Applied Computing and Intelligence

https://www.aimspress.com/journal/aci

Applied Computing and Intelligence
5(2): 191–212.
DOI: 10.3934/aci.2025012
Received: 18 June 2025
Revised: 28 August 2025
Accepted: 30 August 2025
Published: 08 September 2025

Research article

A variational autoencoder and neural network approach to generating synthetic data in well-being research

Joonas Tuomikoski[1,*], Ville Vesterinen[2], Rami Luisto[1], Ilkka Pölönen[1] and Sami Äyrämö[1,3]

[1] Faculty of Information Technology, University of Jyväskylä, Finland

[2] Finnish Institute of High Performance Sport KIHU

[3] Wellbeing Services County of Central Finland, Jyväskylä, Finland

* **Correspondence:** Email: joonas.a.s.tuomikoski@jyu.fi.

Academic Editor: Pasi Fränti

Abstract: Although the field of machine learning in health and well-being has experienced significant growth in recent years, the use of sensitive data in these applications is often restricted. While sports and well-being data may appear less sensitive, they are frequently subject to the same standards as health data. Synthetic data generation has emerged as a potential solution to privacy issues, aiming to replicate the properties of real data without disclosing identifiable information. In this study, we generated synthetic recreational runner data using a combined variational autoencoder and neural network model based on baseline measurements and training responses from a three-month training period. We then evaluated the synthetic data by training predictive models and comparing their performance to models trained on real data, with additional metrics measuring statistical similarity and privacy. While some challenges remain, particularly regarding the modeling of rare cases, handling of missing data, and ensuring privacy, our results demonstrate that synthetic data could be used to train predictive models with performance comparable to that of models trained on real data.

Keywords: synthetic data; variational autoencoder; neural network; well-being; machine learning; privacy

1. Introduction

The field of machine learning in health and well-being has witnessed substantial growth in the last few years. It holds great promise in supporting clinical diagnoses (e.g., [1, 2]) and facilitating research (e.g., [3]). However, the effective utilization of machine learning in these fields is often constrained by the sensitive nature of the data, which is subject to regulatory frameworks such as the European General Data Protection Regulation (GDPR) [4]. While these are necessary to protect individuals'

privacy, they impede research and development of new machine learning methods which could yield substantial societal benefits.

Although exercise and well-being data may appear less sensitive than personal health data, certain variables, such as maximal oxygen uptake, are categorized as health information, subjecting entire datasets to the same standards as health data. This means that generally all sports and well-being data need to be processed according to the same standards and laws as personal health data, and the conventions of health data should be adopted in the field of sports and well-being. This also means that methods developed on sports and well-being data should be applicable to health data, making the field intriguing for research and development. As the datasets of studies with laboratory tests are often limited in scope, investigating possible synthetic data generation methods specifically for data similar to those used in this study is essential.

Synthetic data has emerged as a promising solution to the issues with privacy and data sharing [5]. The primary objective of synthetic data is to produce datasets that closely replicate the properties of real data while minimizing the risk of disclosing identifiable information. Unlike traditional anonymization techniques, synthetic data preserves the structural and relational characteristics of the original dataset, thereby enabling more effective model development and evaluation. This would mean safer distribution of synthetic data for research and development, accelerating research and innovation, and allowing original datasets to be reserved for validation and hypothesis testing. In an ideal scenario, synthetic data could be shared completely freely. Synthetic data have already been used in healthcare in various use cases [6–8].

While this description makes synthetic data sound like a perfect solution to safe data transfer, it is not without its limits. By definition, an imitation of a dataset cannot contain more information than its real counterpart, and some degree of information loss is inevitable [9]. Different approaches have been proposed to estimate to which extent the synthetic data retain the information of the original data [10, 11]. These are typically categorized into two dimensions: *resemblance* (sometimes called *fidelity*), which quantifies the similarity of the distributions of the original and synthetic data, and *utility*, which measures the effectiveness of synthetic data when used in practice [12, 13].

The third commonly used evaluation dimension is *privacy*. Recent studies have demonstrated that adversarial techniques can, in some cases, recover information about individuals in the original dataset [14–16]. The privacy goals of synthetic data are often in conflict with the resemblance and utility of the data, and different methods for synthetic data generation need to balance this trade-off. Privacy assessments commonly employ simulated attacks and distance-based metrics to quantify the risk of re-identification [14, 17, 18].

Synthetic data generation methods can be broadly categorized into statistical approaches, such as Bayesian networks [19, 20] and kernel density estimation [21], and neural network-based methods. Among the latter, the most common are based on generative adversarial networks [22] and variational autoencoders [23]. Most existing studies utilize large, publicly available datasets [13], such as UCI repositories [24] or MIMIC-III [25].

In this study, we aim to generate synthetic data for recreational runners and test the feasibility of synthetic data in the context of well-being research, where datasets are commonly small and incomplete. We utilize a composite dataset of recreational runners derived from four different studies to emulate a real-world scenario with scarce and imperfect data. We propose a model that combines a variational autoencoder and a fully connected neural network to produce synthetic runner profiles with

plausible baselines and training responses. Finally, we evaluate the synthetic data, focusing on the aspect of utility but also considering the fidelity and privacy of the synthetic data. The overarching goal is to establish a practical methodology for synthetic data generation and assessment in well-being research.

2. Materials and methods

To investigate the possibilities of synthetic data in the field of well-being, a methodology was used that would correspond to a real-world situation: a synthetic dataset, which should preserve the features of the original data and thus be usable as a substitute, was generated. Since it is not realistic to model the structure of the sparse longitudinal dataset at hand, we limited the modeling aim to developing and validating variational autoencoder and neural network models for generating realistic baseline measurements and the training responses to the three-month training period, thus dividing the generation process into two parts. Both of these parts are described in detail in Sections 2.2 and 2.3. Finally, the synthetic data was evaluated in Section 2.4 by comparing the performance of models trained on synthetic and real data to measure utility, conducting statistical tests to assess resemblance, and using nearest neighbor distance ratios and membership inference attacks to estimate privacy. The overall structure of the methodology is illustrated in Figure 1.

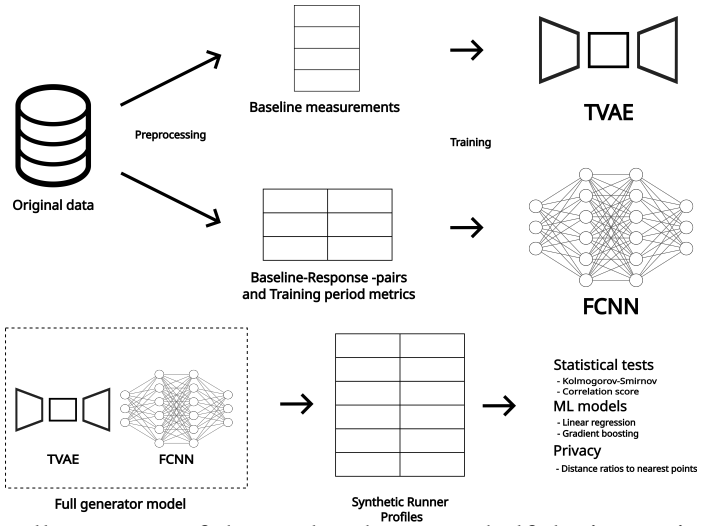

Figure 1. The overall structure of the study: the upper half depicts training the model, and the lower half the synthetic data generation and testing.

2.1. Data description

The dataset utilized in this study was compiled from four longitudinal endurance training projects conducted by the Finnish Institute of High Performance Sport KIHU between 2008 and 2014. Each project implemented comparable training interventions and laboratory testing protocols, facilitating the integration of the datasets for subsequent analysis.

These datasets included three to four laboratory tests per participant, spaced by approximately three-month training periods. Each consisted of demographic characteristics (e.g., gender, age), training volume and intensity distributions (% at low, moderate, and high intensity training zones), physiological load and recovery measures (e.g., testosterone, cortisol, HRV, night HR), and physical performance metrics (e.g., VO2max, HRmax). Importantly, these tests followed the same protocols and were conducted by the same personnel using the same laboratory equipment, which allowed the

merging of the datasets and further restructuring described in Sections 2.2 and 2.3. The combining of the datasets was justified by the laborious and expensive nature of laboratory tests, which makes it impractical to collect large datasets in a single study.

Following the integration of the four datasets, the final sample comprised 122 subjects (71 male, 51 female), with a mean age of 34.4 ± 7.1 years, mean height of 173.1 ± 8.2 cm, mean baseline body mass of 71.0 ± 10.9 kg, mean training history of 9.0 ± 7.7 years, and mean baseline VO2max of 3.5 ± 0.7 l/min. A detailed summary of all variables included in the analysis is provided in Table 1.

