ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

k^m-Ανωνυμοποίηση Συλλογών Δεδομένων με Συνεχή Γνωρίσματα

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΣΩΤΗΡΗ Α. ΑΓΓΕΛΗ

Επιβλέπων : Ιωάννης Βασιλείου
Καθηγητής Ε.Μ.Π.

Αθήνα, Φεβρουάριος 2014

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

k^m-Ανωνυμοποίηση Συλλογών Δεδομένων με Συνεχή Γνωρίσματα

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΣΩΤΗΡΗ Α. ΑΓΓΕΛΗ

Επιβλέπων : Ιωάννης Βασιλείου
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 11η Φεβρουαρίου 2014.

.....	.....	.....
Ιωάννης Βασιλείου	Κωνσταντίνος Κοντογιάννης	Ιωάννης Σταύρακας
Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Ερευνητής Β' ΙΠΣΥ/Ε.Κ. «Αθηνά»

Αθήνα, Φεβρουάριος 2014

.....

ΣΩΤΗΡΗΣ Α. ΑΓΓΕΛΗ

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Περίληψη

Πολλοί οργανισμοί, επιχειρήσεις ή δημόσιοι φορείς συλλέγουν και διαχειρίζονται προσωπικά δεδομένα από διάφορους χρήστες, τα οποία μπορούν να δημοσιεύσουν ή να διανέμουν για ερευνητικούς σκοπούς και στατιστικές μελέτες. Η διανομή και χρήση αυτών των δεδομένων μπορεί βλάψει την ιδιωτικότητα των ατόμων, αφήνοντας εκτεθειμένα ευαίσθητα προσωπικά στοιχεία.

Για την προστασία της ιδιωτικότητας των δεδομένων, έχουν προταθεί διάφορες τεχνικές οι οποίες εφαρμόζουν αλγόριθμους ανωνυμοποίησης στα δεδομένα. Η εφαρμογή όμως τέτοιων αλγορίθμων συνεπάγεται απώλεια πληροφορίας. Η ανωνυμοποίηση θα πρέπει να τηρεί τις εγγυήσεις για την προστασία της ευαίσθητης πληροφορίας, χωρίς ταυτόχρονα να καθιστά τα δεδομένα άχρηστα για στατιστική μελέτη.

Η παρούσα διπλωματική εργασία ασχολείται με τη διασφάλιση της ιδιωτικότητας σε συλλογές δεδομένων μέσω της k^m-ανωνυμίας. Η εγγύηση αυτή, εξασφαλίζει ότι η βάση δεδομένων είναι k-ανώνυμη όταν ο επιτιθέμενος γνωρίζει έως και m τιμές από το σύνολο του ψευδο-αναγνωριστικού μιας εγγραφής. Εστιάζει σε σύνολα δεδομένων όπου τα διάφορα γνωρίσματα λαμβάνουν τιμές από ένα κοινό συνεχές πεδίο τιμών.

Ο αλγόριθμος που προτείνεται υλοποιεί την k^m-ανωνυμοποίηση του συνόλου δεδομένων, εκμεταλλευόμενος τις ιδιότητες των συνεχών γνωρισμάτων του συνόλου. Επιτυγχάνει χωρίς την χρήση κάποιας ιεραρχίας γενίκευσης τη διατήρηση περισσότερης πληροφορίας στα δημοσιευμένα δεδομένα.

Λέξεις Κλειδιά:

k^m-ανωνυμία,

ανωνυμοποίηση δεδομένων,

συνεχή γνωρίσματα

προστασία ιδιωτικότητας

Abstract

Many organizations, enterprises or public services collect and manage personal data from various users. These data may be published or distributed for research and statistical studies. However, releasing this information can harm the privacy of individuals by exposing sensitive person-specific data.

Several methods have been proposed to guarantee data privacy in published datasets, using anonymization techniques. However, implementation of such algorithms causes information loss in the released data. As a result, the anonymization should protect sensitive information, without making the data useless for statistical study.

This diploma thesis deals with ensuring privacy protection using the k^m-anonymity algorithm of set-valued data. K^m-anonymity guarantees that a database is k-anonymous, if the attacker has background knowledge of up to m items of a transaction. We focus on collections of itemsets. All items take values from a common continuous domain.

We develop an algorithm that implements k^m-anonymization of the given dataset by taking into account the properties of a continuous attribute. The algorithm preserves more information in the published dataset in comparison to other anonymization algorithms that use generalization hierarchy trees.

Keywords:

k^m -anonymity

anonymization algorithm

continuous attributes

privacy protection

Ευχαριστίες

Με την ολοκλήρωση της διπλωματικής μου εργασίας στο Εργαστήριο Συστημάτων Βάσεων Γνώσεων και Δεδομένων του τμήματος Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του ΕΜΠ, νιώθω βαθύτατα την ανάγκη να ευχαριστήσω θερμά όσους συνέβαλαν με οποιονδήποτε τρόπο στην επιτυχή εκπόνηση της παρούσας εργασίας.

Αρχικά θα ήθελα να ευχαριστήσω τον καθηγητή του ΕΜΠ κύριο Ιωάννη Βασιλείου, για την ευκαιρία που μου έδωσε να ασχοληθώ με ένα τόσο ενδιαφέρον αντικείμενο, καθώς και για την επιστημονική καθοδήγησή του κατά την διάρκεια της διεκπεραίωσης της εργασίας αυτής.

Στη συνέχεια, ευχαριστώ ιδιαιτέρα την κυρία Όλγα Γκουντούνα, υποψήφια διδάκτορα ΕΜΠ, για την αμέριστη βοήθεια και για την συμπαράσταση που μου παρείχε από την αρχή της επίβλεψης της παρούσας εργασίας. Η εξαιρετική συνεργασία που είχαμε, και ευελπιστώ ότι θα συνεχίσουμε να έχουμε και στο μέλλον, ήταν ο βασικός παράγοντας που με βοήθησε να ξεπεράσω τις όποιες δυσκολίες αντιμετώπισα. Την ευχαριστώ θερμά για τις ιδέες που μου προσέφερε, καθώς και για την προθυμία που έδειχνε να ασχοληθεί με οποιοδήποτε ζήτημα είχε προκύψει κατά τη διάρκεια εκπόνησης αυτής της εργασίας.

Επιπρόσθετα, οφείλω ένα μεγάλο ευχαριστώ στους συμφοιτητές μου, που έκαναν τα φοιτητικά μου χρόνια μια αξέχαστη εμπειρία ζωής, καθώς και για την ανυπολόγιστη βοήθεια τους όλη αυτή την περίοδο. Δεν θα μπορούσα βέβαια να μην αναφερθώ και στους φίλους μου και να τους ευχαριστήσω που βρίσκονται πάντα δίπλα μου.

Το μεγαλύτερο ευχαριστώ χρωστάω στον πατέρα μου Ανδρέα και στη μητέρα μου Μαρία, όχι μόνο για την ηθική και υλική στήριξη που μου προσέφεραν κατά τη διάρκεια της φοίτησης μου, αλλά και για τις θυσίες που ανιδιοτελώς έκαναν για εμένα σε κάθε βήμα της ζωής μου, προκειμένου να μπορέσω να ολοκληρώσω τις σπουδές μου και να εκπληρώσω απερίσπαστα τους στόχους μου.

Τέλος, νιώθω την ανάγκη να αφιερώσω την παρούσα εργασία στις αδερφές μου Μύρια και Κατερίνα, εκφράζοντας την ευγνωμοσύνη μου για την ακούραστη και συνεχή στήριξη τους όλα αυτά τα χρόνια.

Σωτήρης Α. Αγγελή
Αθήνα, Φεβρουάριος 2014

Πίνακας περιεχομένων

1 Εισαγωγή ... 1

1.1 Προστασία ιδιωτικότητας από επιτιθέμενους ... 1

1.2 Αντικείμενο διπλωματικής ... 3

1.2.1 Συνεισφορά ... 4

1.3 Οργάνωση κειμένου ... 5

2 Θεωρητικό υπόβαθρο ... 7

2.1 Εισαγωγή στην προστασία ιδιωτικότητας ... 7

2.1.1 Χρήσιμοι ορισμοί προστασίας ιδιωτικότητας ... 9

2.1.2 Ιεραρχία Γενίκευσης ... 10

2.1.3 Απώλεια πληροφορίας ... 11

2.2 Επιθέσεις αναγνώρισης ταυτότητας ... 12

2.2.1 k-Ανωνυμία (k-Anonymity) ... 12

2.2.1.1 Γενίκευση (Generalization) ... 13

2.2.1.2 Απόκρυψη Εγγραφών (Suppression) ... 14

2.2.2 Παράμετροι k-Ανωνυμίας ... 14

2.2.3 Αλγόριθμοι εύρεσης k-ανώνυμων πινάκων ... 15

2.2.3.1 Ο αλγόριθμος Incognito ... 15

2.2.3.2 Ο αλγόριθμος Mondrian ... 18

2.3 Επιθέσεις αναγνώρισης τιμής ευαίσθητων δεδομένων ... 20

2.3.1 Αδυναμίες k-Ανωνυμίας ... 21

2.3.2 l-Διαφορετικότητα (l-Diversity) ... 23

2.3.2.1 Περιγραφή μεθόδου ... 23

2.3.2.2 Αδυναμίες l-diversity ... 25

2.3.3 Ανατομία (Anatomy) ... 26

2.3.4 τ-Εγγύτητα (t-Closeness) ... 27

2.4 μ-Αμεταβλητότητα (m-Invariance) ... 28

2.5 δ-Παρουσία (δ-Presence) ... 31

2.6 k^m-Ανωνυμία (k^m-Anonymity) ... 32

2.6.1 Περιγραφή μεθόδου ... 32

2.6.2 Μοντέλο γενίκευσης *33*

2.6.3 Apriori αλγόριθμος ανωνυμοποίησης *34*

3 Ορισμός Προβλήματος **35**

3.1 Μοντέλο δεδομένων 37

3.2 Απειλές κατά της ιδιωτικότητας 38

3.3 Μετρική Κόστους Απώλειας Πληροφορίας 38

3.4 Πιθανές Λύσεις 40

3.4.1 Χρήση k^m-ανωνυμοποίησης με ιεραρχία γενίκευσης *40*

3.4.2 Χρήση αλγόριθμου χωρίς ιεραρχίες γενίκευσης *41*

3.4.2.1 Παράμετρος Μέγιστης Διαφοροποίησης NCP 42

4 Περιγραφή αλγόριθμου **43**

4.1 Θεωρητικό υπόβαθρο 43

4.2 Δέντρο Συχνοτήτων (Count Tree) 44

4.2.1 Συγχώνευση στο δέντρο συχνοτήτων με αύξοντες αριθμούς εγγραφών *46*

4.3 Γενικεύσεις σε συνεχή γνωρίσματα 46

4.4 Υλοποίηση 48

4.5 Παράδειγμα υλοποίησης 52

5 Αξιολόγηση **57**

5.1 Παράμετροι αξιολόγησης 57

5.1.1 Μετρική απώλεια πληροφορίας *58*

5.1.2 Χρόνος εκτέλεσης *58*

5.2 Οργάνωση πειραμάτων 59

5.2.1 Συνθετικά Αριθμητικά Δεδομένα *59*

5.2.2 Πραγματικά Οικονομικά Δεδομένα *59*

5.2.3 Διαδικασία πειραμάτων *60*

5.3 Αποτελέσματα πειραμάτων 61

5.3.1 Συνθετικά Αριθμητικά Δεδομένα *61*

5.3.1.1 Απώλεια πληροφορίας 61

5.3.1.2 Χρόνος εκτέλεσης 65

5.3.2 Πραγματικά Οικονομικά Δεδομένα *68*

5.3.2.1 Απώλεια πληροφορίας 68

5.3.2.2 Χρόνος εκτέλεσης 72

6 Τεχνικές λεπτομέρειες 75

6.1 Λεπτομέρειες υλοποίησης 75

6.1.1 Μορφή δεδομένων εισόδου-εξόδου 76

6.1.2 Εισαγωγή παραμέτρων από το χρήστη 76

6.1.3 Δομές Δεδομένων 77

6.2 Ανάλυση Κλάσεων 78

6.2.1 Κλάση Αρχικοποίησης Initialize 78

6.2.2 Κλάση Ανωνυμοποίησης km_Anonymize 79

6.2.3 Κλάση Υπολογισμού Κόστους Compute_Cost 80

6.3 Ανάλυση Βασικών Μεθόδων 80

6.3.1 Κύρια συνάρτηση 80

6.3.2 Διαδικασία αρχικοποίησης 81

6.3.3 Διαδικασία δημιουργίας συστάδων 81

6.3.4 Διαδικασία ανωνυμοποίησης 82

6.3.5 Διαδικασία υπολογισμού μετρικής απώλειας πληροφορίας 83

6.4 Αλγόριθμος k^m-ανωνυμίας (με ιεραρχίες γενίκευσης) 85

6.4.1 Λεπτομέρειες υλοποίησης 85

6.4.2 Ανάλυση βασικών κλάσεων και μεθόδων 86

6.4.2.1 Κύρια συνάρτηση 86

6.4.2.2 Κλάση αρχικοποίησης Initialize 86

6.4.2.3 Διαδικασία Clustering 87

6.4.2.4 Διαδικασία ανωνυμοποίησης 87

7 Επίλογος 89

7.1 Σύνοψη και συμπεράσματα 89

7.2 Μελλοντικές επεκτάσεις 90

8 Βιβλιογραφία 91

1

Εισαγωγή

1.1 Προστασία ιδιωτικότητας από επιτιθέμενους

Τα τελευταία χρόνια ολοένα και περισσότερα προσωπικά δεδομένα συλλέγονται από διάφορους οργανισμούς προκειμένου να δημοσιοποιηθούν για σκοπούς έρευνας. Ιδιαίτερα με τη ραγδαία ανάπτυξη του διαδικτύου, η συλλογή τέτοιας πληροφορίας στις μέρες μας παρουσιάζεται σε μεγάλο βαθμό και μπορεί να αφορά σε κοινωνικές σχέσεις από διάφορα κοινωνικά δίκτυα, σε ιατρικής φύσης δεδομένα, είτε ακόμα και σε επιχειρηματική και εμπορική δραστηριότητα.

Η διαθεσιμότητα όμως τόσο αναλυτικών προσωπικών δεδομένων θέτει σημαντικούς κινδύνους στην ιδιωτικότητα του καθενός. Ακόμη και με την απόκρυψη στοιχείων που προσδιορίζουν μοναδικά ένα άτομο όπως είναι το ονοματεπώνυμο ή ο Αριθμός Φορολογικού Μητρώου (ΑΦΜ), ο συνδυασμός άλλων στοιχείων όπως ο ταχυδρομικός κώδικας, το φύλο και η ηλικία του, θα μπορούσαν να λειτουργήσουν σαν *ψευδο-αναγνωριστικά (quasi-identifiers)* και να οδηγήσουν στην ταυτοποίηση του ατόμου αποκαλύπτοντας ευαίσθητα προσωπικά δεδομένα (Ασθένεια, Μηνιαία Έσοδα κλπ).

Ένα παράδειγμα ταυτοποίησης εγγραφών αποτελεί το σύνολο δεδομένων μιας φορολογικής βάσης από την οποία αφαιρείται το ονοματεπώνυμο και παραμένουν μόνο πληροφορίες όπως {*Ταχυδρομικός Κώδικας, Οδός, Εισοδήματα*} και έναν τοπικό τηλεφωνικό κατάλογο με δεδομένα όπως {*Όνομα, Επώνυμο, Ταχυδρομικός Κώδικας, Οδός, Τηλέφωνο*}. Αν οι τιμές

των γνωρισμάτων της φορολογική βάσης οδηγούν τον επιτιθέμενο σε μία μοναδική εγγραφή, μπορεί να συμπεράνει τα εισοδήματα του φορολογούμενου, χωρίς σε αυτήν να δημοσιεύεται το ονοματεπώνυμό του, το οποίο μπορεί να ανακαλύψει από τον τηλεφωνικό κατάλογο.

Ο Paul Ohm, αναπληρωτής καθηγητής στη Νομική σχολή του πανεπιστημίου του Κολοράντο δηλώνει ότι, σχεδόν για κάθε άνθρωπο πάνω στη γη, υπάρχει τουλάχιστον μια πληροφορία αποθηκευμένη σε μια βάση δεδομένων η οποία καταγράφει κάποιο γεγονός της ζωής του. Η πληροφορία αυτή μπορεί εύκολα να χρησιμοποιηθεί από κάποιο κακόβουλο πρόσωπο, προκειμένου να βλάψει το θύμα νομικά και ηθικά. Είτε αυτό πρόκειται για εκβιασμό, είτε για παρενόχληση ή ακόμα και για κλοπή ταυτότητας.

Ο τομέας της προστασίας της ιδιωτικότητας ασχολείται με την ανάπτυξη διαφόρων αλγόριθμων και τεχνικών ανωνυμοποίησης, οι οποίοι αφαιρούν αναγνωριστική πληροφορία από τα δεδομένα που παρέχονται έτσι ώστε να μην μπορεί ο επιτιθέμενος να προσδιορίσει μονοσήμαντα ένα άτομο. Μια συνηθισμένη τεχνική που χρησιμοποιείται στα μοντέλα ανωνυμοποίησης είναι αυτή της γενίκευσης. Ως *γενίκευση (generalization)* ορίζεται η διαδικασία κατά την οποία η αρχική τιμή που εμφανίζεται στα δεδομένα, αντικαθίστανται με μία πιο γενική τιμή, όπως για παράδειγμα ένα σύνολο τιμών που μπορεί να την περιέχει. Το σύνολο των πιθανών γενικεύσεων των τιμών κάθε *ψευδο-αναγνωριστικού* γνωρίσματος σχηματίζουν την *ιεραρχία γενίκευσης*.

Με την τεχνική της γενίκευσης χάνεται ένα μέρος της χρήσιμης πληροφορίας που εμφανίζεται στα αρχικά δεδομένα. Για το λόγο αυτό, αναζητούνται εγγυήσεις ιδιωτικότητας οι οποίες αποτρέπουν τη διεξαγωγή προσωπικής πληροφορίας αλλά ταυτόχρονα αφαιρούν όσο το δυνατό λιγότερη πληροφορία από τα αρχικά δεδομένα διατηρώντας την χρηστικότητα τους για εκείνους που θέλουν να τα αξιοποιήσουν. Η βασική εγγύηση ιδιωτικότητας που αποτρέπει σε ικανοποιητικό βαθμό την αναγνώριση της ταυτότητας κάποιας εγγραφής είναι η *k*-ανωνυμία. Σκοπός της συγκεκριμένης εγγύησης είναι κανένας συνδυασμός τιμών να μην εμφανίζεται στη βάση δεδομένων σε λιγότερες από k-εγγραφές.

Λόγω του ότι η διαθέσιμη πληροφορία που μπορεί να κατέχει ο επιτιθέμενος μπορεί να έχει πολλές μορφές και να προέρχεται από παράλληλες δημοσιεύσεις σε πολλές πηγές, έχουν προταθεί διάφορες παραλλαγές της k-ανωνυμίας που εκμεταλλεύονται κάθε φορά τη μορφή της γνώσης του επιτιθέμενου. Μία από αυτές είναι και η k^m-ανωνυμία, η οποία εγγυάται ότι η βάση είναι k-ανώνυμη ακόμα και αν ο επιτιθέμενος γνωρίζει m τιμές από το σύνολο του *ψευδο-αναγνωριστικού* μιας εγγραφής, και προσπαθεί να εντοπίσει την εγγραφή αυτή στα δημοσιευόμενα δεδομένα.

1.2 Αντικείμενο διπλωματικής

Στην παρούσα εργασία εξετάζεται ένα πρόβλημα ιδιωτικότητας που αφορά βάσεις δεδομένων με συνεχή γνωρίσματα από ένα κοινό πεδίο τιμών. Κατά την δημοσίευση των δεδομένων ο συνδυασμός των τιμών κάποιων γνωρισμάτων μπορεί να λειτουργήσει σαν *ψευδο-αναγνωριστικό* και να οδηγήσει στην ταυτοποίηση κάποιας εγγραφής στα δεδομένα.

Ο επιτιθέμενος έχοντας σαν γνωστικό υπόβαθρο ένα σύνολο m τιμών μιας εγγραφής του συνόλου δεδομένων, προσπαθεί με την μερική γνώση που κατέχει να προσδιορίσει σε ποια εγγραφή αντιστοιχούν οι τιμές αυτές. Επιχειρείται η κατάλληλη τροποποίηση των δεδομένων έτσι ώστε ο επιτιθέμενος να μην μπορεί να προσδιορίσει μοναδικά μια εγγραφή.

Χαρακτηριστικό παράδειγμα του συγκεκριμένου προβλήματος εμφανίζεται κατά τη δημοσίευση των φορολογικών δεδομένων που περιέχουν τα εισοδήματα φορολογούμενων πολιτών. Σε περίπτωση της δημοσίευσης των αρχικών τιμών υπάρχει ο κίνδυνος αναγνώρισης κάποιας εγγραφής από όποιον επιτιθέμενο γνωρίζει ένα μέρος των εισοδημάτων ενός φυσικού προσώπου.

Η k-ανωνυμία όπως προτείνεται στο [Swe02] θα μπορούσε να αποτελεί μια λύση για το πιο πάνω πρόβλημα. Η μέθοδος αυτή εξασφαλίζει ότι ο επιτιθέμενος δεν θα μπορεί να προσδιορίσει μοναδικά μια εγγραφή στη βάση δεδομένων, αφού υπάρχουν άλλες k-1 εγγραφές με τις ίδιες τιμές. Η χρήση όμως της k-ανωνυμοποίησης, προκαλεί αρκετές φορές υπεργενίκευση στα δεδομένα, με αποτέλεσμα να προκύπτει μεγάλη απώλεια πληροφορίας. Σύμφωνα με την βιβλιογραφία, την καλύτερη προτεινόμενη λύση για την διασφάλιση της ιδιωτικότητας των εγγραφών στο πρόβλημα αυτό, αποτελεί η k^m-ανωνυμοποίηση των δεδομένων [TMK08]. Η τροποποίηση των δεδομένων μέσω της k^m-ανωνυμοποίησης, μπορεί μέσω της διαδικασίας γενίκευσης με τη χρήση προκαθορισμένης ιεραρχίας, να προστατέψει το σύνολο των δεδομένων από τέτοιου είδους επιθέσεις. Ο επιτιθέμενος δεν θα μπορεί να προσδιορίσει μοναδικά κάποια εγγραφή, για οποιοδήποτε σύνολο μερικής γνώσης κατέχει πάνω στα δημοσιευμένα δεδομένα, και σε αντίθεση με την k-ανωνυμία προκαλεί λιγότερη απώλεια πληροφορίας στα δεδομένα.

Παρ' όλα αυτά, ο αλγόριθμος που προτείνεται στη βιβλιογραφία δεν εκμεταλλεύεται το γεγονός ότι η βάση περιέχει συνεχή γνωρίσματα, με αποτέλεσμα αρκετές φορές κατά τη διαδικασία της γενίκευσης να γενικεύονται τιμές σε υψηλά επίπεδα ιεραρχίας, χωρίς αυτό να είναι απαραίτητο. Ο αλγόριθμος που προτείνεται στην παρούσα εργασία, εκμεταλλεύεται την φύση των δεδομένων και τροποποιεί τα δεδομένα χωρίς τη χρήση ιεραρχίας γενίκευσης. Σε κάθε περίπτωση, γενικεύει τις τιμές σε πιο μεγάλα διαστήματα που περιέχουν τις αρχικές τιμές, διατηρώντας έτσι μεγαλύτερο ποσοστό πληροφορίας στα δεδομένα. Με τον τρόπο αυτό αποτρέπεται η παραβίαση της ιδιωτικότητας των εγγραφών από επιθέσεις αυτής της μορφής

και παράλληλα διατηρείται σημαντικά μεγαλύτερο ποσοστό χρήσιμης πληροφορίας στα δημοσιευμένα δεδομένα. Στη λογική αυτή βασίζεται ο αλγόριθμος που παρουσιάζεται με στόχο την επίλυση του προβλήματος που ορίζεται στη συνέχεια.

Όπως αξιολογείται και από τα πειράματα, ο αλγόριθμος που προτείνεται στην παρούσα διπλωματική εργασία υπερέχει αυτού που παρουσιάζεται στη βιβλιογραφία, μιας και με τη χρήση του εξασφαλίζεται η διατήρηση της ανωνυμίας των εγγραφών που συμμετέχουν στα δημοσιευμένα δεδομένα και παράλληλα, τα δεδομένα εμφανίζουν μεγαλύτερη χρηστικότητα για εκείνους στους οποίους απευθύνονται.

1.2.1 Συνεισφορά

Προκειμένου να λυθεί το πρόβλημα ιδιωτικότητας όπως παρουσιάζεται πιο πάνω αναπτύχθηκε ευριστικός αλγόριθμος γενίκευσης, ολικής ανακωδικοποίησης, την οποία υπολογίζει λαμβάνοντας υπόψη την μερική γνώση πάνω στις τιμές των γνωρισμάτων κάποιας εγγραφής που θεωρητικά γνωρίζει ο επιτιθέμενος. Στόχος του αλγορίθμου είναι η k^m-ανωνυμοποίηση του συνόλου των δεδομένων, με τη χρήση του apriori αλγόριθμου ανωνυμοποίησης όπως περιγράφεται και στο [TMK08], εκμεταλλευόμενος τις ιδιότητες που έχουν τα συνεχή γνωρίσματα του συνόλου δεδομένων.

Στην παρούσα εργασία:

1. Μελετήθηκε σχετική βιβλιογραφία που αφορά την προστασία της ιδιωτικότητας και την ανωνυμοποίηση σε συλλογές δεδομένων με σκοπό την εύρεση της κατάλληλης εγγύησης ιδιωτικότητας για το πρόβλημα.
2. Αναπτύχθηκε και υλοποιήθηκε αλγόριθμος για την επίλυση του προβλήματος της βέλτιστης k^m-ανωνυμοποίησης των δεδομένων με στόχο την προστασία των προσωπικών δεδομένων από τις επιθέσεις με μερική γνώση, σε συλλογές με συνεχή γνωρίσματα, χωρίς τη χρήση ιεραρχίας γενίκευσης.
3. Υλοποιήθηκε apriori αλγόριθμος για την k^m-ανωνυμοποίηση δημοσιευθέντων δεδομένων με χρήση ιεραρχιών γενίκευσης, που στόχο έχει την προστασία των προσωπικών δεδομένων από επιθέσεις με μερική γνώση, σε συλλογές με συνεχή γνωρίσματα.
4. Εκτελέστηκαν πειράματα σύγκρισης των δύο αυτών αλγορίθμων για διαφορετικές συλλογές δεδομένων και διαφορετικές παραμέτρους ανωνυμίας.
5. Αξιολογήθηκαν τα συγκριτικά αποτελέσματα των πειραμάτων και διαπιστώθηκε ως αποδοτικότερη λύση στο πρόβλημα των επιθέσεων με μερική γνώση, η χρήση του

προτεινόμενου αλγορίθμου χωρίς τη χρήση ιεραρχών γενίκευσης, όταν οι συλλογές δεδομένων περιλαμβάνουν συνεχή γνωρίσματα.

1.3 Οργάνωση κειμένου

Η δομή του κειμένου της παρούσας εργασίας έχει ως εξής:

Στο δεύτερο κεφάλαιο παρουσιάζεται η βιβλιογραφία που αφορά έννοιες σχετικές με την προστασία της ιδιωτικότητας σε δημοσιευμένα δεδομένα που περιέχουν ευαίσθητη πληροφορία, τις εγγυήσεις ιδιωτικότητας και τους αντίστοιχους αλγορίθμους που έχουν οριστεί για την τροποποίηση δεδομένων, με σκοπό την ανωνυμοποίηση τους.

Στο τρίτο κεφάλαιο παρουσιάζεται το πρόβλημα της k^m-ανωνυμοποίησης δεδομένων, που στόχο έχει την προστασία ιδιωτικότητας από επιθέσεις με μερική γνώση σε συλλογές με συνεχή γνωρίσματα, και αναλύονται οι πιθανές λύσεις που προτείνονται για την επίλυση του προβλήματος.

Στο τέταρτο κεφάλαιο παρουσιάζεται και αναλύεται ο αλγόριθμος που προτείνεται για την k^m-ανωνυμοποίηση των δεδομένων, ο οποίος επιτυγχάνει λιγότερη απώλεια πληροφορίας λαμβάνοντας υπόψιν την φύση των δημοσιευθέντων δεδομένων

Στο πέμπτο κεφάλαιο περιγράφεται η πειραματική διαδικασία που εκτελέστηκε με σκοπό τη σύγκριση του αλγορίθμου με την έως τώρα προσφερόμενη βέλτιστη λύση για την ανωνυμοποίηση αντίστοιχων συνόλων δεδομένων, καθώς και τα αποτελέσματα που προέκυψαν σχετικά με την απόδοση τους ως προς τον χρόνο εκτέλεσης, την διασφάλιση της ιδιωτικότητας και την διατήρηση της χρήσιμης πληροφορίας των αρχικών δεδομένων.

Στο έκτο κεφάλαιο καταγράφονται οι τεχνικές λεπτομέρειες υλοποίησης των δύο αλγορίθμων που αναπτύχθηκαν σε γλώσσα προγραμματισμού C++, αναλύονται οι βασικές μέθοδοι τους και περιγράφονται οι βασικές δομές δεδομένων που χρησιμοποιήθηκαν.

Στο έβδομο κεφάλαιο συνοψίζονται τα αποτελέσματα της εργασίας σχετικά με την βέλτιστη λύση του προβλήματος ιδιωτικότητας από επιτιθέμενους με μερική γνώση, σε συλλογές δεδομένων με συνεχή γνωρίσματα και προτείνονται μελλοντικές επεκτάσεις του αλγορίθμου.

2

Θεωρητικό υπόβαθρο

Το πρόβλημα της προστασίας ιδιωτικότητας, είναι ένα ανοικτό ζήτημα στο χώρο της επιστήμης των υπολογιστών. Ολοένα και περισσότερα δεδομένα που αφορούν σε ευαίσθητους τομείς της ανθρώπινης δραστηριότητας δημοσιεύονται για στατιστικούς και ερευνητικούς λόγους αλλά και για λόγους διαφάνειας. Είτε πρόκειται για δημοσίευση των δεδομένων σε μια ιστοσελίδα, είτε σε περιορισμένο κύκλο, το ζήτημα σχετικά με την προστασία της ιδιωτικότητας παραμένει, και αποτελεί σήμερα ερευνητικό ενδιαφέρον για πολλούς επιστήμονες στο χώρο της πληροφορικής.

2.1 Εισαγωγή στην προστασία ιδιωτικότητας

Σύμφωνα με μελέτη που έγινε το 2000 με επικεφαλής την καθηγήτρια του πανεπιστημίου του Harvard, Latanya Sweeny, το 87% (216 από τα 248 εκατομμύρια) του πληθυσμού των Ηνωμένων Πολιτειών της Αμερικής, μπορεί να προσδιοριστεί μοναδικά μόνο με τη χρήση του ταχυδρομικού κώδικα, φύλου και ημερομηνίας γεννήσεως κάθε πολίτη [Swe00]. Στην ίδια μελέτη αναφέρεται ότι περίπου ο μισός πληθυσμός των Η.Π.Α. (53%) μπορεί να προσδιοριστεί μοναδικά ακόμα και με γνώση πιο γενικών στοιχείων όπως είναι ο χώρος κατοικίας (πόλη ή δήμος), το φύλο και η ημερομηνία γεννήσεως του καθενός.

Σχήμα 2.1 Συνδυασμός Δεδομένων από Διαφορετικά Σύνολα Εγγραφών

Σε άρθρο της Sweeny το 2002, παρουσιάζεται ένα χαρακτηριστικό παράδειγμα που δείχνει πόσο εύκολα μπορεί ένας κακόβουλος τρίτος ο οποίος κατέχει μια γνώση στα δεδομένα, να προσδιορίσει μοναδικά ένα πολίτη [Swe02]. Ένας απλός συνδυασμός δύο βάσεων δεδομένων που η πρώτη αφορούσε εκλογικά στοιχεία των πολιτών από τους δημόσιους καταλόγους ψηφοφορίας και η δεύτερη ανωνυμοποιημένα ιατρικά δεδομένα από οργανισμό ασφάλισης υγείας, ήταν αρκετά για να ανευρεθεί ο ιατρικός φάκελος του κυβερνήτη της Μασαχουσέτης.

Συγκεκριμένα τα στοιχεία του τότε κυβερνήτης της Μασαχουσέτης, ήταν καταχωρημένα στον οργανισμό ασφάλισης υγείας. Ο κυβερνήτης ζούσε στο Cambridge της Μασαχουσέτης. Σύμφωνα με τους καταλόγους ψηφοφορίας 6 άτομα ήταν γεννημένοι την ίδια ημερομηνία με τον κυβερνήτη, μόνο τρεις από αυτούς ήταν άντρες και μόνο ο κυβερνήτης είχε το συγκεκριμένο ταχυδρομικό κώδικα.

Το πιο πάνω παράδειγμα δείχνει πόσο εύκολα μπορεί να προσδιοριστεί μοναδικά μια εγγραφή με απευθείας σύνδεση κοινών χαρακτηριστικών (attributes) δύο πινάκων. Για να γίνει αυτό πιο κατανοητό, θεωρείται ότι ένας ιατρικός οργανισμός δημοσιεύει τα ιατρικά δεδομένα των ασθενών του. Προκειμένου να εξασφαλίσει προστασία ιδιωτικότητας, ο οργανισμός αποκρύπτει από τον πίνακα που δημοσιεύει, χαρακτηριστικά του κάθε ασθενούς όπως είναι το ονοματεπώνυμο, το τηλέφωνο και ο αριθμός ταυτότητας του. Έστω ότι υπάρχει και ένας άλλος πίνακας με τα εκλογικά δεδομένα μιας περιοχής που έχει κοινά γνωρίσματα με τον ανωνυμοποιημένο πίνακα (ημερομηνία γεννήσεως, φύλο, ταχυδρομικό κώδικα), όπως φαίνεται στους πίνακες 2.1 και 2.2.

Από τη σύνδεση των δυο πιο πάνω συνόλων εγγραφών εύκολα μπορεί κάποιος να προσδιορίσει την ασθένεια του Ανδρέα, ο οποίος εισήχθηκε στο νοσοκομείο με γρίπη. Ο λόγος είναι γιατί υπάρχει μόνο μία εγγραφή με έτος γέννησης 1988, ταχυδρομικό κώδικα 53771 και φύλο Άρρεν.

ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ			
Ημερομηνία Γεννήσεως	**Φύλο**	**Ταχυδρ. Κώδικας**	**Ασθένεια**
1988	Άρρεν	53771	Γρίπη
1978	Θήλυ	53772	Ηπατίτιδα
1987	Άρρεν	53771	Γρίπη
1966	Άρρεν	53710	Βρογχίτιδα
1976	Θήλυ	53712	Κάταγμα Χεριού
1966	Άρρεν	53711	Διάστρεμμα

Πίνακας 2.1: Στοιχεία ιατρικού οργανισμού

ΕΚΛΟΓΙΚΑ ΔΕΔΟΜΕΝΑ			
Όνομα	**Ημερομηνία Γεννήσεως**	**Φύλο**	**Ταχυδρ. Κώδικας**
Ανδρέας	1988	Άρρεν	53771
Αλίκη	1988	Θήλυ	55410
Βασιλική	1984	Θήλυ	90210
Δημήτρης	1987	Άρρεν	02174
Ελένη	1965	Θήλυ	02237
Ζωή	1975	Θήλυ	04356
Ηλίας	1974	Άρρεν	12734

Πίνακας 2.2: Στοιχεία εκλογικού καταλόγου

2.1.1 Χρήσιμοι ορισμοί προστασίας ιδιωτικότητας

Προκειμένου να γίνει πιο κατανοητή η μελέτη του προβλήματος της προστασίας ιδιωτικότητας κρίνεται σκόπιμο σε αυτό το σημείο να αναλυθούν κάποιοι ορισμοί που χρησιμοποιούνται συχνά στην επιστημονική κοινότητα και αφορούν την σχεσιακή βάση δεδομένων.

- **Προσωπικά Δεδομένα:** το σύνολο των προσωπικών πληροφοριών ενός ατόμου όπως το φύλο του, το επάγγελμά του, η ηλικία του ή ο μισθός του τα οποία το καθορίζουν. Τα δεδομένα αυτά συνήθως συγκεντρώνονται σε βάσεις δεδομένων, για να μπορεί να είναι πιο εύκολη η μαζική επεξεργασία και μεταφορά τους.

- **Πίνακας (Table)**: τα δεδομένα που βρίσκονται αποθηκευμένα σε μια βάση, είναι οργανωμένα σε μορφή πίνακα $RT\ (A_1, A_2,..,A_n)$ σχεσιακής βάσης δεδομένων όπου τα $A_1, A_2,..,A_n$ είναι οι στήλες-γνωρίσματα του. Στο πίνακα 2.1 τα *Ιατρικά Δεδομένα* αφορούν πίνακα με σύνολο γνωρισμάτων *(Ημερομηνία Γεννήσεως, Φύλο, Ταχυδρομικός Κώδικας, Ασθένεια)*.

- **Πλειάδα (Tuple):** ένα σύνολο τιμών στον πίνακα σχεσιακής βάσης δεδομένων. Πρόκειται για μια εγγραφή η οποία αφορά ένα άτομο και τις τιμές του στα αντίστοιχα πεδία πληροφορίας. Στον πίνακα 2.1 το σύνολο τιμών *(1976, Θήλυ, 53712, Κάταγμα χεριού)* αφορά μια πλειάδα και αναφέρεται σε συγκεκριμένο άτομο.

- **Στήλη – γνώρισμα (Attribute):** κάθε στήλη του πίνακα σχεσιακής βάσης δεδομένων αναφέρεται σε ένα ξεχωριστό γνώρισμα, που αντιπροσωπεύει μια κατηγορία πληροφορίας και έχει ένα σύνολο πιθανών τιμών. Για παράδειγμα στον πίνακα 2.1 η στήλη-γνώρισμα *{Ταχυδρομικός Κώδικας},* έχει πεδίο τιμών τους ταχυδρομικούς κώδικες της περιοχής.

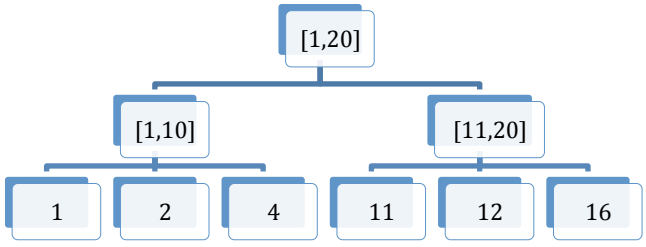
- **Ιδιότητες κλειδιά (Key Attributes):** τιμές-γνωρίσματα τα οποία δεν πρέπει να δημοσιευθούν, γιατί προσδιορίζουν άμεσα κάποιο φυσικό πρόσωπο (π.χ. Αριθμός Δελτίου Ταυτότητας ή Αριθμός Φορολογικού Μητρώου).
- **Ψευδο-αναγνωριστικά (Quasi Identifiers):** το ελάχιστο σύνολο γνωρισμάτων $QI=A_1, A_2,.., A_d$ με το οποίο ένας πίνακας *RT* μπορεί να διασταυρωθεί με κάποιες εξωτερικές πληροφορίες και να αναγνωριστούν ατομικές εγγραφές [MGK+06]. Στο παράδειγμα των πινάκων 2.1 και 2.2, το σύνολο *QI={Ημερομηνία Γεννήσεως, Φύλο, Ταχυδρομικός Κώδικας}*.

- **Ευαίσθητα γνωρίσματα (Sensitive attributes):** πεδία ενός πίνακα RT τα οποία θέλουμε να αποκρύψουμε τις πληροφορίες τους. Στο παράδειγμά το ευαίσθητο γνώρισμα ήταν το πεδίο *{Ασθένεια}*.

