Privacy-Preserving Data Visualization using Parallel Coordinates

Aritra Dasgupta and Robert Kosara

UNC Charlotte, USA

ABSTRACT

The proliferation of data in the past decade has created demand for innovative tools in different areas of exploratory data analysis, like data mining and information visualization. However, the problem with real-world datasets is that many of their attributes can identify individuals, or the data are proprietary and valuable. The field of data mining has developed a variety of ways for dealing with such data, and has established an entire subfield for *privacy-preserving data mining*. Visualization, on the other hand, has seen little, if any, work on handling sensitive data. With the growing applicability of data visualization in real-world scenarios, the handling of sensitive data has become a non-trivial issue we need to address in developing visualization tools.

With this goal in mind, in this paper, we analyze the issue of privacy from a visualization perspective and propose a privacy-preserving visualization technique based on clustering in parallel coordinates. We also outline the key differences in approach from the privacy-preserving data mining field and compare the advantages and drawbacks of our approach.

Keywords: privacy, k-anonymity, visualization, parallel coordinates

1. INTRODUCTION

Databases with sensitive information are ubiquitous today. Census records, medical databases, transaction databases are all examples of data sources which are of interest for analytical purposes. In many cases, the data contained in them is sensitive, either because it can identify an individual and link him or her to private information (e.g., medical data), or because it is highly valuable and would provide a significant advantage to a competitor if it became known. In addition to the value of the data, there are also regulations that make such disclosures punishable by law, like the *Health Insurance Portability and Accountability Act* (HIPAA) in the United States.

Many techniques for publication and analysis of de-identified sensitive data have been developed in the field of privacy-preserving data mining[1] (PPDM). The problem with de-identified data is that a lot of its fine details are lost. There is a trade-off between the amount of privacy gained and the amount of utility retained after de-identification.

In the field of information visualization, dealing with sensitive data is an equally important issue. With the growing number of applications of visualization techniques with real-world data, the handling of sensitive data is becoming inevitable. Visualization has been adopted widely to analyze and present abstract data, and due to the increase in size and complexity of data available, the need for more powerful and effective information visualization methods and techniques is growing. Handling of sensitive data will be a key challenge in this respect. The data hiding techniques proposed in privacy-preserving data mining would, in principle, apply in visualization. But there are some important distinctions. The ultimate goal of visualization being enabling the user to gather insight, the visualized data must maximize utility and therefore be at an appropriate level of granularity. Moreover, protecting privacy in a dynamic environment is a bigger challenge because user interaction may lead to potential privacy breach scenarios and have to be handled properly.

1.1 Privacy-Preserving Data Mining (PPDM)

PPDM addresses the scenario where data with quasi-identifiers is sanitized by the data owner and handed over to an untrusted third party for analysis purposes. Sanitization of data for mining purposes achieves *input privacy* (Figure 1a). This is an area where a lot of work has been done in the field of PPDM. The main techniques for sanitization are a) randomization, where random noise is added to the data for the purpose of perturbation, b) suppression, where some values are hidden and c) generalization, where data values are generalized to a higher level of granularity. The data is then either published or used for analysis purposes while minimizing disclosure of sensitive information.

The basic goal of PPDM is preventing identification of individual data through linking. This happens when a set of attributes co-occur in both private databases containing *sensitive attributes* and publicly available databases like voter registration data. For example, sensitive attributes can be disease names found in hospital records of individuals. On the other hand, age, zip-code, and gender are attributes which co-occur in both types of databases. These attributes are not sensitive themselves, but when linked with the hospital records, they can identify which individual suffers from what disease, thereby compromising his or her privacy. Hence, the set of these attributes constitute a *quasi-identifier.* A key assumption in all PPDM work is that the data owner is aware of what the quasi-identifiers are. The mining output is heavily dependent on the choice and number of quasi-identifiers.

1.2 Privacy-Preserving Data Visualization (PPDV)

We propose the development of privacy-preserving data visualization (PPDV) techniques which, like PPDM, aim to prevent linking sensitive data with externally available information. We see PPDV not just as a means to protect personal data, but also for a company's proprietary data. These two cases are different as there are no quasi-identifiers in the latter case, but we want to protect the attributes that are identified as *private* by the data owner.

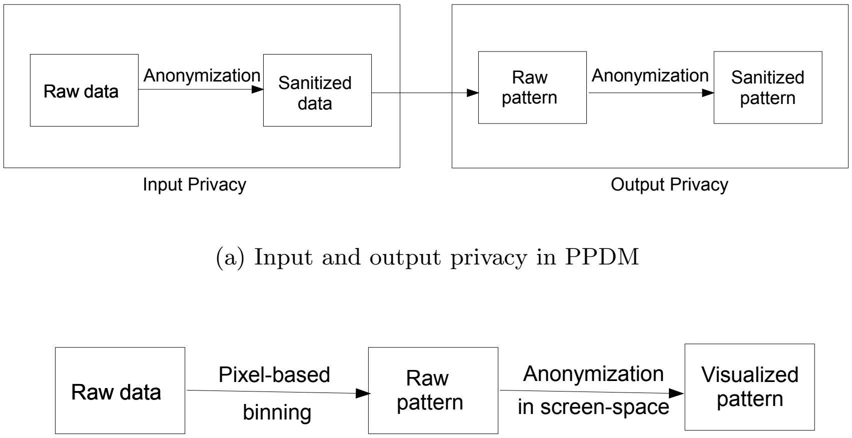
The key advantage of the visualization approach is that it provides the analyst with an adaptive tool where he or she can analyze the data according to his or her needs without breaching privacy. A visualization tool, which deals with de-identification at the screen-space and not the data-space, also potentially preserves more data fidelity than the PPDM approach, thereby making it more useful from the perspective of utility. The trade-off between privacy and utility in sanitized data has been well documented in the PPDM literature.[2,3] Several researchers have shown that even with minor privacy guarantees there is a heavy cost to utility. We aim to have an optimum balance between privacy and utility in our privacy-preserving data visualization approach; we believe that this can be achieved more effectively by having the visualization tightly coupled to the de-identification step (Figure 1b). Imposing privacy-preserving constraints in the screen-space, rather than the data space, meets these criteria. We demonstrate this idea in this paper through a PPDV technique based on screen-space clustering in parallel coordinates.

