BALANCING BETWEEN DATA UTILITY AND PRIVACY PRESERVATION IN DATA MINING

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Bachelor of Technology
In
Computer Science and Engineering

By

ANKIT TANDON
SACHIN KUMAR JAIN

Under the Guidance of
Prof. S.K. JENA

Department of Computer Science Engineering
National Institute of Technology
Rourkela
2010

National Institute of Technology

Rourkela

CERTIFICATE

This is to certify that the thesis entitled, "Balancing between Data Utility and Privacy preservation" submitted by Sachin kumar Jain and Ankit Tandon in partial fulfillments for the requirements for the award of Bachelor of Technology Degree in Computer Science and Engineering at National Institute of Technology, Rourkela is an authentic work carried out by them under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University / Institute for the award of any Degree or Diploma.

Date:

Prof. S. K. Jena
Dept. of Computer Science Engineering
National Institute of Technology
Rourkela - 769008

ACKNOWLEDGEMENT

On the submission of my Thesis report, we would like to extend our gratitude & sincere thanks to our supervisor Prof. S.K.Jena, Professor, Department of Computer Science and Engineering, NIT Rourkela for his constant motivation and support during the course of my work in the last one year. I truly appreciate and value his esteemed guidance and encouragement from the beginning to the end of this thesis. He has been my source of inspiration throughout the thesis work. A special acknowledgement goes to Prof. Korra Sathya Babu for extending his support during entire duration of the project and giving us insights into the subject matter. We would also like to convey our sincerest gratitude and indebtedness to all other faculty members and staff of the Department of Computer Science and Engineering, NIT Rourkela, who bestowed their great effort and guidance at appropriate times without which it would have been very difficult on our part to finish the project work.

Sachin Kumar Jain
Roll No: 10606046

Ankit Tandon
Roll No: 10606050

CONTENTS

List of Tables

List of Figures

ABSTRACT

Data Mining plays a vital role in today’s information world where it has been widely applied in various organizations. The current trend needs to share data for mutual benefit. However, there has been a lot of concern over privacy in the recent years .It has also raised a potential threat of revealing sensitive data of an individual when the data is released publically. Various methods have been proposed to tackle the privacy preservation problem like anonymization and perturbation. But the natural consequence of privacy preservation is information loss. The loss of specific information about certain individuals may affect the data quality and in extreme case the data may become completely useless. There are methods like cryptography which completely anonymize the dataset and which renders the dataset useless. So the utility of the data is completely lost. We need to protect the private information and preserve the data utility as much as possible. So the objective of the thesis is to find an optimum balance between privacy and utility while publishing dataset of any organization. Privacy preservation is hard requirement that must be satisfied and utility is the measure to be optimized.

One of the methods for preserving privacy is K-anonymization which also preserves privacy to a good extent. K-anonymity demands that every tuple in the dataset released be indistinguishably related to no fewer than k respondents. We used K-means algorithm for clustering the dataset and followed by k-anonymization. Decision stump classification is used to determine utility and privacy is determined by firing random queries on the anonymized dataset. The balancing point is where the utility and privacy curves intersect or they tend to converge. The balancing point will vary from dataset to dataset and the choice of Quasi-identifier and sensitive attribute. For our experiment the balancing point is found to be around 50-60 percent which is the intersecting point of privacy and utility curves.

CHAPTER 1

INTRODUCTION

1.1. Background

The amount of data that need to be processed to extract some useful information is increasing. Therefore different data mining methods are adopted to get optimum result with respect to time and utility of data. The amount of personal data that can be collected and analyzed has also increased. Data mining tools are increasingly being used to infer trends and patterns. In many scenarios, access to large amounts of personal data is essential in order for accurate inferences to be drawn. However, publishing of data containing personal information has to be restricted so that individual privacy is not hampered. One possible solution is that instead of releasing the entire database, only a part of it is released which can answer the adequate queries and do not reveal sensitive information. Only those queries are answered which do not reveal sensitive information. Sometimes original data is perturbed and the database owner provides a perturbed answer to each query. These methods require the researchers to formulate their queries without access to any data. Sanitization approach can be used to anonymize the data in order to hide the exact values of the data. But conclusion can't be drawn with surety. Another approach is to suppress some of the data values, while releasing the remaining data values exactly. But suppressing the data may hamper the utility. A lot of research work has been done to protect privacy and many models have been proposed to protect databases. Out of them, k-anonymity has received considerable attention from computer scientist. Under k-anonymity, each piece of disclosed data is equivalent to at least k-1 other pieces of disclosed data over a set of attributes that are deemed to be privacy sensitive.

1.2. Layout of this Thesis

The thesis has been divided into three chapters. The first chapter "Important Concepts" consists of those concepts which have been used for implementation and experiments. The second chapter "Algorithms" explains the algorithms that we have studied and implemented to get the results. This is followed by the Chapter "Implementation and Results" wherein we show and explain the results obtained by implementing our algorithms. The last chapter is "Conclusion and Future Work".

CHAPTER 2

IMPORTANT CONCEPTS

2.1. What is Data Mining?

Data mining is a technique that helps to extract useful information from a large database. It is the process of extracting relevant information from large databases through the use of certain data mining algorithms. As the amount of data doubles every three years, data mining is becoming an increasingly important tool to transform this data into information. Data mining techniques takes a long time which requires long process of research and product development. This evolution started with storing of business data on computers, continued with improvements in data access, and more recently, generated technologies that allow users to search their data in real time. Data mining is ready for application in the business community because it is supported by three technologies that are now sufficiently mature:

- Massive data collection
- Powerful multiprocessor computers
- Data mining algorithms

2.2. Methods of Data Mining

The Amount of data that need to be processed to extract some useful information is increasing. So the methods used for extracting information from huge amount of data must be optimum. As described in [1] the various data mining algorithms can be classified into two broad categories.

1. Heuristic-based approaches
    - additive noise
    - multiplicative noise
    - k-anonymization
    - statistical disclosure control based approaches
2. Cryptography -based approaches

2.2.1. Additive-Noise-based Perturbation Techniques

Random noise is added to the actual data in additive-noise-based perturbation technique. The privacy is measured by evaluating how closely the original values of a modified attribute can be determined. In particular, if the perturbed value of an attribute can be estimated, with a confidence c, to belong to an interval [a, b], then the privacy is estimated by (b−a) with confidence c. However, this metric does not work well because it does not take into account the distribution of the original data along with the perturbed data.

2.2.2. Multiplicative-Noise-based Perturbation Techniques

As shown in [2] Additive random noise can be filtered out using certain signal processing techniques with very high accuracy. This problem can be avoided by using random projection-based multiplicative perturbation techniques as proposed in [3]. Instead of adding some random values to the actual data, random matrices are used to project the set of original data points to a randomly chosen lower-dimensional space. However, the transformed data still preserves much statistical aggregate regarding the original dataset so that certain data mining tasks can be performed on the transformed data in a distributed environment (data are either vertically partitioned or horizontally partitioned) with small errors. High degree of privacy of original data is ensured in this approach. Even if the random matrix is disclosed, it only approximate value of original data can be estimated. It is impossible to get back the original data. The variance of the approximated data is used as privacy measure.

2.2.3. k- Anonymization Techniques

K-anonymization technique for privacy preservation is introduced by Samarati and Sweeney [4, 5]. A database is k-anonymous with respect to quasi-identifier attributes (defined later in this thesis) if there exist at least k transactions in the database having the same values according to the quasi-identifier attributes. In practice, in order to protect sensitive dataset T, before releasing T to the public, T is converted into a new dataset T* that guarantees the k-anonymity property for a sensible attribute. This is done by generalizations and suppression on quasi-identifier attributes. Therefore, the degree of uncertainty of the sensitive attribute is at least 1/k.

2.2.4. Statistical-Disclosure-Control-based Techniques

To anonymize the data to be released (such as person, household and business) which can be used to identify an individual, additional information publicly available need to be considered as described in [6]. Among these methods specifically designed for continuous data, the following masking techniques are described: additive noise, data distortion by probability distribution, resampling, rank swapping, etc. The privacy level of such method is assessed by using the disclosure risk, that is, the risk that a piece of information be linked to a specific individual.

2.2.5. Cryptography-based Techniques

The cryptography-based technique usually guarantees very high level of data privacy. Generally solution is based on the assumption that each party first encrypts its own item sets using commutative encryption, then the already encrypted item sets of every other party. The two communicating party must share a common key which is used for encryption and decryption. Sometimes two key is used known as public key and private key. Public key is known to everybody that wants to communicate with you and private key is used for decryption in a secure communication. Though cryptography-based techniques can well protect data privacy, they may not be considered good with respect to other metrics like efficiency.

2.3. Privacy

Privacy means how an individual controls who has access to his personal information. From another point of view, Privacy may be how the data is collected, shared and used by the customers. So definition of privacy varies from one environment to the other. So the definition of privacy as described in [1] is as follows:

- Privacy as the right of a person to determine which personal information about himself/ herself may be communicated to others.
- Privacy as the control over access to information about oneself.
- Privacy as limited access to a person and to all the features related to the person.

From our experiment point of view privacy is defined in [1] as “The right of an entity to be secure from unauthorized disclosure of sensible information that are contained in an electronic repository or that can be derived as aggregate and complex information from data stored in an electronic repository”.

2.4. Data Utility

The utility of the data must be preserved to certain extent at the end of the privacy preserving process, because in order for sensitive information to be hidden, the database is essentially modified through the changing of information (through generalization and suppression) or through the blocking of data values. Sampling is a privacy preserving technique which does not modify the information stored in the database, but still, the utility of the data falls, since the information is not complete in this case. As we go on changing the data for preserving privacy, the less the database reflects the domain of interest. So, one of the evaluation parameter for the measuring data utility should be the amount of information that is lost after the application of privacy preserving process. Of course, the measure used to evaluate the information loss depends on the specific data mining technique with respect to which a privacy algorithm is performed. As defined in [7] information loss in the context of association rule mining will be measured either in terms of the number of rules that were both remaining and lost in the database after sanitization, or even in terms on the reduction/increase in the support and confidence of all the rules. For the case of classification, we can use metrics similar to those used for association rules. Finally, for clustering, the variance of the distances among the clustered items in the original database and the sanitized database can be the basis for evaluating information loss in this case.

2.5. Generalization and Suppression

Various method have been proposed for providing anonymity in the release of micro data, the k-anonymity proposal focuses on two techniques in particular: generalization and suppression, which, unlike other existing techniques, such as scrambling or swapping, preserve the truthfulness of the information. In the following paragraph we have described it in detail.

The mapping is stated by means of a generalization relationship $\leq_d$. Given two domains D_i and D_j Є Dom, $D_i \leq_d D_j$ states that values in domain D_j are generalizations of values in D_i. The generalization relationship $\leq_d$ defines a partial order on the set Dom of domains, and is required to satisfy the following conditions as stated in [4, 6]

C1: $\forall D_{i,} D_{j,} D_z \in$ Dom:

$(D_i \leq_D D_j)_, (D_i \leq_D D_Z) \Rightarrow (D_j \leq_D D_Z) \vee (D_Z \leq_D D_j)_{,,}$

C2: all maximal elements of Dom are singleton.

Condition C1 states that for each domain D_i, the set of domains generalization of D_i is totally ordered and, therefore, each D_i has at most one direct generalization domain D_j. It ensures determinism in the generalization process. Condition C2 ensures that all values in each domain can always be generalized to a single value. The definition of a generalization relationship implies the existence, for each domain D ∈ Dom, of a totally ordered hierarchy, called domain generalization hierarchy, denoted DGHD. A value generalization relationship is denoted as $\leq_V$ which associates with each value in domain D_i a unique value in domain D_j, direct generalization of D_i. The value generalization relationship implies the existence, for each domain D, of a value generalization hierarchy, denoted VGHD.

2.5.1. k-Minimal Generalization (with Suppression)

Definition 3 (Generalized table - with suppression). Let T_i and T_j be two tables defined on the same set of attributes. Table T_j is said to be a generalization (with tuple suppression) of table T_i, denoted $T_i \leq T_j$, if:

1. $| T_j| \leq | T_i|$

2. The domain dom(A, T_j) of each attribute A in T_i is equal to, or a generalization of, the domain dom(A, T_i) of attribute A in T_i

3. It is possible to define an injective function associating each tuple t_j in T_j with a tuple t_i in T_i, such that the value of each attribute in t_i is equal to, or a generalization of, the value of the corresponding attribute in t_i.

