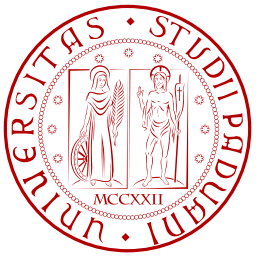


UNIVERSITY OF PADOVA

DEPARTMENT OF INFORMATION ENGINEERING
Ph.D. Course on Information Engineering
Curriculum: Bioengineering
Series: XXXV

Prediction of blood glucose concentrations and hypoglycemic events in Type 1 Diabetes by linear and nonlinear algorithms

Course director:
Andrea Neviani

Advisor:
Andrea Facchinetti

Co-advisor:
Simone Del Favero

Ph.D. candidate:
Francesco Prendin

A thesis submitted
for the degree of
PhilosophiæDoctor (Ph.D.)

Abstract

Type 1 diabetes (T1D) is a metabolic disease which impairs insulin production, and it results in altered glucose homeostasis. As a consequence, subjects must frequently self-administer exogenous insulin, consume corrective/fast-acting carbohydrates, follow dietary measures and exercise routines to maintain glycemia into a desired range (usually [70-180] mg/dL) along the day. Indeed, limiting blood glucose (BG) excursions reduces the risk of mortality, as well as, the long/short-term consequences of hyperglycemia (i.e., BG > 180 mg/dL) and hypoglycemia (i.e., BG<70 mg/dL). Minimally invasive continuous glucose monitoring (CGM) sensors have become a widely used tool by T1D individuals to keep track, and eventually correct, their BG levels. These devices provide frequent BG measurements (commonly one every 5 minutes) for several days, and embed visual and acoustic alerts when the hypo-/hyperglycemic thresholds are crossed, thus helping patients in taking corrective actions like hypotreatments and corrective insulin boluses. However, timely preventive alerts coupled with targeted corrective strategies would be even more helpful to avoid or mitigate the onset of impending, adverse events. For this reason, the real-time forecasting of BG levels has a key role in the development of i) advanced decision support systems (DSS), which are software for helping patients in the decision-making process, and ii) artificial pancreas systems (APS), which are devices for automatizing insulin delivery. The large plethora of data provided by CGM devices (but also insulin pumps, wearable devices, electronic diaries and dedicated mobile applications), coupled with the technological advancements in artificial intelligence, have driven the diabetes technology community to intensively focus on developing glucose predictive algorithms, exploiting methodologies already employed in the fields of time series forecasting, system identification, machine and deep learning. Among the possible approaches for glucose prediction, two main categories can be identified: algorithms fed only by the past history of the CGM signal or fed by CGM data plus additional information such as insulin, carbo-

hydrates or physical exercise. One main open issue is that none of the literature studies have systematically investigated how and/or how much different input information as well as complex algorithms contribute to improve glucose prediction on datasets recorded in daily-life conditions.

To address this gap, this PhD thesis presents the development and application of several linear and nonlinear algorithms for the forecasting of BG levels and hypoglycemic events, and investigates how and how much different input information and model complexity play a role in the prediction.

The thesis is organized as follows: a brief introduction on T1D complications, management, technologies and a picture of the main state-of-art predictive algorithms is presented in Chapter 1.

Chapter 2 proposes a head-to-head comparison between linear and nonlinear state-of-art models employing only CGM data as input. In such a scenario, we demonstrate that the best performance is achieved by individualized models, particularly by Autoregressive Integrated Moving Average (ARIMA) models and feed forward neural networks (NN), evidencing the key role of model parameter individualization. However, one of the main limitations of all the CGM-based predictive algorithms is that any metabolic disturbance, e.g., a meal, would deteriorate the accuracy of the predicted BG concentrations. Therefore, the use of additional sources of information should be considered to improve the accuracy of prediction algorithms.

Thus, in Chapter 3 we investigate how much adding information on meal time and seasonality to CGM data can improve prediction performance. This is achieved by exploiting a novel methodology based on clustering and seasonal stochastic local models. The novel approach is able to outperform CGM-only algorithms and to achieve similar performance to other linear and nonlinear methods, but fed with more information (i.e., CGM, meal and insulin data). However, despite achieving a satisfactory performance in forecasting the future value of BG concentration all over the glycemic range, the prediction of hypoglycemic events, which involves the comparison of predicted versus measured BG values and requires the creation of a classification-like framework, is still a task that poses a challenge to all the algorithms.

For this reason, in Chapter 4 we focus on improving hypoglycemia forecasting by employing Autoregressive Integrated Moving-Average with eXogenous input (ARIMAX) models identified using a glucose specific metric, which allows to better weight hypoglycemia, and equipped with a prediction-funnel alarm strategy. Results show that this approach significantly improve the pre-

diction of impending hypoglycemic events.

So far, we mainly focused on black-box approaches, Chapter 5 investigates whether the use of a physiological white box model identified from easily accessible patient recorded data (i.e., CGM, meal and insulin) and embedded within a particle filter can improve the predictive performance. Unfortunately, in this case the results do not support the hypothesis since advanced nonparametric and deep learning black-box methods turned out to significantly outperform the proposed physiological model.

Finally, we tackle one of the key problems when using black-box machine- and deep-learning approaches: the interpretability of the outcome. Whilst machine and deep-learning models can grant accurate performance, their results can be difficult for users to explain, thus limiting their usability in real-life application. In addition, when two predictive models present a similar performance, the model selected to be used in practice should be the one providing the easiest and most straightforward interpretation. Chapter 6 addresses this issue by: i) introducing a novel tool able to describe the output of each models' prediction and ii) developing a case-of-study in which interpretability should be preferred over prediction accuracy in the choice of the model.