2. RELATED WORK

To our knowledge there has not been much work in the area of privacy-preserving data visualization. The only instance we are aware of uses graph-based abstraction of web data for privacy-preserving manifold visualization.[4] The authors, however, do not explicitly model privacy nor do they provide a detailed analysis of their approach, but just mention using a privacy-related parameter for controlling the clustering process. We aim to fill the void that exists with regards to privacy-preserving techniques in visualization and lay a foundation for this class of visualization techniques.

2.1 Output Privacy

Most techniques in the current PPDM literature focus on protection of *input privacy* where sanitized data is used for the mining input. However the model we follow uses raw data and deals with privacy protection in screen-space and not the data space. This scenario is similar to the concept of *output privacy*[5,6] and the anonymize-and-mine approach[7] (Figure 1a). Output privacy means that the mining output is manipulated in such a way that it cannot be reverse-engineered to breach privacy. In the visualization scenario, the user does



(a) Input and output privacy in PPDM

(b) Output privacy in visualization

Figure 1: Conceptualizing privacy in data mining and visualization: As we can observe, in our proposed visualization model the number of steps to achieve privacy is lesser. Therefore there is a higher likelihood of preserving the fidelity of the represented data.

not have access to raw data. Output privacy in the context of visualization is illustrated in Figure 1b. Here, even with interaction, he or she is not able to guess the finer granularity of the patterns seen on the screen. The advantage of this approach is that it keeps data distortion to a minimum and therefore maximizes the utility of the visualized data.

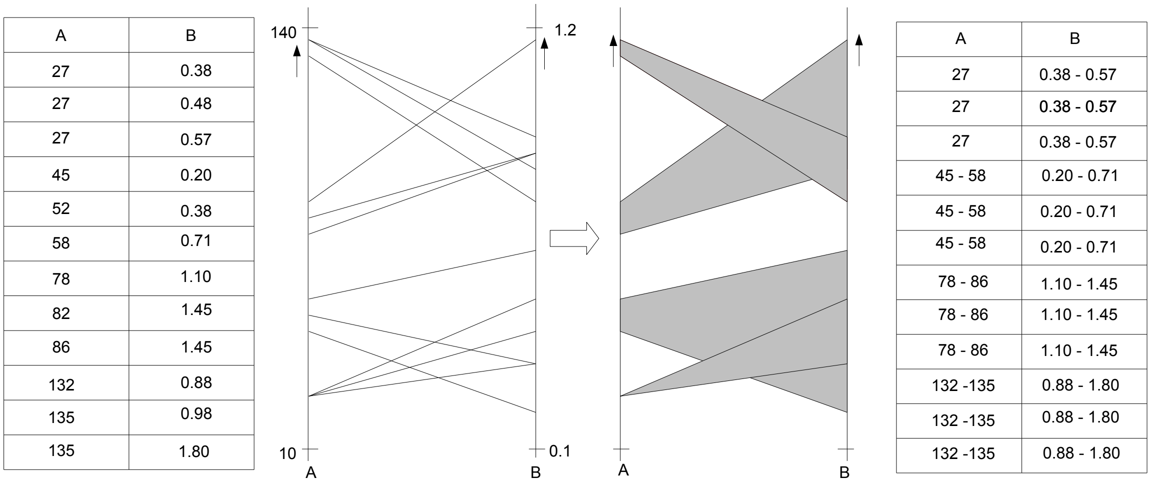
2.2 Approaches to Privacy Protection

The k-anonymity model is an approach to protect individual records from identification. It works by ensuring each data record in the table is indistinguishable from $k-1$ other records with respect to the quasi-identifiers in the table.[7,8] Conventional k-anonymity does not manipulate sensitive attributes and therefore cannot guarantee complete privacy. To overcome this shortcoming, the l-diversity,[9] α-k anonymity[10] approaches were proposed to introduce diversity in the sensitive attributes so that attackers with background knowledge about individuals cannot breach the privacy. In this paper, however we focus on the basic k-anonymity approach and investigate how it can be applied in parallel coordinates. Figure 2 illustrates the idea behind this approach.

k-anonymity problem has been shown to be NP-hard[11] and therefore many approximation algorithms have been proposed.[12, 13] We use the k-member clustering algorithm proposed by Byun et al.[12] This algorithm differs from the conventional k-means clustering algorithm in the sense that it restricts the cluster size to be at least k. While we adopt the overall algorithm proposed by Byun et al., we use a different criterion for seeding and a different cost function. We also adapt the algorithm to work individually for each axis pair, rather than across all dimensions at once.

