Universiteit Leiden

Master Computer Science

Applying Counterfactual Explanations and Multivariate Forecasting to Medical Prediction Tasks

Name:	Tomke Meyer
Student ID:	s2231086
Date:	dd/mm/2025
Specialisation:	Bioinformatics
1st supervisor:	Jan van Rijn
2nd supervisor:	Panos Papapetrou

Master's Thesis in Computer Science

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Abstract

Contents

1 Introduction

Time series forecasting plays a crucial role in a number of different applications, such as stock market predictions, weather forecasting, healthcare monitoring, and personalized treatment planning. The main goal of time series forecasting is to use historical observations to predict future values. For this task there are various models, including statistical approaches such as ARIMA, regression-based methods like Linear Regression, as well as deep learning models like N-Beats and transformer-based models such as DLinear. Especially the latter models have shown promising results in recent years, achieving very accurate predictions. Despite these recent developments, many deep-learning based forecasting systems are considered "black-box"-models, as it is challenging to interpret and understand both the modelling process and the forecasting outcome. This is particularly problematic in healthcare applications, where especially interpretability is very important, as clinicians need to understand the reasoning behind the predictions to make informed decisions. One approach to tackle this problem is working with counterfactual explanations, which aim to identify small changes in the input variables needed to change a model's forecast to a more desirable outcome.
In healthcare, time series forecasting has been applied to a number of different tasks, especially the analysis of electronic health records (EHRs). These contain information about a patient's health over time, allowing predictions about the disease progression, treatment efficacy or patient mortality. For example, forecasting glucose levels in diabetic patients or predicting heart failure with preserved ejection fraction (HFpEF) can aid early interventions and optimize treatment plans.

1.1 Diabetes

For patients with conditions such as type 1 diabetes mellitus (T1DM), closely tracking their glucose levels is a necessity. To reduce the risk of complications such as hyperglycaemia or hypoglycaemia, these patients rely on continuous glucose monitoring (CGM) devices and automated insulin delivery. Using machine learning (ML) to model CGM, can help to gain a better understanding of predicting abnormal glucose events and help with insulin dosage planning. By also incorporating variables such as insulin intake, carbohydrate consumption, and physical activity, a predictive model can allow timely interventions through the generation of actionable recommendations for patients or healthcare providers. This allows for more dynamic treatment based on these forecasted expected glucose trends, which can reduce the long-term risk of diabetes-related complications[Com25].

1.2 HFpEF

Another possible application is identifying early warning signs of heart failure or more specific HFpEF and suggest either lifestyle or treatment changes. Using the vital signs, as for example the heart rate and blood pressure, of a patient as well as other factors like gender and possible comorbidities, allows specific and personal monitoring of disease progression. This way, early warning signs of worsening heart condition can be identified and personalized modifications to lifestyle or medication can be suggested. Such proactive monitoring can help reduce hospital readmissions and improve patient outcomes.

1.3 Counterfactual Explanations

Traditional time series forecasting focuses mainly on predicting future values given historical observations, but modifying past values is not feasible in real-world settings, especially in healthcare applications. Instead, a more practical approach is to explore how changing exogenous variables during the forecast horizon could lead to a desired outcome. This approach allows continuous monitoring of patients, making it possible to dynamically adjust treatment plans to lead towards more optimal results.

1.4 Related Work

Recent research has explored various deep learning (DL) models for time series forecasting, including recurrent neural network (RNN)-based models such as gated recurrent units (GRU) and long short-term memory (LSTM), as well as attention-based architectures like transformers. Transformer-based models, including Autoformer and Informer, have demonstrated strong performance in both univariate and multivariate forecasting tasks by capturing long-range dependencies more effectively than traditional RNN approaches.

In the clinical domain, deep learning models have been applied extensively to glucose forecasting. For instance, Deep Multi-Output Forecasting [FAJ+18] introduced a multi-step forecasting framework that explicitly models the distribution of future glucose values over a prediction horizon using a multi-output deep architecture. Similarly, WaveNet has been adapted for glucose forecasting by leveraging dilated convolutional neural networks (CNNs) to model long-term dependencies [ZLH+18]. In addition, transfer learning techniques have been employed to enhance predictive performance by fine-tuning pre-trained models on patient-specific data while incorporating exogenous covariates such as insulin dosage and carbohydrate intake [MB20].

Beyond predictive performance, explainability remains a critical challenge in deep learning-based forecasting models. Traditional statistical models, such as ARIMAX and VARIMAX, are able to quantify relationships between exogenous factors and the target variable, but their forecasting accuracy is often outperformed by DL approaches. Recent research has focused on integrating explainability into forecasting models to combine the strengths of both interpretability and predictive performance. For example, NBEATSx extends the NBEATS framework by incorporating exogenous variables into its deep architecture, enabling a more structured decomposition of trend and seasonality. However, its interpretability remains static and does not fully capture the dynamic nature of forecasting outcomes.

To address the need for explainability, counterfactual explanations have gained traction in time series analysis. Initial efforts focused on time series classification, where counterfactuals were generated through instance-based modifications and gradient-based perturbations [AALC21]. This was done by introducing a framework for generating counterfactual explanations for multivariate time series classification, identifying minimal input modifications needed to alter the model's decision, providing interpretability for high-dimensional time series models.

More recently, counterfactual explanations have been extended to time series forecasting. ForecastCF [WMSP23] proposed a deep learning-based method for generating counterfactuals in time series forecasting by identifying minimal input changes required to achieve desired prediction outcomes. Building on this, COMET [WSMP24] extended counterfactual explanations to multivariate time series forecasting, focusing on modifying exogenous variables (e.g., insulin, carbohydrates, and exercise) to generate actionable recommendations for glucose management.

In addition to counterfactual forecasting, diagnostic tools like TimeTuner [HSYZ23] have been developed to analyse how time representations influence forecasting models. By employ-

ing counterfactual explanations, TimeTuner enables the evaluation of multivariate time series representations and their impact on model predictions.
Despite these advances, counterfactual explanations for multivariate time series analysis remain an emerging research area. While existing methods demonstrate the feasibility of generating counterfactuals for univariate forecasts, their generalization to multivariate forecasting and real-world clinical applications remains limited. Our work aims to extend on these existing methods by integrating counterfactual reasoning with multivariate forecasting models, focusing on modifying exogenous variables within the prediction horizon to provide actionable and interpretable interventions.