2.6. k-Anonymity and k-Anonymous Tables

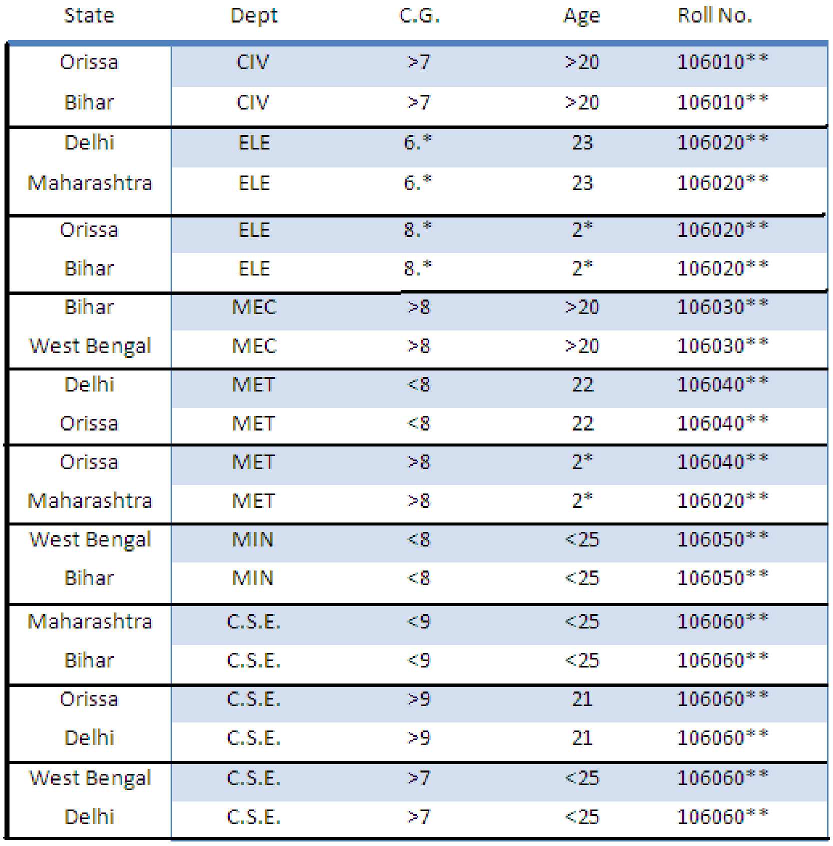
The concept of k-anonymity requires that the released private table (PT) should be indistinguishably related to no less than a certain number of respondents which is followed by all statistical community and by agencies. The set of attributes included in the private table, also externally available and therefore exploitable for linking, is called quasi-identifier. The k-anonymity requirement described in [6] states that every tuple released cannot be related to fewer than k respondents.

Definition 1 (k-anonymity requirement): Each release of data must be such that every combination of values of quasi-identifiers can be indistinctly matched to at least k respondents.
To guarantee the k-anonymity requirement, k-anonymity requires each quasi identifier value in the released table to have at least k occurrences, as stated in [6]

Definition 2 (k-anonymity): Let $T(A_1.........A_m)$ be a table, and QI be a quasi-identifier associated with it. T is said to satisfy k-anonymity with respect to QI if each sequence of values in T[QI] appears at least with k occurrences in T[QI].

This is a sufficient condition for k-anonymity requirement. If a set of attributes of external tables appears in the Quasi identifier associated with the private table PT, and the table satisfies Definition 2, the combination of the released data with the external data will never allow the recipient to associate each released tuple with less than k respondents. For example with respect to the student data table in Fig.1 and quasi identifier { Dept, C.G., Age, Roll NO} it easy to see that the table satisfies k-anonymity with k = 2 only, since there are single occurrences of values over the considered quasi-identifier (e.g., two occurrence of (" CIV, >7, >20, 106010**").

For k-anonymization we need to identify the quasi identifier from a set of attributes present in the original table. The quasi-identifier depends on the external information available to the recipient which determines the extent of linking (not all possible external tables are available to every possible data recipient). Therefore, although the identification of the correct quasi-identifier for a private table can be a difficult task, it is assumed that the quasi-identifier has been properly recognized and defined. For instance, in the student dataset of Fig.1 the quasi-identifiers are {Dept, C.G., Age, Roll NO}.



State	Dept	C.G.	Age	Roll No.
Orissa	CIV	>7	>20	106010**
Bihar	CIV	>7	>20	106010**
Delhi	ELE	6.*	23	106020**
Maharashtra	ELE	6.*	23	106020**
Orissa	ELE	8.*	2*	106020**
Bihar	ELE	8.*	2*	106020**
Bihar	MEC	>8	>20	106030**
West Bengal	MEC	>8	>20	106030**
Delhi	MET	<8	22	106040**
Orissa	MET	<8	22	106040**
Orissa	MET	>8	2*	106040**
Maharashtra	MET	>8	2*	106020**
West Bengal	MIN	<8	<25	106050**
Bihar	MIN	<8	<25	106050**
Maharashtra	C.S.E.	<9	<25	106060**
Bihar	C.S.E.	<9	<25	106060**
Orissa	C.S.E.	>9	21	106060**
Delhi	C.S.E.	>9	21	106060**
West Bengal	C.S.E.	>7	<25	106060**
Delhi	C.S.E.	>7	<25	106060**

Table 2.1: 2-anonymized table

2.7. Attacks on k-Anonymized Datasets

Sufficient care must be taken while selecting the quasi identifier because a solution that adheres to k-anonymity can still be vulnerable to attacks. Some possible attacks identified by Sweeney [8] are described below.

2.7.1. Unsorted matching attack against *k*-anonymity:

This attack is based on the order in which tuples appear in the released table. It can be corrected of course, by randomly sorting the tuples of the solution table. Otherwise, the release of a related table can leak sensitive information. For example a PT having two attributes is released twice. The quasi identifier is different in the two released table T1 and T2. If the orders of tuples are same in T1 and T2 then both tables can be linked to get back the original table.

2.7.2. Complementary release attack against k-anonymity:

It is more common that the attributes that constitute the quasi-identifier are themselves a subset of the attributes released. Therefore, subsequent releases of the same privately held information must consider all of the previously released attributes of T, so that it can prohibit linking on T.

2.7.3. Temporal attack against k-anonymity:

Data collections are dynamic. Tuples are added, changed, and removed constantly. As a result, releases of generalized data over time can be subject to a temporal inference attack. It is described in [8] by Sweeney. Let table T_0 be the original privately held table at time *t*=0. Assume a *k*-anonymity solution based on T_0, which is called table RT_0, is released. At time *t*, assume additional tuples were added to the privately held table T_0, so it becomes R_t. Let RT_t be a *k*-anonymity solution based on T_t that is released at time *t*. Because there is no requirement that RT_t respect RT_0, linking the tables RT_0 and RT_t may reveal sensitive information and thereby compromise *k*-anonymity protection. To combat this problem, RT_0 should be considered as joining other external information. Therefore, either all of the attributes of RT_0 would be considered a quasi identifier for subsequent releases, or subsequent releases themselves would be based on RT_0.

2.7.4. Homogeneity Attack:

When the non sensitive information of an individual is known to the attacker then sensitive information may be revealed based on the known information. It occurs if there is no diversity in the sensitive attributes for a particular block. This method of getting sensitive information is also known as positive disclosure. This suggests that in addition to k-anonymity, the sanitized table should also ensure "diversity" – all tuples that share the same values of their quasi-identifiers should have diverse values for their sensitive attributes.

2.7.5. Background Knowledge Attack:

If the user has some extra demographic information which can be linked to the released data which helps in neglecting some of the sensitive attributes, then some sensitive information about an individual might be revealed. This method of revealing information is also known as negative disclosure.

- To eliminate the homogeneity and background knowledge attack diversity in the sensitive information is necessary. The method of diversifying the sensitive attributes in a block is called l-diversity.

2.8. Privacy Principles:

The information published in the anonymized table is prone to attack due to the background knowledge of the adversary as described in [9]. So the private information might be revealed in two ways: positive disclosure and Negative disclosure.

2.8.1. Positive disclosure:

The original table T published after anonymization as T* results in a positive disclosure if the adversary can correctly identify the value of a sensitive attribute with high probability; i.e., given a $\delta > 0$, there is a positive disclosure if β (q, s, T*) > $(1 - \delta)$ and there exists t Є T such that t[Q] = q and t[S] = s.

2.8.2. Negative disclosure:

The original table T after anonymization is published as T* results in a negative disclosure if the adversary can correctly eliminate some possible values of the sensitive attribute with high probability; i.e., given an Є > 0, there is a negative disclosure if β (q, s, T*) < Є and there exists a t Є T such that t[Q] = q but t[S] != s.

- As described by Machanavajjhala in [9] all positive disclosures are not disastrous neither all negative disclosure. If the prior belief was that α (q, s) > $1-\delta$, the adversary would not have learned anything new. Hence, the ideal definition of privacy can be based on the following principle:

2.8.3. Uninformative Principle:

The published table should provide the adversary with little additional information beyond the background knowledge. In other words, there should not be a large difference between the prior and posterior beliefs.

Suppose the published table T* has two constants ρ1 and ρ2, we say that a (ρ1, ρ2)-privacy breach has occurred when either α (q, s) < ρ1 ∧ β (q, s, T*) > ρ2 or when α (q, s) > 1 − ρ1 ∧ β (q, s, T*) < 1−ρ2. If a (ρ1, ρ2) privacy breach has not occurred, then table T* satisfies (ρ1, ρ2)-privacy.

2.9. l-Diversity:

The drawback of k-anonymization due to the background knowledge attack can be removed by diversifying the values of sensitive attribute within a block. The l-diversity model is a very useful model for preventing attribute disclosure and it has been introduced in [9].

- **l-Diversity Principle:** A q*-block is l-diverse if it contains at least l well-represented values for the sensitive attribute S. A table is l-diverse if every q*-block is l-diverse. The l-diversity principle advocates ensuring l well-represented values for the sensitive attribute in every q-block, but does not clearly state what well-represented means.

2.9.1. Properties:

- Knowledge of the full distribution of the sensitive and non-sensitive attributes is not required in l-diversity.
- l-diversity does not even require the data publisher to have as much information as the adversary. The larger the value of l, the more information is needed to rule out possible values of the sensitive attribute.
- Different adversaries can have different background knowledge leading to different inferences. It simultaneously protects against all of them without the need for checking which inferences can be made with which levels of background knowledge.

2.9.2. Distinct l-diversity:

The term "well represented" in the definition of l-diversity would be to ensure there are at least l distinct values for the sensitive attribute in each equivalence class. Distinct l-diversity does not prevent probabilistic inference attacks. It may happen that in an anonymized block one value appear much more frequently than other values, enabling an adversary to conclude that an entity in the equivalence class is very likely to have that value.

2.9.3. Entropy l-diversity:

The entropy of an equivalence class E is defined to be

$$Entropy(E) = -\sum_{s \epsilon S} p(E,s) \log p(E,s)$$

Where S is the domain of the sensitive attribute, and p(E, s) is the fraction of records in E that have sensitive value s. A table is said to have entropy l-diversity if for every equivalence class E, Entropy(E) ≥ log l. Entropy l- diversity is strong than distinct l-diversity. In order to have entropy l-diversity for each equivalence class, the entropy of the entire table must be at least log (l). Sometimes this may too restrictive, as the entropy of the entire table may be low if a few values are very common.

2.9.4. Recursive (c, l)-diversity:

Recursive (c, l)-diversity ensure that the most frequent value does not appear too frequently, and the less frequent values do not appear too rarely. Let m be the number of values in an equivalence class, and r_i, $1 \leq i \leq m$ be the number of times that the i^{th} most frequent sensitive value appears in an equivalence class E. Then E is said to have recursive (c, l)-diversity if $r_1 < c(r_l + r_{l+1} + ... + r_m)$. A table is said to have recursive (c, l)-diversity if all of its equivalence classes have recursive (c, l)-diversity.

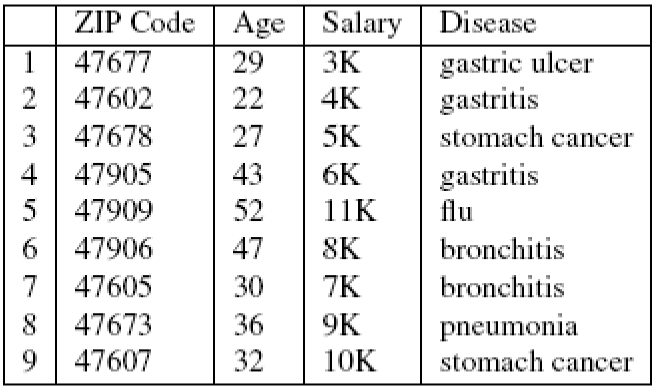
2.10. Attacks on l-diverse data:

2.10.1. Skewness Attack:

l-diversity does not prevent attribute disclosure if the overall distribution is skewed. Consider an equivalence class has an equal number of positive records and negative records. It satisfies distinct 2-diversity, entropy 2-diversity, and any recursive (c, 2)-diversity requirement that can be imposed. However, this presents a serious privacy risk, because anyone in the class would be considered to have 50% possibility of being positive, as compared with the 1% of the overall population. Now consider an equivalence class that has 49 positive records and only 1 negative record. It would be distinct 2-diverse and has higher entropy than the overall table (and thus satisfies any Entropy l-diversity that one can impose), even though anyone in the equivalence class would be considered 98% positive, rather than 1% percent. In fact, this equivalence class has exactly the same diversity as a class that has 1 positive and 49 negative record, even though the two classes present very different levels of privacy risks.