Chapter 7 provides a description of the main findings, and a discussion on possible applications and rooms for improvement.

Sommario

Il diabete di tipo 1 (T1D) è una malattia metabolica caratterizzata da una mancanza di produzione di insulina che provoca un'alterazione dei livelli di glucosio nel sangue (BG). Di conseguenza, per mantenere la glicemia in un adeguato range fisiologico (generalmente [70-180] mg/dL) durante la giornata, i soggetti diabetici devono somministrarsi insulina esogena, assumere carboidrati ad azione rapida, seguire una dieta equilibrata ed eseguire attività fisica. Infatti, limitare le escursioni della glicemia consente di ridurre il rischio di mortalità e le conseguenze, a lungo e breve termine, causate da eventi iperglicemici (BG > 180 mg/dL) e ipoglicemici (BG < 70 mg/dL). I sensori minimamente invasivi per il monitoraggio in continua della glicemia (CGM) sono ampiamente utilizzati dai soggetti diabetici per monitorare, ed eventualmente, correggere i loro livelli glicemici. Generalmente, questi dispositivi forniscono una misurazione della glicemia ogni 5 minuti, per diversi giorni, e integrano allarmi acustici o visivi quando i livelli di glucosio nel sangue oltrepassano le soglie di ipo/iperglicemia. Questi allarmi consentono ai pazienti di prendere delle azioni correttive, come assunzioni di carboidrati, in caso di ipoglicemia, o boli insulinici, in caso di iperglicemia. Tuttavia, l'utilizzo di allarmi preventivi, generati con un adeguato anticipo temporale, assieme all'uso di specifiche strategie correttive consentirebbe di evitare, o almeno mitigare, il verificarsi di futuri eventi critici. Per questo motivo, la predizione in *real-time* dei livelli glicemici svolge un ruolo chiave nello sviluppo di: i) sistemi avanzati per il supporto alla decisione (DSS), i.e., software che forniscono assistenza ai pazienti durante il processo terapeutico-decisionale, e ii) sistemi di pancreas artificiale (APS), i.e., dispositivi che automatizzano l'infusione di insulina. La grande disponibilità di dati fornita dai dispositivi CGM (così come dalle pompe insuliniche, dai dispositivi indossabili, dai diari elettronici e da app dedicate), assieme all'avanzamento tecnologico nel campo dell'intelligenza artificiale, ha guidato la comunità tecnologica del diabete a concentrarsi intensivamente sullo sviluppo di algoritmi predittivi sfruttando tecniche già utiliz-

zate nei campi della predizione delle serie temporali, dell'identificazione dei sistemi, machine e deep learning. In generale, fra tutti i possibili approcci predittivi, possiamo distinguere due categorie principali: algoritmi che utilizzano solo la storia passata del segnale CGM o algoritmi che utilizzano, assieme ai dati CGM, anche altre informazioni aggiuntive come insulina, carboidrati o attività fisica. Una questione aperta, che nessuno degli studi di letteratura ha sistematicamente investigato, è come e/o quanto i diversi input e gli algoritmi, più o meno complessi, contribuiscono a migliorare la predizione dei livelli glicemici, sfruttando dati acquisiti in condizioni di vita quotidiana.

Per affrontare tale *gap*, questo lavoro di tesi presenta lo sviluppo e applicazione di diversi algoritmi, lineari e non lineari, per la predizione dei livelli glicemici e di eventi ipoglicemici, inoltre valuta il ruolo che diversi input, e la complessità del modello in esame, svolgono nella predizione.

La presente tesi è organizzata come segue: una breve introduzione sulle complicanze, la gestione, le tecnologie del diabete di tipo 1 e un quadro generale degli algoritmi predittivi presenti in letteratura vengono riportati nel Capitolo 1.

Il Capitolo 2 propone un confronto esaustivi tra tecniche lineari e nonlineari allo stato dell'arte, che utilizzano solo dati CGM come input. In questo scenario, le migliori performance sono ottenute da modelli individualizzati, in particolare da modelli *Autoregressive Integrated Moving Average* (ARIMA) e da *feed forward neural network* (NN), evidenziando il ruolo chiave della personalizzazione dei parametri del modello. Tuttavia, una delle limitazioni principali di tutti gli algoritmi basati su dati CGM è che qualsiasi disturbo metabolico, ad esempio un pasto, potrebbe deteriorare l'accuratezza della predizione dei livelli glicemici. Quindi, l'utilizzo di informazioni aggiuntive dovrebbe essere considerato per migliorare l'accuratezza degli algoritmi predittivi.

Perciò, nel Capitolo 3 si valuta quanto le informazioni relative al tempo del pasto e alla *seasonality*, in aggiunta ai dati CGM, contribuiscono a migliorare le performance di predizione. Questo è possibile grazie all'utilizzo di una nuova metodologia basata su *clustering* e su modelli stocastici locali. Il nuovo approccio è in grado di fornire delle performance migliori rispetto agli algoritmi che utilizzano solo dati CGM e risultati comparabili ad altri metodi (lineari e non lineari) che utilizzano una maggiore quantità di informazione in input (CGM, pasti e insulina). Tuttavia, nonstante le performance ottenute relative alla predizione dei livelli glicemici siano soddisfacenti lungo tutto il range glicemico, la predizione di eventi ipoglicemici, che prevede il confronto

tra il glucosio predetto vs misurato e la creazione di un framework di classificazione, resta un obiettivo che pone una prova difficile a tutti gli algoritmi.

Per questo motivo, nel Capitolo 4 ci siamo focalizzati sul migliorare la predizione delle ipoglicemie utilizzando modelli *Autoregressive Integrated Moving Average eXogenous input* (ARIMAX), identificati con una metrica specifica per la glicemia, che consente di pesare in maniera migliore le ipoglicemie, e forniti di una nuova strategia di allarme basata sul *funnel* di predizione. Questo approccio migliora significativamente la predizione degli eventi ipoglicemici.

