MetaSynth

A synthetic data method

Raoul Schram

August 31, 2022

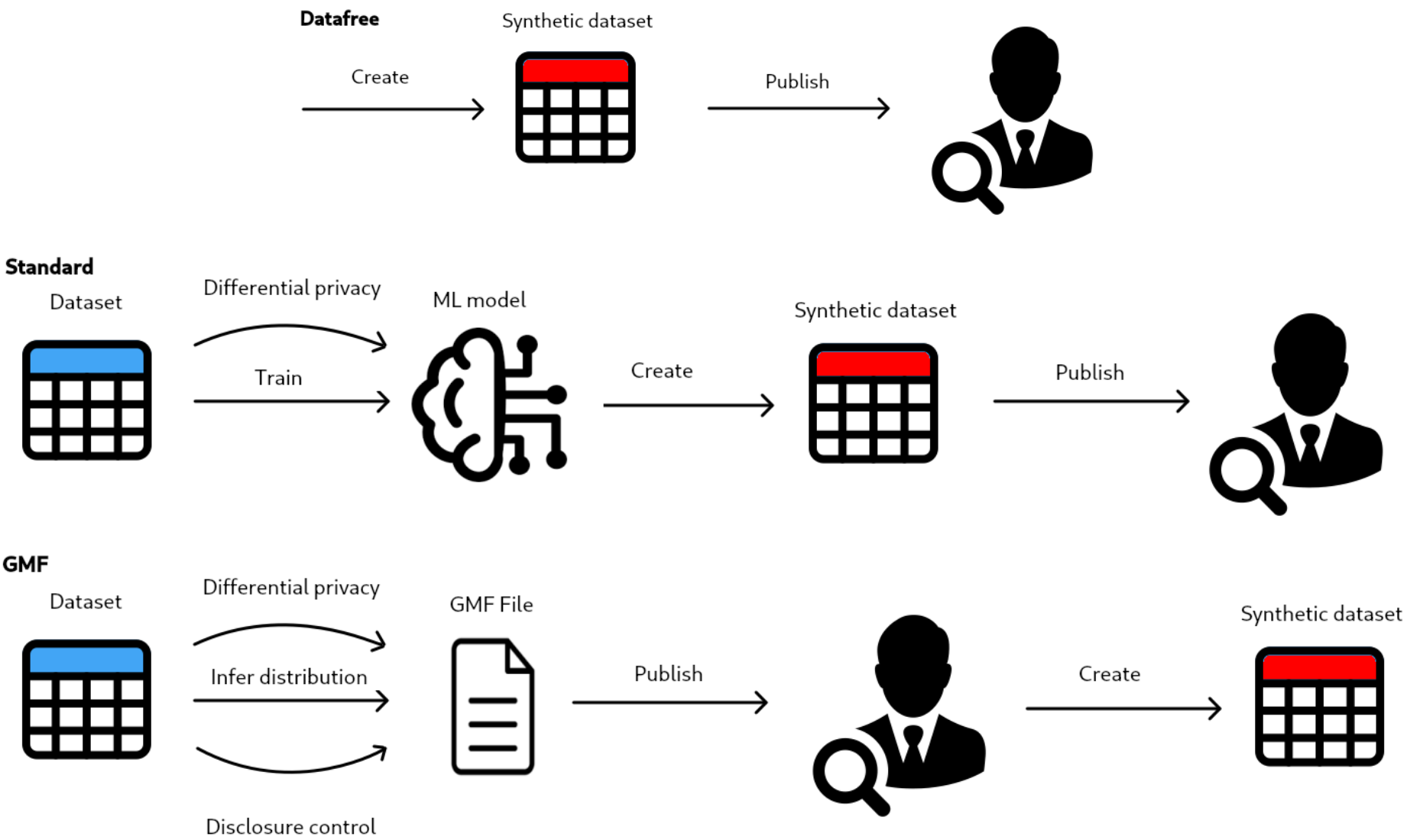
Utrecht University

MetaSynth

- Metasynth project started in April 2022
- Erik-Jan van Kesteren: Assistant professor in the methodology & statistics department
- Raoul Schram: Research Engineer at the IT department at the UU
- Find a way to help researchers test their code on private data
  - CBS datasets
  - Preserve some statistical information
- Create a standardized file format

Utrecht University

Synthetic data pipeline comparison



Generative Metadata File

- Advantages:
  - GMF file human readable.
  - Level of privacy is high
  - Privacy issues can be manually or automatically fixed.
  - GMF file is standardized, with a JSON schema.
- Disadvantages:
  - Utility is low.
  - No relationships between columns.

Utrecht University

GMF structure


```
{
    "n_rows": 891,
    "n_columns": 11,
    "provenance": {
        "created by": {
            "name": "MetaSynth",
            "version": "0.1.0+1.ga7cddcb.dirty",
            "privacy": null
        },
        "creation time": "2022-08-25T12:37:10.347845"
    },
    "vars": [
        {
            "name": "PassengerId",
            "type": "discrete",
            "dtype": "int64",
            "prop_missing": 0.0,
            "distribution": {
                "name": "DiscreteUniformDistribution",
                "parameters": {
                    "low": 1,
                    "high": 892
                }
            }
        },
```


Utrecht University

GMF data types and distributions

- string
  - RegexDistribution, UniqueRegexDistribution, FakerDistribution
- categorical
  - MultinomialDistribution
- float
  - UniformDistribution, NormalDistribution, TruncatedNormalDistribution, ExponentialDistribution, LogNormalDistribution
- int
  - DiscreteUniformDistribution, PoissonDistribution, UniqueKeyDistribution
- date, time, datetime
  - UniformDateDistribution, UniformTimeDistribution, UniformDateTimeDistribution

Utrecht University

MetaSynth

- GitHub
- Python >= 3.7
- Start from pandas `DataFrame`
- Implementation of the GMF standard.
  - Inference of distributions.
  - Creation of synthetic dataset.
- Extensible:
  - New distributions.
  - Override distributions (disclosure control, differential privacy).
- Documentation on readthedocs
- Automated tests
- Package on PyPi
  - `pip install metasynth`

Utrecht University

Outlook

- Implement privacy extensions (differential privacy, disclosure control).
- Implement more distributions (as add-ons).
- Get user feedback.

Utrecht University

Workshop

- Online tutorial:
  - Go to `https://github.com/sodascience/metasynth`
  - Click on "launch binder" badge
- Use your own data:
  - On binder if data not so privacy sensitive.
  - Install MetaSynth locally if privacy sensitive (`pip install metasynth`)
- Use freely available data
  - Download data from `https://data.fivethirtyeight.com`
  - Use binder tutorial to process it.

Utrecht University