Evaluation of Classification Algorithms on Synthesized Data Based on Differential Privacy

Rachit Shaha[1]

Abstract

Data Synthesis is the process to generate new data from an existing dataset so as to increase the scalability and flexibility of machine learning practitioners to increase the performance of model and predict the output for even hypothetical scenarios which may come in the future. Data Synthesis is also applied on datasets containing Personally Identifiable Information, to hide these attributes to maintain fairness and privacy. This is done by adding noise to the distributions to preserve privacy. In this paper I have included the results of my evaluation of implementing various classification algorithms on the original dataset and the synthetic dataset to measure the difference of performance in both of these datasets.

1. Introduction

In this era of computer revolution data is one of the most important elements of processing. Now this data maybe anything belonging to any sector containing information of anything in the world. On a few occasions this data can represent some sensitive information of individuals, groups of people, any business-related transactions etc. which may not be appropriate to access by other agencies. Yet to process the data it has to be handled by some third-party agency which cannot be prevented so, to maintain the privacy we use Data Synthesis based on Differential Privacy such that the data is still in a state of processing but the information is secured. This is achieved by adding noise to data in such a way that the knowledge and value the data possess isn't lost. But adding this noise is definitely going to have some effect when machine learning algorithms are implemented. I have evaluated this performance effect for some classification algorithms namely Naïve Bayes, Support Vector Machines, Logistic Regression Based, some Native Boosting and Tree Based Classification Algorithm. I have used the Adult Income per year dataset available on UCI Machine Learning Repository which contains some personally identifiable attributes like race, age, sex. Further I have generated a synthetic dataset for this original dataset using DataSynthesizer Tool available on this site: https://github.com/DataResponsibly/DataSynthesizer

2. Working

2.1. Data Pre-Processing

The Adult dataset contains a total of 14 attributes and 48842 instances. To demonstrate the concept of Differential Privacy I have selected a few attributes among all 14 which actually affect the performance of model. These attributes were age, sex, education, marital-status, relationship and salary. Further to identify each instance uniquely I considered a candidate key SSN representing the Social Security Number for each instance(person).

2.2. Synthetic Data Generation

After preprocessing, I generated the synthetic data for this dataset using the DataSynthesizer Tool. The data I generated contained 1000 instances and the pattern was very close to the actual dataset. The histogram representation for each instance of both, the original as well as the synthetic dataset is as follows:

2.2.1. For attribute Age:

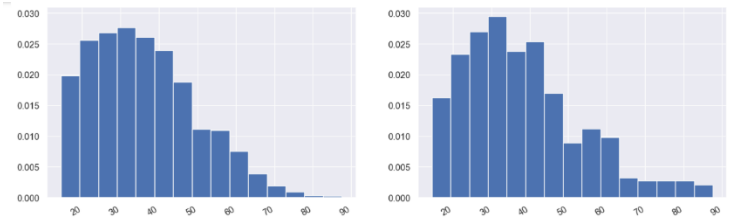

Figure 1. Histogram Representation of attribute Age in Original Dataset vs Synthetic Dataset.

2.2.2. For attribute Education:

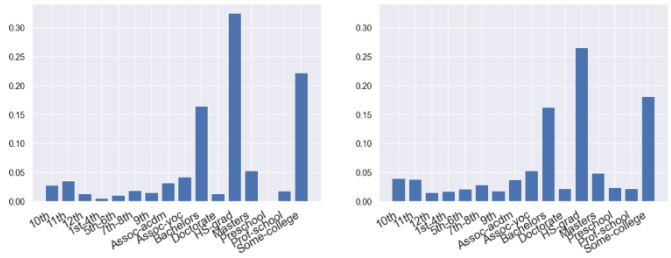

Figure 2. Histogram Representation of attribute Education in Original Dataset vs Synthetic Dataset.