1.5 Main Contributions

Time series forecasting can be quite useful in healthcare by predicting how well a treatment works or predicting the risk of complications, relapse or mortality. Recent studies, such as COMET [WSMP24], have been focusing on counterfactuals in time series forecasting, but these methods work by altering historical observations. Our work aims to bridge this gap by developing a counterfactual forecasting mechanism that identifies optimal changes in exogenous variables during the forecast horizon to achieve desirable outcomes. More specifically, we propose a method that learns the relationship between the forecasted targets and exogenous variables, which leads to a more effective and interpretable decision-making in healthcare.
The main contributions of this paper can be summarized as follows:

- We propose a new model for counterfactual time series forecasting to achieve a desired constrained forecast by modifying exogenous variables within the forecast horizon.
- We incorporate existing forecasting models, such as SARIMAX, OLS, GRU and NBEATSx, for learning the relationship between exogenous variables and a target variable to ensure actionable and interpretable predictions.
- We evaluate the models on two applications in healthcare, specifically for glucose level prediction and HFpEF management, demonstrating the practical utility of our approach.

2 Methodology

2.1 Problem Formulation

1) Let $\mathbf{X} := (\mathbf{x}_i)_{i\in\{1,\dots,n\}}$ be a time series of length n (*back horizon*), with each $x_i \in \mathbb{R}^{m+1}$ composed of the target variable $y_i \in \mathbb{R}$ and the exogenous variables

$$z_i = \begin{pmatrix} z_{1,i} \\ \vdots \\ z_{m,i} \end{pmatrix} \in \mathbb{R}^m.$$

Then $\mathbf{X}$ can be denoted as the combined matrix of the target vector $\mathbf{y} \in \mathbb{R}^{1\times n}$ and the exogenous matrix $\mathbf{Z} \in \mathbb{R}^{m\times n}$:

$$\mathbf{X} = \begin{pmatrix} \mathbf{y} \\ \mathbf{Z} \end{pmatrix} := \begin{pmatrix} y_1 & \dots & y_n \\ z_{1,1} & \dots & z_{1,n} \\ \vdots & \ddots & \vdots \\ z_{m,1} & \dots & z_{m,n} \end{pmatrix}.$$

The relationship between $\mathbf{y}$ and $\mathbf{Z}$ can be described by the function:

$$r : \mathbb{R}^{m\times n} \to \mathbb{R}^{1\times n}$$

which calculates the target vector from the exogenous matrix:

$$r(\mathbf{Z}) = \mathbf{y}.$$

2) Given a time series forecasting model f that predicts the next t values (*forecasting horizon*) of $\mathbf{X}$, we define the forecast as:

$$f(\mathbf{X}) = \mathbf{X}' := (x_{n+i})_{i\in\{1,\dots,t\}}.$$

Additionally, consider a lower and upper bound set of constraints for each time step in the forecasting horizon, denoted as:

$$\boldsymbol{\alpha} = (\alpha_{n+i})_{i\in\{1,\dots,t\}}, \boldsymbol{\beta} = (\beta_{n+i})_{i\in\{1,\dots,t\}}.$$

The objective is to generate a counterfactual time series sample $\mathbf{Z}^*$, such that $\mathbf{y}^* = r(\mathbf{Z}^*)$ satisfies the given bounds:

$$\alpha_i \leq y_i^* \leq \beta_i, \forall y_i^* \in \mathbf{y}^*, i \in \{n+1, \dots, n+t\}.$$

Summarized research objective: Given a target vector $\mathbf{y}$ affected by the exogenous matrix $\mathbf{Z}$, a forecast horizon t, the original forecasted vector $\mathbf{y}'$ and the original forecasted exogenous matrix $\mathbf{Z}'$, the goal is to modify $\mathbf{Z}'$ to $\mathbf{Z}^*$ such that the corresponding target vector $\mathbf{y}^*$ is within constraints $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$.

2.2 Algorithm

The developed algorithm takes as input time series data, split into the target variable and the exogenous variables. Additionally a number of parameters are defined, such as the learning rate, the desired bounds, the clipping range, the maximum number of iterations as well as different forecasting and regression models. The output is then the desired exogenous counterfactuals with the desired target outcome. First the multivariate forecasting is done, resulting in a target and exogenous variables that are then used for the counterfactual generation. The counterfactual generation starts with selecting test samples to generate the necessary bounds, as well as introducing an activity temporal constraint, that allows incorporating the constraint vector c in the loss calculation to encourage counterfactual changes when there is already a planned activity such as meal or exercise. After initializing the loss function, the counterfactuals

are generated using gradient perturbation with an Adam optimizer. The gradient perturbation includes two different constraints, a clipping constraint that ensures that the exogenous variables are within a given constraint range and a historical constraint, to ensure that the counterfactuals are more similar to historical input values. Algorithm 1 shows a pseudocode of the algorithm described above.

Algorithm 1 Counterfactual Hybrid Forecasting

Input: Time series data: target t, exogonous variables E, learning rate η, desired bounds (α, β), clipping range (ρ, ϕ), max iterations max_iter, differentiable forecaster $f(\cdot)$ differentiable regressor $r(\cdot)$, w, $\mathcal{G}$
Output: Counterfactual E' with desired outcome t'
$(t^*, E^*) \leftarrow (t, E)$
$(\hat{t}^*, \hat{E}^*) \leftarrow f(t^*, E^*)$
$\mathcal{S} \leftarrow$ SelectTestSamples
$(\alpha, \beta) \leftarrow$ GenerateBounds$(\mathcal{S})$
$C \leftarrow$ ActivityTemporalConstraint???
$loss \leftarrow L((t^*, E^*), \alpha, \beta, (t, E), C)$???
$t \leftarrow 0$
while $(\hat{t}^* > \beta$ **or** $\hat{t}^* < \alpha) \wedge (t <$ max_iter$)$ **do**
    $\hat{E}^* \leftarrow$ AdamOptimize$(\hat{E}^*, loss, \eta)$
    $\hat{E}^* \leftarrow$ Clip$(\hat{E}^*, \rho, \phi)$???
    $\hat{t}^* \leftarrow r(\hat{E}^*)$
    $C \leftarrow$ HistValueConstraint$(\hat{E}^*, \mathcal{G})$
    loss $\leftarrow L(\hat{E}^*, w, \alpha, \beta, X^*, C)$???
    $t \leftarrow t + 1$
end while
$(t', E') \leftarrow (t, E)$
return (t', E')

2.3 Experiments

2.3.1 Data preparation

There are many possible applications for the proposed model, including domains outside of health care. However, this research will focus only on medical applications, specifically on trying to identify optimal treatment plans for two different use cases: type 1 diabetes management and heart failure with preserved ejection fraction (HFpEF). The datasets used in this study contain physiological and treatment-related variables, allowing personalized forecasts and counterfactual interventions. To prepare the data for the proposed model, different preprocessing steps have been taken. The data is split into training, validation and test sets and normalized using min-max scaling to ensure stable model training, as well as split into the target and exogenous data. Then a sequence generator is used to slice the data into sequences of back horizon input and horizon targets for the prediction task and into sequences of back horizon input and horizon of both the exogenous and target features. This creates overlapping time series segments, of which any sequences with missing values are discarded.

