UNIVERSITAT DE BARCELONA

Final Degree Project

Biomedical Engineering Degree

“Developing a GAN-Based Blood Glucose T1DM Outcome Prediction Model for Clinical Use“

Barcelona, 5th of June 2024

Author: Oriol Bustos Martínez

Director: Josep Vehí Casellas

Tutor: Margarita Giménez Álvarez

Abstract

This project presents the development and evaluation of a novel outcome prediction model for blood glucose levels in patients with Type 1 Diabetes Mellitus (T1DM) using a Wasserstein Conditional Generative Adversarial Network (WCGAN). This Generative Deep Learning (GDL) model is trained on the ReplaceBG dataset, which includes time series data from 226 T1DM patients, comprising Plasma Insulin (PI) administration, Rate of Appearance (RA) of carbohydrates, demographic and temporal information. The WCGAN, comprising over 3 million parameters, was trained to iteratively generate synthetic blood glucose profiles. These would allow producing long series of data in order to analyze the effect of different therapy strategies on the patient the model is mimicking, aiding clinical decisions. The training was conducted over 250 thousand steps using the GeForce RTX 4070 Ti GPU. Once adjusted, this conditional GAN dynamically generated glucose level predictions based on past and present inputs from the three aforementioned variables: insulin, carbohydrates, and time. The effectiveness of the model was tested by assessing the statistical similarity between the synthetic and real glycemic outcomes, with key metrics showing significant results. The model demonstrated physiological glucose-insulin dynamics, a causal relationship between inputs and outputs, and the possibility to control the variability of the latter modifying the latent space ($Z \in \mathbb{R}^3$) sampling. Additionally, we showed clear overlap between real and generated data distributions ($\mathcal{D}_R$ and $\mathcal{D}_G$), as well as success in reconstructing missing parts of the first. Despite this, the effect of including time showed mixed results in improving the quality of the outputs. This thesis can be seen as a proof of concept on incorporating the moment of the day into the GAN-based outcome prediction model, and further establishes its feasibility with ten-fold more data than previous work.

Keywords: Generative Deep Learning, Type 1 Diabetes, Blood Glucose Prediction, Generative Adversarial Networks, Circadian Rhythms

Acknowledgements

My sincere gratitude goes to Josep Vehí for providing me with the chance to start my work at Micelab. His leadership in the research team has equipped me with essential tools including software, hardware, data, and expert knowledge, allowing me to kickstart this project. Special acknowledgment goes to Omer Mujahid, for his invaluable guidance, and for so many never-ending discussions in the lab. The significant role played by Margarita Giménez, shedding light into the medical aspects, was also vital. Lastly, I am eternally grateful to my family for their support throughout all my journeys.

Glossary of Abbreviations

T1DM/T2DM: Type 1/2 Diabetes Mellitus
GAN: Generative Adversarial Network
CNN: Convolutional Neural Network
DNN: Deep Neural Network
CGM: Continuous Glucose Monitoring
MDI: Multiple Dose Insulin (therapy)
HCP: Healthcare Professionals
FDA: US Food and Drug Administration
EMA: European Medicines Agency
AP: Artificial Pancreas
CHO: Carbohydrate
UI: Units of Insulin
PI: Plasma Insulin
RA: Rate of Appearance
TM: Time Bin
HR: Heart Rate
BG: Blood Glucose
L2/MSE: Mean Squared Error
IOB: Insulin On Board
I_{fa}: Fast acting Insulin in plasma
I_{la}: Long acting Insulin
HbA1c: Hemoglobin A1c
ML: Machine Learning
GDL: Generative Deep Learning
GDPR: General Data Protection Regulation (UE)
TIR: Time in Range
G & D: Generator & Discriminator
LS: Latent Space

List of Figures

1 Basic structure of the Artificial Pancreas [5] . . . 2
2 Diagram of the processes involved in glucose homeostasis [12] . . . 4
3 Average hourly basal rate values by age group [25]. . . . 6
4 Diagram of the Hovorka model of the glucose-insulin dynamics [29] . . . 7
5 ['ML' AND 'diabetes'] query on Pubmed [35]. . . . 9
6 ['generative deep learning' AND 'diabetes'] query on Pubmed [36]. . . . 9
7 Diagram of basic GANs architecture [38]. . . . 10
8 Cubic splines interpolation over real data points with 1 and 10 minute sampling rates 23
9 Distribution of samples across time bins. 1: 00-6h, 2: 6-12h, 3: 12-18h, 4: 18-00h . . 26
10 Percentage distribution of samples across gender and ethnicity . . . 26
11 Percentage distribution of samples across diagnostic age, weight and height . . . 27
12 Graphical representation of the critic model . . . 27
13 Graphical representation of the generator model . . . 28
14 Graphical representation of the GAN composite model . . . 28
15 Q-Q plot on generated `coeff._variation` . . . 33

16 Q-Q plot for real `coeff._variation` . . . 33
17 AGP report using aggregated data from realtest patients . . . 34
18 AGP report using aggregated data from generated test patients . . . 35
19 Sample 248-32 . . . 35
20 Sample 234-52 . . . 35
21 Sample 250-7 . . . 36
22 Sample 239-35 . . . 36
23 Sample 250-12 . . . 36
24 Sample 239-27 . . . 36
25 Critic's loss on real samples . . . 36
26 Critic's loss generated samples . . . 36
27 Generator's overall loss . . . 37
28 Generator's adversarial loss . . . 37
29 Generator's mean square loss . . . 37
30 Convergent Cross Mapping skill, or correlation, evolution for test patients . . . 39
31 Cross mapping of PI$\rightarrow$BG 250 . . . 39
32 Cross mapping of RA$\rightarrow$BG 250 . . . 39
33 Bi-directional cross mapping of TM$\leftrightarrow$BG 250 . . . 39
34 PCA analysis (PCs=2) of of real and syntethic BG values (BG dim=2) for test patients 1-5 . . . 40
35 t-SNE analysis (PCs=2) of real and syntethic BG values (BG dim=2) for test patients 1-5 40
36 t-SNE analysis (PCs=2) of real and syntethic BG values (BG dim=2) for test patients 5-14 . . . 40
37 Visualizing the latent space Z, sampling normal and truncated Gaussian distributions 41
38 PCA and t-SNE analysis (PCs=2) of real and syntethic BG values (BG dim=2) with truncated Latent Space . . . 41
39 PCA and t-SNE analysis (PCs=2) of real and syntethic BG values (BG dim=2) with the complementary truncated Latent Space . . . 41
40 Transformation from normal to symmetrized blood glucose scale. Numerical and clinical center coincide and hypo- and hyper- ranges become symmetric [75]. . . . 46
41 WBS of the project . . . 54
42 PERT of the project with critical path outlined in red . . . 55
43 PERT tasks precedents . . . 55
44 Head of the GANTT spreadsheet . . . 55
45 GANTT diagram of the project . . . 56
46 Milestone plan of the project . . . 56
47 SWOT Analysis of the project . . . 58

48 Discriminator architecture . . . 73
49 Generator architecture . . . 74
50 GAN composite . . . 74
51 Real AGP report from subpatient 248-30 . . . 75
52 Generated AGP report from subpatient 248-30 . . . 75
53 Q-Q plot for real `mean_glucose` . . . 81
54 Q-Q plot for real `time_in_range` . . . 81
55 Q-Q plot on generated `mean_glucose` . . . 81
56 Q-Q plot for generated `time_in_range` . . . 81

List of Tables

1 T1DM datasets overview . . . 17
2 Characteristics of the selected datasets . . . 19
3 Summary of the variables that will be present in the model . . . 21
4 Summary of Model Configurations and Architectural Choices . . . 31
5 Kolmogorov-Smirnov test results for generated data . . . 32
6 Kolmogorov-Smirnov test results for real data . . . 32
7 Comparison of Metrics Between Real patient's CGM and Synthetic CGM Presented as Average or IQR (25th;75th) Percentile, with the p-value resulting from the Student's t-test (shaded), Wilcoxon Signed-Rank Test (not shaded) . . . 37
8 Count of validated metrics (p-value>0.05) using the corresponding statistical test (Student's t-test or Wilcoxon Signed Rank). Comparing real vs. synthethic data, grouped by time bins (TM=X), using data with fixed time inputs simulations (SIM-TMX) . . . 38
9 Correlation ('strength of causality') and p-value ('significance') using `causal-ccm` for aggregated patients from the test set for PI→BG . . . 38
10 Correlation ('strength of causality') and p-value ('significance') using `causal-ccm` for aggregated patients from the test set for RA→BG . . . 38
11 Correlation ('strength of causality') and p-value ('significance') using `causal-ccm` for aggregated patients from the test set for TM→BG . . . 38
12 PC Acquisition Costs . . . 60
13 Development Costs . . . 60
14 Comparison of Metrics Between synthethic patient's CGM simulated with fixed TM values, presented as Average or IQR (25th;75th) Percentile . . . 81

Contents

1 Introduction **1**
1.1 Motivation . . . 1
1.2 Objectives and Scope . . . 2
1.3 Institutions and Work Period . . . 2

2 Background: General Concepts & State Of The Art **3**
2.1 Diabetes and Glucose Homeostasis . . . 3
2.1.1 Other Factors Affecting Glucose Homeostasis . . . 5
2.2 Models for T1DM . . . 6
2.3 Artificial Intelligence . . . 8
2.3.1 Generative Adversarial Networks . . . 9

3 Market Analysis **11**
3.1 Potential Users for the Model . . . 11
3.2 Added Value . . . 12
3.3 Sectors Analysis . . . 12

4 Conception Engineering **15**
4.1 Features . . . 15
4.2 Datasets Study and Selection . . . 16
4.3 Models . . . 19
4.4 Selected Approach . . . 20

5 Detail Engineering **22**
5.1 Data Preprocessing . . . 22
5.2 Data Analysis . . . 25
5.3 Implementation of the Model . . . 26
5.4 Validation . . . 31
5.5 Testing . . . 31
5.6 Distribution of the Data . . . 33

6 Results **34**
6.1 Overview . . . 34

6.2 Validation . . . 37
6.3 Data Distribution Representation: $\mathcal{D}_R$ and $\mathcal{D}_G$ . . . 40

7 Discussion 42
7.1 Interpretation of Results . . . 42
7.1.1 Overview . . . 42
7.1.2 Validation . . . 45
7.1.3 Data Distribution Representation . . . 49
7.2 Challenges and Limitations . . . 51
7.3 Simulating in the Clinical Setting . . . 53

8 Execution Cronogram 54
8.1 WBS . . . 54
8.2 PERT & GANTT . . . 54
8.3 Milestone Plan . . . 56

9 Technical Feasibility 57
9.1 Infrastructure and Resources . . . 57
9.2 SWOT Analysis . . . 58

10 Economical Feasibility 59
10.1 Research Group Budgeting . . . 59
10.2 Project Budget . . . 59

11 Regulatory Affairs 61
11.1 Medical Device Regulations . . . 61
11.2 Data Protection and Privacy . . . 62
11.3 AI: Ethics and Regulation . . . 62
11.4 Thesis Approval, Risk Management and Responsibility . . . 63

12 Conclusions 64

13 References 67

14 Annex 73

1 Introduction

1.1 Motivation

In a world where diabetes affects approximately 420 million people globally, innovative solutions are desperately needed to improve the lives of these patients and to reduce the burden on healthcare systems. With technological advancements and the rise of artificial intelligence, we find ourselves in an unprecedented position to better the management of this disease [1].

Diabetes is more than a medical condition; it's a never-ending challenge, requiring continuous monitoring and lifestyle adjustments. It is estimated that T1DM patients take over 180 diabetes-related decisions, every day [2]. Poorly managed diabetes can lead to further complications. Thus, devising more efficient treatment strategies is not just a health necessity, but a societal imperative for improving the lives of millions globally. The focus of this project will be the development of a realistic simulator for diabetes patients,using a data-driven model. This simulator's primary purpose is to help the development of decision support algorithms for both diabetics and Healthcare Professionals (HCPs). Unlike other glucose models that focus solely on immediate prediction, such as avoiding the next hypoglycemic event, our aim is to predict and understand how changes in therapy will impact long-term outcomes for the patient. This would provide a platform for HCPs and/or patients to understand, test and refine their therapeutic strategies. With the ability to forecast the consequences of a particular therapeutic decision on the patient's metabolic parameters, HCPs can ensure they provide evidence-based and personalized recommendations. This will help to optimize glycemic control, minimize the risk of hypoglycemia and other complications, and consequently, enhance the quality of life for patients living with diabetes. Currently, HCPs primarily rely on general population, rule-based recommendations, or diabetes standards of care. However, optimal control can only be an individualized one [3].

This simulator could as well enhance the testing and training of the artificial pancreas (AP). AP systems, which combine insulin pumps and continuous glucose monitors to automatically control glucose levels, are becoming a very good alternative to the still missing cure of the disease [4]. However, their efficiency is highly dependent on their ability to adapt to individual patient's physiological responses. This is where our realistic simulator comes in, which will incorporate new additional relevant features to ensure that the model mirrors the complexity of real–life scenarios. By providing a realistic representation of the patient's glucose-insulin dynamics, the simulator can significantly improve the artificial pancreas system's capacity to tailor and generalize the therapy to the individual. A diagram showing the basic AP components can be seen in Fig. 1.

In essence, the development of this simulator could be a significant leap forward in the diabetes management field. It would not only facilitate advancements in the performance of artificial pancreas systems but would also empower HCPs to deliver superior, individualized care.

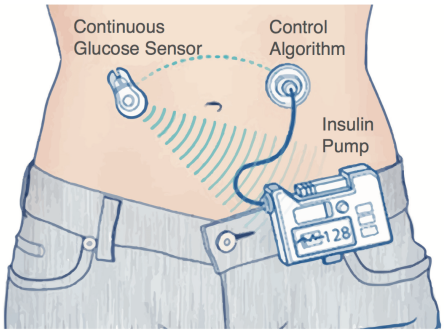

Figure 1: Basic structure of the Artificial Pancreas [5]

1.2 Objectives and Scope

We define the objectives as follows, while we discuss their conception in Section 4.

1. Develop an outcome prediction data-based model for a cohort of diabetic patients using a Generative Adversarial Network.
2. To guarantee that the generated profiles are plausible from the physiological point of view.
3. Ouputs must be caused by past and present input values of insulin, carbohydrates, and time.
4. Model the effect of circadian rhythms including the moment of the day in the architecture.

The scope of this work is to carry out an investigation process during the first semester of 2024. In that period we will develop a deep-learning blood glucose model for T1DM patients in Python, that allows the input of carbohydrates, insulin, and time features using pre-existing datasets. If successful, the conducted research will finally be presented in the form of a scientific article describing the contributions. Looking further ahead, the goal is to push the boundaries of the knowledge in diabetes modeling.

1.3 Institutions and Work Period

This research project will be conducted in collaboration with several academic and research institutions. It will mainly take place in collaboration with Micelab at the Parc Científic i Tecnològic from University of Girona (c/ Pic de Peguera, 15, 17003 Girona).

Micelab is an interdisciplinary research team from the Institute of Informatics and Applications at Universitat de Girona, headed by Prof. Josep Vehí. It is actively participating in both national and international research and transfer projects. It comprises seasoned researchers from control engineering, holding specialized knowledge in biomedical systems [6].

Guidance will be provided by Prof. Josep Vehí himself, and supervised by Dra. Marga Giménez, specialist from the Endocrinology and Nutrition service, and head of the Diabetes Unit from Hospital Universitat de Barcelona. Both will ensure a high standard of academic rigour. The main working period for the project will be between February 2024 and June 2024.

2 Background: General Concepts & State Of The Art

In this section we will explore the theoretical background of diabetes mellitus and glucose homeostasis, overview vital concepts regarding Generative Deep Learning (GDL), and review the latest relevant research developments.

2.1 Diabetes and Glucose Homeostasis

Diabetes mellitus, more commonly known as diabetes, is a metabolic disorder characterized by chronic hyperglycemia, defined as the abnormally high levels of glucose in the blood. This is typically due to impaired insulin production, ineffectiveness of the produced insulin, or a combination of both. The disease manifests in several different forms, including Type 1, Type 2, gestational diabetes and others. It was estimated that by 2021 over 537 million adults worldwide were living with diabetes, with projections foreseeing a rise to 700 million by 2045 [7].

Glucose homeostasis, the balance of the secretion of the hormones insulin and glucagon by the pancreas to maintain blood glucose, is critical for energy production and overall physiological health. This intricate process is regulated by the pancreatic islet cells, primarily the alpha and beta cells, which secrete glucagon and insulin respectively [8]. When the balance in this system is compromised, it can result in conditions like diabetes. We can better visualize the relationship between glucose, glucagon and insulin in the body with the diagram in Fig. 2.

Insulin, produced by beta cells, promotes the uptake of glucose into tissues, reduces glucose production in the liver, and suppresses the action of glucagon, an insulin antagonist. In T1DM, an autoimmune response destroys these beta cells, leading to a lack of insulin and uncontrolled glucose levels [9]. Insulin signaling pathways involve a much more complex network of proteins and regulate various physiological processes beyond glucose metabolism [10].

On the other hand, Type 2 diabetes (T2DM) is characterized by insulin resistance, a state in which tissues become less responsive to insulin. This condition encourages the beta cells to produce more insulin, causing hyperinsulinemia. Then, the alpha cells become resistant to this excess insulin, and continue to secrete glucagon, leading to further increased blood glucose levels [11].

Putting the focus on the management of T1DM, understanding the dual role of insulin is crucial. Insulin administration in diabetic care is typically separated into basal and bolus insulin. Basal insulin provides a steady, continuous dose that helps maintain glucose levels in a fasting state, mimicking the consistent low-level secretion of insulin by the pancreas in healthy individuals. Bolus insulin, on the other hand, is injected before mealtimes to counteract the sudden influx of glucose resulting from carbohydrate ingestion and digestion. Balancing these two insulin types helps maintain healthy glucose levels, although it requires careful and personalized planning.

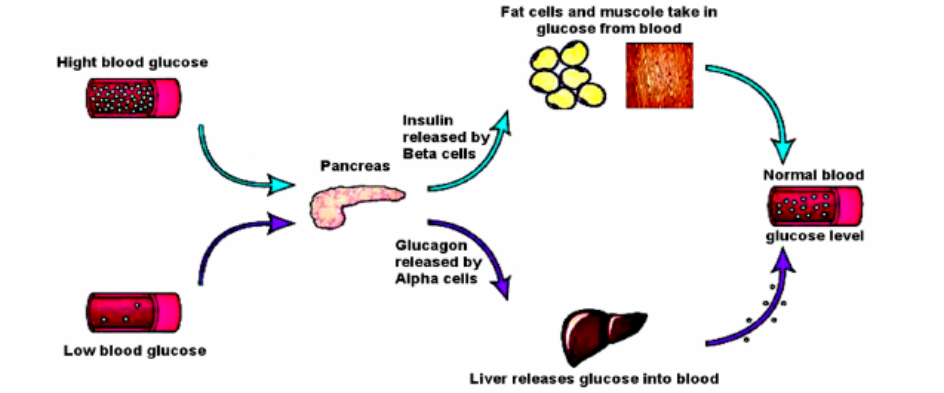

Figure 2: Diagram of the processes involved in glucose homeostasis [12]

Carbohydrates play a central role in determining blood glucose levels due to their conversion into glucose during digestion. Not all carbohydrates, however, are created equal, making each meal have its own rate of appearance (RA) in blood glucose levels. The glycemic index (GI) of a carbohydrate refers to the relative rate at which it raises blood glucose levels post digestion (postprandial). High-GI foods cause a rapid rise and fall in blood glucose, whereas low-GI foods result in a more gradual, sustained release of glucose [13].

Yet, glucose homeostasis is influenced by more than just insulin and carbohydrates. A wider variety of factors can affect glucose dynamics, including temperature, age, sex, lifestyle, exercise, circadian rhythms, stress, and hormones. For instance, environmental temperature can affect insulin sensitivity and therefore glucose levels [14]. And lifestyle factors, such as diet and physical activity, are known to modulate glucose homeostasis significantly [15].

Exercise, in particular, can lower blood glucose levels by increasing the uptake of glucose into muscle cells [16]. However, not all exercise is the same, the type of activity, intensity, and duration of the exercise can have varying effects on blood glucose levels. Similarly, circadian rhythms, defined as the 24-hour cycles that govern physiological processes, can also influence glucose metabolism [17]. Variations in the circadian rhythm can impact the effectiveness of insulin, thereby affecting glucose levels. Furthermore, stress can trigger a release of hormones such as cortisol and adrenaline, which significantly affect glucose homeostasis and increase blood glucose levels [18].

Recent advancements in the understanding of glucose homeostasis have illustrated the role of other hormones such as glucagon-like peptide-1 (GLP-1) and gastric inhibitory polypeptide (GIP) that also regulate in some sense blood glucose levels. Additionally, understanding the interplay between insulin and these hormones has led to the development of novel therapeutic strategies for diabetes [19].

As we went through the basic physiology of the studied disease and explored the state of the art in diabetes research, we have uncovered the complex web of interactions that govern glucose homeostasis.

2.1.1 Other Factors Affecting Glucose Homeostasis

Many factors affect glucose levels in blood, only the most relevant to the project are discussed here.

Exercise

The concept of physical activity as a tool for the prevention and management of diabetes dates back to the early understanding of the condition. Ancient physicians like Hippocrates and Galen often recommended a combination of dietary modifications and physical exercises to manage prominent diseases, including what we today recognize as diabetes [20]. With the pass of time these ancient observations have been scientifically validated, revealing the multiple benefits of physical activity in diabetes management. Physical activity plays a critical role in the regulation of glucose metabolism. Regular physical exercise enhances insulin sensitivity, improves glycemic control, and has been associated with a reduced risk of complications related to diabetes such as heart disease and stroke [21]. It also aids in maintaining optimal body weight, a key aspect in preventing and controlling T2D.

Circadian Rhythms

Although exercise has long been known to play a big part in glucose dynamics, the role of circadian rhythms in diabetes has only recently come to the fore, as research into chronobiology, the study of biological rhythms, has advanced. The circadian rhythm, 24h cycles regulated by endogenous molecular oscillators called the circadian clock, prepare the body for events that take place throughout the day, including food intake and energy expenditure. It is known to regulate a wide variety of physiological processes, including metabolism. Disruptions in this rhythm, such as shift work or sleep disturbances, have been linked to an increased risk of developing metabolic disorders [22] [23]. The molecular mechanisms underlying these effects are still being investigated, but evidence points towards a complex interplay between clock genes, metabolic genes, and environmental factors like light exposure and meal timing. The concept of 'chrono-nutrition', which considers not just what we eat but when we eat, is an emerging field that could offer novel dietary strategies for diabetes management. If we want to accurately model glucose dynamics, we need to open the model's eyes and allow it to see a representation of these cycles, maybe leading to it being able to find the correlations we observe in real life.

It has been studied and clinically demonstrated that insulin secretion varies during the day, which in turn proves variable insulin sensitivity. These are responsible for phenomenon such as the dawn effect. The dawn phenomenon refers to an early morning increase in blood glucose levels, typically between 2 AM and 8 AM. In T1DM it is primarily caused by a decrease in insulin sensitivity at dawn, rather than changes in insulin clearance or insulin action. Nocturnal spikes of growth hormone secretion have been shown to be responsible for the decreased insulin sensitivity at dawn, affecting both the liver

and peripheral tissues [24]. Schneiner et al. [25] studied the hourly insulin sensitivity variability in several patients, showing basal insulin peaks tend to occur during the early morning hours (Fig. 3) with variations in magnitude and duration across different age groups. These higher insulin requirements align with the supposition that insulin sensitivity drops during these periods.

Overall, understanding the role of both physical activity and circadian rhythms influence have proven indispensable in the fight against diabetes. The exploration of circadian rhythms in diabetes is relatively novel but is rapidly uncovering new insights on its effect on metabolism and pushing the discovery of new potential therapeutic strategies. By combining these two essential players, we can hope to better diabetes models and thus improve the quality of life for patients.

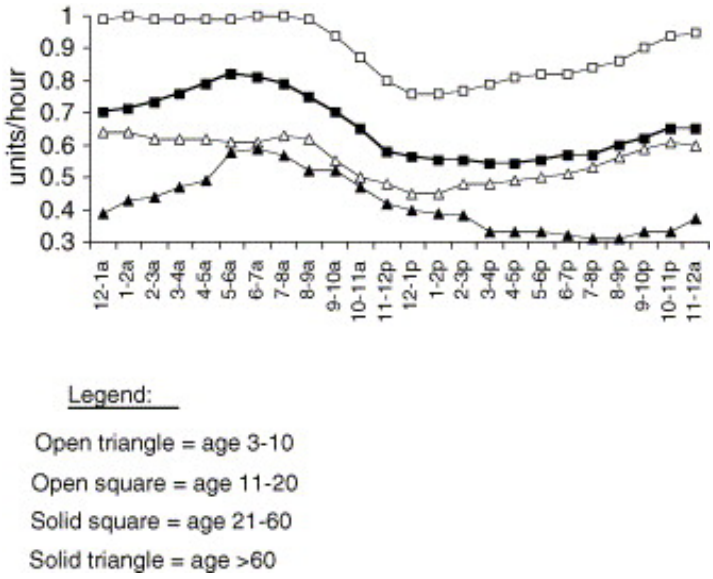

Figure 3: Average hourly basal rate values by age group [25].

2.2 Models for T1DM

The scientific journey towards developing robust models for T1DM has a rich history, marked by the contributions of numerous researchers. The early stages of diabetes research can be traced back to the 1920s, when Banting and Best's discovery of insulin marked the first major breakthrough [26]. Their work provided fundamental knowledge about the role of insulin in glucose metabolism and laid the groundwork for all subsequent diabetes models.

One of the earliest influential models was the minimal model, introduced by Richard N. Bergman in the late 1970s. This model, as described in Bergman's paper 'Toward Physiological Understanding of Glucose Tolerance' [27] was an attempt to understand glucose metabolism and insulin kinetics in the human body. The minimal model uses frequent glucose tolerance tests data to estimate insulin

sensitivity and glucose effectiveness. Although simple, the minimal model was key in advancing the understanding of glucose metabolism.

In the 2000s, Roman Hovorka's work [28] significantly advanced the field of diabetes modeling. His work was notable for its physiological relevance and complexity, incorporating carbohydrate absorption, insulin action, and glucose utilization. Hovorka's model was particularly pivotal in the development of closed-loop insulin delivery systems or artificial pancreases. Fig. 4 is provided as a visual example of the compartments that form these types of models.

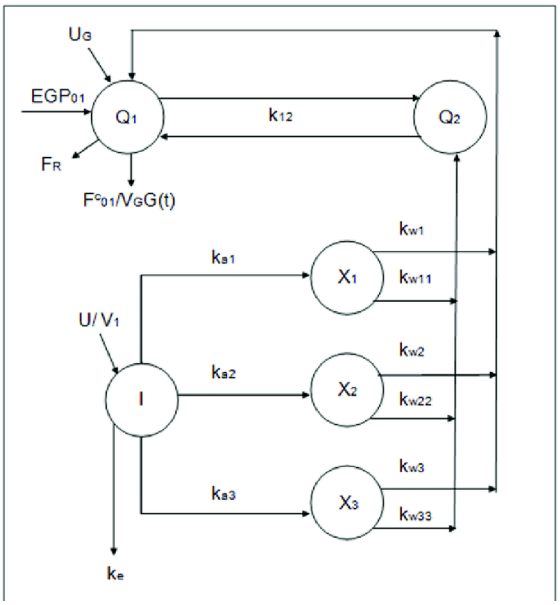

Figure 4: Diagram of the Hovorka model of the glucose-insulin dynamics [29]

In the 2010s, there was a major shift towards using patient-specific data for modeling. In 2008 the University of Virginia, and Dalla Man and her team at the University of Padova developed the 'UVA/-Padova Simulator' [30]. The virtual patients from the simulator, were approved by the FDA. The in silico model mimics glucose-insulin dynamics in patients with type 1 diabetes. It integrates a meal model, exercise model, and insulin administration model, providing a comprehensive simulation environment that has been extensively used for testing diabetes management strategies, particularly the development and testing of AP algorithms.