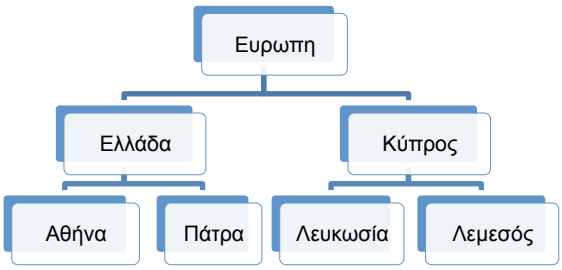
2.1.2 Ιεραρχία Γενίκευσης

Η μέθοδος της γενίκευσης είναι μια πολύ χρήσιμη τεχνική στο χώρο της προστασίας ιδιωτικότητας. Με την χρήση της, επιτυγχάνεται η αντικατάσταση της αρχικής τιμής ενός πεδίου, με μια άλλη τιμή πιο γενική. Αυτό έχει σαν αποτέλεσμα να διατηρείται μέρος της πληροφορίας που περιέχει η αρχική τιμή χωρίς να αλλοιώνεται πλήρως, όπως συμβαίνει στην περίπτωση της απόκρυψης της αρχικής τιμής. Με τη σειρά της αυτή η γενικευμένη τιμή μπορεί να γενικευτεί ξανά σε μια πιο γενικευμένη τιμή διατηρώντας πάλι την ίδια σημασιολογία με την αρχική τιμή του πεδίου κοκ. Όλα τα επίπεδα γενίκευσης στα οποία μπορεί να γενικευθεί μια τιμή μιας βάσης δεδομένων αποτελούν μια ιεραρχία γενίκευσης και συνήθως απεικονίζονται σε μορφή δέντρου.

Το πεδίο τιμών ενός γνωρίσματος μπορεί να αφορά είτε αριθμητικά δεδομένα είτε κατηγορικά δεδομένα. Στην πρώτη περίπτωση, κάθε αριθμός γενικεύεται σε ένα όριο τιμών και στη συνέχεια αυτό το όριο σε ένα πιο μεγάλο όριο, όπως φαίνεται στο σχήμα 2.2.



Σχήμα 2.2: Ιεραρχία γενίκευσης αριθμητικού πεδίου τιμών



Σχήμα 2.3: Ιεραρχία γενίκευσης κατηγορικού πεδίου τιμών

Στην δεύτερη περίπτωση των κατηγορικών δεδομένων, κάθε πεδίο τιμών γενικεύεται σε μια πιο γενική τιμή βάση της σημασιολογίας των αρχικών τιμών, όπως φαίνεται στο πιο πάνω σχήμα.

Υπάρχουν δύο τρόποι εφαρμογής της τεχνικής γενίκευσης σε μια βάση δεδομένων, η *ολική γενίκευση* και η *τοπική γενίκευση*. Στην πρώτη περίπτωση όλες οι αρχικές τιμές της βάσης αντικαθίστανται μια νέα γενικευμένη τιμή. Εναλλακτικά, στην περίπτωση της τοπικής γενίκευσης, ένα υποσύνολο των αρχικών τιμών αντικαθίσταται με την νέα γενικευμένη τιμή. Με αυτό τον τρόπο η κάθε τιμή στη βάση ανήκει σε διαφορετικό επίπεδο γενίκευσης. Η μέθοδος της τοπικής γενίκευσης δίνει αρκετή ευελιξία στους αλγόριθμους ανωνυμοποίησης, αυξάνει όμως πάρα πολύ τον χώρο των πιθανών λύσεων [TMK08].

2.1.3 Απώλεια πληροφορίας

Σκοπός της δημοσίευσης μιας βάσης δεδομένων είναι η εκμετάλλευση της χρήσιμης πληροφορίας που περιέχουν τα δεδομένα, για στατιστικούς ή ερευνητικούς σκοπούς. Όπως αναφέρθηκε και πιο πάνω, προκειμένου να εξασφαλισθεί η ιδιωτικότητα των εγγραφών πρέπει τα δεδομένα να τροποποιηθούν. Οι αρχικές τιμές των δεδομένων αντικαθιστώνται με πιο γενικές τιμές, προκειμένου να διαφυλαχθεί η προσωπική πληροφορία και να μην δημοσιεύεται.

Μια τέτοια γενίκευση έχει σαν αποτέλεσμα, την απώλεια χρήσιμης πληροφορίας που έχουν τα αρχικά δεδομένα, με αποτέλεσμα τα συμπεράσματα που μπορεί να βγουν από μια στατιστική μελέτη των δεδομένων να μην είναι τόσο ακριβή όσο θα ήταν αν δημοσιεύονταν οι αρχικές τιμές.

Ο όγκος της χρήσιμης πληροφορίας που χάνεται κάθε φορά, είναι το πιο σημαντικό μέγεθος στην προστασία της ιδιωτικότητας των δεδομένων. Για αυτό το λόγο, όλοι οι αλγόριθμοι ανωνυμοποίησης που χρησιμοποιούνται για να εξασφαλίσουν την ιδιωτικότητα στις βάσεις δεδομένων, εξετάζονται ως προς την αποδοτικότητα τους σχετικά με αυτήν.

2.2 Επιθέσεις αναγνώρισης ταυτότητας

Υπάρχουν δύο ειδών επιθέσεις στον τομέα της προστασίας ιδιωτικότητας, οι επιθέσεις αναγνώρισης ταυτότητας και οι επιθέσεις αναγνώρισης τιμής ευαίσθητων δεδομένων. Σε αυτή την ενότητα εξετάζεται η πρώτη περίπτωση και στην επόμενη ενότητα το δεύτερο είδος επιθέσεων.

Στην πρώτη περίπτωση, ο επιτιθέμενος προσπαθεί να ταυτοποιήσει μια εγγραφή του δημοσιευμένου πίνακα με κάποιο φυσικό πρόσωπο. Ο επιτιθέμενος μπορεί να εξακριβώσει την τιμή του *ψευδο-αναγνωριστικού* ενός ατόμου με τη χρήση προηγούμενης γνώσης, ή με τον συνδυασμό άλλων δημοσιευμένων πινάκων.

Στον πίνακα 2.3, αν ο επιτιθέμενος γνωρίζει κάποιες τιμές του *ψευδο-αναγνωριστικού*, για παράδειγμα το σύνολο{*1988, Άρρεν*} μπορεί αμέσως να βρει σε ποια εγγραφή αντιστοιχούν, και ότι ο συγκεκριμένος ασθενής πάσχει από γρίπη.

ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ			
Ημερομηνία Γεννήσεως	**Φύλο**	**Ταχυδρ. Κώδικας**	**Ασθένεια**
1988	Άρρεν	53771	Γρίπη
1978	Θήλυ	53772	Ηπατίτιδα
1987	Άρρεν	53771	Γρίπη
1966	Άρρεν	53710	Βρογχίτιδα
1976	Θήλυ	53712	Κάταγμα Χεριού
1966	Άρρεν	53711	Διάστρεμμα

Πίνακας 2.3: Επίθεση αναγνώρισης ταυτότητας

Οι συγκεκριμένες επιθέσεις συναντούνται συχνά στο χώρο της προστασίας ιδιωτικότητας, μιας και είναι πολύ πιθανές να γίνουν, όταν ο πίνακας δημοσιεύεται με την αρχική του μορφή. Σκοπός των αλγόριθμων ανωνυμοποίησης, είναι να τροποποιήσουν τα δεδομένα του αρχικού πίνακα έτσι ώστε να μπορεί να δημοσιευθεί όσο το δυνατό περισσότερη πληροφορία και ταυτόχρονα να προστατεύονται τα συμμετέχοντα πρόσωπα από επιθέσεις που σκοπό έχουν την αναγνώριση ταυτότητας.

2.2.1 k-Ανωνυμία (k-Anonymity)

Στόχος λοιπόν, της προστασίας ιδιωτικότητας είναι να μειωθεί η πιθανότητα να προσδιοριστεί μοναδικά μια συγκεκριμένη οντότητα, ακόμα και με τη διασταύρωση δημοσιευμένων εγγραφών που μπορεί να είναι ανωνυμοποιημένες. Για να γίνει αυτό, πρέπει ουσιαστικά να υπάρχει μια μέγιστη πιθανότητα το πολύ k, ένας επιτιθέμενος να μπορεί να ανακαλύψει με κάποια σύνδεση πινάκων, σε ποιο άτομο ανήκει μια εγγραφή.

2-ΑΝΩΝΥΜΟΣ ΠΙΝΑΚΑΣ			
ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ			
Ημερομηνία Γεννήσεως	**Φύλο**	**Ταχυδρ. Κώδικας**	**Ασθένεια**
198*	Άρρεν	53771	Γρίπη
197*	Θήλυ	53772	Ηπατίτιδα
198*	Άρρεν	53771	Γρίπη
1966	Άρρεν	5371*	Βρογχίτιδα
197*	Θήλυ	53772	Κάταγμα Χεριού
1966	Άρρεν	5371*	Διάστρεμμα

Πίνακας 2.4 Στοιχεία εκλογικού καταλόγου σε 2-Ανωνυμία

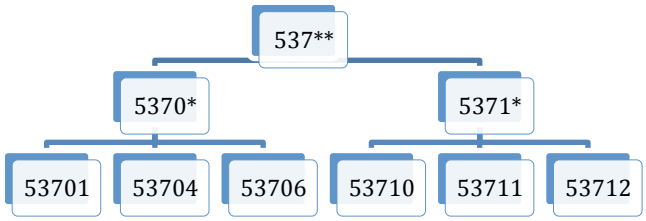
Το 2002 η Sweeny προτείνει την μέθοδο της k-Ανωνυμίας (k-Anonymity) [Swe02]. Με τη μέθοδο αυτή εξασφαλίζεται ότι, η πιθανότητα ανακάλυψης της ταυτότητας μιας εγγραφής είναι το πολύ 1/k. Για την καλύτερη κατανόηση της μεθόδου της k-Ανωνυμίας, χρησιμοποιείται το παράδειγμα του Πίνακα 2.1. Από αυτόν παράγεται ο πίνακα 2.4, οποίος έχει επεξεργαστεί για να ικανοποιεί την k-Ανωνυμία (k=2).

Αν κάποιος κακόβουλος, προσπαθήσει να διασταυρώσει τα δεδομένα του ανωνυμοποιημένου πίνακα, με τον εξωτερικό πίνακα 2.2, τότε μπορεί να κατασκευάσει με πιθανότητα ½ κάθε εγγραφή. Ο λόγος είναι γιατί ο πίνακας έχει χωριστεί σε διάφορα υποσύνολα *(κλάσεις ισοδυναμίας)*, έτσι ώστε σε κάθε υποσύνολο να υπάρχουν τουλάχιστον δύο εγγραφές (*k=2*) οι οποίες να έχουν τις ίδιες τιμές για τα ψευδο-αναγνωριστικά *{Ημερομηνία Γεννήσεως, Ταχυδρομικός Κώδικας, Φύλο}*. Ο πίνακας ικανοποιεί την *2-Ανωνυμία* γιατί για κάθε δύο εγγραφές του συνόλου δεδομένων υπάρχουν κοινές τιμές στα εν λόγω γνωρίσματα.

Σύμφωνα με τα πιο πάνω ορίζεται ότι, ένας πίνακας RT είναι *k-Ανώνυμος*, αν κάθε εγγραφή του πίνακα είναι ίδια ως προς *τα ψευδο-αναγνωριστικά (QI)* πεδία του, με *k-1* άλλες εγγραφές.

2.2.1.1 Γενίκευση (Generalization)

Μέσω της μεθόδου της γενίκευσης οι τιμές των πεδίων QI μετατρέπονται σε μια πιο γενική μορφή, προκειμένου να δημιουργηθούν κλάσεις ισοδυναμίας. Για την γενίκευση ακολουθείται ένα δέντρο ιεραρχίας. Στον πίνακα 2.4, για τη γενίκευση των τιμών του πεδίου {*Ταχυδρομικός Κώδικας*} ακολουθήθηκε η πιο κάτω ιεραρχία:



Σχήμα 2.4: Δέντρο ιεραρχίας γενίκευσης για ταχυδρομικό κώδικα

1966	Άρρεν	53710	Βρογχίτιδα
1966	Άρρεν	53711	Διάστρεμμα

Generalization

1966	Άρρεν	5371*	Βρογχίτιδα
1966	Άρρεν	5371*	Διάστρεμμα

Σχήμα 2.5: Διαδικασία γενίκευσης για δημιουργία κλάσης ισοδυναμίας

Με αυτό τον τρόπο επιτυγχάνεται η δημιουργία κλάσης ισοδυναμίας για τις δύο πιο πάνω εγγραφές του πίνακα, έτσι ώστε να ικανοποιείται η *2-Ανωνυμία*. Σε αυτή την περίπτωση, όσο πιο ψηλά βρίσκεται η γενίκευση στο δέντρο ιεραρχίας, τόσο πιο μεγάλη είναι και η απώλεια πληροφορίας (*information loss*).

2.2.1.2 Απόκρυψη Εγγραφών (Suppression)

Η δεύτερη μέθοδος που χρησιμοποιείται για την ανωνυμία είναι η μέθοδος suppression. Σε αυτή την περίπτωση αφαιρούνται δεδομένα από το σύνολο εγγραφών, τα οποία παραβιάζουν την k-Ανωνυμία, προκειμένου να ελαχιστοποιηθεί το επίπεδο γενίκευσης και να μειώνεται η απώλεια πληροφορίας στα δεδομένα.

Ημερομηνία Γεννήσεως	Φύλο	Ταχυδρ. Κώδικας
1988	Άρρεν	53771
1978	Θήλυ	53772
1987	Άρρεν	53771
1966	Άρρεν	53710
1999	Άρρεν	43654
1976	Θήλυ	53712
1966	Άρρεν	53711

Suppression
+
Generalization

Ημερομηνία Γεννήσεως	Φύλο	Ταχυδρ. Κώδικας
198*	Άρρεν	53771
197*	Θήλυ	53772
198*	Άρρεν	53771
1966	Άρρεν	5371*
197*	Θήλυ	53772
1966	Άρρεν	5371*

Σχήμα 2.6: Απόκρυψη εγγραφών κατά τη διαδικασία ανωνυμοποίησης

2.2.2 Παράμετροι k-Ανωνυμίας

Όπως έχει αναφερθεί, στόχος κατά την ανωνυμοποίηση μιας βάσης δεδομένων είναι να επιτευχθεί η καλύτερη δυνατή γενίκευση με τη λιγότερη απώλεια πληροφορίας, προκειμένου η βάση να είναι αξιοποιήσιμη για σκοπούς έρευνας. Οι παράμετροι οι οποίοι περιγράφουν το πρόβλημα της *k-Ανωνυμίας* και είναι ανταγωνιστικές όσον αφορά την απώλεια πληροφορίας είναι:

1. **Απόκρυψη εγγραφών (Suppression)**: ο αριθμός των εγγραφών που αφαιρούνται από τα δεδομένα, στη διαδικασία της ανωνυμοποίησης

2. Γενίκευση (Generalization): ο όγκος της πληροφορίας που χάνεται γενικεύοντας τα δεδομένα σε κάποιο επίπεδο γενίκευσης (όσο πιο ψηλά στην ιεραρχίας γενίκευσης τόσο μεγαλύτερη απώλεια πληροφορίας)

3. Ανωνυμία (Anonymity): το ελάχιστο ανεκτό μέγεθος k για κάθε κλάση ισοδυναμίας

2.2.3 Αλγόριθμοι εύρεσης k-ανώνυμων πινάκων

Στα πλαίσια της συγκεκριμένης διπλωματικής εργασίας μελετήθηκαν δύο αλγόριθμοι εύρεσης k-ανώνυμων πινάκων, ο Incognito και ο Mondrian. Και οι δύο αλγόριθμοι είναι ευρέως διαδεδομένοι στο χώρο της προστασίας ιδιωτικότητας, χρησιμοποιώντας και οι δύο την τεχνική της γενίκευσης.

Σκοπός τους είναι η γενίκευση ενός συνόλου δεδομένων στα πλαίσια της k-ανωνυμίας με τη λιγότερη δυνατή απώλεια πληροφορίας. Και οι δύο αλγόριθμοι στοχεύουν στην εφαρμογή της k-ανωνυμίας στα δεδομένα, παρέχοντας όμως διαφορετικής ποιότητας ανωνυμοποιήσεις λόγω των διαφορετικών μεθόδων που ακολουθούν.

2.2.3.1 Ο αλγόριθμος Incognito

Ο αλγόριθμος Incognito χρησιμοποιεί *γενίκευση πλήρους πεδίου*. Αντιστοιχίζει κάθε τιμή ενός γνωρίσματος, με την ίδια γενικευμένη τιμή σε όλες τις τιμές του πίνακα [LDR05]. Χρησιμοποιώντας την προκαθορισμένη ιεραρχία γενίκευσης δημιουργεί ένα πλέγμα γενίκευσης πολλαπλών γνωρισμάτων. Στο πλέγμα παρουσιάζονται σχηματικά όλοι οι δυνατοί συνδυασμοί μεταξύ των επιπέδων των ιεραρχιών γενίκευσης των γνωρισμάτων του ψευδο-αναγνωριστικού, όπου εκφράζονται ουσιαστικά όλες οι δυνατές γενικεύσεις των πλειάδων. Οι συνδυασμοί αυτοί, ελέγχονται για την ικανοποίηση της *k*-ανωνυμίας. Στόχος του αλγορίθμου είναι η εύρεση της ελάχιστης γενίκευσης πλήρους πεδίου, προκειμένου να υπάρχει η λιγότερο δυνατή απώλεια πληροφορίας.

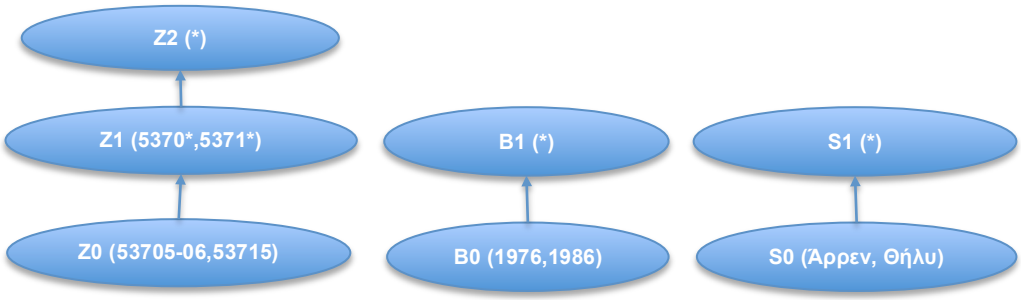
Ο αλγόριθμος Incognito χρησιμοποιεί την *ιδιότητα του υποσυνόλου (subset property),* σύμφωνα με την οποία, αν ένας πίνακας T είναι k-ανώνυμος ως προς ένα σύνολο γνωρισμάτων Q της βάσης δεδομένων, τότε είναι k-ανώνυμος και ως προς οποιοδήποτε υποσύνολο γνωρισμάτων $P \subseteq Q$.

Στο πιο κάτω παράδειγμα παρουσιάζονται τα ιατρικά δεδομένα ενός οργανισμού και η διαδικασία που ακολουθεί ο αλγόριθμος Incognito προκειμένου να ελέγξει όλες τις δυνατές γενικεύσεις στο πλέγμα.

ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ			
Ημερομηνία Γεννήσεως	**Φύλο**	**Ταχυδρ. Κώδικας**	**Ασθένεια**
1976	Άρρεν	53715	Γρίπη
1986	Θήλυ	53715	Ηπατίτιδα
1976	Άρρεν	53703	Βρογχίτιδα
1976	Άρρεν	53703	Κάταγμα Χεριού
1986	Θήλυ	53706	Πυρετός
1976	Άρρεν	53706	Διάστρεμμα

Πίνακας 2.5: Ιατρικά Δεδομένα Οργανισμού

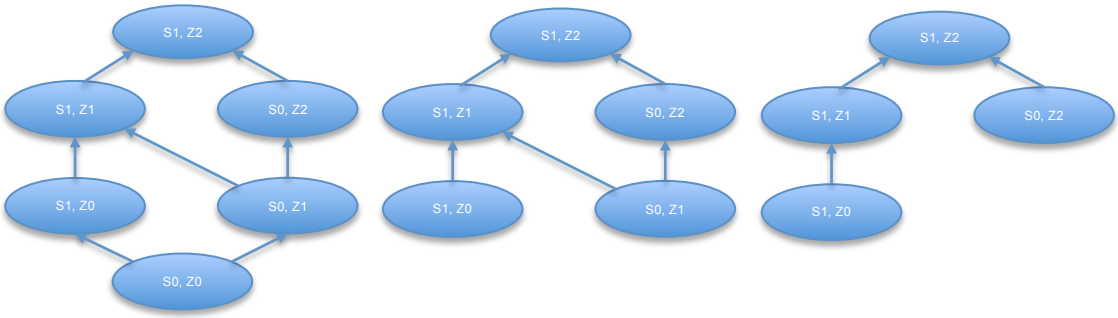
Στην πρώτη επανάληψη για *i=1*, ο αλγόριθμος ελέγχει αν ο πίνακας T είναι k-ανώνυμος για γενικεύσεις ενός συνόλου γνωρισμάτων με μέγεθος *i=1*. Ελέγχει αρχικά, την ανωνυμία του πίνακα T, αν κρατήσει μόνο το πεδίο <*Ημερομηνία Γεννήσεως*>, και απομακρύνει τα πεδία <*Φύλο*> και <*Ταχυδρομικός Κώδικας*>. Βρίσκει ότι ο πίνακας είναι k-ανώνυμος με βάση αυτό το υποσύνολο και άρα με όλες τις γενικευμένες τιμές που ορίζονται από τα domain αυτού *(ιδιότητα γενίκευσης)*. Επαναλαμβάνει την διαδικασία και για τα υπόλοιπα γνωρίσματα του ψευδο-αναγνωριστικού.



Σχήμα 2.7: Βήμα1 Incognito: Ιεραρχίες γενίκευσης πεδίων {Ημερομηνία Γεννήσεως, Ταχυδρομικός Κώδικας, Φύλο}

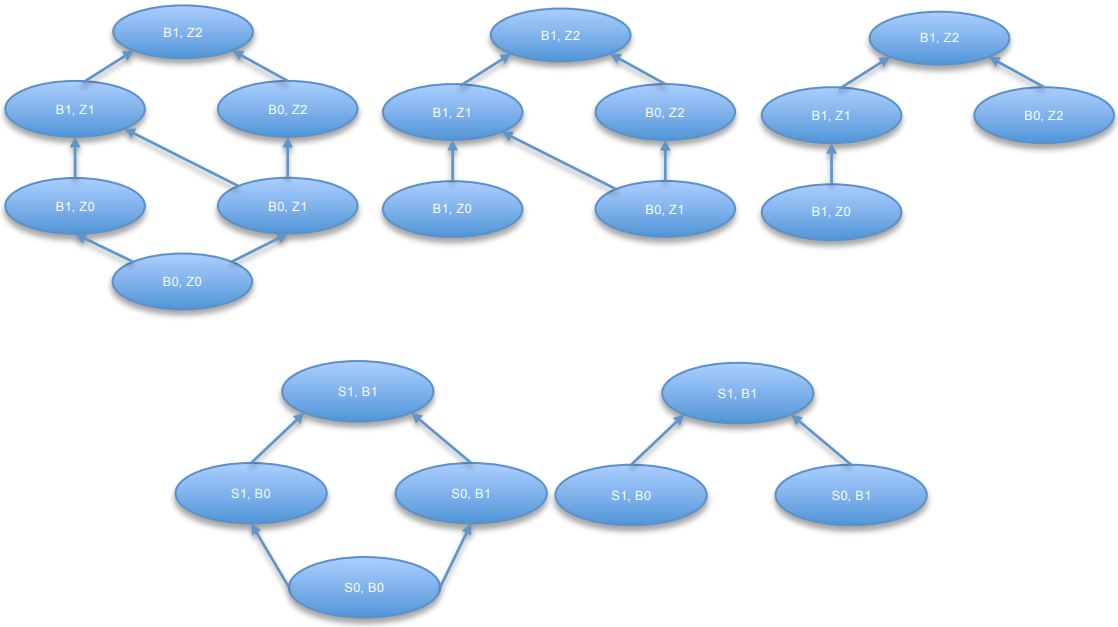
Στη δεύτερη επανάληψη ο αλγόριθμος θα ελέγξει αν ισχύει η k-Ανωνυμία για τα υποσύνολα γνωρισμάτων με μέγεθος *i=2*, <*Ημερομηνία Γεννήσεως* , *Φύλο*> , <*Ημερομηνία Γεννήσεως, Ταχυδρομικός Κώδικας*> και <*Φύλο, Ταχυδρομικός Κώδικας*>.

Για παράδειγμα ο αλγόριθμος ελέγχει αρχικά το frequency set του συνόλου <S0,Z0> και βλέπει ότι δεν ικανοποιείται η k-ανωνυμία, οπότε περνά στον έλεγχο του <S1,Z0> και <S0,Z1>. Με βάση το <S1,Z0> ο πίνακας ικανοποιεί το k-anonymity και κατ' επέκταση όλες οι γενικεύσεις του το επιτυγχάνουν *(ιδιότητα γενίκευσης)*. Στη συνέχεια ελέγχει το frequency set του <S0,Z1> και βλέπει ότι δεν ικανοποιείται το k-anonymity οπότε και απορρίπτεται. Ο συνδυασμός <S1,Z1> δεν ελέγχεται γιατί είναι γενίκευση του <S1, Z0>. Ο επόμενος έλεγχος είναι το σύνολο <S0,Z2> με βάση το οποίο ικανοποιείται η k-ανωνυμία, οπότε και σταματά ο έλεγχος όπως φαίνεται και πιο κάτω.



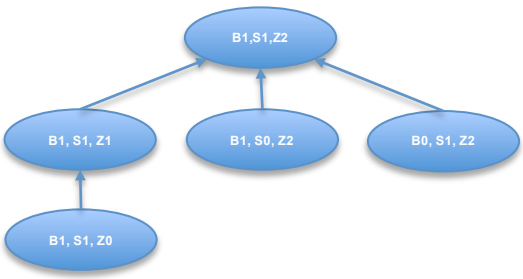
Σχήμα 2.8: Βήμα 2 Incognito: Απόρριψη κόμβων στο υποσύνολο γνωρισμάτων <Φύλο, Ταχυδρομικός Κώδικας>

Η ίδια διαδικασία ακολουθείται για όλα τα υποσύνολα γνωρισμάτων. Στο πιο κάτω σχήμα φαίνονται ποιοι κόμβοι απορρίπτονται στην $2^{η}$ επανάληψη του αλγόριθμου για τα υπόλοιπα σύνολα γνωρισμάτων μεγέθους i=2.



Σχήμα 2.9: Βήμα 2 Incognito: Απόρριψη κόμβων στο υποσύνολο γνωρισμάτων <Φύλο, Ημερ. Γεννήσεως> και <Ταχυδρ. Κώδικας, Ημερ. Γεννήσεως>

Στο τελευταίο βήμα, ο αλγόριθμος αντιστρέφει την ιδιότητα του υποσυνόλου. Έπεται πως αν ένα υποσύνολο γνωρισμάτων του ψευδο-αναγνωριστικού δεν ικανοποιεί την k-ανωνυμία, το ίδιο θα ισχύει και για κάθε σύνολο γνωρισμάτων που το περιέχει. Με τη χρήση αυτής, δημιουργεί το τελικό πλέγμα όλων των k-ανώνυμων συνδυασμών των διαφορετικών επιπέδων γενίκευσης όλων των γνωρισμάτων του ψευδο-αναγνωριστικού, όπως φαίνεται και στο πιο κάτω σχήμα, από όπου και επιλέγεται ο πιο αποδοτικός.



Σχήμα 2.10: Βήμα 3 Incognito: Τελικό πλέγμα γενίκευσης αλγόριθμου Incognito

Η πολυπλοκότητα του αλγόριθμου είναι τελικά εκθετική ως προς το μέγεθος του συνόλου των γνωρισμάτων του ψευδο-αναγνωριστικού. Πρόκειται για ένα ορθό και πλήρη αλγόριθμο ως προς την k-ανωνυμοποίηση που παράγει, με σημαντικό όμως μειονέκτημα το γεγονός ότι παράγει όλες τις δυνατές ανωνυμοποιήσεις του πίνακα από τις οποίες χρησιμοποιείται μόνο η πιο αποδοτική, με αποτέλεσμα ο αλγόριθμος να είναι χρονοβόρος.

2.2.3.2 Ο αλγόριθμος Mondrian

Όπως έχει ήδη αναφερθεί, ο αλγόριθμος Incognito επιστρέφει μία *πλήρους πεδίου γενίκευση*. Αυτό έχει σαν μειονέκτημα την υπεργενίκευση των δεδομένων, με αποτέλεσμα να είναι άχρηστα τις παραπάνω φορές, για εκείνους στους οποίους απευθύνονται. Για παράδειγμα, σε μία βάση με αριθμητικά δεδομένα, μία γενίκευση πλήρους πεδίου σημαίνει αντικατάσταση όλων των αρχικών τιμών με σταθερά μη επικαλυπτόμενα μεταξύ τους διαστήματα ή πλήρη απόκρυψή τους.

Τέτοιου είδους προβλήματα έρχεται να λύσει ο αλγόριθμος Mondrian [LDR06], προσφέροντας υψηλότερης ποιότητας ανωνυμοποίηση, λόγω του *πολυδιάστατου μοντέλου τοπικής ανακωδικοποίησης* με το οποίο μπορεί να εφαρμοστεί. Με βάση αυτό το μοντέλο, ορίζεται ένας χώρος μ-διαστάσεων, όπου μ το πλήθος των ψευδο-αναγνωριστικών quasi identifiers. Χωρίζοντας αυτό το χώρο σε διαμερίσεις (partitions), αναζητείται μια k-ανώνυμη λύση.

Στο πιο κάτω παράδειγμα ορίζονται τα δύο μοντέλα ανωνυμοποίησης (μονοδιάστατο και πολυδιάστατο) για τον πίνακα των ιατρικών δεδομένων ενός οργανισμού.

Όπως περιγράφεται και στο [LDR06], στη μονοδιάστατη ανωνυμοποίηση, σε κάθε επίπεδο της ιεραρχίας γενίκευσης βρίσκονται τιμές από συγκεκριμένα μη επικαλυπτόμενα διαστήματα. Σε αντίθεση με την πολυδιάστατη ανωνυμοποίηση, που σε κάθε περίπτωση στην ιεραρχία γενίκευσης επιτρέπονται τα επικαλυπτόμενα διαστήματα τιμών (Πίνακας 2.6).

ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ			
Ηλικία	**Φύλο**	**Κώδικας**	**Ασθένεια**
25	Άρρεν	53711	Γρίπη
25	Θήλυ	53712	Ηπατίτιδα
26	Άρρεν	53711	Βρογχίτιδα
27	Άρρεν	53710	Κάταγμα Χεριού
27	Θήλυ	53712	Πυρετός
28	Άρρεν	53711	Διάστρεμμα

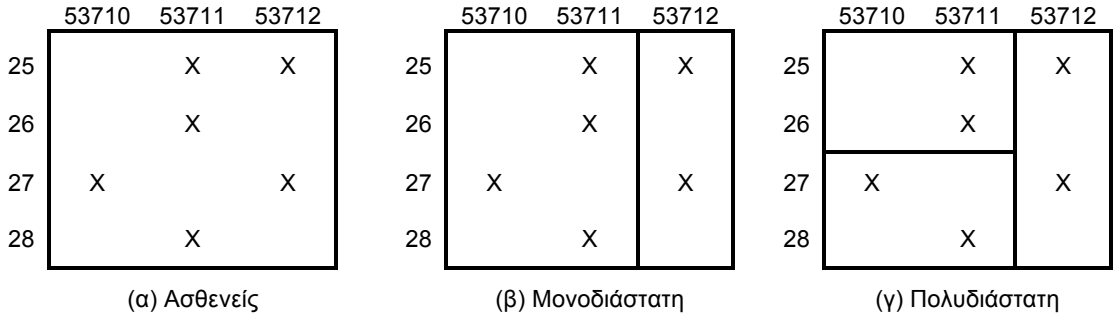
Πίνακας 2.6: Ιατρικά Δεδομένα Οργανισμού

ΜΟΝΟ-ΔΙΑΣΤΑΤΗ ΑΝΩΝΥΜΟΠΟΗΣΗ			
Ηλικία	**Φύλο**	**Κώδικας**	**Ασθένεια**
[25-28]	Άρρεν	[53710-53711]	Γρίπη
[25-28]	Θήλυ	53712	Ηπατίτιδα
[25-28]	Άρρεν	[53710-53711]	Βρογχίτιδα
[25-28]	Άρρεν	[53710-53711]	Κάταγμα Χεριού
[25-28]	Θήλυ	53712	Πυρετός
[25-28]	Άρρεν	[53710-53711]	Διάστρεμμα

ΠΟΛΥ-ΔΙΑΣΤΑΤΗ ΑΝΩΝΥΜΟΠΟΗΣΗ			
Ηλικία	**Φύλο**	**Κώδικας**	**Ασθένεια**
[25-26]	Άρρεν	53711	Γρίπη
[25-27]	Θήλυ	53712	Ηπατίτιδα
[25-26]	Άρρεν	53711	Βρογχίτιδα
[27-28]	Άρρεν	[53710-53711]	Κάταγμα Χεριού
[25-27]	Θήλυ	53712	Πυρετός
[27-28]	Άρρεν	[53710-53711]	Διάστρεμμα

Πίνακας 2.7: Μονοδιάστατη και πολυδιάστατη ανωνυμοποίηση πίνακα ασθενών [LDR06]

Ουσιαστικά, στο μονοδιάστατο μοντέλο για να οριστούν οι σωστές διαμερίσεις (partitions), ο χώρος μοιράζεται τραβώντας παράλληλες ευθείες ως προς τους άξονες και αυτές οι ευθείες διασχίζουν όλο τον χώρο. Αντίθετα στο πολυδιάστατο μοντέλο ορίζονται δύο υποχώροι τραβώντας μια ευθεία παράλληλη ως προς τον ένα άξονα. Στη συνέχεια σε αυτούς τους δύο υποχώρους αναδρομικά ορίζονται άλλοι υποχώροι τραβώντας ευθείες ως προς οποιονδήποτε άξονα, αρκεί οι ευθείες αυτές να μην τέμνουν άλλους υποχώρους. Στους υποχώρους κάθε εγγραφή μπορεί να αναπαρασταθεί σαν ένα σημείο στο χώρο όπως φαίνεται στο πιο κάτω σχήμα. Για να επιλυθεί το k-anonymity αρκεί σε κάθε υποχώρο να υπάρχουν τουλάχιστον k εγγραφές.



Σχήμα 2.11: Χωρική αναπαράσταση ασθενών και διαμερίσεων (QI: Ηλικία, Ταχυδρομικός κώδικας) [LDR06]

Η βασική ιδέα του Mondrian, είναι η αναδρομική διαμέριση του χώρου μ-διαστάσεων με τη χρήση ενός άπληστου αλγόριθμου. Ο αλγόριθμος ακολουθεί την πιο κάτω διαδικασία:

1. Επιλέγει την διάσταση σύμφωνα με την οποία θα γίνει η διαμέριση του χώρου.
2. Υλοποιεί τη διαμέριση βάσει της πιο πάνω διάστασης, από την οποία προκύπτουν δύο υποχώροι R1 και R2.
3. Για κάθε ένα από τους δύο υποχώρους R1 και R2, επαναλαμβάνεται η διαδικασία μέχρι να μην υπάρχει άλλη επιτρεπόμενη τομή για διαμέριση σε καμία διάσταση.
4. Προκύπτει η βέλτιστη πολυδιάστατη διαμέριση και συνεπώς η κατάλληλη πολυδιάστατη γενίκευση που θα χρησιμοποιηθεί.

Ακολουθώντας την πιο πάνω διαδικασία ο αλγόριθμος επιτυγχάνει να βρει τη βέλτιστη πολυδιάστατη διαμέριση, σε κάθε περιοχή της οποίας ανήκουν περισσότερες από k-εγγραφές και συνεπώς ικανοποιείται η k-ανωνυμία.

Το πρόβλημά εύρεσης της βέλτιστης διαμέρισης είναι NP-hard. Ο αλγόριθμος Mondrian δεν δίνει την βέλτιστη λύση, δίνει όμως μια ικανοποιητική προσέγγιση σε σύγκριση με άλλα μοντέλα που έχουν προταθεί και σε αρκετά ικανοποιητικό χρόνο, αφού έχει πολυπλοκότητα O(*nlogn*), όπου n ο αριθμός των εγγραφών του πίνακα.

Πρόβλημα του αλγόριθμου παραμένει το γεγονός ότι δεν μπορεί να υπολογίσει την απώλεια πληροφορίας.

2.3 Επιθέσεις αναγνώρισης τιμής ευαίσθητων δεδομένων

Στη δεύτερη περίπτωση επιθέσεων, ο επιτιθέμενος ενδιαφέρεται να συμπεράνει την τιμή που λαμβάνει ένα άτομο σε ένα ευαίσθητο γνώρισμα στο δημοσιευμένο σύνολο δεδομένων. Για παράδειγμα, στην περίπτωση που ο επιτιθέμενος γνωρίζει ότι το άτομο βρίσκεται στον δημοσιευμένο πίνακα, καθώς και κάποιες από τις τιμές που λαμβάνει σε κάποια από τα γνωρίσματα του συνόλου, μπορεί να ταυτοποιήσει το άτομο με κάποια ή κάποιες εγγραφές και στη συνέχεια μπορεί να διεξάγει συμπεράσματα για τις τιμές του στα υπόλοιπα γνωρίσματα, με το ενδιαφέρον κυρίως στις τιμές των γνωρισμάτων που αναφέρονται σε απόρρητη πληροφορία.

Τέτοιου είδους επιθέσεις αφορούν την αναγνώριση τιμής ευαίσθητων δεδομένων και είναι δύσκολο να αντιμετωπιστούν με τη χρήση της k-ανωνυμίας, όσο καλά και αν επιλεγούν τα ψευδο-αναγνωριστικά.

2.3.1 Αδυναμίες k-Ανωνυμίας

Το πρώτο πρόβλημα αφορά την σειρά που εμφανίζονται οι πλειάδες στον ανωνυμοποιημένο πίνακα. Στο πιο κάτω παράδειγμα από τον αρχικό πίνακα προκύπτουν, μετά από γενίκευση, οι δημοσιευμένοι πίνακες 1 και 2 οι οποίοι ικανοποιούν την k-Ανωνυμία για k=2. Η σειρά εμφάνισης των πλειάδων τόσο του αρχικού πίνακα, όσο και των δύο δημοσιευμένων πινάκων είναι η ίδια. Σε περίπτωση που δημοσιευθεί πρώτα ο πίνακας 1 και ύστερα ακολουθήσει ο πίνακας 2, είναι δυνατό με μια σύνδεση (direct linking) να κατασκευαστούν όλες οι εγγραφές του αρχικού πίνακα.

ΑΡΧΙΚΟΣ ΠΙΝΑΚΑΣ	
Δήμος	**Έτος Γέννησης**
Περιστέρι	1987
Περιστέρι	1999
Περιστέρι	1986
Περιστέρι	1998
Πετρούπολη	1987
Πετρούπολη	1999
Πετρούπολη	1986
Πετρούπολη	1998

ΠΙΝΑΚΑΣ 1	
Δήμος	**Έτος Γέννησης**
Αττική	1987
Αττική	1999
Αττική	1986
Αττική	1998
Αττική	1987
Αττική	1999
Αττική	1986
Αττική	1998

ΠΙΝΑΚΑΣ 2	
Δήμος	**Έτος Γέννησης**
Περιστέρι	198*
Περιστέρι	199*
Περιστέρι	198*
Περιστέρι	199*
Πετρούπολη	198*
Πετρούπολη	199*
Πετρούπολη	198*
Πετρούπολη	199*

Πίνακας 2.8: Αδυναμία k-Anonymity με πίνακες ίδιας σειράς πλειάδων

Υπάρχουν πολλές περιπτώσεις που ακόμα και αν η σειρά εμφάνισης των πλειάδων σε δύο δημοσιευμένους πίνακες είναι διαφορετική, ένας πίνακας συνεχίζει να είναι ευάλωτος σε επιθέσεις ακόμη και αν ικανοποιείται η k-Ανωνυμία.

