

SyntheRela - Synthetic Relational Data Generation Benchmark



Installation

To install only the benchmark package, run the following command:

```
pip install .
```

Replicating the paper's results

We divide the reproducibility of the experiments into two parts: the generation of synthetic data and the evaluation of the generated data. The following sections describe how to reproduce the experiments for each part.

To reproduce some of the figures the synthetic data needs to be downloaded first. The tables can be reproduced with the results provided in the repository or by re-running the benchmark.

First, create a `.env` file in the root of the project with the path to the root of the project. Copy `.env.example`, rename it to `.env` and update the path.

Download synthetic data and results

The data and results can be downloaded and extracted with the below script, or are available on google drive [here](#).

```
conda activate reproduce_benchmark
chmod +x experiments/reproducibility/download_data_and_results.sh
./experiments/reproducibility/download_data_and_results.sh
```

Evaluation of synthetic data

To run the benchmark and get the results of the metrics, run:

```
conda activate reproduce_benchmark
chmod +x experiments/reproducibility/evaluate_relational.sh
./experiments/reproducibility/evaluate_relational.sh

chmod +x experiments/reproducibility/evaluate_tabular.sh
./experiments/reproducibility/evaluate_tabular.sh

chmod +x experiments/reproducibility/evaluate_utility.sh
./experiments/reproducibility/evaluate_utility.sh
```

Generation of synthetic data

Depending on the synthetic data generation method a separate python environment is needed. The instruction for installing the required environment for each method is provided in docs/INSTALLATION.md.

After installing the required environment, the synthetic data can be generated by running the following commands:

```
conda activate reproduce_benchmark
chmod +x ./experiments/reproducibility/generation/generate_sdv.sh
./experiments/reproducibility/generation/generate_sdv.sh

conda activate rctgan
chmod +x ./experiments/reproducibility/generation/generate_rctgan.sh
./experiments/reproducibility/generation/generate_rctgan.sh

conda activate realtabformer
chmod +x ./experiments/reproducibility/generation/generate_realtabformer.sh
./experiments/reproducibility/generation/generate_realtabformer.sh

conda activate tabular
chmod +x ./experiments/reproducibility/generation/generate_tabular.sh
./experiments/reproducibility/generation/generate_tabular.sh

conda activate gretel
python experiments/generation/gretel/generate_gretel.py --connection-uid
python experiments/generation/gretel/generate_gretel.py --connection-uid
```

```
cd experiments/generation/clavaddpm
chmod +x generate_clavaddpm.sh
./generate_clavaddpm.sh <dataset-name> <real-data-path> <synthetic-data-path>
```

To generate data with MOSTLYAI, instructions are provided in `experiments/generation/mostlyai/README.md`.

Further instructions for GRETELAI are provided in `experiments/generation/gretel/README.md`.

Visualising Results

To visualize results, after running the benchmark you can run the below script. The figures will be saved to `results/figures/` :

```
conda activate reproduce_benchmark
chmod +x ./experiments/reproducibility/generate_figures.sh
./experiments/reproducibility/generate_figures.sh
```

Reproducing Tables

To reproduce the tables you can run the below script. The tables will be saved as `.tex` files in `results/tables/` :

```
conda activate reproduce_benchmark
chmod +x ./experiments/reproducibility/generate_tables.sh
./experiments/reproducibility/generate_tables.sh
```

Adding a new metric

The documentation for adding a new metric can be found in `docs/ADDING_A_METRIC.md`.

Synthetic Data Methods

Open Source Methods

- SDV: The Synthetic Data Vault
- RCTGAN: Row Conditional-TGAN for Generating Synthetic Relational Databases
- REaLTabFormer: Generating Realistic Relational and Tabular Data using Transformers
- ClavaDDPM: Multi-relational Data Synthesis with Cluster-guided Diffusion Models
- IRG: Generating Synthetic Relational Databases using GANs

- Generating Realistic Synthetic Relational Data through Graph Variational Autoencoders*
- Generative Modeling of Complex Data*
- BayesM2M & NeuralM2M: Synthetic Data Generation of Many-to-Many Datasets via Random Graph Generation*

* Denotes the method does not have a public implementation available.

Commercial Providers

A list of commercial synthetic relational data providers is available in [docs/SYNTHETIC_DATA_TOOLS.md](#).

Conflicts of Interest

The authors declare no conflict of interest and are not associated with any of the evaluated commercial synthetic data providers.