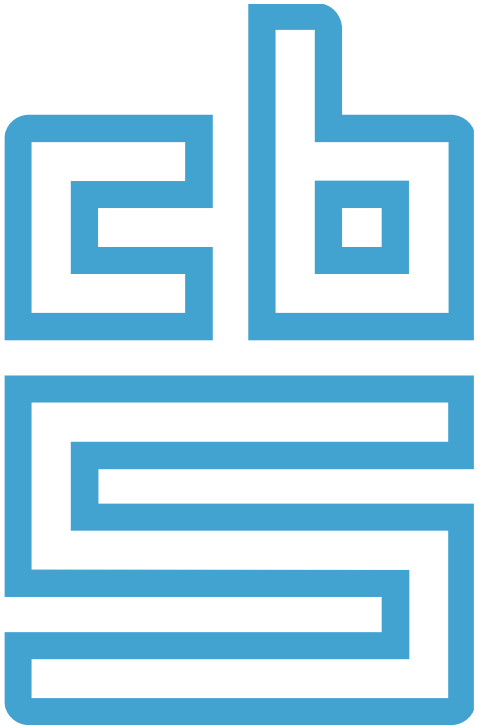


Robust and fast data synthesis with the synthesizer package.

Use of R in Official Statistics 2025 #uRos2025

Mark van der Loo[1,2], Marije Sluijskes[2], Mishca Jacobs[2] and Maria Anthoulaki[2]
[1]Statistics Netherlands (CBS), [2]Leiden University

12-11-2025 | www.markvanderloo.eu

Interface

```
synthesize()        # synthesize a vecor or data.frame

make_synthesizer() # create a function that samples
                     from the synthetic distribution

synthesize(rho=r)  # lower rank correlation between
                   # original and synthetic data to r
                   # (possibly per variable)

synthesize(na.rm=TRUE)  # Remove missings before
                        # synthesizing and get a complete
                        # dataset
```

The synthesizer algorithm

For each variable:

1. Sample *n* values from empirical distribution.
2. Reorder so that the rank of the sample matches the rank of the original *n* values.

Rank order trick: M. Sluijskes (2021): *Imputation of business survey data: A systematic comparison between ratio and random forest-based imputation methods. Leiden University*

The synthesizer algorithm

- Integer, categorical data: resample from observations.
- Numeric data: linear intepolation of formal ECDF

For each variable:

1. Sample *n* values from empirical distribution.
2. Reorder so that the rank of the sample matches the rank of the original *n* values.

Rank correlation between original and synthetic data equals 1.

The synthesizer algorithm: privacy

- Integer, categorical data: resample from observations.
- Numeric data: linear intepolation of formal ECDF