2.3 Parallel Coordinates

Parallel coordinates[14] has been recognized as an effective tool for visualizing multidimensional, multivariate data. In the parallel coordinates research literature, there have been several approaches to clustering. Clustering has been used mainly for clutter reduction and overcoming the problem of over-plotting. Aggregation of lines and their traceability across different dimensions is a problem in parallel coordinates, even with relatively small datasets. Much of the clustering work, therefore, focuses on improving the perceptual aspects of parallel coordinates plots. Zhou et al. propose geometrically deforming and grouping poly-lines to overcome edge clutter.[15] Johansson et al. look at overcoming the problem of over-plotting by using high-precision textures.[16] Their goal is to maximize within-cluster information and show information at a detailed level of granularity. This is different from ours



A	B
27	0.38
27	0.48
27	0.57
45	0.20
52	0.38
58	0.71
78	1.10
82	1.45
86	1.45
132	0.88
135	0.98
135	1.80

A	B
27	0.38 - 0.57
27	0.38 - 0.57
27	0.38 - 0.57
45 - 58	0.20 - 0.71
45 - 58	0.20 - 0.71
45 - 58	0.20 - 0.71
78 - 86	1.10 - 1.45
78 - 86	1.10 - 1.45
78 - 86	1.10 - 1.45
132 -135	0.88 - 1.80
132 -135	0.88 - 1.80
132 -135	0.88 - 1.80

Figure 2: De-identification of lines in parallel coordinates by clustering. The left view shows visualization of the corresponding raw data. The right view shows visualization of the corresponding anonymized data

where we try to limit within-cluster information as much as possible. Artero et al.[17] apply clustering with the help of frequency and density information from the dataset to convey the distribution of the data effectively to the user. Ankerst et al.[18] propose clustering of dimensions based on some similarity metric. We also find clustering based on the data space properties using common clustering algorithms.[19, 20] The goal for clustering in our work is different from all of the above, since our primary goal is to protect the identity of individuals. This means that clustering needs to be done in such a way that the minimum size of clusters is guaranteed.

3. MODEL

The primary goal of privacy-preserving data visualization is that the user should not be able to drill down to the individual values for quasi-identifiers and/or the sensitive attributes in the raw data. Therefore, we intentionally hide information from the user by imposing de-identification constraints in the screen-space. To achieve this, we exploit two types of information loss: the inherent information loss in parallel coordinates and additional loss from grouping together records as a necessary step to achieve a desired level of privacy.

3.1 Information Loss

In parallel coordinates, there are several effects that lead to information loss: over-plotting of lines where the start and end values for different records map to the same pixels; strong convergence, especially in the case of categorical data, where many lines converge onto a single pixel; clutter, where too many lines are crossing between adjacent axes to make it impossible to identify individual records. In previous work[21] we have proposed a set of screen-space metrics to quantify these properties.

Some of this loss is desirable for the purpose of privacy protection, but it has to happen in a controlled manner. While we cannot always reduce the amount of information loss, we can increase it if necessary to combine enough records into a group (Figure 2). We currently only focus on loss through over-plotting, and use line convergence to seed clusters, but do not model all factors for information loss. The more we can model, however, the more we can make use of existing information loss, and the less we need to degrade the display.

3.2 Pixel-based Histograms

All computations are done in pixel coordinates. We first compute axis histograms for each data dimension: each bin counts the number of lines starting or ending in it (Figure 3). We term the number of data points per bin as its *degree* and utilize it for selecting the seed records in our clustering process (Section 4.1). The higher the

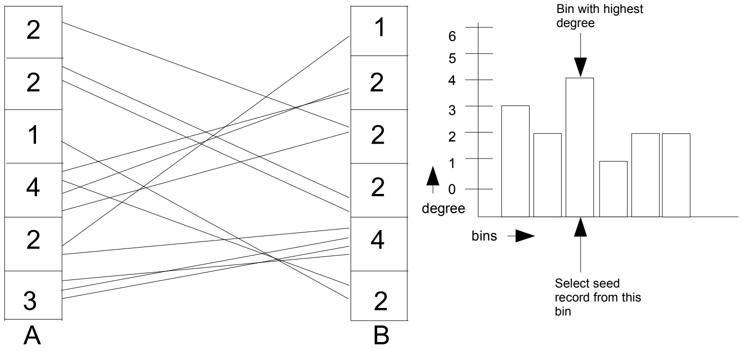


Figure 3: Seed record selection based on the degree histogram. Numbers in the boxes denote the degrees of the corresponding bins. The first seed record is chosen from the bin with the highest number of records.

degree, the higher is the aggregation of records in the bin, which makes for a natural starting point to cluster similar values together.

3.3 Axis Pairs

In conventional PPDM, while publishing data, the user sees the entire dataset. In contrast, the user is most likely to see structures between pairs of adjacent axes in parallel coordinates. As pointed out by Li et al.,[22] finding correlations in parallel coordinates means finding patterns between pairs of axes. Taking this idea further, we have shown that a parallel coordinates visualization can actually be considered as a sequence of axis-pairs.[21] While it is certainly possible to follow poly-lines across multiple dimensions in small datasets, it is not generally feasible in larger datasets and for all records.

We exploit this property by clustering the data by axis pairs, rather than clustering all dimensions at once. As we will demonstrate below, this leads to better preservation of visual structures and smaller cluster sizes.

3.4 Remote Visualization

While we have implemented the technique, the rest of the infrastructure is still a proof-of-concept. If the data is present on the user's machine, there are ways for the user to bypass our program and access it. A full implementation of this idea therefore would require a client-server model where the raw data resides on a server, and the visualization client can make requests to display it. The server sanitizes the data before it is sent to the visualization front-end. Since the server is told about the user's display resolution, it can create the appropriate clustering.