Fin qui sono stati analizzati approcci *black-box*, per questo il Capitolo 5 studia se l'utilizzo di un modello fisiologico *white-box* fisiologico, identificato utilizzando dati facilmente accessibili del paziente (CGM, pasti e insulina), e integrato in un *particle filter* può migliorare le performance predittive. Sfortunatemente, in questo caso, i risultati non supportano l'ipotesi dato che sia i modelli non parametrici che i metodi di deep learning testati risultano migliorare significativamente le performance fornite dal modello fisiologico proposto.

Infine, viene affrontato uno dei problemi principali legato all'utilizzo di approcci *black-box* di machine e deep learning: l'interpretabilità dell'*outcome*. Infatti, mentre questi modelli sono in grado di fornire delle buone performance, i loro risultati possono essere difficilmente interpretabili per gli utenti, limitandone così la loro utilizzabilità nelle applicazioni reali. Inoltre, quando due modelli predittivi presentano una performance simile, il modello da utilizzare in pratica dovrebbe essere quello più semplice e con la più diretta intepretazione. Il Capitolo 6, affronta questa tematica: i) introducendo un nuovo strumento per descrivere l'output di ogni predizione di modello e ii) proponendo un caso di studio in cui, per la scelta del modello, l'interpretabilità dovrebbe essere preferita rispetto ai risultati di predizione.

Il Capitolo 7 fornisce una descrizione dei risultati principali e una discussione su possibili applicazioni e margini di miglioramento.

Contents

Abbreviations

ADA American Diabetes Association

AI Artificial intelligence

AIC Akaike information criterion

ANOVA Analysis of variance

AP Artificial pancreas

AR Autoregressive

ARMA Autoregressive Moving Average

ARIMA Autoregressive Integrated Moving Average

ARX Autoregressive with eXogenous input

ARMAX Autoregressive Moving Average with eXogenous input

ARIMAX Autoregressive Integrated Moving Average with eXogenous input

BIC Bayesian information criterion

BG Blood glucose

BW Body weight

CEG Clarke error grid

CF Correction factor

CGM Continuous glucose monitoring

CHO Carbohydrates

CNN Convolutional neural network

COD Coefficient of variation

CR Carbohydrate-to-insulin ratio

CSII Continuous subcutaneous insulin infusion

CV Cross validation

DL Deep learning

DSS Decision support system

DW Detection window

F1 F1-score

FDA Food and Drug Administration

FN False negative

FP False positive

gMSE Glucose mean square error

GRU Gated recurrent unit

HBGI High blood glucose index

HE Hypoglycemic event

HT Hypotreatment

LIME Local interpretable model-agnosticexplanation

LSTM Long short-term memory neural neutwork

MAE Mean absolute error

MDI Multiple daily injections

ML Machine learning

MCMC Markov chain Monte Carlo

MPC Model predictive control

MSE Mean square error

NAR Nonlinear autoregressive neural network

NC Not countable

NN Neural network (feed forward)

NP Nonparametric

NN-X Neural network with eXogenous input (feed forward)

P Precision

PP Postprandial period

PDSFCM Partial distance fuzzy C-means

PEM Prediction error method

PH Prediction horizon

PHY Physiological model

PP Postprandial period

qHE Quasi hypoglycemic event

R Recall

regRF Regression random forest

RKHS Reproducing kernel hilbert space

RLS Recursive least squares

RMSE Root mean square error

RNN Recurrent neural network

SAP Sensor-augmented pump

SARIMA Seasonal autoregressive integrated moving average

SMBG Self-monitoring of blood glucose

SHAP Shaple addictive explanation

SSD Statistical significant difference

SVR Support vector regression

T1D Type 1 diabetes

T1DS Type 1 diabetes simulator

TAR Time above range

TBR Time below range

TCN Temporal convolutional network

TG Time gain

TIR Time in range

TN True negative

TP True positive

List of Figures

List of Tables

Chapter 1

Type 1 diabetes: blood glucose forecasting and thesis aim

Type 1 diabetes (T1D) is a chronic disease characterized by a lack of insulin production due to the autoimmune destruction of the pancreatic β-cells, resulting in blood glucose (BG) levels that exceed the normal glucose range, usually defined between 70 and 180 mg/dL. As a consequence, individuals affected by T1D frequently need to counteract with various corrective measures, such as insulin injections and fast acting carbohydrates to keep blood glucose levels within a safe range. In the recent years, the use of continuous glucose monitoring (CGM) devices, insulin pumps and dedicated mobile applications has significantly improved T1D management. However, knowing in advance when BG is approaching critical levels has the potential to further revolutionize diabetes care, thus promoting patient well-being. This chapter describes the main complications related to T1D, the widely used technologies and the main literature contributions that has addressed the challenging task of BG forecasting.

1.1 Type 1 diabetes (T1D): description of the disease and its therapy

Type 1 diabetes (T1D) is a life-long disease that cannot be prevented or cured and it is caused by an autoimmune reaction in which the body's immune system attacks the pancreatic β-cells responsible for the endogenous insulin production [2, 3]. In healthy individuals, insulin has the crucial role of accurately regulating blood glucose (BG) homeostasis by promoting the transport of glu-

cose from the bloodstream to muscles, fat and liver cells. So, the absence of endogenous insulin in patient affected by T1D results in elevated BG concentrations. The first line treatment to lower BG levels consists in multiple daily injections of exogenous insulin. Unfortunately, excessive insulin dosing could lead patients to experience very low BG concentrations, which is dangerous even in the short-term since it could cause fainting, light-headiness, coma and even death. So, to keep BG within the desired physiological range of 70-180 mg/dL (named *normoglycemia/euglycemia*), T1D patients face a variety of burdensome tasks, such as: frequent monitoring of BG concentrations, self-administration of corrective insulin boluses, estimation of the correct amount of carbohydrate (CHO) at each meal, intake of fast-acting corrective CHO, etc. As a matter of fact, maintaining BG within this narrow physiological range, allows reducing the risk of death and the long- and short-term consequences of hyperglycemia (i.e., BG concentrations greater than 180 mg/dL) and hypoglycemia (i.e., BG concentrations lower than 70 mg/dL).