Στο παράδειγμα που ακολουθεί οι πίνακες Α1 και Α2 προκύπτουν από την ανωνυμοποίηση του αρχικού πίνακα Α με ψευδο-αναγνωριστικά (quasi identifiers) το σύνολο *QI={Χώρα, Ημερομηνία Γεννήσεως, Φύλο, Ταχυδρομικός Κώδικας*} και ικανοποιούν την 2-Ανωνυμία (k=2).

ΑΡΧΙΚΟΣ ΠΙΝΑΚΑΣ				
ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ				
Α/Α	**Ταχυδρ. Κώδικας**	**Ηλικία**	**Εθνικότητα**	**Ασθένεια**
1	13053	28	Ρωσία	Καρδιοπάθεια
2	13068	29	Αμερική	Καρδιοπάθεια
3	13068	21	Ιαπωνία	Αμυγδαλίτιδα
4	13053	23	Αμερική	Αμυγδαλίτιδα
5	14853	50	Ινδία	Πυρετός
6	14853	55	Ρωσία	Καρδιοπάθεια
7	14850	47	Αμερική	Αμυγδαλίτιδα
8	14850	49	Αμερική	Αμυγδαλίτιδα
9	13053	31	Αμερική	Πυρετός
10	13053	37	Ινδία	Πυρετός
11	13068	36	Ιαπωνία	Πυρετός
12	13068	35	Αμερική	Πυρετός

Πίνακας 2.9: Πίνακας ιατρικών δεδομένων

ΠΙΝΑΚΑΣ Α1				
ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ				
Χώρα	**Ημ/νία Γεννήσεως**	**Φύλο**	**Ταχυδρ. Κώδικας**	**Ασθένεια**
Ελλάδα	1965	Άρρεν	15141	Αμυγδαλίτιδα
Ελλάδα	1965	Άρρεν	15141	Ανεμοβλογιά
Ευρώπη	1965	Θήλυ	1513*	Ερυθρά
Ευρώπη	1965	Θήλυ	1513*	Ίλιγγος
Ελλάδα	1964	Θήλυ	15138	Πυρετός
Ελλάδα	1964	Θήλυ	15138	Ανεμοβλογιά
Κύπρος	1964	Άρρεν	1513*	Αμυγδαλίτιδα
Ευρώπη	1965	Θήλυ	1513*	Υπέρταση
Κύπρος	1964	Άρρεν	1513*	Πυρετός
Κύπρος	1964	Άρρεν	1513*	Παχυσαρκία
Κύπρος	1967	Άρρεν	15138	Χοληστερόλη
Κύπρος	1967	Άρρεν	15138	Ωτίτιδα

ΠΙΝΑΚΑΣ Α2				
ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ				
Χώρα	**Ημ/νία Γεννήσεως**	**Φύλο**	**Ταχυδρ. Κώδικας**	**Ασθένεια**
Ελλάδα	1965	Άρρεν	15141	Αμυγδαλίτιδα
Ελλάδα	1965	Άρρεν	15141	Ανεμοβλογιά
Ελλάδα	1965	Θήλυ	15138	Ερυθρά
Ελλάδα	1965	Θήλυ	15138	Ίλιγγος
Ελλάδα	1964	Θήλυ	15138	Πυρετός
Ελλάδα	1964	Θήλυ	15138	Ανεμοβλογιά
Κύπρος	1960-1969	Άρρεν	15138	Αμυγδαλίτιδα
Κύπρος	1960-1969	[Αρρ/Θήλυ]	15139	Υπέρταση
Κύπρος	1960-1969	[Αρρ/Θήλυ]	15139	Πυρετός
Κύπρος	1960-1969	[Αρρ/Θήλυ]	15139	Παχυσαρκία
Κύπρος	1960-1969	Άρρεν	15138	Χοληστερόλη
Κύπρος	1960-1969	Άρρεν	15138	Ωτίτιδα

Πίνακας 2.10: Πίνακες ιατρικών δεδομένων με k-Anonymity (k=2)

Έστω ότι ο πίνακα Α1 δημοσιεύεται. Εάν στη συνέχεια δημοσιευθεί και ο πίνακας Α2, η προστασία k-Ανωνυμίας δεν μπορεί πλέον να εγγυηθεί ακόμη και αν η σειρά των εγγραφών είναι διαφορετική. Αν συνδυαστούν οι πίνακες Α1 και Α2 με βάση το γνώρισμα {Ασθένεια}, μπορεί να κατασκευαστεί ο πίνακας Σ1. Τα γνωρίσματα [Κύπρος, 1964, Άρρεν, 15138] και [Κύπρος, 1965, Θήλυ, 15139] είναι μοναδικά στον πίνακα Σ1, και έτσι ο πίνακας δεν ικανοποιεί την k-Ανωνυμία.

Αυτό το πρόβλημα δεν θα υπήρχε εάν ο πίνακας Α2 είχε κατασκευαστεί με quasi identifiers το σύνολο QI ∪{Ασθένεια}, ή αν ο Α2 είχε σαν βάση τον Α1.

ΠΙΝΑΚΑΣ Σ1				
ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ				
Χώρα	**Ημ/νία Γεννήσεως**	**Φύλο**	**Ταχυδρ. Κώδικας**	**Ασθένεια**
Ελλάδα	1965	Άρρεν	15141	Αμυγδαλίτιδα
Ελλάδα	1965	Άρρεν	15141	Ανεμευλογιά
Ελλάδα	1965	Θήλυ	15138	Ερυθρά
Ελλάδα	1965	Θήλυ	15138	Ίλιγγος
Ελλάδα	1964	Θήλυ	15138	Πυρετός
Ελλάδα	1964	Θήλυ	15138	Ανεμευλογιά
Κύπρος	1964	Άρρεν	15138	Αμυγδαλίτιδα
Κύπρος	1965	Θήλυ	15139	Υπέρταση
Κύπρος	1964	Άρρεν	15139	Πυρετός
Κύπρος	1964	Άρρεν	15139	Παχυσαρκία
Κύπρος	1967	Άρρεν	15138	Χοληστερόλη
Κύπρος	1967	Άρρεν	15138	Ωτίτιδα

Πίνακας 2.11: Πίνακας μετά από σύνδεσή Α1 και Α2 (δεν ικανοποιεί 2-Anonymity)

Το τρίτο πρόβλημα της k-Ανωνυμίας αφορά την δυναμική αλλαγή των στοιχείων των πινάκων. Ανά πάσα στιγμή η πρόσθεση, η αφαίρεση και η αλλαγή πλειάδων σε ένα σύνολο εγγραφών μπορεί να εκθέσει τη βάση σε κίνδυνο.

Έστω ότι, για t=0 υπάρχει ένας πίνακας T_0 και από αυτόν προκύπτει ένας ανωνυμοποιημένος πίνακας A_0 ο οποίος ικανοποιεί την k-Ανωνυμία. Αν σε χρόνο t, προστεθούν στον αρχικό πίνακα κάποιες εγγραφές τότε προκύπτει ο πίνακας T_t.

Με γενίκευση του T_t προκύπτει ο πίνακας A_t, ο οποίος επίσης ικανοποιεί την k-ανωνυμία. Λόγω του ότι δεν υπάρχει καμία εγγύηση η οποία να εξασφαλίζει ότι ο πίνακας A_t έχει σαν βάση τον A_0, τότε όπως και στο προηγούμενο παράδειγμα η σύνδεση των δύο πινάκων μπορεί να μην ακολουθεί την k-Ανωνυμία.

2.3.2 l-Διαφορετικότητα (l-Diversity)

2.3.2.1 Περιγραφή μεθόδου

Σκοπός της προστασίας ιδιωτικότητας σε ένα σύνολο εγγραφών, δεν είναι μόνο η ασφάλεια της ταυτότητας μιας εγγραφής (identity disclosure). Είναι και ταυτόχρονα η διασφάλιση ότι ο επιτιθέμενος δεν θα μπορέσει εύκολα να βρει από αυτό το σύνολο εγγραφών, προσωπικά στοιχεία για ένα άτομο (attribute disclosure).

Για να γίνει αυτό πιο κατανοητό εξετάζονται οι πίνακες του σχήματος 2.12. Παρόλο που το σύνολο εγγραφών του δεύτερου ικανοποιεί την k-Ανωνυμία, αν κάποιο άτομο ανήκει στην τρίτη κλάση ισοδυναμίας, εύκολα ένας αντίπαλος καταλαβαίνει ότι εισήχθηκε στο νοσοκομείο με πυρετό. Το k-Anonymity δεν είναι σε θέση να εξασφαλίσει ότι ο επιτιθέμενος δεν μπορεί να εξάγει με επιτυχία κάποιες πληροφορίες για κάποιες εγγραφές. Αυτό το πρόβλημα είναι γνωστό στη βιβλιογραφία ως *homogeneity attack* [MGK+06].

ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ				
Α/Α	**Ταχυδρ. Κώδικας**	**Ηλικία**	**Εθνικότητα**	**Ασθένεια**
1	130**	<30	*	Καρδιοπάθεια
2	130**	<30	*	Καρδιοπάθεια
3	130**	<30	*	Αμυγδαλίτιδα
4	130**	<30	*	Αμυγδαλίτιδα
5	1485*	>=40	*	Πυρετός
6	1485*	>=40	*	Καρδιοπάθεια
7	1485*	>=40	*	Αμυγδαλίτιδα
8	1485*	>=40	*	Αμυγδαλίτιδα
9	130**	3*	*	Πυρετός
10	130**	3*	*	Πυρετός
11	130**	3*	*	Πυρετός
12	130**	3*	*	Πυρετός

Πίνακας 2.12: Πρόβλημα Homogeneity Attack

Εκτός αυτού, ένα άλλο πρόβλημα που αντιμετωπίζει συχνά η προστασία ιδιωτικότητας δεδομένων, αφορά το γνωστικό υπόβαθρο του επιτιθέμενου πάνω στα δεδομένα *(background knowledge attack)*. Για παράδειγμα έστω ότι ο επιτιθέμενος γνωρίζει ότι μια φοιτήτρια από την Ιαπωνία έχει εισαχθεί στο νοσοκομείο. Τα στοιχεία της βρίσκονται επίσης στον πίνακα του σχήματος 2.12. Η συγκεκριμένη είναι 21 χρονών και διαμένει στην περιοχή με ταχυδρομικό κώδικα 13068. Σύμφωνα με τον πίνακα η κοπέλα ανήκει στις εγγραφές με αριθμό 1,2,3,4. Χωρίς οποιαδήποτε άλλη πληροφορία ο αντίπαλος δεν μπορεί να ξέρει αν η εν λόγω φοιτήτρια πάσχει από καρδιοπάθεια ή αμυγδαλίτιδα. Είναι όμως γενικά γνωστό ότι οι Ιάπωνες δεν αντιμετωπίζουν συχνά προβλήματα με καρδιοπάθειες, οπότε εύκολα συμπεραίνεται ότι η φοιτήτρια εισήχθηκε στο νοσοκομείο με αμυγδαλίτιδα. Η μέθοδος k-Anonymity δεν είναι σε θέση να αντιμετωπίσει ούτε και αυτή την μορφή της επίθεσης.

Η μεθοδολογία που έχει προταθεί για την επίλυση των πιο πάνω αδυναμιών του k-Anonymity, ονομάζεται *l-διαφορετικότητα (l-diversity)* [MGK+06]. Η συγκεκριμένη μέθοδος επιτρέπει στον αντίπαλο να ανακαλύψει με πιθανότητα *1/l* τα ευαίσθητα δεδομένα ενός ατόμου, ανεξάρτητα σε ποια εγγραφή ανήκει.

Ένας πίνακας ικανοποιεί το *l-diversity* αν σε κάθε κλάση ισοδυναμίας του, υπάρχουν τουλάχιστον *l* διαφορετικές τιμές για το σύνολο των ευαίσθητων δεδομένων του.

Για να γίνει αυτό πιο κατανοητό εξετάζεται το παράδειγμα του πίνακα 2.13, ο οποίος βασίζεται στον αρχικό πίνακα 2.12. Ο συγκεκριμένος πίνακας ικανοποιεί το *l-diversity* με *l=3*. Με μια διαφορετική διάταξη των εγγραφών και διαφορετική γενίκευση επιτυγχάνεται η αντιμετώπιση του προβλήματος που παρουσιάστηκε στον προηγούμενο πίνακα. Σε κάθε κλάση ισοδυναμίας υπάρχουν τουλάχιστον τρεις διαφορετικές τιμές για το ευαίσθητο γνώρισμα {Ασθένεια}.

ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ				
Α/Α	**Ταχυδρ. Κώδικας**	**Ηλικία**	**Δήμος**	**Ασθένεια**
1	1305*	<=40	*	Καρδιοπάθεια
4	1305*	<=40	*	Αμυγδαλίτιδα
9	1305*	<=40	*	Πυρετός
10	1305*	<=40	*	Πυρετός
5	1485*	>40	*	Πυρετός
6	1485*	>40	*	Καρδιοπάθεια
7	1485*	>40	*	Αμυγδαλίτιδα
8	1485*	>40	*	Αμυγδαλίτιδα
2	1306*	<=40	*	Καρδιοπάθεια
3	1306*	<=40	*	Αμυγδαλίτιδα
11	1306*	<=40	*	Πυρετός
12	1306*	<=40	*	Πυρετός

Πίνακας 2.13: 3-Διαφορετικός πίνακας

2.3.2.2 Αδυναμίες l-diversity

Ένας βασικός περιορισμός της *l-διαφορετικότητας* είναι η δυνατότητα εξαγωγής συμπερασμάτων από τον αντίπαλο, με μεγάλη βεβαιότητα. Για παράδειγμα, έστω ότι σε μια κλάση ισοδυναμίας υπάρχουν 10 εγγραφές. Στο πεδίο {Ασθένεια} υπάρχει μια τιμή *Πυρετός*, μια τιμή *Αμυγδαλίτιδα* και οι υπόλοιπες οκτώ είναι *Γρίπη*. Η συγκεκριμένη κλάση ικανοποιεί την 3-διαφορετικότητα, αλλά ο επιτιθέμενος μπορεί να συμπεράνει με βεβαιότητα 80% ότι η νόσος του ατόμου-στόχου είναι η γρίπη.

Μια άλλη βασική αδυναμία της συγκεκριμένης μεθόδου είναι η δυσκολία να επιτευχθεί το *l-diversity* ακόμα και σε σχετικά μικρές βάσεις δεδομένων. Σε ένα σύνολο εγγραφών με 10000 εγγραφές χρειάζονται 100 κλάσεις ισοδυναμίας για να ικανοποιείται το *2-diversity*.

Εκτός αυτού η μέθοδος δεν μπορεί να εγγυηθεί με σιγουριά ότι ο επιτιθέμενος δεν θα μπορέσει να βρει από αυτό το σύνολο εγγραφών, προσωπικά στοιχεία για ένα άτομο (attribute disclosure). Για παράδειγμα ο πίνακας 2.14 ικανοποιεί την 3-διαφορετικότητα με σύνολο ευαίσθητων δεδομένων τα γνωρίσματα{*Μισθός, Ασθένεια*}

Αν ο επιτιθέμενος γνωρίζει ότι ο Γιάννης είναι 26 χρονών και διαμένει στην περιοχή με ταχυδρομικό κώδικα 47672, τότε ανήκει στην πρώτη κλάση ισοδυναμίας του πιο πάνω πίνακα. Από αυτόν μπορεί να συμπεράνει ότι ο μισθός του Γιάννη είναι σχετικά μικρός (από 30-50 χιλιάδες) και πάσχει από κάποια καρδιακή πάθηση (*Στεφανιαία Νόσος, Αρτηριακή Υπέρταση ή Καρδιακή Αρρυθμία*). Αυτό συμβαίνει γιατί το *l-diversity* μπορεί μεν να εξασφαλίζει την διαφορετικότητα των ευαίσθητων τιμών σε κάθε ομάδα, δεν λαμβάνει όμως καθόλου υπόψιν τη σημασιολογική εγγύτητα των τιμών αυτών.

ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ			
Ταχυδρ. Κώδικας	**Ηλικία**	**Μισθός**	**Ασθένεια**
476**	2*	30,000	Στεφανιαία Νόσος
476**	2*	40,000	Αρτηριακή Υπέρταση
476**	2*	50,000	Καρδιακή Αρρυθμία
4790*	>40	60,000	Πυρετός
4790*	>40	110,000	Γρίπη
4790*	>40	80,000	Αμυγδαλίτιδα
476**	3*	70,000	Γρίπη
476**	3*	90,000	Αμυγδαλίτιδα
476**	3*	100,000	Καρδιακή Αρρυθμία

Πίνακας 2.14: 3-Διαφορετικός πίνακας (Similarity attack)

2.3.3 Ανατομία (Anatomy)

Με την ανωνυμοποίηση των δεδομένων, χάνεται ένα μεγάλο μέρος της πληροφορίας που περιέχουν, με αποτέλεσμα να μην μπορούν να αξιοποιηθούν. Αυτό οφείλεται στις γενικεύσεις που προκαλούνται στα δεδομένα προκειμένου να δημιουργηθούν κλάσεις ισοδυναμίας. Εκτός αυτού πολλές φορές δεν προστατεύεται η συσχέτιση της κάθε εγγραφής με την ευαίσθητη τιμή της.

Με την τεχνική της ανατομίας, όπως περιγράφεται και στο [XT06] επιτυγχάνεται η μείωση της απώλειας πληροφορίας. Αυτό συμβαίνει γιατί στα δημοσιευμένα δεδομένα παραμένουν οι αρχικές τιμές τόσο των τιμών του ψευδο-αναγνωριστικού, όσο και των τιμών του ευαίσθητου γνωρίσματος. Αυτό που αποκρύπτεται, είναι η συσχέτιση κάθε εγγραφής με την ευαίσθητη τιμή της.

A/A	Ηλικία	Ταχυδρ. Κώδικας	Φύλο	Μισθός
1	25	14540	Θήλυ	500
2	27	14530	Άρρεν	1000
3	34	14550	Άρρεν	1000
4	31	14544	Άρρεν	800
5	37	17430	Θήλυ	950
6	39	18600	Θήλυ	900
7	40	17650	Άρρεν	900
8	38	18200	Θήλυ	700

Πίνακας 2.15: Μισθολόγιο εργαζομένων

Στον πίνακα 2.15, το σύνολο των γνωρισμάτων του *ψευδο-αναγνωριστικού* είναι {*Ηλικία, Ταχυδρομικός Κώδικας, Φύλο*} και το ευαίσθητο γνώρισμα είναι ο {*Μισθός*}.

Αρχικά, ο πίνακας γενικεύεται με τέτοιο τρόπο ώστε να δημιουργούνται κλάσεις ισοδυναμίας που να ικανοποιούν την 4-ανωνυμία και 3-διαφορετικότητα, όπως φαίνεται πιο κάτω.

A/A	Ηλικία	Ταχυδρ. Κώδικας	Φύλο	Μισθός
1	<=35	[14530-14550]	*	500
2	<=35	[14530-14550]	*	1000
3	<=35	[14530-14550]	*	1000
4	<=35	[14530-14550]	*	800
5	>35	[17430-18600]	*	950
6	>35	[17430-18600]	*	900
7	>35	[17430-18600]	*	900
8	>35	[17430-18600]	*	700

Πίνακας 2.16: Ανωνυμοποιημένος πίνακας 4-ανωνυμίας και 3-διαφορετικότητας

Στη συνέχεια δημιουργείται ένας καινούριος πίνακας με τις αρχικές τιμές για τα γνωρίσματα του ψευδο-αναγνωριστικού προσθέτοντας μια καινούρια στήλη η οποία περιέχει τον αριθμό της κλάσης ισοδυναμίας που ανήκει η κάθε εγγραφή.

Α/Α	Ηλικία	Ταχυδρ. Κώδικας	Φύλο	Αριθμός Κλάσης Ισοδυναμίας
1	25	14540	Θήλυ	1
2	27	14530	Άρρεν	1
3	34	14550	Άρρεν	1
4	31	14544	Άρρεν	1
5	37	17430	Θήλυ	2
6	39	18600	Θήλυ	2
7	40	17650	Άρρεν	2
8	38	18200	Θήλυ	2

Πίνακας 2.17: Αρχικός πίνακας με αριθμό κλάσης ισοδυναμίας

Στο τελευταίο βήμα σχηματίζεται συγκεντρωτικός πίνακας, που περιλαμβάνει τις αρχικές τιμές του ευαίσθητου γνωρίσματος, μαζί με τον αριθμό της κλάσης ισοδυναμίας, και τον αριθμό εμφανίσεων της συγκεκριμένης τιμής μέσα στην κλάση ισοδυναμίας.

Ομάδα	Μισθός	Αριθμός Εμφανίσεων
1	500	1
1	1000	2
1	800	1
2	950	1
2	900	2
2	700	1

Πίνακας 2.18: Πίνακας ευαίσθητων τιμών του αρχικού πίνακα

Με τη χρήση της ανατομίας και τη δημοσίευση των δύο τελευταίων πινάκων, ο επιτιθέμενος γνωρίζοντας κάποιες τιμές του ψευδο-αναγνωριστικού, μπορεί μεν να προσδιορίσει αν το άτομο που αναζητά ανήκει σε κάποια εγγραφή, αλλά δεν μπορεί να συσχετίσει με απόλυτη βεβαιότητα καμία εγγραφή με την ευαίσθητη τιμή της κλάσης ισοδυναμίας που ανήκει, αφού κάθε ομάδα ικανοποιεί την 3-διαφορετικότητα.

Η ανατομία προτιμάται αντί της γενίκευσης, στις περιπτώσεις εκείνες που ο επιτιθέμενος γνωρίζει τις τιμές του ψευδο-αναγνωριστικού μιας εγγραφής και είναι βέβαιος ότι το άτομο-στόχος βρίσκεται στις δημοσιευμένες εγγραφές.

2.3.4 τ-Εγγύτητα (t-Closeness)

Σαν αποτέλεσμα των πιο πάνω, κατανομές οι οποίες έχουν το ίδιο επίπεδο ποικιλομορφίας προσφέρουν διαφορετικά επίπεδα ιδιωτικότητας, ανάλογα (α) με τις σημασιολογικές σχέσεις ανάμεσα στις ευαίσθητες τιμές τους, (β) τα διαφορετικά επίπεδα ευαισθησίας των πεδίων και (γ) την ολική κατανομή δεδομένων στη βάση.

Η μέθοδος της τ-εγγύτητας (t-closeness) εξασφαλίζει ότι, η κατανομή ενός ευαίσθητου πεδίου (sensitive attribute) σε κάθε κλάση ισοδυναμίας διαφέρει από την κατανομή του

συγκεκριμένου πεδίου σε όλη την βάση δεδομένων το πολύ κατά ένα κατώφλι t (threshold). Όσο πιο μικρή είναι η τιμή του t, τόσο πιο κοντά βρίσκονται οι δύο κατανομές.

Στον πίνακα 2.19 εξετάζεται το πρόβλημα του similarity attack. Σύμφωνα με το [LLV07] η απόσταση μεταξύ της κατανομής *{Στεφανιαία Νόσος, Αρτηριακή Πίεση, Καρδιακή Αρρυθμία}* και της ολικής κατανομής ισούται με 0.5. Ενώ η απόσταση για την κατανομή *{Πυρετός, Γρίπη, Αμυγδαλίτιδα}* ισούται με 0.278. Αντίστοιχα, η απόσταση για την κατανομή *{30.000, 40.000, 50.000}* υπολογίζεται στα 0.375 ενώ για την κατανομή *{60.000, 110.000, 80.000}* στα 0.278.

Προκειμένου να ελαχιστοποιηθούν οι τιμές του t στην πρώτη κλάση ισοδυναμίας, ανακατασκευάζεται ο πίνακας όπως φαίνεται πιο κάτω.

ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ			
Ταχυδρ. Κώδικας	**Ηλικία**	**Μισθός**	**Ασθένεια**
4767*	<=40	30,000	Στεφανιαία Νόσος
4767*	<=40	50,000	Καρδιακή Αρρυθμία
4767*	<=40	90,000	Αμυγδαλίτιδα
4790*	>=40	60,000	Πυρετός
4790*	>=40	110,000	Γρίπη
4790*	>=40	80,000	Αμυγδαλίτιδα
4760*	<=40	40,000	Αρτηριακή Υπέρταση
4760*	<=40	70,000	Γρίπη
4760*	<=40	100,000	Καρδιακή Αρρυθμία

Πίνακας 2.19: Πίνακας με 0.167-εγγύτητα για {Μισθό} και 0.278-εγγύτητα για {Ασθένεια}

Ο πίνακας 2.19, εξασφαλίζει ιδιωτικότητα ακόμα και μια επίθεση τύπου similarity attack. Για παράδειγμα, ο επιτιθέμενος δεν μπορεί πλέον να συμπεράνει ότι ο Γιάννης είναι χαμηλόμισθος, ούτε ότι πάσχει από κάποια καρδιακή πάθηση.

Η μέθοδος της τ-εγγύτητας προστατεύει από επιθέσεις που σκοπό έχουν την αποκάλυψη ευαίσθητων γνωρισμάτων στο σύνολο εγγραφών (attribute disclosure), αλλά δεν εγγυάται καθόλου την προστασία από επιθέσεις που στόχο έχουν την αποκάλυψη της ταυτότητας μιας εγγραφής στη βάση δεδομένων (identity disclosure). Για αυτό το λόγο πολλές φορές είναι προτιμότερο να χρησιμοποιείται ταυτόχρονα και η k-Ανωνυμία και η τ-εγγύτητα για σκοπούς προστασίας ιδιωτικότητας.

2.4 μ-Αμεταβλητότητα (m-Invariance)

Ένα πρόβλημα και των τριών μεθόδων που περιγράφηκαν πιο πάνω είναι ότι δεν εγγυούνται κάποια ιδιωτικότητα για τα δυναμικά δεδομένα. Καμιά μεθοδολογία δεν υποστηρίζει την εκ νέου δημοσίευση των δεδομένων μετά από τυχόν αλλαγές στη βάση, όπως είναι η προσθήκη

και η διαγραφή δεδομένων. Αυτό αποτελεί σημαντικό πρόβλημα για μια βάση η οποία πρέπει απαραίτητα να μένει πάντα ενημερωμένη. Στο [XT07] προτείνεται μια μεθοδολογία για το *l-diversity*, που ονομάζεται *m-invariance*. Η δεύτερη μέθοδος αποτελεί επέκταση της πρώτης, ώστε να μπορούν να ανωνυμοποιηθούν και δυναμικά δεδομένα.

Έστω ότι ένα νοσοκομείο δημοσιεύει τα δεδομένα των ασθενών του κάθε εξάμηνο. Στον πίνακα 2.20, ο πίνακας T(1) είναι ο αρχικός πίνακας ο οποίος χρησιμοποιείται σαν βάση για να ανωνυμοποιηθεί και να δημοσιευθεί ο πίνακας T*(1). Μετά από έξι μήνες με βάση τα δεδομένα του αρχικού πίνακα T(2) δημοσιεύεται ο πίνακας T*(2).

	ΑΡΧΙΚΟΣ ΠΙΝΑΚΑΣ T (1)		
	ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ		
Όνομα	**Ηλικία**	**Όνομα**	**Ηλικία**
Βασίλης	21	12000	Δυσπεψία
Αλίκη	22	14000	Βρογχίτιδα
Ανδρέας	24	18000	Γρίπη
Δημήτρης	23	25000	Γαστρίτιδα
Γιώργος	41	20000	Γρίπη
Έλενα	36	27000	Γαστρίτιδα
Ιωάννα	37	33000	Δυσπεψία
Κώστας	40	35000	Γρίπη
Λίζα	43	26000	Γαστρίτιδα
Παύλος	52	33000	Δυσπεψία
Σταύρος	56	34000	Γαστρίτιδα

	ΓΕΝΙΚΕΥΜΕΝΟΣ ΠΙΝΑΚΑΣ T* (2)		
	ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ		
Κλάση	**Ηλικία**	**Ταχυδρ. Κώδικας**	**Ασθένεια**
1	[21,22]	[12K-14K]	Δυσπεψία
1	[21,22]	[12K-14K]	Βρογχίτιδα
2	[23,24]	[18K-25K]	Γρίπη
2	[23,24]	[18K-25K]	Γαστρίτιδα
3	[36,41]	[20K-27K]	Γρίπη
3	[36,41]	[20K-27K]	Γαστρίτιδα
4	[37,43]	[26K-35K]	Δυσπεψία
4	[37,43]	[26K-35K]	Γρίπη
4	[37,43]	[26K-35K]	Γαστρίτιδα
5	[52,56]	[33K-34K]	Δυσπεψία
5	[52,56]	[33K-34K]	Γαστρίτιδα

Πίνακας 2.20: Αρχικός πίνακας T(1) και γενικευμένος T*(1) κατά την πρώτη δημοσίευση

	ΑΡΧΙΚΟΣ ΠΙΝΑΚΑΣ T (2)		
	ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ		
Όνομα	**Ηλικία**	**Ταχυδρ. Κώδικας**	**Ασθένεια**
Βασίλης	21	12000	Δυσπεψία
Δημήτρης	23	25000	Γαστρίτιδα
Αμαλία	*25*	*21000*	*Γρίπη*
Ιωάννα	37	33000	Δυσπεψία
Λίζα	43	26000	Γαστρίτιδα
Γιώργος	41	20000	Γρίπη
Μαρία	*46*	*30000*	*Γαστρίτιδα*
Θάνος	*54*	*31000*	*Δυσπεψία*
Σταύρος	56	34000	Γαστρίτιδα
Θωμάς	*60*	*44000*	*Γαστρίτιδα*
Χρήστος	*65*	*36000*	*Γρίπη*

	ΓΕΝΙΚΕΥΜΕΝΟΣ ΠΙΝΑΚΑΣ T* (2)		
	ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ		
Κλάση	**Ηλικία**	**Ταχυδρ. Κώδικας**	**Ασθένεια**
1	[21,23]	[12K-25K]	Δυσπεψία
1	[21,23]	[12K-25K]	Γαστρίτιδα
2	[25,43]	[21K-33K]	Γρίπη
2	[25,43]	[21K-33K]	Δυσπεψία
2	[25,43]	[21K-33K]	Γαστρίτιδα
3	[41,46]	[20K-30K]	Γρίπη
3	[41,46]	[20K-30K]	Γαστρίτιδα
4	[54,56]	[31K-34K]	Δυσπεψία
4	[54,56]	[31K-34K]	Γαστρίτιδα
5	[60,65]	[36K-44K]	Γαστρίτιδα
5	[60,65]	[36K-44K]	Γρίπη

Πίνακας 2.21: Αρχικός πίνακας T(2) και γενικευμένος T*(2) κατά την δεύτερη δημοσίευση

Οι ασθενείς *Αλίκη, Ανδρέας, Έλενα, Κώστας* και *Παύλος* έχουν διαγραφεί από τη βάση δεδομένων. Αντίστοιχα έχουν προστεθεί οι ασθενείς *Αμαλία, Μαρία, Θάνος, Θωμάς* και *Χρήστος*. Παρόλο που και οι δύο δημοσιευμένοι πίνακες ικανοποιούν το *2-anonymity* και το *2-diversity*, ο επιτιθέμενος μπορεί να προσδιορίσει μοναδικά την ταυτότητα ενός ασθενή, με τη σύνδεση των T*(1) και T*(2).

Για παράδειγμα έστω ότι, ο αντίπαλος γνωρίζει την ηλικία και τον ταχυδρομικό κώδικα του Βασίλη και ταυτόχρονα ξέρει ότι τα στοιχεία του είναι δημοσιευμένα και στους δύο πίνακες (η θεραπεία του κράτησε παραπάνω από έξι μήνες). Από τον πίνακα Τ*(1) ο επιτιθέμενος είναι βέβαιος ότι ο Βασίλης πάσχει είτε από *δυσπεψία* είτε από *βρογχίτιδα*. Με βάση τον πίνακα Τ*(2) ο αντίπαλος μπορεί να βρει ότι ο Βασίλης πάσχει είτα από *δυσπεψία* είτε από *γαστρίτιδα*. Με τη σύνδεση των δυο αυτών γνώσεων ο επιτιθέμενος είναι σίγουρος πλέον ότι ο Βασίλης πάσχει από *δυσπεψία*.

Σύμφωνα με την μέθοδο του m-invariance, ο πίνακας Τ*(2) αντικαθίσταται με τον πίνακα Τ(3), όπως φαίνεται πιο κάτω. Συγκεκριμένα ο πίνακας Τ(3) περιλαμβάνει τις γενικευμένες τιμές του πίνακα Τ(2) μαζί με δύο πλαστές εγγραφές Π1 και Π2. Οι 13 εγγραφές κατανέμονται σε 6 κλάσεις ισοδυναμίας.

		ΠΙΝΑΚΑΣ Τ(3) ΜΕ ΠΛΑΣΤΕΣ ΕΓΓΡΑΦΕΣ		
		ΙΑΤΡΙΚΑ ΔΕΔΟΜΕΝΑ		
Όνομα	**Κλάση**	**Ηλικία**	**Ταχυδρ. Κώδικας**	**Ασθένεια**
Βασίλης	1	[21,22]	[12K-14K]	Δυσπεψία
Π1	*1*	*[21,22]*	[12K-14K]	*Βρογχίτιδα*
Δημήτρης	2	[23,25]	[21K-25K]	Γαστρίτιδα
Αμαλία	2	[23,25]	[21K-25K]	Γρίπη
Ιωάννα	3	[37,43]	[26K-33K]	Δυσπεψία
Π2	*3*	*[37,43]*	*[26K-33K]*	*Γρίπη*
Λίζα	3	[37,43]	[26K-33K]	Γαστρίτιδα
Γιώργος	4	[41,46]	[20K-30K]	Γρίπη
Μαρία	4	[41,46]	[20K-30K]	Γαστρίτιδα
Θάνος	5	[54,56]	[31K-34K]	Δυσπεψία
Σταύρος	5	[54,56]	[31K-34K]	Γαστρίτιδα
Θωμάς	6	[60,65]	[36K-44K]	Γαστρίτιδα
Χρήστος	6	[60,65]	[36K-44K]	Γρίπη

Πίνακας 2.22: Πίνακας Τ*(2) με πλαστές εγγραφές

Από την πλευρά του επιτιθέμενου, μια πλαστή εγγραφή δεν ξεχωρίζει από τις υπόλοιπες εγγραφές στην ίδια κλάση ισοδυναμίας. Για παράδειγμα, οι κλάσεις ισοδυναμίας στους πίνακες Τ*(1) και Τ(3) έχουν πλέον το ίδιο σύνολο ευαίσθητων εγγραφών {*δυσπεψία, βρογχίτιδα*}, οπότε ο επιτιθέμενος δεν μπορεί να προσδιορίσει με βεβαιότητα πάνω από 50% την ασθένεια του Βασίλη.

Οι δύο δημοσιεύσεις των πινάκων Τ*(1) και Τ(3), έχουν μια πάρα πολύ σημαντική ιδιότητα και σε αυτήν στηρίζεται και η μέθοδος της μ-αμεταβλητότητας. Εάν μια εγγραφή εμφανίζεται και στις δύο δημοσιεύσεις, θα γενικευτεί σε δύο κλάσεις ισοδυναμίας με τα ίδια ευαίσθητα χαρακτηριστικά και για τις δύο δημοσιεύσεις. Για παράδειγμα η εγγραφή <*Ιωάννα, 37, 33K, δυσπεψία*>, εμφανίζεται και στους δύο πίνακες Τ(1) και Τ(2). Μετά από τη γενίκευση των πινάκων, η εγγραφή ανήκει στις κλάσεις ισοδυναμίας 4 και 3 στους πίνακες Τ*(1) και Τ(3), αντίστοιχα. Οι δύο αυτές κλάσεις ισοδυναμίας έχουν το ίδιο σύνολο ευαίσθητων

γνωρισμάτων {*δυσπεψία, γρίπη, γαστρίτιδα*}. Αυτό οφείλεται στην προσθήκη της πλαστής εγγραφής Π2 στον πίνακα Τ(3).

Για την ακρίβεια το *m-invariance* απαιτεί την ικανοποίηση του *m-diversity* και ταυτόχρονα μια εγγραφή να ανήκει πάντα σε μια κλάση ισοδυναμίας, η οποία έχει το ίδιο σύνολο ευαίσθητων ιδιοτήτων, για όλες τις δημοσιεύσεις.

2.5 δ-Παρουσία (δ-Presence)

Μια άλλη μετρική η οποία χρησιμοποιείται για την προστασία ιδιωτικότητας σε βάσεις δεδομένων είναι η *δ-παρουσία*. Η μέθοδος εγγυάται ότι με την ανωνυμοποίηση της βάσης δεδομένων, ένας επιτιθέμενος δεν θα μπορεί να είναι σε θέση να προσδιορίσει αν κάποιο άτομο συμπεριλαμβάνεται στη συγκεκριμένη βάση με βεβαιότητα μεγαλύτερη από δ.

Οι μέθοδοι του *k-Anonymity* και του *l-diversity* εγγυούνται προστασία ιδιωτικότητας, με την προϋπόθεση ότι ο επιτιθέμενος γνωρίζει πληροφορίες για ένα άτομο και είναι σίγουρος ότι τα στοιχεία του ατόμου αυτού είναι δημοσιευμένα στο σύνολο εγγραφών.

Σε αρκετές περιπτώσεις όμως αυτό δεν είναι αρκετό. Έστω μια βάση η οποία περιέχει τους διαβητικούς μιας χώρας. Ο αντίπαλος δεν πρέπει να είναι βέβαιος ότι το θύμα συμπεριλαμβάνεται σε αυτό τον πίνακα, γιατί τότε ξέρει με βεβαιότητα ότι πάσχει από διαβήτη.

Άλλα παραδείγματα τέτοιου είδους συλλογών δεδομένων, μπορεί να είναι μια βάσης που περιέχει πληροφορίες σχετικά με τις εικαζόμενες τρομοκρατικές ομάδες ή ένα σύνολο δεδομένων ασθενών με ένα συγκεκριμένο τύπο καρκίνου. Και στις δύο περιπτώσεις, προσδιορίζοντας ότι ένα άτομο ή μια ομάδα περιλαμβάνεται στη βάση δεδομένων μπορεί να αποβεί επιζήμιο για την ιδιωτικότητα του ατόμου.

Στο [NAC07] εξετάζεται κατά πόσο τα δεδομένα θεωρούνται επαρκώς ανώνυμα, μέσω της ανάλυσης του κινδύνου επιβεβαίωσης της συμμετοχής ή όχι ενός φυσικού προσώπου στα ανωνυμοποιημένα δεδομένα. Στο συγκεκριμένο άρθρο ορίζεται το πρόβλημα της απόκρυψης παρουσίας ατόμων σε μια βάση δεδομένων και αποδεικνύεται η αδυναμία της k-ανωνυμίας σε περιπτώσεις δημοσίευσης των τιμών των ευαίσθητων γνωρισμάτων των εγγραφών.

2.6 k^m-Ανωνυμία (k^m-Anonymity)

2.6.1 Περιγραφή προβλήματος

Παρ' όλες τις διάφορες τεχνικές που έχουν αναπτυχθεί, και τις διάφορες εγγυήσεις που προτείνονται στη βιβλιογραφία, πολλοί κίνδυνοι για την ιδιωτικότητα παραμένουν χωρίς αποτελεσματική αντιμετώπιση. Η διαθέσιμη πληροφορία που μπορεί να κατέχει ο επιτιθέμενος μπορεί να έχει πολλές μορφές. Παράλληλα, τα μοντέλα των δημοσιευμένων δεδομένων μπορεί να διαφέρουν κάθε φορά, με αποτέλεσμα η κάθε περίπτωση να απαιτεί διαφορετική επεξεργασία προκειμένου να εξασφαλίζεται η ιδιωτικότητα των βάσεων δεδομένων.

