BUYER BEWARE: UNDERSTANDING THE TRADE-OFF BETWEEN UTILITY AND RISK IN CART BASED MODELS USING SIMULATION DATA

Berlin,
7-8. Oktober, 2024

Jonathan Latner, PhD
Dr. Marcel Neuenhoeffer
Prof. Dr. Jörg Drechsler

SECTION 1: INTRODUCTION

BACKGROUND

- We don’t know.
- We think this idea is interesting.
- We think others should know about it.
- But we are still developing it.
- Thoughts and suggestions would be helpful.

OVERVIEW

- It is well established that there is a trade-off between utility and privacy when generating synthetic data
- Utility and privacy in CART based synthesizers is high (Little et al., 2022; Danker and Ibrahim, 2021)
- Therefore, CART models are less sensitive to this trade-off than other SDGs (i.e. higher utility, lower risk)
- Using simulation data (Reiter et al., 2014), we show that synthetic data from CART models are disclosive
- The problem:
  - Disclosive in ways that are not observable with common privacy metrics
  - It is possible to increase protection (by reducing utility)

WHATS THE GOAL OF SYNTHETIC DATA?

- Synthetic data can accelerate development by replacing sensitive values with synthetic ones with minimal distortion of the statistical information contained in the original data set. (Jordan et al., 2022; Nowak et al., 2016)
- Low disclosure risk (R)
- High data utility (U)
- Visualize the trade-off using the R-U confidentiality map (Duncan et al., 2004)

WHATS THE PROBLEM?

- High data utility – It must be similar to and different from the original data.
  - At the extreme, if the goal is high utility, why not just release the original data?
- Low disclosure risk – Synthetic data is not automatically private.
  - At the extreme, if the goal is low privacy risk, why should we release any data?
- Many measures of utility and privacy exist
  - Therefore, its not clear if data have high utility or low risk
  - 2 problems
    - More specifically, how can we map R-U trade-off if there are multiple measures of both?
    - More generally, how do we know if the data have high levels of utility and low levels of privacy?

WHAT DO WE KNOW?

- SDG with CART is a good option
- Utility
  - Drechsler and Reiter (2011) found that CART models offered the best results in terms of preserving the information from the original data.
  - Other comparisons also found CART is superior (Little et al., 2022; Danker and Ibrahim, 2021)
- Privacy
  - Some evidence also suggests CART is superior (Little et al., 2022)
  - However, other evidence indicates that CART-based synthesis simply replicates most of the original records (Manrique-Vallier and Hu, 2018)

WHAT DO WE NOT KNOW?

- How are CART models so good at minimizing trade-off between risk and utility?

SECTION 2: DATA AND METHODS

DATA AND METHODS

- Data - simulated data with unique record (Reiter et al., 2024)
- Utility measures (synthpop - Raab et al., 2021)
  - Voas Williamson
  - Freeman-Tukey
  - Jensen-Shannaon divergence
  - Kolmogorov-Smirnov statistic
  - Propensity score mean-squared error
  - Bhattacharyya distances
- Privacy measures (synthpop - Raab et al., 2024)
  - Identity disclosure measure: the ability to identify individuals in the data from a set of known characteristics (i.e. ‘keys’).
  - Attribute disclosure measure: the ability to find out from the keys something, not previously known
  - Replicated uniques

SIMULATE DATA WITH A UNIQUE RECORD

Borrowing from Reiter et al. (2014), we set the first 999 observations to be a random sample from a multinomial distribution for all combinations of $var1(0,1)$, $var2(0,1)$, $var3(0,1)$, $var4(0,1)$ except (var1=1,var2=1,var3=1,var4=1), which we set to be the 1000th observation.

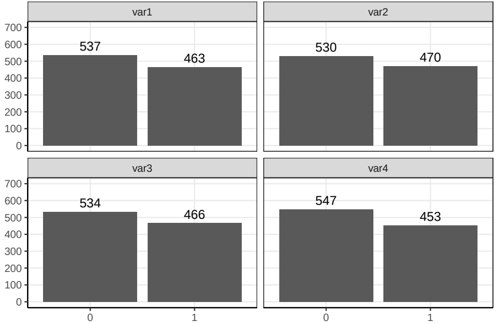


Figure 1: Frequency

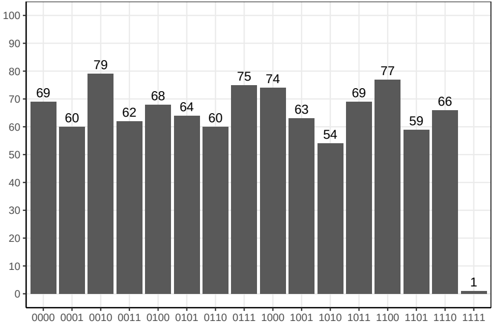


Figure 2: Histogram

GENERATE SYNTHETIC DATA WITH CART (SYNTHPOP)