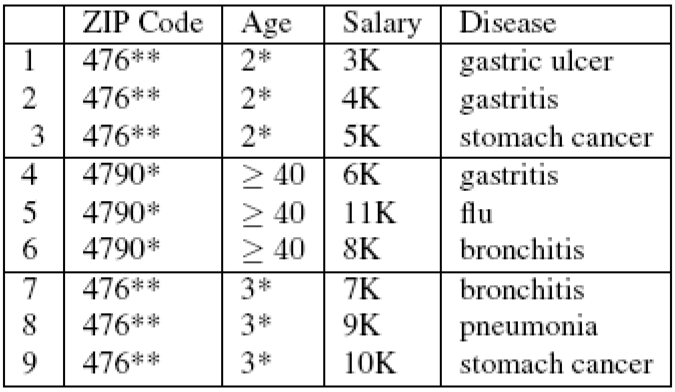
2.10.2. Similarity Attack:

When the sensitive attribute values in an equivalence class are distinct but semantically similar, an adversary can learn important information. Consider the following example. Table 2.2 is the original table, and Table 2.3 shows an anonymized version satisfying distinct and entropy 3-diversity. There are two sensitive attributes: Salary and Disease. Suppose one knows that Bob's record corresponds to one of the first three records, then one knows that Bob's salary is in the range [3K–5K] and can infer that Bob's salary is relatively low. This attack applies not only to numeric attributes like "Salary", but also to categorical attributes like "Disease". Knowing that Bob's record belongs to the first equivalence class enables one to conclude that Bob has some stomach-related problems, because all three diseases in the class are stomach-related.



	ZIP Code	Age	Salary	Disease
1	47677	29	3K	gastric ulcer
2	47602	22	4K	gastritis
3	47678	27	5K	stomach cancer
4	47905	43	6K	gastritis
5	47909	52	11K	flu
6	47906	47	8K	bronchitis
7	47605	30	7K	bronchitis
8	47673	36	9K	pneumonia
9	47607	32	10K	stomach cancer

Table 2.2: Original Salary/Disease table



	ZIP Code	Age	Salary	Disease
1	476**	2*	3K	gastric ulcer
2	476**	2*	4K	gastritis
3	476**	2*	5K	stomach cancer
4	4790*	≥ 40	6K	gastritis
5	4790*	≥ 40	11K	flu
6	4790*	≥ 40	8K	bronchitis
7	476**	3*	7K	bronchitis
8	476**	3*	9K	pneumonia
9	476**	3*	10K	stomach cancer

Table 2.3: A 3-diverse version of table 2.1

This leakage of sensitive information occurs because while l-diversity requirement ensures "diversity" of sensitive values in each group, it does not take into account the semantical closeness of these values.

CHAPTER 3

Algorithms

3.1. Samarati’s Algorithm for K-anonymization:

Samarati [4] proposed an algorithm for k-anonymization in 2001. This algorithm uses generalization and tuple suppression over quasi-identifiers to obtain a k-anonymized table with maximum suppression of MaxSup tuples. This algorithm uses binary search on the generalization hierarchy to save time. It assumes that a table PT with more than k attributes is present which is to be k-anonymized.

Given a table PT and a generalization hierarchy, different possible generalizations exist. Not all generalizations, however, can be considered equally satisfactory. For instance, the trivial generalization bringing each attribute to the highest possible level of generalization, thus collapsing all tuples in T to the same list of values, provides k-anonymity at the price of a strong generalization of the data. Such extreme generalization is not needed if a more specific table (i.e., containing more specific values) exists which satisfies k-anonymity. A naïve approach to compute a k-minimal generalization would then consist in following each generalization strategy (path) in the domain generalization hierarchy stopping the process at the first generalization that satisfies k-anonymity. However this approach becomes impractical when number of paths increase. A better approach to find k-minimal generalization is proposed in [4]. In this approach concept of distance vector is induced and exploited. Let PT be a table and x,y Є PT be two tuples such that x =(v_1.........v_n) and y=(v_1’……v_n’) where v_i and v_i’ are values in domain D_i The distance vector between x and y is the vector $V_{x,y}$ = [d_1……..d_n] where d_i is the (equal) length of the two paths from v_i and v_i’ to their closest common ancestor in the value generalization hierarchy $VGHD_i$ (or, in other words, the distance from the domain of v_i and v_i’ to the domain at which they generalize to the same value v_i). For example consider table PT illustrated in Table 1 and the generalization hierarchies for different attributes illustrated in Fig. 2. Assume Dept, C.G., Age and Roll No. to be a quasi-identifier. The distance vector between (CIV, 7.5, 20, 10601012) and (CIV, 8.6, 21, 10601026) is [0,1,1,1], at which they both generalize to (CIV,>7,>20,106010**).

State	Dept	C.G.	Age	Roll No.
Orissa	CIV	7.5	20	10601012
Bihar	CIV	8.6	21	10601026
Delhi	ELE	6.8	23	10602035
Maharashtra	ELE	6.4	23	10602039
Orissa	ELE	8.3	24	10602029
Bihar	ELE	8.4	25	10602025
Bihar	MEC	8.4	22	10603042
West Bengal	MEC	9.5	21	10603059
Delhi	MET	7.2	22	10604068
Orissa	MET	6.8	22	10604022
Orissa	MET	8.9	23	10604053

Table 3.1: Private Table (PT)

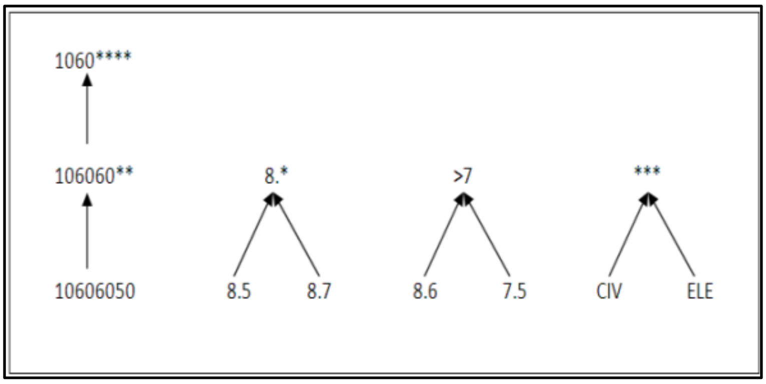


Figure 3.1: Generalization Hierarchy

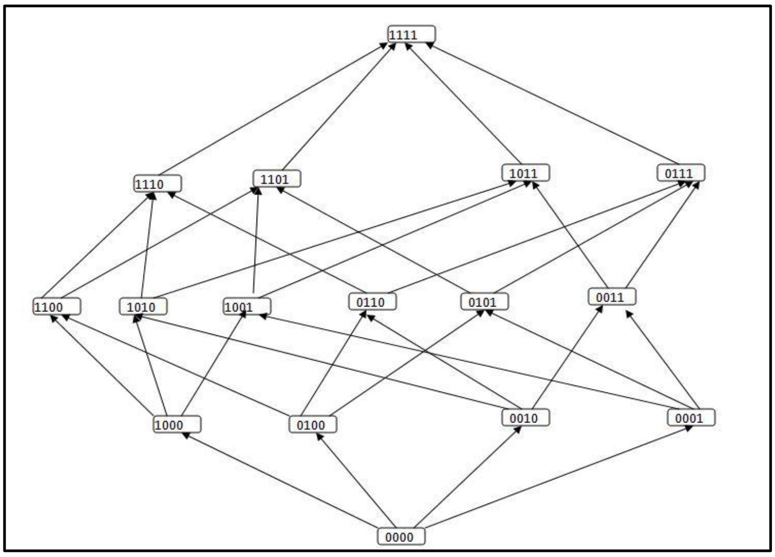
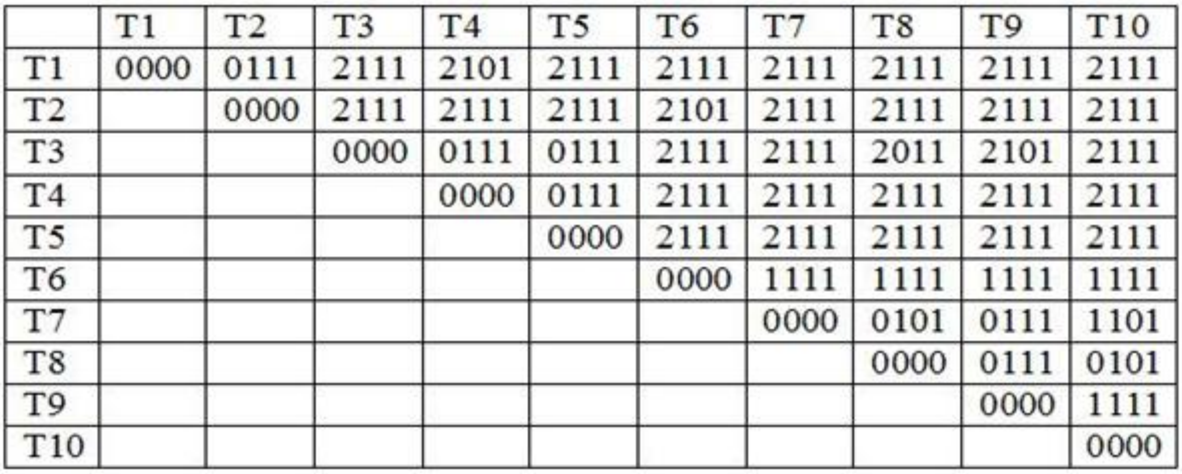


Figure 3.2: Distance Vector Lattice for PT



	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
T1	0000	0111	2111	2101	2111	2111	2111	2111	2111	2111
T2		0000	2111	2111	2111	2101	2111	2111	2111	2111
T3			0000	0111	0111	2111	2111	2011	2101	2111
T4				0000	0111	2111	2111	2111	2111	2111
T5					0000	2111	2111	2111	2111	2111
T6						0000	1111	1111	1111	1111
T7							0000	0101	0111	1101
T8								0000	0111	0101
T9									0000	1111
T10										0000

Figure 3.3: Distance Vector Matrix for PT

Algorithm:

Input: Table Ti =PT[QI] to be generalized, anonymity requirement k, suppression threshold MaxSup, lattice VLDT of distance vectors corresponding to generalization hierarchy DGHDT, where DT is the tuples of the domain of quasi-identifier attributes.

Output: The distance vector solution of generalized table GTsol, that is k-minimal generalization of PT[QI].

Method: Executes a binary search on VLDT based on height of vectors in lattice.

```
Low:=0; high=height(T, $VL_{DT)}$); sol:=T
While (low < high) do
    try:=$\left\lfloor\frac{(low+high)}{2}\right\rfloor$
    Vectors:={vec|height(vec, $VL_{DT}$)=try}
    reach_k:= false
    while vectors ≠ Φ ^ reach_k ≠ true do
        select and remove vec from vectors
        if satisfies (vec,k,$T_i$,MaxSup) then sol:=vec; reach_k:=true
        end If
    if reach_k = true then high:= try else low:=try+1
    end If
    12. End of while
    13. End of while
    14. Return sol
```

3.2. One-Pass K-Means Algorithm

This algorithm was proposed by Jun-Lin and Meng-Cheng in 2008 [12]. It is derived from the standard k-means algorithm but it runs for one iteration. This algorithm has two stages first is the clustering stage and second is the adjustment stage.

Clustering stage:

Let n be the total number of records present in the table T to be anonymized. Then N = $\left\lfloor\frac{\mathrm{n}}{\mathrm{k}}\right\rfloor$ where k is the value of k-anonymity. Clustering stage proceeds by sorting all the records and then randomly picking N records as seeds to build clusters. Then for each record r remaining in the dataset, algorithm checks to find the cluster o which this record is closest and assigns the record to the cluster and updates its centroid. The difference between the traditional k-means algorithm and OKA is that in OKA whenever a record is added to the cluster its centroid is updated thus improving the assignments in future and the centroid represents the real centre of the cluster. In OKA the records are first sorted according to the quasi-identifiers thus making sure that similar tuples are assigned to the same cluster. The algorithm has a complexity of $\mathrm{O}\left(\frac{\mathrm{n}^2}{\mathrm{k}}\right)$.