It should be noted that the visualization front-end cannot access more data than allowed by misrepresenting its display resolution: if the resolution is reported as higher than it is, the seeding only gets degraded, and thus produces larger clusters; the clusters do not get smaller, however. If it is reported as lower, the clusters end up being larger due to the assumed larger pixel sizes, as well.

4. IMPLEMENTATION

We have implemented a privacy-preserving variant of parallel coordinates. Based on the idea of k-anonymity, our program combines k records into one cluster, and displays it as a trapezoid instead of as individual lines (Figure 2). We have adapted an existing clustering mechanism in order to maximize the utility of the resulting clusters for visualization.

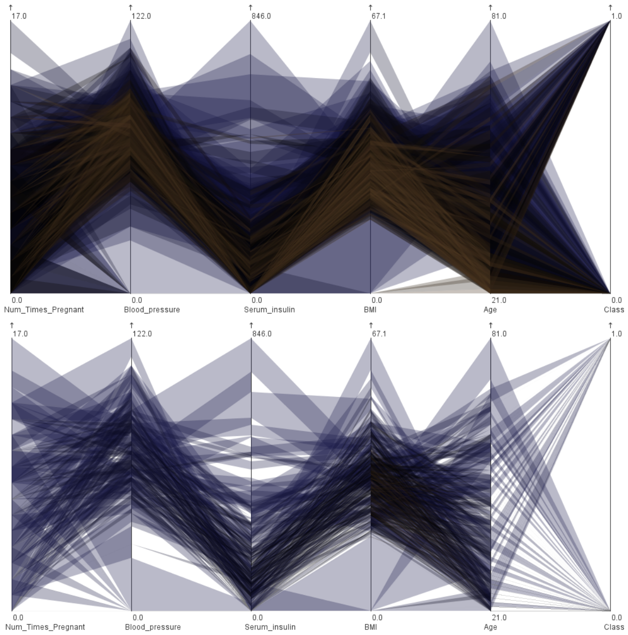


Figure 4: Multi-dimensional (left) and clustering by axis pair (right). Separate clustering for each axis pair creates smaller clusters that represent the structure of the data much better than clustering across all dimensions.

4.1 Clustering

Clustering is a classic means of grouping together similar objects into equivalence classes, which is similar to the idea behind k-anonymity. We adapt a technique called k-members clustering,[12] which uses an information loss metric to create clusters of records containing k records. The algorithm consists of the following three stages:

- *Seeding stage.* Given a set of n records in screen space, we choose the best one to start clustering with. This is often done randomly, but we use the axis histograms described above to start in areas of maximum density.
- *Cluster stage.* The next record is chosen in such a way that the chosen cost function is minimal. This is repeated until a size of k is reached for individual clusters and then repeated until the number of data items left is less than k.
- *Adjustment stage.* After the clustering process there might be some left-over records, which are then added to existing clusters so that the cost function is minimal.

4.1.1 Seed Dimension and Seed Record

The output of a clustering algorithm is strongly influenced by the seeding process. To create dense, small clusters, we use the axis histograms (Section 3.2) to select seed records in areas with a high degree (Figure 3). We are currently investigating different ways of picking the best axis to use for this, but have found that most axes that show an uneven distribution of records (categorical or at least have clear concentration points) to work well. When clustering by axis pair, we use the first dimension of each pair.

The seed is selected by finding the bin with the maximum count in the histogram, and using its associated pixel coordinate to find the first record. The value in this bin is then decremented by one. Each time a record is added to a cluster (and thus removed from the set of remaining records), its corresponding bin count is also decremented by one. The histogram thus tracks the distribution of values, and always provides an accurate picture of the distribution of the remaining records.

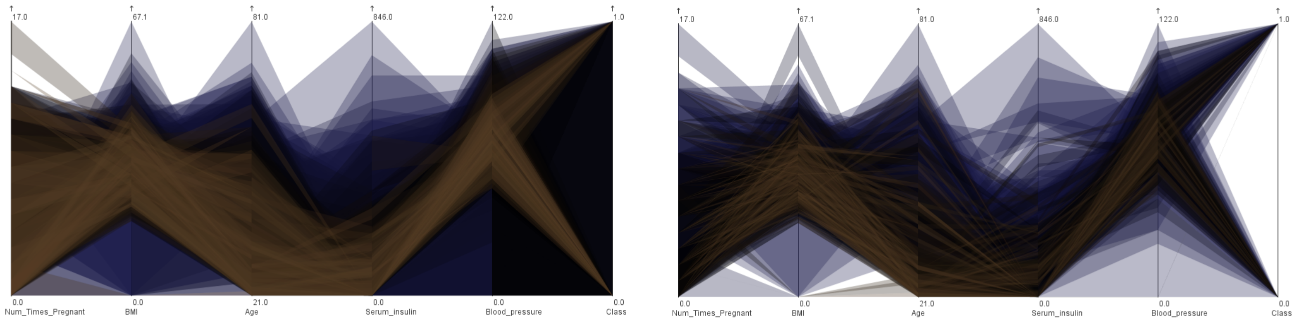


Figure 5: Cost functions used for clustering: Information loss (left) and Manhattan distance (right) used in multi-dimensional clustering. Manhattan distance leads to smaller clusters and more visible structures.

