Synthetic data for digital twins – the why and how

Lucy Mosquera
Senior Director, Data Science & Operations

Typical process for estimating treatment effects

For individual *i* the effect of a treatment on an outcome is would be:

$$e_i = [Potential\ Outcome \mid Treatment] - [Potential\ Outcome \mid Control]$$

In the real world, only one potential outcome is typically observable. Instead RCT will estimate the Average Treatment Effect (ATE) across groups of individuals:

$$ATE = [Expected\ Outcome \mid Treatment] - [Expected\ Outcome \mid Control]$$

ATE is a good estimate of the effect of a treatment when:

- receipt of the treatment and control is random
- participants in the trial are randomly selected from the population **
- participants in the trial truly receive the treatment they are assigned to **

Why digital twins?

Amy, 42F
High BP

Joins trial

Amy randomized to treatment

Amy, 42F
Normal BP

Synthetic Amy randomized to control

Synthetic
Amy, 42F
High BP

With a digital twin, you can recover the missing potential outcome

If Amy participates in a trial and is randomized to receive the treatment, her digital twin can model what would have happened had she received the control

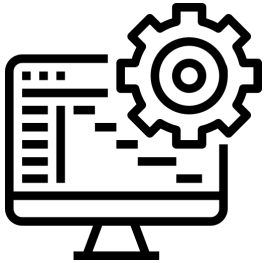
How do we achieve this magic?

Digital twins require:

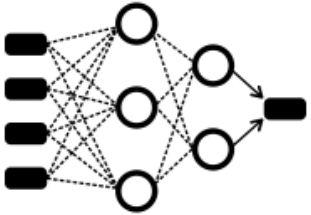
- Excellent synthetic data generation (SDG) methods
- SDG models need to be trained on reliable data
    - Digital twins often used for control arm allocation due to greater data availability although they would need to be updated as standard of care changes
- Robust validation to show that the SDG model is generalizable and able to produce realistic outcomes for a range of clinical trial patients
- Thoughtful clinical trial design to alter randomization rates (e.g., 2:1 randomization to treatment vs control) and alter statistical analyses methods

Synthetic data generation for digital twins can use a range of methods

Statistical or mathematical models



Machine learning



Deep learning

Statistical or mathematical models

Aims to produce a statistical or mathematical model of the dataset or the underlying process.

Can be produced with or without an example of the data you are aiming to synthesize and may incorporate subject area knowledge.

Advantages	Disadvantages
• High degree of explain-ability • Can synthesize data with or without access to a real dataset for training • Computationally efficient to generate large amounts of data • May be combined with subject area knowledge	• May be based on incorrect assumptions or models • Difficult to parameterize correctly • To represent more complex relationships and patterns the user must design a more complex model

Examples:

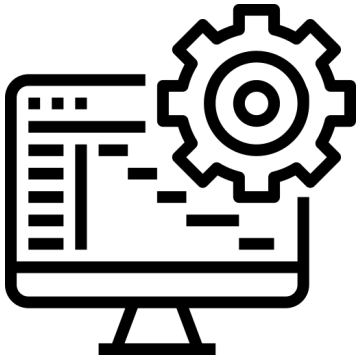
- Gaussian process models
- Monte-Carlo simulations
- Sampling from a probability distribution
- Kernel density smoothing

Machine learning

Models that aim to learn patterns in datasets and then leverage those learned patterns to produce new synthetic observations.

ML models make fewer assumptions than mathematical or statistical models.

Advantages	Disadvantages
• Synthetic data captures a wide range of patterns and relationships • Data driven modelling approach with fewer assumptions • Easy to automate and scale to large datasets	• Requires representative input data to train • More computationally intensive • Lower explain-ability • Input data may require pre-processing before model training



Examples:

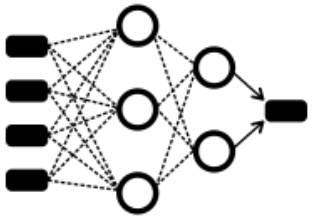
- Decision tree models (random forests, gradient boosted decision trees)
- Clustering based synthesis models
- Naïve Bayes

Deep learning

Neural network models that dynamically and iteratively learn from large input datasets.

Compared to ML models, deep learning models don't just learn the patterns, they learn how to learn the patterns in your data.

Advantages	Disadvantages
• Able to learn and synthesize very complex relationships • Works with wide range of data types (tabular, text, images, audio, video) • Data driven modelling approach with fewer assumptions	• Requires very large amounts of representative data to train • May be prone to overfitting • Computationally demanding to train and generate data • Lower explain-ability • Input data may require extensive pre-processing before model training



Examples:

- Large Language Models like ChatGPT
- Generative Adversarial Networks (GANs)
- Transformers
- Variational Autoencoders (VAEs)

How can we validate synthetic data?

The assessment strategies will depend on the data generation process (i.e., type of model, whether real data is available), intended use case (is it known or unknown?), and the nature of the data

Utility	Privacy	Generalizability
Can the synthetic data be used to draw the same conclusions as real data would? • Quantitative comparisons between real and synthetic data • Benchmarking against population statistics • Replicating analyses on real vs synthetic data • Quantify bias	*Can identities of real individuals be recovered from the synthetic data?* • Assessments of membership disclosure & identity disclosure • Assess how outliers in the data have been synthesized • Ensure compliance with privacy regulations in how data is handled, shared, etc.	*How well representative are the patients generated of the population at large?* • Assess performance of synthetic data in new contexts • Compare performance at both individual patient level generation and population level inference

Thank you!