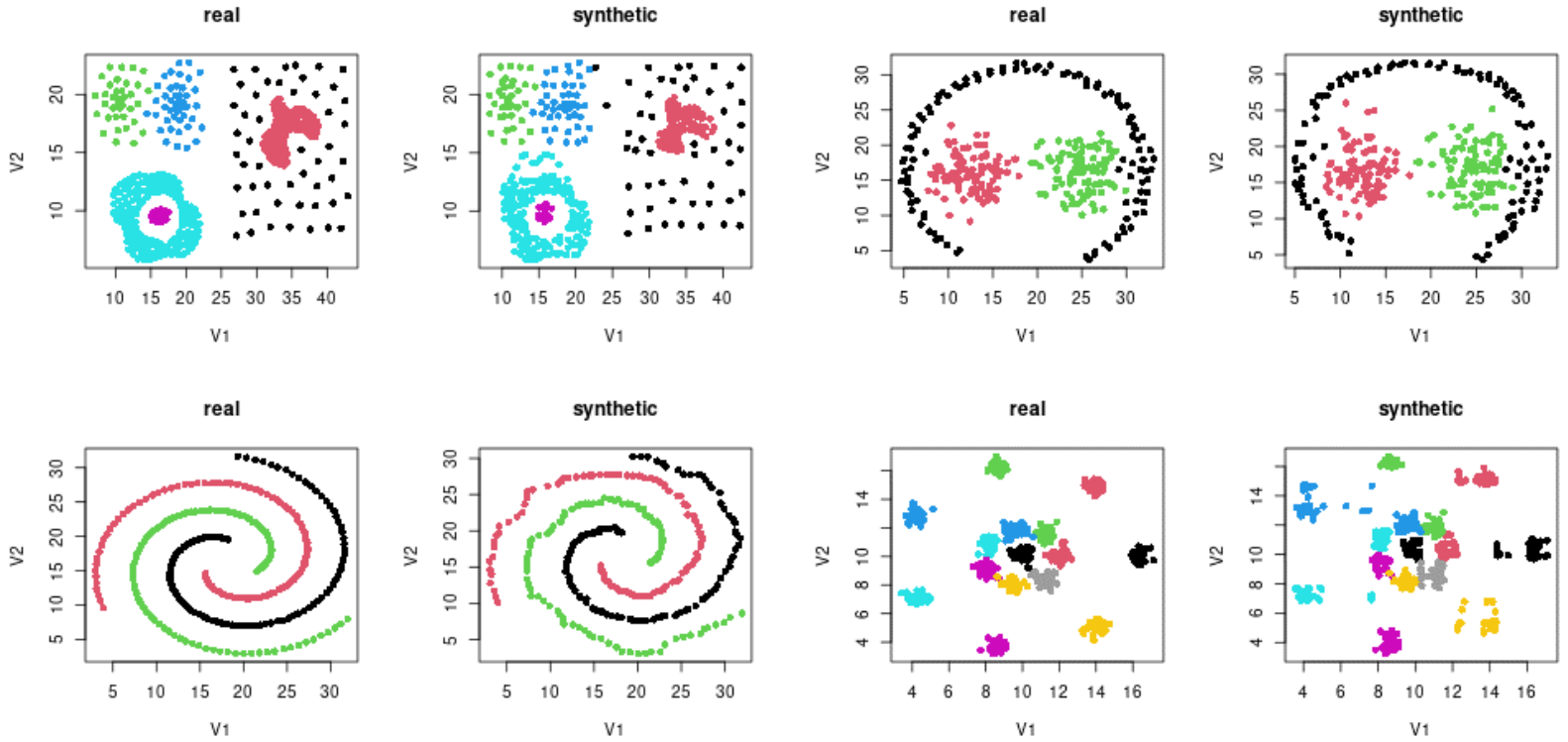
For each variable:

1. Sample n values from empirical distribution.
2. Reorder **with a chosen level of randomisation** so that the rank of the sample matches the rank of the original n values **to a chosen level.**

$$\#values\ to\ randomly\ permute = n(1-\rho)$$

Rank correlation between original and synthetic data equals a chosen value.