4.1.2 Cost Function

Clustering involves minimization of a cost function. The choice of an appropriate cost function is guided by the goal of a clustering process. Here the goal is finding similar records, and therefore minimizing distance among records within a single cluster. Byun et al.[12] use the information loss metric as the cost function. We initially also used metric for clustering, but it resulted in large clusters and a lot of occlusion (Figure 5, left).

The choice of distance metric is an important step in our algorithm. Aggarwal et. al. have shown through empirical evidence that in higher dimensional spaces relevance of L_t norm (where $t = 1, 2, ...n$) to higher values of t worsens faster with increasing number of dimensions.[23] Therefore lower values of t are preferable for large number of dimensions. This implies that L_1 norm or Manhattan distance outperforms Euclidean distance in such cases.

Besides the theoretical aspect, using the Manhattan distance also translates to better visual clarity in parallel coordinates. What can be seen much more clearly in the display than the Euclidean distance (which is part of the information loss metric used by Byun et. al.) is the Manhattan distance: the sum of distances on all axes. Using this metric, we obtain much denser clusters that preserve visual structures much better than the Euclidean distance.

4.1.3 Clustering Stage

The clustering uses a greedy approach that looks for the best fitting record for the current cluster.

1. Initialize the next cluster with the next seed record.
2. Compute the cluster centroid.
3. Select the next record in the cluster where the cost function relative to the centroid is minimal.
4. Iterate steps 2 and 3 until cluster size is k.
5. Repeat from step 1 until fewer than k records are left.
6. Repeat for each dimension if clustering by axis pair.

4.1.4 Adjustment Stage

If the number of records n is not a multiple of k, there will be records left over after the clustering described in the previous section is done. These records are assigned to the existing cluster which minimizes the cost function. A small number of clusters (fewer than k) thus end up with more than k records. For realistic dataset sizes and values of k, this makes no difference in practice.

4.1.5 Multidimensional Clustering and Clustering by Axis Pair

Traditional clustering results in a visualization that does not disclose private information, but also contains so many large clusters that almost all patterns in the data are lost (Figure 4, left). We therefore devised a new clustering approach, which clusters the data between each pair of axes separately, but shows them all in the same display. This is motivated by the fact that we are looking at parallel coordinates mostly as a collection of adjacent axes, rather than a true multi-dimensional visualization.[22]

Our axis-pair method results in much smaller clusters (Figure 4, right), which means less clutter as the overlap among clusters between adjacent axis pairs is minimized, and better preservation of salient structures in the data.

4.2 Rendering

Aggregation of lines and line crossings between axis pairs cause occlusion in parallel coordinates. This is an even bigger problem when we render clusters instead of lines. We therefore sort the clusters by size, and then draw them from the largest to the smallest. This kind of layering has been proven to be effective in other techniques like Parallel Sets,[24] because it prevents smaller polygons from being hidden behind larger ones.

In addition, we want to provide a stronger separation of the polygons by using different colors. Continuing with the idea of depth, we assign a blue color to the largest polygon (that is furthest "away") and an orange color to the smallest (and "closest"). This creates a slight depth-from-color effect[25] that makes the polygons easier to differentiate.

4.3 Interaction

Interaction is an important component of any visualization technique. The goal is to enable the user to explore the data according to his needs and drill down to individual values if necessary for his analysis. However, in the privacy-preserving context the constraint is that the user should not be able to see the individual values, especially on the quasi-identifier dimensions. Keeping this constraint in mind, we have designed a mouse-over interaction where connected clusters are highlighted between each axis-pair. Connected clusters are those which contain the same records.

In the case of multi-dimensional clustering, the clusters are continuous and appear as a connected aggregated structure. But in case of clustering by axis pairs, the clusters appear discontinuous, as they potentially split on each axis (Figure 8).

5. DISCUSSION

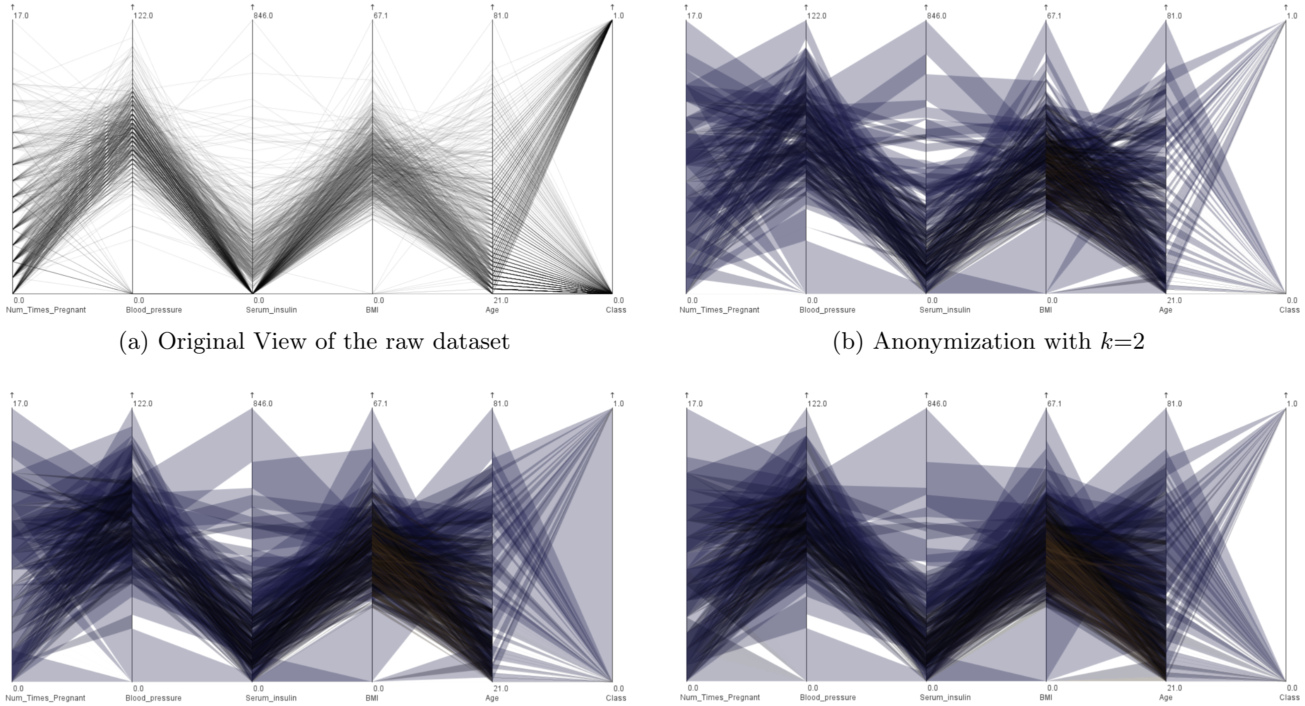
We use two metrics to discuss the utility of the visualizations created by our technique. These metrics have been chosen so that they quantify the data quality and the visual quality in both axis pair-wise and multi-dimensional clustering.

