International Journal of Advance Research in Science and Engineering
Volume No.07, Special Issue No.02, February 2018
www.ijarse.com
IJARSE
ISSN: 2319-8354

PRIVACY Vs UTILITY

Ch.LakshmiBala

Asst. professor Dept. of CSE, SVCET, Srikakulam (India)

ABSTRACT

There has been a lot of concern over privacy in the recent years due to sharing of data.It has also raised a potential threat of revealing sensitive data of an individual when the data is released publically. Though various methods have been proposed to tackle the privacy preservation problem (like anonymization and perturbation), 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

Keywords: Anonymization, Data Mining, k-means, Privacy, Utility.

I.INTRODUCTION

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. 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.

The paperis organized as follows:Section 2 details out the preliminaries. Section 3 describes the various algorithms used for this paper. Results are described in Section 4 followed by conclusions in Section 5. sults obtained by implementing our algorithms. The last chapter is "Conclusion and Future Work".

II.IMPORTANT CONCEPTS

There exist a number of data mining algorithms for information extraction. This section details about various preliminaries that are required for the rest of the section.

2.1. DATA MINING TECHNIQUES:

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.

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.

***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.

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, resembling, 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.

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.

Privacy:

Privacy means how an individual control 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".

2.2. 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 on 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.

2.3. GENERALIZATION AND SUPPRESSION:

Various method has 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 ≤d. Given two domains Di and Dj ∈ Dom, Di ≤d Dj states that values in domain Dj are generalizations of values in Di. The generalization relationship <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: ∀ Di,Dj, Dz ∈ Dom:

(Di ≤D Dj), (Di ≤D Dz) => (Dj ≤D Dz) V (Dz ≤D Dj),,

C2: all maximal elements of Dom are singleton.

Condition C1 states that for each domain Di, the set of domains generalization of Di is totally ordered and, therefore, each Di has at most one direct generalization domain Dj. 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 ≤v which associates with each value in domain Di a unique value in domain Dj, direct generalization of Di. The value generalization relationship implies the existence, for each domain D, of a value generalization hierarchy, denoted VGHD.

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_i in T_i 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.

2.4. 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

2.5. 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.

2.5.1. 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 $\beta(q, s, T^*) > (1 - \delta)$ and there exists $t \in T$ such that $t[Q] = q$ and $t[S] = s$.

2.5.2. 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 $\epsilon > 0$, there is a negative disclosure if $\beta(q, s, T^*) < \epsilon$ and there exists a $t \in T$ 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 $\alpha(q, s) > 1-\delta$, the adversary would not have learned anything new. Hence, the ideal definition of privacy can be based on the following principle:

2.5.3. 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 $\rho1$ and $\rho2$, we say that a ($\rho1$, $\rho2$)-privacy breach has occurred when either $\alpha(q, s) < \rho1 \wedge \beta(q, s, T^*) > \rho2$ or when $\alpha(q, s) > 1 - \rho1 \wedge \beta(q, s, T^*) < 1-\rho2$. If a ($\rho1$, $\rho2$) privacy breach has not occurred, then table T* satisfies ($\rho1$, $\rho2$)-privacy.

III.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 isproposed in [protecting respondent data]. 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 pathsfrom vi and vi' 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 vi).

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.

1. Low:=0; high=height(T, VL_{DT}); sol:=T
2. While (low < high) do
3. try:=$\left\lfloor \frac{(low + high)}{2} \right\rfloor$
4. Vectors:={vec|height(vec, VL_{DT})=try}
5. reach_k:= false
6. while vectors ≠ Φ ^ reach_k ≠ true do
7. select and remove vec from vectors
8. if satisfies (vec,k,T_i,MaxSup) then sol:=vec; reach_k:=true
9. end If
10. if reach_k = true then high:= try else low:=try+1
11. 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 = $\left\lfloor \frac{n}{k} \right\rfloor$ 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 center 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 $O\left(\frac{n^2}{k}\right)$.

Algorithm: Clustering stage:

Input: a set T of n records; the value k for k-anonymity

Output: a partitioning P = $\{P_1, \ldots, P_K\}$ 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 $r_1, \ldots, r_N$ belongs to T ;
4. Let $P_i := \{r_i\}$ for i = 1 to N;
5. Let T := T \ $\{r_1, \ldots, r_N\}$;
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;
10. Let T := T \ {r};
11. 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 $O\left(\frac{n^2}{k}\right)$.

Algorithm: Adjustment Stage:

Input: a partitioning P = {P_1, . . . , P_K } of T

Output: an adjusted partitioning P = {P_1, . . . , P_K } of T

1. Let R := null ;
2. For each cluster P belongs to p with |P| > k do
3. Sort tuples in P by distance to centroid of P;
4. While (|P| > k) do
5. r belongs to P is the tuple farthest from centroid of P;
6. Let P := P \ {r}; R := R [{r};
7. End of While
8. End of For
9. While (R != null) do
10. Randomly select a record r from R;
11. Let R := R \ {r};
12. If P contains cluster Pi such that |Pi| < k then
13. Add r to its closest cluster Pi satisfying |Pi| < k;
14. 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, . . . , 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

IV.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.

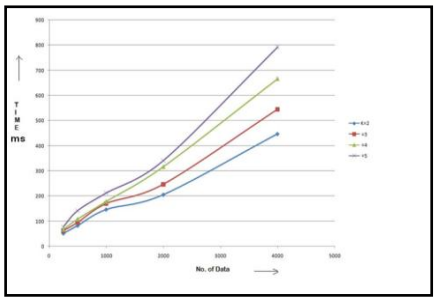

Figure 4.1: Performance of OKA with Varying K

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.

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.