Synthesizer reproduces complex relations

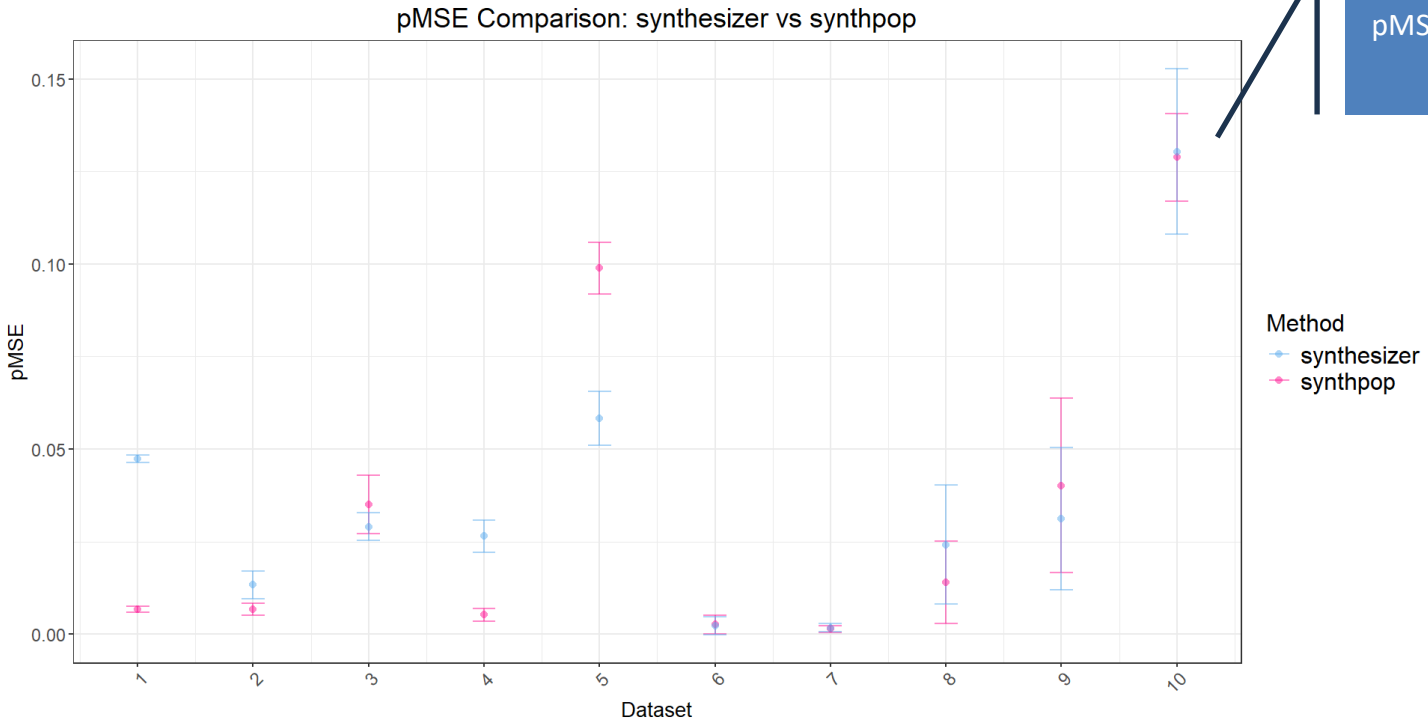


Data: P. Fänti and S. Sieranoja. *K-means properties on six clustering benchmark datasets* Applied Intelligence, 48 (12), 4743-4759, December 2018 https://doi.org/10.1007/s10489-018-1238-7

Synthesizer works in many cases

- ☑ Univariate and multivariate data
- ☑ Numeric, integer, categorical or mixed data
- ☑ Missing data
- ☑ Mixed distributions (e.g. inflated zeros)
- ☑ Univariate and multivariate time series
- ☒ Logical restrictions / structural zero’s
- ☒ Relations between records

Experiments on SBS data



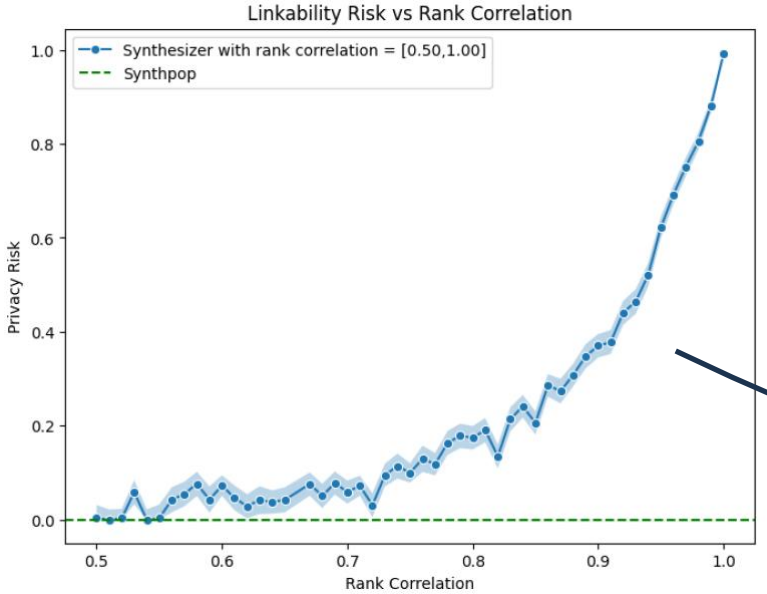
pMSE: lower is better

Image credit: M. Jacobs (2025) *Testing the Performance of synthesizer*. MSc thesis, Leiden University

Synthesizer has controllable SDC

Dataset	Risk	95% Confidence Interval
synthesizer ($rankcor = 1$)	0.23894	(0.21212, 0.26577)
synthesizer ($rankcor = 0.9$)	0.09144	(0.07223, 0.11064)
synthpop	0.16790	(0.14388, 0.19192)