Table 1. Variable list: the vertical lines divide the variables into subject characteristics, test result variables, and training period variables.

	mean	std	min	max	missing %
Strength group	2.7	1.1	1.0	4.0	49.2
Endurance group	3.9	2.0	1.0	7.0	0.0
Age	34.4	7.1	20.0	49.0	0.8
Training years	9.0	7.7	1.0	30.0	0.0
Training times (1/wk) prev. 2mo	4.82	1.43	1.0	10.50	0.0
Running (km/wk) prev. 2mo	29.0	17.1	5.0	80.0	2.5
Height (cm)	173.1	8.2	152.5	188.0	0.0
Weight (kg)	70.9	10.7	47.8	99.3	13.9
Fat %	19.3	6.3	5.2	38.0	13.9
Cortisol (nmol/l)	524.0	153.3	233.0	1142.0	19.7
Testosterone (nmol/l)	10.5	8.6	0.0	31.2	19.7
1 RM (kg)	165.5	35.7	92.5	250.0	55.1
CMJ (cm)	27.2	6.4	13.4	44.6	21.1
VO2max (l/min)	3.570	0.702	1.734	5.259	16.4
Vmax (km/h)	15.0	1.7	10.6	19.0	14.5
HR max (bpm)	186.6	9.8	162.0	213.0	14.5
Lamax (mmol/l)	10.5	2.2	3.9	18.3	14.5
VAnT (km/h)	12.4	1.4	8.0	15.9	14.3
VAerT (km/h)	10.0	1.3	7.0	13.1	14.8
MART Vmax (m/s)	6.16	0.70	4.72	8.18	57.2
MART Lamax (mM)	13.80	2.52	6.30	21.13	57.2
R economy (ml/kg/km)	210.8	13.1	177.9	247.1	48.2
Night HR (bpm)	52.73	6.15	39.10	76.52	32.6
SDRR (ms)	123.4	28.6	72.6	214.6	64.5
RMSSD (ms)	72.9	32.4	23.6	185.8	64.5
Abs. relaxation HRV (ms)	93.8	10.5	62.1	117.5	49.4
Abs. stress HRV (ms)	78.3	26.3	36.8	202.5	49.4
VLF ($\ln(ms^2)$)	5.39	0.57	4.06	6.99	65.2
LF ($\ln(ms^2)$)	8.05	0.71	3.79	9.59	32.6
HF ($\ln(ms^2)$)	7.83	0.87	5.06	10.16	32.6
HF2 ($\ln(ms^2)$)	8.11	1.00	5.27	10.34	65.2
TP ($\ln(ms^2)$)	8.86	0.74	5.80	10.50	49.4
Training period (d)	61.5	14.8	23.0	104.0	19.1
Training avg (h/wk)	5.96	1.92	1.18	14.29	19.1
Training times (1/wk)	5.06	1.56	0.13	14.25	19.1
Running (km/wk)	31.26	16.58	0.00	88.41	18.3
Low intensity %	83.24	11.26	24.65	100.0	19.1
Moderate intensity %	13.74	10.01	0.00	72.62	19.1
High intensity %	3.02	2.95	0.00	19.29	19.1

In the merged dataset there was 31.4% overall missingness. This was mainly due to certain variables having a high number of missing values, as they were not measured in each of the four studies (e.g., 1-rep max), and two of the studies had three measurement points instead of four. This further motivated the further restructuring of data to initial measurements and training responses, as it allowed us to discard the data points with structural missingness, leading to 22.2% missingness. While this still leaves some missingness due to not every variable being measured in every test and thus not being random, it was deemed not to be dependent on the other values and acceptable to be imputed together with values missing at random.

2.2. Baseline measurement generation

To generate synthetic baseline measurements, the merged dataset was restructured such that each individual measurement occasion (three to four per participant) was treated as an independent observation. This allowed us to artificially almost quadruple the number of data points and simultaneously mitigated the impact of missing data, as if a subject was missing a measurement point, it could be excluded without discarding the entire participant's data. Consequently, each participant contributed three or four baseline observations to the training dataset. Rows with more than 50% missing values after restructuring were discarded.

The resulting dataset was divided into training and testing sets using 10-fold cross-validation. Importantly, this split was performed at the subject level, ensuring that all measurement occasions from a given participant were assigned exclusively to either the training or testing set, thereby preventing data leakage. Each fold underwent independent imputation using the machine learning-based model Hyperimpute [26], which imputes the data column-wise, selecting between several commonly used imputation methods for each column. It was chosen as the best-performing method when compared to methods such as k-nearest-neighbor imputation. The training data were used to train a Tabular Variational Autoencoder to generate the synthetic baseline measurements.

The imputed training data were then used to train a Tabular Variational Autoencoder (TVAE) [27], a specialized variant of the Variational Autoencoder (VAE) architecture, tailored for the generation of synthetic tabular data. VAEs, first introduced by Kingma and Welling in 2013 [23], are generative neural network models characterized by their encoder-decoder structure. Unlike traditional autoencoders, VAEs incorporate a stochastic latent space, allowing the model to learn a probabilistic embedding of the input data. The encoder Q maps the input data into a multivariate latent distribution (typically Gaussian), from which the decoder P reconstructs the data. The model is trained by simultaneously minimizing the Kullback-Leibler divergence of the encoder and the reconstruction loss of the decoder, which together form the loss function

$$\mathcal{L} = KL[Q(z|X)|P(z)] - \log(P(X|z)), \tag{1}$$

where $KL[Q(z|X)|P(z)]$ is the Kullback-Leibler divergence and $\log(P(X|z))$ is the reconstruction loss. During data generation the encoder Q is discarded and decoder P is used to generate the data. For example, if multivariate Gaussian $N(0, I * \sigma^2)$ is learned as the latent distribution, synthetic data can be generated by feeding the decoder P samples drawn from $N(0, I * \sigma^2)$.

TVAE extends the standard VAE by incorporating variational Gaussian mixture modeling (VGM) [28] to divide each univariate distribution into multiple Gaussian distributions to ease the problems arising from multimodal and non-Gaussian distributions. The modes found with VGM are

then individually normalized, and the network is trained using the mixture model. When generating new data, it is returned to the original form using the inverse transform of the mixture model. The loss function and layer architecture are otherwise inherited from VAE.

After training on the training data consisting of the restructured baseline measurement, TVAE was used to generate one thousand synthetic baseline samples. In addition to these synthetic baselines, another instance of TVAE was used to generate synthetic training period metrics to connect the baseline and response measurements. These were then used to generate the synthetic training responses in the subsequent stage described in Section 2.3.

2.3. Training response generation

To generate the synthetic training responses, the original dataset was reshaped into a set of baseline-response pairs. The baseline measurements were constructed in an analogous manner to that in the previous section, excluding the final measurement of each participant. The corresponding response measurements were constructed by calculating the difference between the baseline measurement and the following measurement, while the metrics of the training period in between were used to link these two time points. This restructuring allowed each participant to contribute up to three baseline-response pairs to the dataset. Again, rows with more than 50% missing values were discarded, and Hyperimpute was used for the remainder of the missing data by first imputing the baseline measurements, after which the training responses were imputed. The resulting dataset was then partitioned into training and testing sets using 10-fold cross-validation, maintaining the same subject assignments as in the baseline measurement generation to prevent data leakage and ensure comparability.

A fully connected neural network (FCNN) was employed to model and predict the training responses using the baseline measurement and the training period metrics. The FCNN architecture consisted of multiple dense layers with scaled exponential linear unit (SELU) activation functions and incorporated dropout layers for regularization. The network was trained to minimize the mean squared error between predicted and actual training responses. The trained FCNN was then used to predict plausible training responses for the synthetic baseline measurements and the training period metrics generated in the previous step with TVAE. The architecture and parameters of the FCNN are shown in Figure 2.

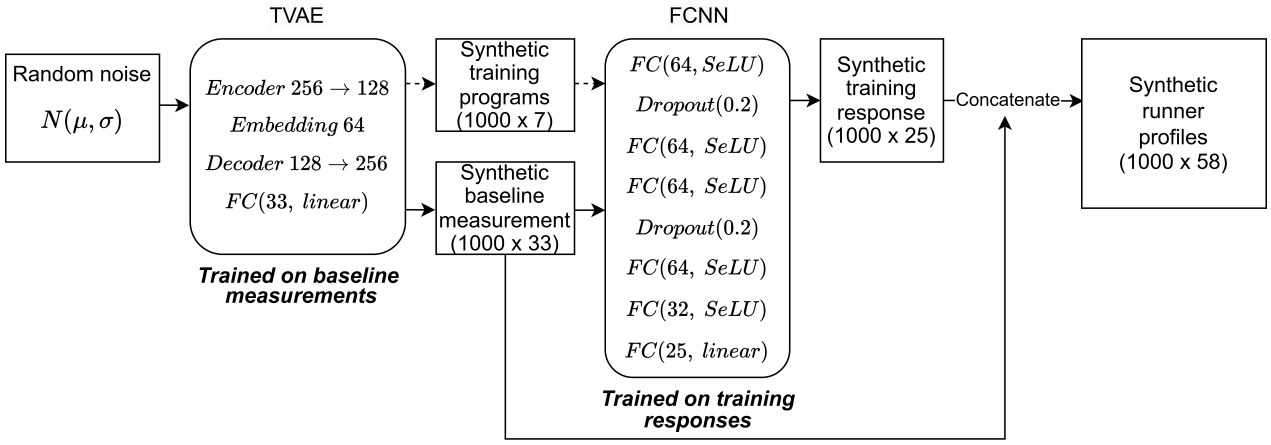

Figure 2. Synthetic data generation model.