Algorithm: Clustering stage

Input: a set T of n records; the value k for k-anonymity
Output: a partitioning P = $\{P_1, . . . , P_K\}$ of T

```
1. Sort all records in dataset T by their quasi-identifiers;
2. Let N := $\left\lfloor\frac{n}{k}\right\rfloor$;
3. Randomly select N distinct records $r_1, . . . , r_N$ belongs to T ;
4. Let $P_i$ := $\{r_i\}$ for i = 1 to N;
5. Let T := T \ $\{r_1, . . . ,r_N\}$;
6. While (T != null ;) do
7. Let r be the first record in T ;
8. Calculate the distance between r to each $P_i$;
9. Add r to its closest $P_i$; update centroid of $P_i$;
```

```
    Let T := T \ {r};
End of While
```

Adjustment Stage:

In the clustering stage the clusters that are formed can contain more than k tuples and there can be some clusters containing less than k tuples, therefore when these clusters are anonymized will not satisfy condition for k-anonymity. These clusters need to be resized to contain at least k tuples. The goal of this adjustment stage is to make the clusters contain at least k records, while minimizing the information loss. This algorithm first removes the extra tuples from the clusters and then assigns those tuples to the clusters having less than k tuples. The removed tuples are farthest from the centroid of the cluster and while assigning the tuples to the clusters it checks the cluster which is closest to the tuple before assigning it, thus minimizing the information loss. If no cluster contains less than k tuples and some records are left they are assigned to this respective closest clusters. The time complexity of this algorithm is $\mathrm{O}\left(\frac{n^2}{k}\right)$

Algorithm: Adjustment Stage

Input: a partitioning $P = \{P_1, \ldots, P_K\}$ of T

Output: an adjusted partitioning $P = \{P_1, \ldots, P_K\}$ of T

```
Let R := null ;
For each cluster P belongs to p with |P| > k do
    Sort tuples in P by distance to centroid of P;
    While (|P| > k) do
        r belongs to P is the tuple farthest from centroid of P;
        Let P := P \ {r}; R := R [ {r};
    End of While
End of For
While (R != null) do
    Randomly select a record r from R;
    Let R := R \ {r};
    If P contains cluster Pi such that |Pi| < k then
        Add r to its closest cluster Pi satisfying |Pi| < k;
    Else
```

```
15. Add r to its closest cluster;
16. End If
17. End of While
```

3.3. K-Anonymization Algorithm Based on OKA

Once the table T is organized into clusters having at least K tuples, we can apply generalization hierarchy on the clusters to form a K-anonymized table. This algorithm uses the output of OKA and produces a K-anonymized table. The generalization hierarchy which is made should be complete which can map all possible values of the attribute to a single value. The time complexity of the algorithm is O(n).

Algorithm:

Input: an adjusted partitioning P = $\{P_1, \ldots, P_K\}$ of T and a generalization hierarchy for attributes

Output: A k-anonymized table T

```
1. For each Partition Pi of T do
2. For each quasi-identifier in Pi do
3. if attribute values for partition Pi are not same do
4. Use Generalization hierarchy to generalize
5. If attribute values for partition Pi are not same do
6. Go To 4
7. End If
8. End If
9. End of For
10. End of For
```

CHAPTER 4

Implementation and Results

4.1. Tools Used:

NetBeans: NetBeans is an integrated developing environment(IDE) written in the Java programming language, which can be used for developing with java, JavaScript, PHP, Python, Ruby, Groovy, C, C++ and much more. We have used NetBeans 6.0 to implement the algorithms as described in the previous chapter using java.

WEKA: Waikato Environment for Knowledge Analysis (WEKA) is a popular suite of machine learning software written in Java, developed at the University of Waikato. WEKA is free software available under the GNU General Public License. It contains a collection of machine learning algorithms for data mining tasks. The algorithms can either be applied directly to a dataset or called from your own Java code. It contains tools for data pre-processing, classification, regression, clustering, association rules, and visualization. We have used WEKA 3.6 for clustering and classification.

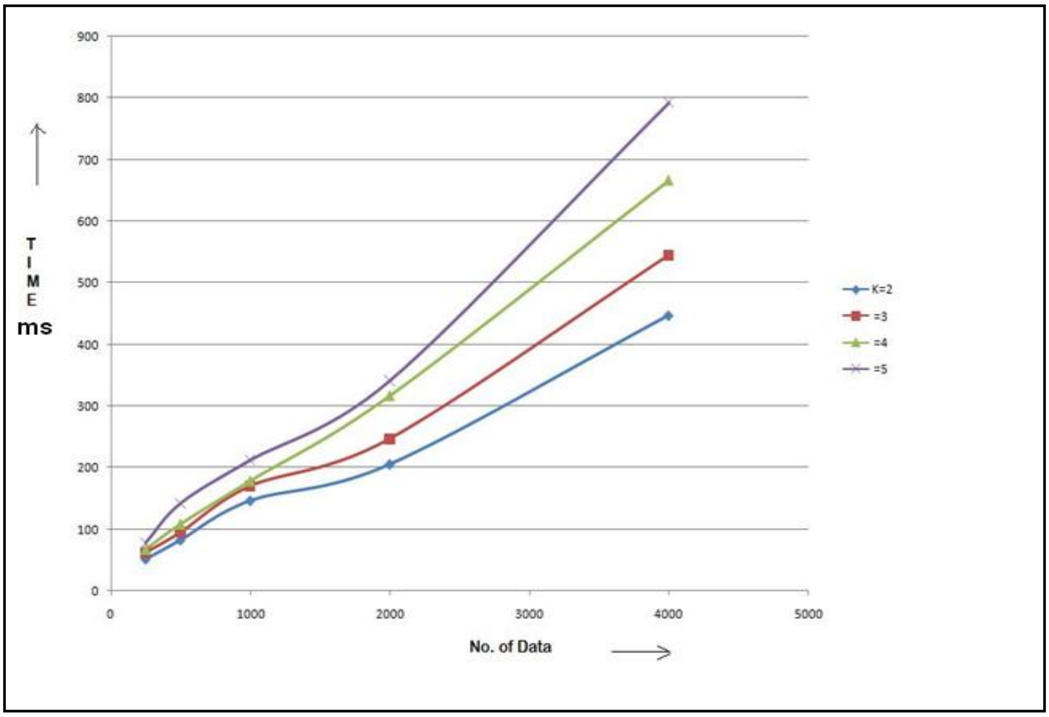
4.2. Implementation of OKA Algorithm

As described in the previous chapter OKA has two stages: Clustering Stage and Adjustment Stage. We have implemented the Clustering Stage using java and observed the time required to cluster with varying number of records and varying K-values. This algorithm was tested on a sample dataset shown in Figure 4.1. We implemented this algorithm for 3 attributes: Two of them were numerical attributes which is used for centroid calculation and other one is categorical attributes. The result is shown in figure 4.2.

Name	Roll No.	CGPA
Ankit	10405067	8.9
Sachin	10402061	8.5
Piyush	10406002	9.5
Rahul	10407008	9.1
Sunil	10406045	7.8
Manish	10402038	9.4
Sweta	1040506	7.2

Table 4.1: Sample Dataset

We found that as the value of k increases, the time required to cluster the data also increases. With same k value also with increase in no of tuples, time required to cluster increases.



.

Figure 4.1: Performance of OKA with Varying K

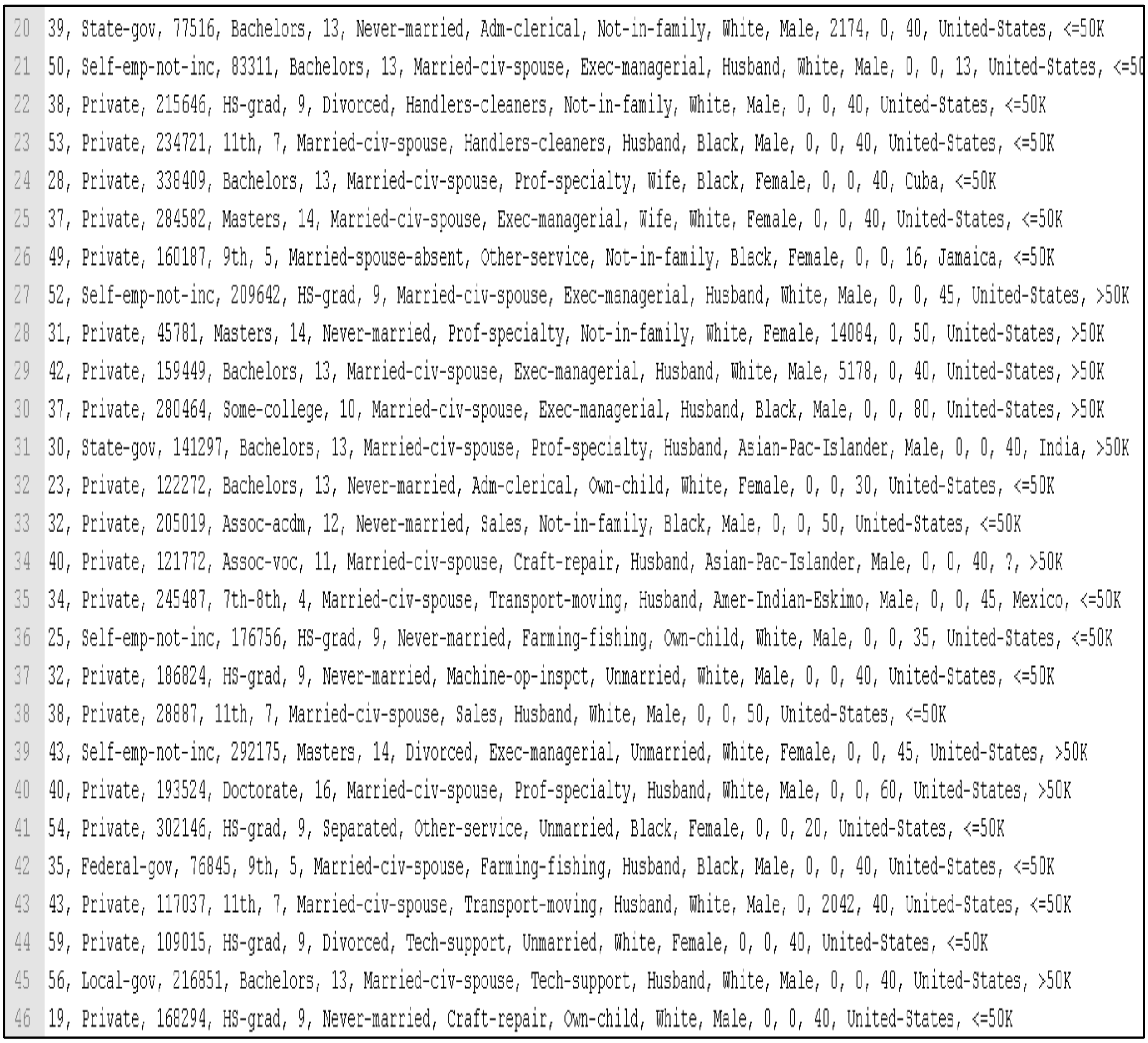
4.3. Experimental Set-up:

We carried out the experiments on the standard adult database from **UCI (University of California Irvine)** machine learning repository with 32,564 records. It contains numerical as well as categorical attributes which is suitable for generalization required in our experiment. It contains the following 15 attributes which may take values as described below:

Attribute	Attribute Values
age	continuous
work class	Private, Self-emp-not-inc, Self-emp-inc, Federal-gov, Local-gov, State-gov, Without-pay, Never-worked.
fnlwgt:	continuous.
education	Bachelor’s, Some-college, 11th, HS-grad, Prof-school, Assoc-acdm, Asso-voc, 9th, 7th-8th, 12th, Masters, 1st-4th, 10th, Doctorate, 5th-6th.
education-num	continuous
marital-status	Married-civ-spouse, Divorced, Never-married, Separated, Widowed, Married-spouse-absent, Married-AF-spouse.
occupation	Tech-support, Craft-repair, Other-service, Sales, Exec-managerial, Prof-specialty, Handlers-cleaners, Machine-op-inspct, Adm-clerical.
relationship	Wife, Own-child, Husband, Not-in-family, Other-relative, Unmarried.
race	White, Asian-Pac-Islander, Amer-Indian-Eskimo, Other, Black
sex	Female, Male.
capital-gain	continuous.
capital-loss	continuous.
hours-per-week	continuous.
native-country	United-States, Cambodia, England, Puerto-Rico, Canada, Germany, Outlying-US(Guam-USVI-etc), India, Japan, Greece, South, China.
Income	>50K, <=50K