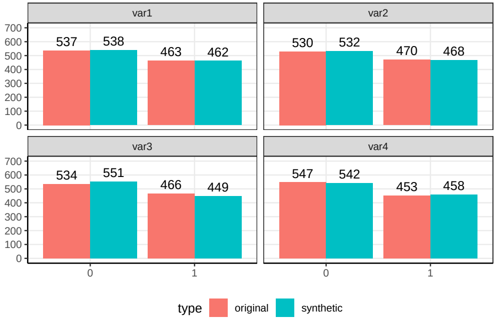


Figure 3: Frequency

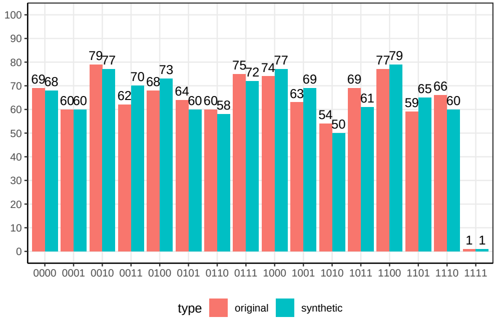
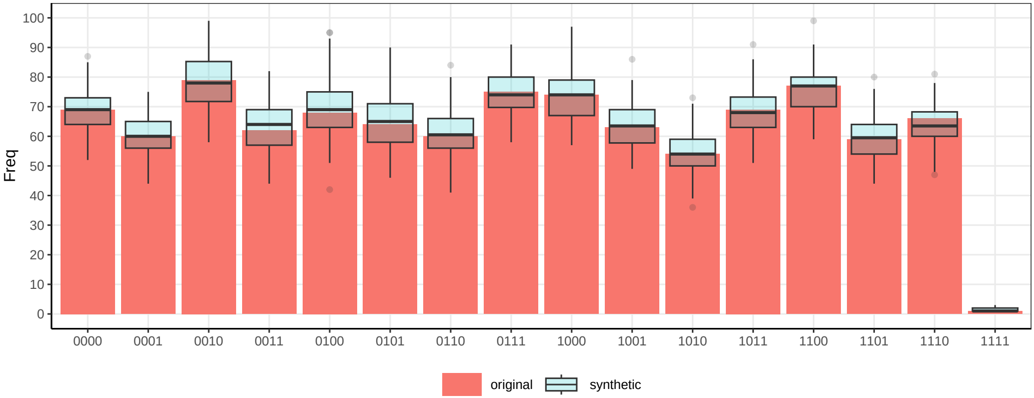


Figure 4: Histogram

COMPARE HISTOGRAM X 100 SYNTHETIC DATASETS

Figure 5



COMPARING UTILITY MEASURES

Table 1

name	var1	var2	var3	var4	average
Voas Williamson utility measure	0.00	0.02	1.16	0.10	0.32
Freeman-Tukey utility measure	0.00	0.02	1.16	0.10	0.32
Jensen-Shannaon divergence	0.00	0.00	0.00	0.00	0.00
Kolmogorov-Smirnov statistic	0.00	0.00	0.02	0.01	0.01
propensity score mean-squared error	0.00	0.00	0.00	0.00	0.00
Bhattacharyya distances	0.00	0.00	0.01	0.00	0.00

COMPARING PRIVACY MEASURES (SET.SEED = 1237, I.E. UNIQUE = 1)

```
> print(t1, plot = FALSE, to.print = "ident")
Disclosure risk for 1000 records in the original data

Identity disclosure measures
from keys: var1 var2 var3
For original  ( UiO )  0 %
For synthetic ( repU ) 0 %.
> print(t1, plot = FALSE, to.print = "attrib")

Table of attribute disclosure measures for var1 var2 var3
Original measure is  Dorig and synthetic measure is DiSCO
Variables Ordered by synthetic disclosure measure

       attrib.orig attrib.syn check1 Npairs check2
1 var4           0          0             0
```

```
> replicated.uniques (sds, df_ods)
    var1 var2 var3 var4
973    1    1    1    1
Uniques and replicated uniques for  1  synthesised data set(s)
 from keys:  var1 var2 var3 var4

Uniques in  original data:
 1 from  1000 records ( 0.1 %)
Uniques in synthetic data:
 1 from  1000 records ( 0.1% )

Replicated uniques:
 1
as a % of uniques in synthetic  100%
as a % of original records (repU) 0.1%
```

COMPARING PRIVACY MEASURES (SET.SEED = 1240, I.E. UNIQUE = 3)

```
> print(t1, plot = FALSE, to.print = "ident")
Disclosure risk for 1000 records in the original data

Identity disclosure measures
from keys: var1 var2 var3
For original  ( UiO )  0 %
For synthetic ( repU ) 0 %.
> print(t1, plot = FALSE, to.print = "attrib")

Table of attribute disclosure measures for var1 var2 var3
Original measure is  Dorig and synthetic measure is DiSCO
Variables Ordered by synthetic disclosure measure

       attrib.orig attrib.syn check1 Npairs check2
1 var4           0          0              0
```

```
> replicated.uniques (sds, df_ods)
Uniques and replicated uniques for  1  synthesised data set(s)
 from keys:  var1 var2 var3 var4

Uniques in  original data:
 1 from  1000 records ( 0.1 %)
Uniques in synthetic data:
 0 from  1000 records ( 0% )

Replicated uniques:
 0
as a % of uniques in synthetic  NaN%
as a % of original records (repU) 0%
```