2.4. Evaluation

The evaluation of synthetic data in this study was structured around three principal dimensions: resemblance, utility, and privacy [12, 13]. While alternative ways to evaluate synthetic data exist (e.g., [10, 11]), these three were deemed the most applicable to this study. We focused on the utility aspect and evaluated the quality of the synthetic data samples by repeating training and testing of predictive machine learning models on both synthetic and real data. The resemblance was tested by using the Kolmogorov-Smirnov test and pairwise Pearson correlations to evaluate the variable distributions. Finally, the privacy of the synthetic data was evaluated by comparing the distances between real and synthetic data points.

To assess the practical value of the synthetic data, we used a predictive modeling framework in which machine learning models were trained and evaluated on both synthetic and real datasets. Specifically, the commonly used gradient boosting [29] and linear regression [30] models were employed to predict training responses using baseline measurement variables. The aim of this utility testing framework, shown in Figure 3, is to quantify how different the results would be if synthetic data were used instead of the original data. For each response variable, two model instances were trained: one on real data and one on synthetic data. The models were evaluated with the test sets from the 10-fold cross-validation split, and root mean squared errors (RMSE) and mean average errors (MAE) of predictions were used as the primary metrics. Welch's t-test was applied to determine the statistical significance of performance differences between models trained on real and synthetic data. All variables were standardized using z-score scaling prior to model training.

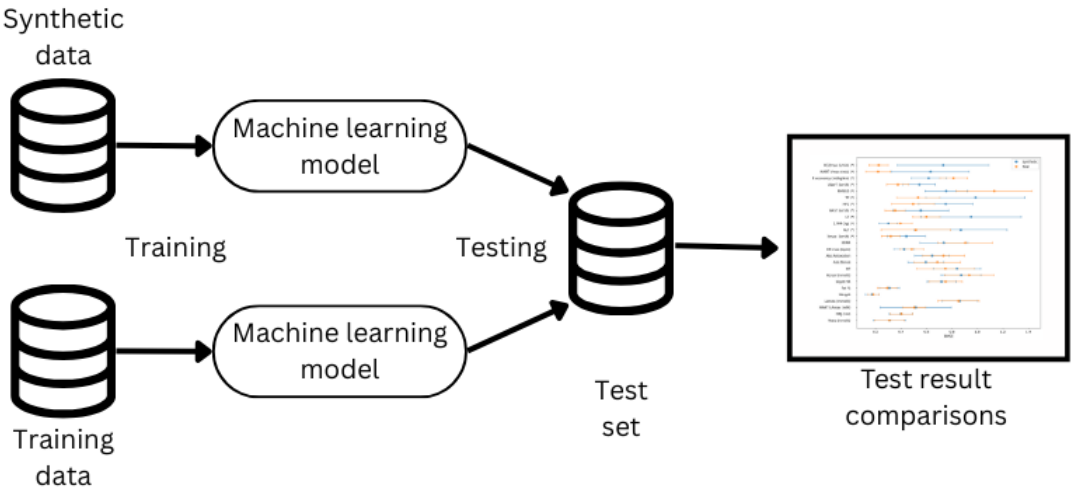

Figure 3. Utility testing framework (TSTR-TRTR).

The resemblance between synthetic and real data was evaluated using both univariate and multivariate statistical measures. The Kolmogorov-Smirnov (KS) test was used to compare univariate similarity. It uses the maximum distance between the cumulative distribution functions of real and synthetic data to measure the similarity of two distributions, and its test statistic is defined as

$$D_{r,s} = \sup_x |F_r(x) - F_s(x)|, \tag{2}$$

where F_r and F_s denote the empirical cumulative distribution functions of the real and synthetic data, respectively.

In addition to the KS test, Wasserstein distance [31] was used as a metric for univariate similarity. The Wasserstein-1 distance for two distributions P and Q over a variable x is defined as

$$W(P, Q) = \inf_{\gamma \in \Gamma(P,Q)} \int |x - y| \, d\gamma(x, y), \tag{3}$$

where $\Gamma(P, Q)$ denotes the set of all joint distributions with marginals P and Q. In practice, the Wasserstein distance was computed between the empirical distributions of each variable in the real and synthetic datasets to provide an additional measure of univariate similarity.

Correlation scores were used to evaluate how well the relations between the variables were preserved in the synthetization process. Using the pairwise Pearson correlation matrices for both real and synthetic datasets to calculate the absolute difference between the matrices yields the correlation score matrix

$$C_{\text{score}} = |\rho(x_{i,\text{synth}}, x_{j,\text{synth}}) - \rho(x_{i,\text{real}}, x_{j,\text{real}})|. \tag{4}$$

Each value in this matrix represents how well the relation between two variables has been preserved in the synthetization with scores ranging from 0 to 2, where lower scores indicate better preservation. The mean of the off-diagonal elements of C_{score} was used to summarize the overall preservation of correlation, and the individual values were shown as a heatmap.

Privacy was assessed by analyzing the nearest neighbor distance ratio (NNDR) between the real and synthetic data. To verify that the synthetic data were not clustered near the real data, for each synthetic point the Euclidean distances to the nearest and the second-nearest real data points were measured and expressed as the distance ratio

$$\text{NNDR} = \frac{d(x_s, x_{r,i})}{d(x_s, x_{r,j})}, \tag{5}$$

where x_s is a synthetic data point, $d(x_s, x_{r,i})$ is the distance to the nearest real data point, and $d(x_s, x_{r,j})$ the distance to the second-nearest. Min-max normalization was applied to all variables prior to distance calculation. The values of the distance ratios range from 0 (the synthetic data point has copied a real data point) to 1 (the synthetic data point is exactly in between two real data points). For reference, the same metric was computed for test data points relative to the training data. The difference between these is referred to as the *privacy loss*, where ideally a value close to 0 is reached.

Membership inference attacks [14] were also deployed as a measure of information leakage. They rely on the fact that models have a smaller loss or error on training data than on independent data. In synthetic data generation, this means that if synthetic data have overfitted to the training data, a different model trained on synthetic data has a smaller error on the training data than on independent test data. The membership inference attack was conducted by assuming that the adversary has acquired 60 records, from which they suspect some have been used to train the data generator, which in turn has been used to train a gradient boosting regression model. If the model has overfitted to the data, predictions with smaller errors may be assumed to have belonged to the training set. We used Gaussian mixture models to attempt to cluster these errors (and thus the records) into either belonging to the training set or to the test set. For simplicity, the records were balanced to a 30/30 split in the true labels.

2.5. *Model selection*

As the missingness of the data was high, the choice of imputation method was important. Four different methods tested are shown in Figures 4 and 5. While on KS tests MICE was the best-performing imputation model (Figure 4), HyperImpute was the best at preserving pairwise correlations (Figure 5). The correlations of the variables were deemed to be more important, and for this reason, HyperImpute was chosen as the imputation method.

To justify the choice of the VAE+FCNN method against temporal models, we also tested the performance of two methods implemented in the *synthcity* library, *timevae* and *timegan*. While these tests were not as extensive as the VAE+FCNN tests and were done on out-of-the-box models, the preliminary results were not promising enough to warrant further investigation for the purposes of this study. It can be seen in Figure 6 that neither model generated data that followed the distribution of the real data. The scatter plot of weight in the baseline measurement against weight in the response in Figure 7 illustrates this: Both models suffered from mode collapse and did not generate diverse values.