Cluster size: The size of the clusters is an important metric both in terms of data representation and visual quality. The larger the cluster size, the more it blurs the data. Also, visually clusters cause more clutter and are harder to discern in case of a larger dataset.

Cluster size is computed as the sum of the sizes of the cluster trapezoid on both axes. This gives us an estimate of information loss, because a larger cluster on each axis means that more precision of the data points is lost. As shown in Figure 7a, cluster size increases with increasing k. In the special case where $k = 1$ (i.e., the raw data), cluster size is zero. The increase in cluster size with k is vastly different between the two different types of clustering. Clustering by axis pair creates a much smaller increase than clustering across all dimensions. This can also be readily observed from the appearance of the visualization (Figure 4).

The differences between different axes given the same k are much smaller as shown in Figure 7b. Categorical dimensions tend to create diverging patterns, and thus much smaller cluster sizes. Most numerical dimensions ended up creating roughly equally-sized clusters in our testing.

Branching Factor: In clustering by axis pair, many clusters appear discontinuous because they get split on each axis. We compute the branching factor as the average number of clusters on the adjacent axis, into which



(a) Original View of the raw dataset

(b) Anonymization with k=2

(c) Anonymization with k=3

(d) Anonymization with k=4

Figure 6: k-anonymized view for the Diabetes dataset for different values of k

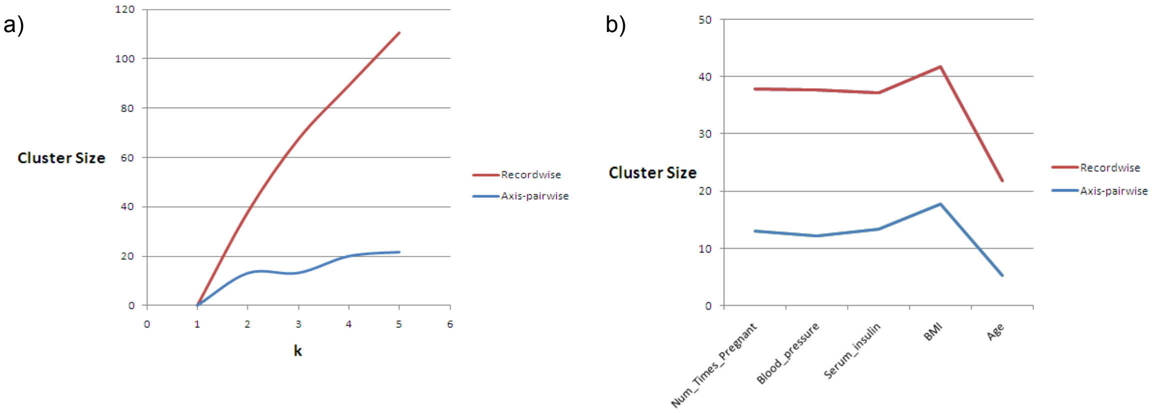


Figure 7: Variation of cluster sizes for axis-pairwise and multi-dimensional clustering. a) Variation of cluster sizes for different values of k; b) Variation of cluster sizes when k=2

clusters on one axis get split. This is zero in case of multi-dimensional clustering. In the diabetes dataset (Section 5.1), the average branching factor for an axis for k = 3 is 2.54. The larger the branching factor, the more we lose out on precision of data values. This affects the utility of parallel coordinates but at the same time has a positive effect on privacy protection as we demonstrate below.