2.3.1.1 SimGlucose

The SimGlucose dataset is generated using the FDA-approved UVA/PADOVA type 1 diabetes simulator [Xie18]. This Python-based simulator models the physiological responses of patients with type 1 diabetes and contains 30 virtual patients (10 adults, 10 adolescents, 10 children). It provides CGM measurements along with insulin dosage and carbohydrate intake. The dataset is generated with a predefined CGM sampling frequency and insulin pump settings according to the developed model [MML+14]. For this study, the measurements of ten adults over the course of one week were simulated. The blood glucose levels are used as the target variable, while the insulin dosages and carbohydrate intakes are used as exogenous variables. After generating the simulated data, the data is preprocessed to be useable for the predictive model. One example can be seen in figure 1, where BG is the blood glucose level, CHO the carbohydrate intake and Insulin the insulin dosage. The risk index shows the hyperglycaemic or hypoglycaemic risk for the patient. For the blood glucose the level the green range shows normal glucose levels between 70 and 180, and the red ranges are hyper- or hypoglycaemic glucose levels.

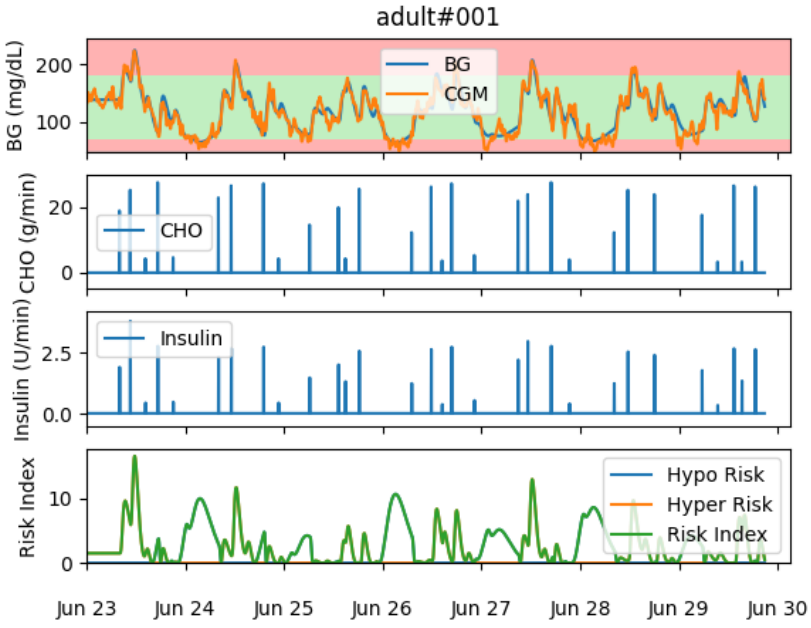


Figure 1: Simulation of an adult patient over the span of one week.

2.3.1.2 OhioT1DM:

The OhioT1DM dataset [MB20] contains real-world glucose monitoring data from 12 individuals with type-1 diabetes, collected by Ohio University over 8 weeks. According to earlier studies [CHN+21], [WSMP24], we extracted the most clinically relevant features, which includes the CGM measurements, basal insulin, bolus insulin, carbohydrate intake and physical activity. Compared to the SimGlucose dataset, this dataset includes more exogenous variables such as bolus and basal insulin, carbohydrate intake and physical activity, as. Since this dataset is collected from real patients, it includes missing data and irregular sampling intervals, which are handled using interpolation techniques and resampling. The availability of more exogenous

variables allows for more complex counterfactual interventions, improving the performance of the forecasting model. Figure 2 shows example measurement in a 24-hour window of one of the patients.

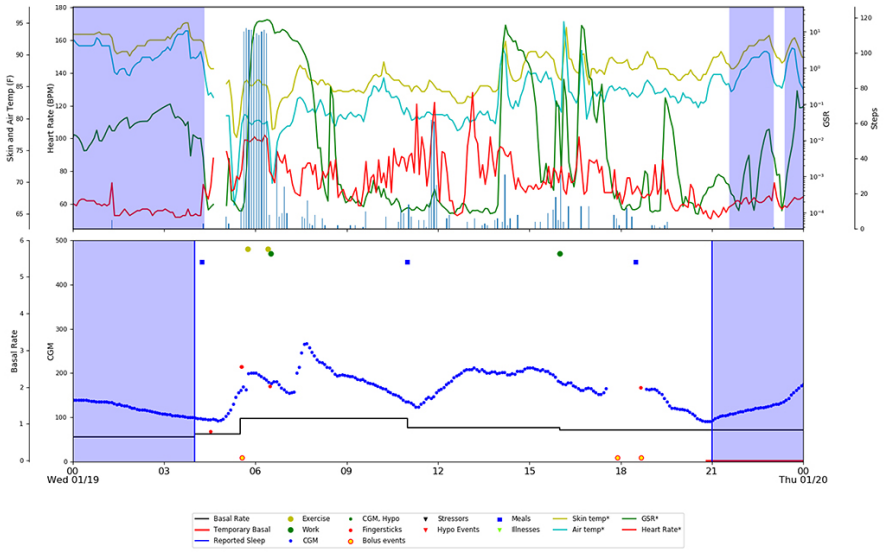


Figure 2: 24-hour measurements of one patient from the OhioT1DM Viewer [MB20].

2.3.1.3 MIMIC-III:

Beyond diabetes forecasting, the proposed model aims to generalize to other medical applications, such as predicting disease progression in HFpEF patients. The MIMIC-III dataset [JPS+16] contains de-identified electronic health records (EHRs) of ICU patients, including vital signs (heart rate, blood pressure, oxygen saturation, etc.), medication records and laboratory results. For this study, a subset of MIMIC-III focusing on cardiovascular patients is used. The target variable is the mortality risk, splitting into two groups (death within 30 days, death within 1 year), while the exogenous variables include vital signs and laboratory values. Gender and comorbidities are used to split the data into multiple cohorts. By analysing the data in different cohorts, it is possible to get more specific and accurate counterfactual interventions.