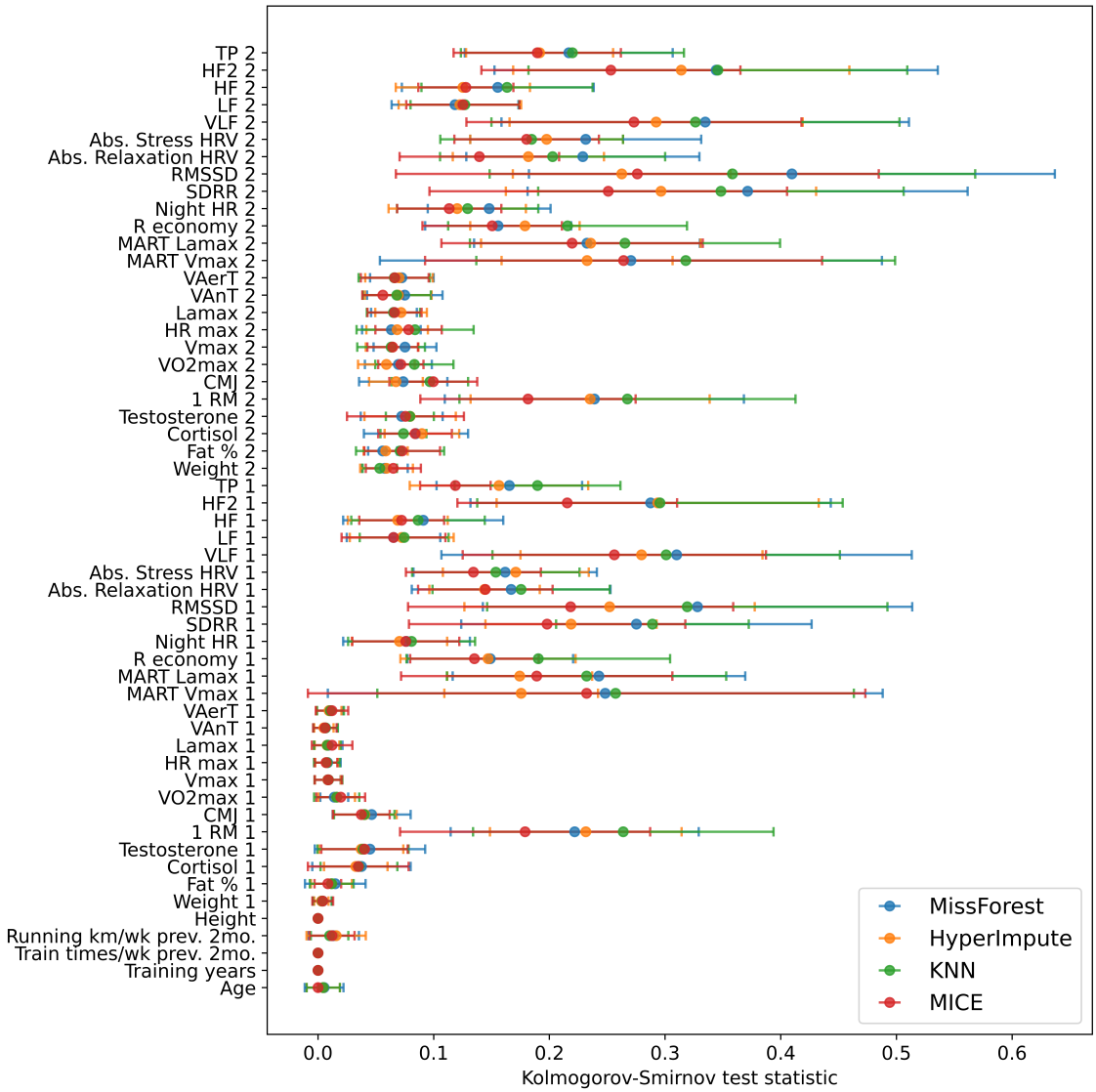

Figure 4. KS test statistics for imputation methods.

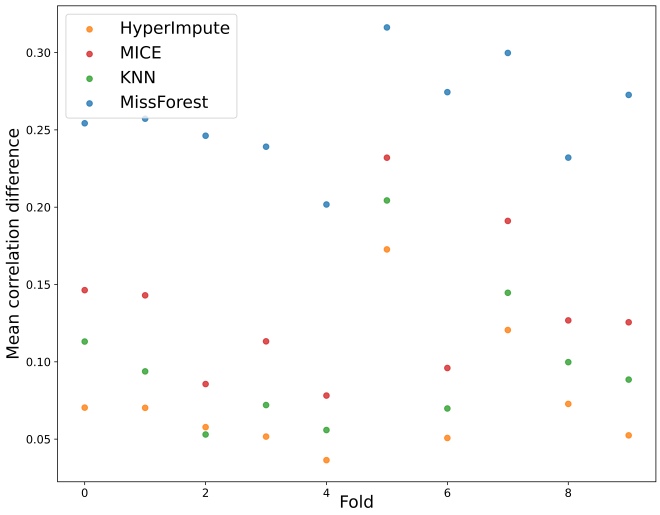

Figure 5. Correlation scores for imputation methods.

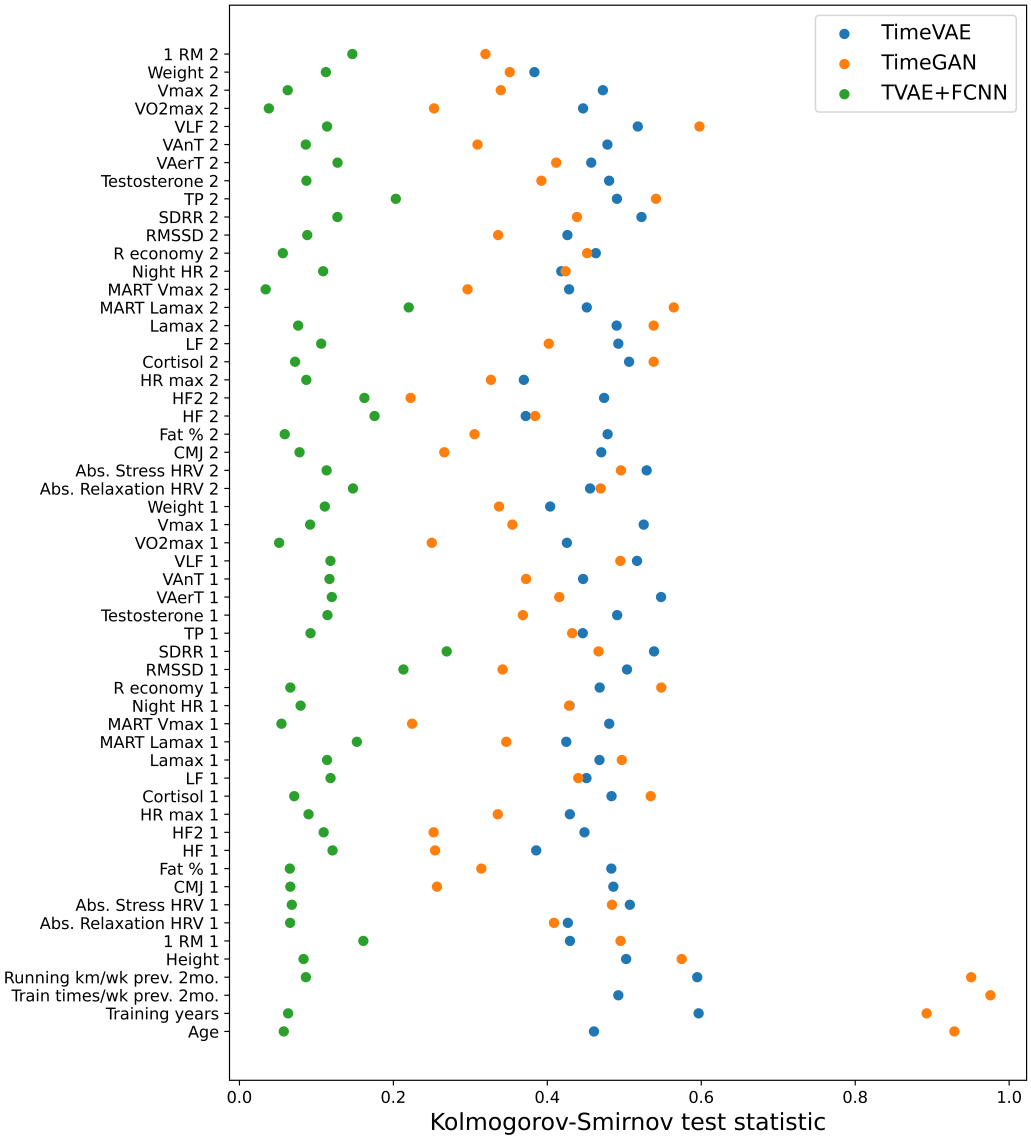

Figure 6. KS test statistics of temporal models.

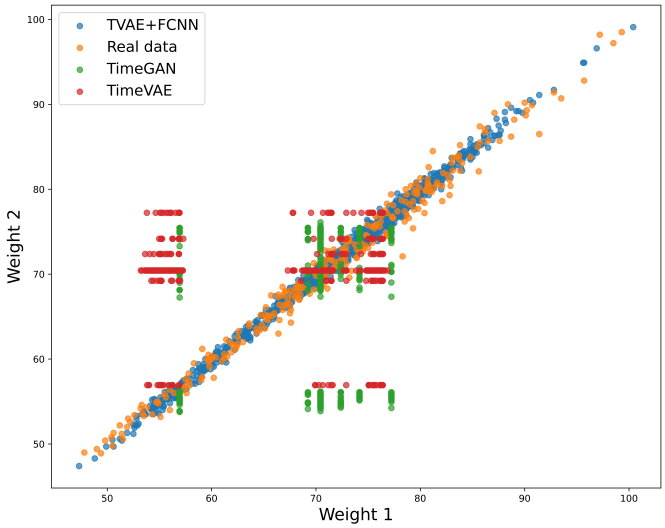

Figure 7. Distributions of the weights of the two time steps.

2.6. Model implementation and training parameters

The implementation of TVAE was from the Synthetic Data Vault [32]. It was used to generate synthetic baseline measurements after training for 3000 epochs on baseline measurements training data (Figure 8), using ADAM [33] for optimization ($\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, *weight decay* $= 10^{-5}$). TVAE was chosen over other generative models such as GANs or Bayesian networks due to the best average KS statistics and correlation scores in preliminary testing. The architecture and parameters were chosen by varying the number of nodes in the hidden layers in powers of two (e.g., 32, 64, 128, 256) and selecting the configuration with the best average KS statistics. Three different sizes of synthetic datasets were generated for comparison purposes. The smallest had the same n as the training set, varying from 265 to 273 depending on the cross-validation fold. Datasets with $n = 1000$ and $n = 10000$ were synthesized as the larger sets.