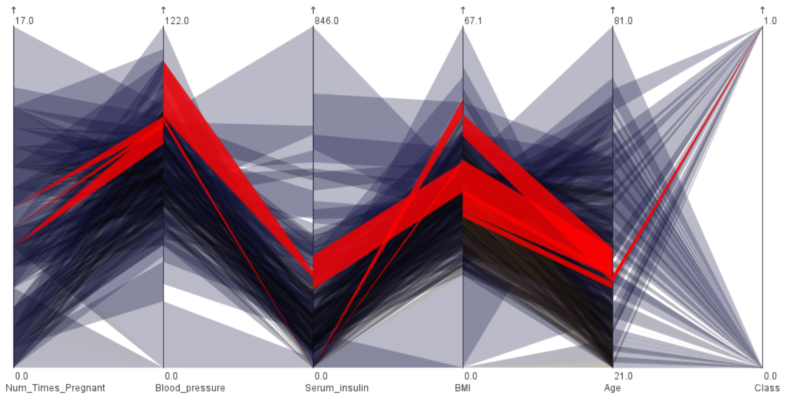


Figure 8: Visualizing split clusters with interaction in axis-pairwise clustering

5.1 Case Study: *Diabetes* Dataset

We use the *diabetes* dataset[26] for illustration of our approach. Datasets containing sensitive information are not typically publicly available. Therefore we treat the data dimensions in this dataset as sensitive attributes and quasi-identifiers. The dataset has 768 records and consists of 6 dimensions: *number of times pregnant*, *blood pressure*, *serum insulin level*, *body mass index (BMI)*, *age*, and the binary attribute *class*. The sensitive dimension is the *class* attribute, all others are considered to either make up the quasi-identifier attribute or constitute the set of attributes marked as private by a company which releases this tool.

Figure 6a shows the parallel coordinates display of the original data. Figure 6b-6d show the anonymized image for different values of k. Since we see aggregated values instead of single ones, represented by polygons instead of clusters, it is not possible to point to particular values on the axes. At the same time, much of the overall structure is still visible in the visualization, even though individual records cannot be identified anymore.

We want to emphasize the differences between multidimensional clustering and clustering by axis pair. Firstly, the latter helps us to better preserve the different structures in parallel coordinates (Figure 4). Secondly, the axis pair approach potentially achieves more privacy protection: clustering is done independently on each axis pair, so clusters tend to split with a high probability leading to non-zero branching factor. As shown in Figure 8, because of the cluster splits between all the dimensions from *number of times pregnant* to *age*, the average branching factor is greater than 2, leading to a lot of diversity among visualized clusters. Clusters already hide individual records and the splits provide an additional layer of privacy protection. For example if an attacker knows the range of values for the number of times pregnant dimension and therefore knows which cluster to look for, the diversity due to cluster splits makes it hard for him or her to exactly guess the values or ranges for the other dimensions.

This is similar, in principle, to the l-diversity approach[9] proposed by Machanavajjhala et. al., where the diversity in the sensitive attribute makes it difficult to link a quasi-identifier attribute to a unique sensitive attribute. In this case, the distinction is that the diversity exists among the different attributes of the quasi-identifier set. The absence of unique links makes a privacy breach more difficult. While we are not yet able to formally show how strong our approach is in this regard, we believe it to generally perform reasonably well.

5.2 Advantages and Drawbacks

Privacy-preserving visualization has the following advantages:

Minimize the loss of data fidelity. In the visualization approach, we work with the raw data and apply privacy protection at the screen space level. As was shown in Figure 1, reducing the number of transformation steps means that we retain the fidelity of the data as much as possible. Moreover, our approach is specifically tailored to visualization, thus giving us much better results than just aggregating values.

Dynamic exploration. Visualization offers a dynamic environment, where an analyst can interact with the data to glean important details. Even the data owner can try out different settings with the tool and manipulate what the untrusted user accesses. Our approach lets us dynamically react to what the user is seeing, and adjust the display accordingly.

The main drawback of our approach is a general issue with k-anonymity in PPDM: The method does not ensure sufficient diversity of values within each cluster. This can make it possible to get knowledge of the sensitive attribute for individuals even if only the clustered data can be seen. Given the high amount of information loss and the discontinuities of clusters in our approach, we do not believe this to be an issue, though.

Another issue is that even though we generally are closer to the visualization, we still lose a lot of visual clarity. We hope to exploit more metrics for seeding and clustering to more closely approximate the amount of information loss from the top.

6. CONCLUSION AND FUTURE WORK

In this paper we have proposed a privacy-preserving data visualization model based on the k-anonymity approach. We have discussed the differences with PPDM and shown that the visualization approach is a step towards achieving better usability of sensitive data. There are several possible extensions to this work. Currently, we have restricted ourselves to numeric data and in the future we would like to find out how we can achieve privacy protected visualization with categorical data. Moreover, we have only addressed the case where there is a single sensitive attribute. Multiple sensitive attributes will bring more complexity to the visualization process and we want to devise ways to handle this case. For the quasi-identifiers also, there can be multiple levels of privacy, because all identifiers are not identifiable with equal probability. To improve visibility of the clusters, we plan to improve our rendering algorithm so that there is less clutter.

On the interaction side we plan to bring in a number of enhancements: there should be more options for the data owner to change the settings of quasi-identifiers and sensitive attributes and see different candidate visualizations before releasing the tool, we also plan to show range of values on mousing over the clusters. We will also try out the enhanced privacy-preserving techniques like l-diversity and not only handle record-level privacy but also protect privacy of the clusters where required.

In addition to the enhancements to the privacy-preserving technique, a higher level goal is to achieve a tighter coupling between information loss in visualization and protection of data privacy. As an extension of our earlier work[21] we have demonstrated a practical application of quantifying information loss through a set of screen-space metrics. We plan on using other existing visualization techniques and pursue the question of information loss further to investigate which privacy guarantees already exists in these tools and how we can improve them.

REFERENCES

[1] Agrawal, R. and Srikant, R., "Privacy-preserving data mining," *ACM Sigmod Record* **29**(2), 439–450 (2000).