Preprocessing: Following previous work [SBA+24], ICD-codes were used for preprocessing. By using the appropriate ICD-9 and ICD-10 codes seen in Table 1, the hospital admissions of patients ≥ 18 with HFpEF as a primary diagnosis have been identified. Since the diagnosis was based on ICD codes, the clinical notes where analysed to validate the chosen patients. This was done by filtering the clinical notes on mentions of the left ventricular ejection fraction (LVEF) value, using regular expressions. Table 1 also shows the number of hospital admissions per diagnosis, here only the latest admission of patient that had been admitted earlier with a similar diagnosis was counted. Prior admissions were added as a comorbidity. In total there were 2043 admissions after filtering with the clinical notes.

The extracted features are listed in table 2, split into three categories. These are the targets (Death within 30 days, Death within 1 year) as well as vital signs and laboratory values, and comorbidities. Gender and prior admission was included in the list of comorbidities, since these

The study sample consisted of 3,235

features are used for the clustering and not the forecasting.

Diagnosis	ICD Code	Frequency
Unspecified diastolic (congestive) heart failure	I5030	33
Diastolic heart failure, unspecified	42830	71
Acute diastolic (congestive) heart failure	I5031	84
Acute diastolic heart failure	42381	169
Chronic diastolic (congestive) heart failure	I5032	249
Chronic diastolic heart failure	42382	428
Acute on chronic diastolic (congestive) heart failure	I5033	411
Acute on chronic diastolic heart failure	42833	598

Table 1: The corresponding Diagnosis and ICD-9 and ICD-10 codes for HFpEF.

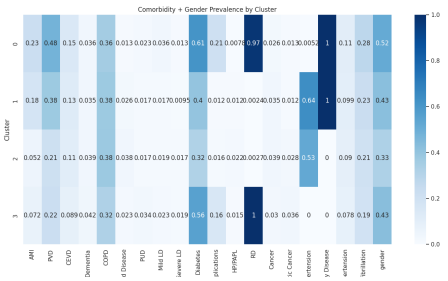
Clustering: The MIMIC data does not only include a wide range of features, but also a great number of comorbidities. Since these comorbidities can influence the chances of HFpEF, it is important to include these in the analysis. In this study, the comorbidities were used to cluster the patients, to get more specific counterfactuals. For this, the data was split according to patients having similar comorbidities to get factual cohorts. Another split was made by first dividing the data on gender and then clustering the data. Figure 3 shows the prevalence of different comorbidities of the MIMIC patients in the clusters, divided into four sub figures depending on the clustering coefficient and the gender. Figure 4 shows the Principal Component Analysis of the different comorbidities, again divided into the same subgroups.

2.3.2 Experimental Setup

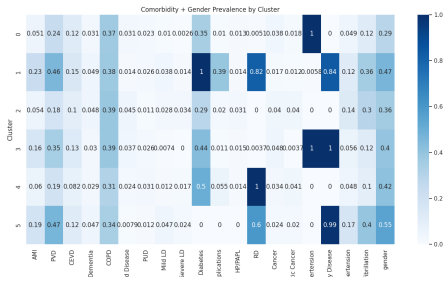
The model can be split into two main parts. Initially a multivariate forecasting model is used to make a first forecast for both the target variable and the exogenous variables. This forecast is then used for the second part, where different regression models are used to change the exogenous and target variable to get the desired outcome. For the forecasting part we used GRU and NBEATS and for the second part we used four different models. Here the focus lies on using very different kinds of models, like a statistical (SARIMAX), a regression based (OLS), and two different deep learning based (GRU and NBEATSx) models.

2.3.2.1 Multivariate Forecasting:

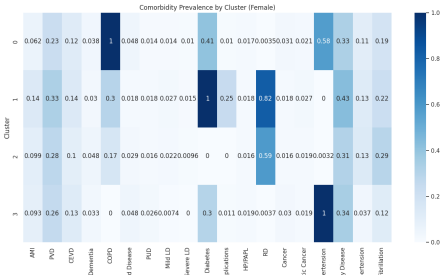
For the multivariate forecasting, two main models have been used: (1) a 2-layer GRU model with 200 units at each layer and a linear output; and (2) a 4-layer NBEATS model with a forecast and backcast layer as well as a linear output. For both models early stopping with a patience of 10 epochs was applied to prevent over-fitting, with a fixed learning rate of 0.0001. For the forecasting tasks different back horizons and forecast windows were used and analysed. To evaluate the DL models two metrics were used: symmetric mean absolute percentage error (sMAPE) and root mean squared error (RMSE), where lower scores indicated a better predictive performance. The best performing model was then used for the counterfactual generation.



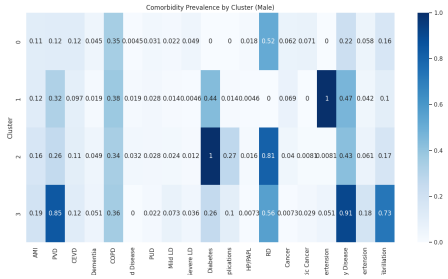
(a) Comorbidities in 4 clusters.



(b) Comorbidities in 6 clusters.



(c) Comorbidities for female patients in 4 clusters.



(d) Comorbidities for male patients in 4 clusters.

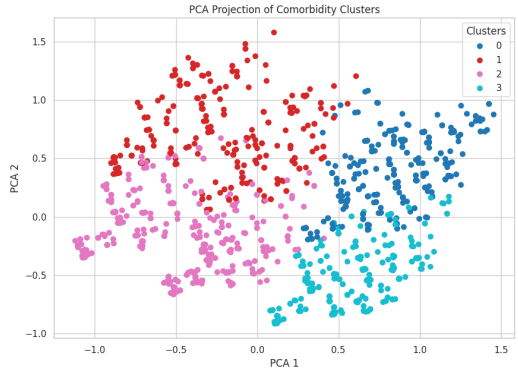
Figure 3: Prevalence of different comorbidities of the MIMIC patients in the clusters. There are 4 subgroups, clustering by comorbidity with $k = 4$, clustering by comorbidity with $k = 6$, clustering only the female patients with $k = 4$ and clustering only the male patients with $k = 4$.