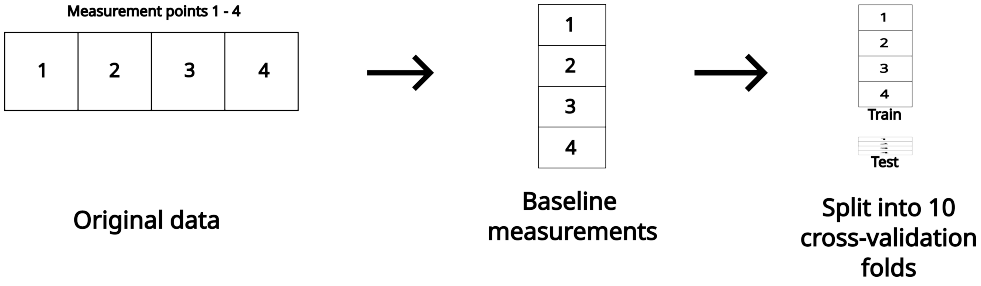

Figure 8. Preprocessing workflow for TVAE. Cross-validation folds were constructed at the subject level to prevent data leakage.

The FCNN used to predict the training responses was built on TensorFlow (2.16.1). It was trained on the baseline-response pairs, which comprised of two concurrent measurement points together with the training program metrics (Figure 9). The inputs for the FCNN were the baseline measurements and the training period metrics, the predicted variables were the differences from the baseline to the response, and it was trained for 1000 epochs using ADAM for optimization ($\alpha = 0.001$, $\beta_1 = 0.9$,

$\beta_2 = 0.999$, $\epsilon = 10^{-8}$, *weight decay* = 0). The architecture and parameters were chosen using the average KS statistics and correlation scores, with a similar search protocol as in TVAE. The details for both components of the synthetic data generation model are shown in Figure 2.

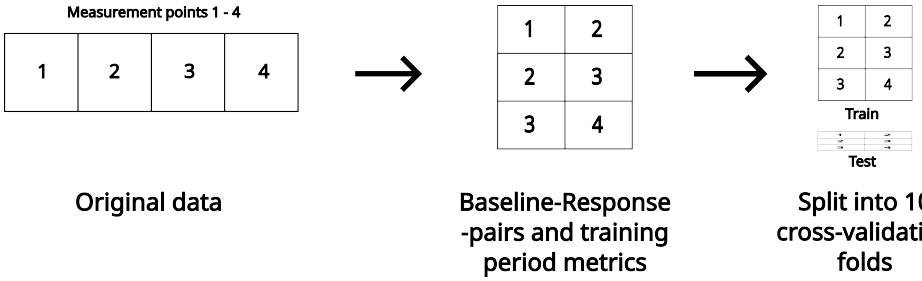

Figure 9. Preprocessing for training response generation. The same cross-validation folds were used as in the baseline measurement generation.

The use of models designed for temporal data was also considered. However, after preliminary testing on two models from the Synthcity library [34] (*TimeGAN* and *TimeVAE*) we decided to focus on our model. These models should be revisited in a future study with larger and more longitudinal datasets, where they may prove better than our model.

The implementations of the utility evaluation models (*GradientBoostingRegressor* and *LinearRegression*) were from the machine learning library Scikit-Learn (1.5.0) with default parameters. They were chosen from a variety of machine learning models as the best and worst performers when trained on real data. The statistical test implementations (Kolmogorov-Smirnov, Welch's t-test) were from SciPy (1.13.1). For everything else, NumPy (1.26.4) was used.

3. Results

3.1. Machine learning models

Table 2 shows the root mean square errors (RMSE) for gradient boosting and linear regression models when tested on the same test sets, averaged over the 10-fold cross-validation, when the models were trained on the smallest synthetic dataset ($n = n_{\text{training set}}$). Linear regression models trained on synthetic data had similar performance to gradient boosting models trained on synthetic data, while on models trained on real data, gradient boosting models tended to outperform the linear models. On models trained on larger synthetic datasets ($n = 1000$ and $n = 10000$), the pattern was similar and can be seen in Figure 10.

The statistical significance of the observed differences was assessed using Welch's t-test, and the significance levels can be seen in Table 2. On two out of 25 variables, Welch's t-tests showed a statistically significant difference between gradient-boosting models trained on synthetic data and on real data. On both of these variables, the model trained on synthetic data had better performance. For linear regression models, 19 variables had significantly different RMSEs, and on all of these, the RMSE was lower with synthetic data. The same results were also observed in the results of models trained on the larger synthetic datasets, and in models trained on n_{10000} on gradient boosting models four of the predictions were significantly different (Figure 10). The RMSEs of linear models did not differ significantly when trained on the larger synthetic datasets (Figure 11).

Table 2. RMSEs for models trained on synthetic ($n = n_{\text{training set}}$) and real data when tested on individual test set (significance levels 0.05 > * > 0.01 > ** > 0.001 > ***).

	Gradient boosting			Linear regression		
	TSTR	TRTR	sign.	TSTR	TRTR	sign.
Weight (kg)	0.161	0.172		0.135	0.166	*
Fat %	0.283	0.285		0.289	0.329	
Cortisol (nmol/l)	0.819	0.801		0.800	1.277	***
Testosterone (nmol/l)	0.303	0.306		0.301	0.377	
1 RM (kg)	0.329	0.403	*	0.326	0.560	***
CMJ (cm)	0.416	0.393		0.401	0.587	*
VO2max (l/min)	0.309	0.285		0.276	0.324	
Vmax (km/h)	0.326	0.328		0.301	0.363	*
HR max (bpm)	0.415	0.454		0.406	0.570	**
Lamax (mmol/l)	0.769	0.820		0.795	1.095	
VAnT (km/h)	0.398	0.384		0.405	0.494	
VAerT (km/h)	0.439	0.412		0.442	0.506	
MART Vmax (m/s)	0.281	0.307		0.253	0.332	*
MART Lamax (mM)	0.474	0.520		0.457	0.726	**
R economy (ml/kg/km)	0.612	0.719	*	0.640	1.365	**
Night HR (bpm)	0.749	0.722		0.752	1.007	*
SDRR (ms)	0.764	0.811		0.736	1.252	**
RMSSD (ms)	0.896	1.005		0.744	1.230	***
Abs. relaxation HRV (ms)	0.700	0.697		0.671	1.061	***
Abs. stress (ms)	0.648	0.653		0.653	1.188	***
VLF ($\ln(ms^2)$)	0.736	0.713		0.785	1.249	**
LF ($\ln(ms^2)$)	0.756	0.807		0.873	1.253	*
HF ($\ln(ms^2)$)	0.753	0.885		0.679	1.396	***
HF2 ($\ln(ms^2)$)	0.632	0.637		0.564	1.065	***
TP ($\ln(ms^2)$)	0.633	0.681		0.616	1.161	***

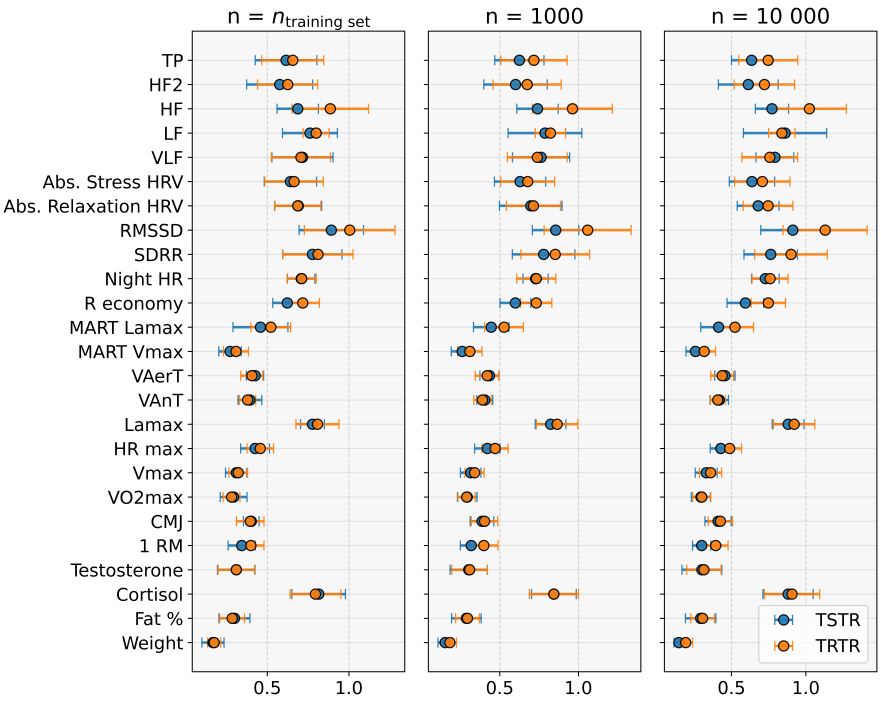

Figure 10. RMSEs of gradient boosting models.