Ένα παράδειγμα τέτοιας περίπτωσης έρχεται να λύσει η εφαρμογή της k^m-ανωνυμίας [TMK08]. Σε αυτό το πρόβλημα κάθε εγγραφή αποτελείται από σύνολα δεδομένων που παίρνουν τιμές από ένα κοινό πεδίο τιμών. Ο επιτιθέμενος κατέχει μερική γνώση πάνω στα δεδομένα, γνωρίζοντας m τιμές μιας εγγραφής, και προσπαθεί να εντοπίσει τις υπόλοιπες τιμές της εγγραφής και να τις αντιστοιχίσει με ένα φυσικό πρόσωπο.

Σε αντίθεση με τις προηγούμενες εγγυήσεις ιδιωτικότητας, δεν υπάρχει σαφής διαχωρισμός μεταξύ ευαίσθητων γνωρισμάτων και ψευδο-αναγνωριστικού. Σε κάθε περίπτωση ένα υποσύνολο των τιμών της εγγραφής σχηματίζει το σύνολο του *ψευδο-αναγνωριστικού* και οι υπόλοιπες τιμές σχηματίζουν το σύνολο των *ευαίσθητων γνωρισμάτων*. Η κάθε εγγραφή έχει διαφορετικό μέγεθος, σε αντίθεση με τις σχεσιακές βάσεις δεδομένων που το μέγεθος της κάθε εγγραφής είναι σταθερό.

Σύμφωνα με το [TMK08] πρόκειται για μια νέα εκδοχή της *k*-ανωνυμίας, στην οποία κάθε συνδυασμός τιμών μεγέθους *m*, εμφανίζεται τουλάχιστον *k* φορές στο σύνολο δεδομένων. Ένα παράδειγμα τέτοιων δεδομένων αποτελεί μια βάση η οποία αποθηκεύει τις καθημερινές αγορές πελατών μιας υπεραγοράς. Αν ο επιτιθέμενος γνωρίζει ένα μέρος των αγορών ενός πελάτη μπορεί με ευκολία να προσδιορίσει τις υπόλοιπες αγορές του.

Για την επίλυση του συγκεκριμένου προβλήματος προτείνεται το μοντέλο της k^m-ανωνυμίας. Σύμφωνα με το μοντέλο αυτό, εάν ο επιτιθέμενος γνωρίζει το πολύ *m* τιμές από μια εγγραφή, δεν θα μπορεί να εντοπίσει την εγγραφή αυτή στη βάση γιατί θα υπάρχουν άλλες k-1 εγγραφές με τις ίδιες τιμές.

Στο [TMK08] ένα σύνολο δεδομένων *D* ικανοποιεί την *k^m-ανωνυμία*, αν οποιοσδήποτε συνδυασμός *m* τιμών, παρουσιάζεται σε τουλάχιστον *k* διαφορετικές εγγραφές.

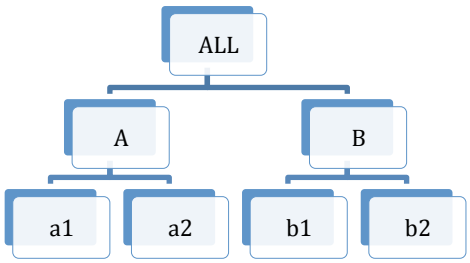
2.6.2 Μοντέλο γενίκευσης

Για την διαδικασία της ανωνυμοποίησης επιλέγεται η *τεχνική της ολικής γενίκευσης*, σύμφωνα με την οποία μια τιμή στη βάση δεδομένων αντικαθίσταται με μια πιο γενική τιμή η οποία περιέχει την αρχική, χωρίς να αλλάζει η σημασιολογία της. Στο παράδειγμα των καθημερινών αγορών μιας υπεραγοράς, η τιμή «γάλα» θα μπορούσε να γενικευτεί σε «γαλακτοκομικά προϊόντα» και η τιμή «ρύζι» σε «δημητριακά».

Το σύνολο των δυνατών γενικεύσεων μιας βάσης δεδομένων αποτελεί το δέντρο της ιεραρχίας γενίκευσης όπως φαίνεται στο πιο κάτω σχήμα. Όσο πιο ψηλά βρίσκεται η γενίκευση τόσο μεγαλύτερη είναι η απώλεια πληροφορίας που παρουσιάζουν τα δεδομένα. Σε μια συλλογή δεδομένων, όλες οι τιμές που υπάρχουν στη βάση πρέπει να βρίσκονται στο δέντρο ιεραρχίας όπως δείχνει το πιο κάτω σχήμα.

Α/Α	ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ
1	{a1, b1, b2}
2	{a2, b1}
3	{a2, b1, b2}
4	{a1, a2, b2}

Πίνακας 2.23 Σύνολο δεδομένων D



Σχήμα 2.12 Ιεραρχία γενίκευσης

Στο συγκεκριμένο παράδειγμα το σύνολο δεδομένων D δεν ικανοποιεί την k^m-ανωνυμία, αφού για $k=2$ και $m=2$ ο συνδυασμός τιμών $\{a1,b1\}$ εμφανίζεται μόνο μια φορά. Η εφαρμογή της γενίκευσης $\{a1, a2\} \rightarrow A$ στη βάση δεδομένων μπορεί να δώσει λύση στο πρόβλημα, αφού πλέον οποιοσδήποτε συνδυασμός $m=2$ τιμών στην βάση, εμφανίζεται σε τουλάχιστον $k=2$ εγγραφές.

Α/Α	ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ
1	{A, b1, b2}
2	{A, b1}
3	{A, b1, b2}
4	{A, A, b2}

Πίνακας 2.24 Σύνολο ανωνυμοποιημένων δεδομένων D′

2.6.3 Apriori αλγόριθμος ανωνυμοποίησης

Σύμφωνα με το [TMK08] ένας αποδοτικός ευριστικός αλγόριθμος που εφαρμόζει k^m-ανωνυμία σε σύνολα δεδομένων είναι αυτός που εκμεταλλεύεται την αρχή της *apriori* ιδιότητας. Σύμφωνα με την ιδιότητα αυτή εάν ένα σύνολο *J* παραβιάζει την ιδιωτικότητα της βάσης, τότε το ίδιο θα συμβαίνει και για οποιοδήποτε υπερσύνολο του *J*. Ο αλγόριθμος ξεκινά και ελέγχει για παραβιάσεις ιδιωτικότητας υποθέτοντας ότι ο επιτιθέμενος γνωρίζει μόνο μια τιμή από το σύνολο του ψευδο-αναγνωριστικού, στη συνέχεια επαναλαμβάνει τον έλεγχο για 2 τιμές και συνεχίζει μέχρι να ελέγξει για *m* τιμές. Το πλεονέκτημα του συγκεκριμένου αλγόριθμου είναι ότι εκμεταλλεύεται τις γενικεύσεις που έγιναν στο βήμα *i* με αποτέλεσμα να μειώνεται ο αριθμός των γενικεύσεων στο βήμα *i+1*.

Ο αλγόριθμος εφαρμόζει τη διαδικασία γενίκευσης σε όλους τους συνδυασμούς τιμών μεγέθους $i = \{1,2,..,m\}$. Σε κάθε βήμα επανάληψης *i*, ο apriori αλγόριθμος καταγράφει σε ένα δέντρο συχνοτήτων *count-tree* τις εμφανίσεις του κάθε συνδυασμού τιμών μεγέθους *i* της βάσης δεδομένων. Στη συνέχεια εντοπίζει στο δέντρο συχνοτήτων τις τιμές στους κόμβους-φύλλα που παρουσιάζουν συχνότητα εμφάνισης μικρότερη από *k*. Για κάθε μια από αυτές τις τιμές ανατρέχει στο δέντρο ιεραρχίας γενίκευσης, και αντικαθιστά τις προβληματικές τιμές με πιο γενικευμένες με στόχο να αυξήσει τη συχνότητα εμφάνισης της κάθε τιμής σε πλήθος μεγαλύτερο από k. Ο αλγόριθμος επαναλαμβάνει τη διαδικασία για όλες τις προβληματικές τιμές του δέντρου συχνοτήτων.

Ακολουθεί ο ψευδοκώδικας του αλγόριθμου που βασίζεται στην apriori ιδιότητα.

Apriori Αλγόριθμος Ανωνυμοποίησης

```
    AA (D, I, k, m)
1:  αρχικοποίηση δέντρου ιεραρχίας γενίκευσης
2:  για ι:=1 μέχρι m                    ⇒ για όλα τα μεγέθη εγγραφών
3:      δημιούργησε νέο δέντρο συχνοτήτων count-tree
4:      για όλες τις εγγραφές t ∈ D
5:          ενημέρωσε το count-tree με όλους τους συνδυασμούς μεγέθους i της εγγραφής
6:          για όλα τα φύλλα v του δέντρου συχνοτήτων
7:          εάν το support(v) < k
8:              βρες γενίκευση στο δέντρο ιεραρχίας τέτοια ώστε support(v)>=k
9:              ανωνυμοποίησε τα δεδομένα και ενημέρωσε δέντρο συχνοτήτων
```

3

Ορισμός Προβλήματος

Στις μέρες μας πολλοί δημόσιοι φορείς και επιχειρήσεις συλλέγουν προσωπικά δεδομένα με σκοπό την αξιοποίηση της πληροφορίας που περικλείουν, στοχεύοντας είτε να αυξήσουν το κέρδος τους, είτε να βελτιώσουν την εξυπηρέτηση του κοινού, ή ακόμα για σκοπούς μελέτης και έρευνας. Με την αυξανόμενη χρήση του διαδικτύου και με τα σύγχρονα τεχνολογικά μέσα, τα συγκεκριμένα δεδομένα μπορούν να φτάσουν στα χέρια οποιουδήποτε επιχειρήσει να έχει πρόσβαση σε αυτά, προκειμένου να τα χρησιμοποιήσει κακόβουλα. Ο έλεγχος της νομιμότητας κατοχής και της διαχείρισης τέτοιου είδους δεδομένων είναι πολύ δύσκολο να ελεγχθεί, με αποτέλεσμα να απειλείται η ιδιωτικότητα των ατόμων που συμμετέχουν σε αυτά.

Σύνολα δεδομένων όπως είναι η φορολογική βάση ενός κράτους, ή το πληροφοριακό σύστημα των ιατρικών δεδομένων των πολιτών, μπορούν να χρησιμοποιηθούν κακόβουλα από επιτιθέμενους, που σε συνδυασμό με χρήση εξωτερικής πληροφορίας να μπορέσουν να αποκαλύψουν πολλές ιδιωτικές πληροφορίες για τα άτομα που περιλαμβάνονται στη βάση. Ακόμα και με την αφαίρεση μοναδικών αναγνωριστικών όπως είναι το Ονοματεπώνυμο, ο Αριθμός Φορολογικού Μητρώου ή ο Αριθμός Δελτίου Ταυτότητας, ο επιτιθέμενος μπορεί να συνδυάσει πληροφορίες από άλλα δημοσιευμένα σύνολα, όπως για παράδειγμα ένα τηλεφωνικό κατάλογο και να μπορέσει να προσδιορίσει μοναδικά ένα φυσικό πρόσωπο που περιλαμβάνεται στη βάση δεδομένων.

Στο Κεφάλαιο 2, μελετήθηκαν αλγόριθμοι ανωνυμοποίησης και εγγυήσεις ιδιωτικότητας που μπορούν να προστατεύσουν ένα σύνολο δημοσιευμένων δεδομένων από τέτοιους είδους κακόβουλες επιθέσεις.

Παρ' όλες όμως, τις τεχνικές που έχουν αναπτυχθεί, πολλοί κίνδυνοι για την ιδιωτικότητα παραμένουν χωρίς αποτελεσματική αντιμετώπιση. Αυτό οφείλεται στο γεγονός ότι, η διαθέσιμη πληροφορία που μπορεί να κατέχει ο επιτιθέμενος μπορεί να έχει πολλές μορφές και να προέρχεται από παράλληλες δημοσιεύσεις σε πολλές πηγές. Παράλληλα, τα μοντέλα των δημοσιευμένων δεδομένων κάθε φορά μπορεί να διαφέρουν, με αποτέλεσμα η κάθε περίπτωση δημοσίευσης δεδομένων να απαιτεί διαφορετική επεξεργασία προκειμένου να εξασφαλίζεται η ιδιωτικότητα των ατόμων που βρίσκονται σε αυτή. Η παρούσα διπλωματική εργασία επιχειρεί να λύσει ένα πρόβλημα ιδιωτικότητας στο οποίο ο επιτιθέμενος κατέχει μερική γνώση των γνωρισμάτων μιας εγγραφής και προσπαθεί, σε συνδυασμό με εξωτερική πληροφορία, να προσδιορίσει μοναδικά ένα φυσικό πρόσωπο. Το συγκεκριμένο μοντέλο βάσης δεδομένων χρησιμοποιεί συνεχή γνωρίσματα από ένα κοινό πεδίο τιμών που αναπαριστούν το ίδιο είδος πληροφορίας. Δεν υπάρχει ένα συγκεκριμένο σύνολο ευαίσθητων γνωρισμάτων στη βάση δεδομένων. Σε κάθε περίπτωση η μερική γνώση του επιτιθέμενου θεωρείται το σύνολο του ψευδο-αναγνωριστικού, ενώ τα υπόλοιπα γνωρίσματα αφορούν την ευαίσθητη πληροφορία.

Η καταγραφή των ετήσιων εισοδημάτων κάθε ατόμου είναι χαρακτηριστικό παράδειγμα του συγκεκριμένου μοντέλου βάσης δεδομένων. Τα δεδομένα αυτά μέσω της ανάρτησης δεδομένων καταγραφής προηγούμενων ετών, διατίθενται δημόσια για ερευνητικούς σκοπούς, με αποτέλεσμα να μπορούν να ανακτηθούν από οποιονδήποτε επιθυμεί να τα χρησιμοποιήσει κακόβουλα. Ο επιτιθέμενος, γνωρίζοντας ένα μέρος των εσόδων ενός ατόμου και με συνδυασμό εξωτερικής πληροφορίας μπορεί να προσδιορίσει μοναδικά μια εγγραφή και να βρει και τα υπόλοιπα έσοδα που αντιστοιχούν στην εγγραφή αυτή.

Η τροποποίηση των δεδομένων μέσω της k^m-ανωνυμοποίησης, όπως έχει οριστεί από [TMK08], μπορεί μέσω της διαδικασίας γενίκευσης με τη χρήση προκαθορισμένης ιεραρχίας, να προστατέψει το σύνολο των δεδομένων από τέτοιου είδους επιθέσεις. Ο επιτιθέμενος δεν θα μπορεί να προσδιορίσει μοναδικά κάποια εγγραφή, για οποιοδήποτε σύνολο μερικής γνώση κατέχει πάνω στα δημοσιευμένα δεδομένα. Ο αλγόριθμος όμως δεν εκμεταλλεύεται πλήρως το μοντέλο δεδομένων που περιγράφτηκε πιο πάνω, με αποτέλεσμα αρκετές φορές κατά τη διαδικασία της γενίκευσης να γενικεύονται τιμές σε υψηλά επίπεδα ιεραρχίας, χωρίς αυτό να είναι απαραίτητο. Εφόσον το μοντέλο δεδομένων του προβλήματος αφορά σε σύνολα δεδομένων με συνεχή γνωρίσματα, η χρήση του δέντρου ιεραρχίας είναι αχρείαστη γιατί σε κάθε περίπτωση γενίκευσης αρκεί οι τιμές να γενικευτούν σε πιο μεγάλα διαστήματα τιμών που περιέχουν τις αρχικές.

Με τον τρόπο αυτό αποτρέπεται η παραβίαση της ιδιωτικότητας των εγγραφών από επιθέσεις αυτής της μορφής μιας και ο επιτιθέμενος δεν μπορεί να προσδιορίσει μοναδικά κάποια εγγραφή και κατά συνέπεια κάποια τιμή που αυτή λαμβάνει. Παράλληλα διατηρείται σημαντικά μεγαλύτερο ποσοστό χρήσιμης πληροφορίας στα δημοσιευμένα δεδομένα. Στη λογική αυτή βασίζεται ο προτεινόμενος αλγόριθμος με στόχο την επίλυση του προβλήματος που ορίζεται στη συνέχεια.

3.1 Μοντέλο δεδομένων

Η παρούσα διπλωματική εργασία εστιάζει σε βάσεις δεδομένων με αριθμητικά γνωρίσματα. Σε κάθε περίπτωση η μερική γνώση του επιτιθέμενου θεωρείται το σύνολο του ψευδο-αναγνωριστικού, ενώ τα υπόλοιπα γνωρίσματα αφορούν την ευαίσθητη πληροφορία.

Το εξεταζόμενο μοντέλο αφορά μια βάση δεδομένων D με ένα σύνολο $|D|$ εγγραφών. Κάθε εγγραφή $t \in D$ αποτελείται από ένα σύνολο τιμών V και παίρνει τιμές από το κοινό πεδίο τιμών $I \subseteq \mathbb{R}$. Το μέγεθος $n=|V|$ της κάθε εγγραφής στη βάση είναι μεταβλητό.

Έστω ένα σύνολο $PK(i_1, i_2, .., i_m)$ υποσύνολο του V, για το οποίο ισχύει ότι $|PK| = m$. Το υποσύνολο αυτό, αναπαριστά τη μερική γνώση που θεωρείται πως κατέχει ο επιτιθέμενος πάνω στις τιμές κάποιας εγγραφής. Το σύνολο αυτό είναι και το σύνολο του ψευδο-αναγνωριστικού. Με τη χρήση αυτών, ορίζεται το σύνολο $SI(i_1, i_2, .., i_{n-m})$, για το οποίο ισχύει ότι $SI = V \backslash PK \; και \; |SI| = n - m$. Το υποσύνολο αυτό, αναπαριστά την ευαίσθητη πληροφορία που προσπαθεί να προσδιορίσει ο επιτιθέμενος σε κάθε περίπτωση.

Η χρήση βάσεων δεδομένων που αντιπροσωπεύουν το παραπάνω μοντέλο συναντάται συχνά στην καθημερινότητα. Εμφανίζεται για να περιγράψει τα φορολογικά στοιχεία των ατόμων που συμμετέχουν στα δεδομένα, τα ποσά από τις καθημερινές οικονομικές τους συναλλαγές ή ακόμα και βάσεις δεδομένων με λεπτομερή στοιχεία από τις αγορές ενός ατόμου από ένα ηλεκτρονικό κατάστημα στο Διαδίκτυο. Το παράδειγμα του Πίνακα 3.1 είναι μία περίπτωση αυτού του μοντέλου δεδομένων.

A/A	Σύνολο τιμών
1	{6490, 2030, 2300, 4002, 2400}
2	{4100, 5300, 7400,230,1262}
3	{1002, 1100, 10400, 400}
4	{1232, 10000, 2300, 430, 1200, 860}
5	{120, 5400, 230, 7340, 100}
6	{1000}

Πίνακας 3.1 Μοντέλο δεδομένων

3.2 Απειλές κατά της ιδιωτικότητας

Σε βάσεις δεδομένων που ακολουθούν το συγκεκριμένο μοντέλο, μπορεί να επιχειρηθεί η παραβίαση της ιδιωτικότητας των ατόμων που συμμετέχουν στη βάση αυτή, με στόχο την αναγνώριση της ταυτότητάς τους. Στο δεδομένο πρόβλημα, ο επιτιθέμενος έχει πληροφορία για κάποιες από τις τιμές των γνωρισμάτων που εμφανίζονται στη βάση για ένα άτομο. Συνδυάζοντας τα στοιχεία που έχει και εκείνα που του παρέχονται από τη δημοσίευση της βάσης, προσπαθεί να ταυτοποιήσει κάποια συγκεκριμένη εγγραφή με το άτομο αυτό. Στο παράδειγμα του Πίνακα 3.1 ο επιτιθέμενος μπορεί να γνωρίζει ότι κάποια από τα εισοδήματα ενός ατόμου είναι {1002, 1100} και συνεπώς να το αναγνωρίσει ως την εγγραφή 4.

Ο επιτιθέμενος μπορεί να καταλήξει με βεβαιότητα σε μια μοναδική εγγραφή του συνόλου των δημοσιευμένων δεδομένων, όταν δεν υπάρχουν άλλες εγγραφές με το ίδιο σύνολο τιμών στα επιμέρους εισοδήματα.

3.3 Μετρική Κόστους Απώλειας Πληροφορίας

Προκειμένου να εκτιμηθεί η αποδοτικότητα ενός αλγόριθμου ανωνυμοποίησης, χρησιμοποιείται σαν παράμετρος σύγκρισης η απώλεια πληροφορίας που παρατηρείται στα δημοσιευμένα δεδομένα. Στην παρούσα εργασία, για να συγκριθούν οι διάφορες μέθοδοι ανωνυμοποίησης ως προς την απώλεια πληροφορίας των ανωνυμοποιημένων δεδομένων, χρησιμοποιείται η *Κανονικοποιημένη Ποινή Βεβαιότητας (Normalized Certainty Penalty)* όπως ορίζεται από [XWP+06].

Συγκεκριμένα στο προς εξέταση μοντέλο, κάθε εγγραφή στο σύνολο δεδομένων αποτελείται από τιμές της μορφής:

$$v = (v_1, v_2, \ldots, v_n)\ , \quad v_i \in \mathbb{R} \ \ \forall\ i \in [1, n]$$

και γενικεύεται σε τιμές της μορφής:

$$v = ([y_1, z_1], [y_2, z_2], \ldots, [y_n, z_n]), \quad y_i \le v_i \le z_i\ \forall\ i\ \in [1, n].$$

Η Κανονικοποιημένη Ποινή Βεβαιότητας (NCP) για κάθε γενίκευση ορίζεται σαν:

$$NCP([yi, zi]) = \begin{cases} 0, & z_i = y_i \\ \dfrac{z_i - y_i}{|I|}, & z_i \neq y_i \end{cases}$$

όπου $|I|$ το μέγεθος του πεδίου τιμών *I* των γνωρισμάτων της κάθε εγγραφής.

Βάσει αυτού, προκύπτει η Κανονικοποιημένη Ποινή Βεβαιότητας όλης της βάσης D:

$$NCP(D) = \frac{\sum_{vi \in D}(C_{v_i} \cdot NPC(v_i))}{\sum_{vi \in D}(C_{v_i})}$$

όπου C_{v_i} η συχνότητα εμφάνισης της κάθε τιμής v_i στη βάση δεδομένων.

Η Κανονικοποιημένη Ποινή Βεβαιότητας αποτελεί ένα πολύ καλό εργαλείο σύγκρισης για την εκτίμηση της πληροφορίας που χάνεται από τις αρχικές τιμές μετά από τη διαδικασία της γενίκευσης που χρησιμοποιούν οι αλγόριθμοι ανωνυμοποίησης. Με τη χρήση αυτής μπορεί να επιλεγεί ο αλγόριθμος που βέλτιστα ικανοποιεί την ανωνυμία, ενώ παράλληλα διατηρεί το μεγαλύτερο δυνατό ποσοστό χρήσιμης πληροφορίας από τα αρχικά δεδομένα.

Ένα παράδειγμα χρήσης της Κανονικοποιημένης Ποινής Βεβαιότητας φαίνεται στο παράδειγμα του Πίνακα 3.2.

A/A	Σύνολο τιμών
1	{100, 100, 200, 400, 400}
2	{100, 300, 400}
3	{100, 100, 400, 400}

Πίνακας 3.2

Αν ο αλγόριθμος προβεί σε μια γενίκευση της μορφής {200, 300} → [200, 300], τότε θα προκύψει ο παρακάτω πίνακας

A/A	Σύνολο τιμών
1	{100, 100, [200,300], 400, 400}
2	{100, [200,300], 400}
3	{100, 100, 400, 400}

Πίνακας 3.3

Η Κανονικοποιημένη Ποινή Βεβαιότητας (NCP) για τη συγκεκριμένη γενίκευση υπολογίζεται σαν:

$$NCP([200,300]) = \frac{(300 - 200 + 1)}{400 - 100 + 1} = \frac{101}{301} = 0.33$$

Για τις υπόλοιπες τιμές της βάσης, το NCP ισούται με μηδέν αφού διατηρούν τις αρχικές τους τιμές. Κατ' επέκταση η Κανονικοποιημένη Ποινή Βεβαιότητας όλης της βάσης D υπολογίζεται σαν:

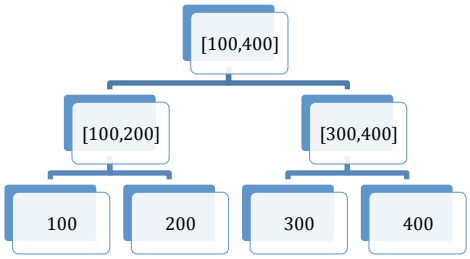
$$NCP(D) = \frac{0.33 \cdot 2}{12} = 0.055$$

3.4 Πιθανές Λύσεις

Για την προστασία ιδιωτικότητας μιας βάσης δεδομένων με συνεχή γνωρίσματα, επιχειρείται η δημοσίευση ανωνυμοποιημένων δεδομένων όσο πιο κοντά στην αρχική τιμή, χωρίς όμως ο επιτιθέμενος να μπορεί να προσδιορίσει μοναδικά μια εγγραφή με τη μερική γνώση που κατέχει. Όπως περιγράφηκε και στο Κεφάλαιο 2, η εφαρμογή k^m-ανωνυμοποίησης με χρήση ιεραρχιών γενίκευσης θα μπορούσε να χρησιμοποιηθεί για να επιλύσει το πρόβλημα. Η συγκεκριμένη τεχνική όμως δεν εκμεταλλεύεται το γεγονός ότι η βάση δεδομένων αποτελείται από αριθμητικά δεδομένα, με αποτέλεσμα να παρουσιάζεται μεγάλη απώλεια πληροφορίας.

3.4.1 Χρήση k^m-ανωνυμοποίησης με ιεραρχία γενίκευσης

Με τη χρήση της k^m-ανωνυμοποίησης με ιεραρχία γενίκευσης ο αλγόριθμος θα δημιουργούσε για τον Πίνακα 3.2 ένα δέντρο ιεραρχίας με την πιο κάτω μορφή.



Σχήμα 3.1 Ιεραρχία γενίκευσης για σύνολο δεδομένων πίνακα 3.2

Κατά τη διαδικασία της ανωνυμοποίησης για $m=2$ και $k=2$, ο αλγόριθμος θα έπρεπε να γενικεύσει τις τιμές {200, 300}, οι οποίες παραβιάζουν την ιδιωτικότητα στη βάση δεδομένων. Η τιμή {200} σύμφωνα με το δέντρο ιεραρχίας, γενικεύεται σε [100, 200]. Λόγω του ότι ο αλγόριθμος χρησιμοποιεί τεχνική ολικής ανακωδικοποίησης στα δεδομένα, όλες οι τιμές {100, 200} θα πρέπει να αντικατασταθούν με την γενικευμένη τιμή [100, 200]. Αυτό έχει σαν αποτέλεσμα να ανωνυμοποιηθεί η τιμή {100} η οποία δεν προκαλούσε στη βάση οποιαδήποτε παράβαση ιδιωτικότητας. Με την ίδια λογική η τιμή {300} γενικεύεται σε [300, 400] επηρεάζοντας και την τιμή {400}. Με το πέρας του αλγόριθμου k^m-ανωνυμοποίησης θα προκύψει το πιο κάτω ανωνυμοποιημένο σύνολο δεδομένων:

Α/Α	Γενικευμένο Σύνολο τιμών
1	{[100,200] , [100,200] , [100,200], [300,400], [300,400]}
2	{[100,200] , [300,400] , [300,400]}
3	{[100,200] , [100,200], [300,400], [300,400]}

Πίνακας 3.4 Σύνολο ανωνυμοποιημένων δεδομένων

Στην περίπτωση αυτή ο επιτιθέμενος δε μπορεί να αναγνωρίσει με βεβαιότητα καμία εγγραφή βάσει της μερικής του γνώσης. Ο αλγόριθμος της κλασσικής k^m-ανωνυμίας εξαιτίας της γενίκευσης πλήρους πεδίου με χρήση ιεραρχίας που εφαρμόζει, ικανοποιεί την k-ανωνυμία στο σύνολο των δεδομένων, όμως υπεργενικεύει τα δεδομένα. Αυτό έχει ως αποτέλεσμα την σημαντική απώλεια χρήσιμης πληροφορίας κατά την δημοσίευσή τους, χωρίς αυτό να είναι απαραίτητο.

Με τη χρήση της Κανονικοποιημένης Ποινής Βεβαιότητας προκύπτει:

$$NCP([100,200]) = \frac{(200 - 100 + 1)}{400 - 100 + 1} = \frac{101}{301} = 0.33$$

$$NCP([300,400]) = \frac{(400 - 300 + 1)}{400 - 100 + 1} = \frac{101}{301} = 0.33$$

$$NCP(D) = \frac{0.33 \cdot 6 + 0.33 \cdot 6}{12} = 0.33$$

3.4.2 Χρήση αλγόριθμου χωρίς ιεραρχίες γενίκευσης

Στην περίπτωση της δημοσίευσης δεδομένων με συνεχή γνωρίσματα, είναι δυνατή η k^m-ανωνυμοποίηση με σημαντικά καλύτερη απόδοση σύμφωνα με τον αλγόριθμο που εξετάζεται στην παρούσα εργασία. Η πιθανή προτεινόμενη λύση εφαρμόζει ολική γενίκευση με σκοπό την k^m-ανωνυμοποίηση των δεδομένων, όπως ακριβώς και στο πιο πάνω παράδειγμα. Η διαφορά με την προηγούμενη λύση, βρίσκεται στο ότι η κατάλληλη γενίκευση για κάθε γνώρισμα δεν ακολουθεί κάποια προκαθορισμένη ιεραρχία. Ο αλγόριθμος εκμεταλλευόμενος την φύση των δεδομένων προσπαθεί κάθε φορά να βρει τη λύση διευρύνοντας το διάστημα τιμών κάθε γνωρίσματος, με τέτοιο τρόπο ώστε να διατηρείται η Κανονικοποιημένη Ποινή Βεβαιότητας μικρότερη από μια μέγιστη επιτρεπτή ποινή. Η εφαρμογή του αλγορίθμου για το ίδιο σύνολο δεδομένων παρουσιάζεται στον Πίνακα 3.5.

A/A	Γενικευμένο Σύνολο τιμών
1	{100 , 100 , [200,300] , 400,400}
2	{100 , 400 , [200,300]}
3	{100, 100, 400, 400}

Πίνακας 3.5 Σύνολο ανωνυμοποιημένων δεδομένων

Κατά τη διαδικασία της ανωνυμοποίησης για $m=2$ και $k=2$, ο αλγόριθμος θα πρέπει να γενικεύσει τις τιμές {200,300}, οι οποίες παραβιάζουν την ιδιωτικότητα στη βάση δεδομένων. Ο αλγόριθμος ελέγχοντας τις πιθανές γενικεύσεις βρίσκει ότι, η τιμή {200}

μπορεί να γενικευτεί σε [200,300] διατηρώντας την Κανονικοποιημένη Ποινή Βεβαιότητας σε χαμηλά επίπεδα, μικρότερη κάθε φορά από μια παράμετρο μέγιστης επιτρεπτής ποινής.

Με τη χρήση και πάλι της Κανονικοποιημένης Ποινής Βεβαιότητας προκύπτει:

$$NCP([200,300]) = \frac{(300 - 200 + 1)}{400 - 100 + 1} = \frac{101}{301} = 0.33$$

$$NCP(D) = \frac{0.33 \cdot 2}{12} = 0.055$$

Στην περίπτωση της λύσης αυτής, ο επιτιθέμενος και πάλι δε μπορεί να αναγνωρίσει με βεβαιότητα καμία εγγραφή βάσει της μερικής του γνώσης. Ο αλγόριθμος εξαιτίας της γενίκευσης χωρίς χρήση ιεραρχίας, ικανοποιεί την *k*-ανωνυμία στο σύνολο των δεδομένων, χωρίς να υπεργενικεύει τα δεδομένα. Αυτό έχει σαν αποτέλεσμα, οι γενικευμένες τιμές να διατηρούνται κοντά στις αρχικές και κατ' επέκταση το ανωνυμοποιημένο σύνολο δεδομένων να εμφανίζει μικρή απώλεια πληροφορίας.

3.4.2.1 Παράμετρος Μέγιστης Διαφοροποίησης NCP

Από τον αλγόριθμο της εργασίας δίνεται επιπλέον η επιλογή της μέγιστης διαφοροποίησης της ποινής βεβαιότητας από μια γενίκευση σε μια άλλη, κάτι που δεν εμφανίζεται στους υπόλοιπους αλγορίθμους. Η επιλογή αυτή γίνεται μέσω μίας παραμέτρου η οποία αναπαριστά τη μέγιστη επιτρεπτή διαφοροποίηση της Κανονικοποιημένη Ποινή Βεβαιότητας κατά την διάρκεια εκτέλεσης του αλγόριθμου. Αυτή η τιμή κυμαίνεται από 0 μέχρι 1 όπου για NCP(D)=0 δεν υπάρχει απώλεια πληροφορίας στη βάση δεδομένων αφού δεν επιτρέπεται καμία γενίκευση και για NCP(D)=1, παρουσιάζεται η μέγιστη δυνατή απώλεια πληροφορίας που μπορεί να υπάρξει στη βάση. Στη δεύτερη περίπτωση όλες οι τιμές της βάσης γενικεύονται στην ίδια τιμή.

4

Περιγραφή αλγόριθμου

4.1 Θεωρητικό υπόβαθρο

Στην παρούσα διπλωματική εργασία αναπτύσσεται αλγόριθμος ο οποίος επιχειρεί την ικανοποίηση της k^m-ανωνυμίας δημοσιευμένων συλλογών δεδομένων που αποτελούνται από συνεχή γνωρίσματα. Ο προτεινόμενος αλγόριθμος εκμεταλλεύεται την apriori ιδιότητα όπως αυτή παρουσιάστηκε στο Κεφάλαιο 2 και περιγράφεται αναλυτικά στο [TMK08]. Εφαρμόζεται ολική ανακωδικοποίηση στις τιμές των γνωρισμάτων των εγγραφών, με τη χρήση της τεχνικής της γενίκευσης.

Η k^m-*ανωνυμία,* απαιτεί κάθε εγγραφή που εμφανίζεται στο σύνολο δεδομένων, να μην μπορεί να αναγνωρισθεί ανάμεσα από τουλάχιστον άλλες *k-1* εγγραφές του συνόλου, ακόμα και αν ο επιτιθέμενος γνωρίζει *m* τιμές μιας συγκεκριμένης εγγραφής. Αυτό επιτυγχάνεται αν *k* τουλάχιστον εγγραφές από το σύνολο, εμφανίζουν τις ίδιες τιμές ή αντίστοιχα τα ίδια διαστήματα τιμών σε *m* διαφορετικά γνωρίσματα. Ο αλγόριθμος που παρουσιάζεται εξασφαλίζει την k^m-ανωνυμία.

Η είσοδος του αλγορίθμου είναι ένα σύνολο δεδομένων D από εγγραφές μεταβλητού μήκους, τα στοιχεία των οποίων λαμβάνουν τιμές από ένα κοινό πεδίο τιμών *I,* η παράμετρος ανωνυμίας *k*, η παράμετρος μερικής γνώσης του επιτιθέμενου *m* και η παράμετρος *d*, μια μεταβλητή που αντιπροσωπεύει την μέγιστη επιτρεπτή Κανονικοποιημένη Ποινή Βεβαιότητας NCP(D) της βάσης.

4.2 Δέντρο Συχνοτήτων (Count Tree)

Κάθε φορά που ο αλγόριθμος ελέγχει για μια πιθανή γενίκευση, πρέπει να είναι σε θέση να μετρά αποδοτικά τις εμφανίσεις όλων των m-συνδυασμών των στοιχείων που περιλαμβάνονται στη βάση δεδομένων, και πώς αυτές οι εμφανίσεις επηρεάζονται με την εφαρμογή της εκάστοτε γενίκευσης. Για να αποφεύγεται η σάρωση της βάσης δεδομένων κάθε φορά που ο αλγόριθμος ελέγχει για ανακωδικοποιήσεις προτείνεται στη βιβλιογραφία [TMK08] μια ειδικά κατασκευασμένη δομή δεδομένων σε μορφή δέντρου, στην οποία αποθηκεύονται όλοι οι δυνατοί συνδυασμοί m-στοιχείων της βάσης μαζί με το πλήθος των εμφανίσεων τους.

Κρατώντας ενημερωμένο το δέντρο συχνοτήτων (count-tree) κάθε φορά που επιχειρείται μια γενίκευση αποφεύγονται οι αχρείαστες σαρώσεις της βάσης δεδομένων. Εκτός αυτού, ο αλγόριθμος μπορεί μόνο με τη χρήση του δέντρου να αποφασίσει από τις εμφανίσεις του κάθε συνδυασμού, αν υπάρχει παραβίαση ιδιωτικότητας για κάποια εγγραφή. Κάθε κόμβος στο δέντρο συχνοτήτων είναι ταξινομημένος ανάλογα με το πλήθος εμφανίσεων του κάθε συνδυασμού.

Στην παρούσα εργασία χρησιμοποιούμε την συγκεκριμένη δομή δεδομένων προκειμένου να κάνουμε τον αλγόριθμο πιο αποδοτικό. Κάθε κόμβος του δέντρου έχει αποθηκευμένα:

1. Την κάθε τιμή ή το κάθε γενικευμένο διάστημα τιμών της βάσης δεδομένων
2. Το πλήθος των διαφορετικών εγγραφών που η τιμή αυτή συναντάται
3. Τον αύξοντα αριθμό των διαφορετικών εγγραφών

Πιο κάτω φαίνεται ο αλγόριθμος δημιουργίας του δέντρου σε ψευδοκώδικα.

Αλγόριθμος για δημιουργία δέντρου συχνοτήτων

Είσοδος: Σύνολο δεδομένων ***D*** με ***|D|*** εγγραφές
m παράμετρος μερικής γνώσης

Έξοδος: Δέντρο συχνοτήτων (*ctree*)

Βήματα αλγόριθμου

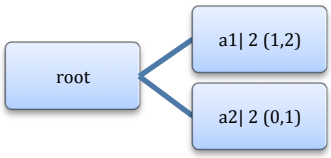
1: Για *i:=1* μέχρι *m*
2: **Για κάθε** εγγραφή $t_d=\{v_1,v_2,..,v_n\} \in D$ $\Rightarrow$ για κάθε μέγεθος εγγραφών ***n***
3: Υπολόγισε όλους τους $\binom{n}{i}$- διαφορετικούς συνδυασμούς ***cmb*** της εγγραφής
4: **Για κάθε** συνδυασμό $cmb=\{v_1,v_2,..,v_{i-1},v_i\}$
5: Βρες το μονοπάτι $\{v_1,v_2,..,v_{i-1}\}$ στο δέντρο *ctree*
6: **Εάν** ο κόμβος v_i είναι παιδί-φύλλο στο μονοπάτι
7: Αύξησε πλήθος v_i κατά 1
Πρόσθεσε τον αύξοντα αριθμό ***d*** στους αριθμούς των εγγραφών
8: **Αλλιώς** πρόσθεσε τον κόμβο v_i σαν παιδί φύλλο του μονοπατιού

Για την καλύτερη κατανόηση του αλγόριθμου, χρησιμοποιείται το παράδειγμα του πιο κάτω πίνακα, όπου επιχειρείται η *2*-ανωνυμοποίηση, όταν ο επιτιθέμενος γνωρίζει το πολύ 2 τιμές από το σύνολο τιμών μιας εγγραφής.