Recently, in the 2020s, AI and machine learning algorithms have started to be employed for predicting and managing T1DM. Van Doorn et al. [31] demonstrated that machine learning could effectively predict glucose values using continuous glucose monitoring data, further personalizing diabetes management. Such models, like the one proposed by Mujahid et al. [32], utilize vast amounts of patient data to create predictive algorithms for glucose levels, insulin needs, or even the risk of developing complications. This approach has the potential to revolutionize the way diabetes is managed, making

it more personalized, predictive, and preventive.

These developments have collectively advanced our understanding of diabetes and paved the way for more personalized and efficient management of the disease. The progress in modeling and simulating type 1 diabetes illustrates the power of combining physiological insights with computational modeling techniques, promising exciting advancements in the future of diabetes care and research.

As a disclaimer, it is important to note that although in this project the terms 'model' and 'simulator' are sometimes used quite interchangeably, they represent slightly different things: the first being a simplified, abstract representation of a specific system or process, while the second refers to a tool or environment that mimics the behavior or actions of that system or process. The reason for this interchangeability is that a simulator is built upon a model (in our case just one, but several can be used). In essence, the model forms the theoretical basis for the simulator.

2.3 Artificial Intelligence

Artificial intelligence (AI) has really changed the game in many fields, and healthcare is a big one. We're seeing AI being routinely used for tasks like disease prediction, risk assessment of patients, amongst many other use cases. The key is that AI can look at huge amounts of information and find patterns that not even experts can spot. As we turn our attention to deep learning, its use in health care is becoming more popular by the day. This is due to the increasing availability of complex biomedical data, such as electronic health records, imaging, genomics, sensor data, and text, which are heterogeneous, poorly annotated, and unstructured. Traditional data mining and statistical learning approaches require feature engineering to obtain effective features from the data before building prediction or clustering models. The latest advances in deep learning provide effective ways to train end-to-end learning models from complex data, without feature engineering. Deep learning approaches have the potential to transform healthcare, by enabling the understanding of those big biomedical data into improved personalized human health. There is still a need for improved methods development and applications, and more notably data [33]. Generative deep learning in particular is showing to be a potentially great tool. Generative models can help create synthetic patient data, or simulate various scenarios that patients might encounter, helping to predict how changes in factors like diet, exercise, medication, and other health conditions might impact blood glucose levels. Not only that, the task of generating data is very different from discriminating it, requiring a much more in-depth and robust understanding of its behavior, dynamics and overall probabilistic distribution. Think about it like this, imagine the difference in required expertise between executing the three tasks of: discriminating a chocolate cake from a cheese cake, rating how good each is, or actually cooking them yourself.

Overall, AI methods are being increasingly established as suitable for use in clinical daily practice and self-management of diabetes. The integration of new technologies, such as Continuous Glucose

Monitors (CGM) devices and the development of the AP, along with the exploitation of data acquired by applying these novel tools, have completely transformed the entire paradigm of diabetes management [34].

In Figs. 5, 6 we can see the scientific publications in Pubmed that include the terms Machine Learning or Generative Deep Learning in combination with Diabetes. It can be clearly noted how deep learning is a much more recent term, while AI already has a long history of scientific work.

Figure 5: ['ML' AND 'diabetes'] query on Pubmed [35].

Figure 6: ['generative deep learning' AND 'diabetes'] query on Pubmed [36].

2.3.1 Generative Adversarial Networks

Generative Adversarial Networks (GANs) are a class of deep learning frameworks designed initially by Goodfellow et al. in 2014 [37]. GANs have been an integral part of the deep learning revolution, providing an innovative approach to generating new, synthetic instances of data that can pass for real ones.

GANs consist of two Neural Networks (NNs): a generator and a discriminator. The generator's goal is to produce data samples that seem as close as possible to the original dataset instances, while the discriminator aims to distinguish the generator's fake forgeries from the real examples. The loss is used as a measure of how well the models are performing and fed back to allow improvement. These two components are trained simultaneously, engaging in a two-player adversarial game, hence the name Generative Adversarial Networks. An explanatory diagram of the system can be seen in Fig. 7

Variations

Over the years, the GAN architecture has evolved and been extended, resulting in a number of variants that have been developed to address some of the challenges and limitations of the original model. These variants include Deep Convolutional GANs (DCGANs)[39], which use convolutional layers in their networks to improve the quality of generated images, Conditional GANs (CGANs), and Wasserstein

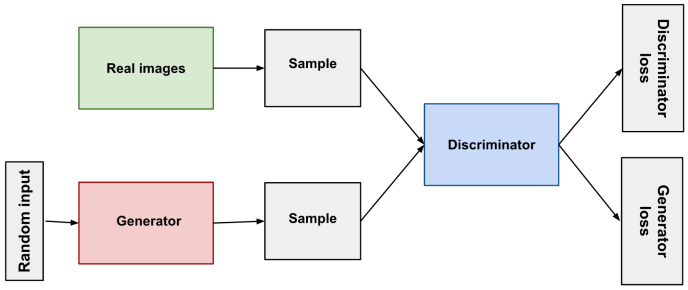

Figure 7: Diagram of basic GANs architecture [38].

GANs (WGANs) [40], which use a different type of objective function to stabilize the training process and avoid a very common problem: mode collapse.

CGANs, introduced in the paper 'Conditional Generative Adversarial Nets' by Mirza et al.[41] in 2014, represent an extension of the original GAN concept. Unlike traditional GANs that generate data from random noise directly, CGANs condition the generation process on certain types of additional information, like a class label or another image. This information is fed to both the generator and discriminator as input, thereby providing control over the types of samples that are generated.

In the current landscape, GANs have been applied to a wide range of domains, from creating realistic human faces to synthesizing novel pharmaceutical molecules. They have also been used in semi-supervised learning, where they leverage both labeled and unlabeled data to improve learning accuracy. Despite their complexity and the challenges associated with training them, GANs have undoubtedly established themselves as a powerful tool in the deep learning toolkit and a vital part of state-of-the-art GDL technology.

Our model will be fundamentally grounded in improving the work of Mujahid et al.[42] specifically his research using CGANs to generate blood glucose profiles conditioned on insulin and carbohydrate inputs. In this context, Mujahid introduces an innovative solution that overcomes the limitations of current diabetes simulators. His proposed CGAN-based system is capable of learning the intricate distribution of data produced by biological systems associated with diabetes. It can then generate synthetic profiles of virtual diabetes patients that closely mimic the complexities of real-life scenarios. The generative model was shown to produce statistically similar blood glucose values and causal relationships between inputs, and the output blood glucose. The goal here is to go a step further including new inputs that can improve the simulator's performance, as well exploring changes in the design, whilst training on a much bigger dataset that allows a more robust validation of the method.

3 Market Analysis

Here we will try to comprehend the current market scenario, trends and target audience. Modeling in diabetis is usually seen as a tool towards the development of bigger concepts, such as the AP, but other use cases for this technology and their current and historical demand will be explored.

3.1 Potential Users for the Model

Potential sectors that could use a simulator for T1DM span across healthcare, education, research, and technology industries. This simulator could be a great asset for the healthtech industry:

Clinicians

- **Pre-Treatment Testing**: HCPs could simulate potential therapies for their patients, personalizing treatment plans and reducing trial-and-error in clinical settings.
- **Patient Education**: Companies like AMIC are already implementing an educative approach for kids [43].

Medical device developers and academic researchers

- **Artificial Pancreas Development**: The model can be employed in the training and development process for APs. It can offer insights into the physiological mechanisms of diabetes, supporting better design and optimization of devices.
- **Testing and Validation**: The model can be used as a test bed for new devices before they reach clinical trial stages, allowing reductions in development time and costs.

Patients

- **Understanding Personal Physiology**: By interacting with the diabetes model, patients gain a deeper understanding of their own condition. This can encourage them to take proactive steps in their self-management.
- **Therapy Exploration**: With this model, patients have a tool to visualize and explore the impact of different therapies, fostering discussions about treatment options with their healthcare providers.

Healthtech companies

- **Health Tracking Applications**: These companies can integrate the model into their software to provide users with real-time feedback on their health status and lifestyle choices. The model can enhance the apps' predictive and analytical features.

Educational institutions

- **Teaching and research**: The model can be used as a teaching aid in medical and health-related courses, offering students a deeper understanding of diabetes and its management.

3.2 Added Value

Non-AI algorithms can struggle to capture the complexity of diabetes-related data, often leading to oversimplified models that might miss crucial aspects and scenarios of the disease. Within AI algorithms, accurately modeling complex systems will only be achievable with deep learning techniques. This is due to their unique capacity to learn intricate patterns directly from large-scale raw data. Among deep learning models, CGANs stand out for their ability to effectively capture the intricate, high-dimensional data distributions, in our case representative of the complex dynamics of glucose and insulin.

In this project the CGANs developed by Mujahid et al. [42] will be further improved by adding efficiently selected features that will allow the model to account for circadian rhythms. As more quality datasets become available, more conditions can be added to the digital twin. Looking into the future, our CGAN-based simulator could become a valuable tool in diabetes research and treatment. It could lead to a better understanding of the disease, realistically simulating the effects of a therapy without putting the patient at risk and aiding the development of more effective therapies and medical devices.

3.3 Sectors Analysis

The Artificial Pancreas Market

There is a growing interest in the medical community towards improving the management of diabetes. And this push is evident in the rise of companies focusing on the development of an AP closed loop system. Especially the ultimate version, the fully automated closed-loop insulin delivery pump, with no intervention needed from the patient. As overviewed in 1.1, the AP is a diabetes control system composed of a sensor, a controller and an actuator. Success in developing such a system requires expertise in building its core components: the CGM, the controller and the insulin pump. And finally training it accordingly in a simulation environment, which is where this project comes in handy. So we'll study the market for diabetes technologies, and especially that of the APs, as demand for it, may result in demand of simulators.

- Historical evolution

Historically, the market evolution of diabetes and personalized treatments has followed the larger trends in medicine. In the earlier part of the 20th century, treatment and management of diabetes were predominantly generalized and one-size-fits-all. The discovery of insulin in the 1920s revolutionized the diabetes market, becoming a lifesaver for many, especially T1DM. As the century progressed, pharmaceutical advancements led to the development of various types of insulin, oral medications for T2DM, and improved testing supplies. However, the personalization of diabetes care truly developed in the 21st century with the rise of digital health and data analytics.

The rise of CGM and insulin pumps allowed tailored insulin dosing and the feasibility of the AP. This concept of an AP emerged about 50 years ago, with the possibility of external blood glucose regulation established through intravenous glucose measurement and insulin infusion. The initial developments

in this field by several teams during the seventies led to the first commercial device, the Biostator, in 1977. However, these intravenous methods were unsuitable for daily patient use, yet they proved the feasibility of external glucose control, encouraging further technological advancements. By 1979, it was proved that the subcutaneous route was feasible for continuous insulin delivery, the insulin pump system, paving the way for the development of wearable AP prototypes. A significant advancement came with the introduction of the MiniMed CGM system in 1999, which led to a surge in academic and industrial efforts focused on developing a fully subcutaneous system for fully automated glucose control [44]. In the early 2000s, technological advancements and increased focus on personalized medicine spurred the development of subcutaneous methods for insulin delivery and glucose measurement. This marked the beginning of a new era in the development of the AP. Companies like Medtronic emerged as early pioneers with the introduction of their first ”sensor-augmented” insulin pump in 2006, which integrated a CGM with an insulin pump [45]. The years following saw incremental advancements towards a fully closed-loop system. In 2013, Medtronic launched the world's first semi-closed loop system, known as the MiniMed 530G, which could automatically suspend insulin delivery when sensor glucose values fall below a certain threshold [46]. This represented an important step in improving the safety of insulin pump therapy. The real breakthrough, however, came in 2016, when Medtronic obtained FDA approval for their MiniMed 670G system, the world's first hybrid closed-loop system. This system automates basal insulin delivery to maximize the time glucose levels are within the target range, but still requires user input for mealtime insulin [47]. In recent years, the hybrid system has established as the most advanced control system widely available in the market, proving its efficacy over normal AID systems in patients of all ages in places like Europe [48]. Finally, the fully automated systems pose a final significant challenge, as no input from the patient means being able to detect meals just from glucose levels, which means control actions will always happen late.

- Current Market Analysis and Main Competitors

The current market for artificial pancreas device systems has evolved considerably, with the global market expected to grow at a compound annual growth rate (CAGR) of 20.34% from 2023 to 2031. The market was valued at $406.76 million in 2022 and is anticipated to reach $1235.19 million by 2028 [49]. The market is segmented based on product types: Threshold Suspend Device System, Control-to-Range (CTR) System, and Control-to-Target (CTT) System, and end-user applications, which include clinicians and hospitals, among others. Some of the prominent corporations working on developing a closed-loop control system for glucose, like Medtronic and Insulet Corp. have been working extensively on AP systems for years. Medtronic's current MiniMed system, is a hybrid closed-loop system that automatically adjusts insulin levels based on sensor readings of blood glucose levels [50]. Insulet's Omnipod Horizon, meanwhile, uses a unique tubeless design and is working towards integrating with CGMs to create a fully automated insulin delivery mechanism [51].

HealthTech Apps Market

The market for HealthTech apps, has experienced substantial growth over the past decade. This includes a wide variety of functionalities, from telemedicine platforms and electronic health records to wearables. Apps that use simulators for patient dynamics would be categorized under this market segment. According to a recent report, the global digital health market size was valued at approximately $227.5 billion in 2022, and is projected to reach around $872.2 billion by 2031, exhibiting a CAGR of 16.1% [52]. The increase in smartphone adoption, heightened health and fitness awareness, and a rise in chronic conditions are some of the major drivers behind this impressive growth. In this context, the market for apps that simulate patient dynamics is anticipated to grow significantly. Such applications have the potential to play a vital role in enhancing treatment plan effectiveness, aiding medical research, and improving patient care quality. A simulation environment can assist healthcare professionals in predicting disease progression, understanding patient responses to treatments, and potentially reducing healthcare costs by optimizing treatment strategies [52].

Some of the main competitors in this market include Dexcom, mySugr, and One Drop. 'mySugr', a subsidiary of Roche, provides an all-around diabetes management platform, with features including food and insulin tracking and a bolus calculator. It also integrates with certain glucose monitors and insulin pumps for automatic data import.

Decision Support Systems

The market for clinical decision support systems (CDSS) represents another promising prospect for simulation demand. A desktop program that incorporates an easy-to-use and fast simulation environment for diabetes care can be highly beneficial for HCPs who must make daily recommendations to their patients. This system could aid experienced clinicians in delivering evidence-based treatment suggestions, enhancing the safety and efficacy of the therapy for the end user. The CDSS market is projected to experience substantial growth in the upcoming years. As previously mentioned, these systems are designed to assist healthcare providers in decision-making. A simulation environment can form a crucial part of such a system, offering clinicians a robust and reliable tool to predict patient responses to treatment and modify their strategies accordingly. According to a report by Expert Market Research, the global clinical decision support systems market size was valued at $1.6 billion in 2022 and is expected to grow at a CAGR of 9.8% to around $3.8 billion by 2031 [53]. The factors driving this growth include the escalating demand for quality healthcare and the adoption of healthcare IT solutions.

Some major competitors include ex-IBM's Watson Health and Google's Verily. Watson Health has been applying its AI technology to a range of healthcare scenarios, including diabetes management. Their system uses cognitive computing to analyze patient data and clinical research, delivering personalized treatment recommendations for healthcare providers. Verily, under its Project Baseline, is using ML algorithms to predict diabetes-related complications before they happen.

4 Conception Engineering

We will now delve into the possible approaches to our problem: creating a glucose model for a cohort of T1D diabetic patients that generates BG profiles iteratively, based on current states like insulin, food and moment of the day. We will formulate initial ideas, select the best suited ones and shape them into a feasible project strategy.

4.1 Features

Unquestionably, insulin and carbohydrates will be included as inputs for the model. They are essential when generating any type of BG data, datasets without them will be immediately discarded. Besides insulin and CHO, we also mentioned how, ideally, we would want the model to also account for physical exercise and circadian rhythms when predicting future blood glucose values. We will explore available variables that represent this phenomenon and select those we consider to have more potential. An important consideration is that due to time and economic constraints, no clinical trial will be conducted to recollect the needed data, so the project will have to rely on already existing and available datasets, thus limited to the use of the features collected in those previous studies.

Exercise

There are many variables which can be considered to represent physical activity or in some way condition it's effect on glucose-insulin dynamics like heart rate (HR), energy expenditure, skin temperature, heart rate variability (HRV), breathing rate, steps, distance, floors climbed etc. The first thing we have to consider is that many of these features, would not even be feasible for continuous monitoring. Furthermore, given no clinical trial will be done, only features present in publicly available datasets can be used. Also, we have to take into account that given the complexity of deep learning, we must include the least possible inputs that give the model the maximum amount of information. So we must avoid using redundant, colinear, or comparatively less informative variables.

Circadian rhythms

Features related to circadian rhythms that in turn affect glucose homeostasis could include: hour of the day, personal habits, hormonal release, eating and sleeping patterns, stress, genetic factors, seasonal changes and many more. Similarly, many of these variables are not easily measurable continuously. Moreover, their causal relationship with blood glucose may be inferred, but its mechanisms are unclear, possibly being overshadowed by the effect of CHO and insulin. So to keep a realistic approach, only the **hour of the day** will be used. This variable is perfect, as it is as easy as monitoring a variable gets, it is already available in all datasets, so no new studies would need to be conducted to train the model. Also, most importantly, it gives the model the ability to find correlations between glucose dynamics and time of the day. Sort of like 'opening its eyes' to the daily cycle. Until now, the scenario looked like a

very long unique day that initiated at the start of the simulation and finished at its end. But now, it can see how there is a 24 hour cycle within which some patterns may appear. It may be able to account for patient routine, meal patterns or common diabetes phenomenon like the dawn phenomenon [54], if any of these are present in the patient. It is important to note how obviously the utility of this parameter will be impaired in patients with inconsistent routines. Also, when training population cohorts instead of individual patients the interpretation will be very different, as individual routines will be averaged out and only common patterns will be captured and even strengthened, like those underlying of the disease like the dawn phenomenon, or those characteristic of a certain region's habits.

Plasma Insulin & Rate of Absorption (of CHO)

In the datasets we will see, insulin (UI or UI/hour for either MDI or CSII) and carb (grams) values are discrete, and must be converted into continuous curves. For insulin, we first must understand two different concepts, Insulin On Board (IOB) and Plasma Insulin (PI). Both IOB and PI are considered "active" insulins, but they refer to different aspects of insulin activity and are used in different contexts. IOB is a term often used in the context of insulin pump therapy and it refers to the amount of insulin that has been administered (either by injection or by an insulin pump) and is still in the body. Divided into two, the S1 compartment includes insulin that is still in the subcutaneous tissue and has not yet entered the bloodstream, and S2, insulin that is in the bloodstream. $IOB = S1 + S2$, is used to prevent "insulin stacking", the accumulation of insulin from multiple doses that can lead to hypoglycemia. On the other hand, PI refers specifically to the amount of insulin that is in the bloodstream and is available to act on tissues to lower blood glucose. It is a more direct measure of the insulin that is currently active, in the sense that it is able to exert its glucose-lowering effect, reason why PI will be the variable used in the model. In other words, all PI is IOB, but not all IOB is PI. Some of the IOB is still in the subcutaneous tissue and has not yet entered the bloodstream to become PI. The distinction between IOB and PI is important in the context of insulin kinetics and the modeling of insulin action in the body. The equations for PI are shown in Subsection 5.1.

4.2 Datasets Study and Selection

An overview of the found datasets is shown in Table 1. This will be data we will try to model. We are interested in publicly available datasets, understand what type of features they contain, how many patients signed up, for how long were they monitored, and the quality of the information. Based on that we will make the final decision on how the model will be designed.

Table 1: T1DM datasets overview

Datasets	Duration (days)	Patients	Availability	CGM	Food/Exercise/Insulin
HypoMin [55]	85	10	Priv.	✓	✓/✓/✓
Chinese [56]	3-14	12	✓	✓	✓/✓/✓
Replace BG [57]	∼220	226	✓	✓	✓/✓/✓
UVA/Padova [58]	cust.	30	✓	✓	✓/×/✓
OhioT1DM [59]	56	12	Cred.	✓	✓/✓/✓
ABC4D [60]	180	10	×	✓	✓/×/✓
KDD18 [61]	1095	40	✓	✓	×/×/×
Yang [62]	1–7	49	×	✓	(Na)/(Na)/(Na)
D1NAMO [63]	4	9	✓	✓	✓/×/×
DiaTrend [64]	509	51	Cred.	✓	✓/×/✓
T1DEXI [65]	30	497	On-wait	✓	✓/✓/✓

At the moment, there is no diabetes dataset containing HRV data. T2DM diabetes datasets were not included in this summary, like in the case for Maastricht study of Maryland. From this first overview we can already discard some cohorts. Firstly, the UVA/Padova as it is simulated data, and we want to validate the results with real data, which is significantly more challenging to replicate. Next, for the Advanced Bolus Calculator for Diabetes (ABC4D) data, the primary source for the development and evaluation of this group appears to be the Imperial College of London, but we were unable to get access to it, and the same for Yang. Regarding D1NAMO, according to the information provided in the article, it includes a myriad of features, but there is no mention of insulin data in the dataset. Next, although KDD18 has a great amount of data with over 550k blood glucose instances, neither does it not contain any of the features we need. Finally, T1DEXI is a huge dataset with many features apart from insulin, food and CGM. However, we are still waiting to get access to it. We are left with 5 possible datasets which are going to discuss in more detail below.

REPLACE-BG

The dataset is based on a 26-week trial carried out in 2015 including 226 individuals with T1D. The core of the study was to exhibit the reliability of CGM systems in managing blood glucose levels. All participants were using an insulin pump. The participants were adults with a mean age of 44 $\pm$ 14 years, all of whom had been diagnosed with T1D for at least one year (mean duration 24 $\pm$ 12 years). They were all on insulin pump therapy and had an HbA1c level of $<$9.0% (mean 7.0 $\pm$ 0.7%). Prior to the study, 47% of the participants were already CGM users. The dataset includes CGM data over a span of 26 weeks. The data was segmented into two groups: one with participants using only CGM (n = 149), and the other with participants using both CGM and BGM (n = 77). The primary outcome metric was the ”time in range” (70–180 mg/dL) over the 26-week trial period. It also contains insulin data in the form of bolus and carbohydrate intake, but not basal rates from the pump. The REPLACE-BG study

[1]The UVA/Padova patients are virtual patients from their FDA-approved cohort.

[2]Of CGM, with insulin pump only 152.

was a randomized clinical trial that aimed to determine whether the use of CGM without Blood Glucose Measurement (BGM) is as safe and effective as using CGM with BGM in adults with well-controlled T1D. The participants were randomly assigned to either the CGM-only or CGM+BGM group, with a 2:1 ratio. The CGM usage averaged 6.7 $\pm$ 0.5 and 6.8 $\pm$ 0.4 days/week in the CGM-only and CGM+BGM groups, respectively, over the 26-week trial period.

Chinese Dataset

The Shanghai T1DM dataset is part of a larger initiative to foster data-driven machine learning in the domain of diabetes research, particularly within the Chinese demographic. It contains CGM data spread across 3 to 14 days for 12 T1DM patients. Some patients have multiple CGM recording periods due to different hospital visits, which are stored in separate excel tables within the dataset. In total, having 8 patients with 1 period of CGM recording and 2 patients having 3 periods of CGM recording. The age group of the T1DM patients is around 57.8 $\pm$ 11.1 years. Most of these patients (10 out of 12) are classified under a subtype of T1DM known as "latent autoimmune diabetes in adults", characterized by slow autoimmune β-cell destruction and an older mean age at the onset of diabetes. The gender distribution is such that women constitute 58.3% of the T1DM group. The dataset encapsulates a range of information including CGM data recorded every 15 minutes, capillary blood glucose (CBG) values, blood ketone levels, self-reported dietary intake, insulin doses, and non-insulin hypoglycemic agents. Insulin administration data covers CSII using an insulin pump, MDI with an insulin pen, and intravenous insulin infusion in cases of extremely high BG levels. This dataset has the highest quality of all those found as there are no missing values and provides very detailed meal information. However, it is small and meal composition requires significant processing.

OHIOT1DM Dataset

This dataset comprises CGM readings, insulin doses, meal carbohydrate intakes, HR measurements, self-reported life-event data amongst other features from 12 CSII patients over approximately 8 weeks. Specifically, some individuals were documented to be using Medtronic 530G insulin pumps, which are capable of delivering both basal and bolus insulin doses to manage blood glucose levels. The Medtronic Enlite CGM sensors were used by some participants to continuously monitor their glucose levels. Problems with the dataset include that only half of the patients actually wore a biometrics tracker wristband and thus have heart rate data. Plus there is a significant amount of missing values in all features.

HypoMin

The Hypomin clinical trial at Hospital Clinic Barcelona involved 10 adults with T1DM on MDI therapy, using the Freestyle Libre CGM system from Abbott for 12 weeks. They monitored glucose (CGM, BG), carbohydrates, insulin (fast and slow acting), heart rate, steps, calories, and sleep. Inclusion criteria required a history of frequent hypoglycemia or hypoglycemia unawareness, with a T1DM duration of over

5 years and an HbA1C between 6.5% and 9.5%. Exclusion criteria included recent ketoacidosis, severe comorbidities, or mental conditions. Participants also tracked exercise, illnesses, and disturbances at home, alongside physiological data using a Fitbit Alta HR wristband.

DiaTrend

The DiaTrend dataset comprises longitudinal data from 54 T1D patients, including 27,561 days of CGM and 8,220 days of insulin pump data. This dataset is designed to support diabetes research. Participants are aged 19-74, with a gender distribution of 17 males and 37 females. Insulin pump data logs basal and bolus doses, carbohydrate intake, and other settings. It is important to note certain limitations of the DiaTrend dataset. There is an imbalance in the representation of subjects across the dimensions of race, gender, and age, with a majority being non-Hispanic White/Caucasian and a lower representation of males compared to females. Additionally, the dataset lacks full temporal alignment in the CGM and insulin pump data for each participant, and various forms of missing data are present, which might limit some research efforts. Despite these limitations, DiaTrend is one of the largest open-source datasets in the diabetes domain.

Table 2: Characteristics of the selected datasets

Datasets	Days	Therapy	HR/Cal.	Food	Characteristics
HypoMin [55]	∼850	MDI	✓/✓	✓	Hypo-prone
Chinese [56]	∼150	Various	×/×	Qual.	Chinese
Replace BG [57]	∼40k	CSII	×/×	✓	General
OhioT1DM [59]	∼700	CSII	Some	✓	Half HR/Cal
DiaTrend [64]	27.6k	CSII	×/×	✓	High HbA1C

4.3 Models

Modelling blood glucose concentration in type 1 diabetic patients can be performed via two primary techniques: physiological models and data-driven models. Physiological models leverage a predefined set of rules and equations which reflect our understanding of human biology and metabolism. These models have been crucial in advancing our understanding of diabetes, but their limitation lies in their inherent assumption-based nature. They cannot account for the numerous variables that may affect a person's response to the disease, and thus, may miss critical information unique to each individual.