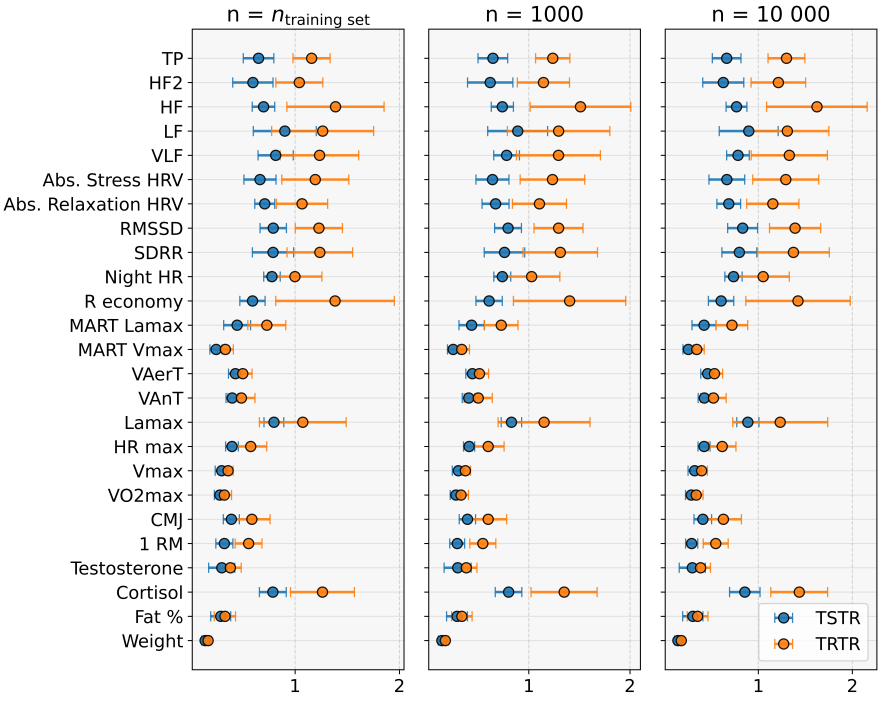

Figure 11. RMSEs of linear regression models.

The mean absolute errors of the predictions yielded similar results to the RMSEs, except with more variables with significant differences. They also had slightly more deviation, as seen in Figure 12.

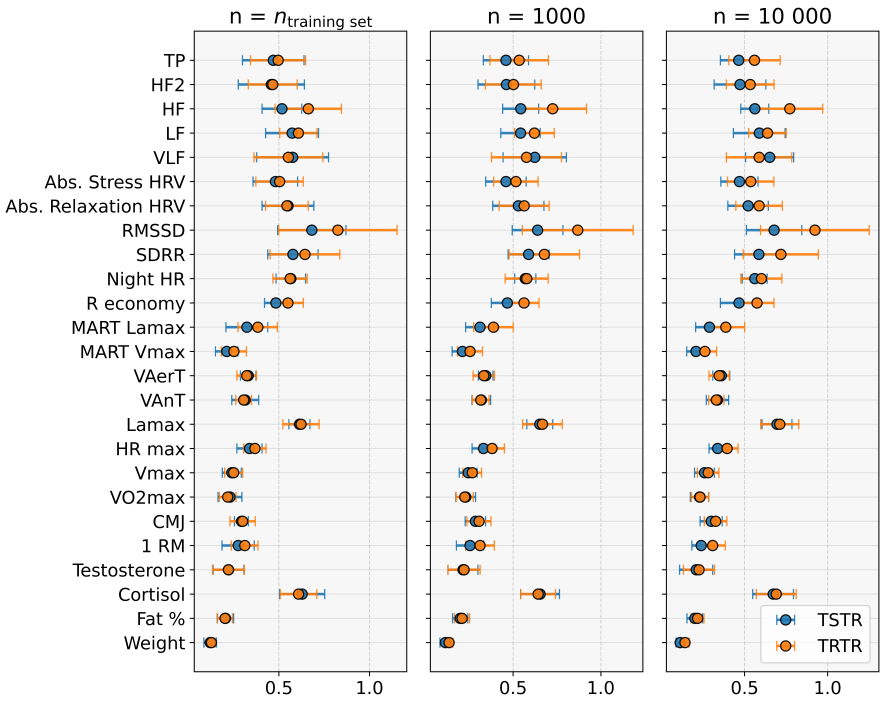

Figure 12. MAEs of gradient boosting model predictions.

3.2. Statistical similarity tests

The KS tests showed a moderate variance in the test statistics between different variables, as can be seen in Figure 13. For individual variables, the KS test statistic ranged from 0.049 (VO2max 1) to 0.314 (RMSSD 2), averaged across the ten cross-validation folds. The correlations between the means and standard deviations in the KS test statistics of the three synthetic datasets were 0.700, 0.564, and 0.628 (n_{training}, n_{1000}, and n_{10000}), indicating that variables with a higher test statistic tended to vary slightly more. The total average KS test statistics were 0.135, 0.128, and 0.126 with standard deviations 0.0604, 0.060, and 0.059. The test statistics were on average higher for the second measurement variables. This was also seen in the Wasserstein distances (Figure 13). These being calculated for the categorical variables showed that there was a notable difference between the distributions of strength group designations between real and synthetic data.

The average absolute differences in correlation were 0.064, 0.060, and 0.060 (n_{training}, n_{1000}, and n_{10000}), with the real dataset's correlation matrix calculated from unimputed values. Correlations that were not possible to calculate due to missing values were replaced with zeroes, which made the average differences slightly higher. Using imputed values, the average correlation differences were 0.045, 0.040, and 0.0384. The correlation differences were higher in the response variables, which can be seen in the lower right quadrants of heatmaps in Figure 14.

The NNDRs for the datasets can be seen in Table 3. There was no notable difference between the different dataset sizes, and while NNDR is larger in the test set than in the the datasets and there is some loss of privacy, none of these differences were significant. Although this might indicate a tendency for the synthetic data to have more similarity to the training data, some level of difference in the distance ratios of the independent test data was expected.

The results of the membership inference attack on the n_{1000} synthetic dataset can be seen in Table 4. While the specificity values indicate that many of the records not used in training could be ruled out, the other metrics show that it is difficult to discern whether a record was used in training.

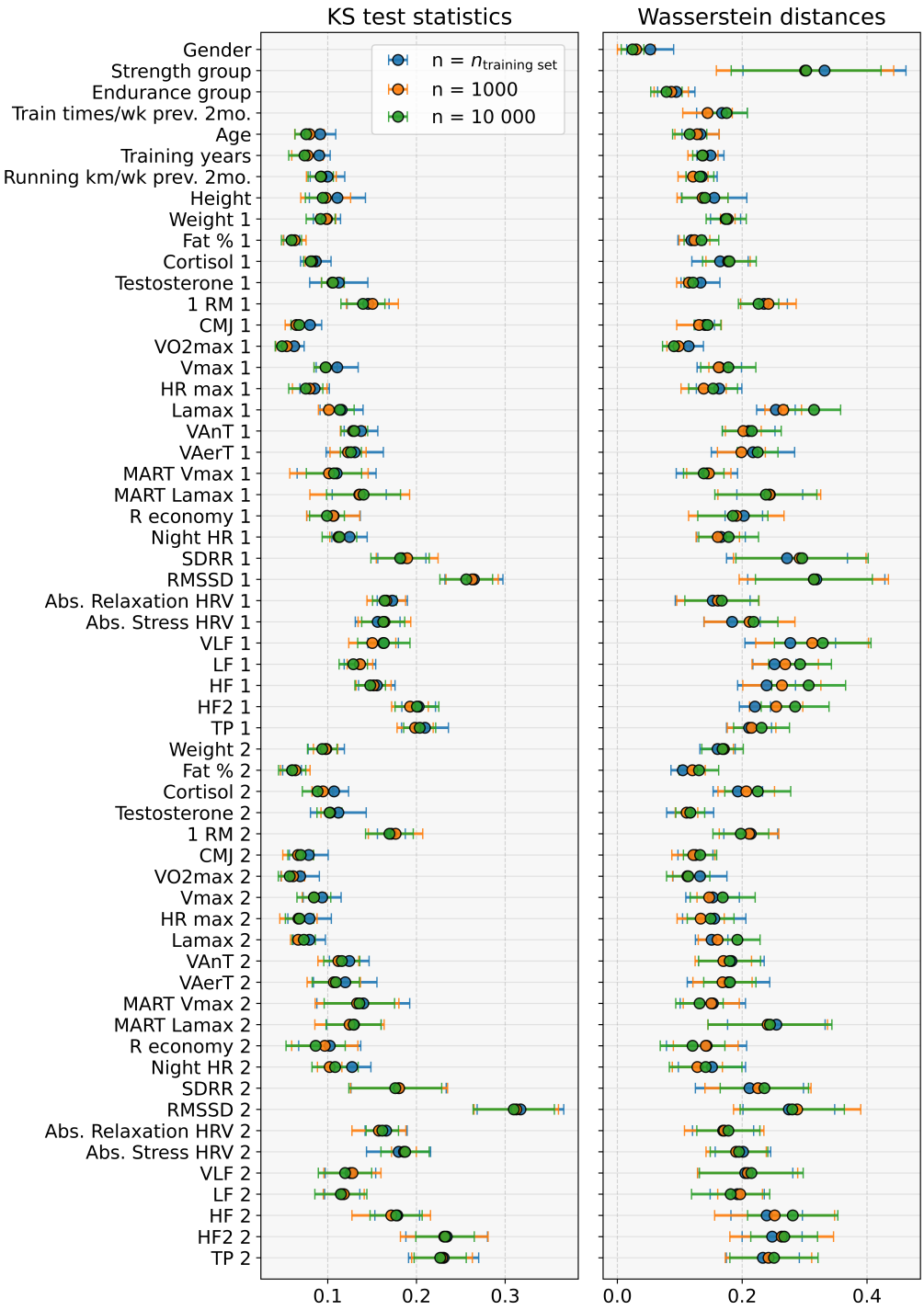

Figure 13. Univariate comparisons between real and synthetic data.