[2] Brickell, J. and Shmatikov, V., "The cost of privacy: destruction of data-mining utility in anonymized data publishing," in [*Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*], 70–78, ACM, New York, NY, USA (2008).

[3] Li, T. and Li, N., "On the tradeoff between privacy and utility in data publishing," in [*Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*], 517–526, ACM Press (2009).

[4] Zhang, X., Cheung, W. K., and Li, C. H., "Graph-based abstraction for privacy preserving manifold visualization," in [*WI-IATW '06: Proceedings of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*], 94–97, IEEE Computer Society, Washington, DC, USA (2006).

[5] Bu, S., Lakshmanan, L. V. S., Ng, R. T., and Ramesh, G., "Preservation of patterns and input-output privacy," in [*International Conference on Data Engineering*], 696–705 (2007).

[6] Wang, T. and Liu, L., "Butterfly: Protecting Output Privacy in Stream Mining," in [*IEEE 24th International Conference on Data Engineering*], 1170–1179 (2008).

[7] V. Ciriani, S. De Capitani di Vimercati, S. F. and Samarati, P., "k-anonymous data mining: A survey," in [*Privacy-Preserving Data Mining: Models and Algorithms*], Aggarwal, C. C. and Yu, P. S., eds., 105–136, Springer-Verlag (2007).

[8] Sweeney, L., "k-anonymity: A Model for Protecting Privacy," *IEEE Security And Privacy* **10**(5), 1–14 (2002).

[9] Machanavajjhala, A., Kifer, D., Gehrke, J., and Venkitasubramaniam, M., "l-diversity: Privacy beyond k-anonymity," *ACM Transactions on Knowledge Discovery from Data (TKDD)* **1**(1), 3 (2007).

[10] Wong, R. C.-W., Li, J., Fu, A. W.-C., and Wang, K., "(α, k)-anonymity: an enhanced k-anonymity model for privacy preserving data publishing," in [*Proceedings Conference on Knowledge Discovery and Data Mining (KDD)*], 754–759, ACM Press (2006).

[11] Meyerson, A. and Williams, R., "On the complexity of optimal k-anonymity," in [*PODS '04: Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*], 223–228, ACM, New York, NY, USA (2004).

[12] Byun, J.-W., Kamra, A., Bertino, E., and Li, N., "Efficient k-anonymization using clustering techniques," in [*Proceedings of the 12th international conference on Database systems for advanced applications*], 188–200 (2007).

[13] Aggarwal, G., Feder, T., Kenthapadi, K., Motwani, R., Panigrahy, R., Thomas, D., and Zhu, A., "Approximation algorithms for k-anonymity," in [*Journal of Privacy Technology*], (2005).

[14] Inselberg, A. and Dimsdale, B., "Parallel coordinates: A tool for visualizing multi-dimensional geometry," in [*IEEE Visualization*], 361–378, IEEE CS Press (1990).

[15] Zhou, H., Yuan, X., Qu, H., Cui, W., and Chen, B., "Visual clustering in parallel coordinates," in [*Computer Graphics Forum*], **27**(3), 1047–1054 (2008).

[16] Johansson, J., Ljung, P., Jern, M., and Cooper, M., "M.: Revealing structure within clustered parallel coordinates displays," in [*In Proceedings of the 2005 IEEE Symposium on Information Visualization*], 125–132 (2005).

[17] Artero, A., De Oliveira, M., and Levkowitz, H., "Uncovering Clusters in Crowded Parallel Coordinates Visualizations," in [*IEEE Symposium on Information Visualization*], 81–88 (2004).

[18] Ankerst, M., Berchtold, S., and Keim, D., [*Similarity clustering of dimensions for an enhanced visualization of multidimensional data*], Proceedings of the International Conference on Information Visualization (1998).

[19] Fua, Y.-H., Ward, M. O., and Rundensteiner, E. A., "Hierarchical parallel coordinates for exploration of large datasets," in [*Proceedings of the Conference on Visualization*], 43–50, IEEE Computer Society Press, Los Alamitos, CA, USA (1999).

[20] Novotny, M. and Hauser, H., "Outlier-preserving focus+context visualization in parallel coordinates," *IEEE Transactions on Visualization and Computer Graphics* **12**(5), 893–900 (2006).

[21] Dasgupta, A. and Kosara, R., "Pargnostics: Screen-space Metrics for Parallel Coordinates," in [*IEEE Conference on Information Visualization*], (2010, in press).

[22] Li, J., Martens, J.-B., and van Wijk, J. J., "Judging Correlation from Scatterplots and Parallel Coordinate Plots," *Information Visualization* **9**(1), 13–30 (2010).

[23] Aggarwal, C. C., Hinneburg, A., and Keim, D. A., "On the surprising behavior of distance metrics in high dimensional space," in [*Lecture Notes in Computer Science*], 420–434, Springer (2001).

[24] Kosara, R., Bendix, F., and Hauser, H., "Parallel sets: Interactive exploration and visual analysis of categorical data," *Transactions on Visualization and Computer Graphics (TVCG)* **12**, 558–568 (July/August 2006).

[25] Piringer, H., Kosara, R., and Hauser, H., "Interactive focus+context visualization with linked 2d/3d scatterplots," in [*2nd International Conference on Coordinated and Multiple Views in Exploratory Visualization (CMV)*], 49–60 (2004).

[26] "UC Irvine Machine Learning Repository." http://archive.ics.uci.edu/ml/ (Accessed July 2010).