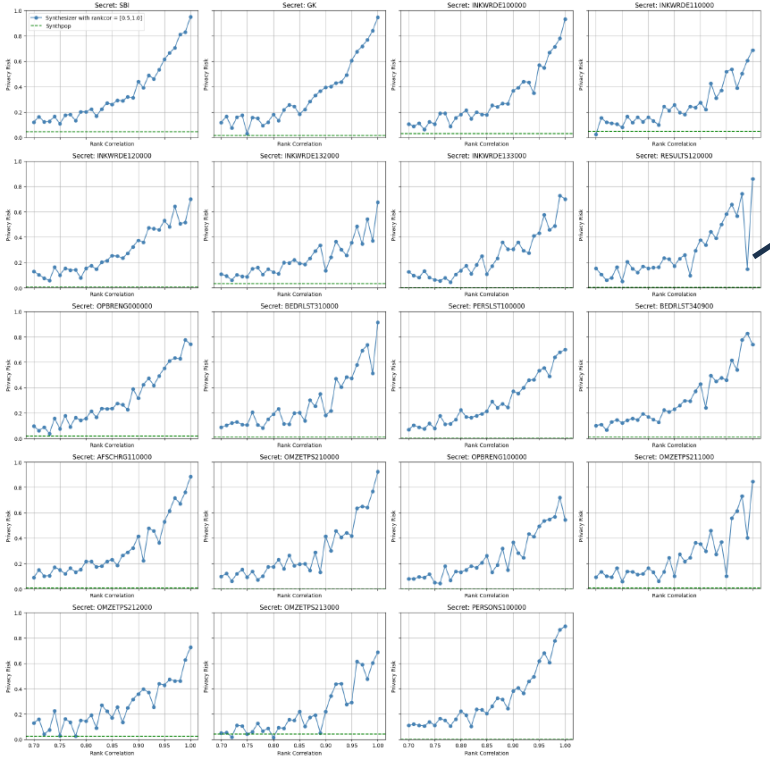
Singling-out risk
Probability that, given a unique pattern in synthetic data, the pattern can be found in the original data



Linkability risk
Probability that records from two data sets without overlapping variables can be linked through synthetic data that overlaps with both.

Image credit: M. Anthoulaki (2025) *Evaluating the Privacy Riscs of data Generated by the Synthesizer Method*. MSc thesis, Leiden University

Synthesizer has controllable SDC



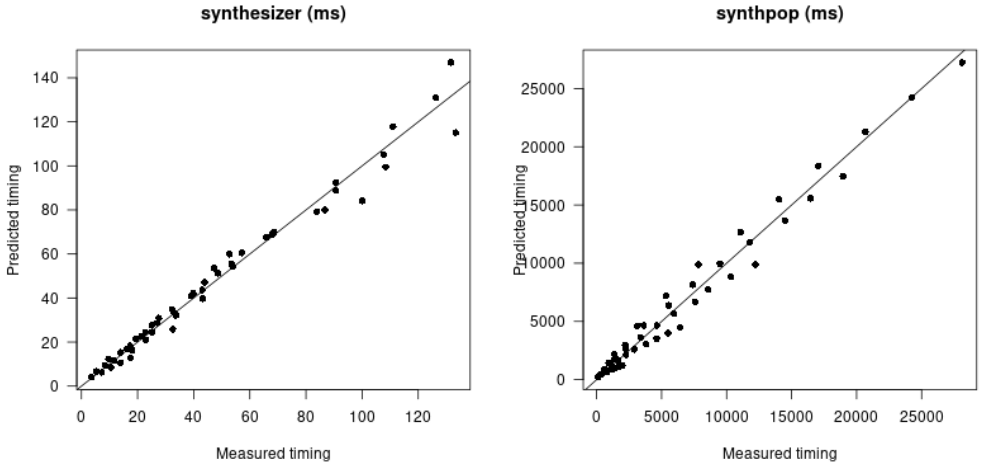
Inference risk:
Risk of disclosure by prediction based on models trained on synthetic data

Riscs measured using the Anonymeter framework in Python[1].