2.3.2.2 Counterfactual Generation:

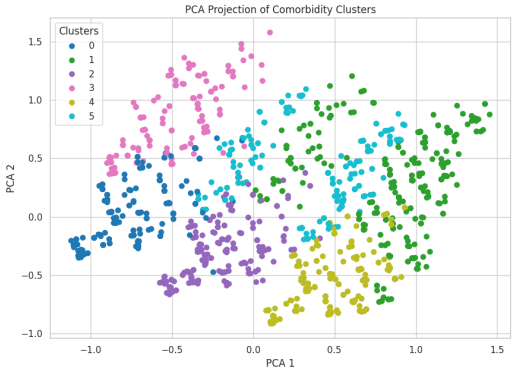
For the counterfactual generation four different models have been used, all following a similar implementation. First two subsets of test samples were selected using two thresholds. This way, the samples were divided into a hyperglycemia set (blood glucose levels ≥ 180 mg/dL) and a hypoglycemia set (blood glucose levels ≤ 70 mg/dL) and then the following procedure was used:

- Bound Generation: Polynomial-based upper and lower bounds were generated for each test sample's predicted output, targeting desired glucose thresholds within a given window.
- Gradient Perturbation: For each forecasting model, the model was inverted by optimizing the exogenous inputs via gradient descent to steer the predicted output within the generated bounds.
- Loss Function: A custom loss function was used to penalize deviation from the bounds, and gradients were approximated using automatic differentiation or finite differences depending on model compatibility.

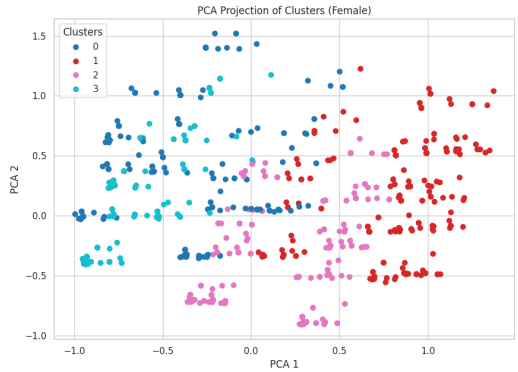
The models used for the counterfactual generation span both traditional statistical methods and deep learning architectures.
SARIMAX: The Seasonal Autoregressive Integrated Moving Average with Exogenous Variables (SARIMAX) model was employed to account for seasonality and external factors. Model



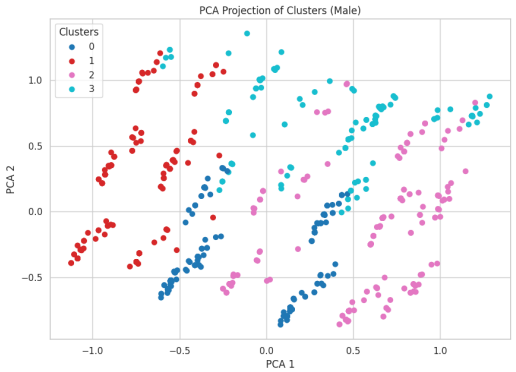
(a) PCA clustering in 4 clusters.



(b) PCA clustering in 6 clusters.



(c) PCA clustering for female patients in 4 clusters.



(d) PCA clustering for male patients in 4 clusters.

Figure 4: Principal component analysis (PCA) clustering of different comorbidities of the MIMIC patients. There are 4 subgroups, clustering by comorbidity with $k = 4$, clustering by comorbidity with $k = 6$, clustering only the female patients with $k = 4$ and clustering only the male patients with $k = 4$.

parameters, including the order of autoregression (p), differencing (d), and moving average (q), were selected based on the Akaike Information Criterion (AIC). Exogenous inputs such as insulin dosage and carbohydrate intake were incorporated as regressors, and model parameters were optimized using maximum likelihood estimation.

OLS: Ordinary Least Squares (OLS) regression was used to examine linear relationships between exogenous variables and the target series. Feature engineering involved the creation of time-lagged variables and interaction terms between insulin and carbohydrate intake. To mitigate overfitting, regularization was applied. Additionally, linear constraints on exogenous variables were enforced post-hoc to facilitate counterfactual generation.

GRU: A Gated Recurrent Unit (GRU) network was implemented. The GRU architecture consisted of two layers, each with 128 hidden units, and dropout regularization set at 0.2 to prevent overfitting. The network was trained using the Adam optimizer with an initial learning rate of 0.001, which was reduced by a factor of 0.1 if the validation loss plateaued over five epochs. Counterfactual explanations were generated by computing gradients of the output with respect to exogenous inputs, enabling identification of minimal interventions.

NBEATSx: The NBEATSx model was utilized, extending on the standard N-BEATS archi-

tecture by incorporating exogenous variables into its forecasting blocks. The model employed trend and seasonality decomposition through fully connected layers, and included exogenous inputs such as carbohydrate intake, insulin dosage, and physical activity. Quantile loss was used during training to capture distributional uncertainty, and feature importance analyses were performed to assess the contribution of exogenous variables to forecast accuracy.

2.3.3 Evaluation Metrics

To evaluate the quality of the counterfactual interventions, multiple evaluation metrics were implemented. First, traditional forecasting performance was measured using Root Mean Squared Error (RMSE) and Symmetric Mean Absolute Percentage Error (sMAPE), giving a quantitative assessment of prediction accuracy across the forecast horizon. For the evaluation of the generated counterfactuals, several additional metrics were introduced. These have been applied to ensure that the newly generated data for the exogenous variables is realistic, plausible and applicable.

RMSE, sMAPE scores sum of MSE scores, normalize for horizon?