Hyperglycemia

Hyperglycemic episodes occur multiple times a day when glucose remains in the bloodstream instead of being used as energy. Among the main contributing factors there are: food and physical activity choices (aerobic vs anaerobic), illness, drugs, skipping or not injecting the correct insulin dose. Furthermore, several hormones that are released by the body in the early morning hours, can cause high blood sugar. This is known as the dawn phenomenon [4]. If hyperglycemia is not treated it can develop into ketoacidosis, where toxic acids build up in the blood. This condition can lead to long-term complications like neuropathy, retinopathy, kidney and micro-/macrovascular heart diseases. To lower BG, prolonged hyperglycemic events [5] can be treated by injecting exogenous insulin boluses which take into account the current BG level, the CHO content of the meal, body weight (BW) and other parameters set by physician, such as carbohydrate-to-insulin (CR) [6] and correction factor (CF) [7] which describe how many grams of CHO are covered by a unit of insulin and how much BG is lowered by each unit of insulin, respectively.

Hypoglycemia

Depending on multiple factors such as skipped meal, intensive physical exercise, alcohol, or excessive insulin treatment [8], hypoglycemic events can be

very disabling. In fact, low BG levels lead to short-term complications such as mental confusion, nausea, headaches, blurred vision. If not promptly treated, e.g., by having fast-acting rescue CHO, hypoglycemic events can become severe and lead to acute cognitive dysfunction, seizure, coma and even death [9]. Moreover, frequent hypoglycemic episodes cause a cascade of physiologic effects and may induce cardiac arrhythmias [10], contribute to sudden cardiac death and cause ischemic cerebral damage [11]. To raise low BG levels, the American Diabetes Association (ADA) suggests the so-called 15-15 rule [5] which requires to have 15 grams of carbohydrate and check BG after 15 minutes. If it is still below 70 mg/dL, patients have to repeat these steps until BG is back into *euglycemia*. It is worth underling that several studies [12, 13] indicate in the hypoglycemia and the fear of its complications the main limiting factor in achieving optimal glucose control.

1.2 Wearable devices and systems for T1D management

Over the past 15 years, technological advances in T1D management have grown rapidly [14]. As a matter of fact, individuals with T1D have begun to replace traditional glucometers with continuous or flash glucose monitoring devices, manual insulin injections with improved insulin delivery systems, and the manual calculation of insulin dosing with dedicated mobile applications that assist patient during the decision making process [14]. It is worth noting that the development of new technologies in diabetes care is rapidly moving toward personalized and user-independent devices that promise to further reduce the burden of the disease. This section reviews glucose monitoring technologies, insulin delivery systems, and applications such as artificial pancreas and decision support systems that integrate new technologies.

1.2.1 Self-monitoring of blood glucose (SMBG)

For a long time, self-monitoring of blood glucose (SMBG) was the only system patients could use to measure their BG concentrations [15]. Briefly, these measurements involve placing a small drop of blood (usually collected from fingertips) on a reagent test-strip, which is then inserted into a measurement device. Typically, these finger-stick measurements were taken 4-5 times per

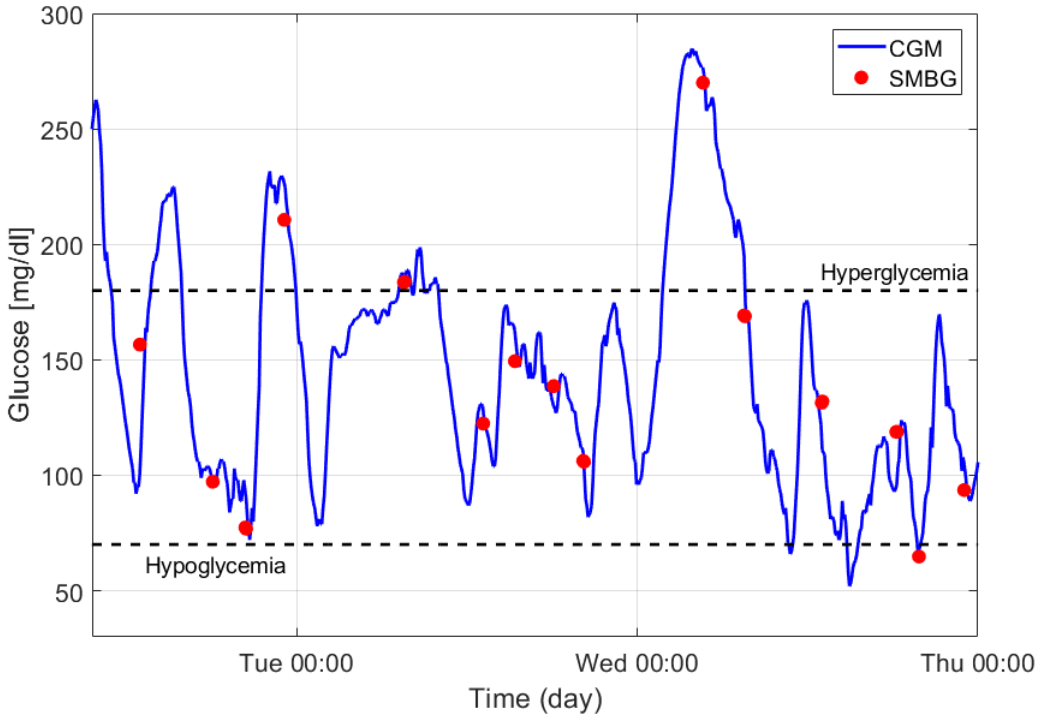


Figure 1.1: Representative glucose monitoring data: SMBG (red dots) and CGM (blue line). Data are extracted from the Dexcom G6 pivotal study (NC:02880267).