Figure 4.2: UCI Dataset Attributes and Values

Figure 4.2 shows a sample of the adult database that we have used for conducting the experiments.



```
39, State-gov, 77516, Bachelors, 13, Never-married, Adm-clerical, Not-in-family, White, Male, 2174, 0, 40, United-States, <=50K
50, Self-emp-not-inc, 83311, Bachelors, 13, Married-civ-spouse, Exec-managerial, Husband, White, Male, 0, 0, 13, United-States, <=5
38, Private, 215646, HS-grad, 9, Divorced, Handlers-cleaners, Not-in-family, White, Male, 0, 0, 40, United-States, <=50K
53, Private, 234721, 11th, 7, Married-civ-spouse, Handlers-cleaners, Husband, Black, Male, 0, 0, 40, United-States, <=50K
28, Private, 338409, Bachelors, 13, Married-civ-spouse, Prof-specialty, Wife, Black, Female, 0, 0, 40, Cuba, <=50K
37, Private, 284582, Masters, 14, Married-civ-spouse, Exec-managerial, Wife, White, Female, 0, 0, 40, United-States, <=50K
49, Private, 160187, 9th, 5, Married-spouse-absent, Other-service, Not-in-family, Black, Female, 0, 0, 16, Jamaica, <=50K
52, Self-emp-not-inc, 209642, HS-grad, 9, Married-civ-spouse, Exec-managerial, Husband, White, Male, 0, 0, 45, United-States, >50K
31, Private, 45781, Masters, 14, Never-married, Prof-specialty, Not-in-family, White, Female, 14084, 0, 50, United-States, >50K
42, Private, 159449, Bachelors, 13, Married-civ-spouse, Exec-managerial, Husband, White, Male, 5178, 0, 40, United-States, >50K
37, Private, 280464, Some-college, 10, Married-civ-spouse, Exec-managerial, Husband, Black, Male, 0, 0, 80, United-States, >50K
30, State-gov, 141297, Bachelors, 13, Married-civ-spouse, Prof-specialty, Husband, Asian-Pac-Islander, Male, 0, 0, 40, India, >50K
23, Private, 122272, Bachelors, 13, Never-married, Adm-clerical, Own-child, White, Female, 0, 0, 30, United-States, <=50K
32, Private, 205019, Assoc-acdm, 12, Never-married, Sales, Not-in-family, Black, Male, 0, 0, 50, United-States, <=50K
40, Private, 121772, Assoc-voc, 11, Married-civ-spouse, Craft-repair, Husband, Asian-Pac-Islander, Male, 0, 0, 40, ?, >50K
34, Private, 245487, 7th-8th, 4, Married-civ-spouse, Transport-moving, Husband, Amer-Indian-Eskimo, Male, 0, 0, 45, Mexico, <=50K
25, Self-emp-not-inc, 176756, HS-grad, 9, Never-married, Farming-fishing, Own-child, White, Male, 0, 0, 35, United-States, <=50K
32, Private, 186824, HS-grad, 9, Never-married, Machine-op-inspct, Unmarried, White, Male, 0, 0, 40, United-States, <=50K
38, Private, 28887, 11th, 7, Married-civ-spouse, Sales, Husband, White, Male, 0, 0, 50, United-States, <=50K
43, Self-emp-not-inc, 292175, Masters, 14, Divorced, Exec-managerial, Unmarried, White, Female, 0, 0, 45, United-States, >50K
40, Private, 193524, Doctorate, 16, Married-civ-spouse, Prof-specialty, Husband, White, Male, 0, 0, 60, United-States, >50K
54, Private, 302146, HS-grad, 9, Separated, Other-service, Unmarried, Black, Female, 0, 0, 20, United-States, <=50K
35, Federal-gov, 76845, 9th, 5, Married-civ-spouse, Farming-fishing, Husband, Black, Male, 0, 0, 40, United-States, <=50K
43, Private, 117037, 11th, 7, Married-civ-spouse, Transport-moving, Husband, White, Male, 0, 2042, 40, United-States, <=50K
59, Private, 109015, HS-grad, 9, Divorced, Tech-support, Unmarried, White, Female, 0, 0, 40, United-States, <=50K
56, Local-gov, 216851, Bachelors, 13, Married-civ-spouse, Tech-support, Husband, White, Male, 0, 0, 40, United-States, >50K
19, Private, 168294, HS-grad, 9, Never-married, Craft-repair, Own-child, White, Male, 0, 0, 40, United-States, <=50K
```

Figure 4.2: Adult Dataset From UCI Reposiotry

The algorithms were implemented in java and executed on a workstation with Intel Dual Core Processor, 1.80 GHz and 1.00 GB of RAM on Window XP SP2 platform.

Clustering: Clustering of the database is done using WEKA. We have used K-means clustering for our experiment. The clustered results produced by WEKA are saved for further use in the experiment. Figure 4.3 shows the clustering results produced by WEKA.

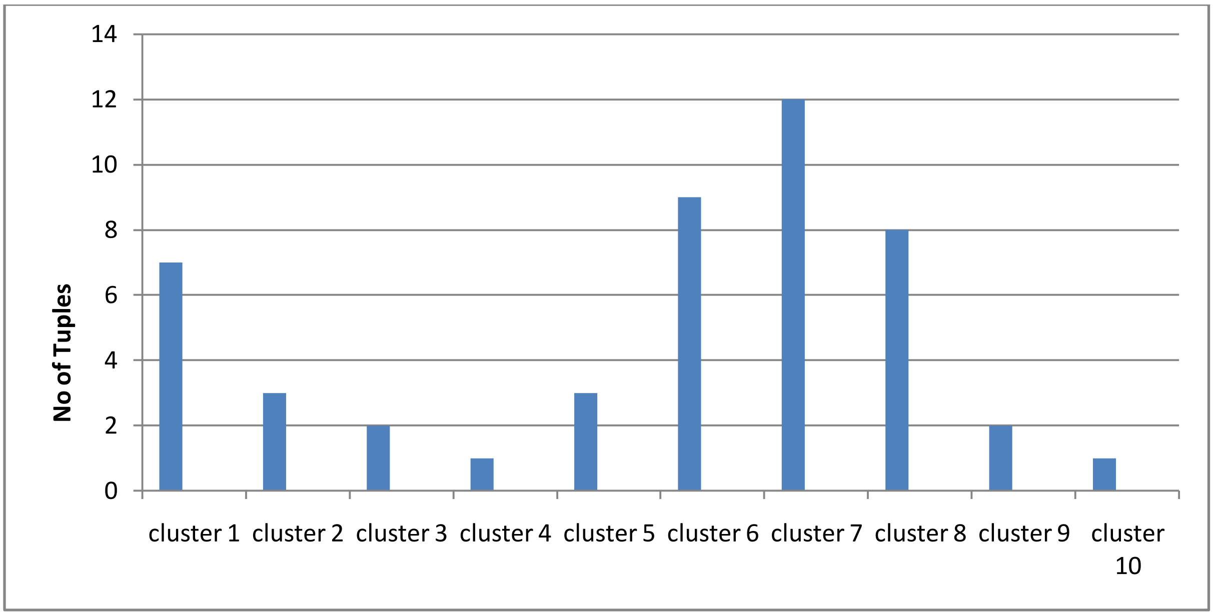


Figure 4.3: Clusters generated by WEKA

Figure 4.3 shows that the clusters are not uniform and cannot be used for k-anonymization. Thus we need to adjust the size of these clusters so that each cluster contains at least k tuples.

4.4. Generalization:

Generalization is done on the clustered dataset from the K-means algorithm. Details of the data and the generalization are shown below. Out of the total 15 attributes we considered 5 attributes as quasi-identifiers and rest as sensitive attributes.

Generalization rules: For age which is a numerical attribute mean of all the tuple values is taken.

$$\text{Mean age} = \frac{\sum_{i=1}^{k} t(i)}{k}$$

Figure 4.4, Figure 4.5, Figure 4.6, figure 4.7 shows generalization hierarchy for education, native-country, race and work class. These generalization hierarchies are used for k-anonymizing evenly clustered data.

