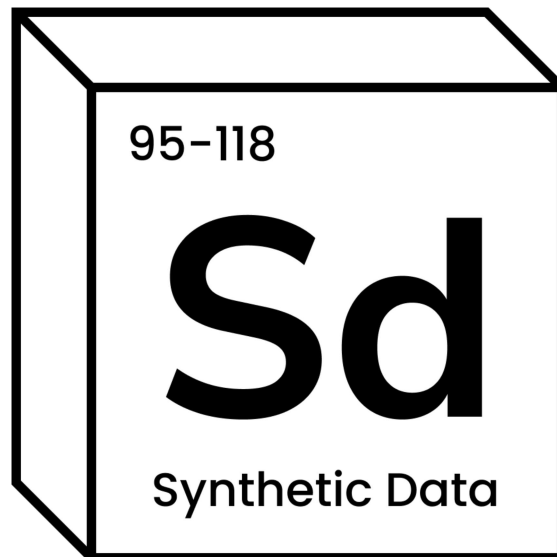


## Overview





## Overview

`ydata-synthetic` is the go-to Python package for **synthetic data generation for tabular and time-series data**. It uses the latest Generative AI models to learn the properties of real data and create realistic synthetic data. This project was created to educate the community about synthetic data and its applications in real-world domains, such as data augmentation, bias mitigation, data sharing, and privacy engineering. To learn more about Synthetic Data and its applications, [check this article](#).

## Current Functionality

- 🤖 **Create Realistic Synthetic Data using Generative AI Models:** `ydata-synthetic` supports the state-of-the-art generative adversarial networks for data generation, namely Vanilla GAN, CGAN, WGAN, WGAN-GP, DRAGAN, Cramer GAN, CWGAN-GP, CTGAN, and TimeGAN. Learn more about the use of [GANs for Synthetic Data generation](#).
- 🔴 **Synthetic Data Generation for Tabular and Time-Series Data:** The package supports the synthesization of tabular and time-series data, covering a wide range of real-world

applications. Learn how to leverage `ydata-synthetic` for [tabular](#) and [time-series](#) data.

-  **Best Generation Experience in Open Source:** Including a guided UI experience for the generation of synthetic data, from reading the data to visualization of synthetic data. All served by a slick Streamlit app.  Here's a [quick overview](#) – ⌚ 1min

#### ? Question

##### Looking for an end-to-end solution to Synthetic Data Generation?

[YData Fabric](#) enables the generation of high-quality datasets within a full UI experience, from data preparation to synthetic data generation and evaluation. Check out the [Community Version](#).

## Supported Data Types

### Tabular Data

**Tabular data** does not have a temporal dependence, and can be structured and organized in a table-like format, where **features are represented in columns**, whereas **observations correspond to the rows**.

Additionally, tabular data usually comprises both *numeric* and *categorical* features. **Numeric** features are those that encode **quantitative** values, whereas **categorical** represent **qualitative** measurements. Categorical features can further divided in *ordinal*, *binary* or *boolean*, and *nominal* features.

Learn more about synthesizing tabular data in this [article](#), or check the [quickstart guide](#) to get started with the synthesization of tabular datasets.

### Time-Series Data

**Time-series data** exhibit a sequential, **temporal dependency** between records, and may present a wide range of patterns and trends, including **seasonality** (patterns that repeat at calendar periods – days, weeks, months – such as holiday sales, for instance) or **periodicity** (patterns that repeat over time).

Read more about generating time-series data in this [article](#) and check this [quickstart guide](#) to get started with time-series data synthesization.

## Supported Generative AI Models

The following architectures are currently supported:

- [GAN](#)
- [CGAN](#) (Conditional GAN)
- [WGAN](#) (Wasserstein GAN)
- [WGAN-GP](#) (Wasserstein GAN with Gradient Penalty)
- [DRAGAN](#) (Deep Regret Analytic GAN)
- [Cramer GAN](#) (Cramer Distance Solution to Biased Wasserstein Gradients)
- [CWGAN-GP](#) (Conditional Wasserstein GAN with Gradient Penalty)
- [CTGAN](#) (Conditional Tabular GAN)
- [TimeGAN](#) (specifically for *time-series* data)
- [DoppelGANger](#) (specifically for *time-series* data)