day: before meals, snacks and physical exercise, when a critical event was suspected and before/after a treatment to improve glucose control. Whilst SMBG provides very accurate information about BG levels [16], the limited number of samples is not sufficient to describe all the glycemic variability that occurs along the day.

As shown in Figure 1.1, most of the SMBG measurements (red dots) fall within the range and only a few of them evidence critical episodes. However, the glycemic profile (blue line) shows two hypoglycemic and three hyperglycemic episodes that are not detected by SMBG.

The inability to provide complete information of glucose dynamics, led to the development of new devices that allow to measure BG concentrations almost continuously, the so-called Continuous Glucose Monitoring systems.

1.2.2 Continuous glucose monitoring (CGM)

Today, BG monitoring is performed using continuous glucose monitoring (CGM) sensors, which allow collecting and visualizing glucose concentrations almost continuously (e.g., every 5 min) for several days [17, 18]. All commercial CGM devices are labeled as minimally invasive since they require either a micronee-

Figure 1.2: Example of CGM device. Dexcom G7 CGM System (Dexcom Inc., San Diego, USA). On the right the CGM sensor and transmitter, and compatible smart devices equipped with Dexcom G7 (mobile app or smartwatch). https://www.dexcom.com

dle or a small capsule to be inserted in the subcutis, and they represent an important innovation in T1D management because they allow reducing the burden of performing multiple daily invasive self-monitoring tests of BG concentrations. Figure 1.2 shows a novel CGM systems, which is composed by three main elements:

- a needle-based sensor, placed in the subcutis it measures the electrical signal which is proportional to BG concentration in the interstitial fluid;
- a transmitter, applied over the sensor aiming to send data to the receiver;
- a receiver, that converts the electrical signal into glucose concentration and displays it on a monitor or on a smartphone equipped with a dedicated app.

CGM devices are currently accepted as standard tools for glucose monitoring and they have proved to improve both insulin therapy and, as a consequence, T1D management [19, 20, 21]. In fact, most of these devices usually provide alerts that warn the subject when the CGM values exceed the normal glucose range. Also, some systems have allowed patients to customize these

alerts by setting high and low limits for different times along the day [22]. It is worth noting that the technological improvement has led to the regulatory approval of CGM used alone (the so-called non-adjunctive use) [23]. In this context, CGM is allowed to be used to make treatment decision without the need of confirmatory fingersticks. The safety of this treatment has been proven by computer simulations [24] and several randomized clinical trials [25, 26]. Finally, the use of CGM enables short-term prediction of future glucose levels and/or hypo-/hyperglycemic episodes. Therefore, targeted preventive measures -such as preemptive hypotreatment (rapid-acting carbohydrate consumption [27]) or corrective insulin boluses [28]- could be taken based on future rather than current blood glucose levels to reduce the occurrence and impact of critical episodes.

1.2.3 Insulin pumps and smart pens

Portable subcutaneous continuous insulin infusion (CSII) pumps are devices that continuously delivers insulin through a small catheter placed into the subcutis [29]. Generally, insulin is delivered in two ways: a continuous infusion of rapid-acting insulin throughout the day and night (called basal insulin) and a discrete infusion, i.e., one-time doses of rapid-acting insulin administered by the user at mealtimes or to correct a high blood glucose level (insulin bolus). Basal insulin delivery replaces the use of the longer-acting exogenous insulin formulations used in multi daily injections (MDI) set ups [30]. Despite several studies demonstrate the benefits of CSII systems in improving glucose control and reducing the risk of hypoglycemia [31, 32], approximately the 60% of patients affected by T1D in the US and 5-15% in Europe currently use insulin pumps [33, 34]. Considering the large number of patients following the MDI therapy, smart insulin pen technology has largely evolved over the last ten years. In fact, smart pens are similar to traditional insulin pens: the user has to prime a needle, set the insulin dose and use the depressing device for insulin delivery. In addition, smart insulin pens can communicate to other devices via Bluetooth connectivity, record and store data of meal and insulin injections (time and amount), as well as calculate the insulin bolus and trigger some reminder alerts [35].

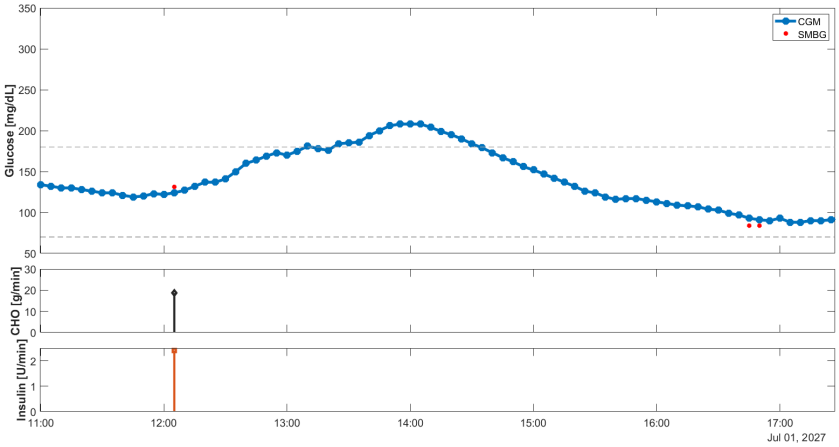
1.2.4 Artificial pancreas and decision support systems