Α/Α	ΣΥΝΟΛΟ ΤΙΜΩΝ
0	{a2, a2}
1	{a1, a1, a2, a2}
2	{a1}

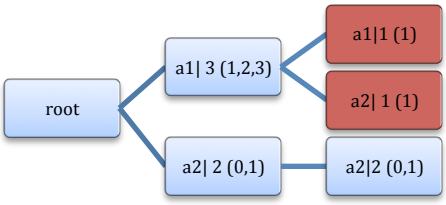
Πίνακας 4.1 Σύνολο δεδομένων

Αρχικά ο αλγόριθμος θα πάρει όλους τους συνδυασμούς τιμών που μπορούν να γίνουν στη βάση με μέγεθος m=1 στοιχείο, $\{a_1\}$ και $\{a_2\}$. Σαρώνει τη βάση και περνά τους συνδυασμούς στο count tree, μαζί με το πλήθος των διαφορετικών εγγραφών που αυτοί εμφανίζονται και τους αύξοντες αριθμούς των εγγραφών αυτών.



Σχήμα 4.1 Δέντρο συχνοτήτων για m=1

Στη συνέχεια ο αλγόριθμος παίρνει όλους τους συνδυασμούς τιμών που μπορούν να γίνουν στη βάση με μέγεθος m=2 στοιχεία, $\{a_1,a_1\}$, $\{a_2,a_2\}$ και $\{a_1,a_2\}$. Σαρώνει τη βάση και περνά τους συνδυασμούς στο δέντρο, μαζί με το πλήθος των διαφορετικών εγγραφών που αυτοί εμφανίζονται και τους αύξοντες αριθμούς των εγγραφών αυτών.



Σχήμα 4.2 Δέντρο συχνοτήτων για m=2

Παρ' όλο που ο συνδυασμός τιμών {a1,a2} εμφανίζεται δύο φορές στην εγγραφή 1, το πλήθος εμφανίσεων του στο δέντρο συχνοτήτων είναι ένα, γιατί το δέντρο αποθηκεύει το πλήθος των διαφορετικών εγγραφών που οι τιμές εμφανίζονται.

4.2.1 Συγχώνευση στο δέντρο συχνοτήτων με αύξοντες αριθμούς εγγραφών

Με τη χρήση της συγκεκριμένης δομής, ο αλγόριθμος δεν κρατά μόνο πληροφορίες για τους συνδυασμούς τιμών που μπορούν να γίνουν στη βάση, αλλά βρίσκει και ποιοι συνδυασμοί τιμών παραβιάζουν την ιδιωτικότητα, χωρίς να χρειαστεί να σαρώσει τη βάση παραπάνω φορές. Στο παράδειγμα οι συνδυασμοί {a1,a2} και {a1,a1} παραβιάζουν την ιδιωτικότητα αφού έχουν πλήθος εμφάνισης μικρότερο από 2, ο καθένας.

Ο αλγόριθμος της k^m-ανωνυμίας θα ψάξει στο δέντρο να βρει πιθανές γενικεύσεις. Μια πιθανή γενίκευση στο συγκεκριμένο παράδειγμα είναι η *{a1,a2}→ {A}*. Στη συγκεκριμένη περίπτωση, ο αλγόριθμος μπορεί - χωρίς να σαρώσει τη βάση δεδομένων – να κρίνει ότι η γενίκευση αυτή δεν λύνει το πρόβλημα ιδιωτικότητας γιατί, με τη συγχώνευση των κόμβων *{a1|1 (1)}* και *{a2|1 (1)}* το πλήθος των διαφορετικών εγγραφών συνεχίζει να είναι ίσο με ένα, αφού και οι δύο τιμές εμφανίζονται μόνο στην εγγραφή με αύξοντα αριθμό 1.

Το γεγονός ότι το δέντρο συχνοτήτων αποθηκεύει και τους αύξοντες αριθμούς των διαφορετικών εγγραφών κάνει τον αλγόριθμο ακόμα πιο αποδοτικό, χωρίς να χρειάζεται να ελέγχεται η βάση δεδομένων για τους αύξοντες αριθμούς των διαφορετικών εγγραφών που εμφανίζονται οι συνδυασμοί.

4.3 Γενικεύσεις σε συνεχή γνωρίσματα

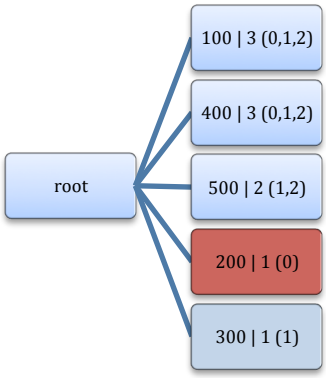
Αυτό που διαφοροποιεί τον αλγόριθμο της παρούσας διπλωματικής εργασίας από τον αλγόριθμο της κλασσικής k^m-ανωνυμίας, είναι ότι εκμεταλλεύεται την φύση των δεδομένων της δημοσιευμένης βάσης, και πιο συγκεκριμένα το γεγονός ότι πρόκειται για συνεχή γνωρίσματα αριθμητικών τιμών.

Το πλεονέκτημα σε τέτοιες βάσεις δεδομένων είναι ότι μπορούμε να πετύχουμε την ελάχιστη δυνατή Κανονικοποιημένη Ποινή Βεβαιότητας (NCP), με πολύ λιγότερους ελέγχους εκμεταλλευόμενοι την εξής ιδιότητα:

Εάν για ένα σύνολο J ισχύει ότι NCP(J)>d τότε, και για οποιοδήποτε υπερσύνολο K του J ισχύει ότι NCP(K)>=NCP(J)>d.

Για την καλύτερη κατανόηση του αλγόριθμου, εξετάζεται το πιο κάτω παράδειγμα, όπου επιχειρείται η *2*-ανωνυμοποίηση, όταν ο επιτιθέμενος γνωρίζει το πολύ 2 τιμές από το σύνολο τιμών μιας εγγραφής. Σε κάποιο από τα βήματα του αλγόριθμου, το δέντρο συχνοτήτων έχει την μορφή του σχήματος 4.3. Προκειμένου να επιλυθεί οποιαδήποτε παράβαση ιδιωτικότητας ο αλγόριθμος επιχειρεί να συνενώσει τους προβληματικούς κόμβους του δέντρου. Από αυτές τις συνενώσεις πρέπει να επιλεγεί αυτή με την μικρότερη ποινή NCP και την λιγότερη

απώλεια πληροφορίας. Στο παράδειγμα ο αλγόριθμος επιχειρεί να γενικεύσει την τιμή {200} σε ένα μεγαλύτερο διάστημα τιμών.



Σχήμα 4.3 Δέντρο συχνοτήτων

Ο αλγόριθμος ταξινομεί τους αδελφικούς κόμβους {200} σε αύξουσα σειρά και βρίσκει όλες τις δυνατές γενικεύσεις που μπορούν να γίνουν, για να λύσουν το πρόβλημα υπολογίζοντας κάθε φορά το NCP(D) της βάσης για κάθε μία από αυτές.

Σχήμα 4.4 Αδερφικοί κόμβοι σε αύξουσα σειρά

ΠΛΗΘΟΣ ΚΟΜΒΩΝ	ΚΟΜΒΟΙ	ΓΕΝΙΚΕΥΣΗ	NCP (D)
2	{100,200}	[100, 200]	0.1
	{200,300}	[200, 300]	0.04
3	{100,200,300}	[100, 300]	0.25
	{200,300,400}	[200, 400]	0.25
4	{100,200,300,400}	[100, 400]	0.64
	{200,300,400,500}	[200, 500]	0.48
5	{100,200,300,400,500}	[100, 500]	1

Πίνακας 4.2 Σύνολο πιθανών γενικεύσεων κόμβου {200}

Είναι φανερό ότι ο αλγόριθμος δεν χρειάζεται να ελέγξει για γενικεύσεις οι οποίες αποτελούν υπερσύνολα άλλων γενικεύσεων που έχουν ήδη ελεγχθεί, γιατί σύμφωνα με την ιδιότητα που περιγράφηκε πιο πάνω θα έχουν σίγουρα μεγαλύτερη ποινή *NCP(D)*.

Στον προτεινόμενο αλγόριθμο αν R μια γενίκευση για την οποία ισχύει ότι *sup(R)>=k*, ο αλγόριθμος κοιτάζει για *NCP(D)*. Αν το *NCP(D)<=d* είναι σίγουρο ότι έχει βρει την πιο

φτηνή αποδεκτή λύση μεταξύ των αδελφικών κόμβων. Αν NCP(D)>d δεν χρειάζεται να ψάξει για μεγαλύτερα όρια τιμών στους συγκεκριμένους αδελφικούς κόμβους γιατί σίγουρα θα έχουν μεγαλύτερη ποινή.

4.4 Υλοποίηση

Ο αλγόριθμος αρχικά, για κάθε εγγραφή $t \in D$ υπολογίζει τις συχνότητες εμφάνισης (αριθμό διαφορετικών εγγραφών) της κάθε τιμής $v_i \in I$, σαρώνοντας την βάση δεδομένων D. Ταξινομεί κάθε εγγραφή t, βάσει των συχνοτήτων εμφάνισης των τιμών της, προκειμένου να φτιάξει το δέντρο συχνοτήτων *(count tree),* όπως έχει ήδη περιγραφεί.

Στη συνέχεια, ο αλγόριθμος εκμεταλλευόμενος την apriori ιδιότητα (Κεφάλαιο 2), ελέγχει για παραβάσεις ιδιωτικότητας στη βάση από συνδυασμούς τιμών μεγέθους i στοιχείων (ξεκινώντας αρχικά για $i=1$), μέχρι να φτάσει στην τελική τιμή m που αντιπροσωπεύει και την πραγματική μερική γνώση του επιτιθέμενου.

Αρχικά, βάσει της ταξινομημένης διάταξης των εγγραφών, δημιουργεί όλους του δυνατούς συνδυασμούς μεγέθους i τιμών που μπορούν να γίνουν σε κάθε εγγραφή. Με βάση αυτούς τους συνδυασμούς δημιουργεί το δέντρο συχνοτήτων *(count tree)*, τοποθετώντας σε κάθε κόμβο του δέντρου τους συνδυασμούς των τιμών, το πλήθος των διαφορετικών εγγραφών που περιέχουν τους συνδυασμούς αυτούς και τους αύξοντες αριθμούς των εγγραφών αυτών.

Ο αλγόριθμος στη συνέχεια, σαρώνει τα φύλλα του δέντρου και βρίσκει ποιοι κόμβοι έχουν συχνότητα εμφάνισης στη βάση, μικρότερη από τη δοσμένη παράμετρο ανωνυμίας k. Αυτοί οι κόμβοι παραβιάζουν την ιδιωτικότητα και τα όρια των τιμών τους πρέπει να διευρυνθούν, προκειμένου να συγχωνευθούν με άλλους κόμβους-φύλλα του δέντρου, έτσι ώστε το πλήθος εμφανίσεων τους στη βάση να γίνει μεγαλύτερο ή τουλάχιστον ίσο με k.

Για κάθε προβληματικό κόμβο L_f, ο αλγόριθμος ελέγχει για πιθανές συγχωνεύσεις του κόμβου, με τους αδελφικούς του κόμβους στο δέντρο $sib(L_f)=\{sib_1,sib_2,..,sib_n\}$, δηλαδή με κόμβους που έχουν τον ίδιο πατέρα με αυτόν.

Σύμφωνα με την ιδιότητα των συνεχών γνωρισμάτων που περιγράφηκε πιο πάνω, ο αλγόριθμος για να εξοικονομήσει χρόνο και να αποφύγει περιττούς ελέγχους ταξινομεί τους αδελφικούς κόμβους $sib(L_f)$ μαζί με τον προβληματικό L_f σε αύξουσα σειρά. Δημιουργεί υποσύνολα των αδερφικών κόμβων με τον προβληματικό, ξεκινώντας αρχικά για μέγεθος ίσο με δύο και ελέγχοντας σταδιακά και για υποσύνολα μεγαλύτερου μεγέθους, με σκοπό να βρει μια πιθανή γενίκευση, η οποία θα εξασφαλίζει την ιδιωτικότητα στη βάση.

Μόλις ο αλγόριθμος εντοπίσει μια πιθανή γενίκευση $R_{leaf} = [v_{min}, v_{max}]$ η οποία ικανοποιεί την k^m-ανωνυμία, σταματά τον έλεγχο και υπολογίζει την Κανονικοποιημένη Ποινή Βεβαιότητας

$NCP(R_{leaf})$ της βάσης. Αν η μεταβολή της ποινής που προκαλείται στη βάση είναι μικρότερη της παραμέτρου d, τότε η γενίκευση είναι αποδεκτή, ο αλγόριθμος τερματίζει τον έλεγχο και προχωρά στην ολική ανακωδικοποίηση των τιμών της βάσης, αντικαθιστώντας όλες τις τιμές v_i που ανήκουν στο όριο R_{leaf} με το νέο διάστημα τιμών.

Διαφορετικά, αν η ποινή $NCP(R_{leaf})$ προκαλεί μεταβολή για τη συγκεκριμένη γενίκευση μεγαλύτερη από την παράμετρο d, τότε ο αλγόριθμος σταματά να ελέγχει τους αδερφικούς κόμβους-φύλλα και προσπαθεί να επιλύσει το πρόβλημα ιδιωτικότητας με συνένωση του προβληματικού κόμβου L_f με άλλους γειτονικούς κόμβους-φύλλα στο δέντρο $adj(L_f)=\{adj_1,adj_2,..adj_n\}$.

Ο αλγόριθμος ταξινομεί σε αύξουσα σειρά τους γειτονικούς κόμβους $adj(L_f)$ μαζί με τον προβληματικό L_f. Με τον ίδιο τρόπο δημιουργεί και πάλι υποσύνολα των γειτονικών κόμβων, ξεκινώντας αρχικά για μέγεθος ίσο με 2, στοχεύοντας να βρει μια γενίκευση, η οποία θα λύνει το πρόβλημα ιδιωτικότητας. Αν η ανωνυμία δεν εξασφαλίζεται, ελέγχει για υποσύνολα μεγαλύτερου μεγέθους.

Μόλις ο αλγόριθμος εντοπίσει πιθανή γενίκευση $R_{leaf}=[v_{adj.min}, v_{adj.max}]$ η οποία ικανοποιεί την k^m-ανωνυμία, σταματά τον έλεγχο και ελέγχει την Κανονικοποιημένη Ποινή Βεβαιότητας $NCP(R_{leaf})$ της βάσης. Αν η μεταβολή της ποινής για τη συγκεκριμένη γενίκευση R_{leaf} είναι μεγαλύτερη από την παράμετρο d, τότε ο αλγόριθμος σταματά να ελέγχει την συγκεκριμένη οικογένεια γειτονικών κόμβων-φύλλων και ψάχνει να λύσει το πρόβλημα ιδιωτικότητας με κόμβους από άλλη οικογένεια γειτονικών φύλλων.

Διαφορετικά, αν η μεταβολή της ποινής για τη συγκεκριμένη γενίκευση είναι μικρότερη της παραμέτρου d, τότε το όριο γενίκευσης R_{leaf} είναι μια πιθανή λύση. Προκείμενου όμως να γίνει αποδεκτή η συγκεκριμένη γενίκευση πρέπει να προηγηθεί η συνένωση των προγόνων του προβληματικού κόμβου L_f με τους προγόνους των γειτονικών κόμβων $adj(L_f)$ στο δέντρο, έτσι ώστε να δημιουργηθεί ένα κοινό μονοπάτι στο δέντρο για τους κόμβους που θα συνενωθούν.

Ο αλγόριθμος, διασχίζοντας αναδρομικά το μονοπάτι των προγόνων του προβληματικού κόμβου $path(L_f)=\{anc_1(L_f), anc_2(L_f),.., anc_n(Lf)\}$ και των προγόνων της οικογένειας των γειτονικών κόμβων $path(adj)=\{anc_1(adj), anc_2(adj),.., anc_n(adj)\}$, ελέγχει κάθε ζεύγος κόμβων $\{anc_i(L_f), anc_i(adj)\}$, που δεν βρίσκονται στο κοινό μονοπάτι και υπολογίζει το κόστος συνένωσης τους $R_{anc,i}=[v_{min.anc}, v_{max.anc}]$ βάσει του $NCP(R_{anc,i})$.

Αν το τελικό κόστος συνένωσης των φύλλων $NCP(R_{leaf})$, και όλων των προγόνων $NCP(R_{anc})$ που σχηματίζουν το κοινό μονοπάτι είναι μικρότερο από d τότε η λύση είναι αποδεκτή, ο αλγόριθμος τερματίζει τον έλεγχο και προχωρά στην ολική γενίκευση των τιμών της βάσης που ανήκουν στα όρια των γενικεύσεων R_{leaf} και R_{anc}. Διαφορετικά, αν το τελικό

κόστος συνένωσης των φύλλων και όλων των προγόνων είναι μεγαλύτερο από το d, η συνένωση δεν είναι αποδεκτή και ο αλγόριθμος ψάχνει για γενίκευση του προβληματικού κόμβου με άλλη οικογένεια γειτονικών κόμβων φύλλων.

Η πιο πάνω διαδικασία επαναλαμβάνεται για όλα τα προβληματικά φύλλα του δέντρου που παρουσιάζουν πλήθος εμφανίσεων μικρότερο από την δοσμένη παράμετρο ανωνυμίας k.

Όταν ο αλγόριθμος επιτύχει τις κατάλληλες γενικεύσεις και επιλύσει το πρόβλημα για αριθμό i συνδυασμών, επαναλαμβάνει εκ νέου την διαδικασία σύμφωνα με τον apriori αλγόριθμο για μέγεθος συνδυασμών $i+1$.

Πιο κάτω παρουσιάζεται ο αλγόριθμος σε μορφή ψευδοκώδικα.

Αλγόριθμος km-ανωνυμοποίησης συνεχών γνωρισμάτων

Είσοδος:	Σύνολο δεδομένων D με $\|D\|$ εγγραφές
	κάθε εγγραφή $t \in D$ *αποτελείται από σύνολο τιμών* $V=\{v_1, v_2,..,v_n\}, V \in I$
	k παράμετρος ανωνυμίας
	m παράμετρος μερικής γνώσης επιτιθέμενου
	d παράμετρος μέγιστης μεταβολής ποινής $NCP(D)$
Έξοδος:	*Ανωνυμοποιημένο σύνολο* D'

Βήματα Αλγόριθμου

1: Βρες συχνότητες εμφάνισης $sup(v_i)$ κάθε τιμής v_i του πεδίου τιμών I
2: Ταξινόμησε κάθε εγγραφή $t \in D$ σύμφωνα με τις συχνότητες εμφάνισης των τιμών της
3: **Για** *i:=1* μέχρι *m* {
4: **Για κάθε** εγγραφή $t=\{v_1,v_2,..,v_n\} \in D$ { ⇒ για κάθε μέγεθος εγγραφών n
5: Υπολόγισε όλους τους $\binom{n}{i}$-συνδυασμούς τιμών cmb της εγγραφής
6: Ενημέρωσε *count tree* με τα supports $sup(cmb)$ των συνδυασμών
7: }
8: **Για κάθε** κόμβο-φύλλο L_f στο *count tree* {
9: **Εάν** η συχνότητα εμφάνισης $sup\ (L_f) < k$, τότε L_f προβληματικός κόμβος:
10: *//Η τιμή v_i του κόμβου L_f πρέπει να γενικευτεί σε μεγαλύτερο εύρος τιμών*
11: Βάλε στα Nodes τις τιμές των αδερφικών κόμβων $Nodes=sib(L_f) + L_f$
12: **Όσο** δεν έχει βρεθεί αποδεκτό όριο τιμών {
13: Ταξινόμησε σε αύξουσα σειρά τα $Nodes=\{v_1,v_2,..,v_i,..,v_n\}$
14: Βρες $R_{leaf}=[v_{sib.min}, v_{sib.max}]$ με το πιο μικρό $NCP(R_{leaf})$, με $sup(L_f)>=k$
15: **Εάν** η μεταβολή της ποινή του ορίου $NCP\ (R_{leaf})<d$:
16: Το όριο $R_{leaf} =[v_{sib.min},v_{sib.max}]$ είναι αποδεκτό

17: Τερμάτισε το βρόγχο

18: **Αλλιώς**:

19: **Όσο** υπάρχουν γειτονικοί κόμβοι-φύλλα $adj(L_f)$του L_f {

20: Βάλε στο Nodes τιμές γειτονικών κόμβων $Nodes=adj(Lf)$

21: Βρες $R_{leaf}=[v_{adj.min}, v_{adj.max}]$ με $sup(L_f)>=k$

22: **Εάν** η μεταβολή της ποινή του ορίου $NCP\ (R_{leaf})<d$:

23: $path(L_f)=\{anc_1(L_f), anc_2(L_f),.., anc_n(Lf)\}$

24: $path(adj)=\{anc_1(adj), anc_2(adj),.., anc_n(adj)\}$

25: **Για κάθε** ζεύγος κόμβων $\{anc_i(L_f), anc_i(adj)\}$ {

26: $R_{anc,i}=[v_{min.anc}, v_{max.anc}]$

27: Υπολόγισε $NCP(R_{anc,i})$

28: }

29: **Εάν** η συνολική ποινή $NCP\ (R_{anc}, R_{leaf}) < d$

30: Τα όρια γενίκευσης R_{anc} και R_{leaf} είναι αποδεκτά

31: Τερμάτισε το βρόγχο

32: **Αλλιώς:**

33: Ψάξε σε άλλους γειτονικούς κόμβους $adj(L_f)$

34: **Αλλιώς:**

35: Ψάξε σε άλλους γειτονικούς κόμβους $adj(L_f)$

36: }

37: }

38: Αντικατέστησε όλες τιμές των εγγραφών με τα νέα διαστήματα R_{anc} και R_{leaf}

39: Ενημέρωσε το *count tree* με τα νέα διαστήματα τιμών

40: }

41: }

■

4.5 Παράδειγμα υλοποίησης

Για την καλύτερη κατανόηση του αλγόριθμου, χρησιμοποιείται το παράδειγμα του πιο κάτω πίνακα, όπου επιχειρείται η *2*-ανωνυμοποίηση, όταν ο επιτιθέμενος γνωρίζει το πολύ 2 τιμές από το σύνολο τιμών μιας εγγραφής. Η μέγιστη μεταβολή της Κανονικοποιημένης Ποινής Βεβαιότητας (NCP) της βάσης, ορίζεται ίση με *d=0.2*. Τα δεδομένα του παραδείγματος αφορούν τα μερικά εισοδήματα ατόμων.

Α/Α	ΣΥΝΟΛΟ ΕΙΣΟΔΗΜΑΤΩΝ
0	{200, 100, 100, 400, 400}
1	{400, 100, 300, 500}
2	{500, 100, 400, 400, 100}

Πίνακας 4.3

Αρχικά, ο αλγόριθμος υπολογίζει τις συχνότητες εμφάνισης κάθε διαφορετική τιμής στη βάση δεδομένων, τον αριθμό των διαφορετικών εγγραφών που αυτή εμφανίζεται και βάσει αυτών ταξινομεί το σύνολο δεδομένων, όπως φαίνεται στον πίνακα 4.5

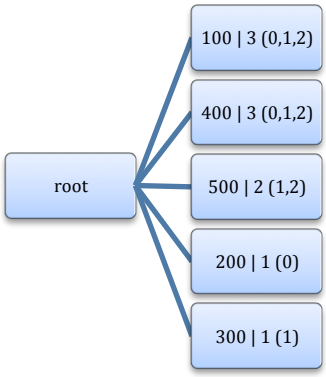
ΤΙΜΗ	ΣΥΧΝΟΤΗΤΑ ΕΜΦΑΝΙΣΗΣ	ΠΛΗΘΟΣ ΕΓΓΡΑΦΩΝ	Α/Α ΕΓΓΡΑΦΩΝ
{100}	5	3	0,1,2
{400}	5	3	0,1,2
{500}	2	2	1,2
{200}	1	1	0
{300}	1	1	1

Πίνακας 4.4 Συχνότητα εμφάνισης κάθε τιμής και αριθμός εγγραφών που εμφανίζεται

Α/Α	ΤΑΞΙΝΟΜΗΜΕΝΟ ΣΥΝΟΛΟ
0	{100, 100, 400, 400, 200}
1	{100 400 500 300}
2	{100, 100, 400, 400, 500}

Πίνακας 4.5 Ταξινομημένο σύνολο δεδομένων

Βάσει του πιο πάνω πίνακα σύμφωνα και με το [TMK08] σαρώνεται η ταξινομημένη βάση και δημιουργείται το δέντρο συχνοτήτων για m=1. Κάθε κόμβος του δέντρου περιλαμβάνει την τιμή της βάσης, το πλήθος των διαφορετικών εγγραφών που αυτή εμφανίζεται και τους αύξοντες αριθμούς των εγγραφών αυτών.



Σχήμα 4.5 Δέντρο συχνοτήτων για m=1

Ο αλγόριθμος βρίσκει τους κόμβους που παραβιάζουν την ιδιωτικότητα ($k<2$) προκειμένου να τους γενικεύσει. Ξεκινώντας από τον προβληματικό κόμβο {200|1} ελέγχει για πιθανές γενικεύσεις του κόμβου με γειτονικούς του κόμβους στο δέντρο. Αρχικά ταξινομεί τους αδελφικούς κόμβους σε αύξουσα σειρά.

Σχήμα 4.6 Αδερφικοί κόμβοι σε αύξουσα σειρά

Ξεκινώντας για πλήθος κόμβων ίσο με 2, βρίσκει τις γενικεύσεις που μπορούν να γίνουν, και εφόσον αυτές λύνουν το πρόβλημα υπολογίζει το NCP(D) της βάσης βάσει των γενικεύσεων αυτών.

$$NCP([100,200]) = \frac{(200-100+1)}{500-100+1} = \frac{101}{401} = 0.25$$

$$NCP(D) = \frac{0.25 \cdot 6}{14} = 0.107$$

$$NCP([200,300]) = \frac{(300-200+1)}{500-100+1} = \frac{101}{401} = 0.25$$

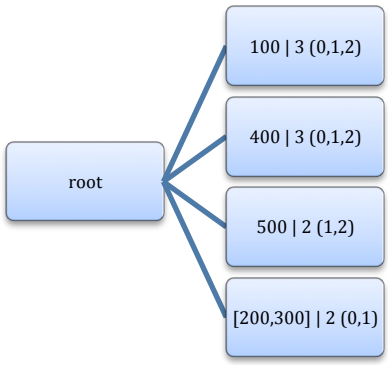
$$NCP(D) = \frac{0.25 \cdot 2}{14} = 0.035$$

Αφού ο αλγόριθμος βρίσκει πιθανές γενικεύσεις για πλήθος κόμβων ίσο με 2, δεν χρειάζεται να ψάξει για μεγαλύτερο πλήθος κόμβων αφού σίγουρα θα έχουν μεγαλύτερη απώλεια πληροφορίας. Από τις δύο πιθανές γενικεύσεις επιλέγει την μικρότερη η οποία ικανοποιεί το $NCP_{curr}(D)$-$NCP_{prev}(D)<d$.

ΠΛΗΘΟΣ ΚΟΜΒΩΝ	ΓΕΝΙΚΕΥΣΗ	ΠΛΗΘΟΣ ΔΙΑΦΟΡΕΤΙΚΩΝ ΕΓΓΡΑΦΩΝ	NCP (D)
2	[100,200]	3	0.107-0 < d
	[200,300]	**2**	**0.035-0 < d**

Πίνακας 4.6 Πιθανές γενικεύσεις του κόμβου {200}

Στη συνέχεια εφαρμόζεται ολική γενίκευση στο δέντρο, με αποτέλεσμα όλοι οι κόμβοι με τιμές {200} και {300} να αντικατασταθούν με τη νέα γενικευμένη τιμή [200,300]. Η μορφή του δέντρου μετά το τέλος της γενίκευσης φαίνεται στο πιο κάτω σχήμα.



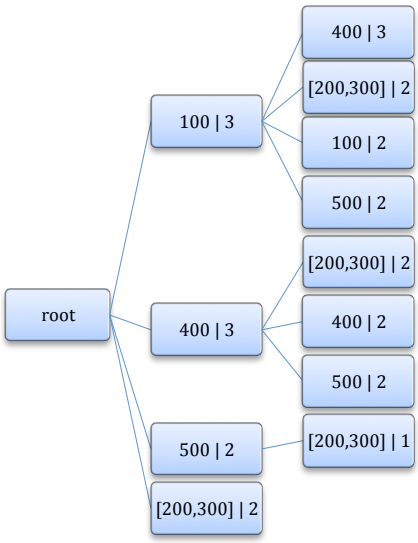
Σχήμα 4.7 Δέντρο συχνοτήτων μετά από γενίκευση [200, 300]

Όπως αποτυπώνεται και στο δέντρο, για *m=1* μετά από τη γενίκευση [200,300] δεν υπάρχει οποιοσδήποτε άλλος προβληματικός κόμβος ο οποίος να παραβιάζει την ιδιωτικότητα για *k=2*.

Στο επόμενο βήμα, ο αλγόριθμος δημιουργεί το δέντρο συχνοτήτων για *m=2*. Περνά όλους τους ανά δύο συνδυασμούς τιμών κάθε εγγραφής στο δέντρο συχνοτήτων, μαζί με το πλήθος εμφανίσεων τους στη βάση.

ΣΥΝΔΥΑΣΜΟΣ	ΠΛΗΘΟΣ ΕΜΦΑΝΙΣΗΣ	Α/Α ΕΓΓΡΑΦΩΝ
{100, 100}	2	0,2
{100, 400}	3	0,1,2
{100, [200,300]}	2	0,1
{400, 400}	2	0,2
{400, [200,300]}	2	0,1
{100, 500}	2	1,2
{500, [200,300]}	1	1
{400, 500}	2	1,2

Πίνακας 4.7 Ανά δύο συνδυασμοί τιμών κάθε εγγραφής

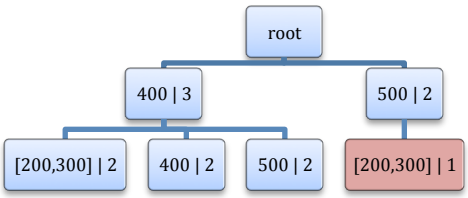


Σχήμα 4.8 Δέντρο συχνοτήτων για m=2

Από το δέντρο συχνοτήτων επιλέγονται οι κόμβοι που παραβιάζουν την ιδιωτικότητα ($k<2$) για $m=2$, προκειμένου να γενικευτούν. Στο παράδειγμα, ο μόνος προβληματικός κόμβος είναι ο κόμβος {[200,300] |1}.

Αφού ο συγκεκριμένος κόμβος δεν έχει *αδελφικούς* κόμβους, ο αλγόριθμος διατρέχοντας αναδρομικά το δέντρο ελέγχει πιθανές γενικεύσεις με *γειτονικούς* κόμβους-φύλλα στο δέντρο. Στο παράδειγμα ο αλγόριθμος γενικεύει τα παιδιά του κόμβου {500|1} με τα παιδιά του κόμβου {400|3}, όπως φαίνεται και στο πιο κάτω σχήμα.

Ξεκινώντας για πλήθος γειτονικών-κόμβων ίσο με 2, βρίσκει τις γενικεύσεις που μπορούν να γίνουν, και εφόσον αυτές λύνουν το πρόβλημα υπολογίζει το NCP(D) της βάσης βάσει των γενικεύσεων αυτών.



Σχήμα 4.9 Συνένωση γειτονικών κόμβων του {[200,300]|1}

ΠΛΗΘΟΣ ΠΡΟΓΟΝΩΝ	ΚΟΜΒΟΙ ΠΡΟΓΟΝΟΙ	ΠΛΗΘΟΣ ΠΑΙΔΙΩΝ	ΚΟΜΒΟΙ ΠΑΙΔΙΑ	ΤΕΛΙΚΕΣ ΓΕΝΙΚΕΥΣΕΙΣ	NCP (D)
2	{400},{500}	**2**	**{[200,300]}, {[200,300]}**	**[400,500]**	**0.16-0.035<d**
		2	{[200,300]}, {400}	[200,500]	0.48-0.035>d
		2	{[200,300]}, {500}	[200,500]	0.48-0.035>d

Πίνακας 4.8 Πιθανές γενικεύσεις του κόμβου {[200,300]|1}

Η απώλεια πληροφορίας στη βάση υπολογίζεται βάσει της Κανονικοποιημένης Ποινής Βεβαιότητας.

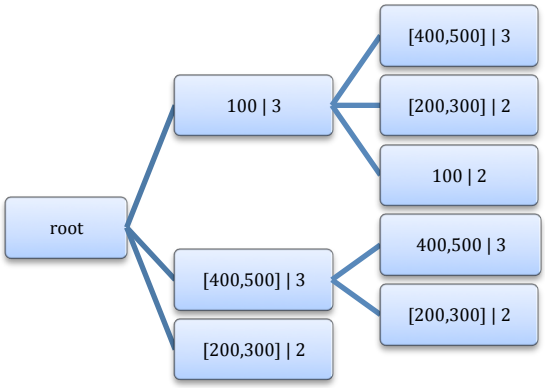
$$NCP([400,500]) = \frac{(500-400+1)}{500-100+1} = \frac{101}{401} = 0.25$$

$$NCP(D) = \frac{0.25 \cdot 2 + 0.25 \cdot 7}{14} = 0.1607$$

$$NCP([200,500]) = \frac{(500-200+1)}{500-100+1} = \frac{301}{401} = 0.75$$

$$NCP(D) = \frac{0.75 \cdot 9}{14} = 0.482$$

Αφού ο αλγόριθμος βρίσκει πιθανές γενικεύσεις για πλήθος προγόνων ίσο με 2 και πλήθος κόμβων ίσο με 2, δεν χρειάζεται να ψάξει για μεγαλύτερο πλήθος κόμβων ούτε για μεγαλύτερο πλήθος προγόνων, αφού σίγουρα θα προκαλούν μεγαλύτερη απώλεια πληροφορίας. Από τις τρεις πιθανές γενικεύσεις επιλέγεται η μικρότερη η οποία ικανοποιεί το $NCP_{curr}(D)$-$NCP_{prev}(D)<d$. Στη συνέχεια, εφαρμόζεται ολική γενίκευση στο δέντρο, με αποτέλεσμα όλοι οι κόμβοι με τιμές {400},{500} να αντικαθιστώνται με τη γενικευμένη τιμή [400,500]. Η μορφή του δέντρου μετά το τέλος της γενίκευσης φαίνεται στο πιο κάτω σχήμα.



Σχήμα 4.10 Δέντρο συχνοτήτων μετά από γενίκευση [400,500]

Όπως αποτυπώνεται και στο δέντρο, για $m=2$ μετά από τη γενίκευση [400,500] δεν υπάρχει οποιοσδήποτε άλλος προβληματικός κόμβος ο οποίος να παραβιάζει την ιδιωτικότητα για $k=2$. Τελειώνοντας τη διαδικασία ανωνυμοποίησης το δημοσιευμένο σύνολο που προκύπτει είναι το ακόλουθο με Κανονικοποιημένη Ποινή Βεβαιότητας $NCP(D)=0.16 < d$

Α/Α	ΑΝΩΝΥΜΟΠΟΙΗΜΕΝΟ ΣΥΝΟΛΟ
0	{100, 100, [400,500], [400,500], [200,300]}
1	{100, [400,500], [400,500], [200,300]}
2	{100, 100, [400,500], [400,500], [400,500]}

Πίνακας 4.9 2-Ανωνυμοποιημένο σύνολο δεδομένων για m=2

5

Αξιολόγηση

Ο προτεινόμενος αλγόριθμος k^m-ανωνυμίας, αξιολογήθηκε μέσω της εφαρμογή του τόσο σε συνθετικά αριθμητικά δεδομένα όσο και σε πραγματικά οικονομικά σύνολα δεδομένων. Στο κεφάλαιο αυτό αναλύονται όλες οι λεπτομέρειες που αφορούν τις παραμέτρους αξιολόγησης του αλγόριθμου, τα σύνολα των δεδομένων, τα πειραματικά αποτελέσματα και τους παράγοντες που επηρεάζουν την απόδοση του.

Παράλληλα, γίνεται σύγκριση του προτεινόμενου αλγόριθμου με αυτόν της k^m-ανωνυμίας με χρήση ιεραρχιών γενίκευσης ως προς την μετρική απώλειας πληροφορίας και το χρόνο εκτέλεσης τους. Και οι δύο οι αλγόριθμοι χρησιμοποιούν το δέντρο συχνοτήτων για να αξιολογήσουν τις υποψήφιες γενικεύσεις, με αποτέλεσμα να αποφεύγεται η σάρωση της βάσης δεδομένων σε κάθε έλεγχο.

5.1 Παράμετροι αξιολόγησης

Οι παράμετροι αξιολόγησης που χρησιμοποιήθηκαν για την διεξαγωγή των πειραμάτων σχετικά με την αποδοτικότητα του αλγορίθμου είναι (α) η μετρική απώλειας πληροφορίας και (β) ο χρόνος εκτέλεσης του. Οι τιμές που λαμβάνει η Κανονικοποιημένη Ποινή Βεβαιότητας αλλά και ο χρόνος εκτέλεσης επηρεάζονται άμεσα από τις τιμές που λαμβάνουν οι παράμετροι εισόδου του αλγορίθμου. Η Κανονικοποιημένη Ποινή Βεβαιότητας εξαρτάται και μεταβάλλεται σημαντικά για διαφορετικές τιμές της παραμέτρου ανωνυμίας *k* αλλά και της παραμέτρου m. Αντίστοιχα, ο χρόνος εκτέλεσης του αλγόριθμου εξαρτάται από το μέγεθος

του συνόλου των δεδομένων, τον αριθμό των εγγραφών, καθώς και από τις τιμές των παραμέτρων k και m.

5.1.1 Μετρική απώλεια πληροφορίας

Η μετρική απώλεια πληροφορίας χρησιμοποιείται βάσει της Κανονικοποιημένης Ποινής Βεβαιότητας (NCP) όπως αυτή έχει οριστεί στο Κεφάλαιο 3. Σύμφωνα με τη μετρική αυτή κάθε γενικευμένη τιμή v_i της μορφής $[yi, zi]$ που προκύπτει στη βάση δεδομένων έχει Κανονικοποιημένη Ποινή Βεβαιότητας:

$$NCP([yi, zi]) = \begin{cases} 0, & z_i = y_i \\ \dfrac{z_i - y_i}{|I|}, & z_i \neq y_i \end{cases}$$

όπου $|I|$ το μέγεθος του πεδίου τιμών I των γνωρισμάτων της κάθε εγγραφής.

Η απώλεια πληροφορίας του k^m-ανώνυμου συνόλου δεδομένων D′ για κάθε γενικευμένη τιμή v_i της μορφής $[y_i, z_i]$ με συνολικό αριθμό εμφανίσεων στη βάση C_{vi}, υπολογίζεται σύμφωνα με την Κανονικοποιημένη Ποινή Βεβαιότητας ίση με:

$$NCP(D') = \frac{\sum_{vi \in D}(C_{v_i} \cdot NPC(v_i))}{\sum_{vi \in D}(C_{v_i})}$$

όπου C_{v_i} η συχνότητα εμφάνισης της κάθε τιμής v_i στη βάση δεδομένων.

Η μετρική αυτή υποδεικνύει την απώλεια πληροφορίας από τα αρχικά δεδομένα μετά την ανωνυμοποίηση τους. Χρησιμοποιήθηκε για την αξιολόγηση των αποτελεσμάτων και των δύο αλγόριθμων που παρουσιάζονται στην εργασία.

5.1.2 Χρόνος εκτέλεσης

Η δεύτερη παράμετρος αξιολόγησης που χρησιμοποιείται στα πειραματικά αποτελέσματα σχετικά με την αποδοτικότητα του αλγόριθμου, είναι ο χρόνος εκτέλεσης του. Ο χρόνος αφορά σε χρόνο επεξεργασίας της Κεντρικής Μονάδας Επεξεργασία (CPU time) και ορίζεται σαν ο χρόνος που δαπανάται από την CPU για την πραγματική εκτέλεση του προγράμματος. Ο χρόνος CPU μετριέται σε δευτερόλεπτα (seconds).