On the other hand, data-driven models possess the potential to overcome these limitations. They are able to capture the complex, multifactorial nature of diabetes by learning from large amounts of patient data, thereby encapsulating a more realistic distribution of patient responses. This not only allows them to make more precise predictions but also helps in uncovering hidden patterns that can advance our understanding of the disease. Within data-driven models, we have two prevalent approaches: traditional machine learning (ML) and deep learning (DL). While ML has been useful in predicting diabetic

conditions, its performance is often constrained by its inability to model highly complex relationships within data. Deep learning, however, with its multi-layered neural networks, has the potential to represent complex non-linear relationships in the data, through the combination of layers with non-linear activation functions. Therefore, they can more accurately capture the true distribution of individual patient responses.

Deep learning itself has a subset of models that are particularly promising for diabetes modelling: generative models. Specifically Large Language Models (LLMs) with Transformers, Generative Adversarial Networks and autoregressive models. ConditionalGANs are a variant of GANs models which can generate synthetic data, conditioned on certain inputs, making them ideal for scenario testing and sensitivity analysis. They can mimic the real distribution of patient data and provide meaningful insights into the progression of the disease under different conditions.

Autoregressive models, on the other hand, make future outcomes predictions in a sequential manner, based on previous data points. This approach is designed to capture temporal dependencies in patient data, such as how a patient's glucose levels change over time, and provide personalized disease progression forecasts. Unlike some other models, they are able to capture the temporal aspect of diabetes progression. Compared to GANs they could offer a less data-demanding option at the cost of not learning the data distribution as accurately.

All in all, for the most realistic and comprehensive modeling of diabetic patients, generative deep learning methods like CGANs and autoregressive models provide promising solutions. These models can not only learn the complex and highly individualistic nature of diabetes but also generate actionable predictions to aid in personalized patient care. Finally, given the extensive experience with CGANs inside Micelab, that with enough data it can better mimic complex distributions, and that comparatively less work has been done with them in diabetes research, **CGANs** will be the selected algorithm for this project. The specific configuration, loss function and overall design will be adressed in the section 5.

4.4 Selected Approach

So, recapitulating, type 1 diabetic patients data will be used to train a Conditional Generative Adversarial Network, a type of Generative Deep Learning algorithm composed of a generator and a discriminator. The generator generates synthetic data samples, while the discriminator distinguishes between real and fake data. Through iterative training, the generator improves its ability to generate realistic data, while the discriminator becomes more adept at identifying fake data. The model will generate blood glucose profiles conditioned to a combination of input variables including: Plasma Insulin, CHO and Time. Regarding data, only one dataset must be selected, provided we aim to make a populational model, so mixing patients from different studies should be avoided. Therefore, given the problems presented by other data collections, and due to its large size, with 226 CSII patients, ReplaceBG will be the selected dataset with which the model will be trained. It does not include heart rate so only

circadian rhythms will be included in the model. This will be done through the inclusion of encoded time bins. A single integer number from 0 to 3 will be input through the time input branch, representing each of the four 6 hour periods that form a 24-hour day. The selection of this 6-hour window comes from the fact that it could allow distinction of typical day partitions (night, morning, afternoon and evening), as well as phenomena like the dawn phenomena. Also, suggested in discussion with clinicians, and from reviewieng literature. E.g. Cabrera et al. [66], use 6 hour windows in their probabilistic prediction models for T1DM.

Input data will be passed onto the model as a set of vectors, and the output will be a single vector. This output vector will forecast the blood glucose profile for the decided prediction window. The insulin vector's values will be computed taking into account the rate of absorption of previous rapid and slow insulin boluses. A similar approach will be used for CHO, calculating the rate of appearance. 10 minutes sampling frequency will be used for blood glucose (measured in $\frac{mg}{dL}$), Plasma Insulin (UI) and Rate of Appearence of carbohydrates (g). Typically, the standard in the field is a 5-minute interval; however, given the not-so-fast dynamics of glucose fluctuations, doubling the sampling interval to 10 minutes allows for a reduction in data volume. This adjustment facilitates faster training of GANs without significantly compromising the resolution of vital data. This strategic reduction balances computational efficiency and the necessity to capture essential metabolic changes, and at the same time acting as a smoothing filter. Finally, no clustering of patients will be done provided this would entail a specific preprocessing, model design, tuning and training for each of the clusters, which is beyond the scope of the project, which aims at proving the concept of including time of the day variability on a general population, with significantly more data than Mujahid et al. [42] used for their GAN.

Table 3: Summary of the variables that will be present in the model

Variable	Unit	Sampling Rate
BG	mg/dL	10 min
Insulin	IU	10 min
Carbohydrates	grams	10 min
Time of the day	Category (0-3)	6 h

To evaluate the synthetic blood glucose curves, we will not focus on point-to-point accuracy. Therefore, we won't be using metrics like MARD, which are not suitable for our purpose. Instead, our analysis will center on metrics like Times in Range that assess the overall patterns and statistical properties of glucose levels over extended periods and scenarios. This approach aims to generate curves that statistically reflect a patient's typical glucose behavior. This is important as we have to acknowledge that each newly generated curve will be different due to the stochastic nature of the model. This method ensures we capture the essential variability and trends in blood glucose, rather than exact point-by-point replication.

5 Detail Engineering

In the detail engineering phase of our project, we aim at transforming our theoretical ideas into a tangible results and functional model. The key parts include preprocessing and analyzing the data before designing and implementing the model. After the GAN has been trained we will tune the hyperparameters by simulating new curves with new unseen data, and test it.

5.1 Data Preprocessing

Here, several important problems are tackled regarding temporal issues, duplicate entries, uneven spacing between measures, missing data, and outliers among other challenges. We also convert the insulin and carbohydrates discrete inputs into continuous curves from where we can sample from. Finally, we normalize, pack, and split the data, saving it for training, validation and testing.

Temporal Issues

Many issues regarding the ReplaceBG dataset arised from the time-related columns. Firstly, we merged all of them into a single 'Time' column in 'YYYY-MM-DD HH-MM-00' format, rounded to the nearest minute. Using the difference in minutes between rows, a classification is done into duplications, close readings (<5min difference, for being the theoretical sampling rate), and time gaps (>10min). These gaps are not missing values in the CGM column, but rather entire sequences of time with missing rows. Most duplicated or close readings were found to be caused by the input of insulin or meal information, which instead of being included in the current CGM read, they were put in a newly created row. This was solved by merging the information into a single row with all the available values. Gaps were subclassified into large and small gaps, with a arbitrary threshold to distinguish between them of 60 minutes. If Patient[n] had a large gap, it would be split into Subpatient[n.1] with the data from the start until the start of the gap, and Subpatient[n.2] from the end of the gap until the next one. k gaps will result in $k + 1$ subpatients. Small gaps will be filled with empty evenly spaced rows for later recompletion with the splines interpolation. The 60 minute threshold was selected to avoid splitting the dataset into too many small parts, as well as to ensure a good enough reconstruction of the curves.

Missing Values

Now that no time jumps are found in the data, complete blood glucose data needs to be ensured. With the same criteria as before, sequences with missing CGM data longer than an hour were further splitted, as having no temporal gaps does not guarantee these rows had non-zero blood glucose measurements. Here, outliers were also detected and corrected using neighboring values, as well as discontinuities, e.g.: some subpatients showed curves that would hover around 50 mg/dl and suddenly move to 240 mg/dl in a single time step. This would confuse the model so these cases are splitted too. Zero sequences at the tails (start and end of all columns) are deleted to make the future job of splines interpolation easier.

After so many splits, subpatients shorter than 3 days samples are filtered. Further filtering is done by deleting subpatients with a percentage of 0 values in their CGM greater than 10%.

Missing blood glucose series shorter than 60 minutes were imputed using cubic splines interpolation. The purpose of this is two-fold, repopulating missing data and resampling the instances from the original's 5 minute sample rate into the goal of 10 minute sample rate. A resampling example can be seen in Fig. 8, where the newly picked 10 minutes sampling rate still represents the data accurately. As it can be seen, splines automatically discard the two missing values around the 800 and 1000 minutes marks, without skewing or significantly modifying the output curve.

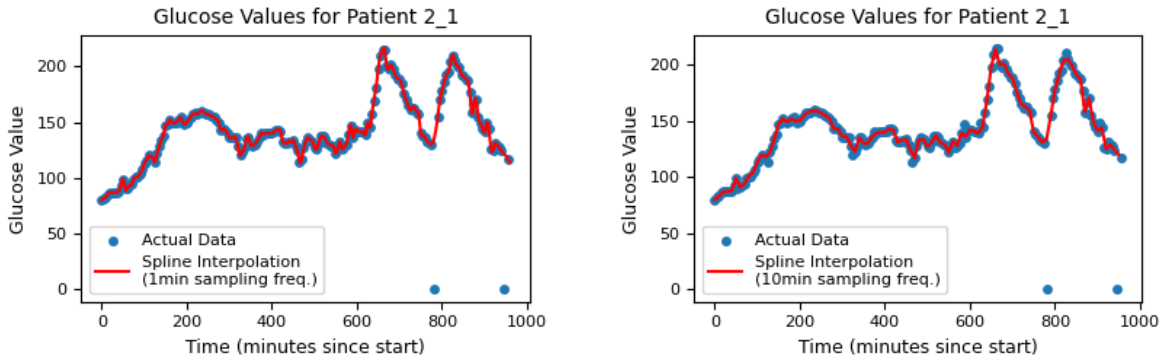

Figure 8: Cubic splines interpolation over real data points with 1 and 10 minute sampling rates

Plasma Insulin

To compute PI in CSII, the therapy followed by the ReplaceBG patients, all injected insulin, bolus and basal rate from the pump is of rapid-acting profile, so all of it can be computed using the I_{fa} compartment of the original Hovorka model seen in 4.[67] This requires first solving the system of differential equations below. After obtaining S2, we can use it to compute I_{fa}.

$$\frac{dS1(t)}{dt} = u(t) - K_{dia} \cdot S1(t) \tag{1}$$

$$\frac{dS2(t)}{dt} = K_{dia} \cdot (S1(t) - S2(t)) \tag{2}$$

$$\frac{dI_{fa}(t)}{dt} = \frac{S_2(t)}{T_{maxI} \cdot V_I} - k_e I_{fa}(t) \tag{3}$$

u(t) - Represents the rate of insulin administration and has units of (Units Insulin/min).
S1(t), **S2(t)** - Represent insulin in different compartments (UI)
K$_{dia}$ - The rate of insulin disappearance from the compartments in (min^{-1}).
I$_{fa}(t)$ - Insulin concentration in its active form (PI). Units of (UI/(L*kg)).
T$_{maxI}$ - Represents the maximum duration of insulin action and has units of minutes.
V_I - Represents the volume of distribution of insulin per unit weight and has units of (L/kg).

$\mathbf{k}_e$ - The elimination rate constant for insulin from the active form and has units of (min^{-1}).

The system is solved using 'odeint' for the boluses in the data, but there is a major issue: there is no basal rates column. Therefore, it will be estimated. Contained in the demographics file, the weight and sex of each patient are found. These are used to compute the individual's estimated Total Daily Insulin Requirement. After that, 40% is considered to be for basal requirements. That way we obtain an informed guess on the daily basal rate, with which we can easily compute the hourly and 5-minute rate from the pump of each subject.

Rate of Absorption

Analogous to PI, the RA curve will be obtained by solving a system of differential equations defined by the Hovorka model. (Equations 4, 5, 6 are Eq. 174, 175, 176 in [67]). Before solving these systems, patients with fewer than 2 inputs of carbohydrates or 2 insulin boluses per day were filtered out.

$$\frac{dD1(t)}{dt} = A_G \cdot D(t) - \frac{1}{\tau_D} \cdot D1(t) \tag{4}$$

$$\frac{dD2(t)}{dt} = \frac{1}{\tau_D} \cdot D1(t) - \frac{1}{\tau_D} \cdot D2(t) \tag{5}$$

$$RA(t) = \frac{D_2(t)}{\tau_D} \tag{6}$$

Min-max Scaling

In GAN architectures, consistent scaling of input data is vital for the model to effectively learn the distribution of data conditioned. This uniformity in scale helps the model in training more efficiently. To align with the output range of the hyperbolic tangent activation function, which goes from -1 to 1, the MinMaxScaler is employed to adjust the data accordingly. This scaling is applied to each column of the dataset, ensuring that the 'BG', 'PI', and 'RA' features are rescaled to have minimum and maximum values of -1 and 1. By doing so, the variability of the data is maintained across a consistent scale, improving the GAN model's efficacy.

Initial tests with the model often led to two seemingly unrelated phenomena. First, occasional mode collapses when training. Secondly, when successful in tuning the model, it would output profiles at different scales. The size difference seemed to be related to the range of values of the inputs. These, in the original data have variable sizes as each patient uses different insulin doses in their therapy, or eats a certain amount of carbohydrates per day. When doing MinMax Scaling the standard way, the range of the inputs is squeezed into a small range, but in a global manner. This means the highest PI value from all the patients will be +1, and the minimum -1. This essentially keeps the difference in input scales between the patients. However, as we train for a single populational model, the generator learns a single relationship between inputs and outputs. Thus, this kept inputs scale breach, will affect the consistency of the outputs. For these reasons, it was decided to deviate from the conventional

method and scale the data on a patient-wise basis. Actually, this approach was used by Lin et al. [68] in their DoppelGANger architecture. They called the process of individually normalizing each time-series: 'auto-normalization' and found to generate good results, tackling mode collapse.

Pack and Split

Data has been processed almost completely, preparation for feeding it into the model is the only remaining challenge. There will be a 2-hour prediction horizon following an input of 4 PI and 4 RA samples, spaced every 20 minutes starting at t_{-60}, plus the current time bin (a single integer from 1 to 4). Packing a subpatient's data this way results in $n-(\frac{120min}{10min})-(4-1) = n-(12)-(3)$ packs, given the start needs to be at position 4 in order to have past data to build from, and end at position $n-12$ so that the training samples can contain BG data 2 hours into the future. This way, now the concept of patients or subpatients disappears, and samples of $4+4+1=9$ conditions and 12 targets, are complete training examples ready to be input into the model. Thus, sampling can be done randomly from the pool of instances obtained. It is important to understand that even though PI and RA are sampled every 20 minutes in these packs, in the data they are still separated by 10 minutes of time, so that when the next sample pack is constructed at t_{+10}, the necessary data to build the block is available. The final step is to split the packed data into training, validation and test, which is done in a 80/10/10 proportion.

5.2 Data Analysis

The analysis is aimed at providing an overview of the dataset, focusing on understanding its structure and assessing its balance. It was preferred to perform it after the data had been cleaned and preprocessed, rather than directly on the raw dataset. This step is important to understand the contents of the data and highlight any specific areas of interest or concern that might impact the final results. The ReplaceBG study was carried out from May 2015 to March 2016 with adults who had been living with type 1 diabetes (T1D) for over a year. These participants were all using insulin pumps and had Hemoglobin A1C (HbA1c) levels of 9.0% (75 mmol/mol) or below, maintained good hypoglycemia awareness, and had not experienced any severe hypoglycemic events recently. Initially, participants used a Dexcom G4 Platinum CGM device without access to the data (a blinded period) before being divided into two groups. One group was assigned to manage their insulin based on real-time CGM data alone, while the other used both real-time CGM data and self-monitoring blood glucose (SMBG) readings for a duration of 26 weeks. Data from CGM and SMBG were collected for all individuals in the study. Both cohorts utilized SMBG readings to calibrate their CGM devices, following the guidelines provided by the device manufacturer. The study included 226 participants in total. The median age of participants was 43 years, with an interquartile range from 31 to 55 years, and the HbA1c levels had a mean of 7.0 $\pm$ 0.7% [69].

The total number of instances across all patients before preprocessing is 14,162,318, whereas the total instances after preprocessing, adding training, validation and testing equals: 3,759,199 + 2 *

469,899 = 4,698,997. More than a third of the instances have been wiped off during the preprocessing, but there are still a vast amount of training examples to work with.

Regarding the balance of data, time bins are equally distributed as seen in Fig. 9. As aforementioned, insulin and carbohydrate input data balance is ensured filtering out patients with less than two inputs per day of each of these two features. Further balance analysis is done using demographics. As it can be seen in Fig. 10, there is a balance in gender distribution but not in ethnicity or race. In Fig. 11, close to normal distributions can be observed in weight and height, and skewed normal distribution in diagnostic age. This analysis ensures a well balanced dataset is being used for all characteristics but race and ethnicity, which will need to be accounted for if it's used in the future with non-white caucasian people.

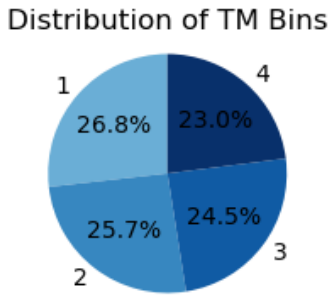

Figure 9: Distribution of samples across time bins. 1: 00-6h, 2: 6-12h, 3: 12-18h, 4: 18-00h

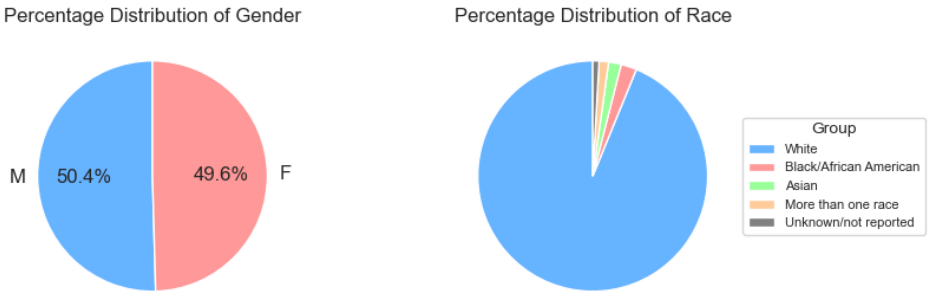

Figure 10: Percentage distribution of samples across gender and ethnicity

5.3 Implementation of the Model

This section delves into the details for the design and implementation of the Wasserstein Conditional Generative Adversarial Network (WGAN) aimed at predicting blood glucose levels based on carbohydrate intake, insulin administration and moment of the day.

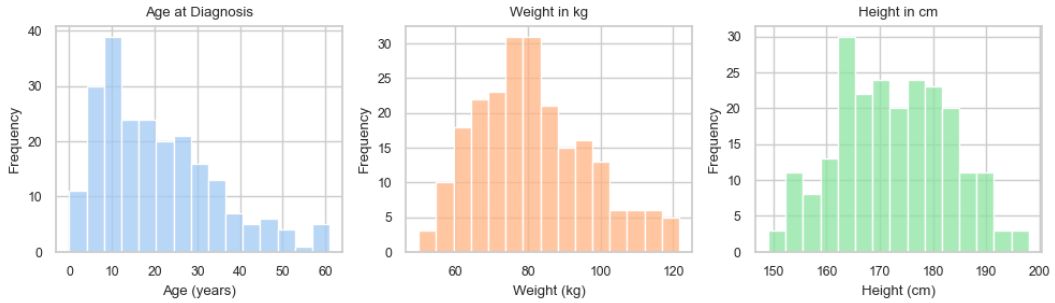

Figure 11: Percentage distribution of samples across diagnostic age, weight and height

Model Architecture

The model's architecture is adapted from the Pix2Pix framework, utilizing a Seq2Seq approach to establish a conditional structure that captures the causal relationships within the data. The architecture is divided into two main components: the generator and the discriminator. A schematic illustration of each can be seen in Figs. 12, 13, along with an overall view of the GAN in Fig. 14. A more in-depth description of the models is presented in Annex 1's Figs. 48, 49, 50.

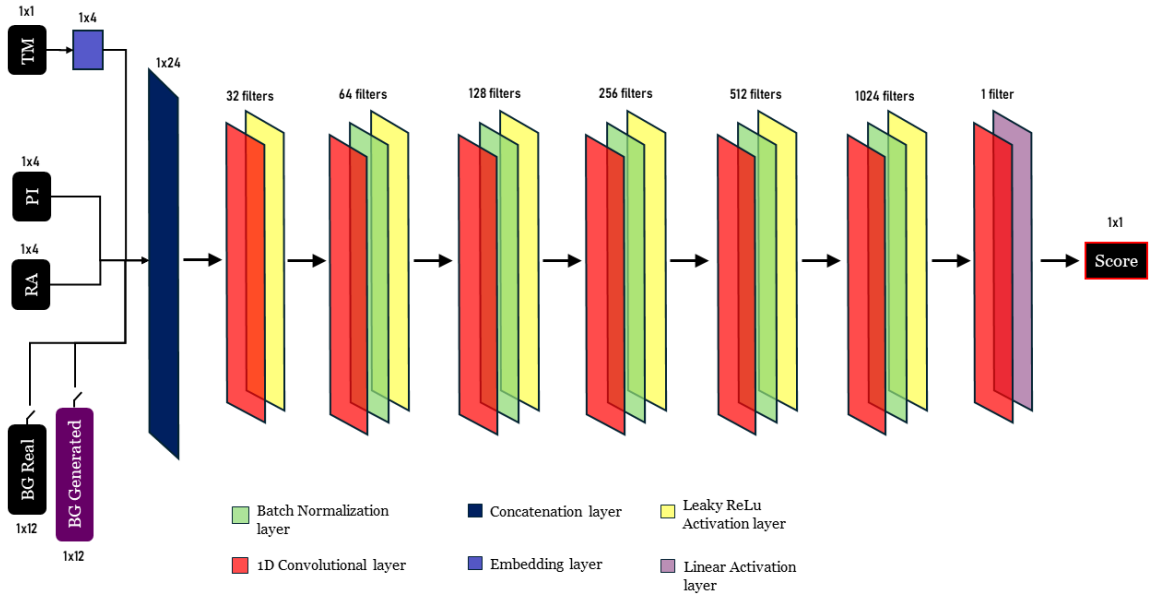

Figure 12: Graphical representation of the critic model

Discriminator The sigmoid activation from the final layer of the traditional discriminator is removed, so that predictions no longer fall within the 0 to 1 range. That means now it is functioning as a critic rather than a discriminator, using a linear activation function whose result is a score of the 'realness' of the sample, which can be in the range $[-\infty, \infty]$. It is built with multiple 1D convolutional layers

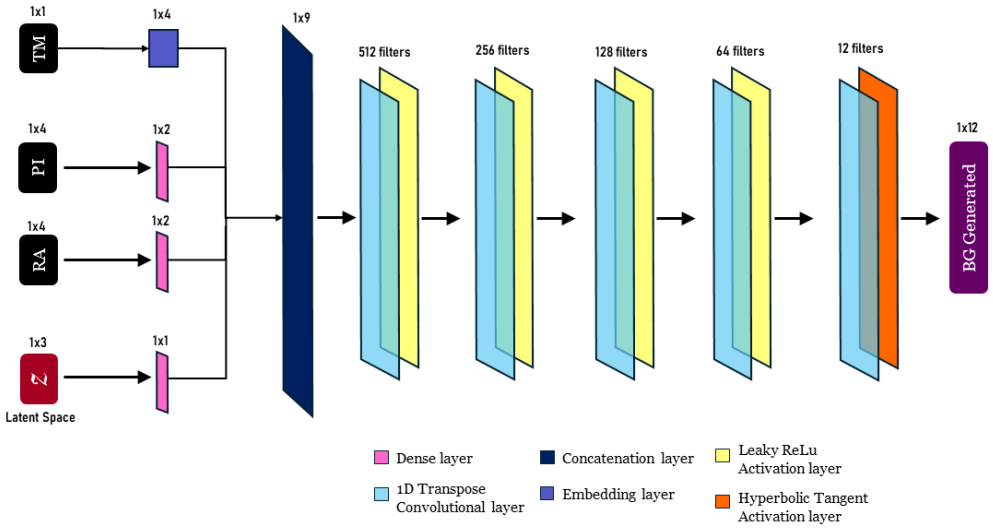

Figure 13: Graphical representation of the generator model

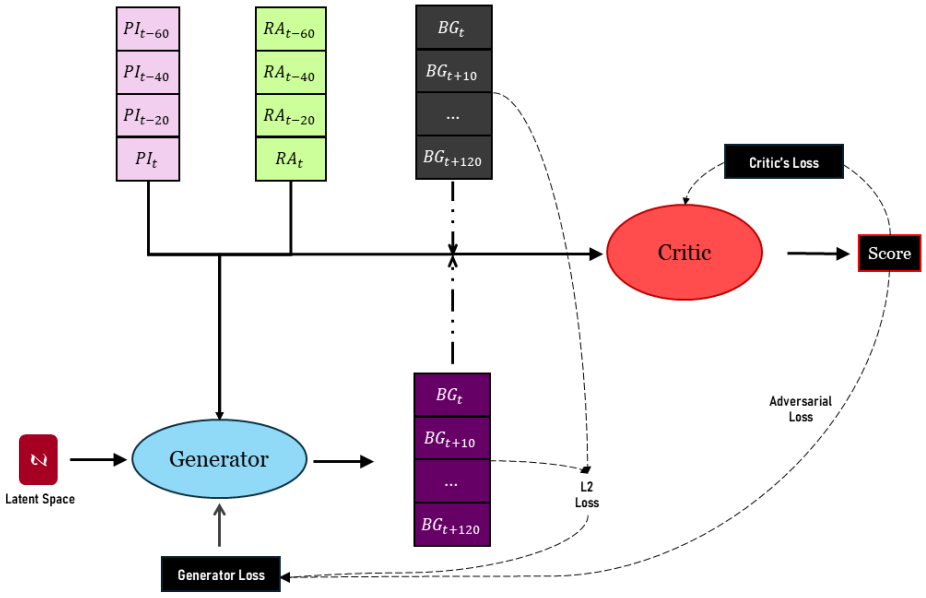

Figure 14: Graphical representation of the GAN composite model

to process temporal sequences, enabling it to guide the generator towards producing realistic glucose value predictions without direct access to glucose data. This network uses both Leaky ReLU activation function at the output of each hidden layer and batch normalization after that. The first one is a key

component to avoid mode collapse, and the second normalizes the output of a previous activation layer by subtracting the batch mean and dividing by the batch standard deviation.

Generator The generator samples from a three-dimensional latent space (LS) that follows a standard normal distribution, to encourage the variability of the outcomes does too. It is conditioned on insulin and carbohydrate intake: $RA_{t-60min}$, $PI_{t-60min}$ - $RA_{t-40min}$, $PI_{t-40min}$ - $RA_{t-20min}$, $PI_{t-20min}$ - RA_t, PI_t, plus the current time bin $T_{\text{bin}}(t)$. Such a small LS is chosen because, unlike typical use cases for GANs, this project works with 1D and not two-dimensional data. Given the MinMaxScaler was used with a range from -1 to +1, the output G layer will take advantage of the hyperbolic tangent which outputs values just in that range. At the end of the inference data is just reverse transformed with the saved scaler instance used previously. Using Leaky ReLU, we introduce non-linearity. Without batch normalization, the generator may have more freedom to generate diverse and realistic samples. It does not use drop-out either, like in Pix2Pix, as the LS already provides the needed stochasticity.