SUMMARY

- Using common privacy and utility measures, CART generates synthetic data with both high utility and low risk
- However (and this is the point):
  - We know there is a problem (because we created it)
  - We know that common measures do not capture the problem
- We are also not alone in identifying this problem (Manrique-Vallier and Hu, 2018)

SECTION 4: SOLUTION

- The good news: we know how to solve the problem
- The bad news: we don’t know how to identify the problem

THE GOOD NEWS: SOLUTIONS

- minumlevels = 5: Ensures the data are treated as categorical
- cp = 0.05 (default = $1e^{-8}$): prevent large trees (i.e. overfitting)
- minbucket = 75 (default = 5): the minimum number of observations in any terminal node
- Other options also exist

VISUALIZING TREES (DEFAULT VS. MODIFIED)

Figure 6: CART (default)

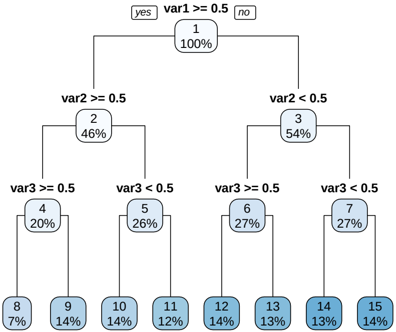
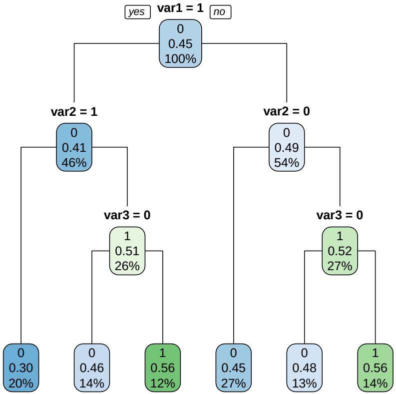


Figure 7: CART (modified)



COMPARE HISTOGRAM X 100 SYNTHETIC DATASETS

Figure 8: CART (default)

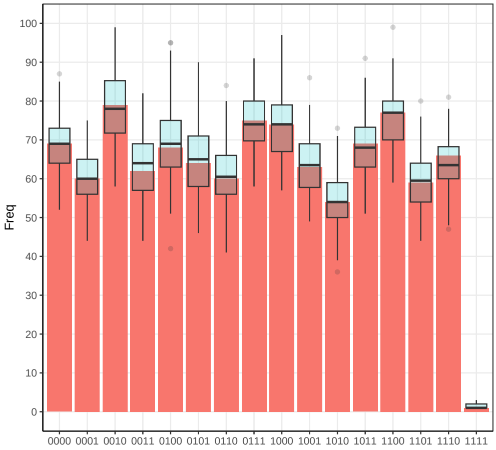
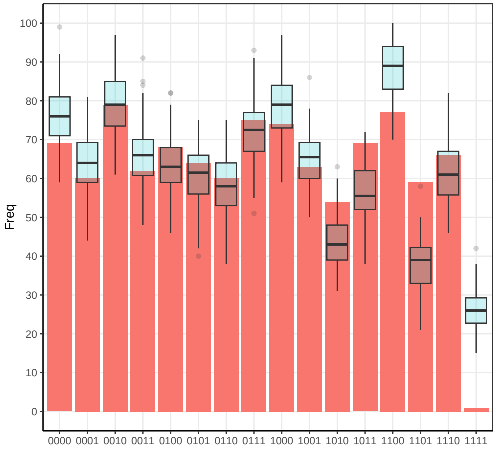


Figure 9: CART (modified)



THE BAD NEWS

- Specifically, we don’t know how to identify the privacy risk
- Generally, we have to know a problem exists before we would do something about it

SECTION 5: CONCLUSION

CONCLUSION

- It has long been understood that there is a trade-off between utility and risk
- Previous research indicated that CART models were less sensitive to this trade-off than other SDGs
- Using a simulated data set, we show that CART are sensitive to this trade-off
- The good news: It is possible to reduce risk in CART with parameters
- The bad news:
  - We must sacrifice utility
  - Common privacy metrics do not capture risk in our simulated data

QUESTIONS

- If you did not know there was a problem, why would you sacrifice utility?
- What do you think?
- Should we develop this further?
- If so, how and in what direction?
- What would be the contribution?

CONTACT

Jonathan Latner

jonathan.latner@iab.de

Reproducible code:

- Github: https://github.com/jonlatner/KEM_GAN/tree/main/latner/projects/simulation