2.2.3. For attribute Sex:

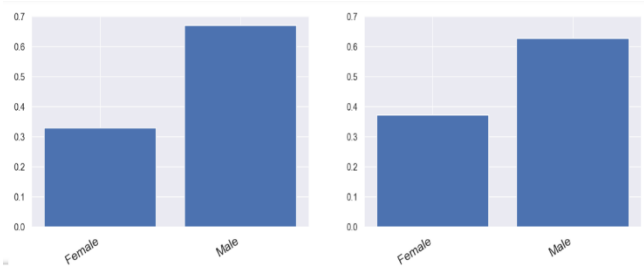

Figure 3. Histogram Representation of attribute Sex in Original Dataset vs Synthetic Dataset.

2.2.4. For Attribute Marital-Status:

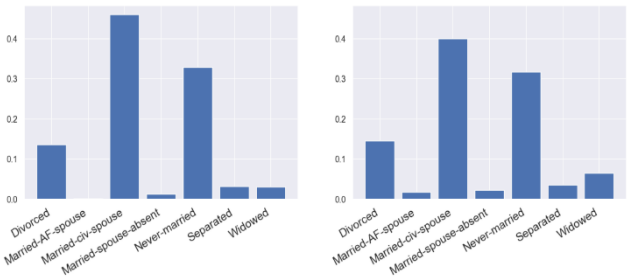

Figure 4. Histogram Representation of attribute Marital-Status in Original Dataset vs Synthetic Dataset.

2.2.5. For Attribute Relationship:

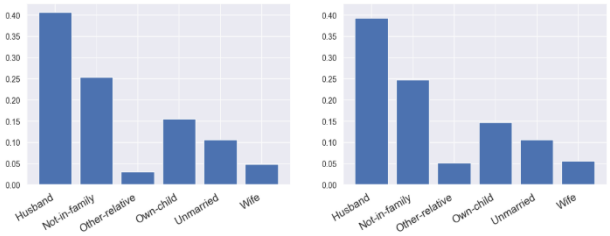

Figure 5. Histogram Representation of attribute Relationship in Original Dataset vs Synthetic Dataset.

2.2.6. For attribute Salary:

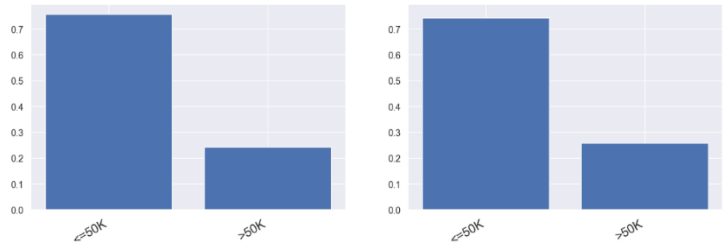

Figure 6. Histogram Representation of attribute Salary in Original Dataset vs Synthetic Dataset.

As seen in the graph the synthetic data set is pretty close to the original dataset in terms of representation of data as whole. We can see some differences here and there like in the original dataset there were very less instances of people who completed their education till preschool but in the synthetic data this number was comparatively more. Same can be seen in the case of people who had Marital Status as Married-AF-spouse.

The co-relation between the attributes in each of this dataset also had a very minor difference. In the case of Original Dataset, the co-relation between age and education was less which was seen increased in the Synthetic Dataset.

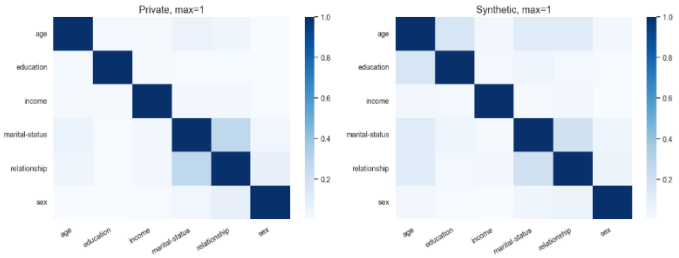

Figure 7. Heatmap Representation of co-relation between attributes in Original Dataset vs Synthetic Dataset.

As we can spot in the Heatmap the attributes age and education have more co-relation (darker square) in Synthetic Dataset. But if these small differences are neglected it is hard to distinguish both datasets.

This can be seen as a very promising fact as we can in fact generate loads of data without changing the knowledge as well as its value which play an important role in processing this information.

2.3. Implementing Classification Algorithms.

I have selected the following algorithms to implement on the above generated datasets.