Training Configuration

Loss Functions Wasserstein Loss approximates what is known as the Earth Mover's Distance, which measures the cost of transforming one probability distribution into another by computing the integrated difference between their cumulative distributions. Literature and experiments suggest it is more stable than usual binary cross-entropy. It provides a meaningful loss, allowing for tracking progress. The critic's and generator's respective loss functions using the Wasserstein distance are shown in Equations 7, 8. Minimizing the first equation can be seen as maximizing what is inside the parentheses. $\mathbb{E}_{k\sim p_K}$ represents the expectation operator applied to a random variable k that is drawn from the probability distribution p_K. In our project we are trying to model the real probability distribution p_R, by training our GAN on the empirical distribution $\mathcal{D}_R$. The obtained generated probability distribution p_G will be represented implicitly in the model's weight and biases. We will then assess its performance by simulating the generated data distribution $\mathcal{D}_G$, and comparing against the original $\mathcal{D}_R$. Going back to the equation, in the first term the expectation is computed over the entire set of real data samples r, drawn from the real data distribution p_R, with each critic score on this sample $D(r)$ multiplied by the corresponding probability of r. In the second term we sample from the latent space $Z \in \mathbb{R}^3$, and obtain the critic's score on the generated samples $G(z)$. In a nutshell, in order to make the critic work, we need to maximize the first term, or the score it gives to real samples, and minimize the second one, the score it gives to the fake ones. The generator, on the other hand, only must maximize the score received by its forgeries.

$$\min_D - (\mathbb{E}_{r\sim p_R}[D(r)] - \mathbb{E}_{z\sim p_Z}[D(G(z))]) \tag{7}$$

$$\min_G - (\mathbb{E}_{z\sim p_Z}[D(G(z))]) \tag{8}$$

In order for the outputs to resemble not only the distribution of the patient $\mathcal{D}_r$, but also the curves themselves, the generator gradients are calculated based on the weighted combination of the adversarial loss (Wasserestein distance), and the L2 loss (mean square error). The suggested ratio to implement is [1,100]. However, in the case for this model, it was found that the L2 loss was not so small compared to the first one, so this ratio led to the squared error overshadowing the adversarial component, leading to instability. A stable configuration was found using a ratio of [1,5].

Optimization and Hyperparameters A very small learning rate (LR) is required for this configuration. Arjovsky et al. [40] used a $5e-05$ learning rate. Numerous experiments with the ReplaceBG data have shown that this LR leads to unstable training and needs to be 5-10 fold smaller. The optimizer, an algorithm used to change the attributes of the neural network, such as weights and learning rate, to reduce the losses has been chosen to be Root Mean Square Propagation (RMSProp). It adjusts the learning rate for each weight of the model individually, using the magnitude of recent gradients for that weight to scale the learning rate. This means that weights with large gradients will have their learning rate decreased, while weights with small gradients will have their learning rate increased. This helps in fast convergence and solves some issues of the simple gradient descent like oscillations. Adam, which is used more commonly, is a very powerful optimizer that works well across a wide range of deep learning models. However, its momentum component can sometimes lead to instability in the training of WGANs, where delicate balance between generator and critic updates is crucial. The adaptive learning rate without momentum, as in RMSProp, tends to offer more stable convergence in this specific scenario. The discriminator is subjected to more intensive training compared to the generator, at a ratio of 5:1, reflecting the need for a robust critic. The generator uses approximately 178,000 trainable parameters, whereas the discriminator is significantly larger, with over 3.5 million parameters.

Batch Size, Epochs, Cleaning and Clipping A compromise between computational efficiency and the learning process needs to be struck. The original paper uses a Batch Size (BS) of 64, while Mujahid [42], used BS=1 and 50 epochs. The latter is not possible since the dataset size is 10 times bigger. Instead we'll use 20 epochs. To avoid the progressive slowing of the training process, after each epoch, G, D and GAN model weights are saved, all variables are deleted, and Python garbage collector run. After this 'cleaning', models would be loaded again and training resumed, with similar to initial iteration speed.

The original paper [40] also explains the importance of enforcing the 1-Lipschitz constraint, which ensures that the gradient of the discriminator's output with respect to its input does not exceed a magnitude of 1. Intuitively, a Lipschitz continuous function is limited in how fast it can change, which is crucial for stabilizing the training of GANs. Gradient penalty is the state of the art way to enforce it, however the simpler method of weight clipping has shown to work.

Table 4: Summary of Model Configurations and Architectural Choices

Feature	Description
Architecture	Wasserstein Conditional GAN
Scaling method	MinMax [-1,+1]
Generator Input	Insulin, Carbohydrates, Time Bin
Generator Output	Size 12 vector with 2 hour glucose prediction horizon
Latent Space (Z)	3 dimensional standard normal distribution
Activation (Generator, Discriminator)	Tanh, Linear
Loss Function (Generator, Discriminator)	Wasserstein Distance + L2 Loss, Wasserestein Distance
Ratio Generator Losses (Adversarial-L2)	1-5
Optimizer Type and LR	RMSprop, $5 \cdot 10^{-6}$
Sample Rate	10 min
Batch Size	64
Epochs	20
Parameters (Generator, Discriminator, Total)	3,511,793 - 178,420 - 3,690,213
Time embeddings size	4
Weight clipping	0.01
Total training steps	972320

5.4 Validation

Once the generator is trained, the discriminator and GAN can be thrown away, and inference can start. In our case, the generator has been trained to produce 12 glucose values starting now, until 120 minutes into the future, sampled every 10 minutes. To produce a BG profile, PI and RA curves from the validation dataset are then sequentially input into the generator. At a given time, t, we use inputs from $RA_{t-60min}$, $PI_{t-60min}$ - $RA_{t-40min}$, $PI_{t-40min}$ - $RA_{t-20min}$, $PI_{t-20min}$ - RA_t, PI_t, the corresponding time of the day $T_{\text{bin}}(t)$. When moving to $t+10min$, the current BG value becomes the average of the forecast from the previous time step and the present prediction for that timestamp. The process of averaging out predictions acts as a kind of moving average.

The concatenation of this process over all the validation set produces a set of simulated curves that represent the blood glucose levels of the patients in response to the real inputs of insulin carbohydrates and time. These are then inversely scaled with the MinMaxScaler saved instance. However, they are not perfectly aligned with the data, showing certain shifts and compressions. They need to be transformed again, which can be done as long as the same is applied equally to all the data points. To decide on the used transformation, an optimization algorithm is used to find the 2 best suited parameters based on comapring `mean-glucose`, `coefficient-of-variation` and `time-in-range` metrics to the real ones. These parameters are then applied to the new data in the next test phase.

5.5 Testing

Finally, once all hyperparameters are selected, the testing phase is conducted using previously unseen data. The simulated and real blood glucose data are compared based on their metrics. To determine the appropriate statistical test for validating our results, we first assessed normality in our dataset, which

includes nearly 250 subpatients. According to the Central Limit Theorem, as explained in [70], "as n increases, the sampling distribution of x is increasingly concentrated around μ and becomes closer and closer to a Gaussian distribution." In other words, it suggests that the distribution of sample means approximates a normal distribution as sample sizes exceed 30, allowing us to assume normality for our large sample size. However, to confirm this, we employed various normality tests, including the Kolmogorov-Smirnov test, which is particularly effective for larger datasets with n>50 due to its sensitivity to differences in the empirical distribution function of the sample and a reference normal distribution [71]. In addition to numerical tests, we also used visual methods to assess normality. Quantile-Quantile (Q-Q) (Figs. 15 to 16, more can be found in Annex 5's Figs. , 54, 55, 56) plots provide graphical views to check how closely the data follow a Gaussian distribution. The Q-Q plot compares the quantiles of the sample distribution to the quantiles of a theoretical normal distribution, providing am additional visual assessment. After confirming the assumption of normality in the key outcomes, we proceeded with the Student's t-test, a statistical test that determines if there are significant differences between the means of two groups. This test is suitable for normally distributed data and helps in making inferences about population means based on sample means. Thus, the combination of these tests and visual assessments supports the validity of our findings under the assumption of normality. For not normally-distributed metrics, the Wilcoxon signed-rank test was used to evaluate statistical similarity, assuming that real and generated outcomes came from the same population of patients. In our case a p-value>0.05 supports the hypothesis of similarity. In Tables 5, 6, the shaded rows indicate data points that are accepted as normal based on the Kolmogorov-Smirnov statistic (K-S stat) and the p-value interpretation. A small K-S stat suggests the sample distribution closely follows a normal distribution. Correspondingly, a high p-value (p>0.05) indicates that there is no significant evidence to reject the hypothesis that the data is normally distributed.

Table 5: Kolmogorov-Smirnov test results for generated data

Variable	K-S Stat	p-value
mean_glucose	0.073	0.138
std_glucose	0.045	0.689
max_glucose	0.04	0.811
min_glucose	0.099	0.014
coefficient_of_variation	0.032	0.953
time_between_70_54	0.116	0.002
time_in_range	0.083	0.06
time_over_180	0.08	0.081
time_over_250	0.16	<0.001
time_under_54	0.131	<0.001

Table 6: Kolmogorov-Smirnov test results for real data

Variable	K-S Stat	p-value
mean_glucose	0.081	0.075
std_glucose	0.055	0.421
max_glucose	0.075	0.118
min_glucose	0.129	<0.001
coefficient_of_variation	0.034	0.916
time_between_70_54	0.149	<0.001
time_in_range	0.051	0.512
time_over_180	0.060	0.330
time_over_250	0.260	0.002
time_under_54	0.118	<0.001

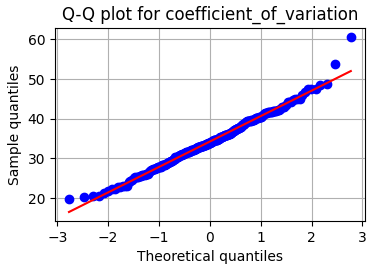

Figure 15: Q-Q plot on generated `coeff._variation`

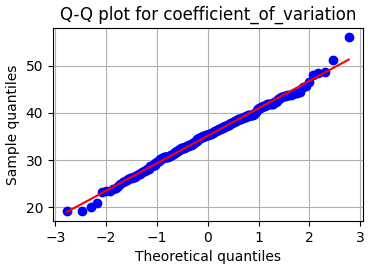

Figure 16: Q-Q plot for real `coeff._variation`

Time Inputs

To validate if time is causing variability in the output profiles, the test set was used to simulate curves with the same PI and RA inputs as before, but modified TM values. Specifically, we will run the simulation with all test patients 4 times, each time with a fixed time bin value. That way, differences in the outcomes can be studied based on the learned dependencies tied to each bin, as well as assessing if using the correct TM provides better results than using incorrect ones.

Causality

To assess causality, we will be using Convergent Cross Mapping (CCM), a statistical method used to determine a potential cause-and-effect relationships between variables. CCM essentially evaluates whether the historical data of one variable can reliably predict the state of another. Its objective is to solve the challenge that correlation does not necessarily imply causation. To do so, it reconstructs the dynamical system based on time-delayed coordinates (X_t-$X_{t-\tau}$), or what is called 'shadow manifolds'. The method involves plotting these manifolds and then cross mapping points from one to another, visually analyzing how well the trajectory of one variable can predict the trajectory of another. CCM is chosen over other causality tests because of its robustness in handling nonlinear relationships and complex dynamics. These are very typical of physiological systems like our. Unlike Granger causality, which requires assumptions about linear interactions and often fails in the presence of feedback loops or interconnected components. The CCM test was conducted using the `causal-ccm` package in Python. Before obtaining the cross mapping plots, data is scaled for better visualization.

5.6 Distribution of the Data

To evaluate if the model learnt the underlying real probability distribution p_R, we will visually compare the empirical distributions of real $\mathcal{D}_R$ and generated data $\mathcal{D}_G$, using two dimensionality reduction techniques: Principal Component Analysis (PCA) and t-distributed Stochastic Neighbor Embedding (t-SNE). Additionally, we will explore the embedded information in different regions of the LS, by seeing the effect sampling from a truncated p_Z has on the output.

6 Results

In this section we present the results obtained after the execution of what is described in Section 5, including training outcomes and tracking, plots of real versus generated curves, and overall metrics for both cohorts. These results will be discussed later in Section 7.

6.1 Overview

AGP Report

An Ambulatory Glucose Profile (AGP) report is a standard way of presenting a summary of the blood glucose control over an extended period of time using continuous monitoring. It allows for a quick visual and statistical inspection. These reports have been obtained using the `iglu` package from `R`.

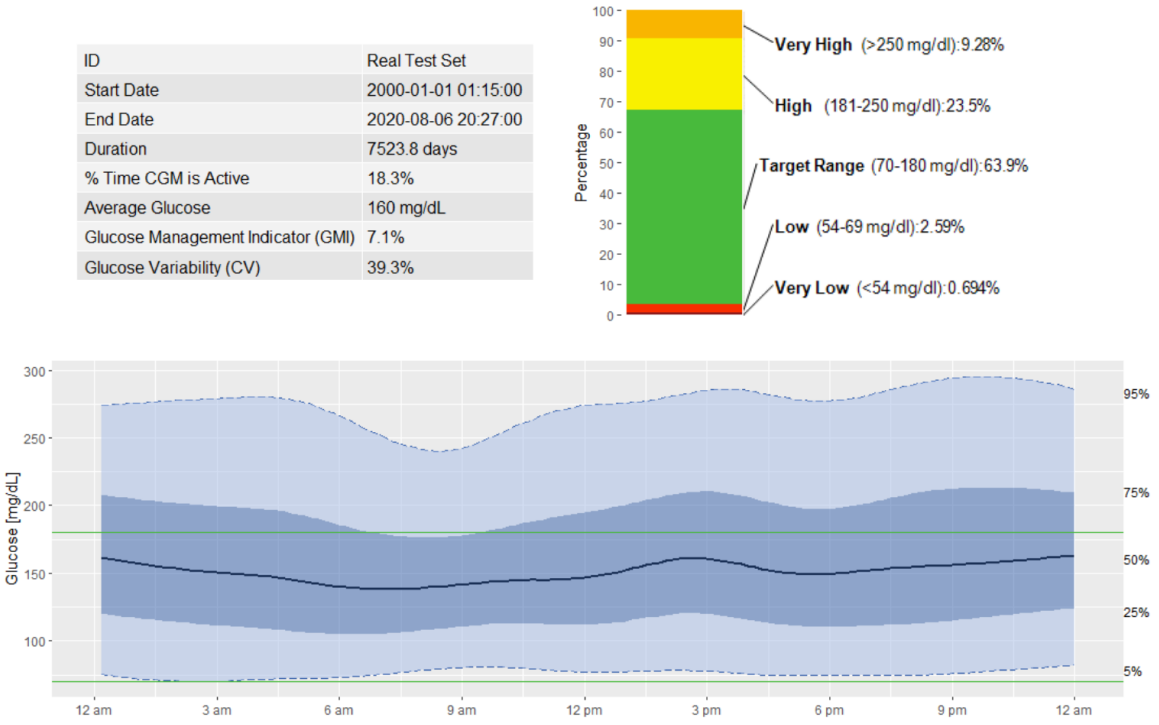

Figure 17: AGP report using aggregated data from realtest patients

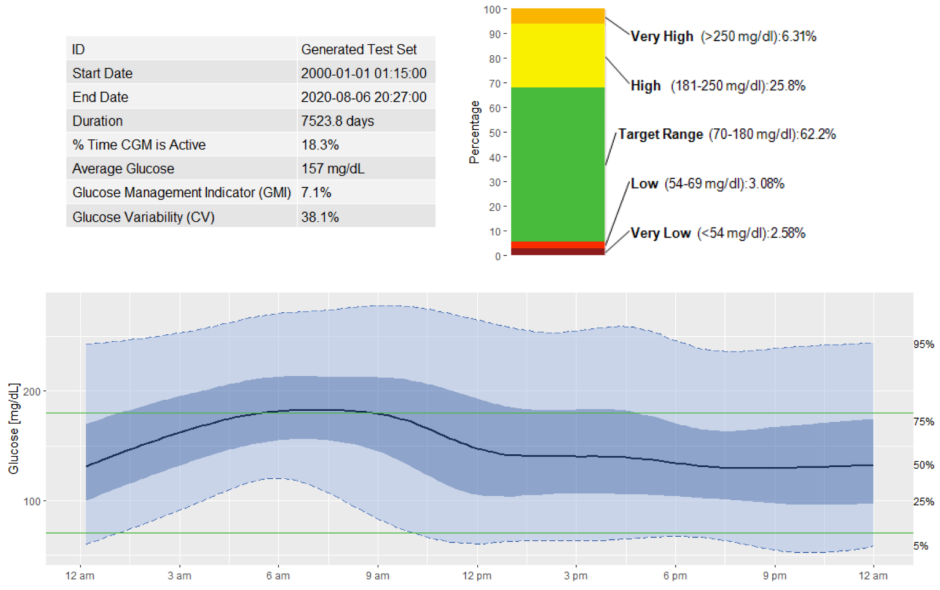

Figure 18: AGP report using aggregated data from generated test patients

Side-To-Side Blood Glucose Profiles

Here, we can see the obtained plots after the process of generation. An overview of the real and synthetic profiles from several subpatients, alongside the corresponding inputs of insulin and carbohydrates are presented below.

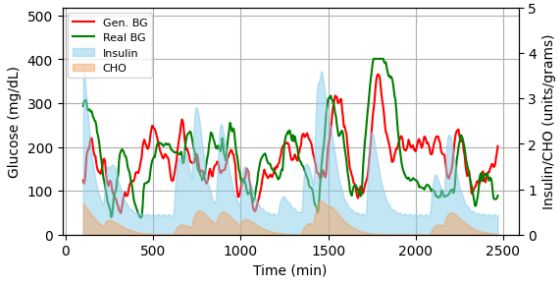

Figure 19: Sample 248-32

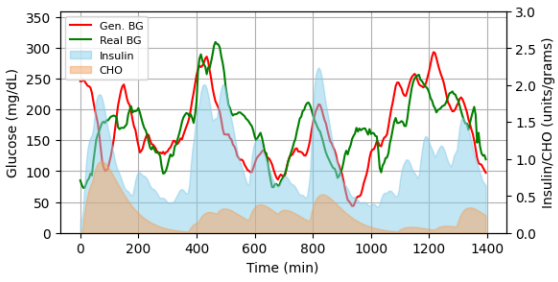

Figure 20: Sample 234-52

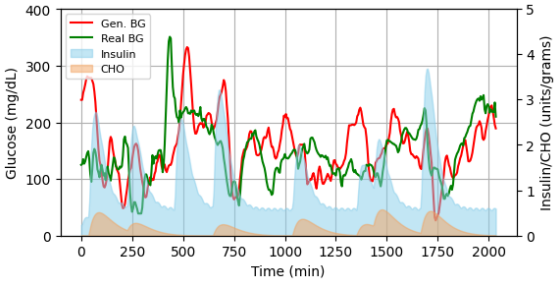

Figure 21: Sample 250-7

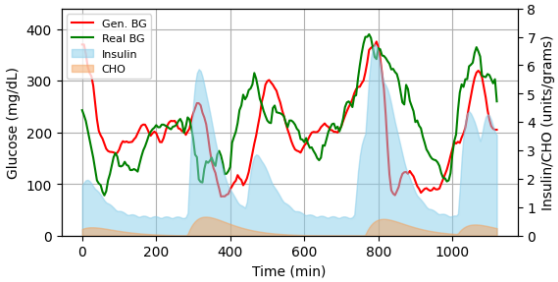

Figure 22: Sample 239-35

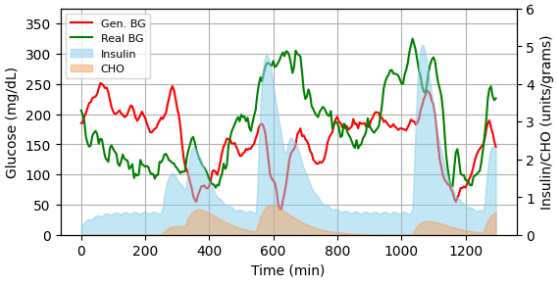

Figure 23: Sample 250-12

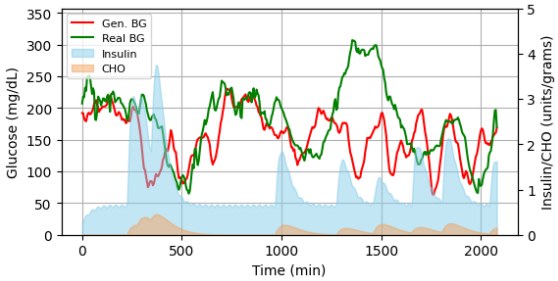

Figure 24: Sample 239-27

Loss Evolutions

During the training process, the losses were logged every 100 iterations which allowed analysis of the evolution of the quality of the samples for possible early stopping.

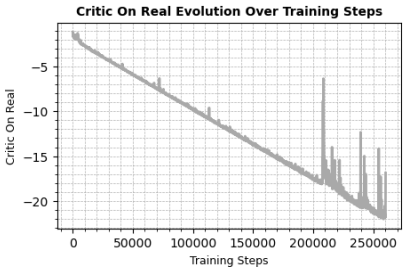

Figure 25: Critic's loss on real samples

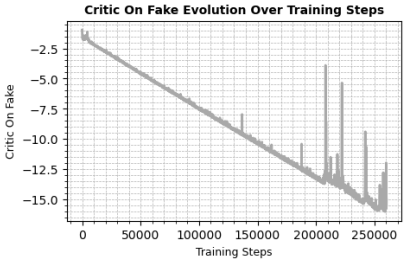

Figure 26: Critic's loss generated samples

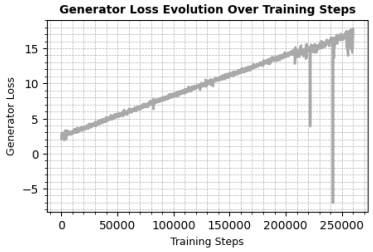

Figure 27: Generator's overall loss

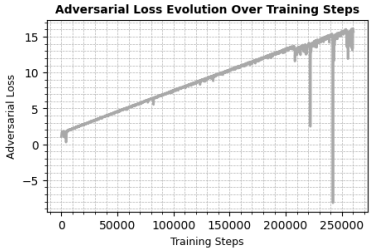

Figure 28: Generator's adversarial loss

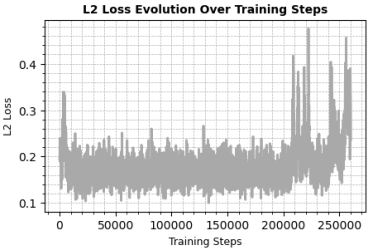

Figure 29: Generator's mean square loss

6.2 Validation

Statistical Analysis

The optimized scale factor and shift found are 4.368 and 3.25 respectively. The number of validation and test samples equal 189,770 and 198,393. Once the curves have been generated, we need to measure how successful we were in replicating the original data distribution in a quantifiable manner. As aforementioned, point-to-point comparison is not suitable for this case and general blood glucose metrics will be used. For statistical assessment, a Student's t-test is performed. Before deciding on using the t-test, normality is checked as seen in Tables 5, 6.

Table 7: Comparison of Metrics Between Real patient's CGM and Synthetic CGM Presented as Average or IQR (25th;75th) Percentile, with the p-value resulting from the Student's t-test (shaded), Wilcoxon Signed-Rank Test (not shaded)

Statistic	Real CGM	Synthetic CGM	p-value
mean_glucose *mg/dL*	160.29 (145.74-175.33)	156.45 (142.65-173.7)	0.063
std_glucose *mg/dL*	56.53 (45.2-66.13)	54.48 (43.8-62.41)	0.095
max_glucose *mg/dL*	328.06 (281.75-375.25)	330.86 (273.87-376.13)	0.635
min_glucose *mg/dL*	50.35 (40.0-57.0)	28.19 (15.66-40.71)	<0.001
coeff.variation %	35.14 (31.47-38.81)	34.58 (30.16-38.55)	0.312
time_between_70_54 %	2.63 (0.83-3.59)	3.18 (2.88-7.47)	0.011
time_in_range %	62.43 (50.13-75.1)	62.09 (50.19-75.24)	0.29
time_between_180_250 %	33.15 (22.43-43.26)	25.74 (18.53-34.33)	0.054
time_under_54 %	0.68 (0.00-1.05)	2.60 (0.92-3.38)	<0.001
time_over_250 %	9.32 (2.38-14.09)	6.40 (1.19-9.73)	<0.001

Time Inputs

As explained in the detail engineering section, to assess the difference in generation for the time labels, the whole test set is simulated 4 times, each with a fixed time value. The resulting metrics for each of the four simulations are presented in Table 14 in Annex 4.

Table 8: Count of validated metrics (p-value>0.05) using the corresponding statistical test (Student's t-test or Wilcoxon Signed Rank). Comparing real vs. synthethic data, grouped by time bins (TM=X), using data with fixed time inputs simulations (SIM-TMX)

	TM=0	TM=1	TM=2	TM=3
SIM-TM0	4	3	4	3
SIM-TM1	6	3	3	3
SIM-TM2	8	4	4	5
SIM-TM3	5	3	2	3

Causality: Convergent Cross Mapping

Table 9: Correlation ('strength of causality') and p-value ('significance') using `causal-ccm` for aggregated patients from the test set for PI→BG

Patient	248	249	250	251	253	254	256	258	263	264	266	MEAN[1]
Corr.	0.0928	0.0445	0.0676	0.0508	0.323	0.0524	0.0607	0.0878	0.051	0.1382	0.0766	0.0958
P-value	3.6e-28	2.41e-05	2.23e-32	6.38e-13	6.76e-50	9.24e-13	2.87e-17	2.6e-23	5.25e-21	2.1e-41	2.16e-14	-

Table 10: Correlation ('strength of causality') and p-value ('significance') using `causal-ccm` for aggregated patients from the test set for RA→BG

Patient	248	249	250	251	253	254	256	258	263	264	266	MEAN[1]
Corr.	0.1673	0.0749	0.1138	0.105	0.1700	0.0972	0.128	0.1094	0.1133	0.1261	0.1596	0.1205
P-value	2.16e-88	1.1e-12	7.6e-89	3.87e-50	1.8e-14	3.21e-40	2.1e-71	2.25e-35	1.54e-97	1.09e-34	1.17e-57	-

Table 11: Correlation ('strength of causality') and p-value ('significance') using `causal-ccm` for aggregated patients from the test set for TM→BG

Patient	248	249	250	251	253	254	256	258	263	264	266	MEAN[1]
Corr.	0.075	0.0585	0.0419	0.0635	0.1886	0.0429	0.019	0.0485	0.0473	0.1079	0.0562	0.0714
P-value	6.34e-19	2.72e-08	2.26e-13	2.63e-19	1.70e-17	4.98e-09	0.0083	3.98e-08	2.47e-18	9.1e-26	2.1e-08	-

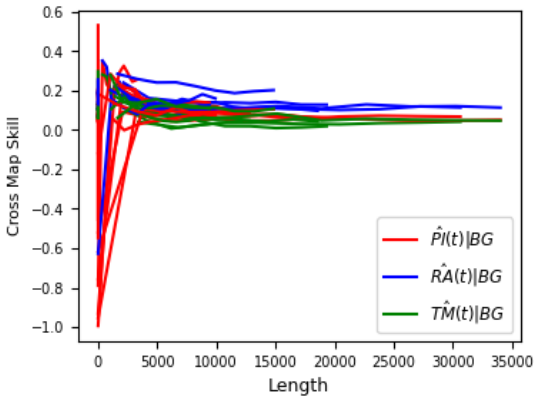

Figure 30: Convergent Cross Mapping skill, or correlation, evolution for test patients

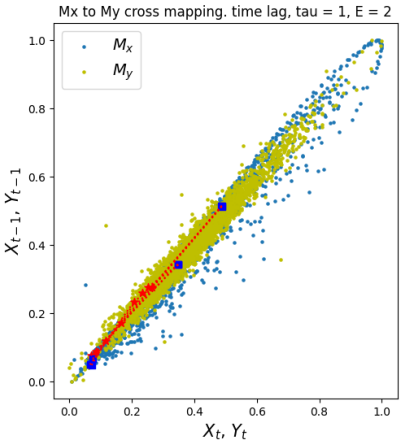

Figure 31: Cross mapping of PI→BG 250

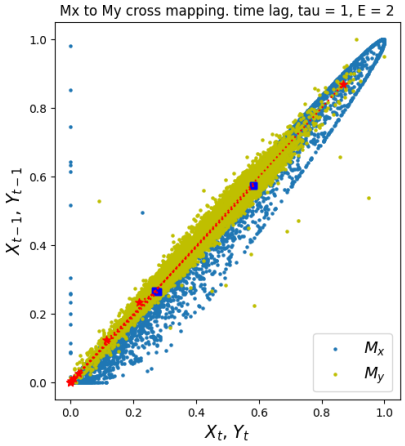

Figure 32: Cross mapping of RA→BG 250

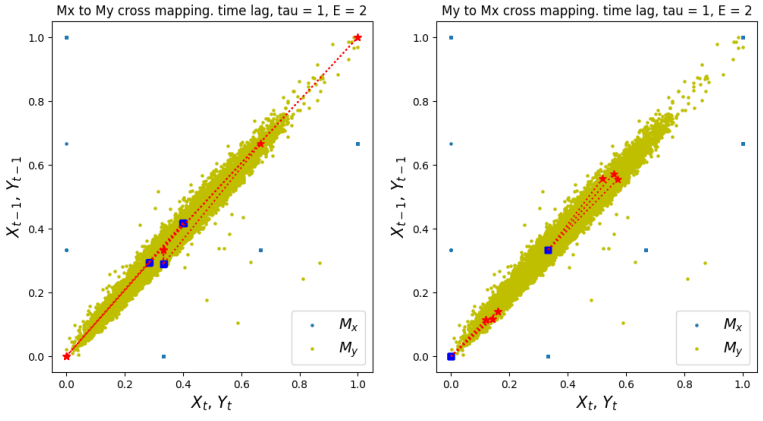

Figure 33: Bi-directional cross mapping of TM↔BG 250

6.3 Data Distribution Representation: $\mathcal{D}_R$ and $\mathcal{D}_G$

Principal Component Analysis (PCA) and t-distributed Stochastic Neighbor Embedding (t-SNE) are two dimensionality reduction techniques that allow representing the distribution of real and syntethic BG values. Only the first 5 test patients are used to avoid clutter in Figs. 34, 35, and the next 10 patients for Fig. 36.