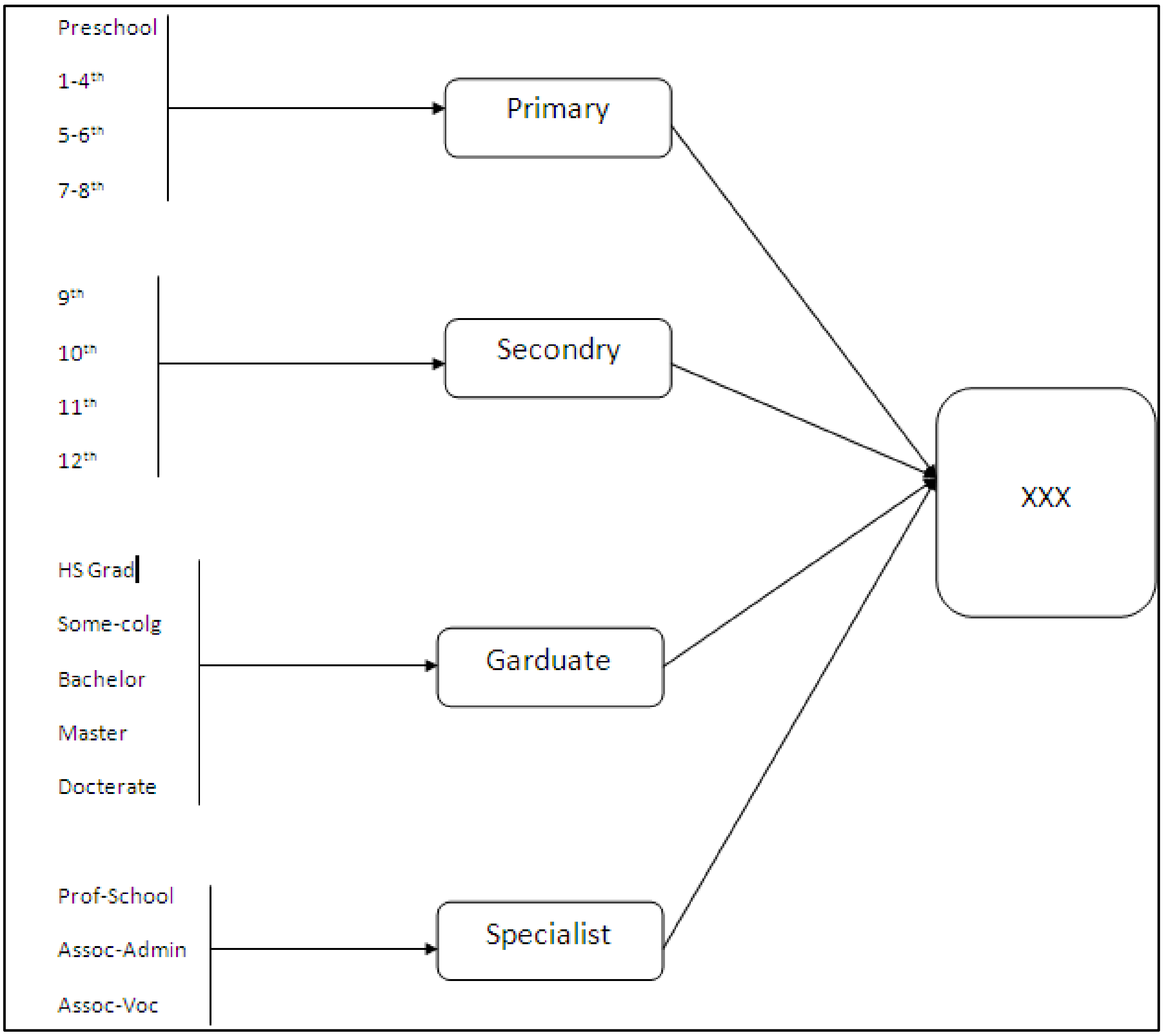


Figure 4.4: generalization heirarchy for education

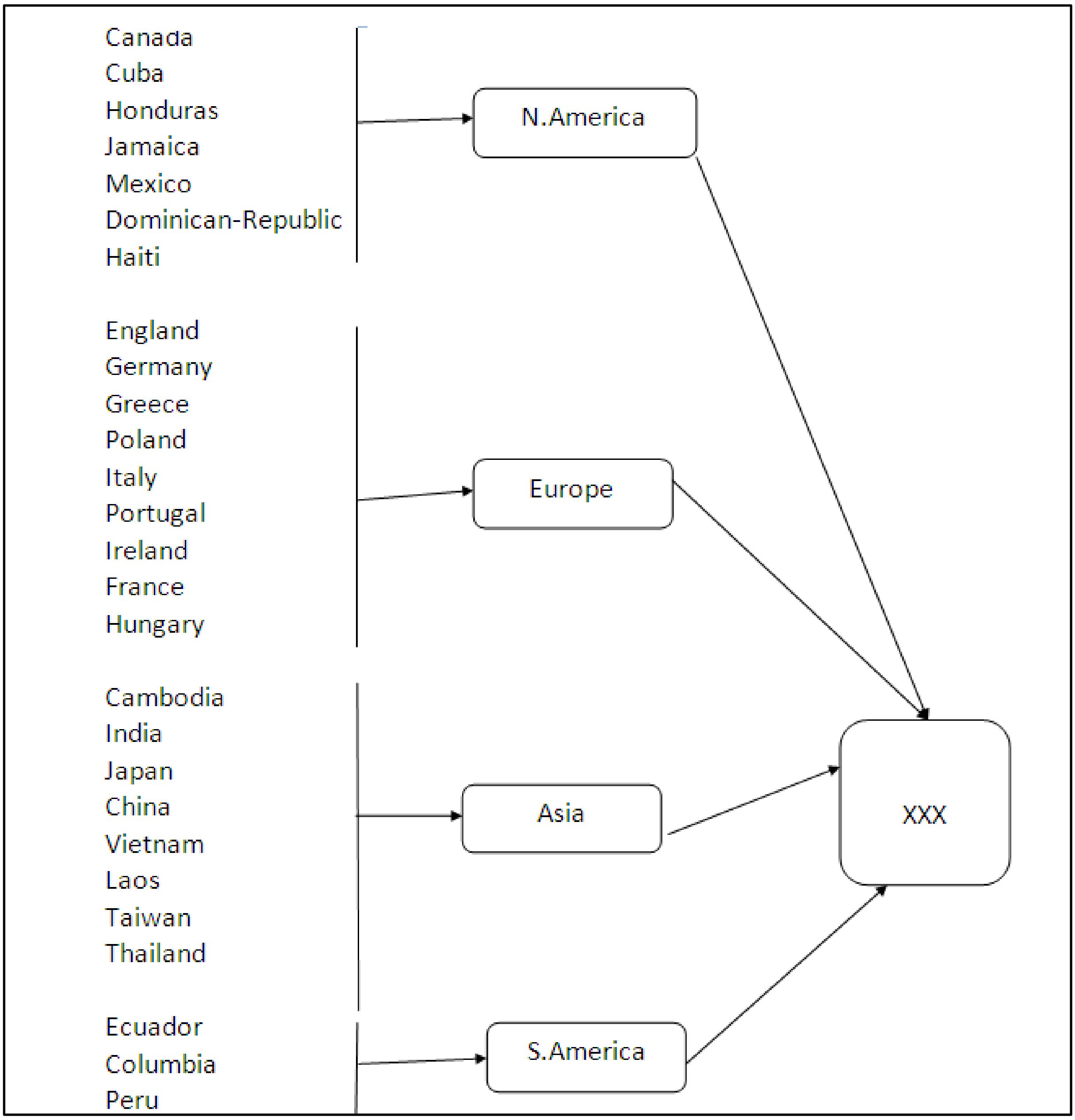


Figure 4.5: generalization Heirarchy for Native-country

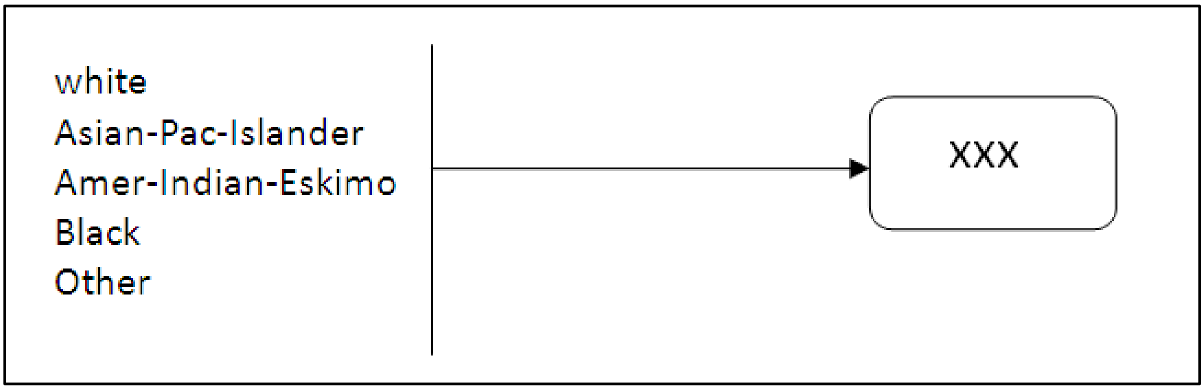


Figure 4.6: Generalization Hierarchy for Race

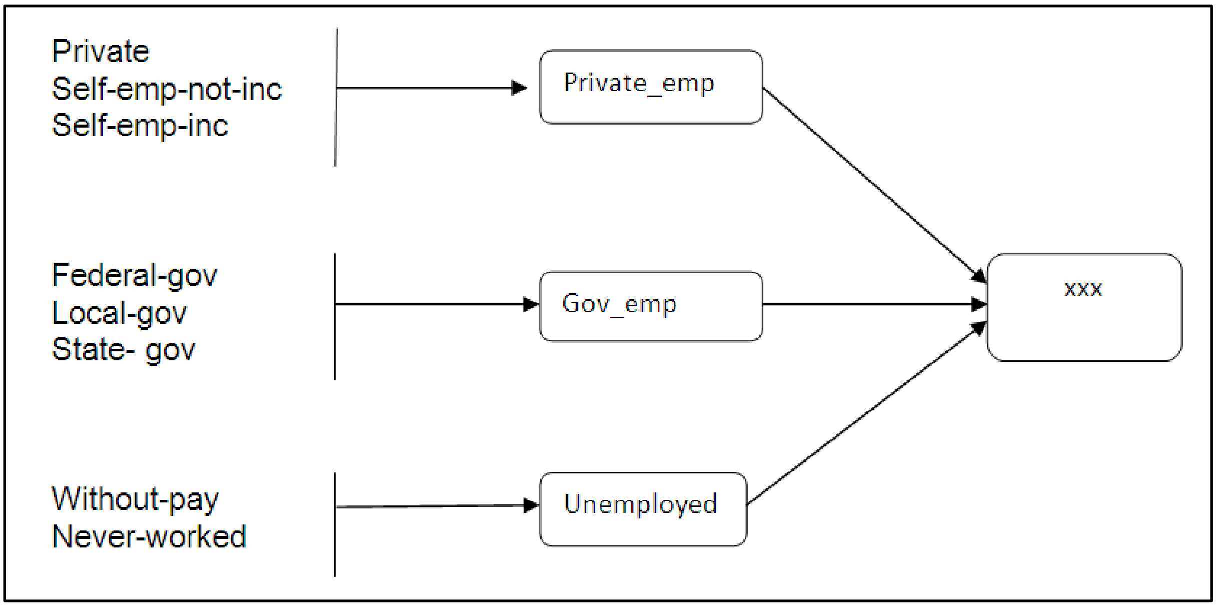


Figure 4.7: Generalization Hierarchy for Work class

4.5. Methodology Used for determining Utility and Privacy:

Utility: To determine the utility of the dataset we have used Decision stump algorithm for classification which is already implemented in WEKA. Decision stump is a machine learning model consisting of a single-level decision tree with a categorical or numeric class label. The results produced by WEKA clearly show percentage of tuples that can be correctly classified using the algorithm.

Privacy: To determine the extent of privacy preserved by the dataset we counted the number of attributes whose values are completely suppressed. For example the generalization hierarchy shown in Figure 4.7 suppresses the attribute value by generalizing the whole domain to 'xxx'. Queries used for calculating privacy for our generalization hierarchy are as follows:

- Select count (*) from dataset where education = 'xxx' ;
- Select count (*) from dataset where country = 'xxx' ;
- Select count (*) from dataset where race = 'xxx' ;
- Select count (*) from dataset where work class = 'xxx' ;

$$\text{Privacy \%} = \left(\frac{\text{Total number of suppressed values}}{\text{Total number of quasi} - \text{identifier values}}\right) * 100$$

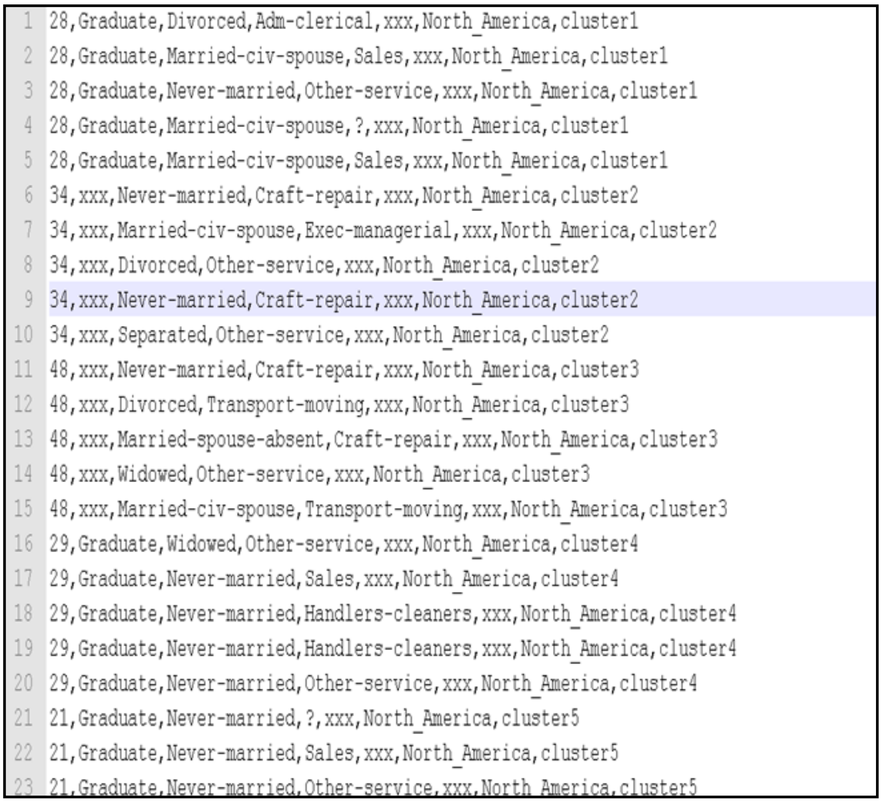
Percentage of privacy preserved in the anonymized dataset is given by the above formula.

Anonymizing sample dataset containing 1000 tuples:

Experiment 1:

In the first experiment we considered only six attributes, age, education, marital status, occupation, race and native-country for our analysis. We randomly selected 1000 tuples from the dataset for anonymization to determine how utility varies with privacy. Age, education, race and country are considered as quasi-identifiers and other two as sensitive attributes. First we used WEKA to arrange the data into clusters according to the value of k. As described in section 4.3 the clusters produced by WEKA may contain less than k tuples, thus an adjustment is required so that each cluster contains at least k tuples.