Average value of change The average value of change quantifies the magnitude of adjustments made to the exogenous variables in order to achieve the desired forecasted outcome. For this the averaged original results of the multivariate forecast were compared to the averaged results of the counterfactual generation. euclidean distance?
The fraction of values to change captures the proportion of exogenous variable entries that required modification, offering insight into the sparsity of the interventions. By evaluating for how many of the timesteps in the time series the data changes, it is possible to analyse the differences that the counterfactuals
Severity of change To further characterize the nature of the interventions, the severity of change metric measures the relative intensity of the modifications compared to the original values. outlier score, LOF-score compare original exog to changed prediction
Fitting of predictions in bounds Additionally, the fitting of predictions within bounds was evaluated, reflecting the extent to which the counterfactual predictions adhered to the predefined upper and lower target constraints. This was done using the euclidean distance of the predicted target to the bounds. how good is the prediction, one in the band better than others?
Compare exogenous variables to those of a healthy patient Finally, to ensure the plausibility of the counterfactual scenarios, the modified exogenous variable profiles were compared to reference profiles from healthy patients. This comparison provides a qualitative and quantitative check on the biological or clinical realism of the generated counterfactual forecasts.
what are the differences in exog? changes from healthy patient

3 Results

3.1 Multivariate Forecasting

To evaluate the performance of the multivariate forecasting and to ensure a realistic basis for the counterfactual generation, two different methods for multivariate forecasting have been tested. For both the GRU and NBeats method, sMAPE and RMSE scores have been used as

a measurement for the forecasting quality. By splitting the data at a timestamp and using the data before that timestamp as a training set and the data after as a test set, it is possible to have true future values as well as the prediction. This allows for a credible assessment of the forecasters through sMAPE and RMSE scores. The results of the evaluation of the multivariate forecasting can be seen in table 3. It shows the quality of the forecast using the sMAPE and RMSE score, for the two different models and the different datasets.

Table 3: Multivariate forecasting training metrics.

Dataset	Back Horizon	Horizon	Model	sMAPE	RMSE
OhioT1DM	6	3	GRU	3.7648	9.3329
			NBEATS	**3.1741**	**9.1723**
	12	6	GRU	6.2525	14.8269
			NBEATS	**6.1397**	**14.3862**
SimGlucose	6	3	GRU	0.4699	1.9799
			NBEATS	**0.4017**	**1.6730**
	12	6	GRU	0.9970	4.3754
			NBEATS	**0.9904**	**3.6312**
	40	10	GRU	1.5436	5.6069
			NBEATS	**1.4608**	**3.0223**

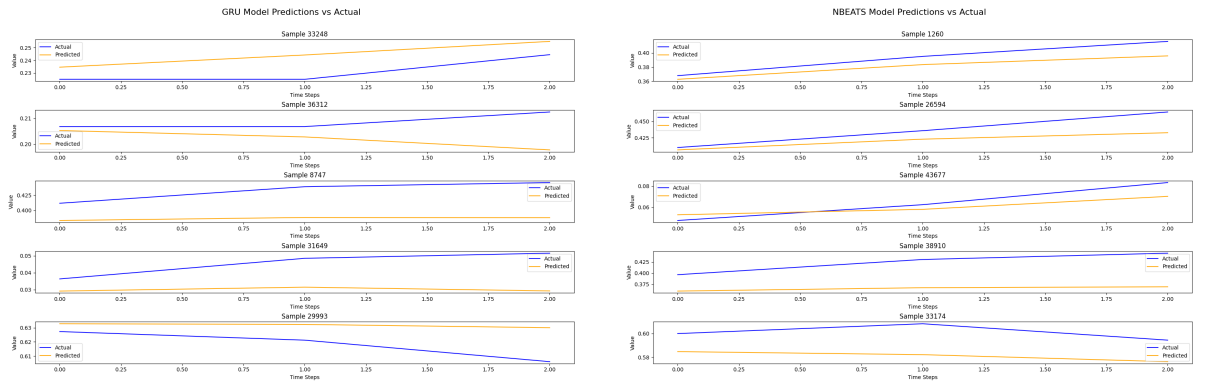
Figure 5 and Figure 6 show some corresponding examples to show the correctness of the forecasting results for the OhioT1DM dataset, while Figure 9 and Figure 7 show the same for the SimGlucose dataset.

3.1.1 OhioT1DM

3.1.2 SimGlucose

3.1.3 MIMIC

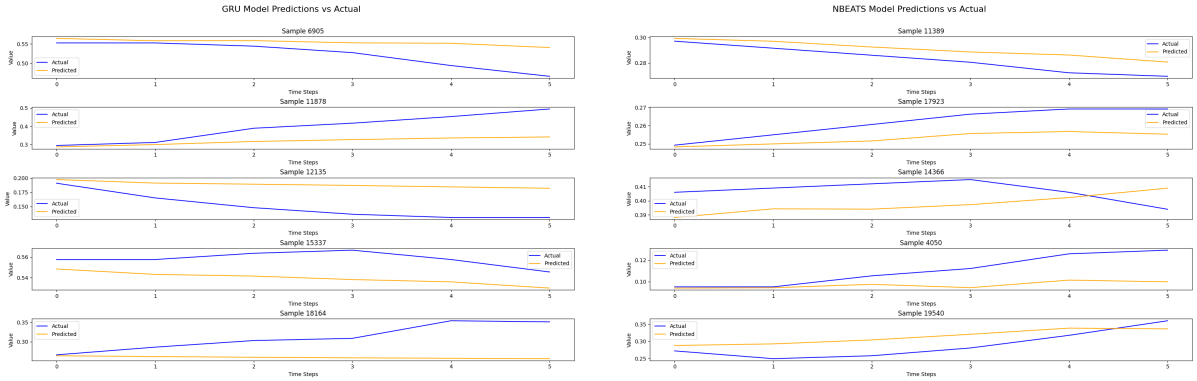
waardes checken



(a) Results of the multivariate forecasting using GRU.

(b) Results of the multivariate forecasting using NBEATS.

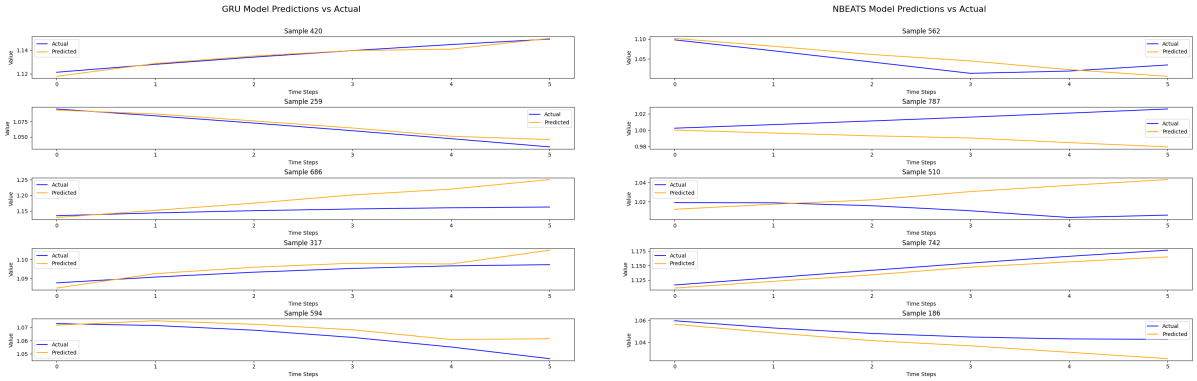
Figure 5: Results of the multivariate forecasting for the OhioT1DM dataset with back horizon = 6 and forecast horizon = 3, showing the accuracy of the forecasting.