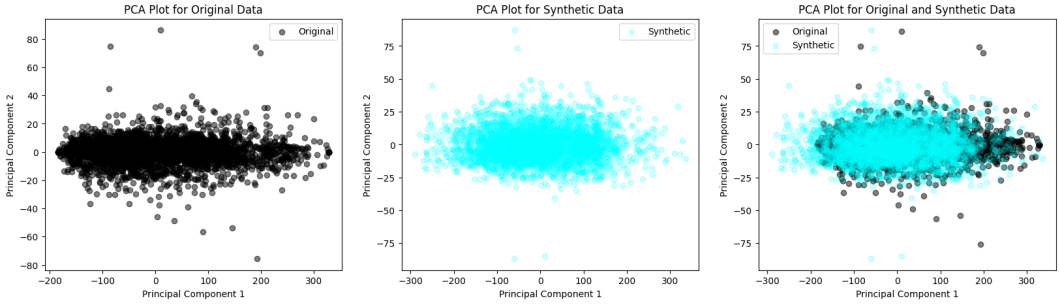

Figure 34: PCA analysis (PCs=2) of of real and syntethic BG values (BG dim=2) for test patients 1-5

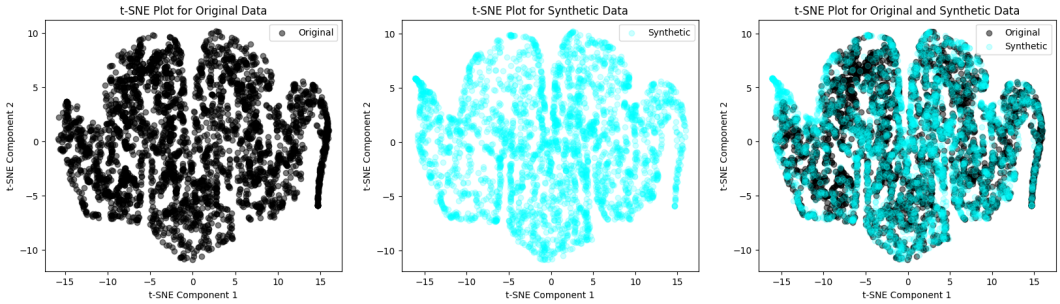

Figure 35: t-SNE analysis (PCs=2) of real and syntethic BG values (BG dim=2) for test patients 1-5

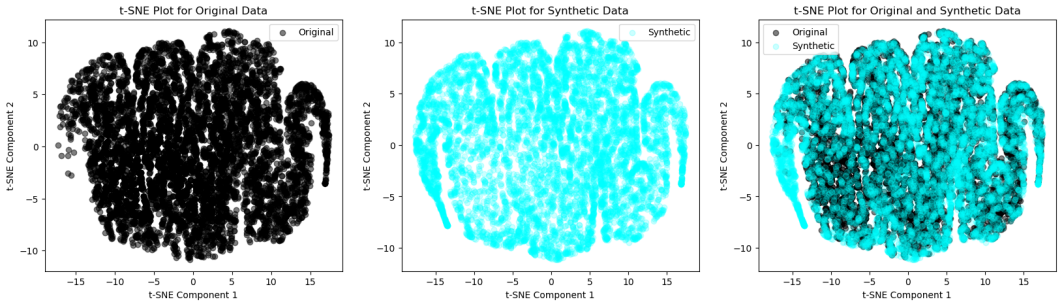

Figure 36: t-SNE analysis (PCs=2) of real and syntethic BG values (BG dim=2) for test patients 5-14

Latent Space Exploration

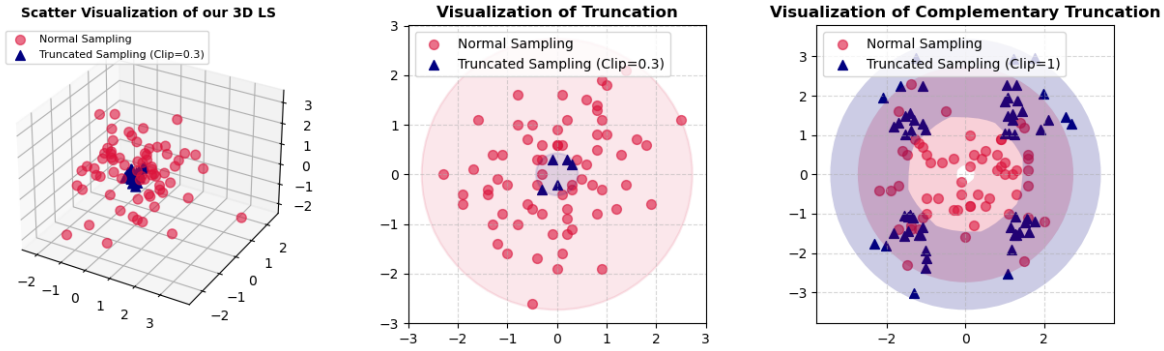

Figure 37: Visualizing the latent space Z, sampling normal and truncated Gaussian distributions

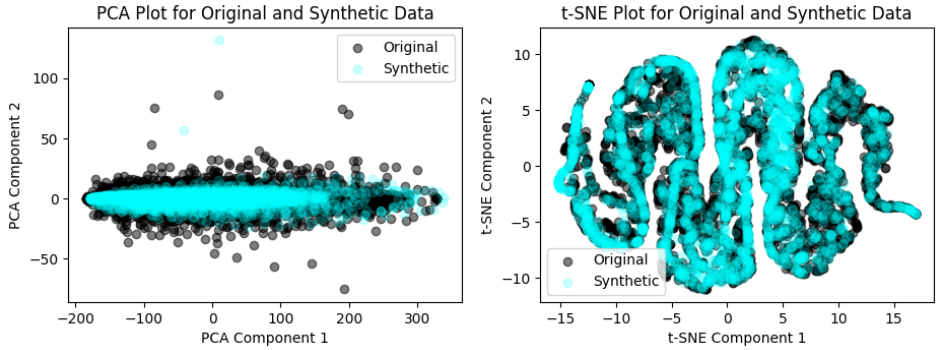

Figure 38: PCA and t-SNE analysis (PCs=2) of real and syntethic BG values (BG dim=2) with truncated Latent Space

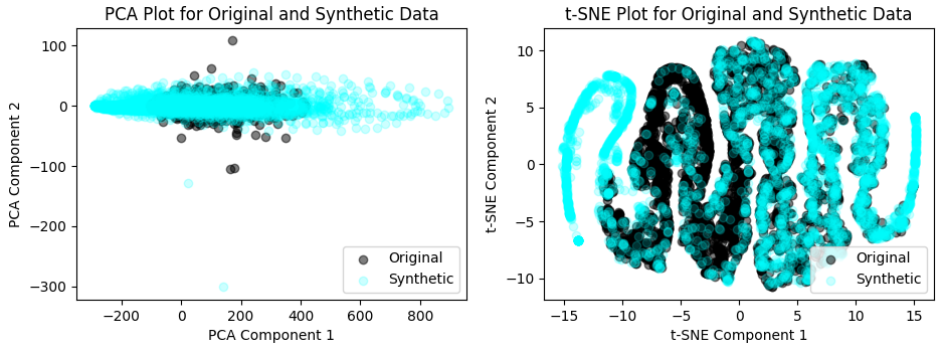

Figure 39: PCA and t-SNE analysis (PCs=2) of real and syntethic BG values (BG dim=2) with the complementary truncated Latent Space

7 Discussion

In this section we'll make an in-depth analysis of the results presented above, evaluating their significance, highlighting key findings, discussing challenges and foreseeing the model's future use.

7.1 Interpretation of Results

7.1.1 Overview

The Final Configuration

The chosen design parameters of the CGAN, as summarized in 4, are not the result of arbitrary decisions, but of research and experimentation. These configurations are of vital importance, as without correct tuning of variables the training process does not evolve correctly and the results are complete noise. The initial LR guesses around the order of magnitude suggested by the original WGAN paper [40] at $5 \cdot 10^{-5}$, seemed to work for several tens of thousands of iterations. However, when used for longer than 100k steps the training started to collapse, with losses fluctuating, and generations losing quality. An equilibrium between batch size, epochs and training time needed to be found, as the preferably high amount of epochs, combined with the ideal low batch size, would be completely out of reach to finish within schedule with the computational power provided by our single GPU. This is due to the amount of data (3,111,479 samples only for training), alongside more than 3.6 M network parameters. To put that into perspective, in the landscape of NN, this size represents a middle ground in terms of complexity, given the increased numbers image or video-based models present. The early convolutional network, LeNet-5, had only 60,000 parameters and was designed for basic digit recognition tasks [72]. The ResNet-50, consisted of about 25 million parameters, significantly boosting performance in image classification [73]. Our GAN is dwarfed by the sheer scale of LLMs like GPT-3, containing 175 bn. parameters, representing a state-of-the-art in language processing capabilities [74].

The decided final configuration (Table 4), expected the total steps to be 972,320, or 20 epochs. After many attempts, only 260 thousand steps were executed, a little over 5 epochs. Several reasons are to blame, including: the recurrent destabilization of the process after 200k iterations. Time limitations, as each attempt that reached such stage, took 3-5 days. And also the degradation of the generations.

Preprocessing of the data left the 226 patients from ReplaceBG split into numerous subpatients. This was done to ensure no missing data was included, and that this was accomplished with as few interpolations as possible. However, this left the dataset split into possibly too many small pieces. Working with short periods of data may not seem a problem at first, but in retrospect, it may have been better to work with at least 15 days, if any temporal pattern is to be learnt. No subpatient contains such lengths, with the maximum being a little shorter than 10 days. This could be accomplished by reducing the filtering, even if that meant allowing the presence of missing sequences. Finally, the decided scaling method involved transforming time-series individually using MinMax scaling, rather than operating globally, which improved results significantly with respect to initial implementations.

The Training Process

If we turn our attention to the loss functions' evolution subsection, the figures present a clear stable trend. Towards the end, the process becomes unstable, while Mean Squared Error (MSE or L2) spikes, so training is stopped manually at step 260k (approximately 27%) (see total steps in Table 4). This MSE loss correctly goes down at the start, as the generator goes from noise to physiologically sound curves. After that, the point-to-point error does not change much as the generator is learning the underlying probability distribution p_R. Given L2's smaller scale, its weight on the overall generator loss is overshadowed by adversarial loss. The latter, seen in Fig. 28, is the critic's score on the output samples crafted by the generator. If samples get better, adversarial loss, should increase, as it does. Figs. 25, 26 show the score on real and fake samples by the critic. They should become similar, as the critic is fooled and unable to distinguish them, it gives similar scores to both.

The Blood Glucose Profiles & AGP

There is an obvious issue with analyzing AGP reports (Figs. 17, 18), which is that for so much data they become averaged out, and few or small oscillations are distinguishable. They also clearly show why intra-day variability or time dependencies might be difficult to learn from such a big dataset, as the real AGP is predominantly flat. The generated one shows a different behavior during the first half of the day which is commented in discussion of time's effect in: 7.1.2. The 7,523 days do not reflect the actual data used, as each subpatient's time vector starts on the first day of the month following the last subpatient's month. The real number is approximately 1400 days. The metrics show high coincidence between synthetic and ground truth, as well as the time in ranges. These glycemic outcomes are quantitatively analyzed in the validation section 6.2. More AGPs, from an actual individual subpatient (found in the Annex: Figs. 51, 52), is used to illustrate how, if we look closer, the tendencies and oscillations are present and modeled, but we can't see them in the aggregated plots.

Next, in comparing the side-to-side BG profiles (subsection 6.1), we can say that despite the generator lacking direct or past BG data, it surprisingly aligns well with patient BG profiles, effectively approximating actual dynamics solely from indirect inputs. This holds true, despite not being the objective, indicating the model's capability to approximate actual dynamics effectively. We can wonder what would have happened, for instance, if past BG was used as input. It is possible the generator took a shortcut and relied heavily on this input for its generations, hindering its capability to learn the dynamics from the PI, RA and TM inputs.

Overviewing the different samples, we can see Figs. 19 through 22 have a good visual resemblance. Patterns are close and it is clear the model is depicting the correct dynamics based on the indirect inputs. Despite not trying to replicate BG point-to-point, this learned dynamics sometimes makes the synthetic profiles be relatively close to the originals. As the GAN learns the representation of the probability distribution implicitly in the weights and biases of the generator (p_G), it is able to behave like the original patients may do. Yet not all profiles align as good as these (Figs. 23, 24). And they shouldn't. Regardless, these two examples present some interesting behaviors we will discuss.

Firstly, during the first 250 minutes in Fig. 23, there is no input of carbohydrates, and only of insulin, but still the model is generating a curve, which is not possible with all types of conditional models. The latter learn to transform the input variables into the output values, if there is no input, there is nothing to transform, so it cannot generate. In our case however, the generative conditional model has learned to transform the sampled point from the latent space, regardless of the presence of inputs. The conditions should only alter the shape of the BG profile but are not essential for its production. This property can make the task of using it for simulation easier. After that initial phase, a small meal and bolus occur. These lead to a short postprandial peak followed by a steep fall into an hypoglycemia. Despite differing from the real curve's behavior, this dynamic is perfectly physiological. It may even be what this patient, would have experienced in a different day, given a different state in the myriad of factors that regulate his glucose dynamics. These can't all be included in the model, but GANs aims at replicating their variable effect by introducing the LS to create stochasticity. Something similar happens with the second and third meal, a meal followed by an hypoglycemia. We could actually be facing a patient with a lower than average insulin sensitivity with respect to the population. Thus when the model, trained on the entire cohort, makes predictions for this specific patient, it fails to predict accurately. Looking ahead, tailored adjustment of the relative of effects of PI and RA would require patient-specific training, or the use of clusters.

Analyzing subpatient 239-27, Fig. 24, we observe a fairly overlapping profile, until around the minute 1400, where real BG goes into a postprandial excursion, which the model fails to replicate. However, if we analyzed the inputs, without knowing the outputs, we probably would have never guessed this postprandial excursion, as no big meal is recorded in the input data, and only a series of small intakes happen after the hyperglycemic event is already in place. All of this happens while insulin boluses keep coming in. So the generator is predicting reasonably. We can make suppositions on things that might happen here, apart from the previously mentioned stochasticity, training data could be to blame. A meal might be consumed and not noted down. If annotated, quantity of carbohydrates might be wrong, due to typing errors or misestimations of the meal. Even if correctly calculated and typed, not all carbs have the same response, which will depend on their Glycemic Index, or the composition of the rest of the macronutrients, which may significantly slow down the rate of appearance. The basal insulin is computed from solving the differential equation of plasma insulin for each time step, using a infusion basal rate from the pump that was computed during the preprocessing from each patient's demographics, but not recorded in the data. This value was kept constant throughout all of their time series, but in reality, it is adjusted recurrently over time. Scenarios are also limited in including any physical activity. As we can see, the data fed into the model is of utmost importance, and we used numerous but imperfect samples for our model training. Despite all of these limitations, results are satisfactory, as we can affirm that by qualitatively analyzing the profiles, we have successfully replicated the physiological insulin-carbohydrates-glucose dynamics, with profiles that behave with great deal of similarity to the originals.

7.1.2 Validation

Statistical analysis

Before discussing the results themselves, some comments can be made about the optimization of the scaling and shifting parameters done using the validation set. (The *validation* set, referring to the cohort of patients used for optimizing the paramaters of scaling and shift, not to be confused with the process of *validating* or confirming the correctness of results.). The process showed to provide good results with the test set. Despite that, it is notable that a scaling factor of more than 4 was needed, even after the generator used the hyperbolic tangent as the last layer's activation function (range from -1 to +1), and data was scaled using MinMax with range [-1,+1]. This way, it was possible for the generator to recreate data at the same scale as the real profiles. We expected that when rescaling the generated data using the MinMax instance fitted to the real data, both vectors would have similar magnitudes. But this was not the case. Regardless, the scaling correction needed was consistent for all generated data, proven by the succesful results. But future versions should try to tackle this problem and try to minimize the need for big scaling factors, as this can be causing part of the discrepancies at the metrics of the extremes.

Going into the statistical tests results (Table 7), comparing the real and generated data metrics, we focus primarily on whether the means of the two datasets are statistically different. Before the tests, normality was assesed (Tables 5, 6). Resemblant results in the Kolmogorov-Smirnov test may indicate, although indirectly, a certain similarity between the data. This is a not strong argument for validating the generated data, as akin results in terms of their normality could be due to the nature of the metric calculation itself (mean, standard deviation, maximum etc.), rather than to the underlying data.

The t-test, conducted for the shaded rows, measures the size of the difference relative to the variation in the sample data. A small p-value suggest a significant difference between the groups. For variables with p-values greater than 0.05, we fail to reject the null hypothesis. This indicates no significant difference between the real and generated data. For the validated metrics, we can affirm that the generated data using the test set scenarios is statistically similar to the actual blood glucose profiles.

Not normally distributed outcomes coincide with those with p-values<0.05 (in their case using the Wilcoxon Signed-Rank test), suggesting a significant difference between the outcomes. The unsatisfactory results in most of these metrics are not surprising given they are the most extreme ends of the curves, which can be the most challenging to represent.

However, we validated the most clinically relevant metrics we find: the mean, coefficient of variation and time range. Comparing the results to the ones obtained by Mujahid et al. seen in Tables 1, 2 and 3 in [42], we see their results apparently show better resemblence, validating all p-values. However, we have to take into account the number of patients used to generate these results. The biggest dataset of the three was Hospital Clinic de Barcelona's T1D dataset, comprising 27 patients in total. This means the statistical test with the test set split from this dataset would have been significantly smaller than ours, originally composed of 226 patients. More data points strengthens statistical significance when

trying to prove a difference in a study. In our case, trying to prove the opposite, similarity, and doing so with so many samples entails a much stricter test. So, overall, even if not all metrics surpass the 5% significance level, those who do, present a much more robust validation of the results. We can see if curves were shifted up, the excess of time under 54 generated profiles have over the real ones, could be reduced. At the same time we would be increasing time over 250, which would make both metrics' error improve. It is not clear how this would affect the rest of the results. Neverteheless, this is an assessment we make after the transformation parameters are already optimized with the validation set, and statistical test are obtained. Parameters should not be adjusted to satisfy the needs of the test set, as we would be overfitting the data, and losing generalization.

Although both extremes (`time_over_250` and `time_under_54`) show differences with respect to the objective, we can see that the differences are more pronounced in the hypoglycemic area. It would be interesting to explore the development of the model using a non-linear scaling in order to balance the importance of both extreme areas. Right now, the hyperglycemic area is over-represented as any value in the range from 180 to 500, is considered in that category. On the other hand only the 70 points from 0 to 70 mg/dL are classified as hypoglycemic. Moreover, the clinical center: the euglycemic range, is not at all at the center of the data. Kovatchev et al. [75] proposed a solution to symmetrize blood glucose data applying a logarithmic transformation as seen in Fig. 40. This could potentially help the model be more accurate in these low regions and overall generate better results.

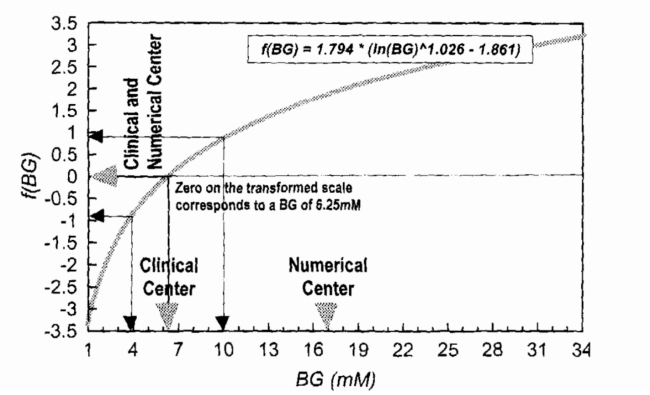

Figure 40: Transformation from normal to symmetrized blood glucose scale. Numerical and clinical center coincide and hypo- and hyper- ranges become symmetric [75].

Time

As explained in 2.1.1, insulin basal rates vary during the day. This is one of the components that makes T1DM so complicated as not only inter-patient, but also intra-patient variability is present. During the early morning hours, basal insulin peaks, while insulin sensitivity troughs. This imbalance is usually behind the dawn effect. If insulin is not adjusted correctly, lower sensitivity to insulin can lead to higher glucose levels. In our simulations with a fixed time label, as seen in Table 14, the highest mean blood

glucose levels correspond to those generations using the night period label, between 12 AM and 6 AM, when the dawn phenomenon usually takes place.

This is also seen in the generated AGP report (Fig. 18), which shows a prolonged peak during this period which also extends into the morning. This peak is not found on the real AGP, which seems odd. We can try to formulate some explanations, like that maybe this phenomenon *is* present in the real data, but the different moments each patient experiences it, cancel out. In turn, the model learned the upwards tendency and embedded it into the 6 hour window, and learned to represent it always alike for all patients, so that the dawn phenomenon is plotted without the same time shift in the generated curves and then the aggregation in the AGP report does not average it out. The extension of this phenomena past the end of the first time window may be caused by the 2-hour prediction horizon. What is clear from this AGP is that the model embed some kind of relationship with time. We can look into the individual AGP, and see that the real patient (Fig. 51) either has breakfast at 5 AM and then eats something aorund 9 AM, or is actually showing the dawn phenomenon at the end of the night and start of the morning. The generated one in turn (Fig. 52), changes a downwards trend during the evening into a clear and steady upward trend during the whole night, ending a little before 6 AM. This is just from 1 subpatient, but aligns with our previous guess, where the higher blood tendency is present in the real data but only for a short period of time, and is maintained for the whole TM=0 period during generation. We can notice how the window from 00:00h to 06:00h might not be ideal, as for this patient, the dawn phenomenon steps into the next time window. Maybe, in the future, the number and position of the windows should be adjusted differently, possibly being more effective if made specific for each cluster.

If we turn our attention to Table 8, the results represent the number of metrics statistically similar to the real profiles. The diagonal values are the results from the periods where the correct input was used, as the 4 hour time period and the TM label for that fixed simulation coincided. In red, all the other combinations for the remaining 18 hours of the day. We would expect, given the time input was having a big impact on the realness of the generations, that these diagonal values were higher than those outside of it. However, the results do not follow this hypothesis. Rather, the moment of the day itself seems to be more important in terms of similarity, as the night period had always the best results. This is logical provided the night is predominantly empty of meals or bolus corrections.

Given the stochastic nature of the model, in reality it would be preferable to make more simulations to understand if these counts are consistent or not. This however is not possible due to time constraints, as making a single complete simulation of the test-set for one TM category took up to 10 hours. Doing so for the 4 categories, repeated 5 times would take around 8 days of continuous simulation. What is evident is that time, although *causing* part of the variability in the output (as seen in the next results Subsection 6.2), is not comparable to the effect of the other two inputs, and may be overshadowed. This can be due to a variety of causes, one of them is the effect of the heterogeneity of the data. The populational model may be great to gather large amounts of data but could have averaged out the circadian rhythms we were looking for. Maybe the selection of bins has not been optimal, or, despite relevant findings showing circadian rhythms are clearly visible in insulin secretion, like seen in subsection 2.1.1,

this does not have such a big of an impact on glucose in the first place, or not in our dataset. This could mean that even if perfect modeling was done, there is not much to learn. Again, we insist on how important clustering data can be in future works so that these problems can be mitigated.

Causality

In subsection 6.2 we present a Convergence Cross Mapping over different lengths of data in Fig. 30, and cross mappings between inputs: PI, RA and TM, and output BG in Figs. 31, 32, 33. Inputs are represented by M_x, and glucose by M_y. In Figs. 31, 32, only the effect of X (either PI or RA) on Y (BG) is shown, whereas in 33 both directions appear. To visualize the shadow manifolds, M_x is represented as a scatter plot of the current value X_t against its past X_{t-1} value. While M_y is plotted similarly using Y_t against Y_{t-1}. These manifolds are visual representations showing how each variable depends on its past, capturing the trajectory each variable takes through time. In our case most points tend to be along a diagonal in the center. This seems logical given we are working with continuous time series data, which does not vary excessively from point to point, even less when considering our sample rate of 10 minutes. So the values of K_{t-1} are close to those of K_t regardless of K's value, giving us this diagonal shape (data was scaled for correct visualization). Distances between points on the manifolds are calculated to find nearby points in terms of temporal dynamics. For several time steps, the nearest points on the opposite manifold are identified. These are visualized with markers: blue squares for the original and red stars for the corresponding points on the opposite manifold. Lines (red dotted) connect these pairs to indicate the prediction or mapping from one to another. The process aims to visualize how well one variable can predict the other by using past values. The cross mapping aims to test if knowing the path or trajectory of X can help predict where Y will be in the future, and vice versa. By mapping points from X's path to Y's path, we can see how much X influences Y and whether there's a potential causal relationship. To interpret this we have to look at the red dotted lines that connect the pairs, and asses whether values from the starting manifold, have either a narrow or spread out cross mapping to values on the other manifold. In the first case, it may be implied that X influences Y, while the second may mean that not much information is transferred from the input to the output. We can see in our case dotted red lines stemming from the blue boxes travel to a limited number of closely located points. In the TM M_x-M_y map, we see some points travelling further away but keeping clear narrow mappings. In its side M_y-M_x map, single blue boxes spread out more diversely, suggesting that the effect blood glucose has on time, is not as strong the one time has on glucose, as should be. [76]

Although cross mappings that tend to be narrow strengthen the hypothesis of causality, it is a non-straightforward and qualitative method of understanding the effect of the inputs on the outputs, which is also analyzed patient by patient. More importantly, it is just: "one of two criteria to conclude causality" [76]. We also need convergence. 'Convergence', as we use it in this section, refers to the principle that the predictive power of the cross-mapping method improves as we increase the amount of available historical data L provided to make the reconstruction. The underlying idea is that a longer time series provides a more complete view of the dynamics of the system, allowing for a more accurate reconstruc-

tion of the ”shared attractor”. This shared attractor is a concept where two or more variables in a system converge in a common, complex pattern of behavior over time, despite their distinct trajectories. This shared attractor represents the underlying state space that both variables influence and are influenced by. The interpretation in this case is much simpler than with the previous cross mappings, and just requires the correlation **stabilizing** at a positive value, as length of the used sequence increases. This criterion focuses on stabilization as the goal rather than just an indefinite increase in correlation. This comes from the fact that adding more points, once the attractor is sufficiently reconstructed, should not alter its structure. Basically, accuracy should stabilize, and new data should not contribute to significant changes in the geometrical properties of the attractor because the representation of the system dynamics is complete. In Fig. 30, this is demonstrated for all the patients from the test set, for the three input variables. We can go further and look at the stabilization values, seeing that RA tends to have the highest correlation stabilization value, over PI. Unsurprisingly, time has the lowest causation on the output. Actual correlation values from the patients are presented in Tables 9, 10, 11, which are coherent with the analysis of the convergence plot. The higher the correlation value, the higher the causation (yes, in this case, correlation means causation). For cases where the p-value is below the 0.05 threshold, the causality is significant. This happens for all patients, for the three types of inputs with extremely high degrees of significance.