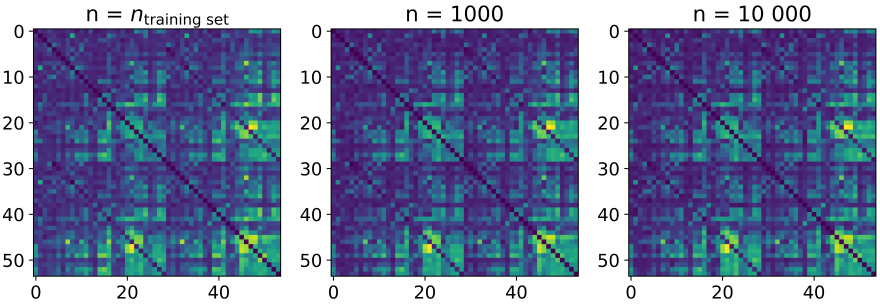

Figure 14. Absolute differences in correlation between real and synthetic data. The indices correspond to the variables in Figure 13.

Table 3. Nearest neighbor distance ratio (NNDR) and difference to NNDR calculated on test set (Privacy Loss).

	NNDR	Privacy Loss
test	0.9537 ± 0.0078	0
n = equal	0.9481 ± 0.0023	0.0057 ± 0.0080
n = 1000	0.9482 ± 0.0022	0.0055 ± 0.0070
n = 10 000	0.9483 ± 0.0018	0.0054 ± 0.0074

Table 4. Membership inference attack on the n_{1000} synthetic dataset.

	precision	Sensitivity	Specificity	F1-score
Weight	0.482	0.273	0.707	0.349
Fat %	0.530	0.32	0.717	0.399
Cortisol	0.478	0.453	0.507	0.466
Testosterone	0.498	0.397	0.6	0.442
1 RM	0.475	0.38	0.58	0.422
CMJ	0.517	0.3	0.72	0.380
VO2max	0.512	0.487	0.537	0.499
Vmax	0.547	0.54	0.553	0.544
HR max	0.525	0.427	0.613	0.471
Lamax	0.490	0.43	0.553	0.458
VAnT	0.458	0.37	0.563	0.410
VAerT	0.529	0.367	0.673	0.433
MART Vmax	0.477	0.41	0.55	0.441
MART Lamax	0.368	0.21	0.64	0.268
R economy	0.471	0.403	0.547	0.4345
Night HR	0.512	0.37	0.647	0.429
SDRR	0.464	0.32	0.63	0.377
RMSSD	0.510	0.35	0.663	0.415
Abs. Relaxation HRV	0.469	0.497	0.437	0.482
Abs. Stress HRV	0.485	0.377	0.6	0.424
VLF	0.5186	0.557	0.483	0.537
LF	0.491	0.367	0.62	0.420
HF	0.502	0.38	0.623	0.433
HF2	0.449	0.28	0.657	0.345
TP	0.452	0.497	0.397	0.473

4. Discussion

The findings of this study demonstrate that synthetic data generated using a combined variational autoencoder and neural network approach can achieve predictive performance comparable to that of real data in the context of recreational runner data. In most instances, particularly with linear regression models, the synthetic data even outperformed the real data. Although most observed differences in model performance were not statistically significant, there was no evidence of substantial degradation in data quality attributable to the synthetic data generation process. Furthermore, the differences in pairwise correlations between real and synthetic datasets were generally small, and the Kolmogorov-Smirnov tests indicated that the distributions of individual variables were well preserved.

The use of gradient boosting to predict training responses yielded similar accuracies when trained on real and synthetic data. However, for the two variables with a statistically significant difference, the model trained on synthetic data outperformed those trained on real data. Linear regression models exhibited highly improved performance when trained on synthetic data, with better results in every variable. This is particularly interesting when the sizes of the synthetic and training sets were the same, as the result cannot be explained by the larger size of the synthetic dataset. The larger size might enhance model robustness or lower the standard deviation in univariate distributions, causing the easy predictions to mask the outliers. The generative modeling might do most of the work in the utility testing framework, as the linear model is created on top of the generative model. This requires further investigation, as it calls into question the usefulness of utility testing using other models.

These results do indicate some usefulness in synthetic data usage, especially for enriching small datasets. It is plausible that synthetic data generation methods, by attenuating outliers, may facilitate more stable model training and improved average predictions. Lower variability has previously been observed in synthetic datasets [35]. This, however, conflicts with the goal of generating diverse synthetic data, as rare cases are often important in healthcare solutions [36].

The results indicate that VAEs have potential as a generative method in well-being research. The field is currently dominated by GANs, which have been shown to be effective in generating synthetic data [13], but they are often difficult to train and require large datasets. The FCNN component of the generative pipeline appeared more susceptible to modeling errors than the VAE, as reflected in both KS test statistics and machine learning model RMSE values. These errors may be minimized with more sophisticated generative models, such as transformer models [37] or diffusion models [38], as recurrent neural networks are known to be less effective for long sequences [39]. The inherent complexity of longitudinal well-being data, characterized by numerous unobserved confounding factors, presents a significant challenge for synthetic data generation, yet addressing this complexity remains essential for modeling health and well-being trajectories [40].

A notable challenge encountered in this study was the prevalence of missing data. Even minor errors introduced during imputation can propagate through the generative process, ultimately diminishing the quality of the synthetic data. As a result, variables with higher rates of missingness generally exhibited poorer KS test statistics and correlation scores. Additionally, missing information regarding inconsistencies in exercise activity between measurement points complicated the prediction of training responses. These are common issues in well-being and health data, where missing data are prevalent due to factors such as participant dropout, measurement errors, or challenges in data collection. While synthetic data generation can mitigate some of these issues [36], such as bias in participant dropout,

the quality of real training data remains crucial for ensuring the reliability of the generated synthetic data.

Privacy is a core subject in synthetic data and warrants further attention in future research. While the results of this study are promising, as the synthetic data did not explicitly contain any copied data points and the membership inference attack did not clearly show signs of overfitting, this alone does not guarantee safety from malicious adversaries. The distance ratios may be skewed by variables that were less successfully modeled, potentially leading to vulnerabilities in the data. Outliers are always the highest risk for information leakage, which may be minimized using differential privacy. The difficulties in balancing the trade-off between data utility and privacy preservation are well known [41]. Future work should explore more rigorous privacy evaluation methods and mitigation strategies to ensure that synthetic data can be shared safely and freely.

5. Conclusions

In this study, we showed that for small datasets, synthetic data have the potential to serve as a surrogate for real data. Given the high costs and logistical challenges associated with collecting physiological and performance data, such as those obtained from incremental treadmill tests, synthetic data may offer a practical alternative for preliminary hypothesis testing and application development. Nevertheless, findings derived from synthetic data should always be validated against real data, as synthetic data are not perfect representations of real phenomena.

Accurately modeling longitudinal data remains a significant challenge, especially when the datasets are limited in size and completeness, which are characteristics common in well-being and health data. While this study focused on synthesizing three-month time steps, well-being data often consist of multiple time steps, which would require more sophisticated models to generate. Furthermore, privacy issues will also need to be addressed if synthetic data are to be regarded as a safer alternative to real data. These can be either methods that by definition generate data with privacy guarantees (such as differential privacy) or more robust metrics to ensure that sensitive information cannot leak from synthetic datasets.

For synthetic data to achieve widespread adoption in well-being research and industry, further methodological advancements and standardization of validation processes are required. Current evaluation practices vary considerably across studies, making it difficult to assess the generalizability and utility of synthetic datasets. Although the methods used in this study indicate that the quality of the generated data is good, it is difficult to determine the usefulness of the data if released. Establishing well-defined validation protocols and acceptable quality thresholds is required for synthetic data to gain the trust needed for widespread use.

Acknowledgments

The work is related to the Wellbeing DataLab project that has received funding from the Regional Council of Central Finland (A80232), and the European Regional Development Fund and Leverage from the EU (2021-2027). This work was also supported by the Research Council of Finland (356158), the City of Jyväskylä, and Polar Electro Oy. The funders had no role in study design, data collection and analysis, the decision to publish, or the preparation of the manuscript.