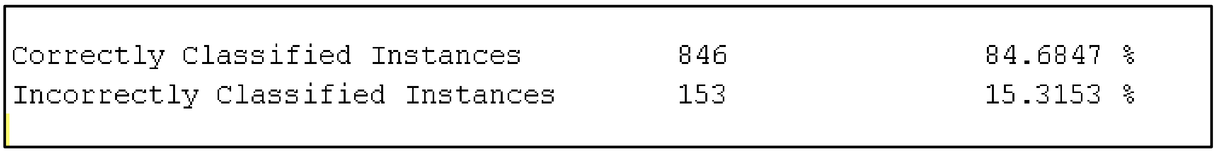
Before applying the generalization clusters are adjusted so that each cluster contains at least k tuples. After adjusting the clusters, k-anonymization is done based on the generalization hierarchy. We have implemented k anonymization algorithm based on OKA to generalize the adjusted clusters. Figure 4.8 shows a 5-k anonymized dataset obtained after anonymizing.



```
28,Graduate,Divorced,Adm-clerical,xxx,North_America,cluster1
28,Graduate,Married-civ-spouse,Sales,xxx,North_America,cluster1
28,Graduate,Never-married,Other-service,xxx,North_America,cluster1
28,Graduate,Married-civ-spouse,?,xxx,North_America,cluster1
28,Graduate,Married-civ-spouse,Sales,xxx,North_America,cluster1
34,xxx,Never-married,Craft-repair,xxx,North_America,cluster2
34,xxx,Married-civ-spouse,Exec-managerial,xxx,North_America,cluster2
34,xxx,Divorced,Other-service,xxx,North_America,cluster2
34,xxx,Never-married,Craft-repair,xxx,North_America,cluster2
34,xxx,Separated,Other-service,xxx,North_America,cluster2
48,xxx,Never-married,Craft-repair,xxx,North_America,cluster3
48,xxx,Divorced,Transport-moving,xxx,North_America,cluster3
48,xxx,Married-spouse-absent,Craft-repair,xxx,North_America,cluster3
48,xxx,Widowed,Other-service,xxx,North_America,cluster3
48,xxx,Married-civ-spouse,Transport-moving,xxx,North_America,cluster3
29,Graduate,Widowed,Other-service,xxx,North_America,cluster4
29,Graduate,Never-married,Sales,xxx,North_America,cluster4
29,Graduate,Never-married,Handlers-cleaners,xxx,North_America,cluster4
29,Graduate,Never-married,Handlers-cleaners,xxx,North_America,cluster4
29,Graduate,Never-married,Other-service,xxx,North_America,cluster4
21,Graduate,Never-married,?,xxx,North_America,cluster5
21,Graduate,Never-married,Sales,xxx,North_America,cluster5
21,Graduate,Never-married,Other-service,xxx,North_America,cluster5
```

Figure 4.8: 5-anonymized dataset

Results: For evaluating utility, we performed the classification mining on the k-anonymized dataset (DT). Classification was performed by using WEKA Data Mining Software considering native-country as classification variable. We considered the percentage of correctly classified tuples as the utility of the dataset. Figure 4.9 shows the results produced by the WEKA on using decision stump algorithm for a 3-anonymized dataset. Privacy was calculated by counting the number of tuples which are generalized to xxx. Privacy percentage is calculated as described in section 4.5. Privacy and utility was calculated by varying the value of k. The balancing point between utility and privacy is the point where privacy and utility curves intersect or tend to converge. Figure 4.10 shows the variation of utility and privacy with k. It clearly follows from the figure that on increasing the value of k privacy provided by the dataset increases but utility decreases. For this sample dataset the balancing point comes between k=8 and k=9, and utility of the dataset at balancing point is around 60%.



```
Correctly Classified Instances         846               84.6847 %
Incorrectly Classified Instances       153               15.3153 %
```

Figure 4.9: WEKA Classification Result for 3-Anonymized Dataset

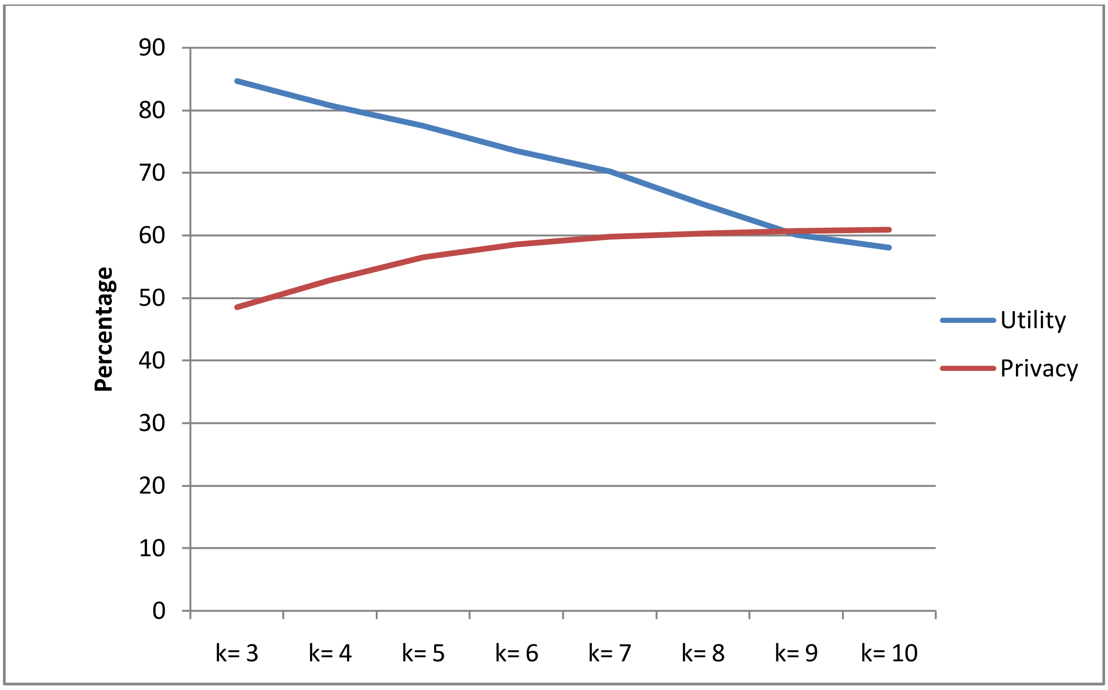
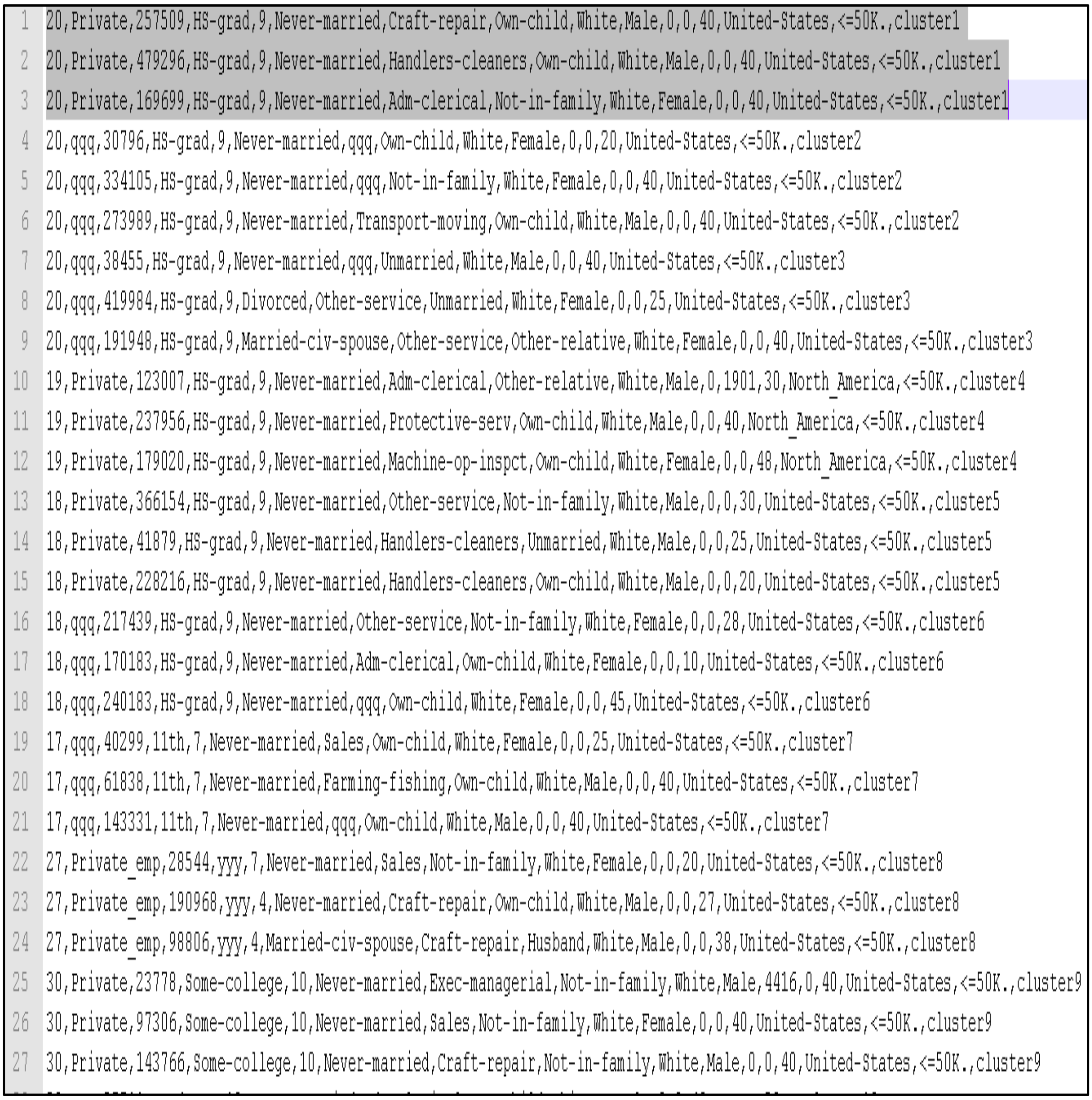


Figure 4.10: Variation of Utility And privacy with anonymization(1000 tuples)

Experiment 2:

In the second experiment we considered all the attributes for our analysis, to study the effect of more number of attributes on the privacy and the utility of the k-anonymized dataset. We randomly selected 1000 tuples from the dataset for anonymization to determine how utility varies with privacy. Age, work class, education, race and native-country are considered as quasi-identifiers and all other attributes as sensitive attributes. Similar steps were followed as in experiment 1 to study the variation of utility and privacy on varying k value. Figure 4.11 shows a 5-k anonymized dataset obtained after anonymizing.



```
20,Private,257509,HS-grad,9,Never-married,Craft-repair,Own-child,White,Male,0,0,40,United-States,<=50K.,cluster1
20,Private,479296,HS-grad,9,Never-married,Handlers-cleaners,Own-child,White,Male,0,0,40,United-States,<=50K.,cluster1
20,Private,169699,HS-grad,9,Never-married,Adm-clerical,Not-in-family,White,Female,0,0,40,United-States,<=50K.,cluster1
20,qqq,30796,HS-grad,9,Never-married,qqq,Own-child,White,Female,0,0,20,United-States,<=50K.,cluster2
20,qqq,334105,HS-grad,9,Never-married,qqq,Not-in-family,White,Female,0,0,40,United-States,<=50K.,cluster2
20,qqq,273989,HS-grad,9,Never-married,Transport-moving,Own-child,White,Male,0,0,40,United-States,<=50K.,cluster2
20,qqq,38455,HS-grad,9,Never-married,qqq,Unmarried,White,Male,0,0,40,United-States,<=50K.,cluster3
20,qqq,419984,HS-grad,9,Divorced,Other-service,Unmarried,White,Female,0,0,25,United-States,<=50K.,cluster3
20,qqq,191948,HS-grad,9,Married-civ-spouse,Other-service,Other-relative,White,Female,0,0,40,United-States,<=50K.,cluster3
19,Private,123007,HS-grad,9,Never-married,Adm-clerical,Other-relative,White,Male,0,1901,30,North_America,<=50K.,cluster4
19,Private,237956,HS-grad,9,Never-married,Protective-serv,Own-child,White,Male,0,0,40,North_America,<=50K.,cluster4
19,Private,179020,HS-grad,9,Never-married,Machine-op-inspct,Own-child,White,Female,0,0,48,North_America,<=50K.,cluster4
18,Private,366154,HS-grad,9,Never-married,Other-service,Not-in-family,White,Male,0,0,30,United-States,<=50K.,cluster5
18,Private,41879,HS-grad,9,Never-married,Handlers-cleaners,Unmarried,White,Male,0,0,25,United-States,<=50K.,cluster5
18,Private,228216,HS-grad,9,Never-married,Handlers-cleaners,Own-child,White,Male,0,0,20,United-States,<=50K.,cluster5
18,qqq,217439,HS-grad,9,Never-married,Other-service,Not-in-family,White,Female,0,0,28,United-States,<=50K.,cluster6
18,qqq,170183,HS-grad,9,Never-married,Adm-clerical,Own-child,White,Female,0,0,10,United-States,<=50K.,cluster6
18,qqq,240183,HS-grad,9,Never-married,qqq,Own-child,White,Female,0,0,45,United-States,<=50K.,cluster6
17,qqq,40299,11th,7,Never-married,Sales,Own-child,White,Female,0,0,25,United-States,<=50K.,cluster7
17,qqq,61838,11th,7,Never-married,Farming-fishing,Own-child,White,Male,0,0,40,United-States,<=50K.,cluster7
17,qqq,143331,11th,7,Never-married,qqq,Own-child,White,Male,0,0,40,United-States,<=50K.,cluster7
27,Private_emp,28544,yyy,7,Never-married,Sales,Not-in-family,White,Female,0,0,20,United-States,<=50K.,cluster8
27,Private_emp,190968,yyy,4,Never-married,Craft-repair,Own-child,White,Male,0,0,27,United-States,<=50K.,cluster8
27,Private_emp,98806,yyy,4,Married-civ-spouse,Craft-repair,Husband,White,Male,0,0,38,United-States,<=50K.,cluster8
30,Private,23778,Some-college,10,Never-married,Exec-managerial,Not-in-family,White,Male,4416,0,40,United-States,<=50K.,cluster9
30,Private,97306,Some-college,10,Never-married,Sales,Not-in-family,White,Female,0,0,40,United-States,<=50K.,cluster9
30,Private,143766,Some-college,10,Never-married,Craft-repair,Not-in-family,White,Male,0,0,40,United-States,<=50K.,cluster9
```

Figure 4.11: 3-anonymized Dataset

Results: As described in previous experiment privacy and utility were calculated by varying the value of k. The balancing point between utility and privacy is the point where privacy and utility curves intersect or tend to converge. Figure 4.12 shows the variation of utility and privacy with k. For this sample dataset the balancing point comes between k=11 and k=12, and utility of the dataset at balancing point is around 52%. Thus on increasing the number of quasi-identifiers considered for analysis the balancing point is shifts down and values of k at which balance is achieved increases.