7.1.3 Data Distribution Representation

In order to represent the empirical distribution of the data we show the results from performing a PCA and a t-SNE analysis using 2 components (or dimensions) in 34, 35, 36. Principal Component Analysis (PCA) and t-distributed Stochastic Neighbor Embedding (t-SNE) are two well-known techniques used for dimensionality reduction, facilitating the visualization and interpretation of high-dimensional data in a lower-dimensional space. PCA is a linear technique that identifies the directions (principal components) along which the variance of the data is maximized. It is particularly useful for highlighting the global structure of the data. On the other hand, t-SNE is a nonlinear technique that preserves local relationships and is useful to illustrate clusters within the data. These techniques are also used in other time-series GAN works, such as Zhu et al.’s [77] GluGAN to visualize the distribution of the data.

To prepare the 1D glucose data for these analysis, the series is first transformed into a higher-dimensional space using delay embedding, based on Takens’ Theorem, which states how lagged versions of a single time series can serve as substitute variables to reconstruct an attractor for the underlying dynamic system [78]. We reshaped the data into a matrix of size (N, dimension) where N is the number of samples in the original data. This will allow us to perform the PCA or t-SNE on the reshaped data as we cannot perform dimensionality reduction on a 1D vector. The dimensions we choose will affect the amount of compression the techniques enforce as they ’squeeze’ the data into a 2D space. If the selected dimension is 2, the number of dimensions is not compressed, and data is just transformed with a change of axis that can make patterns more apparent, which is what we want. It can be observed how real and synthetic distributions ($\mathcal{D}_R$, $\mathcal{D}_G$) are very highly overlapped, 34, 35, which suggests that

even without specialized temporal layers, our GAN successfully approximated the underlying distribution of the glucose time series data p_R. GluGAN [77] for example used Recurrent Neural Networks (RNNs) to model the time series obtaining less overlapping distributions compared to our GAN, which used only a simple 1D Convolutional Neural Network (CNN) to act as a moving average filter. We would expect temporal dynamics to be hindered by this more simplistic approach. These satisfactory results could also strengthen the hypothesis that the inclusion of time in the model and the shifted inputs of carbohydrates and insulin boluses, is helping the model learn these temporal relationships. Referring to Fig. 36, representing a t-SNE analysis on test patients 5 through 14, we can observe how the real distribution $\mathcal{D}_R$ looks 'incomplete' on its left side, likely due to the absence of corresponding points in those samples. Nevertheless, we see how the generator is able to reconstruct that missing part realistically in $\mathcal{D}_G$. This observation shows why GANs have been extensively used for data augmentation tasks in recent years.

Latent Space

In our model, a 3 dimensional Latent Space was used, this small size is enough provided we are working with 1-dimensional data. Typical uses of GANs include image generation, which may require using 100 dimensions or more. The LS serves as a compact representation where different regions may correspond to different features or embedded information of the data being modeled. A bigger latent space allows a better separation of the information encoded, but a space too big to handle may not allow any region to represent anything meaningful. The LS is an essential part of GANs, as it represents its source of stochasticity. A normal NN, once trained and provided no drop-out layers are used during inference, is deterministic. And as discussed earlier, we want to include stochasticity as a way to capture the swarms of untrackable factors influencing glucose homeostasis.

In Fig. 37 we can see different representations of the latent space samples and how the points change if we truncate the bell curve. In the middle plot, we show p_Z truncated at ± 0.1, whereas in the right one we do the same at ± 1, and then we take the complementary, or external part. The effect on the empirical distributions is clear: the PCA for the generated data with the truncated LS is densely packed in the center, not overlapping with the real distribution $\mathcal{D}_R$ in the outer regions, and taking few extreme values with a maximum around $PC1 = 300$. Meanwhile, the PCA plot for the complementary sampling is more sparsely populated in the center, and has many more points outside the range of the original data, with a maxmium at $PC1 = 800$. On the t-SNE side, we observe that the truncated generations tend to concentrate around the center and along the irregular line that goes from end to end of the shape. In the complementary t-SNE plot, the middle part is close to empty, with most points heavily accumulated at the right and left extreme ends. This tells us that the truncation and complementary sampling methods can alter the probability distribution p_G of the generator, affecting the structure, variability and spread of $\mathcal{D}_G$. Deeper study of the LS could allow control over the outputs using specific sampling techniques.

7.2 Challenges and Limitations

Data

Data is justifiably considered the backbone of AI and deep learning. Fortunately, this project enjoyed plenty data to train its model. However, not only size is important but also quality. As seen in the preprocessing, from the 15 million samples gathered initially, only little over 3 remained ready to be used after the process. This happened due to the intense cleaning of large sequences of missing data, especially blood glucose, but to an extent also of CHO and insulin. A missing BG sample is spotted right away, as it is continuously measured, or at least intermittently every 5 minutes. However, insulin or carbs can come at any time, or at no time, so there is no straightforward way to identify missing inputs. The patient may have really missed a meal, or the meal may not have been annotated. To work around this uncertainty, an average of 2 insulin boluses and 2 meals per day was used to filter out patients.

It is possible that there exist some scenarios where food was taken alongside insulin boluses, but only one of them was annotated. This can be especially confusing for the model if carbohyrdates were missed, as an increse in insulin is accompanied by a hyperglycemic event without food in place. During analysis, this is difficult to assess or differentiate from normal glucose fluctuations, so no actions could be taken to mitigate this data quality problem. Moreover, the complete input of all mealtimes does not entail correct input of rate of appearance quantities. First, because as humans we can be especially bad at counting CHO quantities from a meal just by looking at it. Second, because we are making further assumptions by using a set of differential equations from the Hovorka model (for both carbohydrates and insulin), that might represent dynamics different from the ones shown by the different meals (with a range of different Glycemic Indexes), or by our patients. Blood glucose did not need human intervention to be recorded, but that does not prevent it from measuring errors, as CGM's have their own set of limitations themselves, some even disconnecting for variable amounts of time which where reconstructed using cubic splines interpolation, a good but imperfect imputation method.

While this abundance of data for the GAN model is advantageous, the pursuit of a populational model introduces its own set of challenges. Aggregating data from a diverse patient population allows for a more comprehensive understanding of blood glucose dynamics across different individuals. However, it runs the risk of diluting specific patient behaviors in the process. By averaging out individual variations, there's a possibility the the model's performance may suffer, leading to bad predictions for certain scenarios from uncommon patients. Moreover the dataset itself is problematic, missing key information like basal rates, and containing few control actions outside the meal events.

Model

The implementation of a Wasserstein CGAN with approximately 3.7 million parameters introduces several limitations and challenges. Firstly, the increased complexity of the model significantly elevates the computational requirements and training duration. The vast number of samples, even after preprocessing was enough to make state-of-the-art NVIDIA GPU have a hard time. Training with the ideal batch

size of 1, for a reasonable 50 epochs, assuming an optimistic constant speed of 70 epochs/min, would take over 1,500 days of continuous training runtime. As each sample would need to be seen individually, each epoch would consist of 3,111,479 steps, one for each example. It is clear we needed to increase significantly this batch size, which is not optimal. However, this was the only way to fit the training time inside the reasonably acceptable range. With a batch size of 64 and 20 epochs, still 10 days are needed to finish the training. These exorbitant demands for computational resources and training times, made it very tedious and slow to explore various model architectures and hyperparameters. With changes being made week by week, leading to many dead ends.

This problem was exacerbated by the inherent instability of training the adversarial nets. GAN training is notoriously challenging due to the balance between the generator and discriminator networks. The discriminator may overpower the generator, causing it to fail to produce realistic samples. Or we may experience 'mode collapse', a common problem where the G starts creating a limited variety of outputs, which succeed in fooling the discriminator. 5 days of good progress in training, can suddenly go into mode collapse, and require restarting from scratch. Moreover, interpreting the loss functions is not trivial. Traditional metrics like cross-entropy loss are not directly applicable due to the adversarial nature of the training process, making detecting this phenomena not always possible. The Wasserstein distance used in WCGANs provides a more stable training signal than traditional GAN losses, but it still requires careful monitoring and interpretation. Attempting to apply early stopping or other techniques for hyperparameter tuning becomes challenging too in this case. The usual indicators of convergence or divergence may not be reliable due to GANs' oscillatory nature. Loss curves may exhibit sudden spikes or plateaus, making it difficult to discern whether the model is making progress or stuck in a local minimum.

Overall, training a Wasserstein CGAN model with millions of parameters and samples, presented a formidable computational and interpretational challenge. The complexity of the model, combined with the inherent instability of GAN training, required careful experimentation and patience to achieve satisfactory results.

Validating the Results

Once the data is ready, the model is trained and the obtained generator works, the work is not yet over. Validating the results of this CGAN model presents more unique challenges, nothing less intricate than the previous parts, particularly as we are generating data that resembles the distribution of original patient data $\mathcal{D}_R$, but not necessarily the curves themselves. Unlike methods that compare data point-to-point with established metrics like mean absolute error, GAN-generated data might not precisely match the original data in a direct manner. That is not the goal. Assessing whether a generated curve is physiologically sound can be straightforward, using benchmarks such as times in ranges or response to carbohydrate intake or insulin boluses. However, this is not enough to tell if the generator has learnt a patient's probability distribution p_R and is replicating his physiology correctly in order to predict long-term outcomes, which is the goal.

The intricacies of individual patient profiles, including interpatient and intrapatient variability, further complicate validation efforts. There is no general way of identifying distinct patient characteristics within generated data. One of the fundamental challenges lies in the implicit representation of probability density functions, which is embedded within the weights and biases that convert the latent space sample to the output. As said, through the random sampling from this LS, the network introduces stochasticity at each new step, resulting in a different outcome with each generation. Unlike explicit representations of probability distributions, which can be directly analyzed, GANs' implicit nature requires alternative validation strategies. Testing presented particularly challenging as just as before, any process involving such a dataset is computationally demanding. E.g. in fixed labeled simulations, given TM's effect is much smaller than that of PI and RA, coupled with the stochasticity of the generation, it is difficult to know if the different results would be consistent if we had run the simulations numerous times.

7.3 Simulating in the Clinical Setting

The ultimate goal of developing this outcome predictive model is to use it in a real-world clinical setting, reaching the end users in the form of a simulator. Such simulator would enable forecasting blood glucose control outcomes over an extended period of time. To achieve this, Open Loop (OL) and Closed Loop (CL) control methodologies could be used. In an OL system, the inputs to the model would be a generated scenario composed of the meals in the form of grams of carbohydrates, and its corresponding bolus. Based on the grams of carbohydrates consumed, current blood glucose levels, and individualized correction factors (CF) and carbohydrate ratios (CR), the simulator would compute the required insulin bolus for each meal, using a simple formula. This method however, does not adjust insulin delivery in real-time based on glucose readings, like most T1DM patients do. Conversely, CL control systems incorporate a feedback loop that adjusts the insulin delivery based on the glucose readings, or can suggest the patient to take a rescue carbohydrate if a hypoglycemia is predicted. Both setups would require a preliminary analysis to adjust the parameters of the control algorithms so that their style of operation resembles that of the real patient. This would include factors like body weight, CR, CF, hypo- and hyperglycemia response thresholds, basal insulin (I_b), eating habits, and more. I_b in particular, needs special calibration by finding the value for which blood glucose stabilizes into around the desired set point.

We can imagine a situation in a typical clinical setting where a doctor encounters a T1DM patient with suboptimal glucose control. By having the model tuned to the patient's historical data, the physician will have the possibility to explore various therapeutic modifications such as adjusting the basal rate of the pump, increasing the CF, or altering meal timing and composition. The simulator allows the doctor to virtually test different combinations of these changes and observe the predicted outcomes, aiding in formulating a more effective, tailored treatment plan. Ultimately, while the simulator provides data-driven recommendations, the final decision always rests with the healthcare provider.

8 Execution Cronogram

Execution cronograms offer visual roadmaps for our project timeline and tasks. They are vital for efficient project management.

8.1 WBS

The Work Breakdown Structure overview of the project is presented in Fig. 41. The details of each task are further broken down in Annex 3. The personnel costs from section 10.2 are used, for a single person working on the project.

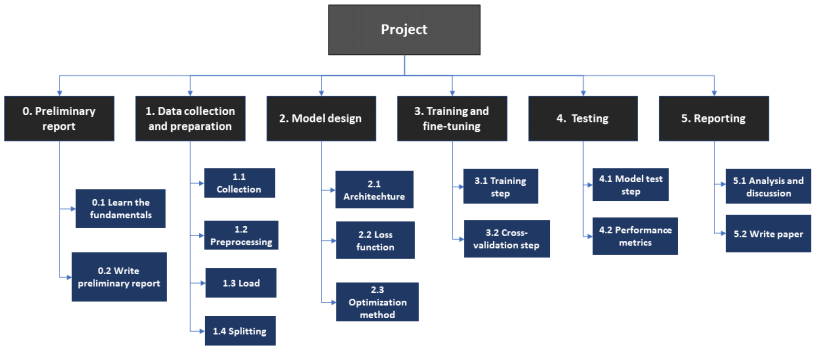

Figure 41: WBS of the project

8.2 PERT & GANTT

The Program Evaluation Review Technique (PERT) is a project management planning tool used to estimate the amount of time it will take to finish our project. The red path in Fig. 42 is the 'critical path', which represents the sequence of activities that determines the minimum duration required for completing the project, where any delay in those activities will directly impact the overall project timeline.

The GANTT diagram in the other hand is a bar chart that represents a project's schedule. It is built from the other cronograms. It can be appreciated how the critcal path coincides with the one of the PERT in Fig. 45.

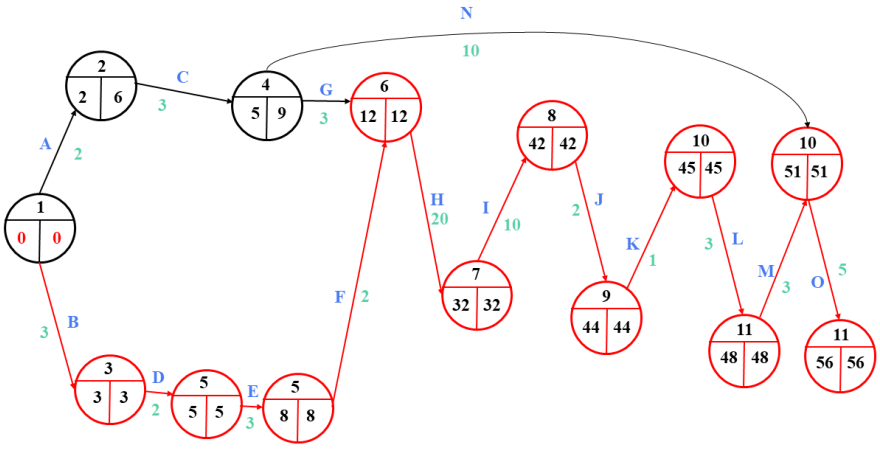

Figure 42: PERT of the project with critical path outlined in red

Task	Precedent	Duration (work days)	Task	Precedent	Duration (work days)
A. Meetings	-	2	H. Training	F, G	20
B. Literature review	-	3	I. Cross Validation	H	10
C. Conceptualization	A	3	J. Exploration	I	2
D. Dataset study & selection	B	2	K. Performance metrics test	J	1
E. Preprocessing	D	3	L. Other validation techniques	K	3
F. Analysis	E	2	M. Discussion	L	3
			N. Write final tesis	C	10
G. Design	C	3	O. Prepare presentation	M, N	5

Figure 43: PERT tasks precedents

PROJECT	Developing a GAN-Based Blood Glucose T1D Outcome Prediction Model for Clinical Use	DATE	01/02/2023
CLIENT	MICELAB	REVIEW	JOSEP VEHÍ & ORIOL BUSTOS
ORDER Nº	1	AUTHOR	ORIOL BUSTOS

START DATE	2/1/2024
END DATE	5/6/2024
DURATION (DAYS)	67.00

Automatic cell completion

Figure 44: Head of the GANTT spreadsheet

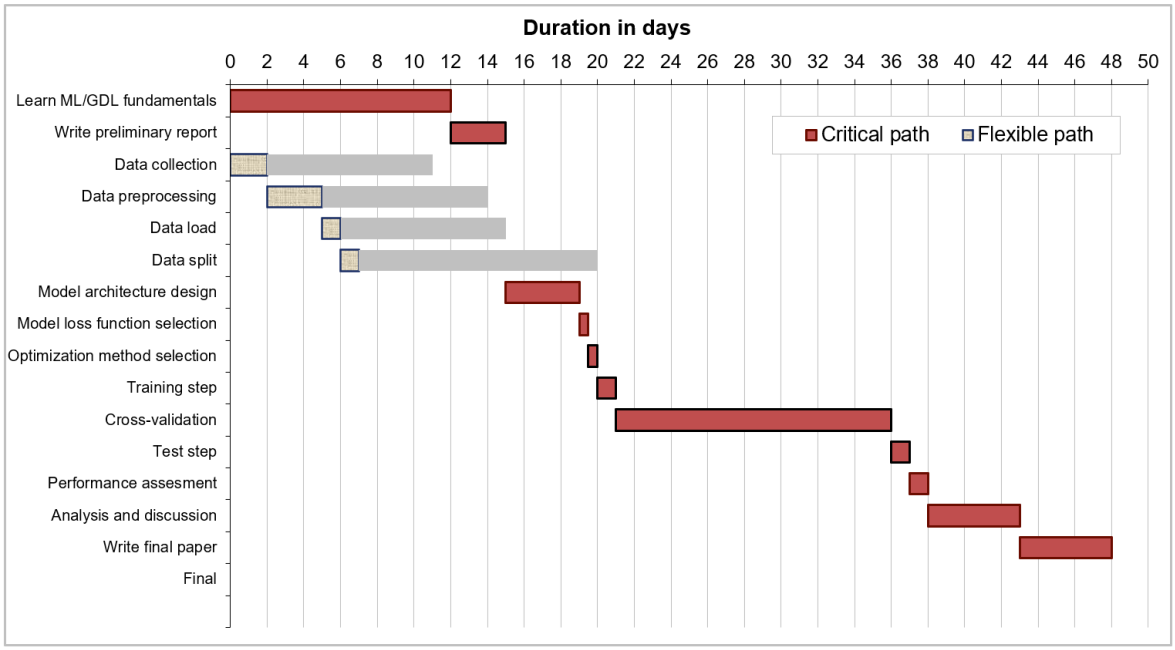

Figure 45: GANTT diagram of the project

8.3 Milestone Plan

In Fig. 46 we can see the major milestones that constitute the project, they can be composed by several deliverables.

MILESTONE PLAN

PROJECT:	Developing a GAN-Based BloodGlucose T1D Outcome Prediction Modelfor Clinical Use
CLIENT:	MICELAB
AUTHOR:	Oriol Bustos DATE: 05/02/2024

ID	DESCRIPTION	RESPONSIBLE	DATE	APPROVED	DELIVERABLES INCLUDED
1	Definition of the objectives and methods to conduct the project	Oriol Bustos	06/02/2024	Josep Vehí & Oriol Bustos	0.1 Preparation
2	Implementation of the model	Oriol Bustos	03/04/2024	Josep Vehí & Oriol Bustos	1. Data, 2. Model implementation
3	Validation of the model	Oriol Bustos	10/05/2024	Josep Vehí & Oriol Bustos	3. Testing
4	Thesis document hand-in	Oriol Bustos	01/06/2024	Josep Vehí & Oriol Bustos	4.1 Discussion, 4.2 Wrie final thesis
5-	Presentation of the thesis	Oriol Bustos	10/06/2024	Josep Vehí & Oriol Bustos	4.3 Prepare presentation

Figure 46: Milestone plan of the project

9 Technical Feasibility

We'll explore the project's feasibility from the technical viewpoint, understanding why we have the necessary technical capabilities and resources to deploy a deep learning model, execute and maintain the project successfully.

9.1 Infrastructure and Resources

Running deep learning models requires considerable computational power, storage, and technical expertise. Our deep learning model has been trained and deployed through a new high-performance PC assembled using a range of cutting-edge components to ensure efficient and effective model training and execution. At the core of the system is the AMD Ryzen 9 7900X processor, which runs at 4.7 GHz [79]. This state-of-the-art processor can handle large-scale computational tasks and is capable of parallel processing, making it well-suited for deep learning applications. The processor is supplemented by an efficient cooling system, the Corsair iCUE H115i RGB ELITE Liquid Cooler, to prevent overheating during intense computation. The PC is equipped with a pair of Corsair VENGEANCE DDR5 RAM in dual channel with 16GB each, clocked at 4800MHz. This memory storage and speed will allow for efficient data handling, crucial for real-time model training and prediction. Additionally, our storage solutions include a Seagate BarraCuda hard drive and a Crucial P3 Plus 1TB M.2 drive. The former provides sufficient storage of 2TB for raw datasets and intermediate data products, while the latter offers superior read/write speeds, facilitating faster access to data and scripts The selected motherboard is the Gigabyte B650 AORUS ELITE AX, which was chosen for its reliable performance and compatibility with our chosen CPU and memory. It also provides a solid foundation for future upgrades, ensuring the longevity of our deep learning system.

The entire system is powered by the EVGA SuperNOVA 850 GT, an 80 Plus Gold certified 850W fully modular power supply. This will provide stable and sufficient power for all the components during peak operation, ensuring consistent performance. The deep learning models are deployed using the Zotac Gaming GeForce RTX 4070 Ti Trinity OC graphics card. The card's 12GB GDDR6X memory and DLSS 3.0 support make it ideal for handling the parallel processing needs of deep learning algorithms. Additionally, the Nvidia CUDA cores in this GPU are leveraged for their high-speed performance in machine learning tasks. Our system is housed in the Corsair iCUE 4000X RGB case, which offers excellent thermal performance and space for future expansion. This case ensures optimal operating conditions for all our hardware components. The Asus ProArt Display is used for visual output. This 27" LED IPS QHD display, coupled with FreeSync USB-C, enables high-resolution for long working hours.

The programming framework includes the open-source coding language Python and common libraries such as `matplotlib`, `pandas`, `numpy`, `TensorFlow`, `Keras` and `Sci-kit Learn`. The different versions of the code have been stored in a Gitea repository, which uses `git` version control, located in Micelab's Synology Server. The documents have been written in `LaTeX` using VSCode, compiling with `xelatex`. We can conclude that the project is **feasible** from a technical viewpoint as we have acquired datasets with the needed features, and the DL algorithm training is backed up by a powerful PC to compute the adjustments of the model. The robustness of this computing setup enables us to carry out our project accordingly.

9.2 SWOT Analysis

The SWOT framework stands as a powerful tool for assessing a project's strengths, weaknesses, opportunities, and threats. By understanding these key elements, we can gain valuable insights to make informed decisions and navigate future uncertainties with confidence.

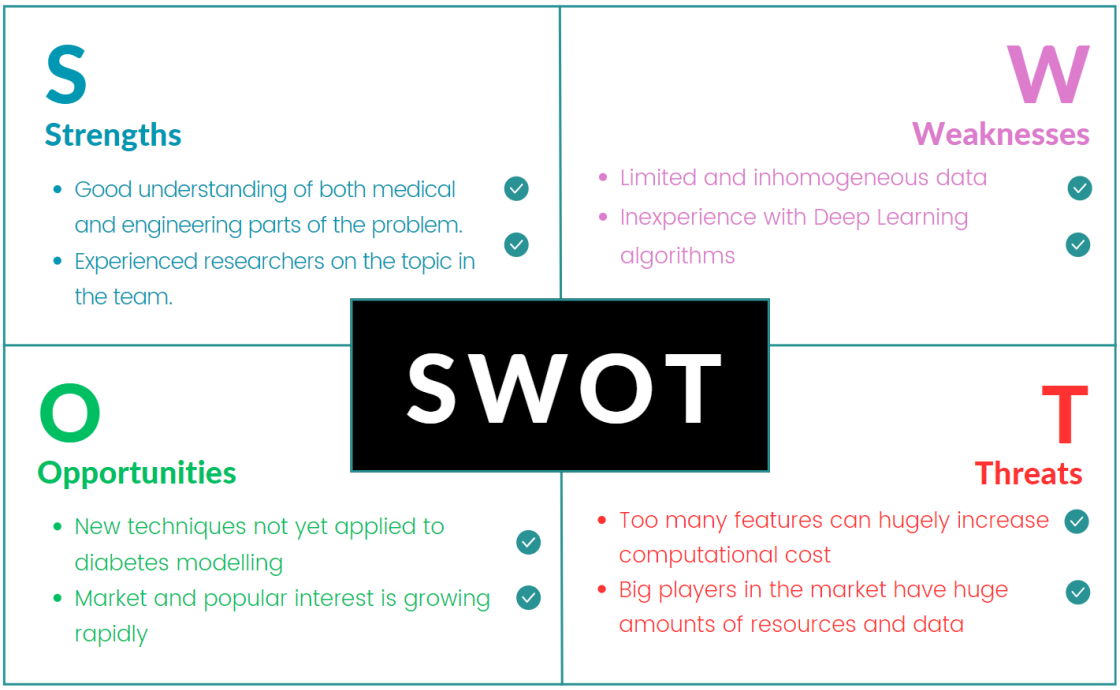

Figure 47: SWOT Analysis of the project

10 Economical Feasibility

We will now evaluate the project's potential for financial success, and capability of being carried out within the budget, while offering potential for return on investment (ROI), in our case non-financially.

10.1 Research Group Budgeting

Micelab is a group from the Institute of Informatics at University of Girona (UdG). The lab is thus located in UdG's Parc Científic i Tecnològic. It is currently involved in several projects, 6 funded by the "Ministerio de Ciencia e Innovación" from the Spanish government, and one funded by the European comission. Overall, the group has been awarded over 2 million euros in competitive funding for research projects focused on diabetes-related technologies [80]. These investments provide the necessary financial support for the experienced researchers working in the team and for the acquisition of expensive hardware. A partnership with the diabetes management app SocialDiabetes has received allocation from the Next Generation funds of the European Union, through the Recovery, Transformation, and Resilience plan of the Ministry of Science and Innovation [81]. Recently, the research group has been nominated to hold the chair in artificial intelligence at University of Girona, partnering also with Dexcom. The lab will receive more funds with the objective of developing new formative activities, dissemination, and knowledge generation and transfer, through research and activities.