- Naïve Bayes
- Logistic Regression Based
- Support Vector Based
- Native Boosting
- Tree Based Classification

First, I implement them on the Original Dataset, record the accuracy and then I use the same model to predict output on the Synthetic Dataset. I used cross-validation in each algorithm to prevent overfitting since the synthetic data generated had only 1000 instances. I set the k value as 10. These were the following results in each case:

2.3.1 NAÏVE BAYES ALGORITHM

Naïve Bayes Algorithm gives an accuracy of 76.7619% on the original dataset.

The confusion matrix is as follows:

a	b	
10682	2967	a => less than 50k
1224	3162	b=> more the 50k

Figure 8. Confusion Matrix for Naïve Bayes on the Original Dataset.

The same model gives an accuracy of 71.25% on the synthetic dataset.

The confusion matrix is as follows:

a	b	
359	59	a => less than 50k
102	40	b=> more the 50k

Figure 9. Confusion Matrix for Naïve Bayes on the Synthetic Dataset.

The difference in accuracy is around 5%, being more on the Original Dataset.

2.3.2 LOGISTIC REGRESSION BASED

Logistic Regression Based algorithm yields an accuracy of 79.2736% on the original dataset. This is more than that of Naïve Bayes.

The confusion matrix is as follows:

a	b	
12852	797	a => less than 50k
2941	1445	b=> more the 50k

Figure 10. Confusion Matrix for Logistic Regression Based Algorithm on the Original Dataset.

The same model gives an accuracy of 72.1429% on the synthetic dataset which is less than accuracy over Original Dataset but it is still more than accuracy over Synthetic Dataset when Naïve Bayes Algorithm was implemented.

The confusion matrix is as follows:

a	b	
401	17	a => less than 50k
139	3	b=> more the 50k

Figure 11. Confusion Matrix for Logistic Regression Based Algorithm on the Synthetic Dataset.

In this case the difference between accuracy is just over 7%.

2.3.3 SUPPORT VECTOR BASED CLASSIFICATION

Support Vector Based Algorithm yields an accuracy of 84.5214% on the original dataset. This is the highest accuracy among all the algorithms.

The confusion matrix is as follows:

a	b	
13671	774	a => less than 50k
2017	1572	b=> more the 50k

Figure 12. Confusion Matrix for Support Vector Based Algorithm on the Original Dataset.

The same model gives an accuracy of 81.76% on the Synthetic Dataset which is again the highest when compared to any other algorithm's accuracy on the Synthetic Dataset.

The confusion matrix is as follows:

a	b	
452	2	a => less than 50k
100	6	b=> more the 50k

Figure 13. Confusion Matrix for Support Vector Based Algorithm on the Synthetic Dataset.

In this case the difference is just below 3%.

2.3.4 NATIVE BOOSTING USING AdaBoost and LogitBoost

AdaBoost Algorithm yields an accuracy of 75.7361% on the original dataset. The confusion matrix is as follows:

a	b	
13287	362	a => less than 50k
4014	372	b=> more the 50k

Figure 14. Confusion Matrix for AdaBoost Algorithm on the Original Dataset

Further, the same model gives an accuracy of 74.6429% on the synthetic dataset having the least difference in the accuracy between Original and Synthetic Dataset among all the algorithms.

The confusion matrix is as follows:

a	b	
418	0	a => less than 50k
142	0	b=> more the 50k

Figure 15. Confusion Matrix for AdaBoost Algorithm on the Synthetic Dataset

The LogitBoost Algorithm yields an accuracy of 79.5897% on the Original Dataset. The confusion matrix is as follows:

a	b	
12714	935	a => less than 50k
2746	1640	b=> more the 50k

Figure 16. Confusion Matrix for LogitBoost Algorithm on the Original Dataset

The same model results in an accuracy of 74.6429% on the synthetic dataset, which is same as AdaBoost Algorithm. The confusion matrix is as follows:

a	b	
416	2	a => less than 50k
140	2	b=> more the 50k

Figure 16. Confusion Matrix for LogitBoost Algorithm on the Synthetic Dataset

2.3.5 TREE BASED CLASSSIFICATION

Decision Tree is used to classify the dataset. For the original dataset the size of tree was 42 and number of leaf nodes were 35. The accuracy was 79.0685% on the original dataset. The confusion matrix is as follows:

a	b	
13302	347	a => less than 50k
3428	958	b=> more the 50k

Figure 17. Confusion Matrix for Tree Based Algorithm on the Original Dataset

The same model when implemented on the Synthetic Dataset yields an accuracy of 74.6429% same as AdaBoost and LogitBoost. The confusion matrix is as follows:

a	b	
418	0	a => less than 50k
142	0	b=> more the 50k

Figure 18. Confusion Matrix for Tree Based Algorithm on the Synthetic Dataset

As per the accuracy the best algorithm that performed well on both the datasets was the Support Vector Based Classification Algorithm. The AdaBoost algorithms performed almost similar on both the datasets but SVM still had the highest accuracy.

3. Experiments and Observations

I conducted a few experiments over different hyperparameters and the attributes in the dataset. While using the cross-validation function I tried the algorithm for various values of k. For its lowest value, which is k = 2, the training accuracy was highest since there may be some overfitting present. As k value increased the accuracy very gradually decreased but after k=10 it was almost constant.

For the attributes, I deleted some attributes and implemented the algorithms to check the effect of absence of a particular variable. For all the attributes, when they were deleted, the accuracy decreased except in the case of attribute marital-status the accuracy increased on an average on 3% in each case.

As per the final accuracy Support Vector Based Algorithm performed best among all the algorithms on both datasets. Even the difference was almost the least when compared to other algorithms.

Table 1. Algorithms and their accuracy on respective datasets.

Algorithm	Accuracy on Original Dataset	Accuracy on Synthetic Dataset	Difference
SVM	84.52	81.76	2.76
LogitBoost	79.58	74.64	4.94
Logistic Regression Based	79.27	72.14	7.13
Tree Based	79.06	74.64	4.42
Naive Bayes	76.76	71.25	5.51
AdaBoost	75.73	74.64	1.09

Following the above table one can infer that, surely there is a drop in the performance of the algorithms but this can be increased by a few things like:

- Filling the missing values in the Original Dataset so that Synthetic Dataset doesn't fill them with some outliers or incorrect data.
- While implementing differential privacy the noise which is added must be in check so it doesn't hurt the dataset information.
- Increasing the data samples in this case can also lead to a good accuracy.

4. Conclusion

Data Synthesis brings a lot of advantages on the table across all the domains of machine learning applications. As we observed that the algorithms predicted the output from the synthetic dataset with almost the same accuracy as they predicted on the original dataset and there also being a scope for improvement, data synthesis can be used in various situations.

1. When original data is too small synthetic data can be generated and a well-functioning model can be built which can perform likewise if there was enough data.

2. Using synthetic data one can simulate some specific hypothetical situation which may not have been occurred in past but are likely to happen in future and the model should make a decision in that case.

3. By using the concept of differential privacy, data can remain secured and can be shared without any restriction to third party agencies for processing like in the sectors of Health-care, Military, Social media, Hiring Agencies etc.

Something like this can be beneficial for achieving Responsible AI in all the applications. Maintaining fairness, security, monitorable and human-centered data can be easily done with the help of Data Synthesis.

With a few improvements more Synthetic Datasets can definitely fill the void for solving the Data Sharing and Limited Data issues.

5. References

Haoyue Ping, julia Stoyanovich, Bill Howe *DataSynthesizer: Privacy-Preserving Synthetic Datasets.* 2017.

Lucas Rosenblatt, Xiaoyan Liu, Samira Pouyanfar, Eduardo de Leon, Anuj Desai, Joshua Allen *Differentially Private Synthetic Data: Applied Evaluations And Enhancements.* 2020

What Is Data Synthesis, and Why Are We Calling It Data Mimicking? https://www.tonic.ai/blog/what-is-data-synthesis-and-why-are-we-calling-it-data-mimicking

UCI Machine Learning Repository. https://archive.ics.uci.edu/ml/datasets/adult

Top 20 Synthetic Data Use Cases & Applications in 2023. https://research.aimultiple.com/synthetic-data-use-cases/