(a) Results of the multivariate forecasting using GRU.

(b) Results of the multivariate forecasting using NBEATS.

Figure 6: Results of the multivariate forecasting for the OhioT1DM dataset with back horizon = 12 and forecast horizon = 6, showing the accuracy of the forecasting.



(a) Results of the multivariate forecasting using GRU.

(b) Results of the multivariate forecasting using NBEATS.

Figure 7: Results of the multivariate forecasting for the SimGlucose dataset with back horizon = 12 and forecast horizon = 6, showing the accuracy of the forecasting.

3.2 Counterfactuals

3.2.1 OhioT1DM

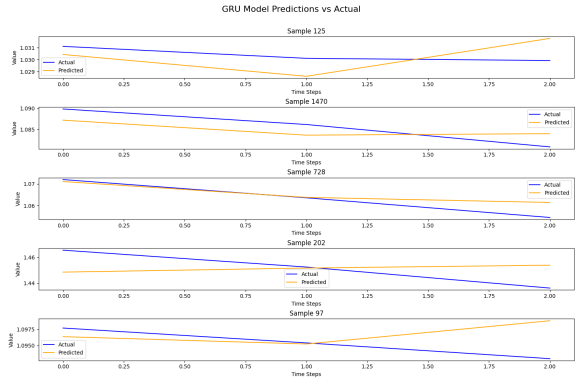
3.2.2 SimGlucose

3.2.3 MIMIC

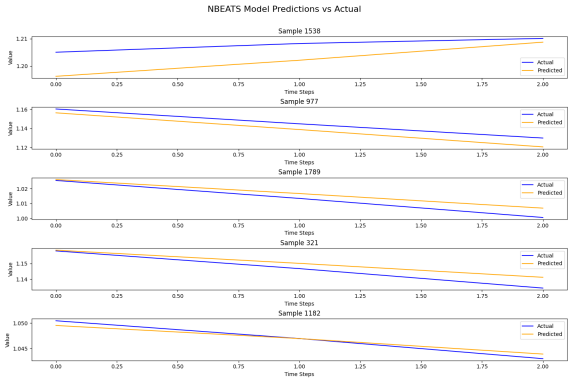
3.3 Measurements

3.3.1 Average value of change

Figure ?? shows two instances of the generated counterfactuals as well as the original exogenous data, using the OhioT1DM data.
Figure 11 shows the averaged absolute change between the original and generated time series of all four features of the OhioT1DM dataset.

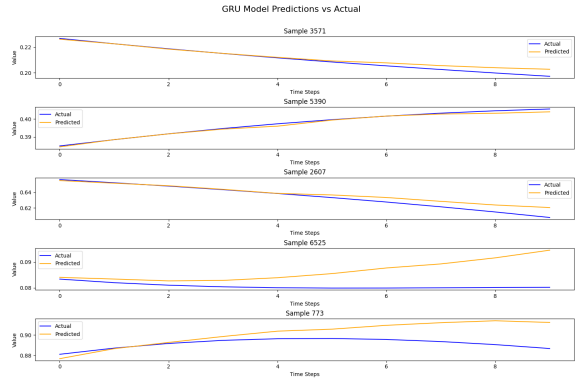


(a) Results of the multivariate forecasting using GRU.

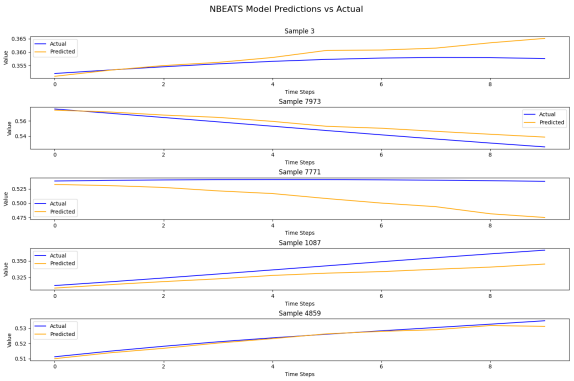


(b) Results of the multivariate forecasting using NBEATS.

Figure 8: Results of the multivariate forecasting for the SimGlucose dataset with back horizon = 6 and forecast horizon = 3, showing the accuracy of the forecasting.



(a) Results of the multivariate forecasting using GRU.



(b) Results of the multivariate forecasting using NBEATS.

Figure 9: Results of the multivariate forecasting for the SimGlucose dataset with back horizon = 40 and forecast horizon = 10, showing the accuracy of the forecasting.