10.2 Project Budget

As aforementioned, the primary objective of this research project is not to generate a financial ROI. Instead, we aim to contribute significantly to the scientific community, offer substantial public benefits, and effectively manage public resources, all within a predefined budget. The project, then, requires an initial investment for the hardware acquisition, which will be useful for future deep learning projects in the lab. The data source for this research will be pre-existing datasets from prior clinical studies, which do not entail additional initial expenses. The acquisition costs are outlined in Table 12. Then, the variable costs include those associated with software, depreciation of the computer and personnel. The software used, Python, along with its library TensorFlow, is open-source, and thus, cost-free. Regarding the workforce, only one biomedical engineer will be working full-time at a rate of 15€/hour. The development costs are outlined in table 13.

The useful life of a PC for depreciation purposes is commonly estimated to be between 3 and 8 years [82]. Considering we have high-end quality materials, but at the same time intensive use for the

Table 12: PC Acquisition Costs

PC Component Name	Price (€)
Assembly by *PC componentes*	37.18
Logitech Desktop MK120 Keyboard and mouse	18.93
Asus ProArt Display PA278CV 27" LED IPS	343.79
GeForce RTX 4070 Ti Trinity OC 12GB GDDR6X	767.77
EVGA SuperNOVA 850 GT 80 Plus Gold 850W	169.41
Corsair iCUE 4000X RGB Cristal Templado USB 3.1	120.65
Seagate BarraCuda 3.5" 2TB SATA 3	39.17
Crucial P3 Plus 1TB SSD M.2 3D NAND NVMe PCIe 4.0	60.66
Corsair VENGEANCE DDR5	100.83
Corsair iCUE H115i RGB ELITE Liquid Kit	141.31
Gigabyte B650 AORUS ELITE AX	217.35
AMD Ryzen 9 7900X 4.7 GHz	393.38
Canon Digital 182592 and 10587442	10.90
Total PC Cost	**3056.63**

PC, we'll consider a midpoint at 5 years of useful lifetime. The PC is expected to be used 5 days a week, accounting for a total of roughly 240 workdays in a year (assuming 2 weeks of holidays). Therefore, over 5 years, we estimate a total of 1,200 useful days.

The depreciation cost of the PC will be its initial cost divided by its useful life in days. This provides a straight-line depreciation method, which is the most common method used. For a PC with an initial cost of 3056.63€, we can calculate the daily depreciation as follows:

$$\text{Depreciation cost per day} = \frac{\text{Initial cost}}{\text{Useful life in days}} = \frac{3056.63\ €}{1200\ \text{days}} = 2.54\ € \text{ per day}$$

Based on our GANTT execution diagram in Fig. 44 we know the project is estimated to last 67 full-time working days. In Table 13 we can see the costs associated with running the project. The acquisition costs of the computer have not been included in this table as it hasn't been purchased uniquely for use in this project, and would not be representantive.

Table 13: Development Costs

Concept	Price (€)
Labor Cost (329 hours (EDT) x 15€/hour)	8040.00
PC Depreciation Cost (67 days at 2.54€/day)	170.18
Total Project Cost	8210.18

We can conclude that Micelab's funding from both the European Union and the Spanish government, make the adquisiton of the PC and the development costs bearable, and therefore the project is economically feasible.

11 Regulatory Affairs

In this section we outline the relevant regulatory framework: the deployment and use of medical devices as software, AI, and data protection. As well as discussing the practical and ethical risks associated.

11.1 Medical Device Regulations

MDR The MDR 2017/745, is a legislative framework designed to supervise the safety and effectiveness of medical devices within the EU. This regulation introduces a more rigorous post-market surveillance system, increased transparency, and a reclassification of medical devices based on their associated risks, leading many devices to be categorized at a higher risk level.

For companies introducing new medical devices, including AI models, several considerations have to be accounted for. The MDR emphasizes strong clinical evidence, asking manufacturers to undergo thorough evaluations that not only demonstrate safety and performance but also require periodic updates. It also mandates the implementation of a Unique Device Identification system to allow device traceability and the aforementioned post-market surveillance system. Additionally, there are stricter labeling requirements to ensure clear communication to patient-users, and higher responsibilities for all parties involved, which could potentially impact supply chain relationships [83][84].

SaMD Software as a Medical Device (SaMD) refers to software intended to be used for medical purposes without being part of a hardware medical device. Unlike the MDR, SaMD specifically targets software. Both frameworks use a risk-based approach, categorizing devices based on the potential harm to patients. Yet, they diverge in their regulatory details, with the MDR having a more region-specific framework, while SaMD guidelines aim for a more harmonized, global approach [85].

ISO/TS 82304-1:2016 - Health Software This standard addresses the general requirements for the safety and performance of health software products. Specifically, it focuses on standalone software applications intended to operate in a healthcare environment, either as a medical device or a software accessory to a medical device. The specification emphasizes the need for risk management, usability, and software lifecycle processes, including development, maintenance, and post-production. By adhering to ISO/TS 82304-1:2016, safetiness and reliability of the model would be certified [86].

11.2 Data Protection and Privacy

Data in Europe: GDPR

Strict compliance with data privacy regulations such as the General Data Protection Regulation (GDPR) in the EU is essential for any data-related project [87]. The GDPR is a comprehensive legal framework instituted by the EU to regulate the collection, storage, and processing of personal data. The central principles include:

1. Ensuring legality and transparency in data use
2. Obtaining explicit user consent
3. Data minimization to only the necessary information
4. Scope definition
5. Ensuring data accuracy
6. Implementing secure storage mechanisms, and safeguarding data integrity and confidentiality.
7. Defining and limiting the conservation term of the data

To comply with these mandates, this project uses publicly available anonymized data, adopting secure data storage locally in a Synology Drive server which can only be accessed within the private Micelab network.

Data in the US: HIPAA The Health Insurance Portability and Accountability Act (HIPAA) is a US-specific regulation introduced in 1996. While the European GDPR broadly governs personal data of EU citizens across all sectors, HIPAA specifically targets the protection of health-related information, ensuring the confidentiality, integrity, and availability of health information. Additionally, it regulates portability of health insurance and healthcare fraud which are outside the reach of GDPR.

11.3 AI: Ethics and Regulation

GDPR for AI The GDPR normative is a regulatory framework that, as seen, considers the legal aspects for all data-related issues. However, AI requires a specific adequation of the guidelines as presented in ”Adecuación al RGPD de tratamientos que incorporan Inteligencia Artificial” [88] by the AEGPD from Spain, where the group is based. It concerns how AI technologies and systems can be designed, developed, and utilized in compliance with these broad-reaching data protection standards. The RGPD protects individual data privacy, which becomes especially critical when applied to AI technologies that may process large volumes of personal data. Achieving adequation means not only meeting the explicit technical and organizational requirements around data processing set out in the RGPD, but also addressing the more abstract principles such as fairness, transparency, and accountability.

11.4 Thesis Approval, Risk Management and Responsibility

This subsection discusses the inherent risks associated with the model and the measures taken to mitigate them. The predictor is specifically designed to output BG profiles resembling the behavior of T1DM patients to assist in therapy change decisions. Importantly, the model is not intended to predict acute events such as hyperglycemia or hypoglycemia, which require immediate medical attention. Patients should not rely on this tool for critical short-term treatment decisions or for anticipating immediate fluctuations in a patient's glucose levels.

While our training process and model design may prove valuable for a particular subset of T1DM patients, its performance will not generalize to all T1DM individuals or to those with other types of diabetes. Like all models, ours is based on the data it was trained on and the assumptions it was built under. Variabilities in lifestyle, genetics, and other health conditions can influence glucose behavior, and these may not be entirely captured in our model. The quality of patient data is vital to the performance and reliability of predictions. Erroneous or incomplete patient data can lead to misrepresentative BG profiles, potentially affecting therapeutic decision-making.

Finally, our model serves as an adjunct to, not a replacement for, the expertise of a qualified healthcare professional. Decisions on therapy adjustments should always be made with a deep understanding of the patient's condition, including clinical context, recent events, and potential external influencing factors which may not be included in our model.

Clinical Trials and Approval Regarding data gathering, the project did not require a clinical trial as it used prior projects' data or existing datasets publicly available on the internet. Addressing the issue of safety, it is worth noting that any large-scale deployment of the model involving real patients would require a separate clinical trial.

Principles and Spanish Legislation This project not only adheres to the Integrity Code in Research, like all UB Final Degree Projects, but also seeks to align with the specific regulatory frameworks stipulated by Spanish law. Regarding data, based on article 16.3 of Law 41/2002, of 14 November, the basic Spanish law regulating patient autonomy and rights and obligations regarding clinical information and documentation, it is legitimate to process personal medical data without direct consent from the individual, as long as the data is either anonymized or pseudonymized. Despite that, we are not using patient data from the hospital but rather taking publicly available datasets, which are already anonymized. Moreover, Spain has not issued specific guidelines for the classification of software as a medical device, thus the guidelines issued by the European Commission are recommended for evaluating whether certain software qualifies as a medical device [89].

12 Conclusions

In order to conclude the project, we will first go back to the start and assess whether the objectives we aimed for were fulfilled.

The first goal was to develop an outcome prediction data-based model for a cohort of diabetic patients using a Generative Adversarial Network. This objective is completed as we have successfully built a generative deep learning model that correctly predicts the outcomes for the main blood glucose control metrics. The t-test proved significant similarity between the synthetic and real patients in terms of several key metrics including mean, coefficient of variation and time in range, among others. The most extreme ones, being time under 54 and over 250, present the most significant deviations (see Table 7).

The second objective, states that it should generate plausible blood glucose profiles from the physiological point of view. This objective is completed as curves show physiological behavior in meal situations, as insulin has a glucose-lowering effect while high-carb meals result in hyperglycemias. Some examples (see Section 6.1) show how qualitatively similar profiles can potentially be, despite this not being the main purpose of the model.

The third one was to guarantee the generated profiles were caused by the past and present conditions of insulin, carbohydrates, and time values. This objective is completed as the model includes all the effects listed, using present and up to 1 hour back into the past information (Fig. 14), and all of them show statistically significant correlation using the CCM causality test (Tables 9, 10, 11).

For the last goal, which included modeling circadian rhythms using the moment of the day, we can say the effect of time has been included and proven to have an effect as already mentioned in objective three, and seen in subsection 6.2. However, the magnitude of this effect seems unclear. This is due to the mixed results that simulations using fixed labels of time showed, with no apparent improvement when the used label coincided with the real one, compared to using any random label (Table 8), likely due to the averaging out of individual circadian patterns by the populational nature of the model.

Regarding the implications, this project serves as a proof of concept for both the GAN-based prediction model and the possibility of including time as an input. Previous works have already tested the ability of GANs to succeed in this task [42], but not with as many patients as we have used here. The addition of time is unprecedented in this type of model. Despite not ideal results, its feasibility has been proven, without the increase in network complexity worsening the actual final curves' quality.

Key Takeaways & Future Directions

This project presents several contributions which should definitely be build upon in projects coming ahead. Many of the points commented here are deeply discussed and put into further context in Section 7. This implementation saw many changes introduced together from the ground up, rather than

incrementally, which makes it challenging to isolate the impact of some of the adjustments.

The preprocessing approach, while thorough, might have been overly strict. In hindsight, less fragmentation of patient data into small subpatients, keeping sequences of at least 15 days, could have been better. Adopting MinMax scaling on an individual basis, rather than a global approach for all patients, considerably improved the outcomes, suggesting it could be a potentially very useful contribution to future GAN-based populational models by ensuring consistent data scales. Another notable modification was the reduction of the sampling rate to 10 minutes, which likely reduced the computational burden of training, as it halved the number of samples compared to the field standard of 5 minutes. Additionally, the model now incorporates three extra past values of both insulin and carbohydrate, instead of just the current one, alongside with the one-hot encoded current time of the day (Fig. 14). This change of sampling rate, and the expansion in input data did not compromise the feasibility of training or degrade the model outputs, maintaining robust results for ten fold more data than previous works. Despite this, we need higher quality datasets containing more diverse scenarios, and that do not lack infusion rates.

Another key take away is that clustering may become essential as datasets start growing and including more heterogeneus data. These techniques should be investigated to better understand patient subgroups, and allow a more accurate modeling of relationships that happen by learning from more homogeneus patient data. This could lead to more personalized glucose predictions and therapy recommendations, enhancing the clinical utility of the model. Symmetrizing the blood glucose range could also potentially improve outcomes at the lower range of blood glucose levels, the hypoglycemic zone, by aligning the numerical and clinical centers and balancing the emphasis on hypo- and hyperglycemic conditions [75].

Findings showed improvements when deviating from the literature-suggested learning rate and generator's loss ratios (opting for 5×10^{-6} and a [1,5] ratio, respectively, over the original 5×10^{-5} and [1,100]). Anyhow, more parallelization will be needed to effectively explore the different configurations in reasonable time frames. Developing this research was notoriously slow, and each change in the configuration took several days to take effect and possibly correct them. More computing power would accelerate the explorations of different configurations, and allow finding more effective architectures. The final model utilized 3.6 million parameters, a middle point in the complexity and computational demand within the field. Interestingly, fewer iterations than initially anticipated were executed, achieving satisfactory results in just 5 of the planned 20 epochs. This was evidenced by the loss functions logs, particularly the increase in L2 loss, which was vital in determining when to do the early stopping. (Summary in Table 4).

The model exhibited a strong understanding of glucose homeostasis dynamics. Despite this, introducing synthetic examples of underrepresented scenarios in the data, or using physics-informed NN methods, could further guide it and improve its capabilities. Although the output ranges were consistent across synthetic patients, significant scaling adjustments were necessary to mimic actual glucose curves. An optimal scaling factor was determined using the validation set, which was then successfully

applied to the test set. Coming research should prioritize minimzing the need for scaling, by ensuring the output range already matches the real data.

Time variability was embedded into the generator's layers, hinting at phenomena such as the dawn effect during the nighttime window in the AGP report (Fig. 18). However, as aforementioned results have room for improvement (Fig. 7), possibly with the use of clustering, and/or by a different selection of the time windows. In the future, the initial design of the networks and selection of input variables could be guided by further analysis of causality. In our case causality tests like Convergent Cross Mapping proved to be very valuable as a validation strategy, yet they are not sufficient and additional methods need to be developed to ensure we can assess and compare results reliably. Inspired by the very foundations of this work, a discriminator-like validator could be trained to distinguish real profiles coming from the training set, from real profiles coming from an external cohort.

Finally, the model succeeded in representing the original distribution of glucose data $\mathcal{D}_R$, as seen in Figs. 34, 35, as well as reconstructing missing portions of it in $\mathcal{D}_G$ (Fig. 36). This is the case even without specialized layers for time series like RNNs, solely with 1D convolutional layers. Moreover, adjustments in the sampling method of the Latent Space, have shown to influence the output data distribution, suggesting a better understanding of the LS in the future, could allow control over the generation style with different sampling strategies. Overall, this and previous evidence shows we were able to approximate the underlying probability distribution of the population p_R, in our WCGAN's p_G.

As we close this chapter on our project, it is clear that it has been a long and challenging journey. This research, has not only met its initial goals but also opened the way for many new questions and pathways to explore. It has become evident how implementing deep learning architectures is more of an exploratory art, rather than a guided and static science. It is an art of finding hyperparameters empirically, just as if you tried to fit the pieces of a jigsaw, or mix the perfect ingredients in a magic formula. All of this as you try to find balance in the tug-of-war between the competing demands of data and parameters, and the ever-increasing computational costs. When such compromises suddenly fit, like a key opening a lock, they reveal the great and beautiful potential of generative deep learning in understanding very complex interconnected relationships, which could ultimately improve the lives of millions of individuals struggling with diabetes around the world.

13 References

[1] World Health Organization. *Diabetes 2023*. https://www.who.int/health-topics.

[2] E. Digitale. “New Research Shows How to Keep Diabetics Safer During Sleep”. In: *Scope Stanford Medicine* (2014).

[3] American Diabetes Association Professional Practice Committee. “Improving care and promoting health in populations: Standards of Medical Care in Diabetes—2022”. In: *Diabetes Care* 45.Suppl. 1 (2022).

[4] Claudio Cobelli, Eric Renard, and Boris Kovatchev. “Artificial pancreas: past, present, future”. In: *Diabetes* 60.11 (2011), pp. 2672–2682.

[5] B. Wang. *Artificial Pancreas Shows Improved Results but Longer Studies Needed to Show Cost Effectiveness*. https://www.nextbigfuture.com/2018/04/artificial-pancreas-shows-improved-results-but-longer-studies-needed-to-show-cost-effectiveness.html.

[6] *Micelab Website*. https://micelab.udg.edu/.

[7] P. Saeedi. “Global and regional diabetes prevalence estimates for 2019 and projections for 2030 and 2045: Results from the International Diabetes Federation Diabetes Atlas, 9th edition”. In: *Diabetes Research and Clinical Practice* 157 (2019), p. 107843.

[8] M. Prentki and C. Nolan. “Islet beta cell failure in type 2 diabetes”. In: *The Journal of Clinical Investigation* 116.7 (2006), pp. 1802–1812.

[9] Mark A Atkinson, George S Eisenbarth, and Aaron W Michels. “Type 1 diabetes”. In: *The Lancet* 383.9911 (2014), pp. 69–82.

[10] Md Saidur Rahman et al. “Role of insulin in health and disease: an update”. In: *International journal of molecular sciences* 22.12 (2021), p. 6403.

[11] Roger H Unger and Lelio Orci. “Paracrinology of islets and the paracrinopathy of diabetes”. In: *Proceedings of the National Academy of Sciences* 107.37 (2010), pp. 16009–16012.

[12] Vincenzo Manca. *The Glucose Homeostasis*. ResearchGate. https://www.researchgate.net/figure/The-glucose-homeostasis_fig1_220284268.

[13] Milton Keynes University Hospital (NHS). *What is the glycaemic index (GI)?*

[14] Ahmed Al-Qaissi et al. “Environmental effects of ambient temperature and relative humidity on insulin pharmacodynamics in adults with type 1 diabetes mellitus”. In: *Diabetes, Obesity and Metabolism* 21.3 (2019), pp. 569–574.

[15] Sheri R Colberg et al. “Physical activity/exercise and diabetes: a position statement of the American Diabetes Association”. In: *Diabetes care* 39.11 (2016), pp. 2065–2079.

[16] Eivind Andersen and Arne T Høstmark. “Effect of a single bout of resistance exercise on postprandial glucose and insulin response the next day in healthy, strength-trained men”. In: *The Journal of Strength & Conditioning Research* 21.2 (2007), pp. 487–491.

[17] Eleonora Poggiogalle, Humaira Jamshed, and Courtney M Peterson. “Circadian regulation of glucose, lipid, and energy metabolism in humans”. In: *Metabolism* 84 (2018), pp. 11–27.

[18] Richard S Surwit, Mark S Schneider, and Mark N Feinglos. “Stress and diabetes mellitus”. In: *Diabetes care* 15.10 (1992), pp. 1413–1422.

[19] Michael A Nauck. “Incretin therapies: highlighting common features and differences in the modes of action of glucagon-like peptide-1 receptor agonists and dipeptidyl peptidase-4 inhibitors”. In: *Diabetes, Obesity and Metabolism* 18.3 (2016), pp. 203–216.

[20] J. A. Hawley and M. J. Gibala. “What’s New Since Hippocrates? Preventing Type 2 Diabetes by Physical Exercise and Diet”. In: *Diabetologia* 55.2 (Jan. 2012), pp. 535–539. DOI: 10.1007/s00125-011-2404-0.

[21] S. Amanat et al. “Exercise and Type 2 Diabetes”. In: *Exercise in Diabetics: Prevention and Management of Type 2 Diabetes*. Springer, 2020, pp. 63–75. ISBN: 978-981-15-1791-4. DOI: 10.1007/978-981-15-1792-1_6.

[22] A. Kalsbeek. “Circadian control of glucose metabolism”. In: *Molecular Metabolism* 3.4 (July 2014), pp. 372–383. DOI: 10.1016/j.molmet.2014.03.002.

[23] C. Henry, B. Kaur, and R. Quek. “Chrononutrition in the management of diabetes”. In: *Nutrition & Diabetes* 10.1 (Feb. 2020), p. 6. DOI: 10.1038/s41387-020-0111-3.

[24] G Perriello et al. “Nocturnal spikes of growth hormone secretion cause the dawn phenomenon in type 1 (insulin-dependent) diabetes mellitus by decreasing hepatic (and extrahepatic) sensitivity to insulin in the absence of insulin waning”. In: *Diabetologia* 33 (1990), pp. 52–59.

[25] Gary Scheiner and Bret A Boyer. “Characteristics of basal insulin requirements by age and gender in Type-1 diabetes patients using insulin pump therapy”. In: *Diabetes research and clinical practice* 69.1 (2005), pp. 14–21.

[26] P. Diem, P.H. Ducluzeau, and A. Scheen. “The discovery of insulin”. In: *Diabetes Epidemiology and Management* 5 (2022), p. 100049. ISSN: 2666-9706.

[27] R. N. Bergman et al. “Quantitative estimation of insulin sensitivity”. In: *American Journal of Physiology-Endocrinology and Metabolism* 236.6 (1979), E667–E677.

[28] R. Hovorka et al. “Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes”. In: *Physiological Measurement* 25.4 (2004), pp. 905–920.

[29] Ayub Md. Som. *Schematic diagram of Hovorka model with modified equations*. ResearchGate. https://www.researchgate.net/figure/Schematic-diagram-of-Hovorka-model-with-modified-equations_fig2_363927735. 2023.

[30] C. Dalla Man, R. A. Rizza, and C. Cobelli. “Meal simulation model of the glucose-insulin system”. In: *IEEE Transactions on Biomedical Engineering* 54.10 (2007), pp. 1740–1749.

[31] WPTM van Doorn et al. “Machine learning-based glucose prediction with use of continuous glucose and physical activity monitoring data: The Maastricht Study”. In: *PLoS ONE* 16.6 (2021), e0253125. DOI: 10.1371/journal.pone.0253125.

[32] O. Mujahid et al. “Conditional Synthesis of Blood Glucose Profiles for T1D Patients Using Deep Generative Models”. In: *Mathematics* 10.20 (2022), p. 3741.

[33] R. Miotto et al. “Deep learning for healthcare: review, opportunities and challenges”. In: *Briefings in Bioinformatics* 19.6 (Nov. 2018), pp. 1236–1246. DOI: 10.1093/bib/bbx044.

[34] I. Contreras and J. Vehi. “Artificial Intelligence for Diabetes Management and Decision Support: Literature Review”. In: *Journal of Medical Internet Research* 20.5 (May 2018), e10775. DOI: 10.2196/10775.

[35] *PubMed Search: ML Learning and Diabetes*. https://pubmed.ncbi.nlm.nih.gov/?term=%28machine+learning%29+AND+%28diabetes%29.

[36] *PubMed Search: Generative Deep Learning and Diabetes*. https://pubmed.ncbi.nlm.nih.gov/?term=%28generative+deep+learning%29+AND+%28diabetes%29.

[37] I. Goodfellow et al. “Generative adversarial networks”. In: *Communications of the ACM* 63.11 (2020), pp. 139–144.

[38] Google. *Descripción general de la estructura de las GAN*. https://developers.google.com/machine-learning/gan/gan_structure?h.

[39] Alec Radford, Luke Metz, and Soumith Chintala. “Unsupervised representation learning with deep convolutional generative adversarial networks”. In: *arXiv preprint arXiv:1511.06434* (2015).

[40] Martin Arjovsky, Soumith Chintala, and Léon Bottou. “Wasserstein generative adversarial networks”. In: *International conference on machine learning*. PMLR. 2017, pp. 214–223.

[41] Mehdi Mirza and Simon Osindero. “Conditional generative adversarial nets”. In: *arXiv preprint arXiv:1411.1784* (2014).

[42] Omer Mujahid et al. “Generative deep learning for the development of a type 1 diabetes simulator”. In: *Communications Medicine* 4.1 (2024), p. 51.

[43] AMIC Health. *Home - AMIC Health*. https://www.amic-health.com/. 2023.

[44] C. Cobelli, E. Renard, and B. Kovatchev. “Artificial Pancreas: Past, Present, Future”. In: *Diabetes* 60 (11 2011), pp. 2672–2682. DOI: 10.2337/db11-0654.

[45] Theresa S Cavaiola and Jeremy Pettus. “Evolution of the artificial pancreas”. In: *American journal of managed care* 19.2 Suppl (2013), S55.

[46] Seth A Berkowitz et al. “Initial choice of oral glucose-lowering medication for diabetes mellitus: a patient-centered comparative effectiveness study”. In: *JAMA internal medicine* 173.12 (2018), pp. 1855–1862.

[47] Satish K Garg et al. “Glucose outcomes with the in-home use of a hybrid closed-loop insulin delivery system in adolescents and adults with type 1 diabetes”. In: *Diabetes technology & therapeutics* 19.3 (2017), pp. 155–163.

[48] Marta Bassi et al. “Automated Insulin Delivery (AID) Systems: Use and Efficacy in Children and Adults with Type 1 Diabetes and Other Forms of Diabetes in Europe in Early 2023”. In: *Life* 13.3 (2023). ISSN: 2075-1729. DOI: 10.3390/life13030783.

[49] *Artificial Pancreas Device Systems Market 2023 Statistical Overview of Sizing Report*. https://www.marketwatch.com/press-release/artificial-pancreas-device-systems-market-2023-statistical-overview-of-sizing-report-2023-05-03. MarketWatch. 2023.

[50] Medtronic. *Sistema de bomba de insulina MiniMed™ 780G con tecnología SmartGuard™*. https://www.medtronic-diabetes.com/es-ES/sistema-integrado-minimed-780g. Medtronic.

[51] Omnipod. *Simplify Life with Omnipod® 5*. https://www.omnipod.com/what-is-omnipod/omnipod-5. Omnipod.

[52] Expert Market Research. *Global Digital Health Market Outlook*. 2023.

[53] Expert Market Research. *Global Clinical Decision Support Systems Market Outlook*. https://www.expertmarketresearch.com/reports/clinical-decision-support-systems-market. 2023.

[54] M. Schmidt. “The dawn phenomenon, an early morning glucose rise: implications for diabetic intraday blood glucose variation”. In: *Diabetes care* 4.6 (Nov. 1981), pp. 579–585. DOI: 10.2337/diacare.4.6.579.

[55] Hospital Clinic of Barcelona Conget I. *Prediction and Prevention of Nocturnal Hypoglycemia in Persons With Type 1 Diabetes Using Machine Learning Techniques*. ClinicalTrials.gov. https://clinicaltrials.gov/ct2/show/NCT03711656.

[56] Q. Zhao, J. Zhu, X. Shen, et al. “Chinese diabetes datasets for data-driven machine learning”. In: *Scientific Data* 10 (2023), p. 35. URL: https://doi.org/10.1038/s41597-023-01940-7.

[57] Parisa Avari et al. “Differences for Percentage Times in Glycemic Range Between Continuous Glucose Monitoring and Capillary Blood Glucose Monitoring in Adults with Type 1 Diabetes: Analysis of the REPLACE-BG Dataset”. In: *Diabetes Technology & Therapeutics* 22.3 (Mar. 2020), pp. 222–227. DOI: 10.1089/dia.2019.0276.

[58] Chiara Dalla Man et al. “The UVA/PADOVA Type 1 Diabetes Simulator: New Features”. In: *Journal of Diabetes Science and Technology* 8.1 (2014), pp. 26–34. DOI: 10.1177/1932296813514502.