Combining insulin pumps and CGM devices allows T1D patients to manually modify the infusion rate according to their BG levels, this is known as sensor augmented pump (SAP) therapy [36]. However, automatizing and optimizing insulin administration is made possible by Artificial Pancreas (AP) systems, emerging technologies that combine: CGM devices, infusion insulin pumps and a closed-loop control algorithm, usually fed by CGM readings, insulin and meal information [37, 38]. In the recent years, a greater emphasis has been given to model predictive control (MPC) algorithms which is based on a mathematical model describing glucose-insulin dynamics. Therefore, the control performances are largely influenced by the quality of the model. Ideally, a personalized whole-body physiologically based model would enable a successful closed-loop algorithm, however a large scale model can be hardly identifiable on a single individual without the use of invasive measurements [39]. Although several literature contributions deal with linear approximation of patient-tailored models, the individualization still represent one of the main challenges to achieve optimal glucose control [40].
Besides AP systems, a recent report on artificial intelligence (AI) applications for diabetes management [41] pointed out that the combined use of CGM devices, insulin pumps and dedicated mobile applications [42] has increased the development of advanced AI-enabled decision support systems (DSS). These composite tools implement multiple software to support the patient in the decision-making process. In particular, a DSS can comprise several modules that allow to reduce the daily burden and challenging routine of T1D management, by: i) computing the optimal insulin dose at meal-time by exploiting machine learning models [43] or by suggesting insulin adjustments from a set of predefined treatments exploiting supervised classifiers as in [44]; ii) counting the correct amount of CHO at meal by exploiting computer vision techniques [45] and deep learning approaches based on historical CGM measures, meals and insulin [46] or detecting an unannounced meal [47]; iii) warning patients via preventive alerts about upcoming impending critical events in order to avoid or mitigate the onset of hyper-/hypoglycemic episodes [44, 48]. This last point, as well as the forecasting of BG levels is a relative mature field that has received vast attention in the scientific community for its potential to revolutionize diabetes care as witnessed by the large number of proposed algorithms and published reviews [49, 50, 51]. The next section provides a brief

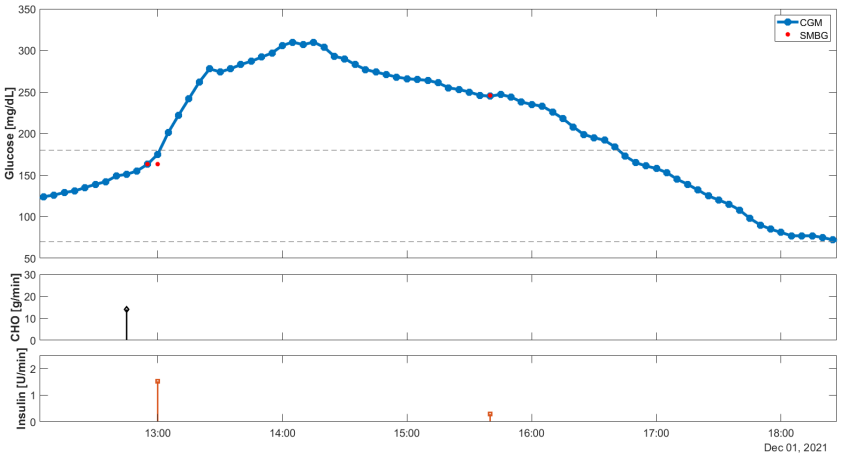
description of the main approaches for BG forecasting and hypoglycemia prediction.

1.3 Blood glucose forecasting in T1D

BG concentrations are influenced by multiple metabolic factors caused by daily activities such as eating, injecting insulin, exercising, driving, etc. [52]. In addition, because of the large inter- and intra-individual variability, the effect over time of these factors on BG may be different. As a consequence, an accurate prediction of future glucose levels faces several challenges.



(a) patient ID: 544



(b) patient ID: 575

Figure 1.3: Postprandial responses for similar amount of CHO and insulin, patient ID 544 (a) patient ID 575 (b). CGM trace (blue dotted line, upper panel), SMBG (red dots, upper panel), CHO content of meals (black diamonds, middle panel) insulin boluses (orange squares, bottom panel). Data are extracted from the Ohio Type 1 Diabetes Mellitus dataset [1].

To better highlight some of these issues, Figure 1.3 shows postprandial glycemic profiles (blue dotted line) along with SMBG (red dots), ingested CHO during meals (black diamonds, middle panel) and injected insulin boluses (orange squares, bottom panel) of two individuals with T1D, patient ID: 544 in Figure 1.3a and patient ID: 575 in Figure 1.3b. The euglycemic range is delimited by the grey dashed lines. It is worth noting that two hours before the selected meals BG was stable and patients did not ingest CHO or inject insulin.