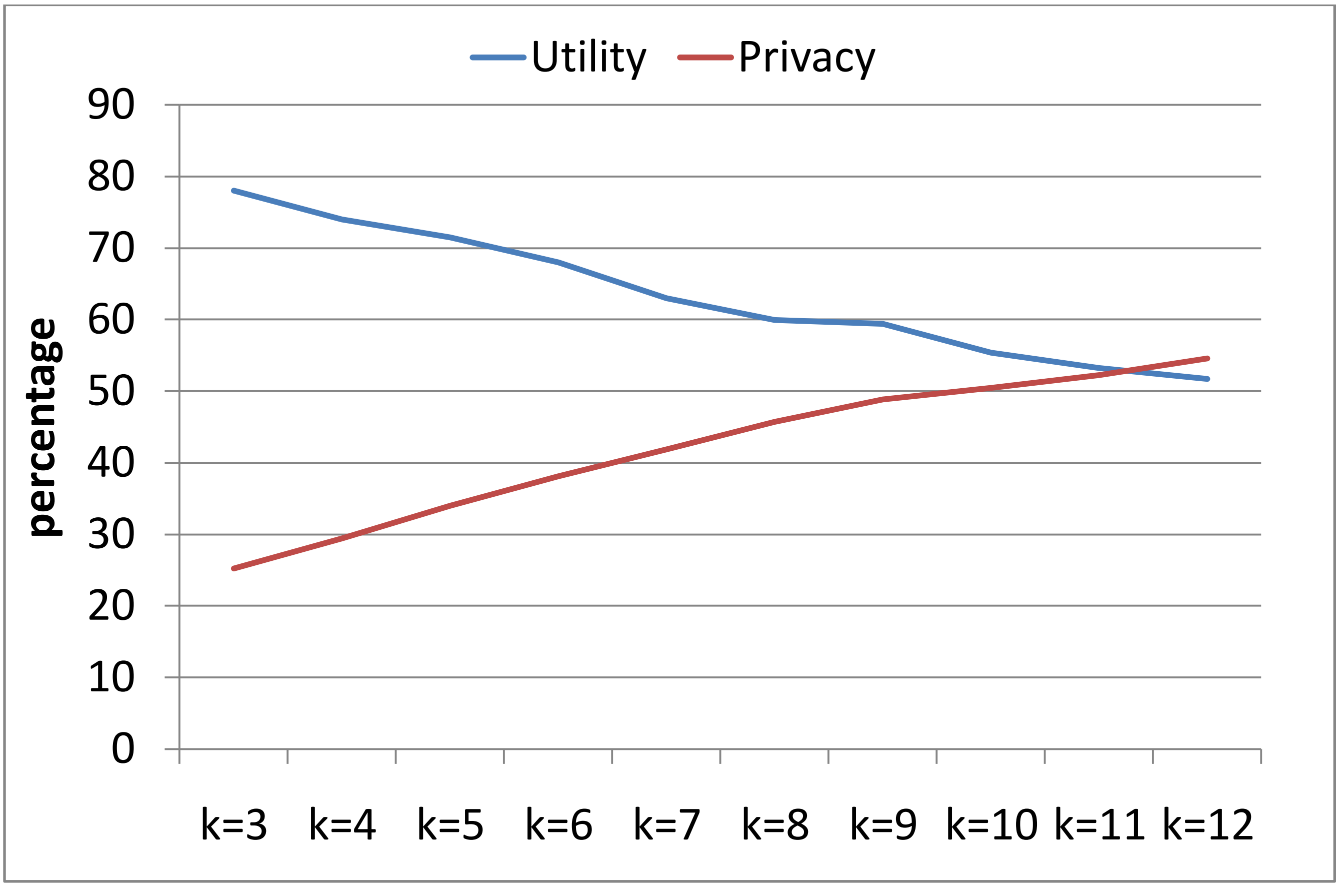


Figure 4.12: Variation of Utility And privacy with anonymization(1000 tuples)

Anonymizing sample dataset containing 3000 tuples:

In this experiment we took 3000 tuples from the adult dataset and carried out the same experiment. We considered all the attributes for our analysis, to study the effect of more number of tuples on the privacy and the utility of the k-anonymized dataset. Age, work class, education, race and native-country are considered as quasi-identifiers and all other attributes as sensitive attributes. Similar steps were followed as in experiment 1 to study the variation of utility and privacy on varying k value. Figure 4.13 shows variation of utility and privacy on varying value of k. For this sample dataset the balancing point comes between k=10 and k=11, and utility of the dataset at balancing point is around 50%.

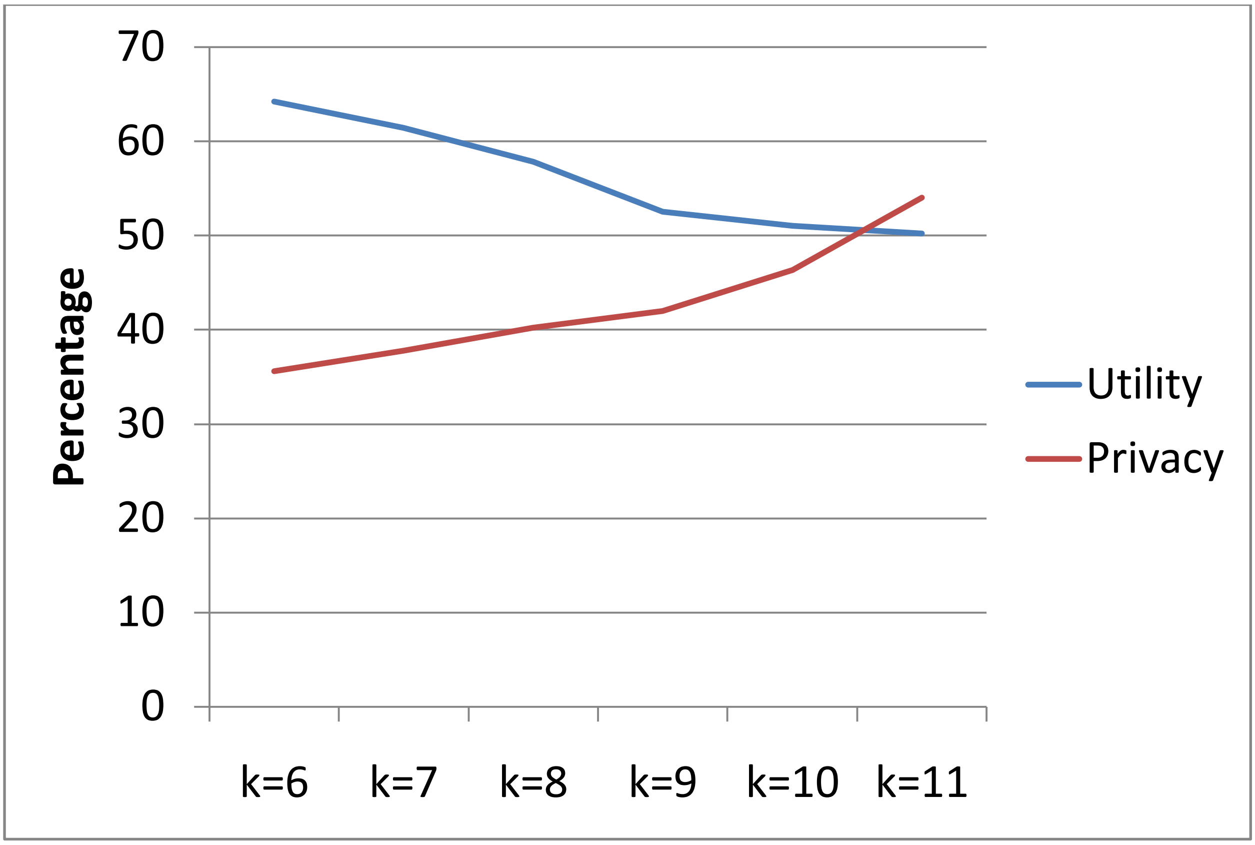


Figure 4.13: Variation of Utility and Privacy with anonymization (3000 tuples)

CHAPTER 5

Conclusion

And

Future Work

Conclusion: In order to improve the privacy offered by the dataset, utility of the data suffers. On conducting the experiments we found that the balancing point between utility and privacy depends on the dataset and value of k cannot be generalized for all datasets such that utility and privacy are balanced.

On varying the number of sensitive attributes in a dataset the balancing point varies. We found that if number of quasi-identifiers increases balancing point moves down and balance between utility and privacy occurs at a higher value of k. Thus if a dataset contains more number of quasi-identifiers then the utility as well as privacy attained at balancing point will be less than the dataset having fewer quasi-identifiers.

We also studied the affect of number of tuples in the data set on the balancing point and found that as the number of tuple increases there is slight shift in the balancing point and the value of k for which balancing occurs. Thus we can approximately predict the balancing point for a huge dataset by conducting experiment on a sample dataset.

Future Work: We tried to find balancing point between privacy and utility using k-anonymity, however there are some drawbacks of using k-anonymization for privacy preserving. Other privacy preserving algorithms can be used to find a balancing point between privacy and utility.

References:

1. E. Bertino, D. Lin, W. Jiang (2008). A Survey of Quantification of Privacy. In: Privacy-Preserving Data Mining. Springer US, Vol 34, pp. 183-205.

2. R. J. Bayardo, R. Agrawal (2005). Data privacy through optimal k-anonymization. In: Proc. of the 21st International Conference on Data Engineering, IEEE Computer Society, pp. 217-228.

3. K. Liu, H. Kargupta, J. Ryan (2006). Random projection-based multiplicative data perturbation for privacy preserving distributed data mining. IEEE Transactions on Knowledge and Data Engineering, Vol 18(1), pp. 92–106

4. P. Samarati (2001). Protecting respondents' identities in microdata release. IEEE Transactions on Knowledge and Data Engineering, VOl 13(6), pp. 1010–1027

5. L. Sweeney (2002). Achieving k-anonymity privacy protection using generalization and suppression. International Journal of Uncertainty, Fuzziness and Knowledge Based Systems, Vol 10(5), pp. 571–588.

6. V. Ciriani, S. De Capitani di Vimercati, S. Foresti, and P. Samarati (2007). k-Anonymity. In: Secure Data Management in Decentralized Systems. Springer US, Vol 33, pp. 323-353.

7. V. S. Verykios, E. Bertino, I. N. F. L. P. Provenza, Y. Saygin, and Y. Theodoridis (2004). State-of-the-art in Privacy Preserving Data Mining. ACM SIGMOD Record, Vol 33(1), pp. 50-57.

8. L. Sweeney (2002). k-anonymity: a model for protecting privacy. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, Vol 10 (5), pp. 557-570.

9. A. Machanavajjhala, J. Gehrke ,D. Kifer, M. Venkitasubramaniam (2007). ℓ-Diversity: Privacy Beyond k-Anonymity. ACM Transactions on Knowledge Discovery from Data, Vol 1(1), Article: 3.

10. R. Agrawal, R. Srikant (2000). Privacy preserving data mining. ACM SIGMOD Record, Vol 29(2), pp. 439–450.

11. M. R. Z. Mirakabad and A. Jantan (2008). Diversity versus Anonymity for Privacy Preservation. The 3rd International Symposium on Information Technology (ITSim2008), Vol 3, pp. 1-7.

12. J.Lin, and M. Wei (2008). An Efficient Clustering Method for k-Anonymization. In: Proceedings of the 2008 international workshop on Privacy and anonymity in information society, Vol. 331, pp. 46-50.

13. UCI Repository of machine learning databases, University of California, Irvine. http:/archive.ics.uci.edu/ml/.

14. M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten (2009). The WEKA Data Mining Software: An Update. SIGKDD Explorations, Vol 11(1).