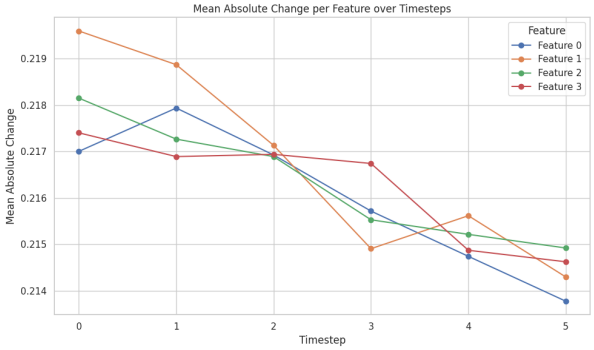
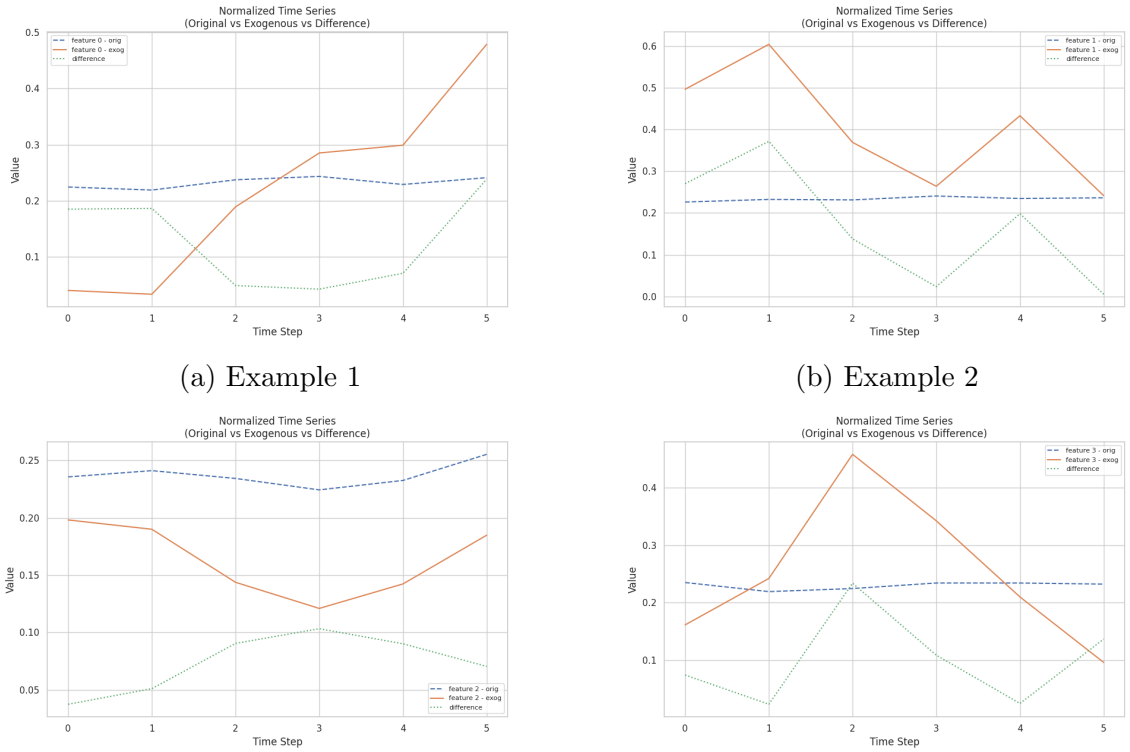


Figure 11: Caption

3.3.2 Coefficient of determination

3.3.3 Severity of change



(a) Example 1

(b) Example 2

(c) Example 2

(d) Example 2

Figure 12: Visualization of the difference between the original exogenous variables and the generated counterfactuals as well as a measured distance per feature.

3.3.4 Target prediction within bounds

3.3.5 Comparison to healthy patient

4 Discussion

4.1 Results

4.2 Limitations

4.3 Future Work

5 Conclusion

References

[AALC21] Emre Ates, Burak Aksar, Vitus J Leung, and Ayse K Coskun. Counterfactual explanations for multivariate time series. pages 1–8, 2021.

[CHN+21] Ran Cui, Chirath Hettiarachchi, Christopher J Nolan, Elena Daskalaki, and Hanna Suominen. Personalised short-term glucose prediction via recurrent self-attention network. pages 154–159, 2021.

[Com25] American Diabetes Association Professional Practice Committee. 6. glycemic goals and hypoglycemia: Standards of care in diabetes—2025. *Diabetes Care*, 48(Supplement_1):S128–S145, 2025.

[FAJ+18] Ian Fox, Lynn Ang, Mamta Jaiswal, Rodica Pop-Busui, and Jenna Wiens. Deep multi-output forecasting: Learning to accurately predict blood glucose trajectories. pages 1387–1395, 2018.

[HSYZ23] Jianing Hao, Qing Shi, Yilin Ye, and Wei Zeng. Timetuner: Diagnosing time representations for time-series forecasting with counterfactual explanations. *IEEE Transactions on Visualization and Computer Graphics*, 30(1):1183–1193, 2023.

[JPS+16] Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9, 2016.

[MB20] Cindy Marling and Razvan Bunescu. The ohiot1dm dataset for blood glucose level prediction: Update 2020. In *CEUR workshop proceedings*, volume 2675, page 71. NIH Public Access, 2020.

[MML+14] Chiara Dalla Man, Francesco Micheletto, Dayu Lv, Marc Breton, Boris Kovatchev, and Claudio Cobelli. The uva/padova type 1 diabetes simulator: new features. *Journal of diabetes science and technology*, 8(1):26–34, 2014.

[SBA+24] Ikgyu Shin, Nilay Bhatt, Alaa Alashi, Keervani Kandala, and Karthik Murugiah. Predicting 30-day and 1-year mortality in heart failure with preserved ejection fraction (hfpef). *medRxiv*, 2024.

[WMSP23] Zhendong Wang, Ioanna Miliou, Isak Samsten, and Panagiotis Papapetrou. Counterfactual explanations for time series forecasting. In *2023 IEEE International Conference on Data Mining (ICDM)*, pages 1391–1396. IEEE, 2023.

[WSMP24] Zhendong Wang, Isak Samsten, Ioanna Miliou, and Panagiotis Papapetrou. Comet: Constrained counterfactual explanations for patient glucose multivariate forecasting. In *2024 IEEE 37th International Symposium on Computer-Based Medical Systems (CBMS)*, pages 502–507. IEEE, 2024.

[Xie18] Jinyu Xie. Simglucose v0.2.1. `https://github.com/jxx123/simglucose`, 2018.

[ZLH+18] Taiyu Zhu, Kezhi Li, Pau Herrero, Jianwei Chen, and Pantelis Georgiou. A deep learning algorithm for personalized blood glucose prediction. pages 64–78, 2018.

Feature types	Specifics	Occurrences
Targets		
	Death within 30 days	133
	Death within 1 year	146
Vital signs and laboratory values		
	Heart Rate	1766
	Systolic BP	1759
	SpO2	1766
	Temperature	1754
	BMI	1766
	Bicarbonate	1749
	Creatinine	1751
	Hemoglobin	1739
	Platelet Count	1738
	WBC Count	1739
	Sodium	1751
	NT-proBNP	74
	Troponin	928
Comorbidities		
	AMI	232
	PVD	562
	CEVD	219
	Dementia	73
	COPD	654
	Rheumatoid Disease	51
	PUD	41
	Mild LD	44
	Moderate/Severe LD	29
	Diabetes	771
	Diabetes + Complications	150
	HP/PAPL	31
	RD	737
	Cancer	64
	Metastatic Cancer	47
	Hypertension	614
	Coronary Artery Disease	736
	Pulmonary Hypertension	175
	Atrial Fibrillation	409
	Prior Admission	
	Male	732
	Female	1034

Table 2: Features, targets, data types