Με τη χρήση του χρόνου εκτέλεσης σαν παραμέτρου αξιολόγησης μπορεί να παρατηρηθεί οποιαδήποτε διαφορά προκύπτει στον χρόνο εκτέλεσης αναφορικά με τα δεδομένα εισόδου, το μέγεθος του συνόλου δεδομένων, την τιμή της παραμέτρου ανωνυμίας k και της παραμέτρου μερικής γνώσης m.

5.2 Οργάνωση πειραμάτων

Τα πειράματα που εκτελέστηκαν βασίστηκαν στην υλοποίηση του αλγορίθμου όπως αυτή περιγράφεται στο Κεφάλαιο 4, με χρήση της αντικειμενοστρεφούς γλώσσας προγραμματισμού C++.

Για την αξιολόγηση του αλγόριθμου εκτελέστηκαν δύο σειρές πειραμάτων. Η πρώτη αφορούσε συνθετικά αριθμητικά δεδομένα και η δεύτερη πραγματικά οικονομικά δεδομένα των κατοίκων των Η.Π.Α, όπως περιγράφονται και πιο κάτω.

5.2.1 Συνθετικά Αριθμητικά Δεδομένα

Στην πρώτη σειρά πειραμάτων τα δεδομένα που χρησιμοποιήθηκαν και παρήγαγαν τα αποτελέσματα αναπαριστούν συνθετικά αριθμητικά γνωρίσματα. Τα δεδομένα αφορούσαν 500,000 εγγραφές με μέγιστο αριθμό τιμών σε κάθε εγγραφή n=5. Τα επιλεγμένα γνωρίσματα, προέρχονται από ένα κοινό πεδίο τιμών με |I|=1000, μιας και όλα αφορούν αριθμητικά δεδομένα. Στον πιο κάτω πίνακα φαίνονται τα χαρακτηριστικά του αρχικού συνόλου των αριθμητικών δεδομένων:

ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ	$\|D\|$	$\|I\|$	$max\ \|t\|$	$avg\ \|t\|$
Συνθετικά Αριθμητικά Δεδομένα	500,000	1,000	5	2.99

Πίνακας 5.1 Χαρακτηριστικά συνόλου συνθετικών δεδομένων (το t συμβολίζει την εγγραφή)

Τα πειράματα εκτελέστηκαν τόσο στο αρχικό σύνολο δεδομένων, όσο και σε υποσύνολα αυτού με μέγεθος 100000, 50000, 25000, 10000, 5000, 1000 το καθένα από το αμέσως μεγαλύτερό του, διαδοχικά. Τα επιλεγμένα γνωρίσματα, προέρχονται από ένα κοινό πεδίο τιμών, αφού όλα αφορούν αριθμητικά δεδομένα.

5.2.2 Πραγματικά Οικονομικά Δεδομένα

Το δεύτερο σύνολο δεδομένων που χρησιμοποιήθηκε, αφορούσε πραγματικά οικονομικά δεδομένα. Συγκεκριμένα, πρόκειται για προσωπικά φορολογικά δεδομένα που προέρχονται από την απογραφή των κατοίκων των Ηνωμένων Πολιτειών Αμερικής τη χρονολογία 1990, όπως αυτά δημοσιεύονται στο UCI Machine Learning Repository [1].

Το αρχικό σύνολο δεδομένων περιείχε 2,458,285 εγγραφές με 68 γνωρίσματα. Από αυτά απομονώθηκαν οκτώ γνωρίσματα που αναφέρονταν σε εισοδήματα του κάθε νοικοκυριού. Τα

γνωρίσματα αφορούν ακέραιους αριθμούς και θεωρείται ότι, προέρχονται από ένα κοινό πεδίο τιμών μιας και όλα αναπαριστούν οικονομικά δεδομένα. Για τη διεξαγωγή των πειραμάτων το πεδίο τιμών περιορίστηκε σε |I|=1000, με την κατάλληλη επεξεργασία των εγγραφών.

Από το αρχικό σύνολο δεδομένων, με 2,458,285 εγγραφές, αφαιρέθηκαν οι εγγραφές με μηδενικές τιμές και προέκυψε ένα σύνολο δεδομένων μεγέθους 1,000,000 εγγραφών. Στον πιο κάτω πίνακα φαίνονται τα χαρακτηριστικά του συνόλου των πραγματικών δεδομένων όπως αυτά χρησιμοποιήθηκαν στην πειραματική διαδικασία.

ΣΥΝΟΛΟ ΔΕΔΟΜΕΝΩΝ	$\|D\|$	$\|I\|$	$max\ \|t\|$	$avg\ \|t\|$
Πραγματικά Οικονομικά Δεδομένα	1,000,000	1,000	8	2.27

Πίνακας 5.2 Χαρακτηριστικά συνόλου οικονομικών δεδομένων (το t συμβολίζει την εγγραφή)

Τα πειράματα εκτελέστηκαν τόσο στο πιο πάνω σύνολο δεδομένων, όσο και σε υποσύνολα αυτού με μέγεθος 500000, 100000, 50000, 25000, 10000, 5000, 1000 το καθένα από το αμέσως μεγαλύτερό του, διαδοχικά. Τα επιλεγμένα γνωρίσματα, προέρχονται από ένα κοινό πεδίο τιμών, αφού όλα αφορούν πραγματικά οικονομικά δεδομένα.

5.2.3 Διαδικασία πειραμάτων

Μετά την κατάλληλη τροποποίηση των δεδομένων, εκτελέστηκαν επαναλήψεις της υλοποίησης του αλγορίθμου για κάθε ένα από τα υποσύνολα δεδομένων και για τις τιμές των παραμέτρων $k=\{2, 5, 10, 50\}$, $m=\{1, 2, 3\}$ και $d=\{10^{-5}, 10^{-4}, 10^{-3}, 1\}$, όπου υπολογίσθηκε η Κανονικοποιημένη Ποινή Βεβαιότητας και ο χρόνος εκτέλεσης.

Αντίστοιχες εκτελέσεις του αλγορίθμου της κλασσικής k^m-ανωνυμίας με χρήση ιεραρχιών γενίκευσης, έγιναν για τα ίδια υποσύνολα δεδομένων και τις ίδιες τιμές της παραμέτρου k και m, ενώ δεν εξετάστηκε κάποια αντιστοιχία για την παράμετρο d. Ο βαθμός του δέντρου ιεραρχίας γενίκευσης ορίστηκε ίσος με *degree=2*. Τα ανωνυμοποιημένα δεδομένα που προκύπτουν από τους δύο αλγορίθμους συγκρίθηκαν ως προς την απώλεια πληροφορίας που εμφανίζουν με χρήση της Κανονικοποιημένης Ποινής Βεβαιότητας. Στη συνέχεια έγινε αξιολόγηση της αποδοτικότητας των δύο αλγορίθμων σύμφωνα με το χρόνο εκτέλεσης τους.

Το σύνολο των πειραμάτων έγινε με τη χρήση προσωπικού υπολογιστή, με επεξεργαστή Intel Core 2 Duo, με CPU 2,53GHz, με RAM 4GB και λειτουργικό σύστημα Mac OS X 10.8.5. Για την διεξαγωγή των πειραμάτων στα Mac OS X χρησιμοποιήθηκε το ολοκληρωμένο περιβάλλον ανάπτυξης (Integrated Development Environment –IDE) Eclipse.

5.3 Αποτελέσματα πειραμάτων

Με βάση τις δύο παραμέτρους αξιολόγησης που ορίστηκαν στο Κεφάλαιο 5.1, εξετάστηκε η απόδοση του αλγορίθμου σε σχέση με τις διαφορετικές τιμές των παραμέτρων εισόδου, για όλα τα υποσύνολα δεδομένων που προέκυψαν μετά τη δειγματοληψία.

Αρχικά, εκτελέστηκε μια σειρά πειραμάτων για τα συνθετικά αριθμητικά δεδομένα και στη συνέχεια με τις ίδιες παραμέτρους αξιολόγησης εξετάστηκε η απόδοση του αλγόριθμου και για τα πραγματικά οικονομικά δεδομένα.

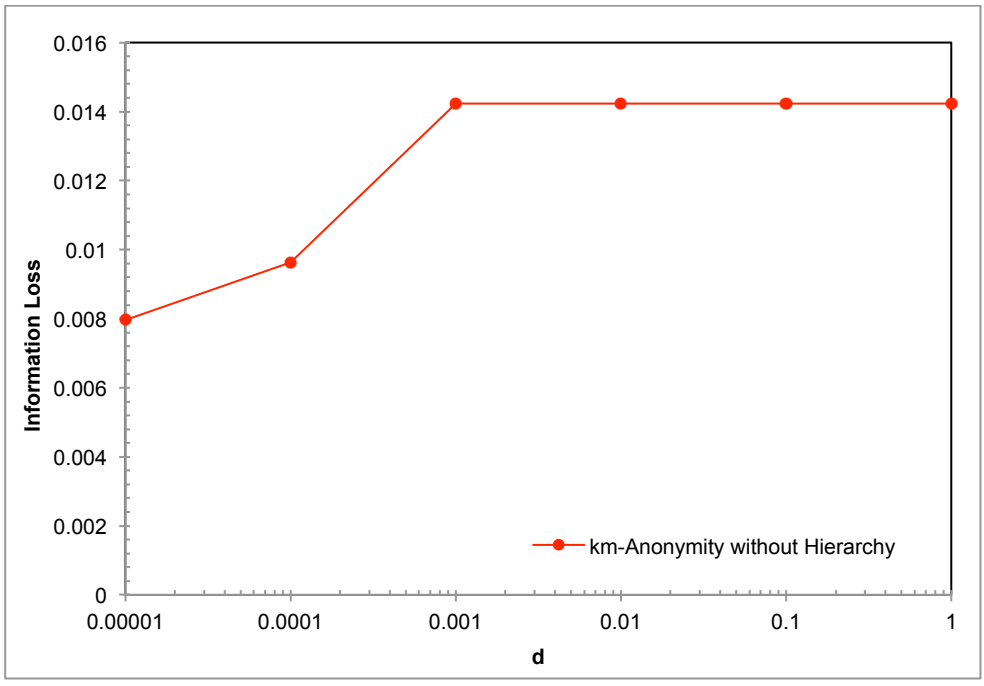
Σε κάθε πειραματική διαδικασία, ο αλγόριθμος εξετάστηκε και συγκρίθηκε με αυτόν της k^m-ανωνυμίας βάσει της Κανονικοποιημένης Ποινής Βεβαιότητας και στη συνέχεια βάσει του χρόνου εκτέλεσης του.

5.3.1 Συνθετικά Αριθμητικά Δεδομένα

5.3.1.1 Απώλεια πληροφορίας

Σε πρώτο στάδιο, μελετήθηκε η Κανονικοποιημένη Ποινή Βεβαιότητας για τις επαναλήψεις του αλγορίθμου και συγκρίθηκε με εκείνη του αλγορίθμου της κλασσικής k^m-ανωνυμίας για τα συνθετικά αριθμητικά δεδομένα.

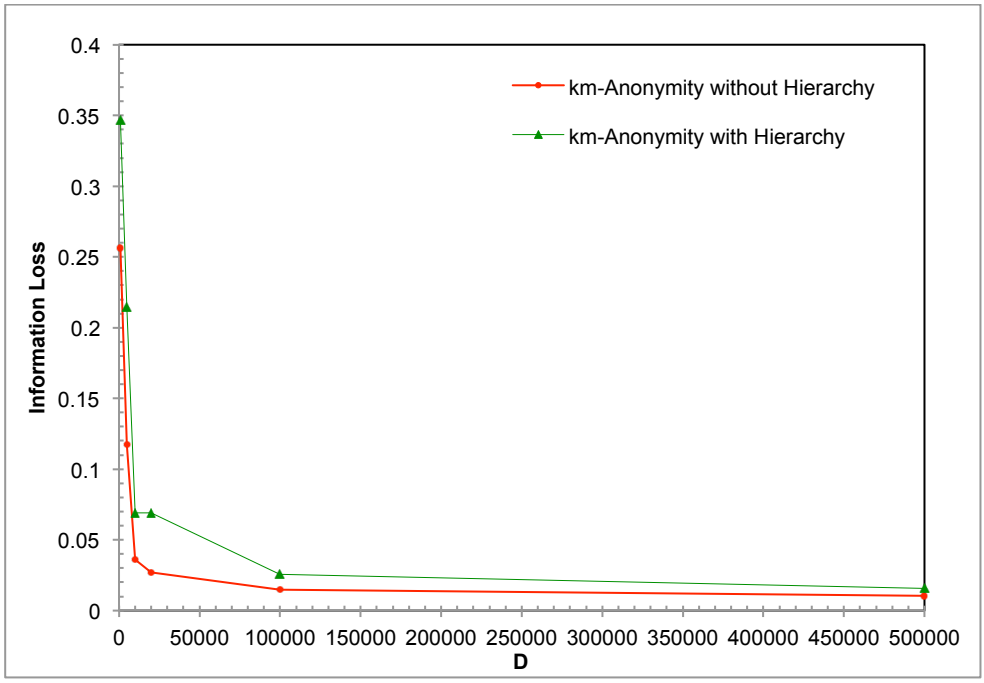
Στην Εικόνα 5.1 φαίνεται πως μεταβάλλεται η απώλεια πληροφορίας καθώς αυξάνεται η παράμετρος μέγιστης μεταβολής της ποινής d, στον αλγόριθμο χωρίς ιεραρχίες γενίκευσης.



Εικόνα 5.1. Μετρική απώλειας πληροφορίας – παράμετρος μεταβολής NCP (k=2 m=2 |D|=100,000)

Όπως ήταν αναμενόμενο, παρατηρείται αύξηση της απώλειας πληροφορίας για μεγαλύτερες τιμές του d, αφού όσο μεγαλύτερη είναι η επιτρεπτή μεταβολή της ποινής σε κάθε γενίκευσης, γίνονται αποδεκτές γενικεύσεις οι οποίες προκαλούν μεγάλη απώλεια πληροφορίας. Από την άλλη όσο μικρότερη είναι η τιμή της παραμέτρου d, τόσο μικρότερη απώλεια πληροφορίας προκαλείται στη βάση σε κάθε γενίκευση.

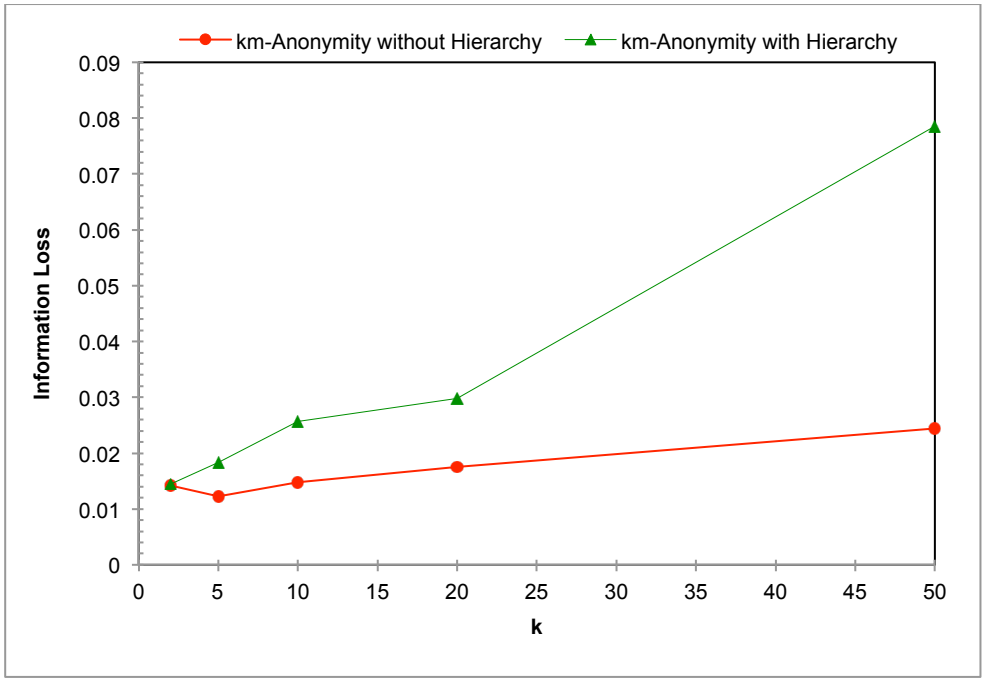
Στη συνέχεια, στην Εικόνα 5.2 παρουσιάζεται η απώλεια πληροφορίας που εμφανίζεται στα k^m-ανωνυμοποιημένα δεδομένα κατά την εκτέλεση των δύο αλγορίθμων για σύνολα δεδομένων διαφορετικού μεγέθους. Οι παράμετροι ανωνυμίας είχαν τις τιμές k=10 και m=2.



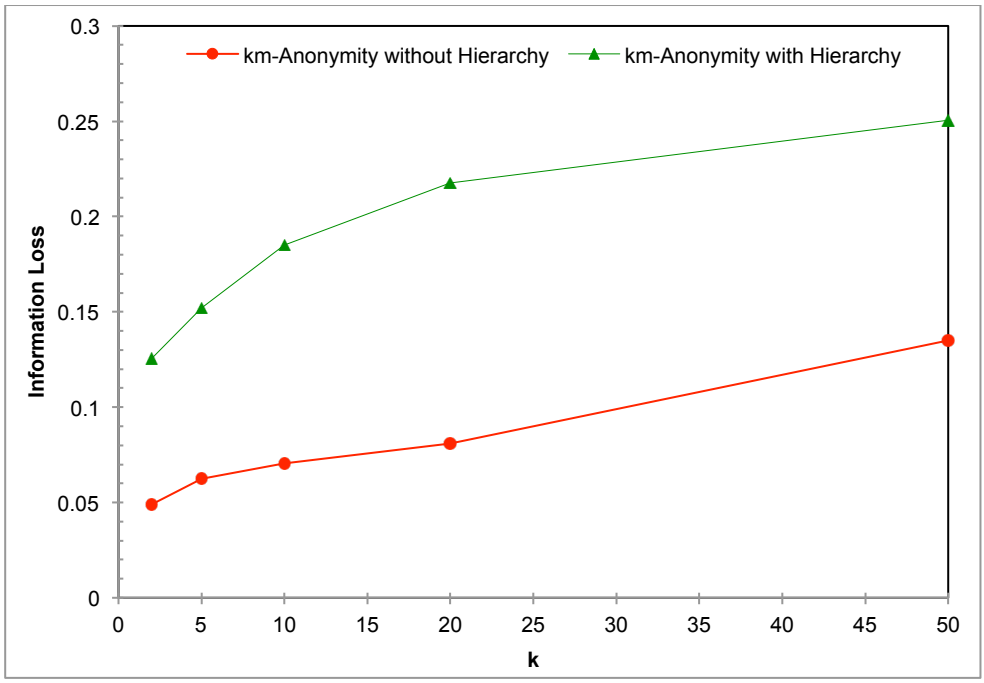
Εικόνα 5.2 Μετρική απώλειας πληροφορίας – πλήθος εγγραφών συνόλου (k=10 m=2 d=0.001)

Συμπερασματικά, ο αλγόριθμος της k^m-ανωνυμίας χωρίς ιεραρχίες, υπερέχει του αλγόριθμου με χρήση ιεραρχιών γενίκευσης αφού παρουσιάζει μικρότερη απώλεια πληροφορίας στα ανωνυμοποιημένα δεδομένα. Και στις δύο περιπτώσεις παρατηρείται αύξηση της απώλειας πληροφορίας για τα μικρότερα σύνολα δεδομένων, γεγονός που τεκμηριώνεται λογικά, αφού για λιγότερες εγγραφές, απαιτείται μεγαλύτερη γενίκευση, λόγω του ότι η πιθανότητα εμφάνισης κοινών τιμών στις εγγραφές είναι μικρή.

Στις Εικόνες 5.3 και 5.4, παρουσιάζεται η απώλεια πληροφορίας που προκύπτει έπειτα από την εκτέλεση και των δύο αλγορίθμων πάνω στο ίδιο σύνολο δεδομένων 100,000 εγγραφών, μεταβάλλοντας την παράμετρο ανωνυμίας k. Η παράμετρος μερικής γνώσης m είχε τιμές τις m=2 και m=3 σε κάθε περίπτωση.



Εικόνα 5.3 Απώλεια πληροφορίας - παράμετρος k (|D|=100,000, m=2, d=0.001)



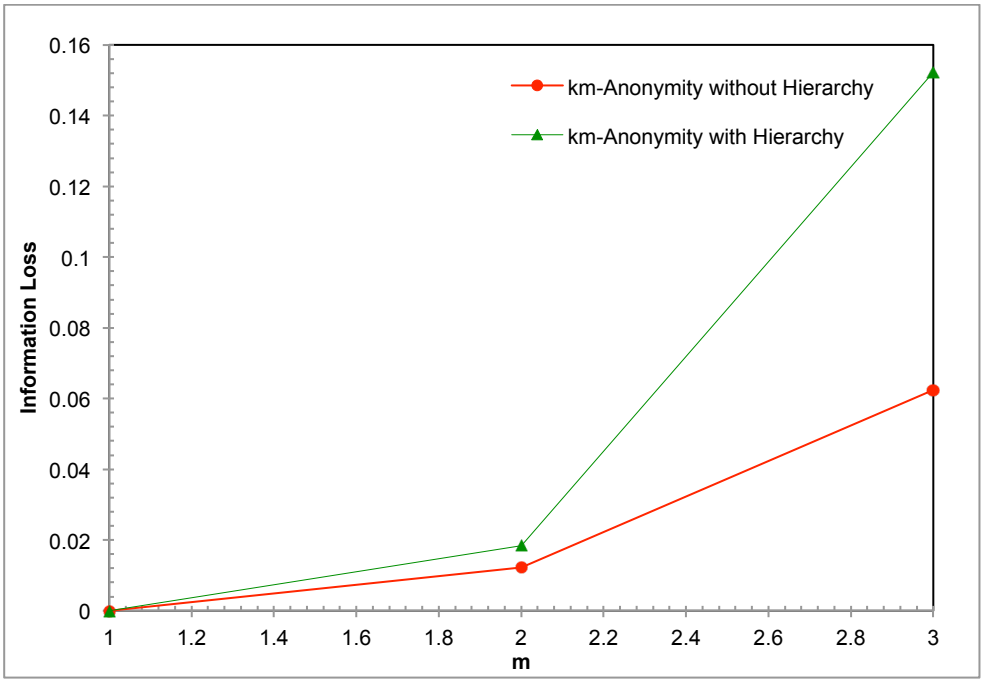
Εικόνα 5.4 Απώλεια πληροφορίας - παράμετρος k (|D|=100,000, m=3, d=0.001)

Από τις δύο πιο πάνω εικόνες διεξάγεται το συμπέρασμα της υπεροχής του προτεινόμενου αλγορίθμου σε σύγκριση με τον αλγόριθμο της k^m-ανωνυμίας με χρήση ιεραρχιών γενίκευσης, αναφορικά με την απώλεια πληροφορίας που εμφανίζουν τα ανωνυμοποιημένα δεδομένα.

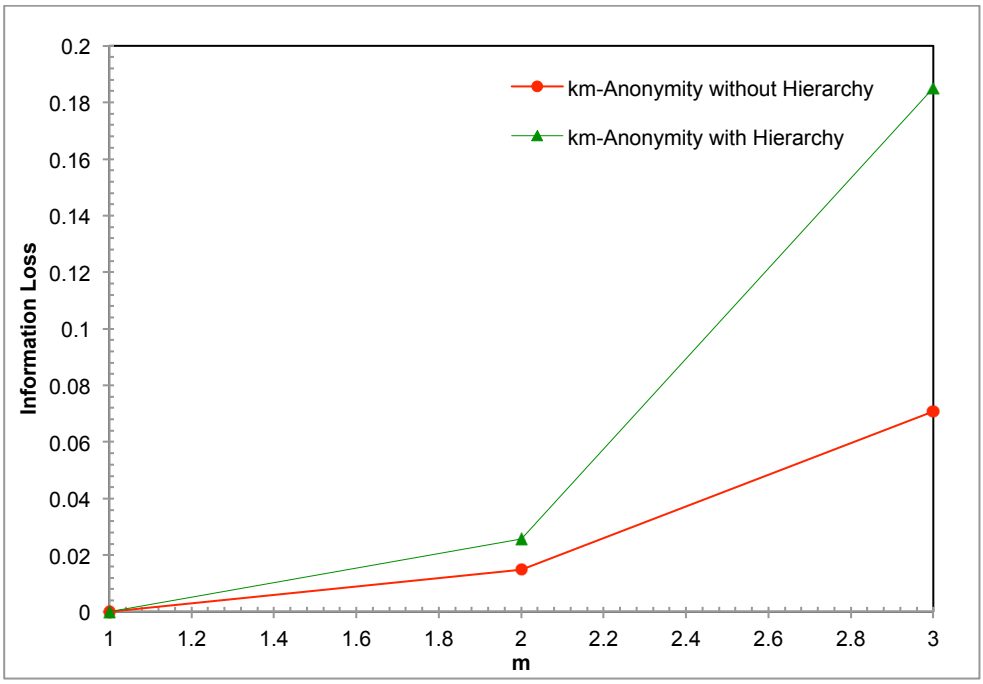
Παρατηρείται σημαντική διαφορά στην απώλεια πληροφορίας μεταξύ των δύο αλγορίθμων για κάθε συνδυασμό των παραμέτρων εισόδου, γεγονός που οφείλεται κατά κύριο λόγο στο

γεγονός ότι ο αλγόριθμος που προτείνεται από την εργασία εκμεταλλεύεται τη φύση των δεδομένων της βάσης.

Ο προτεινόμενος αλγόριθμος διατηρεί την Κανονικοποιημένη Ποινή Βεβαιότητας σε πάρα πολύ χαμηλά επίπεδα. Από την άλλη ο αλγόριθμος με τη χρήση ιεραρχιών αναγκάζεται να υπεργενικεύει τα δεδομένα με βάση το δέντρο ιεραρχίας. Όσο μεγαλύτερα διαστήματα τιμών υπάρχουν στο δέντρο, τόσο μεγαλύτερη είναι και η απώλεια πληροφορίας στη βάση.



Εικόνα 5.5 Απώλεια πληροφορίας - παράμετρος μερικής γνώσης m (|D|=100,000 k=5)



Εικόνα 5.6 Απώλεια πληροφορίας - παράμετρος μερικής γνώσης m (|D|=100,000 k=10)

Τα δύο τελευταία πειράματα για τον έλεγχο της απώλειας πληροφορίας έγιναν συναρτήσει της μερικής γνώσης m που κατέχει ο επιτιθέμενος πάνω στα δεδομένα. Σε σύνολο 100,000 εγγραφών με παραμέτρους k=5 και k=10 αντίστοιχα, ο προτεινόμενος αλγόριθμος υπερέχει του κλασσικού αλγόριθμου της k^m-ανωνυμίας, όσον αφορά την μετρική απώλεια πληροφορίας των ανωνυμοποιημένων δεδομένων.

Και στις δύο περιπτώσεις (Εικόνα 5.5 και Εικόνα 5.6) παρατηρείται αύξηση της απώλειας πληροφορίας για τις μεγαλύτερες τιμές της παραμέτρου m, γεγονός που τεκμηριώνεται λογικά, αφού για μεγαλύτερη γνώση m του επιτιθέμενου, απαιτείται μεγαλύτερη γενίκευση, λόγω του ότι η πιθανότητα εμφάνισης κοινών m συνδυασμών στις εγγραφές είναι μικρή.

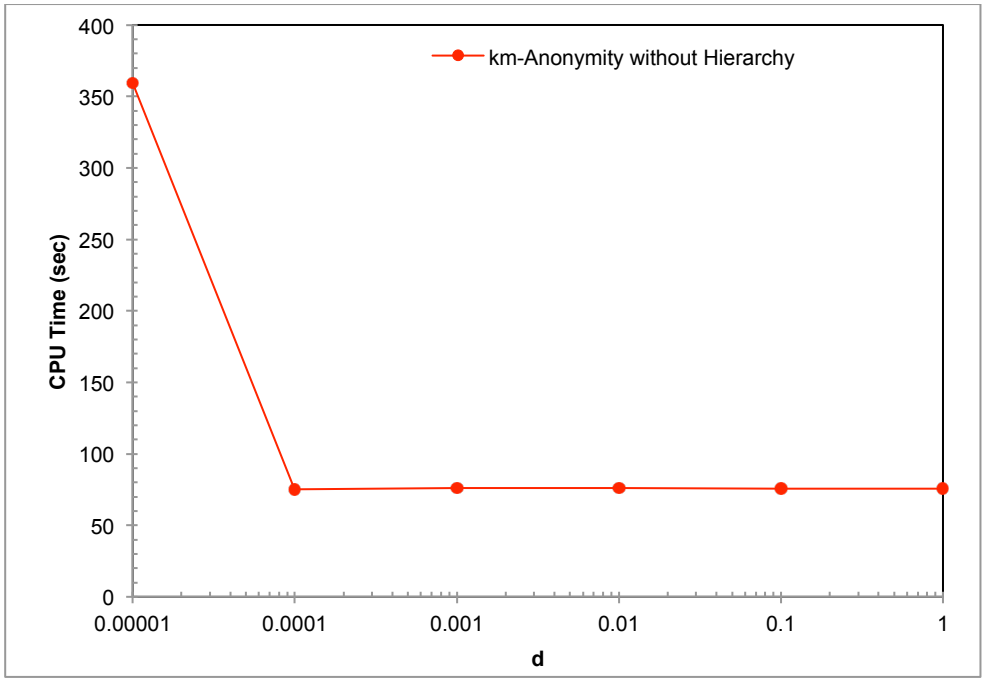
5.3.1.2 Χρόνος εκτέλεσης

Στην επόμενη ενότητα παρουσιάζονται τα αποτελέσματα από τις διαφορετικές εκτελέσεις του αλγορίθμου αναφορικά με τον χρόνο εκτέλεσής του. Ο χρόνος είναι ένα χρήσιμο εργαλείο για την διεξαγωγή συμπερασμάτων ως προς την απόδοση του αλγορίθμου. Συγκεκριμένα μπορεί να παρατηρηθεί οποιαδήποτε διαφορά προκύπτει στον χρόνο εκτέλεσης αναφορικά με τα δεδομένα εισόδου, το μέγεθος του συνόλου δεδομένων, την τιμή της παραμέτρου ανωνυμίας *k* ή της παραμέτρου μερικής γνώσης m.

Είναι αναμενόμενο ότι, ο αλγόριθμος της k^m-ανωνυμίας με τη χρήση ιεραρχιών γενίκευσης θα είναι πιο αποδοτικός σε σχέση με τον προτεινόμενο αλγόριθμο, λόγω του ότι εκτελεί λιγότερα βήματα κάθε φορά που βρίσκει ένα προβληματικό κόμβο στο δέντρο συχνοτήτων. Σε αντίθεση, ο προτεινόμενος αλγόριθμος προκειμένου να βρει μια αποδεκτή γενίκευση με μικρή απώλεια πληροφορίας χωρίς την χρήση ιεραρχίας, εκτελεί περισσότερους ελέγχους και αναδρομές στο δέντρο.

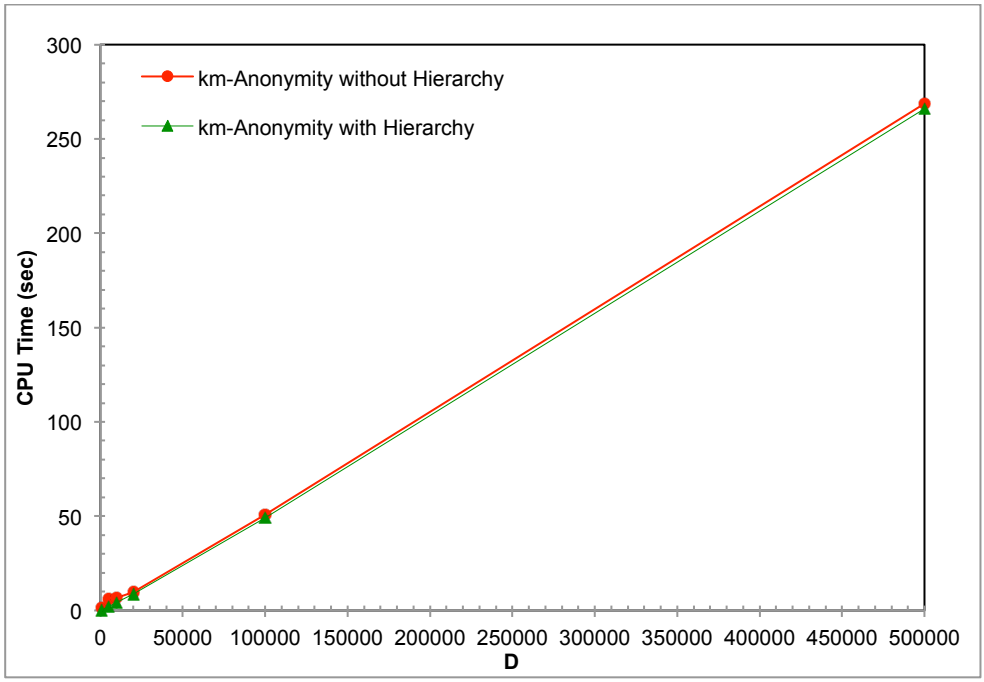
Παρ' όλα αυτά η σύγκριση των δύο αλγορίθμων ως προς τον χρόνο εκτέλεσης τους, δείχνει ότι ο χρόνος του προτεινόμενου αλγόριθμου κινείται σε πολύ χαμηλά επίπεδα και δεν διαφέρει σε μεγάλο βαθμό από αυτόν της κλασσικής k^m-ανωνυμίας, παρ' όλο που εκτελεί περισσότερα βήματα. Αυτό οφείλεται κυρίως σε δύο παράγοντες: (i) οι δομές δεδομένων που χρησιμοποιήθηκαν στο πρόγραμμα είχαν αποδοτικούς χρόνους διαχείρισης και (ii) ο αλγόριθμος, εκμεταλλευόμενος τις ιδιότητες των συνεχών γνωρισμάτων όπως αυτές περιγράφονται στο Κεφάλαιο 3, τερματίζει τους ελέγχους γενίκευσης πολύ γρήγορα.

Στην Εικόνα 5.7 παρουσιάζεται η μεταβολή του χρόνου εκτέλεσης καθώς αυξάνεται η παράμετρος μέγιστης μεταβολής της ποινής d, στον αλγόριθμο χωρίς ιεραρχίες γενίκευσης σε σύνολο δεδομένων 100,000 εγγραφών. Οι παράμετροι ανωνυμίας ήταν m=2 και k=2.



Εικόνα 5.7 Χρόνος CPU – παράμετρος μέγιστης μεταβολής NCP (k=2 m=2 |D|=100,000)

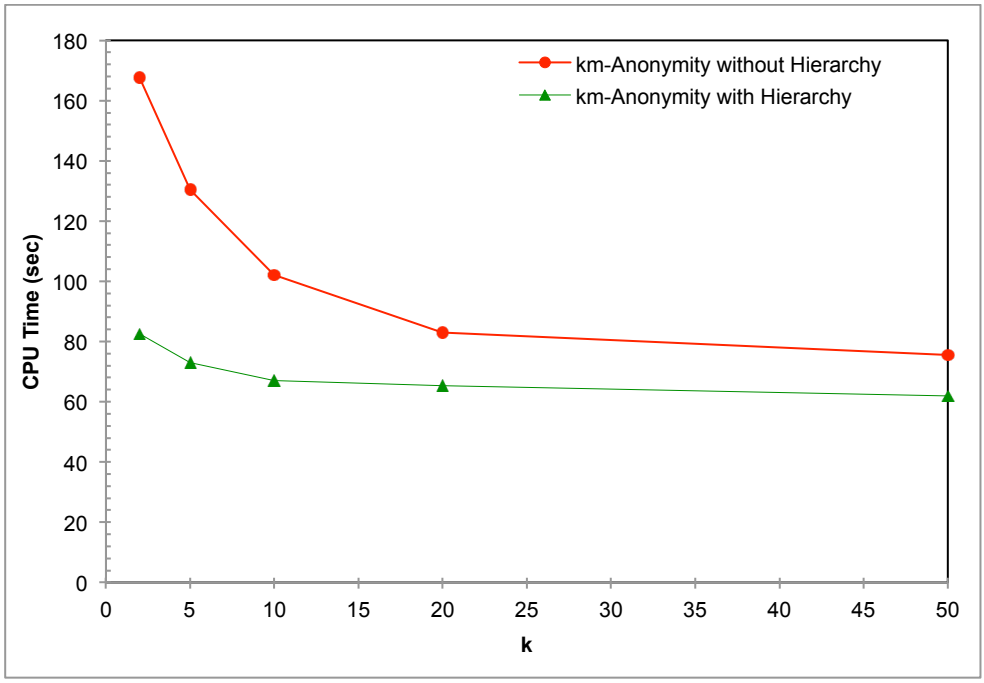
Όπως ήταν αναμενόμενο, ο αλγόριθμος παρουσιάζει μεγαλύτερο χρόνο εκτέλεσης καθώς η παράμετρος d παίρνει μικρότερες τιμές. Για μικρότερες τιμές του d, ο αλγόριθμος ελέγχει περισσότερους κόμβους στο δέντρο συχνοτήτων, προκειμένου να βρει μια γενίκευση που να μην προκαλεί μεταβολή στην ποινή της βάσης μεγαλύτερη από d.



Εικόνα 5.8 Χρόνος CPU – πλήθος εγγραφών συνόλου (k=10 m=3 d=0.001)

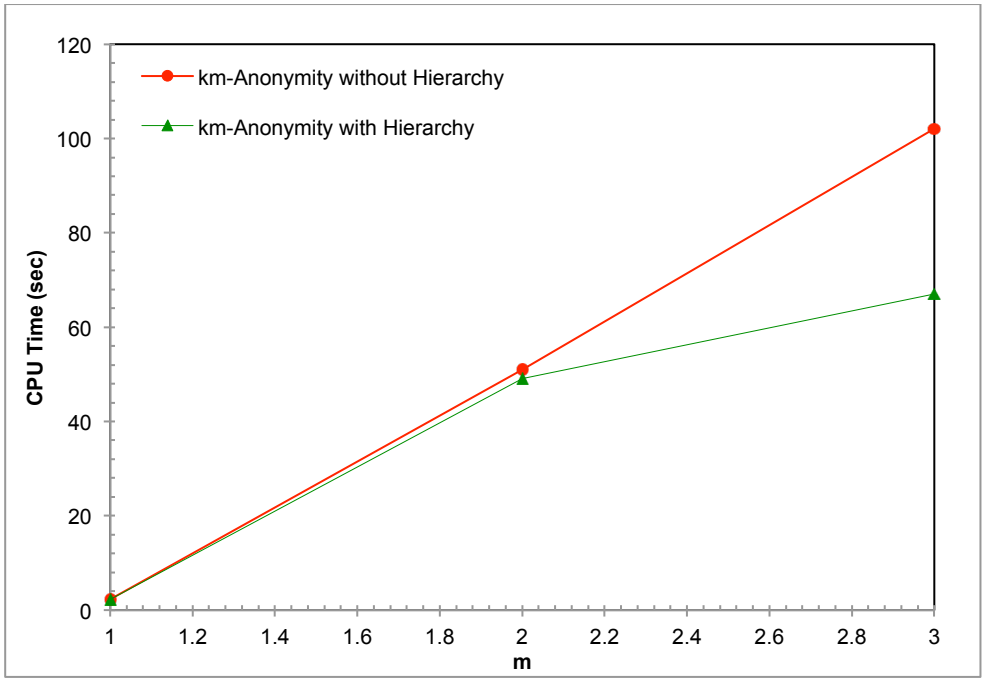
Στην Εικόνα 5.8 καταγράφονται οι χρόνοι εκτέλεσης των δύο αλγορίθμων για σύνολα δεδομένων διαφορετικού μεγέθους, και τιμές ανωνυμίας k=10 και m=3. Ο χρόνος εκτέλεσης και των δύο αλγορίθμων αυξάνεται, καθώς αυξάνεται το μέγεθος του συνόλου των εγγραφών.