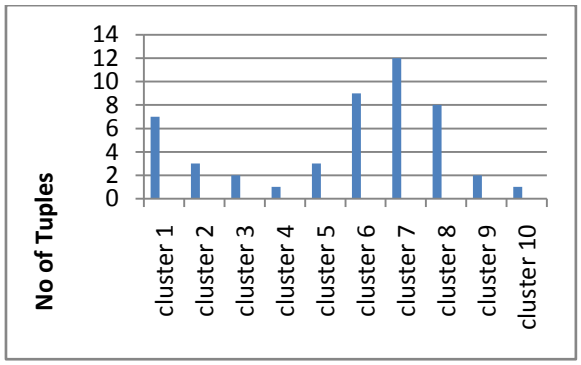

Figure 4.2: Clusters generated by WEKA

Figure 4.2 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.Mean age = $\frac{\sum_{i=1}^{k} t(i)}{k}$

Figure 4.3 shows generalization hierarchy for education. These generalization hierarchies are used for k-anonymizing evenly clustered data.

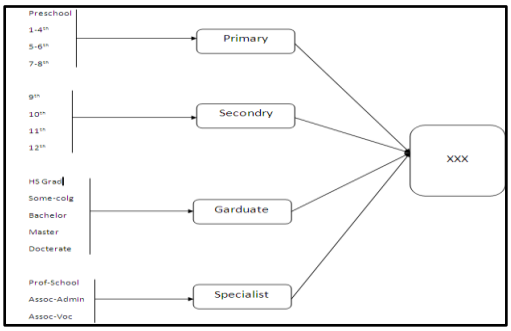

Figure 4.3: generalization heirarchy for education

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

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

Experiment1:

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.

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 4.4 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 4.5 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 4.4: WEKA Classification Result for 3-Anonymized Dataset

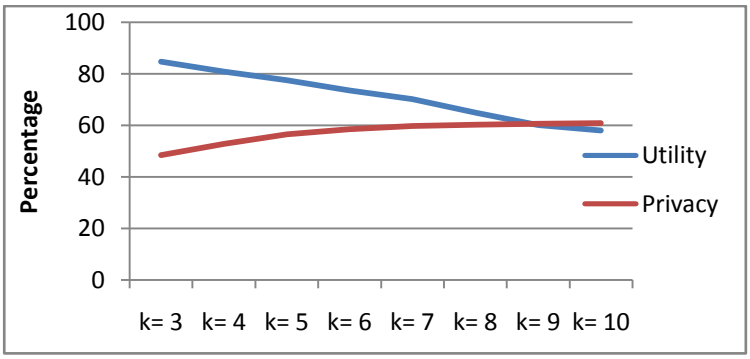

Figure 4.5: Variation of Utility and privacy with anonymization (1000 tuples)

Experiment2:

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.

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 4.6 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.

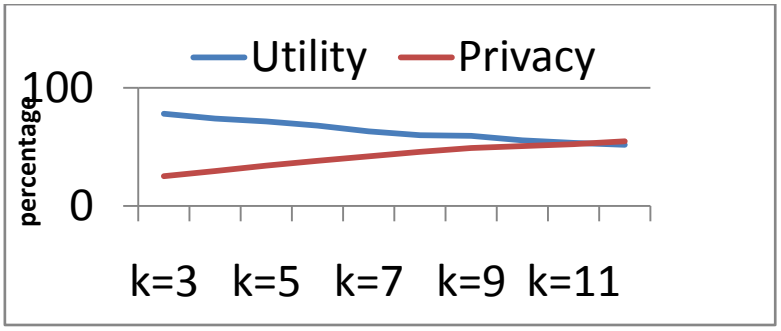

Figure 4.6: Variation of Utility And privacy with anonymization(1000 tuples)

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 4.7 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%.

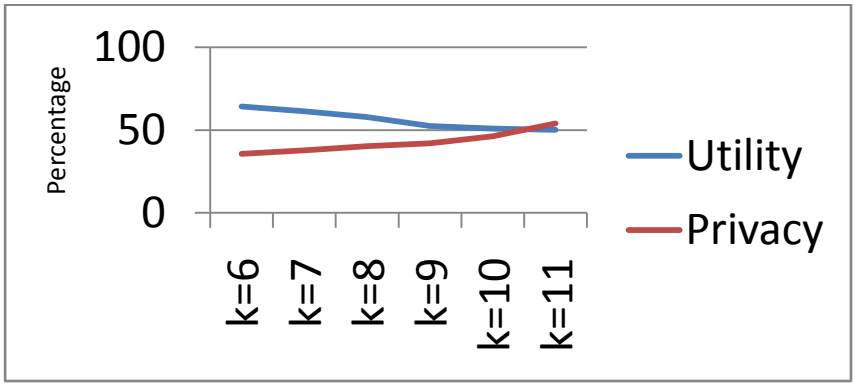

Figure 4.7: Variation of Utility and Privacy with anonymization(3000 tuples)

V.CONCLUSION

We also studied the effect 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.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.

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.] Vassilios S. Verykios, Elisa Bertino, Igor Nai Fovino Loredana Parasiliti Provenza, Yucel Saygin, and Yannis 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.] Ashwin Machanavajjhala, Johannes Gehrke ,Daniel Kifer, Muthuramakrishnan 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.] Mohammad Reza Zare Mirakabad and Aman Jantan (2008).Diversity versus Anonymity for Privacy Preservation. The 3rd International Symposium on Information Technology (ITSim2008), Vol 3, pp. 1-7.

[12.] Jun-Lin Lin and Meng-Cheng 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.

[14.] http:/archive.ics.uci.edu/ml/.

[15.] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, Ian H. Witten (2009); The WEKA Data Mining Software: An Update; SIGKDD Explorations, Vol 11(1).