Although the initial conditions and the total amount of CHO and insulin are similar, the postprandial responses in Figure 1.3a and Figure 1.3b, are different. Specifically, Figure 1.3a shows a smooth increase in glucose concentration of approximately 85 mg/dL two hours after the meal, with a peak value of 208 mg/dL. In contrast, in Figure 1.3b, BG rises very sharply: the postprandial peak occurs about an hour and a half after meal intake, with a glucose excursion of 160 mg/dL and a peak of 310 mg/dL. In this case, the patient experiences a prolonged hyperglycemic event even if an insulin bolus was injected just after mealtime to counterbalance the effect of CHO on BG levels.
Figure 1.3a and Figure 1.3b seem to suggest that BG predictive algorithms should, in principle, use individualized models fed by various information: certainly the past history of glucose concentrations measured by the CGM sensor, but also ingested CHO and injected insulin may play a major role. However, accounting for all these inputs, formalizing in mathematical terms, and extracting useful signals from them is not a trivial task.
Facing these challenges, in the last 20 years, several research groups have developed methodologies for the prediction of future BG concentrations. In particular, literature contributions can be classified according to:

- the model under consideration: physiological vs. black-box, the former using white-box models to describe glucose-insulin dynamics, the latter using techniques from the fields of time-series forecasting, system identification, machine learning, and deep learning;

- the data employed as input: CGM data only vs CGM data plus additional exogenous information.

In the following section we provide a brief overview about the main contributions in the field. Table 1.1 provides a summary of the contributions evidencing the model type, the input data, the number of patients and the performance in terms of root mean square error (RMSE) for several prediction horizon (PH = 30, 45 and 60 minutes). For detailed reviews, we refer the reader to [49, 50, 51].

1.3.1 Black-box models

Time series approaches

Time-series modeling approaches based on autoregressive (AR), autoregressive moving average (ARMA) models represent the first and common strategies employed in literature to forecast BG values. These techniques assume that future BG concentration can be computed as a linear combination of previous CGM readings. Sparacino et al. [53], assessed the possibility of predicting glucose ahead in time by using a first-order AR model which parameter was recursively updated to describe the rapidly changes in glucose dynamics. Reifman et al. [54] developed a regularized AR model of fixed order, which parameters are personalized to patient data. On the contrary, Gani et al. [55], developed a time invariant AR model of order 30, which shows the feasibility of developing a single model that can be used for all subjects. Eren-Oruklu et al. [56], proposed an adaptive univariate ARMA model with fixed model orders, chosen according the Akaike Information Criterion (AIC), that can be used not only to predict BG levels but also to forecast hypoglycemic and hyperglycemic events. In Otoom et al. [57], an online estimation algorithm based on autoregressive integrated moving average (ARIMA) model is found to be the best model for predicting BG levels. Finally, in Yang et al. [58], an ARIMA model with adaptive order selection shows outperforming results both in terms of BG levels predictions and early hypoglycemic alarms.

Models commonly used in system identification

While time series models are fed only with CGM data, this paragraph describes the use of their variants with eXogenous input (ARX, ARMAX and ARIMAX) for the prediction of future glucose concentrations. One of the first attempt in this field was made by Finan et al. [59], where both a time invariant and time variant ARX model based on glucose, insulin and CHO information were assessed on two datasets. Also, the use of multivariate ARMAX models has been considered as in Eren-Oruklu et al. [56]. Of note, a branch of linear model identification has moved towards the use of personalized and recursive approaches (i.e., models' parameters are updated each time a new sample arrives), while the other branch has focused intensively on the study of novel identification strategies. In particular, glucose-insulin model identification using kernel-based and nonparametric methods, as in Georga et al. [60] and in Del Favero et al. [61], has been shown to significantly improve predictive

performance compared with state-of-art parametric strategies, as reported in Faccioli et al. [62]. As a final remark, these models are mainly fed by past history of CGM, meal and insulin information. However, as described in Hobbs et al. [63] and Faccioli et al. [64] linear models can be successfully exploited also for taking into account physical exercise data.

Machine and deep learning strategies

As noted in Oviedo et al. [50] and Woldaregay et al. [51], last years have seen an increasing trend in the use of machine and deep learning techniques. The most used techniques for BG prediction are based on shallow feed-forward neural network (NN) as in Perez-Gandìa et al. [65], Zecchin et al. [66], and Kushner et al. [67]. Also shallow recurrent neural network (RNN) represents a valid tool when referring to its success in temporal sequence processing and regression. In fact, its modified version, the long-short term memory (LSTM) is one of the most used method for BG forecasting as reported in Xie et al. [68], Aliberti et al. [69], and Sun et al. [70]. It is worth noting that a large number of literature works, see for instance Bunescu et al., Mirshekarian et al., Bertachi et al. [71, 72, 73], combined physiological models with machine learning algorithms in order to improve prediction accuracy (support vector regression, SVR; NN and LSTM, respectively). According to Woldaregay et al. [51], less popular strategies are the one based on genetic programming techniques, Contador et al., [74], autoregressive neural network (NAR), Aliberti et al. [69], regression random forest (regRF), Georga et al. [75] and ensemble techniques combining multiple single learners, Wadghiri et al. [76]. Considering the complexity of BG dynamics, deep learning techniques may represent a suitable option due to their ability to automatically learn feature from input data, as reported in Schmidhuber et al. [77], Zhu et al. [78]. Among others, multilayer convolutional neural networks (CNN) have been recently developed and tested in simulated framework showing outperforming results, Li et al. [79], Daniels et al. [80]. However, a common challenge in the machine learning, especially in deep learning field, is the large amount of data required to accurately train subject-specific models. To overcome this issue, some approaches focused on transfer learning strategies. This learning paradigm leverages knowledge from previous training examples to improve the learning for a specific task, as in Daniels et al. [80], and De Bois et al. [81].
As a final comment, more than half of the machine/deep learning algorithms used one or two additional input information, usually insulin doses or CHO or

both, as reported in Woldaregay et al. [51]. Also in this case, the use of physical activity as input is substantially less investigated, as reported in Zarkogianni et al. [82] and in van Doorn et al. [83].