[1] Giomi, Matteo, et al. *A unified framework for quantifying privacy risk in synthetic data* (2022). Proceedings of Privacy Enhancing Technologies Symposium.
Image credit: M. Anthoulaki (2025) *Evaluating the Privacy Riscs of data Generated by the Synthesizer Method*. MSc thesis, Leiden University

Synthesizer is fast

Method	Time complexity (theory)
synthesizer	O($pn\log(n)$)
synthpop (using CART)	O($p^2n\log(n)$)



Experiment:
- 1k-10k records and 5-25 variables (5x)
- Linear model for median timings.
- Adjusted R^2=0.98 (both models)

Synthesizer is grounded in theory

For each variable:

1. Sample *n* values from empirical distribution.
2. Reorder so that the rank of the sample matches the rank of the original *n* values.

$$\sup |\hat{F}_n^* - F| \to 0, a.s. \ (n \to \infty)$$

original distribution

Synthetic distribution

(Proof based on multivariate generalizations of the Glivenco-Cantelli result in terms of copulas)

Summary

- synthesizer offers an easy-to-understand synthetic data method where *the data is the model*.
- The method is fast, retains high utility, and has customizable privacy—utility tradeoff.
- Because of it's simplicity we can formally establish some asymptotic properties

www.markvanderloo.eu

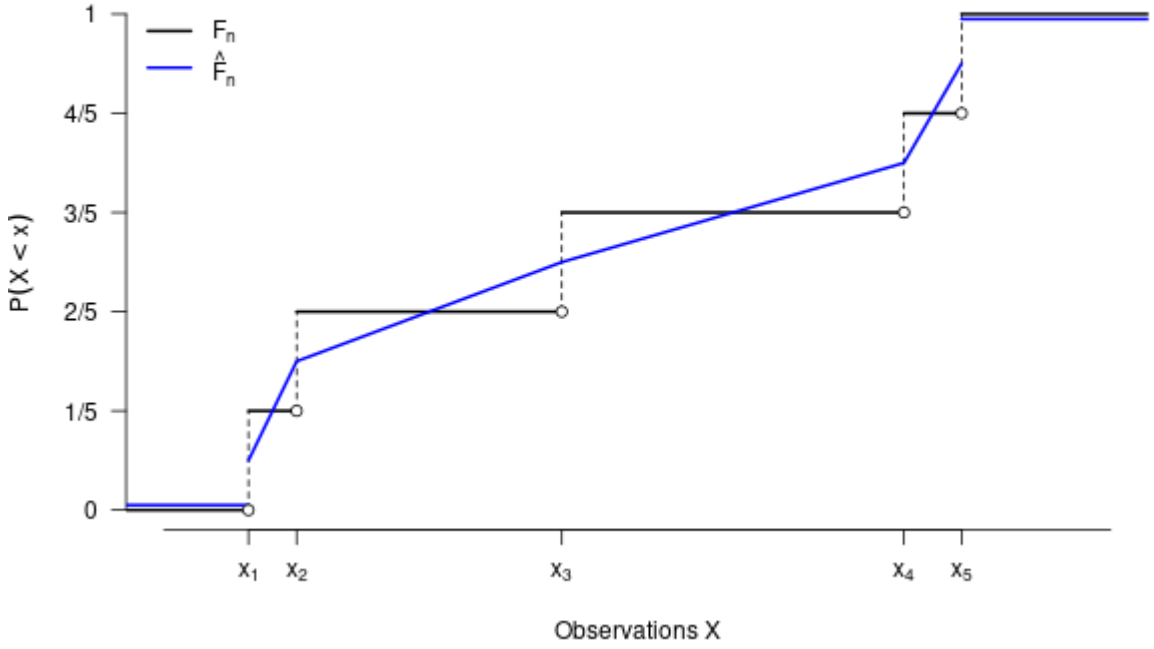
Glad you asked!

Algorithm 1: Synthesize a dataset

Input : An $n \times p$ dataset $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_p\}$
Output: An $n \times p$ synthetic dataset $\mathbf{X}^*$

$\mathbf{X}^* = \{\}$;
for $i \in \{1, 2, \ldots p\}$ **do**
  Create an approximate CDF $\hat{F}_{n,j}$;
  $\mathbf{x}_j^* \leftarrow [x_1, x_2, \ldots, x_n] \sim \hat{F}_{n,j}$; // sample n values from $\hat{F}_{n,j}$
  $\mathbf{x}_j^* \leftarrow \mathsf{sort}(\mathbf{x}_j^*)[\mathsf{rank}(\mathbf{x}_j)]$; // match order of $\mathbf{x}_j$
  $\mathbf{X}^* = \mathbf{X}^* \cup \{\mathbf{x}_j^*\}$;
end

Wow, I did not expect that question!



I totally did not expect this question

$$\#values\ to\ randomly\ permute = n(1-\rho)$$

In expectation

Desired rank correlation