Στην Εικόνα 5.9 αποτυπώνεται ο χρόνος εκτέλεσης του αλγορίθμου, με είσοδό το σύνολο δεδομένων 100,000 εγγραφών και τιμές της παραμέτρου *k* από το σύνολο {2,5,10,50}, με σταθερή παράμετρο μερικής γνώσης m=3. Ο χρόνος εκτέλεσης αυξάνεται, όταν η παράμετρος ανωνυμίας *k* μειώνεται. Αυτό είναι αναμενόμενο αφού, η πολυπλοκότητα του αλγορίθμου εξαρτάται από τον αριθμό των κόμβων του δέντρου. Όσο μικρότερη είναι η παράμετρος ανωνυμίας k τόσο περισσότεροι κόμβοι υπάρχουν στο count tree, με αποτέλεσμα οι έλεγχοι που γίνονται για πιθανές γενικεύσεις σε κάθε επανάληψη να είναι περισσότεροι.



Εικόνα 5.9 Χρόνος CPU – παράμετρος ανωνυμίας k (m=3 |D|=100,000 d=0.001)

Στην Εικόνα 5.10 καταγράφονται οι χρόνοι εκτέλεσης των δύο αλγορίθμων για το σύνολο δεδομένων 100,000 εγγραφών, με παράμετρο ανωνυμίας k=10 και m={1,2,3}. Οι χρόνοι εκτέλεσης και των δύο αυξάνονται, καθώς αυξάνεται η παράμετρος μερικής γνώσης του επιτιθέμενου. Αυτό είναι αναμενόμενο γιατί καθώς η παράμετρος m παίρνει μεγαλύτερες τιμές, καταγράφονται στο δέντρο συχνοτήτων περισσότεροι συνδυασμοί τιμών, με αποτέλεσμα το πλήθος των κόμβων του δέντρου να είναι μεγαλύτερο. Ο αλγόριθμος της κλασσικής k^m-ανωνυμίας για m=3 έχει μικρότερους χρόνους εκτέλεσης γιατί, δημιουργώντας γενικεύσεις μεγαλύτερων διαστημάτων μειώνει το πλήθος των κόμβων του δέντρου συχνοτήτων με αποτέλεσμα να προκαλεί στο τελικό ανωνυμοποιημένο σύνολο δεδομένων περισσότερη απώλεια πληροφορίας σε αντίθεση με τον προτεινόμενο αλγόριθμο.



Εικόνα 5.10 Χρόνος CPU – παράμετρος μερικής γνώσης m (k=10 |D|=100,000 d=0.001)

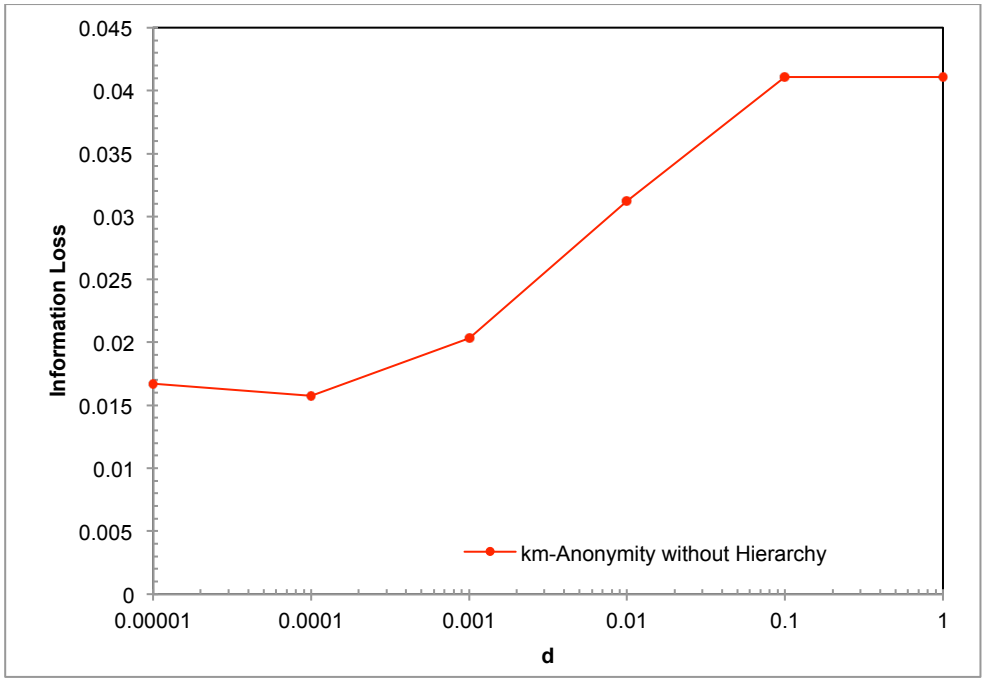
5.3.2 Πραγματικά Οικονομικά Δεδομένα

Στη δεύτερη σειρά πειραμάτων, ο αλγόριθμος αξιολογήθηκε βάσει της εφαρμογής του σε πραγματικά οικονομικά δεδομένα. Τα αποτελέσματα της πειραματικής διαδικασίας φαίνονται στην πιο κάτω ενότητα. Τα συμπεράσματα από τα πειράματα που έγιναν, είναι τα ίδια με αυτά της εφαρμογής του αλγόριθμου σε συνθετικά δεδομένα.

5.3.2.1 Απώλεια πληροφορίας

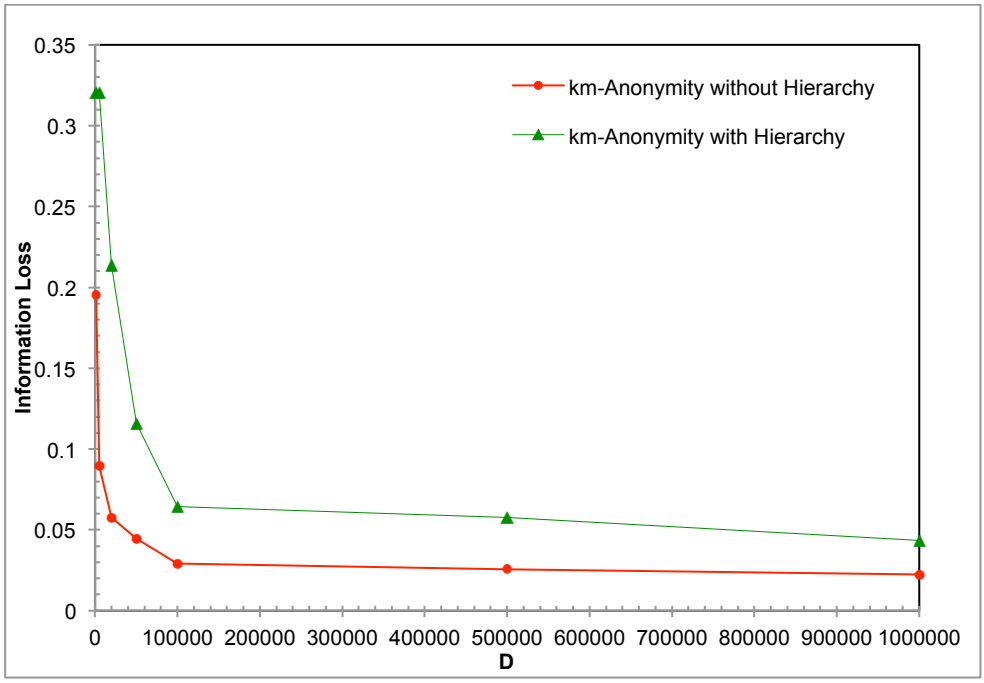
Στην Εικόνα 5.11 φαίνεται πως μεταβάλλεται η απώλεια πληροφορίας καθώς αυξάνεται η παράμετρος μέγιστης μεταβολής της ποινής d, στον αλγόριθμο χωρίς ιεραρχίες γενίκευσης.

Παρατηρείται αύξηση της απώλειας πληροφορίας για μεγαλύτερες τιμές του d, αφού όσο μεγαλύτερη είναι η επιτρεπτή μεταβολή της ποινής, γίνονται αποδεκτές γενικεύσεις οι οποίες προκαλούν μεγάλη απώλεια πληροφορίας. Από την άλλη όσο μικρότερη είναι η τιμή της παραμέτρου d, τόσο μικρότερη απώλεια πληροφορίας προκαλείται στη βάση σε κάθε νέα γενίκευση τιμών.



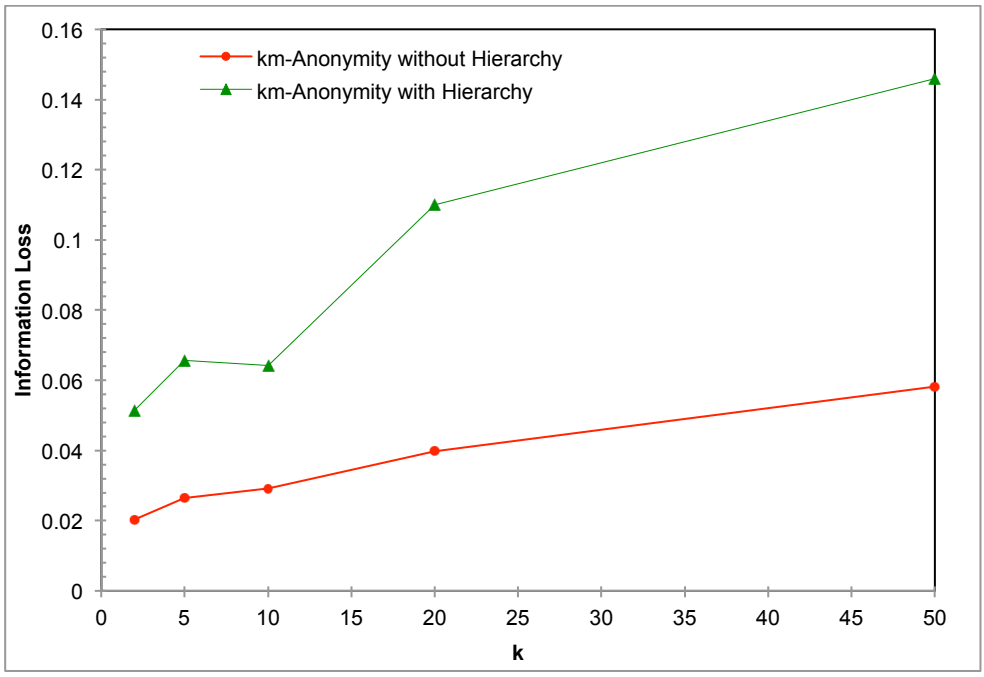
Εικόνα 5.11 Απώλεια πληροφορίας – παράμετρος μέγιστης μεταβολής NCP (k=10 |D|=100,000 d=0.001)

Στην Εικόνα 5.12 αναπαριστάται η απώλεια πληροφορίας που εμφανίζεται στα k^m-ανωνυμοποιημένα δεδομένα κατά την εκτέλεση των δύο αλγορίθμων για σύνολα δεδομένων διαφορετικού μεγέθους. Και στις δύο περιπτώσεις παρατηρείται αύξηση της απώλειας πληροφορίας για τα μικρότερα σύνολα δεδομένων, γεγονός που τεκμηριώνεται λογικά, αφού για λιγότερες εγγραφές, απαιτείται μεγαλύτερη γενίκευση, λόγω του ότι η πιθανότητα εμφάνισης κοινών τιμών στις εγγραφές είναι μικρή.

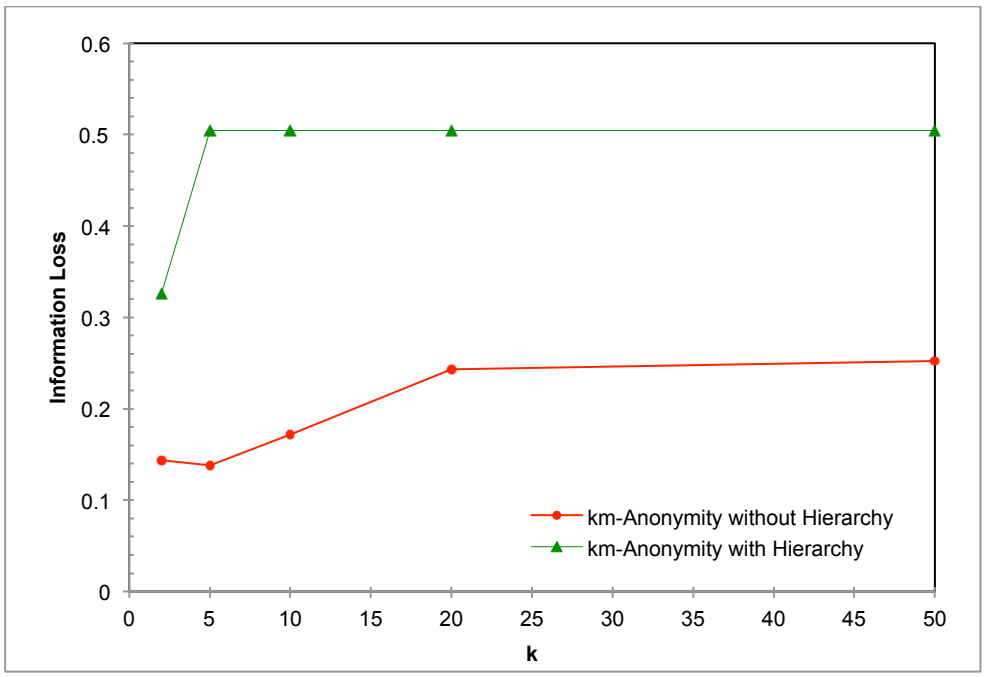


Εικόνα 5.12 Μετρική απώλειας πληροφορίας – πλήθος εγγραφών συνόλου (k=10 m=2 d=0.001)

Στις Εικόνες 5.13 και 5.14, παρουσιάζεται η απώλεια πληροφορίας που προκύπτει έπειτα από την εκτέλεση και των δύο αλγορίθμων πάνω στο ίδιο σύνολο δεδομένων 100,000 εγγραφών. Ο προτεινόμενος αλγόριθμος διατηρεί την Κανονικοποιημένη Ποινή Βεβαιότητας σε πάρα πολύ χαμηλά επίπεδα. Από την άλλη ο αλγόριθμος με τη χρήση ιεραρχιών αναγκάζεται να υπεργενικεύει τα δεδομένα με βάση το δέντρο ιεραρχίας. Όσο μεγαλύτερα διαστήματα τιμών υπάρχουν στο δέντρο, τόσο μεγαλύτερη είναι και η απώλεια πληροφορίας στη βάση.

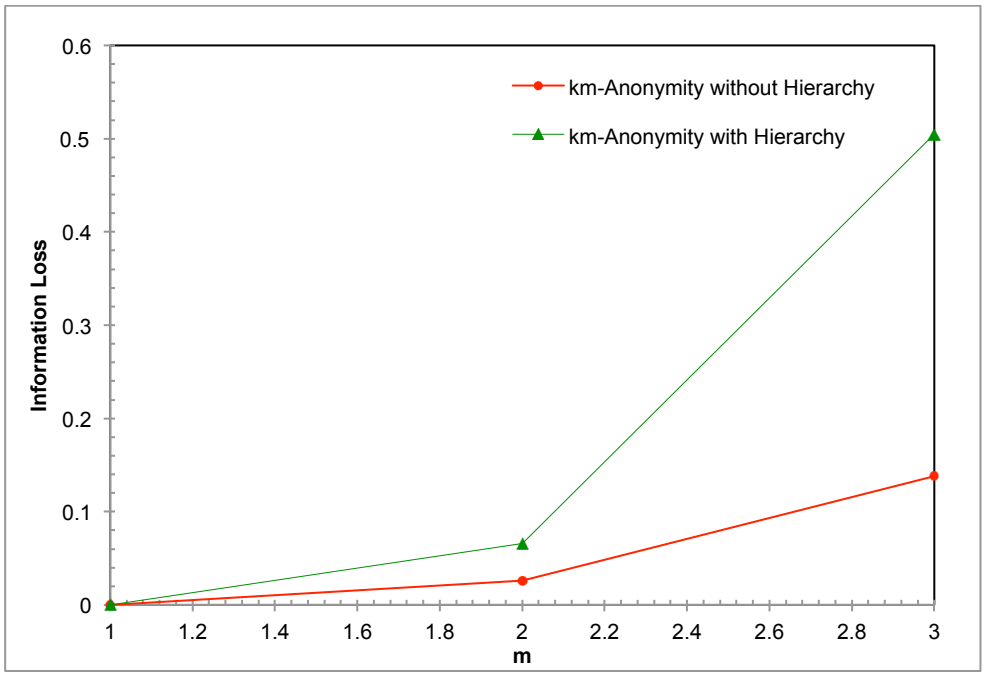


Εικόνα 5.13 Απώλεια πληροφορίας - παράμετρος k (|D|=100,000 m=2 d=0.001)

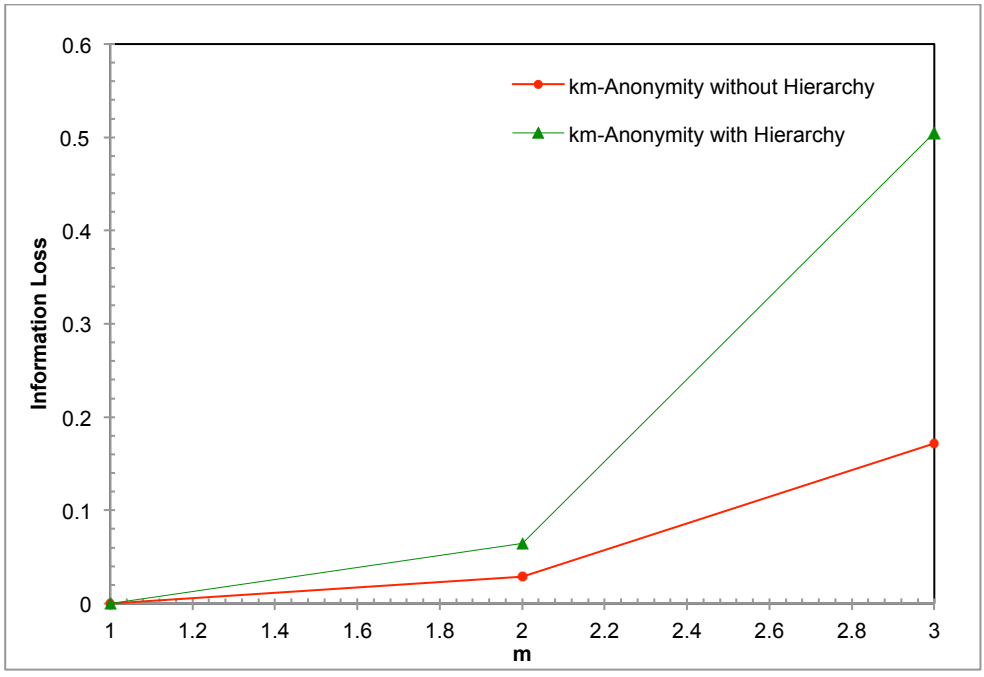


Εικόνα 5.14 Απώλεια πληροφορίας - παράμετρος k (|D|=100,000 m=3 d=0.001)

Τα δύο τελευταία πειράματα για τον έλεγχο της απώλειας πληροφορίας έγιναν συναρτήσει της μερικής γνώσης m που κατέχει ο επιτιθέμενος, σε σύνολο 100,000 εγγραφών. Και στις δύο περιπτώσεις, παρατηρείται αύξηση της απώλειας πληροφορίας για τις μεγαλύτερες τιμές της παραμέτρου m, γεγονός που τεκμηριώνεται λογικά, αφού για μεγαλύτερη γνώση m του επιτιθέμενου, απαιτείται μεγαλύτερη γενίκευση, λόγω του ότι η πιθανότητα εμφάνισης κοινών m συνδυασμών στις εγγραφές είναι μικρή.



Εικόνα 5.15 Απώλεια πληροφορίας - παράμετρος μερικής γνώσης m (|D|=100,000 k=5 d=0.001)

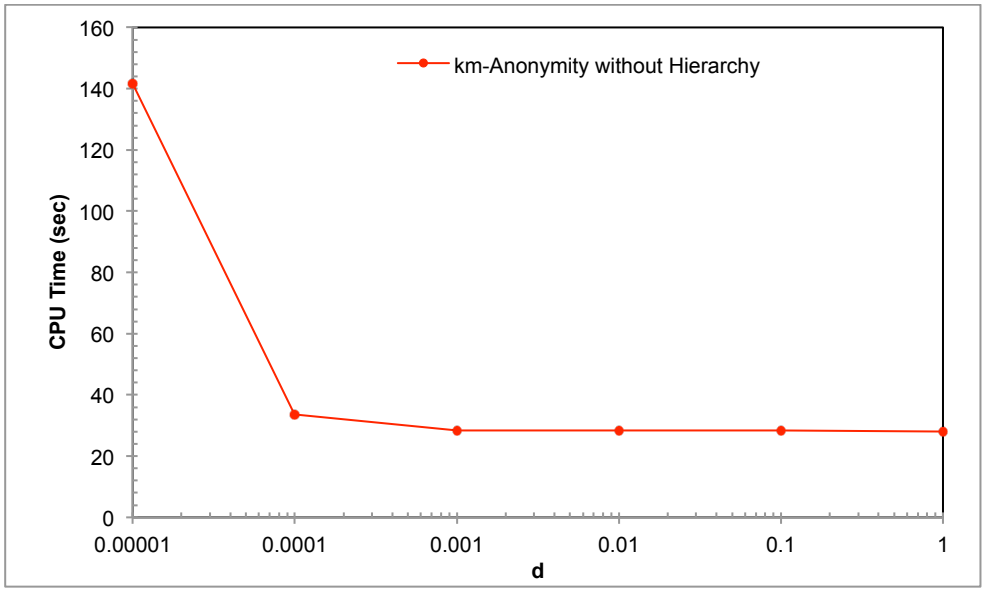


Εικόνα 5.16 Απώλεια πληροφορίας - παράμετρος μερικής γνώσης m (|D|=100,000 k=10 d=0.001)

5.3.2.2 Χρόνος εκτέλεσης

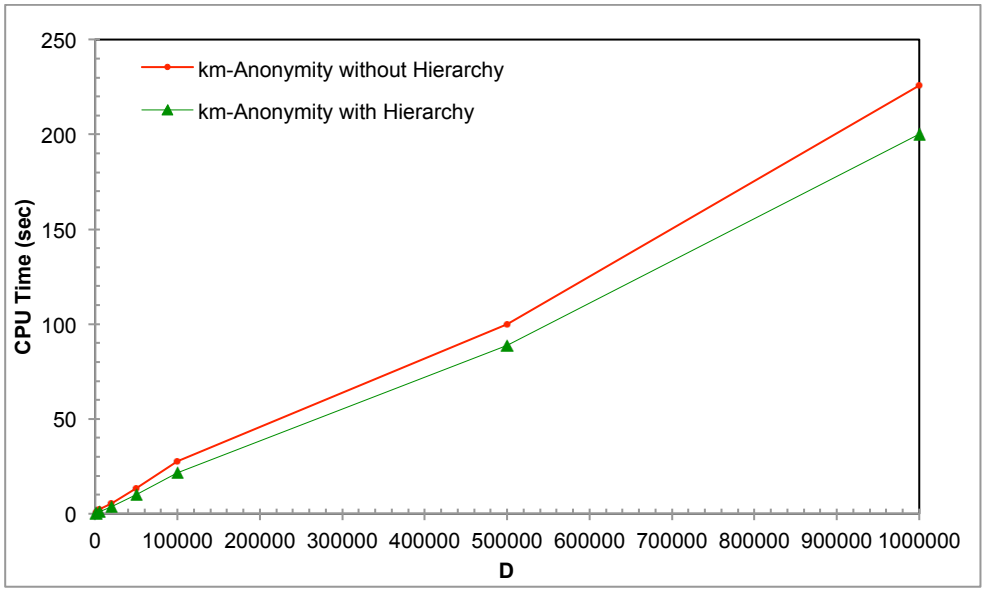
Στην επόμενη ενότητα παρουσιάζονται τα αποτελέσματα από τις διαφορετικές εκτελέσεις του αλγορίθμου αναφορικά με τον χρόνο εκτέλεσής του στα πραγματικά οικονομικά δεδομένα.

Στην Εικόνα 5.17 παρουσιάζεται η μεταβολή του χρόνου εκτέλεσης καθώς αυξάνεται η παράμετρος μέγιστης μεταβολής της ποινής d. Όπως ήταν αναμενόμενο, ο αλγόριθμος παρουσιάζει μεγαλύτερο χρόνο εκτέλεσης καθώς η παράμετρος d παίρνει μικρότερες τιμές.



Εικόνα 5.17 Χρόνος - παράμετρος μέγιστης μεταβολής ποινής NCP (|D|=100,000 k=2 m=2)

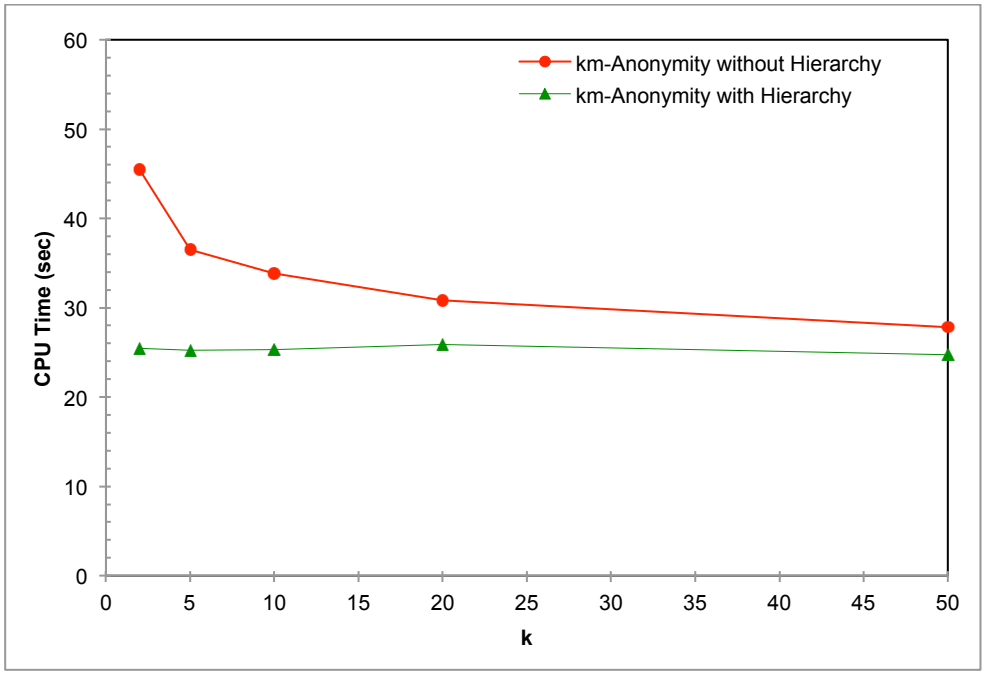
Στην Εικόνα 5.18 καταγράφονται οι χρόνοι εκτέλεσης των δύο αλγορίθμων για σύνολα δεδομένων διαφορετικού μεγέθους, και τιμές ανωνυμίας k=10 και m=2. Ο χρόνος εκτέλεσης και των δύο αλγορίθμων αυξάνεται, καθώς αυξάνεται το μέγεθος του συνόλου των εγγραφών.



Εικόνα 5.18 Χρόνος CPU – πλήθος εγγραφών συνόλου (k=10 m=2 d=0.001)

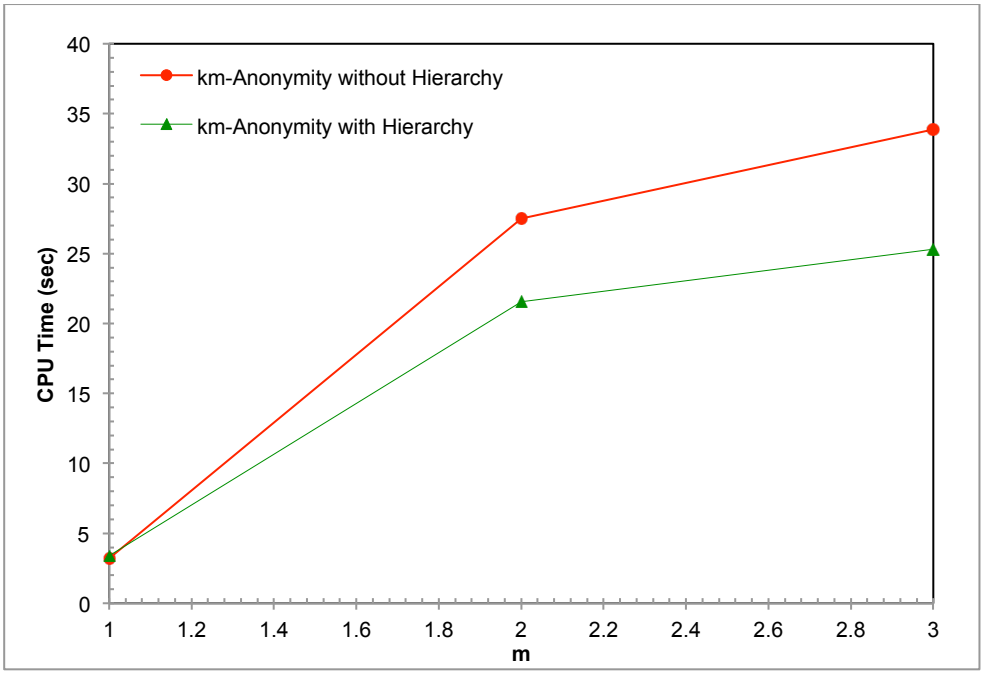
Στην Εικόνα 5.19 αποτυπώνεται ο χρόνος εκτέλεσης του αλγορίθμου, με είσοδό το σύνολο δεδομένων 100,000 εγγραφών και τιμές της παραμέτρου *k* από το σύνολο {2, 5, 10, 50}, με σταθερή παράμετρο μερικής γνώσης m=3.

Ο χρόνος εκτέλεσης αυξάνεται, όταν η παράμετρος ανωνυμίας *k* μειώνεται. Αυτό είναι αναμενόμενο αφού, η πολυπλοκότητα του αλγορίθμου εξαρτάται από τον αριθμό των κόμβων του δέντρου. Όσο μικρότερη είναι η παράμετρος ανωνυμίας k τόσο περισσότεροι κόμβοι υπάρχουν στο count tree, με αποτέλεσμα οι έλεγχοι που γίνονται για πιθανές γενικεύσεις σε κάθε επανάληψη να είναι περισσότεροι.



Εικόνα 5.19 Χρόνος CPU – παράμετρος ανωνυμίας k (m=3 |D|=100,000 d=0.001)

Στην Εικόνα 5.20 καταγράφονται οι χρόνοι εκτέλεσης των δύο αλγορίθμων για το σύνολο δεδομένων 100,000 εγγραφών, με παράμετρο ανωνυμίας k=10 και m={1, 2, 3}. Αυτό είναι αναμενόμενο γιατί καθώς η παράμετρος m παίρνει μεγαλύτερες τιμές, καταγράφονται στο δέντρο συχνοτήτων περισσότεροι συνδυασμοί τιμών, με αποτέλεσμα το πλήθος των κόμβων του δέντρου να είναι μεγαλύτερο. Ο αλγόριθμος της κλασσικής k^m-ανωνυμίας για m=3 έχει μικρότερους χρόνους εκτέλεσης γιατί, δημιουργώντας γενικεύσεις μεγαλύτερων διαστημάτων μειώνει το πλήθος των κόμβων του δέντρου συχνοτήτων με αποτέλεσμα να προκαλεί στο τελικό ανωνυμοποιημένο σύνολο δεδομένων περισσότερη απώλεια πληροφορίας σε αντίθεση με τον προτεινόμενο αλγόριθμο.



Εικόνα 5.20 Χρόνος CPU – παράμετρος μερικής γνώσης m (k=10 |D|=100,000 d=0.001)

6

Τεχνικές λεπτομέρειες

Στο Κεφάλαιο που ακολουθεί, παρουσιάζονται όλες οι τεχνικές λεπτομέρειες σχετικά με την υλοποίηση του προτεινόμενου αλγορίθμου όπως αυτός παρουσιάστηκε στις προηγούμενες ενότητες. Αναλύονται όλες οι δομές δεδομένων που χρησιμοποιήθηκαν, καθώς και όλες οι κλάσεις και οι αντίστοιχες μέθοδοι που αναπτύχθηκαν κατά τη διάρκεια της υλοποίησης. Τέλος, παρουσιάζονται και όλες οι τεχνικές λεπτομέρειες για την υλοποίηση του αλγόριθμου της κλασσικής k^m-ανωνυμίας, με τον οποίο συγκρίθηκε ο προτεινόμενος αλγόριθμος.

6.1 Λεπτομέρειες υλοποίησης

Για την ανάπτυξη του αλγόριθμου χρησιμοποιήθηκε η αντικειμενοστρεφής γλώσσα προγραμματισμού (object-oriented programming language) C++. Κατά τη διάρκεια της υλοποίησης του αλγόριθμου αλλά και για την διεξαγωγή των πειραμάτων χρησιμοποιήθηκε το ολοκληρωμένο περιβάλλον ανάπτυξης (Integrated Development Environment –IDE) Eclipse, με τον μεταγλωττιστή g++.

6.1.1 Μορφή δεδομένων εισόδου-εξόδου

Το αρχικό σύνολο δεδομένων D, εισάγεται στο πρόγραμμα με την μορφή απλού κειμένου. Κάθε γραμμή του κειμένου αντιστοιχεί σε μια εγγραφή, ενώ οι τιμές της εγγραφής αυτής χωρίζονται μεταξύ τους με κενό χαρακτήρα.

Μετά το πέρας εκτέλεσης του αλγόριθμου παράγεται ένα αρχείο απλού κειμένου με τα ανωνυμοποιημένα δεδομένα, τα οποία ικανοποιούν την k^m-ανωνυμία σύμφωνα με τις παραμέτρους ανωνυμίας που εισήγαγε ο χρήστης. Τα νέα γενικευμένα δεδομένα έχουν την μορφή ορίου τιμών (π.χ. [100, 200]).

6.1.2 Εισαγωγή παραμέτρων από το χρήστη

Ο χρήστης πριν τη διαδικασία ανωνυμοποίησης, καλείται να εισάγει στο πρόγραμμα τις διάφορες παραμέτρους ανωνυμίας:

- ***Τιμή παραμέτρου ανωνυμίας k:*** Η τιμή της παραμέτρου k σύμφωνα με την οποία απαιτούμε τα ανωνυμοποιημένα δεδομένα να ικανοποιούν την k^m-ανωνυμία ως προς την μερική γνώση του επιτιθέμενου. Η τιμή της παραμέτρου k πρέπει να είναι ακέραια και μεγαλύτερη από το μηδέν. ($k>0$)
- ***Τιμή παραμέτρου μερικής γνώσης m:*** Η τιμή της παραμέτρου m, σύμφωνα με την οποία καθορίζεται η μερική γνώση του επιτιθέμενου πάνω στα δεδομένα. Η παράμετρος αυτή αντιπροσωπεύει το μέγιστο μέγεθος τιμών που μπορεί να γνωρίζει ο επιτιθέμενος και με την οποία τα ανωνυμοποιημένα δεδομένα θα ικανοποιούν την k^m-ανωνυμία ως προς την μερική γνώση του επιτιθέμενου. Η τιμή της παραμέτρου m πρέπει να είναι ακέραια, μεγαλύτερη από το μηδέν και μικρότερη ή ίση με τον ακέραιο αριθμό n, που αντιπροσωπεύει το μεγαλύτερο μέγεθος εγγραφής στη βάση. ($m \in [1, n]$)
- ***Τιμή παραμέτρου μέγιστης μεταβολής κανονικοποιημένης ποινής βεβαιότητας d:*** Η μέγιστη μεταβολή της Κανονικοποιημένης Ποινής Βεβαιότητας κατά την διαδικασία της k^m-ανωνυμοποίησης. Η τιμή της παραμέτρου d πρέπει να είναι μεγαλύτερη από το 0 και μικρότερη ή ίση από το 1. ($d \in [0, 1]$)

6.1.3 Δομές Δεδομένων

Κατά την ανάπτυξη του αλγορίθμου με χρήση της γλώσσας C++ χρησιμοποιήθηκαν τόσο οι προσφερόμενες δομές δεδομένων από το σύνολο C++ Standard Template Library, καθώς και άλλες δομές δεδομένων πέρα από αυτές που βρίσκονται στην βασική βιβλιοθήκη της C++. Με τη χρήση των κατάλληλων δομών ο αλγόριθμος γίνεται πιο αποδοτικός, λόγω του ότι οι δομές ανταποκρίνονται καλύτερα στην φύση του προβλήματος.

- ***Δενδρική δομή tree:*** Οργανώνει τα δεδομένα σε μορφή δένδρου. Κάθε κόμβος συνδέεται με άλλους κόμβους του δέντρου είτε στο ίδιο επίπεδο (αδερφοί κόμβοι), είτε στο πιο κάτω επίπεδο (κόμβοι παιδιά). Δεν υπάρχει περιορισμός στο πόσα παιδιά μπορεί να έχει ένας κόμβος, σε αντίθεση με τα δυαδικά δέντρα αναζήτησης. Στην κορυφή του δένδρου υπάρχει ένας κόμβος που αποτελεί τη ρίζα. Η πρόσβαση στους κόμβους γίνεται μέσω δεικτών (pointers). Στην παρούσα εργασία η συγκεκριμένη δομή χρησιμοποιήθηκε για την υλοποίηση του δέντρου συχνοτήτων (count tree).
- ***Δυαδικό δέντρο αναζήτησης binarysearchtree:*** Οργανώνει τα δεδομένα σε μορφή δένδρου, με τον περιορισμό ότι κάθε κόμβος έχει το πολύ δύο παιδιά. Τα δυαδικά δέντρα είναι ταξινομημένα ως εξής: για κάθε κόμβο, το αριστερό υποδέντρο του περιέχει μόνο κόμβους με τιμές μικρότερες από αυτή του κόμβου-πατέρα και το δεξί υποδέντρο του περιέχει μόνο κόμβους με τιμές μεγαλύτερες από αυτή του κόμβου-πατέρα. Λόγω του συγκεκριμένου χαρακτηριστικού η αναζήτηση σε δυαδικό δέντρο είναι εύκολη και γρήγορη. Στην παρούσα υλοποίηση η δομή του δυαδικού δέντρου χρησιμοποιήθηκε στην ταξινόμηση των τιμών της κάθε εγγραφής και στην εύρεση των συχνοτήτων εμφάνισης τους.
- ***Ταξινομημένη δομή std::map:*** Συσχετίζει δεδομένα σε μορφής ζεύγους. Κάθε ζεύγος αποτελείται από μία τιμή-κλειδί (key) και την αντίστοιχη τιμή (value) της. Η δομή map ταξινομεί τις εισαγόμενες σε αυτήν τιμές βάσει της τιμής-κλειδί που έχουν, με χρήση δοσμένης συνάρτησης ταξινόμησης. Η πρόσβαση στα δεδομένα του map είναι πολύ γρήγορη αφού συνήθως οι δομές αυτές υλοποιούνται σε μορφή δυαδικού δένδρου. Στην παρούσα εργασία, η δομή map χρησιμοποιήθηκε σαν ένα ταξινομημένο σύνολο από δείκτες σε κόμβους του δέντρου συχνοτήτων. Ως τιμή-κλειδί χρησιμοποιήθηκε η τιμή του κόμβου και ως τιμή value ένα διάνυσμα (vector<iterators>) από δείκτες στους κόμβους του δέντρου με την τιμή αυτή. Με τη χρήση της map η πρόσβαση στους κόμβους του δέντρου συχνοτήτων είναι πολύ πιο αποδοτική και γρήγορη.
- ***Διάνυσμα std::vector*****:** Αποθηκεύει τα δεδομένα του ακολουθιακά όπως ακριβώς συμβαίνει και με τους πίνακες, με το μόνο πλεονέκτημα ότι το μέγεθος του vector μπορεί να αλλάξει δυναμικά κατά την διαδικασία εκτέλεσης. Χρησιμοποιήθηκε τόσο

για τον ορισμό μονοδιάστατων διανυσμάτων αλλά και για την υλοποίηση διανυσμάτων δύο διαστάσεων, διανύσματα στα οποία κάθε θέση τους περιείχε ένα άλλο διάνυσμα (vector<vector>). Η πρόσβαση στα περιεχόμενα του vector είναι πολύ πιο αποδοτική σε σχέση με άλλους τύπους δεδομένων (λίστες, ουρές, συνδεδεμένες λίστες κλπ) για αυτό και επιλέχθηκε στη παρούσα διπλωματική εργασία.

