχεδίων Εκτέλεσης Ερωτημάτων Διαρκείας*, Διπλωματική εργασία στο Εργαστήριο Συστημάτων Βάσεων Γνώσεων και Δεδομένων, ΣΗΜΜΥ, ΕΜΠ, 2007.