Use of AI tools declaration

Microsoft Copilot (GPT4.1) was used to assist with the grammar and spelling of the manuscript.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. A. Esteva, B. Kuprel, R. A. Novoa, J. Ko, S. M. Swetter, H. M. Blau, et al., Dermatologist-level classification of skin cancer with deep neural networks, *Nature*, **542** (2017), 115–118. https://doi.org/10.1038/nature21056
2. W. Gouda, M. Almurafeh, M. Humayun, N. Z. Jhanjhi, Detection of COVID-19 based on chest X-rays using deep learning, *Healthcare*, **10** (2022), 343. https://doi.org/10.3390/healthcare10020343
3. J. Abramson, J. Adler, J. Dunger, R. Evans, T. Green, A. Pritzel, et al., Accurate structure prediction of biomolecular interactions with AlphaFold 3, *Nature*, **630** (2024), 493–500. https://doi.org/10.1038/s41586-024-07487-w
4. *European Parliament and Council of the European Union, Regulation (EU) 2016/679 of the European Parliament and of the Council*, European Union, 2016. Available from: `https://eur-lex.europa.eu/eli/reg/2016/679/oj`.
5. J. Jordon, L. Szpruch, F. Houssiau, M. Bottarelli, G. Cherubin, C. Maple, et al., Synthetic data—what, why and how? arXiv: 2205.03257. https://doi.org/10.48550/arXiv.2205.03257
6. A. Benaim, R. Almog, Y. Gorelik, I. Hochberg, L. Nassar, T. Mashiach, et al., Analyzing medical research results based on synthetic data and their relation to real data results: systematic comparison from five observational studies, *JMIR Med. Inform.*, **8** (2020), e16492. https://doi.org/10.2196/16492
7. A. Gonzales, G. Guruswamy, S. R. Smith, Synthetic data in health care: a narrative review, *PLOS Digit Health*, **2** (2023), e0000082. https://doi.org/10.1371/journal.pdig.0000082
8. M. Frid-Adar, E. Klang, M. Amitai, J. Goldberger, H. Greenspan, Synthetic data augmentation using GAN for improved liver lesion classification, *Proceedings of IEEE 15th international symposium on biomedical imaging*, 2018, 289–293. https://doi.org/10.1109/ISBI.2018.8363576
9. S. Lala, M. Shady, A. Belyaeva, M. Liu, Evaluation of mode collapse in generative adversarial networks, *High Performance Extreme Computing*, 2018, 1–9.
10. P. Eigenschink, T. Reutterer, S. Vamosi, R. Vamosi, C. Sun, K. Kalcher, Deep generative models for synthetic data: a survey, *IEEE Access*, **11** (2023), 47304–47320. https://doi.org/10.1109/ACCESS.2023.3275134
11. K. El Emam, Seven ways to evaluate the utility of synthetic data, *IEEE Secur. Priv.*, **18** (2020), 56–59. https://doi.org/10.1109/MSEC.2020.2992821
12. M. Hernadez, G. Epelde, A. Alberdi, R. Cilla, D. Rankin, Synthetic tabular data evaluation in the health domain covering resemblance, utility, and privacy dimensions, *Methods Inf. Med.*, **62** (2023), e19–e38. https://doi.org/10.1055/s-0042-1760247

13. H. Murtaza, M. Ahmed, N. F. Khan, G. Murtaza, S. Zafar, A. Bano, Synthetic data generation: State of the art in health care domain, *Comput. Sci. Rev.*, **48** (2023), 100546. https://doi.org/10.1016/j.cosrev.2023.100546

14. R. Shokri, M. Stronati, C. Song, V. Shmatikov, Membership inference attacks against machine learning models, *Proceedings of IEEE symposium on security and privacy*, 2017, 3–18. https://doi.org/10.1109/SP.2017.41

15. B. van Breugel, H. Sun, Z. Qian, M. van der Schaar, Membership inference attacks against synthetic data through overfitting detection, arXiv: 2302.12580. https://doi.org/10.48550/arXiv.2302.12580

16. N. Carlini, F. Tramer, E. Wallace, M. Jagielski, A. Herbert-Voss, K. Lee, et al., Extracting training data from large language models, *Proceedings of 30th USENIX security symposium (USENIX Security 21)*, 2021, 2633–2650.

17. G. Gondim-Ribeiro, P. Tabacof, E. Valle, Adversarial attacks on variational autoencoders, arXiv: 1806.04646. https://doi.org/10.48550/arXiv.1806.04646

18. A. Yale, S. Dash, R. Dutta, I. Guyon, A. Pavao, K. P. Bennett, Generation and evaluation of privacy preserving synthetic health data, *Neurocomputing*, **416** (2020), 244–255. https://doi.org/10.1016/j.neucom.2019.12.136

19. H. Ping, J. Stoyanovich, B. Howe, Datasynthesizer: privacy-preserving synthetic datasets, *Proceedings of the 29th International Conference on Scientific and Statistical Database Management*, 2017, 1–5. https://doi.org/10.1145/3085504.3091117

20. D. Kaur, M. Sobiesk, S. Patil, J. Liu, P. Bhagat, A. Gupta, et al., Application of Bayesian networks to generate synthetic health data, *J. Am. Med. Inform. Assoc.*, **28** (2020), 801–811. https://doi.org/10.1093/jamia/ocaa303

21. B. Tang, H. He, KernelADASYN: kernel based adaptive synthetic data generation for imbalanced learning, *Proceedings of IEEE congress on evolutionary computation (CEC)*, 2015, 664–671. https://doi.org/10.1109/CEC.2015.7256954

22. I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, et al., Generative adversarial nets, *Proceedings of the 28th International Conference on Neural Information Processing Systems*, 2014, 2672–2680.

23. D. P. Kingma, M. Welling, Auto-encoding variational Bayes, arXiv: 1312.6114. https://doi.org/10.48550/arXiv.1312.6114

24. *D. Aha, K. Nottingham, R. Longjohn, M. Kelly, P. Murphy, C. Merz, et al., UCI machine learning repository*, UC Irvine, 2007. Available from: `https://archive.ics.uci.edu/ml/index.php`.

25. A. Johnson, T. Pollard, L. Shen, H. Lehman, M. Feng, M. Ghassemi, et al., MIMIC-III, a freely accessible critical care database, *Sci. Data*, **3** (2016), 160035 https://doi.org/10.1038/sdata.2016.35

26. D. Jarrett, B. Cebere, T. Liu, A. Curth, M. van der Schaar, Hyperimpute: generalized iterative imputation with automatic model selection, *Proceedings of the 39th International Conference on Machine Learning*, 2022, 9916–9937.

27. L. Xu, M. Skoularidou, A. Cuesta-Infante, K. Veeramachaneni, Modeling tabular data using conditional GAN, *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 2019, 7335–7345.
28. C. M. Bishop, *Pattern recognition and machine learning*, New York: Springer, 2006.
29. J. H. Friedman, Greedy function approximation: a gradient boosting machine, *Ann. Statist.*, **29** (2001), 1189–1232. https://doi.org/10.1214/aos/1013203451
30. T. Hastie, R. Tibshirani, J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, New York: Springer, 2009. https://doi.org/10.1007/978-0-387-21606-5
31. L. N. Vaserstein, Markov processes over denumerable products of spaces, describing large systems of automata, *Probl. Peredachi Inf.*, **5** (1969), 64–72.
32. N. Patki, R. Wedge, K. Veeramachaneni, The synthetic data vault, *Proceedings of IEEE International Conference on Data Science and Advanced Analytics (DSAA)*, 2016, 399–410. https://doi.org/10.1109/DSAA.2016.49
33. D. P. Kingma, J. Ba, Adam: a method for stochastic optimization, *Proceedings of 3rd International Conference for Learning Representations*, 2015, 6.
34. Z. Qian, B. C. Cebere, M. van der Schaar, Synthcity: facilitating innovative use cases of synthetic data in different data modalities, arXiv: 2301.07573. https://doi.org/10.48550/arxiv.2301.07573
35. M. Hernandez, G. Epelde, A. Beristain, R. Álvarez, C. Molina, X. Larrea, et al., Incorporation of synthetic data generation techniques within a controlled data processing workflow in the health and wellbeing domain, *Electronics*, **11** (2022), 812. https://doi.org/10.3390/electronics11050812
36. B. van Breugel, T. Liu, D. Oglic, M. van der Schaar, Synthetic data in biomedicine via generative artificial intelligence, *Nat. Rev. Bioeng.*, **2** (2024), 991–1004. https://doi.org/10.1038/s44222-024-00245-7
37. A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, et al., Attention is all you need, *Proceedings of the 31st International Conference on Neural Information Processing Systems* , 2017, 6000–6010.
38. J. Ho, A. Jain, P. Abbeel, Denoising diffusion probabilistic models, *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020, 6840–6851.
39. F. M. Shiri, T. Perumal, N. Mustapha, R. Mohamed, A comprehensive overview and comparative analysis on deep learning models: CNN, RNN, LSTM, GRU, arXiv: 2305.17473. https://doi.org/10.48550/arXiv.2305.17473
40. A. Amirahmadi, M. Ohlsson, K. Etminani, Deep learning prediction models based on EHR trajectories: a systematic review, *J. Biomed. Inform.*, **144** (2023), 104430. https://doi.org/10.1016/j.jbi.2023.104430
41. T. Stadler, B. Oprisanu, C. Troncoso, Synthetic data—anonymisation groundhog day, *Proceedings of the 31st USENIX Security Symposium*, 2022, 1451–1468.

AIMS Press