[59] C. Marling. “The OhioT1DMDataset for Blood Glucose Level Prediction: Update 2020”. In: (2020). Unpublished manuscript.

[60] Peter Pesl et al. “An Advanced Bolus Calculator for Type 1 Diabetes: System Architecture and Usability Results”. In: *IEEE Journal of Biomedical and Health Informatics* 20.1 (2016), pp. 11–17. DOI: 10.1109/JBHI.2015.2464088.

[61] Isaac Fox et al. “Deep multi-output forecasting: Learning to accurately predict blood glucose trajectories”. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. 2018, pp. 1387–1395.

[62] Jie Yang et al. “An ARIMA model with adaptive orders for predicting blood glucose concentrations and hypoglycemia”. In: *IEEE Journal of Biomedical and Health Informatics* 23.3 (2018), pp. 1251–1260.

[63] Fabien Dubosson et al. “The open D1NAMO dataset: A multi-modal dataset for research on non-invasive type 1 diabetes management”. In: *Informatics in Medicine Unlocked* 13 (2018), pp. 92–100. ISSN: 2352-9148. DOI: https://doi.org/10.1016/j.imu.2018.09.003.

[64] Temiloluwa Prioleau et al. “DiaTrend: A dataset from advanced diabetes technology to enable development of novel analytic solutions”. In: *arXiv preprint arXiv:2304.06506* (2023).

[65] Michael C. Riddell et al. “Examining the Acute Glycemic Effects of Different Types of Structured Exercise Sessions in Type 1 Diabetes in a Real-World Setting: The Type 1 Diabetes and Exercise Initiative (T1DEXI)”. In: *Diabetes Care* 46.4 (Feb. 2023), pp. 704–713. ISSN: 0149-5992. DOI: 10.2337/dc22-1721.

[66] Alvis Cabrera et al. “Validation of a Probabilistic Prediction Model for Patients with Type 1 Diabetes Using Compositional Data Analysis”. In: *Mathematics* 11.5 (2023), p. 1241. ISSN: 2227-7390. DOI: 10.3390/math11051241.

[67] Marcello Pompa et al. “A comparison among three maximal mathematical models of the glucose-insulin system”. In: *PloS one* 16.9 (2021), e0257789.

[68] Zinan Lin et al. “Using gans for sharing networked time series data: Challenges, initial promise, and open questions”. In: *Proceedings of the ACM Internet Measurement Conference*. 2020, pp. 464–483.

[69] Grazia Aleppo et al. “REPLACE-BG: a randomized trial comparing continuous glucose monitoring with and without routine blood glucose monitoring in adults with well-controlled type 1 diabetes”. In: *Diabetes care* 40.4 (2017), pp. 538–545.

[70] Mathieu Rouaud. *Probability, Statistics and Estimation*. p. 10. Archived from the original on 2022-10-09. Archived, 2013. URL: https://ghostarchive.org/archive/dcUO0.

[71] Prabhaker Mishra et al. “Descriptive statistics and normality tests for statistical data”. In: *Annals of cardiac anaesthesia* 22.1 (2019), pp. 67–72.

[72] Yann LeCun et al. “Gradient-based learning applied to document recognition”. In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[73] Kaiming He et al. “Deep residual learning for image recognition”. In: 2016, pp. 770–778.

[74] Tom B Brown et al. “Language Models are Few-Shot Learners”. In: *arXiv preprint arXiv:2005.14165* (2020).

[75] Boris P. Kovatchev et al. “Risk Analysis of Blood Glucose Data: A Quantitative Approach to Optimizing the Control of Insulin Dependent Diabetes”. In: *Journal of Theoretical Medicine* 3 (1999). Received 10 August 1999; Revised October 1999; In final form 18 January 2000, pp. 1–10.

[76] Students of PhD in Data Science Batch 2023 at the Asian Institute of Management. *Time Series Analysis Handbook*. 2020. URL: https://phdinds-aim.github.io/time_series_handbook/06_ConvergentCrossMappingandSugiharaCausality/ccm_sugihara.html.

[77] Taiyu Zhu et al. “Glugan: Generating personalized glucose time series using generative adversarial networks”. In: *IEEE Journal of Biomedical and Health Informatics* (2023).

[78] Ethan R Deyle and George Sugihara. “Generalized theorems for nonlinear state space reconstruction”. In: *Plos one* 6.3 (2011), e18295.

[79] Geektopia. *Ryzen 9 7900X AMD*. https://www.geektopia.es/es/product/amd/ryzen-9-7900x/.

[80] *Micelab Projects List*. https://micelab.udg.edu/projects/.

[81] SocialDiabetes. *A new partnership between SocialDiabetes and the University of Girona to implement AI-based Digital Therapies for diabetes management*. Oct. 2022. URL: https://blog.socialdiabetes.com/en/a-new-partnership-between-socialdiabetes-and-the-university-of-girona-to-implement-ai-based-digital-therapies-for-diabetes-management/.

[82] Hewlett Packard. *What is the Average Lifespan of a Computer?* 2021. URL: https://www.hp.com/in-en/shop/tech-takes/post/average-computer-lifespan.

[83] European Parliament and Council. *REGULATION (EU) 2017/745 OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL of 5 April 2017 on medical devices*. 2017. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32017R0745.

[84] European Medicines Agency. *Medical Device Regulation comes into application*. 2021. URL: https://www.ema.europa.eu/en/news/medical-device-regulation-comes-application.

[85] U.S. Food and Drug Administration. *Software as a Medical Device (SaMD) by FDA*. URL: https://www.fda.gov/medical-devices/digital-health-center-excellence/software-medical-device-samd.

[86] International Electrotechnical Commission. *IEC 82304-1:2016 - Health software — Part 1: General requirements for product safety*. 2016. URL: https://www.iso.org/standard/59577.html.

[87] Parlamento y Consejo Europeo. *Reglamento (UE) 2016/679*. Diario Oficial de la Unión Europea. Apr. 2016. URL: http://eur-lex.europa.eu/legal-content/ES/TXT/?uri=OJ:L:2016:119:TOC.

[88] Agencia Española de Protección de Datos (AEPD). *Adecuación al RGPD de tratamientos que incorporan Inteligencia Artificial. Una introducción*. Publication by the Agencia Española de Protección de Datos (AEPD). Feb. 2020.

[89] CMS. *DIGITAL HEALTH APPS AND TELEMEDICINE IN SPAIN*. 2023. URL: https://cms.law/en/int/expert-guides/cms-expert-guide-to-digital-health-apps-and-telemedicine/spain.

14 Annex

Annex 1: Extended GAN Architecture

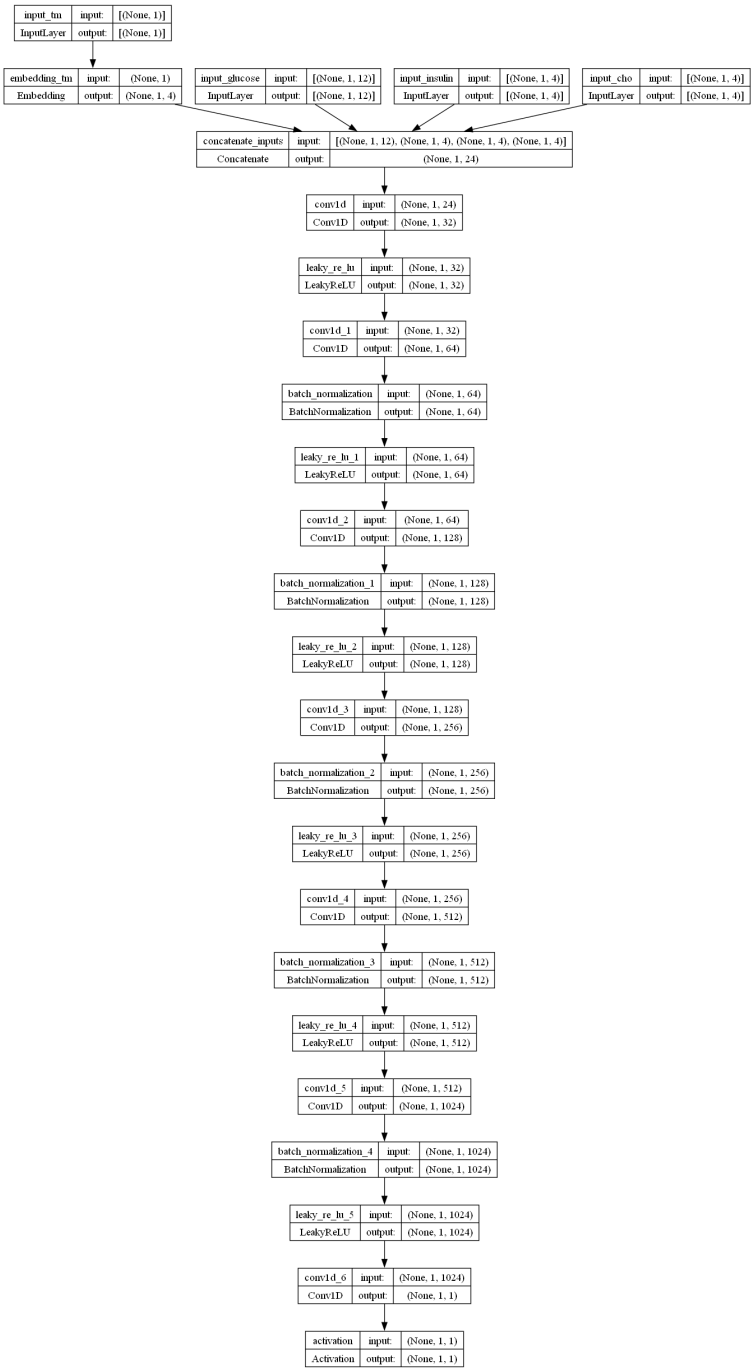

Figure 48: Discriminator architecture

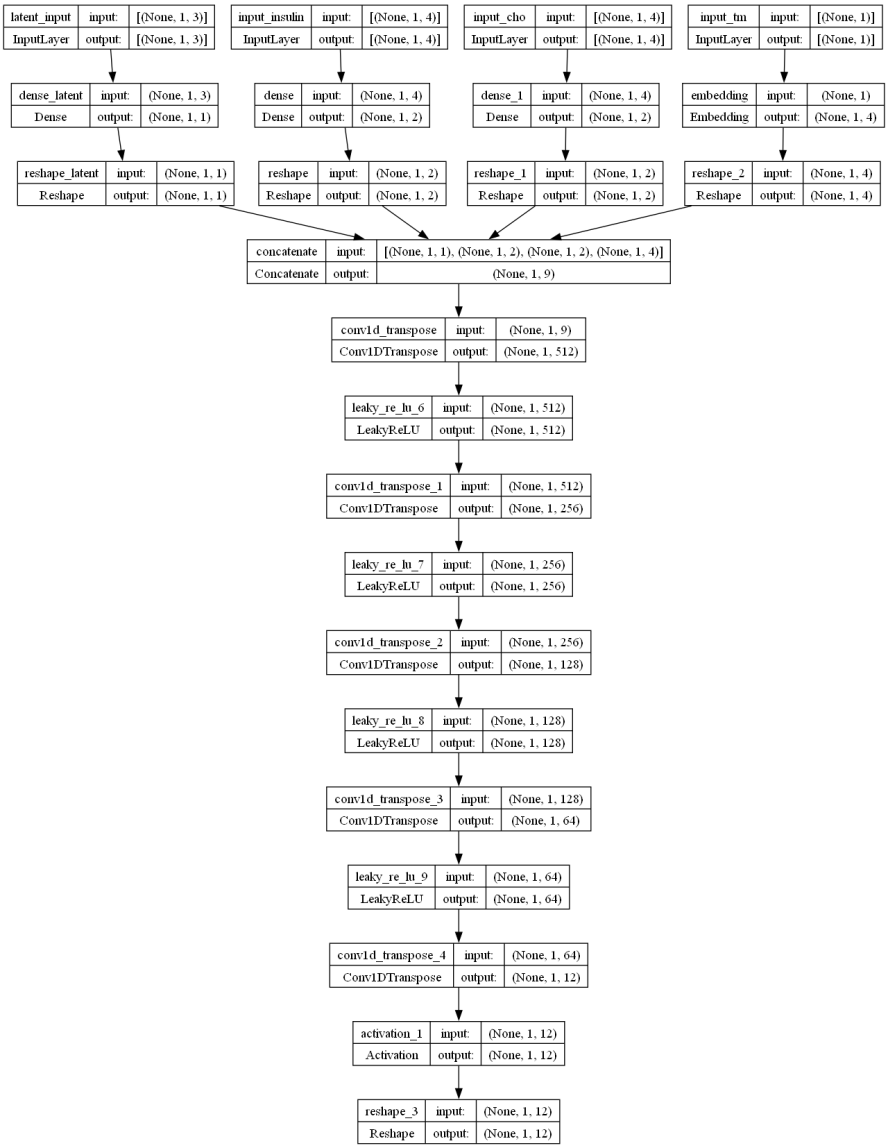

Figure 49: Generator architecture

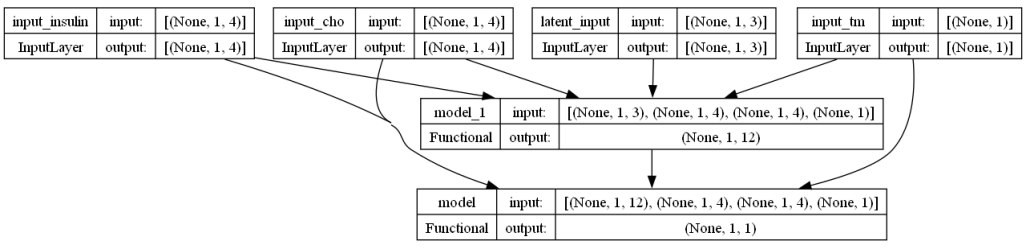

Figure 50: GAN composite

Annex 2: Additional AGP Reports

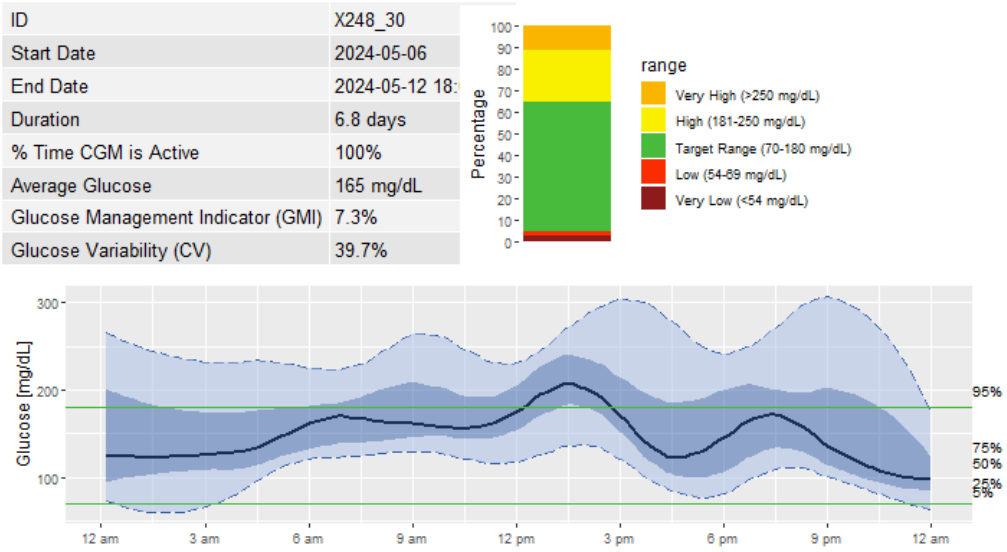

Figure 51: Real AGP report from subpatient 248-30

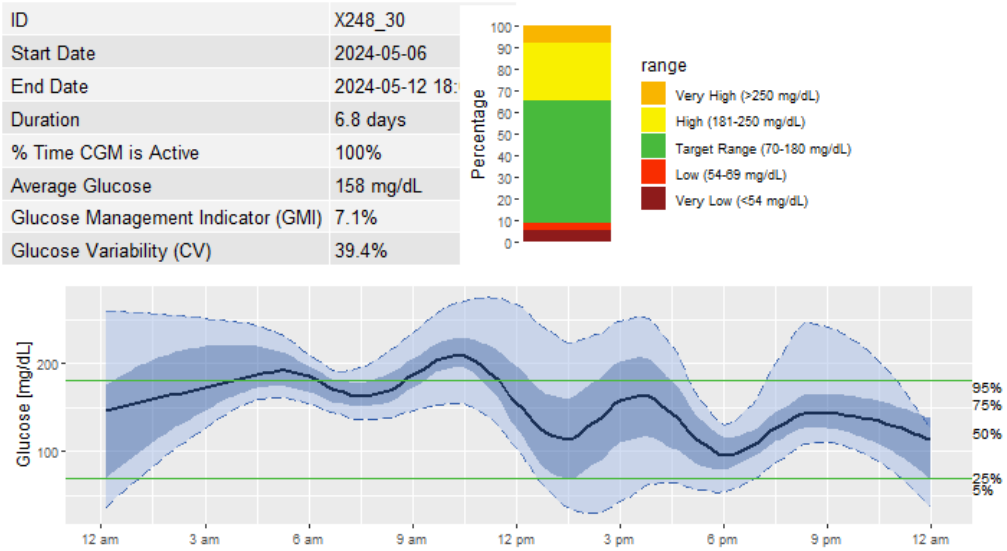

Figure 52: Generated AGP report from subpatient 248-30

Annex 3: WBS Dictionary

Name of the work package	Preliminary project
Code in WBS	0
Description	
Not a working package by itself, composed by three further work packages.	
Estimated cost	920.00 €
Estimated cost	58 hours
Limit date	2/16/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	0.1
Description	
Define the workflow, objectives, methods and deadlines	
Acceptation criterion	
Three way agreement that objectives and methods are clearly and correctly defined	
Approved by:	Oriol Bustos
Deliverables description	
Notes	
Resources	
Zoom, Micelab meeting room	
Estimated cost	200.00 €
Estimated cost	10 hours
Limit date	2/6/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	0.2
Description	
Read articles from past and present academia publications on diabetes modelling and Generative Deep Learning	
Acceptation criterion	
The general concepts are clearly understood, options available listed and documented.	
Approved by:	Oriol Bustos
Deliverables description	
A written section including this content	
Resources	
A biomedical engineer for writing and the tutor for discussion.	
Estimated cost	360.00 €
Estimated cost	24 hours
Limit date	2/7/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	0.3
Description	
Study the available options found and determine how the project will be conducted	
Acceptation criterion	
The process is clearly described from start to finished, with all the possible details and alternative options stated.	
Approved by:	Oriol Bustos
Deliverables description	
A written section including this content	
Resources	
A biomedical engineer for writing and the tutor for discussion.	
Estimated cost	360.00 €
Estimated cost	24 hours
Limit date	2/16/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	1
Description	
Not a working package by itself, composed by three further work packages.	
Estimated cost	780.00 €
Estimated cost	52 hours
Limit date	2/16/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	1.1
Description	
Define the workflow, objectives, methods and	
Acceptation criterion	
Ensure availability for use both technically and legally. Must contain sufficient total days and the necessary features that will be implented in the model, BG, CHO, Insulin, Hour.	
Approved by:	Oriol Bustos
Deliverables description	
A folder with the datasets.	
Resources	
Python, FitBit API, Excel, Synology Drive Server and	
Estimated cost	180.00 €
Estimated cost	12 hours
Limit date	2/9/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	1.2
Description	
Clean and transform raw data, normalization, removing outliers, dealing with missing values and any other necessary preprocessing.	
Acceptation criterion	
Quality and compatibility with the code. Missing values must have been handled and must be standarized.	
Approved by:	Oriol Bustos
Deliverables description	
Folder with csv files or similar	
Resources	
Excel, python, the datasets themselves.	
Estimated cost	300.00 €
Estimated cost	20 hours
Limit date	2/14/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	1.3
Description	
Perform an analysis of the cohorts, will be used for training. Understand our data, statistical metrics, frequency analysis...	
Acceptation criterion	
Main statistical aspects analyzed.	
Approved by:	Oriol Bustos
Deliverables description	
CSV files	
Resources	
Python, excel	
Estimated cost	300.00 €
Estimated cost	20 hours
Limit date	2/16/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	2
Description	
Not a working package by itself, composed by four further work packages.	
Estimated cost	1,860.00 €
Estimated cost	124 hours
Limit date	4/3/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	2.1
Description	
Design a CGAN with both a G and a D networks.	
Acceptation criterion	
Design and implements strategies to prevent mode collapse, enhance performance and efficiency. Both the G and D get better with each iteration.	
Deliverables description	
A python script	
Resources	
Python, TensorFlow.	
Estimated cost	360.00 €
Estimated cost	24 hours
Limit date	2/20/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	2.2
Description	
Design and run the functions that will load the data, pass it through the DNN's and perform the backpropagation and training steps.	
Acceptation criterion	
A function that trains the model adapted to the structure of each cohort. It also prints statements to follow the evolution of the process.	
Approved by:	Oriol Bustos
Deliverables description	
Several python function	
Resources	
Python, TensorFlow, Keras	
Estimated cost	1,200.00 €
Estimated cost	80
Limit date	2/21/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	2.3
Description	
Compare quality of the results between different hyperparameter configurations.	
Acceptation criterion	
The best / selected combination of parameters yields plausible results from the model.	
Approved by:	Oriol Bustos
Deliverables description	
A written document making the comparative.	
Resources	
Python, TensorFlow, Keras, LaTex, Excel	
Estimated cost	150.00 €
Estimated cost	10 hours
Limit date	3/20/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	2.4
Description	
Explore possible improvements: differentiated regions in the latent space, fine-tuning etc.	
Acceptation criterion	
Performance is the maximized	
Approved by:	Oriol Bustos
Deliverables description	
Document explaining what and how was improved.	
Resources	
Python, TensorFlow, Keras	
Estimated cost	150.00 €
Estimated cost	10 hours
Limit date	4/3/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	3
Description	
Not a working package by itself, composed by two further work packages.	
Estimated cost	375.00 €
Estimated cost	25 hours
Limit date	4/10/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	3.1
Description	
Evaluate the model's performance running the test data and computing performance metrics.	
Acceptation criterion	
Coherent results are obtained.	
Approved by:	Oriol Bustos
Deliverables description	
Document with relevant metrics plotted or described, compared with the other results.	
Resources	
Python, Excel	
Estimated cost	75.00 €
Estimated cost	5 hours
Limit date	4/5/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	3.2
Description	
Further analysis techniques like causality tests, frequency analysis, clustering and more.	
Acceptation criterion	
Results or findings are further confirmed.	
Approved by:	Oriol Bustos
Deliverables description	
Performance metrics document.	
Resources	
Excel, python	
Estimated cost	300.00 €
Estimated cost	20 hours
Limit date	4/10/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	4
Description	
Not a working package by itself, composed by three further work packages.	
Estimated cost	1,050.00 €
Estimated cost	70 hours
Limit date	5/6/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	4.1
Description	
Comprehensively discuss, compare results with literature and examining the model's performance metrics.	
Acceptation criterion	
Put results in context, and assesing the quality of the the same.	
Deliverables description	
Written document section.	
Resources	
LaTeX.	
Estimated cost	300.00 €
Estimated cost	20 hours
Limit date	4/14/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	4.2
Description	
Put all the work together in a summarized thesis that describes the work done, how it has been, the results obtained and discusses relevance and future trends.	
Acceptation criterion	
Scientific essay, containing the sections described in guidelines, no ortographic mistakes or technical inaccuracies and referencing its sources.	
Approved by:	Oriol Bustos
Deliverables description	
PDF file	
Resources	
LaTeX.	
Estimated cost	450.00 €
Estimated cost	30 hours
Limit date	4/29/2024
Responsible	Oriol Bustos

Name of the work package	Preliminary project
Code in WBS	4.3
Description	
Decide the topics that will be covered, selecting only the relevant ones, prepare the PPT, script and practice it.	
Acceptation criterion	
The presentation covers all important parts of the project, is accurate and sticks to the time limit.	
Approved by:	Oriol Bustos
Deliverables description	
PPT and talk.	
Resources	
PPT	
Estimated cost	300 €
Estimated cost	20 hours
Limit date	5/6/2024
Responsible	Oriol Bustos

Annex 4: Supplementary Results Tables

Table 14: Comparison of Metrics Between synthethic patient's CGM simulated with fixed TM values, presented as Average or IQR (25th;75th) Percentile

Statistic	TM=0	TM=1	TM=2	TM=3
mean_glucose *mg/dL*	161.99 (148.4-180)	153 (139.8-170)	156.81 (143.0-174.2)	152.37 (139.1-168.9)
std_glucose *mg/dL*	51.81 (42.3-60.6)	51.13 (41.7-60.0)	51.55 (41.5-60.1)	50.95 (41.5-59.4)
max_glucose *mg/dL*	324.86 (269.9-373.8)	319.50 (269.5-363.3)	322.72 (274.4-369.4)	317.14 (263.8-366.9)
min_glucose *mg/dL*	35.32 (19.8-48.3)	16.54 (28.8-43.8)	16.17 (28.4-45.6)	30.57 (28.9-42.5)
coeff.variation *%*	31.73 (27.9-35.6)	33.24 (28.8-37.3)	32.63 (28.4-36.4)	33.18 (28.9-36.9)
time_between_70_54 *%*	2.22 (1.03-3.12)	3.04 (1.68-3.8)	2.69 (1.40-3.68)	3.10 (1.77-3.95)
time_in_range *%*	60.76 (46.5-74.0)	65.43 (52.4-77.9)	63.48 (50.0-78.0)	65.92 (53.1-79.0)
time_between_180_250 *%*	28.86 (21.09-38.18)	24.35 (16.13-34.18)	26.25 (17.53-35.31)	23.99 (15.64-33.29)
time_under_54 *%*	1.54 (0.4-2.3)	2.24 (0.6-3.2)	1.94 (0.7-2.9)	2.26 (0.7-3.1)
time_over_250 *%*	6.70 (1.1-10.2)	5.11 (0.7-8.0)	5.77 (1.2-9.1)	4.92 (0.7-7.6)

Annex 5: Additional QQ Plots

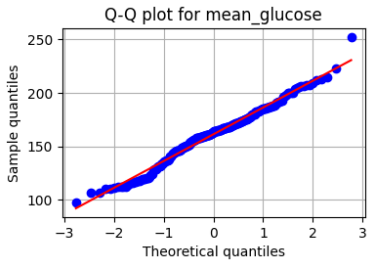

Figure 53: Q-Q plot for real `mean_glucose`

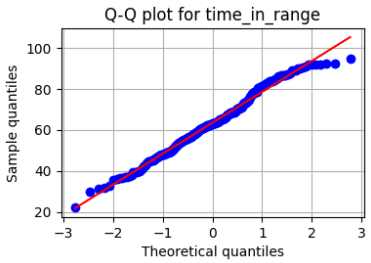

Figure 54: Q-Q plot for real `time_in_range`

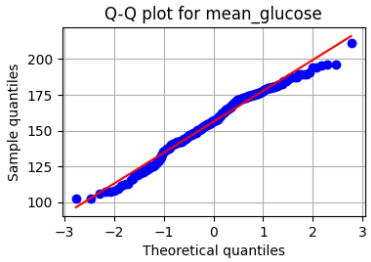

Figure 55: Q-Q plot on generated `mean_glucose`

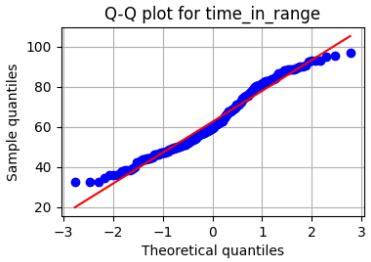

Figure 56: Q-Q plot for generated `time_in_range`