- ***Διάστημα τιμών range<int, int>*:** Αποτελείται από ένα ζεύγος ακεραίων (min, max) και χρησιμοποιήθηκε για την αναπαράσταση των διαστημάτων τιμών. Στη θέση min αποθηκεύεται ο μικρότερος αριθμός του διαστήματος και στη θέση max ο μεγαλύτερος αριθμός. Λόγω του ότι κατά τη διαδικασία της γενίκευσης, μια αριθμητική τιμή αντικαθίσταται από ένα διάστημα τιμών, η συγκεκριμένη δομή είναι απαραίτητη στη διαχείριση, σύγκριση και εκτύπωση των διαστημάτων τιμών κατά την εκτέλεση του προγράμματος.
- ***Κόμβος δένδρου συχνοτήτων node:*** Η συγκεκριμένη δομή, ομαδοποιεί τιμές διαφορετικών τύπων δεδομένων, προκειμένου να χρησιμοποιηθούν για τον κάθε κόμβο του δέντρου συχνοτήτων. Όπως περιγράφηκε και στο Κεφάλαιο 4, το δέντρο κρατά σε κάθε κόμβο, την τιμή ή το διάστημα τιμών της κάθε εγγραφής (range), το πλήθος των διαφορετικών εγγραφών που η τιμή συναντάται (integer), και τους αύξοντες αριθμοί των εγγραφών (vector). Με την ομαδοποίηση των τριών διαφορετικών αυτών τύπων δεδομένων είναι πιο εύκολη η διαχείριση των κόμβων του δέντρου συχνοτήτων.

6.2 Ανάλυση Κλάσεων

Κατά την υλοποίηση του αλγόριθμου αναπτύχθηκαν διάφορες κλάσεις προκειμένου να υπάρχει η σωστή αλληλεπίδραση μεταξύ των δομών και τύπων του προγράμματος. Πιο κάτω παρουσιάζονται οι κλάσεις αυτές και οι βασικές μέθοδοι τους.

6.2.1 Κλάση Αρχικοποίησης Initialize

Η κλάση αυτή περιλαμβάνει βασικές μεθόδους για την προετοιμασία της βάσης δεδομένων και την ταξινόμηση της, προκειμένου να γίνει πιο εύκολη η διαδικασία της ανωνυμοποίησης. Περιλαμβάνει παράλληλα και μεθόδους για την αρχικοποίηση του δέντρου συχνοτήτων και του map list.

- ***Initialize::CountFrequencies:*** Σαρώνει την βάση δεδομένων την πρώτη φορά και περνά τις τιμές κάθε εγγραφής σε ένα δυαδικό δέντρο αναζήτησης μαζί με το πλήθος

εμφάνισης της κάθε τιμής στη βάση και τον αριθμό των διαφορετικών εγγραφών που αυτή εμφανίζεται.

- ***Initialize::CountTreeInit:*** Δημιουργεί το δέντρο συχνοτήτων για m=1. Κάνοντας χρήση του δυαδικού δέντρου αναζήτησης που έχει δημιουργηθεί, η μέθοδος περνά στο δέντρο συχνοτήτων τις τιμές των εγγραφών ξεκινώντας από την πιο συχνά εμφανιζόμενη τιμή. Για κάθε εισαγωγή κόμβου στο δέντρο, ενημερώνει παράλληλα το map list με τους δείκτες του νέου κόμβου. Με το πέρας της συνάρτησης έχει δημιουργηθεί το δέντρο συχνοτήτων (count tree) με όλες τις τιμές των εγγραφών της βάσης για m=1 και παράλληλα η map list περιέχει όλους τους δείκτες των κόμβων του δέντρου.
- ***Initialize::SortDBbyFreq:*** Σαρώνει το αρχείο εισόδου και ταξινομεί την κάθε εγγραφή σύμφωνα με τις συχνότητες εμφάνισης της στη βάση δεδομένων ξεκινώντας από την πιο συχνά εμφανιζόμενη τιμή.

6.2.2 Κλάση Ανωνυμοποίησης km_Anonymize

Στην κλάση αυτή περιλαμβάνονται οι κύριες μέθοδοι για την διαδικασία της ανωνυμοποίησης της βάσης δεδομένων.

- ***km_Anonymize::Clustering:*** Μέθοδος η οποία για m=1 βρίσκει όλους τους κόμβους-φύλλα που παραβιάζουν την k^m-ανωνυμία και ανατρέχοντας στο δέντρο βρίσκει τις πιο φτηνές γενικεύσεις που μπορούν να γίνουν για την επίλυση του προβλήματος, ανεξάρτητα της παραμέτρου d. Με το πέρας της συνάρτησης ο αλγόριθμος έχει βρει την καλύτερη λύση (optimal) για m=1 και η βάση είναι ανωνυμοποιημένη με διαστήματα τιμών που ικανοποιούν την k^m-ανωνυμία για m=1.
- ***km_Anonymize::Generalize:*** Λαμβάνει σαν παράμετρο ένα καινούριο όριο γενίκευσης και εφαρμόζει ολική γενίκευση στις τιμές που ανήκουν στο όριο αυτό. Αντικαθιστά τους κόμβους του δέντρου συχνοτήτων που έχουν τιμή η οποία ανήκει στο όριο τιμών με το νέο διάστημα τιμών και παράλληλα κρατά ενημερωμένη τη δομή map με τα νέα διαστήματα τιμών.
- ***km_Anonymize::MergeNodes:*** Συνενώνει τους αδελφικούς κόμβους του δέντρου που έχουν το ίδιο διάστημα τιμών, κρατώντας πάντα ενημερωμένο το map list. Η μέθοδος με τις κατάλληλες εναλλαγές, κρατά ταξινομημένους τους αδελφικούς κόμβους ανάλογα με τις συχνότητες εμφάνισης τους στη βάση, ξεκινώντας από τις πιο συχνά εμφανιζόμενες τιμές.
- ***km_Anonymize::Anonymization:*** Μέθοος η οποία για m>1 βρίσκει όλους τους κόμβους-φύλλα που παραβιάζουν την k^m-ανωνυμία και ανατρέχοντας στο δέντρο

αναζητά γενικεύσεις οι οποίες μπορούν να επιλύσουν το πρόβλημα, ικανοποιώντας την παράμετρο d. Με το πέρας της μεθόδου η βάση δεδομένων είναι ανωνυμοποιημένη, με διαστήματα τιμών που ικανοποιούν την k^m-ανωνυμία.

6.2.3 Κλάση Υπολογισμού Κόστους Compute_Cost

Σε αυτή την κλάση περιλαμβάνονται οι βασικές μέθοδοι για τον υπολογισμό της Κανονικοποιημένης Ποινής Βεβαιότητας της βάσης δεδομένων τόσο για πιθανές γενικεύσεις, όσο και για γενικεύσεις που έχουν ήδη γίνει.

- ***Compute_Cost::Cost:*** Μέθοδος που δέχεται σαν όρισμα δύο ή περισσότερους κόμβους-φύλλα του δέντρου συχνοτήτων που πιθανόν να συνενωθούν, και υπολογίζει την απώλεια πληροφορίας που θα προκύψει από τη συνένωση αυτή, βάσει της Κανονικοποιημένης Ποινής Βεβαιότητας (NCP). Στην ποινή περιλαμβάνεται τόσο η ποινή για τη γενίκευση των κόμβων-φύλλων όσο και η ποινή για τη γενίκευση των προγόνων των κόμβων που θα συνενωθούν.
- ***Compute_Cost::ncp:*** Υπολογίζει τη Κανονικοποιημένη Ποινή Βεβαιότητας της βάσης με τις γενικεύσεις που έχουν γίνει στη βάση για την στιγμή που ένα αντικείμενο της κλάσης καλεί τη συγκεκριμένη μέθοδο.

6.3 Ανάλυση Βασικών Μεθόδων

6.3.1 Κύρια συνάρτηση

Η βασική συνάρτηση *main()* υλοποιεί το κύριο μέρος του αλγόριθμου, ελέγχοντας και συνδέοντας όλες τις επιμέρους συναρτήσεις μεταξύ τους. Οι κύριες λειτουργίες της συνάρτησης είναι:

- Διαβάζει τις επιλογές του χρήστη για την παράμετρο ανωνυμίας k, την παράμετρο μερικής γνώσης m, την μέγιστη επιτρεπτή ποινή d και ελέγχει αν οι τιμές που εισάγει ο χρήστης είναι μέσα στα επιτρεπτά όρια εμφανίζοντας τα κατάλληλα μηνύματα σφάλματος όπου είναι αναγκαίο.
- Δημιουργεί αντικείμενα της κλάσης αρχικοποίησης *Initialize*() προκειμένου να μετρήσει τις συχνότητες εμφάνισης κάθε εγγραφής και να ταξινομήσει τις εγγραφές της βάσης, σύμφωνα με τις συχνότητες αυτές. Παράλληλα δημιουργεί το δέντρο συχνοτήτων για m=1 και περνά τους δείκτες κάθε κόμβου του δέντρου στο ταξινομημένο σύνολο map.

- Στη συνέχεια δημιουργεί αντικείμενα της κλάσης *km_Anonymize*. Τα αντικείμενα αυτά χρησιμοποιούν τη μέθοδο *km_Anonymize::Clustering()* για την ομαδοποίηση των τιμών του δέντρου για m=1, δημιουργώντας όρια τιμών με τέτοιο τρόπο ώστε να ικανοποιείται η k^m-ανωνυμία για m=1.
- Με τη χρήση των αντικειμένων της κλάσης *km_Anonymize* καλειται η μέθοδος *km_Anonymize::Anonymization()* για την ανωνυμοποίηση της βάση για m>1.
- Τέλος, υπολογίζει την τελική Κανονικοποιημένη Ποινή Βεβαιότητας (NCP) της βάσης για το σύνολο των ανωνυμοποιημένων εγγραφών.

6.3.2 Διαδικασία αρχικοποίησης

Αντικείμενα της κλάσης *Initialize* δημιουργούνται προκειμένου να οργανώσουν και να αρχικοποιήσουν τα δεδομένα με τέτοιο τρόπο έτσι ώστε να είναι έτοιμα για επεξεργασία, σύμφωνα με τις μεθόδους που ορίζει η k^m-ανωνυμία.

- Αρχικά, με τη χρήση της μεθόδου *Initialize*::*CountFrequencies()* σαρώνεται μια φορά η βάση δεδομένων και καταγράφονται οι τιμές της κάθε εγγραφής σε ένα δυαδικό δέντρο αναζήτησης της κλάσης *BinarySearchTree*. Στο δέντρο αυτό καταγράφονται παράλληλα το πλήθος εμφάνισης της κάθε τιμής στη βάση και ο αριθμός των διαφορετικών εγγραφών που αυτή εμφανίζεται. Μέσω της μεθόδου αυτής καταγράφονται και άλλες πληροφορίες της βάσης δεδομένων όπως η μέγιστη και ελάχιστη τιμή του πεδίου τιμών, ο αριθμός των εγγραφών της βάσης, το μέγεθος της μεγαλύτερης εγγραφής προκειμένου να χρησιμοποιηθούν για τον υπολογισμό της Κανονικοποιημένης Ποινής Βεβαιότητας.
- Στη συνέχεια τα αντικείμενα της ίδιας κλάσης, μέσω της μεθόδου *Initialize*::*CountTreeInit()* δημιουργούν το δέντρο συχνοτήτων για m=1. Το δέντρο συχνοτήτων ανήκει στην κλάση *CountTree*. Για κάθε εισαγωγή κόμβου στο δέντρο, ενημερώνεται παράλληλα το map list με τους δείκτες του νέου κόμβου.
- Τέλος σαρώνεται για δεύτερη φορά η βάση δεδομένων. Με τη χρήση της μεθόδου *Initialize*::*SortDBbyFreq()* ταξινομούνται οι εγγραφές της βάσης, ξεκινώντας από την πιο συχνά εμφανιζόμενη τιμή.

6.3.3 Διαδικασία δημιουργίας συστάδων

Αρχικά ο αλγόριθμος ακολουθώντας την apriori ιδιότητα θα επιλύσει τα προβλήματα παραβίασης ιδιωτικότητας για m=1. Ουσιαστικά στο πρώτο βήμα, αντιμετωπίζει το πρόβλημα k^m-ανωνυμίας σαν ένα πρόβλημα k-ανωνυμίας. Για την επίλυση του,

δημιουργούνται συστάδες κόμβων (clusters), που έχουν πλήθος εμφάνισης (support) μεγαλύτερο ή ίσο με k και επιτυγχάνουν την μικρότερη δυνατή ποινή βεβαιότητας. Η κύρια συνάρτηση δημιουργεί αντικείμενα της κλάσης *km_Anonymity* και μέσω της μεθόδου *km_Anonymity::Clustering()* ακολουθείται η πιο κάτω επαναληπτική διαδικασία:

- Σαρώνεται το δέντρο συχνοτήτων και εντοπίζεται ο πρώτος προβληματικός κόμβος με support μικρότερο από k.
- Όλοι οι κόμβοι του δέντρου τοποθετούνται σε ένα διάνυσμα vector και ταξινομούνται ανάλογα με την τιμή τους.
- Δημιουργούνται υποσύνολα των αδελφικών κόμβων, όπως περιγράφηκε στο Κεφάλαιο 4, ξεκινώντας για μέγεθος υποσυνόλου ίσο με δύο, μέχρι να βρεθούν υποσύνολα κόμβων του ίδιου μεγέθους με support μεγαλύτερο από k.
- Υπολογίζονται οι τιμές των Κανονικοποιημένων Ποινών Βεβαιότητας για τα υποσύνολα ίδιου μεγέθους και διατηρείται αυτό με την μικρότερη ποινή. Αυτή θα είναι και η αποδεκτή γενίκευση.
- Τα αντικείμενα αυτά, μέσω της μεθόδου τους *km_Anonymity*::*Generalize()* εφαρμόζουν ολική γενίκευση στις τιμές του δέντρου αντικαθιστώντας τους κόμβους του δέντρου συχνοτήτων με τη νέα αποδεκτή γενικευμένη τιμή και παράλληλα ενημερώνοντας τη δομή map με τα νέα διαστήματα τιμών.
- Τέλος, μέσω της μεθόδου τους *MergeNodes()* συνενώνονται οι αδελφοί κόμβοι του δέντρου που τυγχάνει να έχουν το ίδιο διάστημα τιμών, κρατώντας πάντα ενημερωμένο το map list.

Η διαδικασία επαναλαμβάνεται μέχρι να αποκτήσουν όλοι οι κόμβοι στο δέντρο συχνοτήτων, πλήθος εμφάνισης (support) μεγαλύτερο ή ίσο με k.

6.3.4 Διαδικασία ανωνυμοποίησης

Σύμφωνα με την αρχή της apriori ιδιότητας, όπως περιγράφεται και στο Κεφάλαιο 4, ο αλγόριθμος ξεκινά και ανωνυμοποιεί τα δεδομένα πρώτα για *i=2* (λόγω της διαδικασίας Clustering παραλείπεται το βήμα *i=1*) και σταδιακά αυξάνει το *i* μέχρι την τελική τιμή *m*. Η κύρια συνάρτηση δημιουργεί αντικείμενα της κλάσης *km_Anonymity* και μέσω της μεθόδου *km_Anonymity::Anonymization()* ακολουθείται η πιο κάτω επαναληπτική διαδικασία:

- Σαρώνονται όλες οι εγγραφές της βάσης δεδομένων και τροποποιούνται, αντικαθιστώντας τυχόν παλιές τιμές με τα νέα γενικευμένα διαστήματα τιμών.
- Δημιουργούνται όλοι οι δυνατοί συνδυασμοί τιμών για κάθε εγγραφή και καταγράφονται στο δέντρο συχνοτήτων, κρατώντας ενημερωμένο παράλληλα και το map list με δείκτες σε κόμβους του δέντρου.

- Στη συνέχεια σαρώνονται τα φύλλα του δέντρου συχνοτήτων και εντοπίζεται ο πρώτος προβληματικός κόμβος που εμφανίζει support μικρότερο του k.
- Διασχίζοντας αναδρομικά το δέντρο αναζητούνται λύσεις στο πρόβλημα πρώτα με τυχόν συνενώσεις με αδελφικούς κόμβους-φύλλα και ύστερα με γειτονικούς κόμβους-φύλλα, έτσι ώστε το support του προβληματικού κόμβου να γίνει μεγαλύτερο ή ίσο με k.
- Για κάθε πιθανή γενίκευση, δημιουργούνται αντικείμενα της κλάσης *Compute_Cost* και μέσω της βασικής μεθόδου *Compute_Cost*::*Cost()* υπολογίζεται το κόστος συνένωσης, τόσο των κόμβων φύλλων όσο και των προγόνων τους.
- Τα αντικείμενα της κλάσης *km_Anonymity*, μέσω της μεθόδου τους *km_Anonymity*::*Generalize()* εφαρμόζουν ολική γενίκευση στις τιμές του δέντρου αντικαθιστώντας τους κόμβους του δέντρου συχνοτήτων με τη νέα αποδεκτή γενικευμένη τιμή και παράλληλα ενημερώνοντας τη δομή map με τα νέα διαστήματα τιμών.
- Τέλος, μέσω της μεθόδου τους *MergeNodes()* συνενώνονται οι αδελφοί κόμβοι του δέντρου που τυγχάνει να έχουν το ίδιο διάστημα τιμών, κρατώντας πάντα ενημερωμένο το map list.

6.3.5 Διαδικασία υπολογισμού μετρικής απώλειας πληροφορίας

Τα αντικείμενα της κλάσης *Compute_Cost* μέσω της μεθόδου *Compute_Cost*::*Cost()* ελέγχουν το κόστος μιας πιθανής γενίκευσης για m>1. Η μέθοδος παίρνει σαν όρισμα διανύσματα από κόμβους στο δέντρο που πρέπει να γενικευτούν. Βάσει αυτών των κόμβων βρίσκει τις πιθανές γενικεύσεις που πρέπει να γίνουν και υπολογίζει το NCP για τις γενικεύσεις αυτές. Στη ποινή περιλαμβάνεται τόσο η ποινή για τη γενίκευση των κόμβων-φύλλων όσο και η ποινή για τη γενίκευση των προγόνων των κόμβων που θα συνενωθούν. Σκοπός της μεθόδου είναι ο υπολογισμός της πιο κάτω πράξης για κάθε γενίκευση:

$$NCP([yi, zi]) = \begin{cases} 0, & z_i = y_i \\ \frac{z_i - y_i}{|I|}, & z_i \neq y_i \end{cases}$$

όπου $|I|$ το μέγεθος του πεδίου τιμών *I* των γνωρισμάτων της κάθε εγγραφής.

Για να το πετύχει αυτό ανατρέχει στο map list και εντοπίζει όλα τα διαστήματα τιμών $[y_i, z_i]$ στο σύνολο των δεδομένων. Για κάθε ένα από αυτά διαιρεί με το μέγεθος του πεδίου τιμών *I* που έχει υπολογίσει στην αρχή του προγράμματος. Αποθηκεύει τις τιμές για κάθε ένα όριο σε ένα διάνυσμα vector.

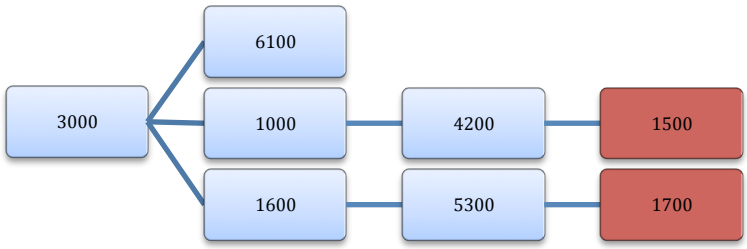
Για να βρει την Κανονικοποιημένη Ποινή Βεβαιότητας του συνόλου των δεδομένων πρέπει να υπολογίσει την πράξη:

$$NCP(D) = \frac{\sum_{vi \in D}(C_{v_i} \cdot NPC(v_i))}{\sum_{vi \in D}(C_{v_i})}$$

όπου C_{v_i} η συχνότητα εμφάνισης της κάθε τιμής v_i στη βάση δεδομένων.

Η μέθοδος παίρνει από το διάνυσμα vector την ποινή κάθε γενίκευσης *NCP([y_i, z_i])* και την πολλαπλασιάζει με τις συχνότητες εμφάνισης της κάθε τιμής που ανήκει στη γενίκευση οι οποίες έχουν υπολογιστεί στην αρχή της εκτέλεσης. Αθροίζει όλες τιμές ($C_{v_i} \cdot NPC([yi, zi])$) και διαιρεί με το συνολικό πλήθος συχνοτήτων.

Στο παράδειγμα που ακολουθεί, αν σε κάποιο βήμα ο αλγόριθμος επιδιώκει τη συνένωση των κόμβων {1500, 1700} θα πρέπει να ελέγξει για τις ποινές συνένωσης και των προγόνων των κόμβων αυτών {4200, 5300} και {1600, 1000}. Σε αυτή την περίπτωση ο αλγόριθμος επιδιώκει μέσω των αντικειμένων της κλάσης *Compute_Cost* και με τη χρήση της μεθόδου *Compute_Cost::Cost()* με όρισμα τα διανύσματα τιμών ({1500, 1700}, {4200, 5300}, {1600, 1000}) να υπολογίζει την ποινή για αυτή τη συνένωση.



Σχήμα 6.1 Παράδειγμα χρήσης της Compute_Cost::Cost()

Η μέθοδος στη συνέχεια θα ελέγξει τις γενικεύσεις και θα καταλήξει στις εξής δύο [1000, 1700] και [4200, 5300] αφού τα άλλα διαστήματα τιμών περιλαμβάνονται σε αυτά.

Αφού υπολογιστούν οι Κανονικοποιημένες Ποινές Βεβαιότητας για τις δύο γενικεύσεις NCP([1000, 1700]) και NCP([4200, 5300]) θα βρεθεί η Κανονικοποιημένη Ποινή Βεβαιότητας της βάσης και θα επιστραφεί μαζί με τα πιθανά όρια γενίκευσης.

6.4 Αλγόριθμος k^m-ανωνυμίας (με ιεραρχίες γενίκευσης)

6.4.1 Λεπτομέρειες υλοποίησης

Προκειμένου να συγκριθεί ο αλγόριθμος της παρούσας διπλωματικής εργασίας με αυτόν της κλασσικής k^m-ανωνυμίας με χρήση ιεραρχιών γενίκευσης, υλοποιήθηκε και δεύτερο πρόγραμμα στην αντικειμενοστρεφή γλώσσα προγραμματισμού C++. Χρησιμοποιήθηκε και πάλι το ολοκληρωμένο περιβάλλον ανάπτυξης (Integrated Development Environment –IDE) Eclipse, με τον μεταγλωττιστή g++.

Το αρχικό σύνολο δεδομένων D, εισάγεται στο πρόγραμμα με την μορφή απλού κειμένου. Κάθε γραμμή του κειμένου αντιστοιχεί σε μια εγγραφή, ενώ οι τιμές της εγγραφής αυτής χωρίζονται μεταξύ τους με τον κενό χαρακτήρα. Μετά το πέρας εκτέλεσης του αλγόριθμου παράγεται ένα αρχείο απλού κειμένου με τα ανωνυμοποιημένα δεδομένα, τα οποία ικανοποιούν την k^m-ανωνυμία σύμφωνα με τις παραμέτρους ανωνυμίας που εισήγαγε ο χρήστης.

Το πρόγραμμα δέχεται από το χρήστη τις παραμέτρους ανωνυμίας *k* και *m*, χωρίς να λαμβάνει την παράμετρο *d* για τη μέγιστη επιτρεπτή μεταβολή της Κανονικοποιημένης Ποινής Βεβαιότητας, όπως συνέβαινε στο προηγούμενο αλγόριθμο. Για το δέντρο ιεραρχίας γενίκευσης δίνεται ο βαθμός *(degree)* του κάθε κόμβου του δέντρου. Ο βαθμός αυτός συμβολίζει τον αριθμό των τιμών που γενικεύονται στο δέντρο. Για παράδειγμα αν αποφασίσουμε δέντρο ιεραρχίας με βαθμό ίσο με *n,* τότε κάθε γενίκευση από το ένα επίπεδο στο άλλο γενικεύει *n* τιμές. Εάν το μέγεθος του πεδίου τιμών δεν διαιρείται ακριβώς με το *n,* δημιουργούνται μικρότερες κλάσεις γενίκευσης.

Κατά την ανάπτυξη του δεύτερου αλγορίθμου χρησιμοποιήθηκαν τόσο οι προσφερόμενες δομές δεδομένων από το σύνολο C++ Standard Template Library, καθώς και άλλες δομές δεδομένων πέρα από αυτές που βρίσκονται στην βασική βιβλιοθήκη, όπως ακριβώς περιγράφονται στην υποενότητα 6.1.3. Για το δέντρο ιεραρχίας γενίκευσης χρησιμοποιήθηκε η δενδρική δομή tree της κλάσης tree.

Εκτός από τις δομές δεδομένων και τις βασικές μεθόδους τους, κατά την υλοποίηση του αλγόριθμου αναπτύχθηκαν οι ίδιες κλάσεις που χρησιμοποιήθηκαν και στην υλοποίηση του προηγούμενου αλγόριθμου με προσθήκη της μεθόδου *Initialize::Hierarchy()* στην κλάση αρχικοποίησης *Initialize.* Τα αντικείμενα της κλάσης μέσω αυτής της μεθόδου, καθορίζουν την ιεραρχία γενίκευσης για όλες τις τιμές των εγγραφών της βάσης δεδομένων.

Η κλάση *Compute_Cost* δεν χρησιμοποιήθηκε στην υλοποίηση του δεύτερου προγράμματος, αφού ο αλγόριθμος δεν αποφασίζει βάσει της Κανονικοποιημένης Ποινής Βεβαιότητας για το

ποια θα είναι η επόμενη γενίκευση, αλλά ακολουθεί τα βήματα του δέντρου ιεραρχίας γενίκευσης.

6.4.2 Ανάλυση βασικών κλάσεων και μεθόδων

6.4.2.1 Κύρια συνάρτηση

Οι κύριες λειτουργίες του αλγόριθμου έχουν ως εξής:

- Διαβάζει τις επιλογές του χρήστη για την παράμετρο ανωνυμίας k, την παράμετρο μερικής γνώσης m, το βαθμό του δέντρου ιεραρχίας και ελέγχει αν οι τιμές που εισάγει ο χρήστης είναι μέσα στα επιτρεπτά όρια, εμφανίζοντας τα κατάλληλα μηνύματα σφάλματος όπου είναι αναγκαίο.
- Με αντικείμενα της κλάσης *Initializ*e, μετρά τις συχνότητες εμφάνισης κάθε εγγραφής και ταξινομεί τις εγγραφές της βάσης, σύμφωνα με τις συχνότητες αυτές. Παράλληλα δημιουργεί το δέντρο συχνοτήτων για m=1 και περνά τους δείκτες κάθε κόμβου του δέντρου στο ταξινομημένο σύνολο map.
- Μέσω της μεθόδου *Initializ*e*::Hierarchy()* δημιουργεί το δέντρο ιεραρχίας γενίκευσης βάσει του βαθμού του δέντρου που δόθηκε από το χρήστη.
- Στη συνέχεια δημιουργεί αντικείμενα της κλάσης *km_Anonymize.* Τα αντικείμενα αυτά χρησιμοποιούν τη μέθοδο *km_Anonymize::Clustering()* για την ομαδοποίηση των τιμών του δέντρου για m=1, δημιουργώντας όρια τιμών με τέτοιο τρόπο ώστε να ικανοποιείται η k^m-ανωνυμία για m=1.
- Με τη χρήση των αντικειμένων της κλάσης *km_Anonymize* καλείται η μέθοδος *km_Anonymize::Anonymization()* για την ανωνυμοποίηση της βάση για m>1.
- Υπολογίζει την τελική Κανονικοποιημένη Ποινή Βεβαιότητας (NCP) της βάσης για το σύνολο των ανωνυμοποιημένων εγγραφών.

6.4.2.2 Κλάση αρχικοποίησης Initialize

Αντικείμενα της κλάσης *Initialize* δημιουργούνται, προκειμένου το πρόγραμμα να οργανώσει και να αρχικοποιήσει τα δεδομένα με τέτοιο τρόπο έτσι ώστε να είναι έτοιμα για επεξεργασία, σύμφωνα με τις μεθόδους που ορίζει η k^m-ανωνυμία. Η συγκεκριμένη κλάση έχει τις ίδιες μεθόδους με το προηγούμενο πρόγραμμα, με τη διαφορά ότι προστίθεται η μέθοδος *Initialize::Hierarchy()* την οποία χρησιμοποιούν τα αντικείμενα της κλάσης για τη δημιουργία του δέντρου ιεραρχίας γενίκευσης.

6.4.2.3 Διαδικασία Clustering

Όπως και στο προηγούμενο πρόγραμμα, ο αλγόριθμος ακολουθώντας την apriori ιδιότητα θα επιλύσει τα προβλήματα παραβίασης ιδιωτικότητας για m=1. Για την επίλυση του δημιουργούνται συστάδες κόμβων (clusters), που έχουν πλήθος εμφάνισης (support) μεγαλύτερο ή ίσο με k και επιτυγχάνουν την μικρότερη δυνατή ποινή βεβαιότητας.

Ο αλγόριθμος ακολουθεί επαναληπτική διαδικασία:

- Σαρώνει το δέντρο συχνοτήτων και βρίσκει τον πρώτο προβληματικό κόμβο με support μικρότερο από k.
- Ανατρέχει στο δέντρο ιεραρχίας γενίκευσης και βρίσκει το διάστημα τιμών που θα γενικευτεί ο προβληματικός κόμβος.
- Δημιουργεί αντικείμενα της κλάσης *km_Anonymity* και μέσω της μεθόδου *km_Anonymity*::*Generalize()* εφαρμόζει ολική γενίκευση στις τιμές του δέντρου αντικαθιστώντας τους κόμβους του δέντρου συχνοτήτων με τη νέα γενικευμένη τιμή, σύμφωνα με το δέντρο ιεραρχίας γενίκευσης και παράλληλα κρατά ενημερωμένη τη δομή map με τα νέα διαστήματα τιμών.
- Τέλος, με τη χρήση της μεθόδου *km_Anonymity::MergeNodes()* συνενώνει αδελφικούς κόμβους του δέντρου που έχουν το ίδιο διάστημα τιμών, κρατώντας πάντα ενημερωμένο το map list.

Η διαδικασία επαναλαμβάνεται μέχρι να αποκτήσουν όλοι οι κόμβοι στο δέντρο συχνοτήτων, πλήθος εμφάνισης (support) μεγαλύτερο ή ίσο με k.

6.4.2.4 Διαδικασία ανωνυμοποίησης

Ακολουθώντας τα πρότυπα του πρώτου αλγόριθμου, η διαδικασία ανωνυμοποίησης ξεκινά και τροποποιεί τα δεδομένα πρώτα για *i=2* και σταδιακά αυξάνει το i μέχρι την τελική τιμή m.

Ο αλγόριθμος για κάθε βήμα του *i* ακολουθεί επαναληπτική διαδικασία:

- Αρχικά διαβάζει όλες τις εγγραφές από τη βάση δεδομένων και τροποποιεί την κάθε εγγραφή αντικαθιστώντας τις παλιές τιμές με τα νέα γενικευμένα διαστήματα τιμών αν υπάρχουν.
- Δημιουργεί τους δυνατούς συνδυασμούς τιμών για κάθε εγγραφή και τοποθετεί τους συνδυασμούς αυτούς στο δέντρο συχνοτήτων, κρατώντας ενημερωμένο παράλληλα και το map list με δείκτες σε κόμβους του δέντρου.
- Στη συνέχεια σαρώνει τα φύλλα του δέντρου συχνοτήτων και βρίσκει τον πρώτο προβληματικό κόμβο που εμφανίζει support μικρότερο του k.

- Ανατρέχει στο δέντρο ιεραρχίας γενίκευσης και βρίσκει το διάστημα τιμών που θα γενικευτεί ο προβληματικός κόμβος.
- Τα αντικείμενα της κλάσης *km_Anonymity*, μέσω της μεθόδου τους *km_Anonymity*::*Generalize()* εφαρμόζουν ολική γενίκευση στις τιμές του δέντρου αντικαθιστώντας τους κόμβους του δέντρου συχνοτήτων με τη νέα γενικευμένη τιμή και παράλληλα ενημερώνοντας τη δομή map με τα νέα διαστήματα τιμών.
- Τέλος, μέσω της μεθόδου τους *MergeNodes()* συνενώνονται οι αδελφοί κόμβοι του δέντρου που τυγχάνει να έχουν το ίδιο διάστημα τιμών, κρατώντας πάντα ενημερωμένο το map list.

7

Επίλογος

7.1 Σύνοψη και συμπεράσματα

Η συγκεκριμένη διπλωματική εργασία, ασχολήθηκε με το πρόβλημα της διασφάλισης της ιδιωτικότητας των εγγραφών σε συλλογές δεδομένων με συνεχή γνωρίσματα. Θεωρήθηκε η περίπτωση επίθεσης κατά την οποία ο επιτιθέμενος έχει μερική γνώση τιμών μιας εγγραφής του συνόλου δεδομένων.

Αναπτύχθηκε ευριστικός αλγόριθμος που εγγυάται την ικανοποίηση της k^m-ανωνυμίας του συνόλου των δεδομένων. Ο αλγόριθμος βασίζεται στην χρήση της ολικής γενίκευσης χωρίς τη χρήση κάποιας ιεραρχίας. Παράλληλα αναπτύχθηκε και δεύτερος αλγόριθμος k^m-ανωνυμίας με χρήση ιεραρχιών γενίκευσης. Η απόδοσή των δύο αλγορίθμων εξετάστηκε ως προς την μετρική απώλειας πληροφορίας (Κανονικοποιημένη Ποινή Βεβαιότητας) και τον χρόνο εκτέλεσης τους σε διαφορετικά δεδομένα εισόδου.

Όπως φαίνεται και από τα αποτελέσματα, η υπεροχή του αλγορίθμου που παρουσιάζεται για το πρόβλημα αυτό είναι ξεκάθαρη, καθώς επιτυγχάνει πολύ μικρότερη απώλεια πληροφορίας από τον αλγόριθμο της k^m-ανωνυμίας με χρήση ιεραρχίας. Αυτό οφείλεται στο γεγονός ότι ο προτεινόμενος αλγόριθμος εκμεταλλεύεται τις ιδιότητες των συνεχών γνωρισμάτων της βάσης.

Συμπεραίνεται πως για το πρόβλημα της προστασίας της ιδιωτικότητας σε σύνολα δεδομένων της παραπάνω μορφής από επιθέσεις με μερική γνώση σε κάποιες τιμές των γνωρισμάτων

του ψευδο-αναγνωριστικού μιας εγγραφής, ο αλγόριθμος που αναπτύχθηκε επιτυγχάνει μικρότερη απώλεια πληροφορίας συγκριτικά με τον αλγόριθμο της k^m-ανωνυμίας με χρήση ιεραρχιών γενίκευσης.

7.2 Μελλοντικές επεκτάσεις

Μια χρήσιμη επέκταση του αλγόριθμου είναι η χρήση τοπικής γενίκευσης στο σύνολο δεδομένων. Αυτό θα έχει σαν αποτέλεσμα την μείωση της απώλειας πληροφορίας στα δημοσιευμένα σύνολα.

Εκτός αυτού, ο αλγόριθμος λόγω της πρακτικής χρησιμότητας του μπορεί να επεκταθεί και σε διαφορετικά μοντέλα επιθέσεων. Μια χρήσιμη επέκταση αφορά την μελέτη επιθέσεων του επιτιθέμενου με σύνθετη μερική και συναθροιστική γνώση. Σε αυτό το μοντέλο, ο επιτιθέμενος έχει σαν γνωστικό υπόβαθρο (i) ένα σύνολο m τιμών μιας εγγραφής του συνόλου δεδομένων και (ii) μια συναθροιστική γνώση πάνω στο σύνολο των γνωρισμάτων του ψευδο-αναγνωριστικού.

Χαρακτηριστικό παράδειγμα του συγκεκριμένου προβλήματος εμφανίζεται κατά τη δημοσίευση των φορολογικών δεδομένων που περιέχουν τα εισοδήματα φορολογούμενων πολιτών. Σε περίπτωση της δημοσίευσης των αρχικών τιμών υπάρχει ο κίνδυνος αναγνώρισης κάποιας εγγραφής από όποιον επιτιθέμενο γνωρίζει ένα μέρος των εισοδημάτων ενός φυσικού προσώπου και ταυτόχρονα γνωρίζει και το συνολικό του εισόδημα.

Τέλος, ένα διαφορετικό μοντέλο επίθεσης θα μπορούσε να αφορά ένα σύνολο δεδομένων τέτοιο ώστε ένα ή περισσότερα γνωρίσματα του ψευδο-αναγνωριστικού να είναι ευαίσθητα. Σε αυτήν την περίπτωση μπορεί να επιχειρηθεί η ικανοποίηση της *l*-διαφορετικότητας για τα ευαίσθητα γνωρίσματα σε συνδυασμό με την k^m-ανωνυμία.

8

Βιβλιογραφία

[LDR05] K. Le Fevre, D. J. De Witt, R. Ramakrishnan, *Incognito: Efficient Full-Domain k-Anonymity*, In Proc. Special Interest Groupon Management of Data, 2005.

[LDR06] K. Le Fevre, D. J. De Witt, R. Ramakrishnan, *Mondrian Multidimensional k-Anonymity*, In Proc. Intl. Conference on Data Engineering, 2006.

[LLV07] N. Li, T. Li, S. Venkatasubramanian. *t-Closeness: Privacy Beyond k-Anonymity and l-Diversity*, InProc. International Conference on Data Engineering, 2007.

[MGK+06] A. Machanavajjhala, J. Gehrke, D. Kifer, and M. Venkitasubramaniam. *l-diversity: Privacy beyond k-anonymity*. In Proc. 22nd International Conference of Data Engineering (ICDE), 2006.

[NAC07] M.E. Nergiz, M. Atzori, C. Clifton. *Hiding the Presence of Individuals from Shared Databases*, In Proc. Special Interest Group on Management of Data, 2007

[Swe00] L. Sweeney. *Uniqueness of Simple Demographics in the U.S. Population*, LIDAPWP4.Carnegie Mellon University, Laboratory for International Data Privacy, Pittsburgh, 2000.

[Swe02] L. Sweeney. *k-anonymity: a model for protecting privacy*. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems (vol.10 no.5), 2002.

[TMK08] M. Terrovitis, N. Mamoulis and P. Kalnis, "*Privacy-preserving Anonymization of Set-valued Data*", PVLDB, vol.1, no.1, 2008.

[XT06] X. Xiao, Y. Tao. *Anatomy: Simple and Effective Privacy Preservation*, In Proc. Very Large Data Bases, 2006.

[XT07] X. Xiao, Y. Tao. *m-Invariance: Towards Privacy Preserving Re-publication of Dynamic Datasets*, In Proc. Special Interest Group on Management of Data, 2007.

[XWP+06] J. Xu, W. Wang, J. Pei, X. Wang, B. Shi, A. Fu, *Utility-Based Anonymization Using Local Recoding*, KDD, 2006

[1] "UCI Machine Learning Repository"
http://archive.ics.uci.edu/ml/datasets/US+Census+Data+%281990%29