1.3.2 Physiological models

While all the literature works presented so far has considered black-box models that focus on the input-output relationship, only a few papers in the literature have dealt with white-box models that take into account the physiological information about the underlying system. There are two main types of physiological models: i) *minimal models*, such as the Bergman et al. model [84], that proposed a simplified descriptions of the physiology with a few differential equations and parameters (to capture the essential glucose-insulin dynamics) and ii) *maximal models*, such as the Hovorka et al. [85] and the Dalla Man et al. [86] model, comprising several differential equations and parameters to provide a more detailed description of the glucose-insulin system. Of note, maximal models are commonly used for computer simulation rather than for prediction purposes. A first attempt to predict BG using a physiological model was proposed by De Pereda et al. [87]. A limitation of these approaches concerns the identification of subject-specific parameters capable of describing the large intra-patient variability in blood glucose levels. For this reason, several contributions investigated different identification strategies as in Laguna-Sanz et al. [88], and Visentin et al., [89]. One of the latest works employing physiological model was proposed by Liu et al. [90], where authors employed a model of glucose regulation composed by the minimal model (i.e., Bergman et al. model, [84]) and by the insulin and CHO absorption models proposed by Hovorka et al. [85] to increase the accuracy in long-term prediction.
As a final remark, the main aim of this brief overview was to show that several predictive algorithms have been developed in the last 15 years and there has been a recent increase in the use of machine and deep learning techniques for BG forecasting, thanks to the large amount of data made available by the improved technologies for diabetes care. As a result of this scientific effort, simple algorithms (mainly based on linear CGM trend extrapolation) have been incorporated into CGM devices (e.g., predictive glucose alert in Dexcom G6/G7 devices) and into commercial SAP systems, as in Zhong et al. [91], Forlenza et al. [92] which have been proven to reduce hypoglycemia by using predictive glucose alerts and a predictive low-glucose insulin suspension system.

Study	Model	Input	Dataset	RMSE [mg/dL]		
				PH = 30 min	PH = 45 min	PH = 60 min
Sparacino et al. [53]	AR	CGM	28 T1D	18.78	34.64	-
Gani et al. [55]	AR	CGM	34 T1D^{+}	3.5	-	-
Reifman et al. [54]	AR	CGM	15 T1D	22.2	-	32.3
Yang et al. [58]	ARIMA	CGM	49 T1D	N.A.	N.A.	N.A.
Finan et al. [59]	ARX	CGM, insulin and meal	6 T1D	27	-	45
Georga et al. [60]	KRLS-ALD	CGM, insulin, meal and physical activity	7 T1D	15.22	-	31.65
Faccioli et al. [62]	NP	CGM, insulin and meal	11 T1D	17.5	-	29.8
Hobbs et al. [63]	VARX	CGM, insulin, meal and heart rate	32 T1D	26.33	-	-
Perez-Gandia et al. [65]	NN	CGM	6 T1D	17.5	27.1	-
Zecchin et al. [66]	NN	CGM and meal data	15 T1D	14	-	-
Kushner et al. [67]	NN	CGM and insulin data	15 T1D	-	-	28
Xie et al. [68]	LSTM	CGM, insulin and meal data	12 T1D	19.67	-	-
Aliberti et al. [69]	LSTM	CGM	451 T1D^{+}	5.93	-	-
Sun et al. [70]	bi-LSTM	CGM	20 T1D	21.74	30.21	36.92
Bunescu et al. [71]	PHY+SVR	CGM, insulin and meal	5 T1D	19.5	-	35.7
Mirshekarian et al. [93]	LSTM	CGM, insulin and meal	5 T1D	20.3	-	35.5
Bertachi et al. [73]	PHY+NN	CGM, insulin and meal	6 T1D	19.33	-	31.72
Georga et al. [75]	regRF	CGM, insulin and meal	27 T1D	8.15	-	9.25
Li et al. [79]	CNN+LSTM	CGM, insulin and meal	10 T1D	21.07	-	33.27
Daniels et al. [80]	CNN+LSTM	CGM, insulin and meal	12 T1D	18.8	25.3	31.8
De Bois et al. [81]	CNN	CGM, insulin and meal	12 T1D	19.61	-	-
van Doorn et al. [83]	LSTM	CGM and physical activity	6 T1D	21.42	-	34.52
Liu et al. [90]	PHY	CGM, insulin and meal	10 T1D	17.67	-	30.36

Table 1.1: A review of different contributions dealing with BG forecasting. Notation: RMSE (root mean square error), AR (autoregressive), ARMA (autoregressive moving average), ARIMA (autoregressive integrated moving average), KRLS-ALD (approximate linear dependency kernel recursive least squares), VARX (vector autoregressive integrated with exogenous input), NP (non-parametric), NN (feed forward neural network), LSTM (long-short term memory neural network), PHY (physiological model), SVR (support vector regression), regRF (regression random forest), CNN (convolutional neural network); $^{+}$ indicates data are filtered; N.A. not available.

1.4 Aim and structure of the thesis

As described in the previous sections, BG forecasting in T1D is a relatively mature field. However, despite the scientific community has made great efforts in developing predictive algorithms, there are still some open issues. First, it is not completely clear what are the most important signals that should be used as input information to improve the forecasting accuracy. As an example, adding insulin and CHO data to CGM is expected to enhance predictive performance. This has been demonstrated in several clinical trials where patients were selected for their ability to count a correct amount of CHO. However, in daily-life setting, the meal information are burdensome for the users to record and prone to error, thus potentially leading to inaccurate predictions. Another understudied point concerns the most appropriate approach for BG forecasting: linear, nonlinear machine/deep learning, black-box or physiological models. As a matter of fact, nonlinear methodologies are likely to be potentially more accurate than linear models because of their ability to learn complex dynamics, but they usually require a large amount of data for training, thus limiting predictive performance if this step is not performed correctly.

To address these gaps, this thesis aims to understand the role of different input information and the contribution of simple vs complex strategies for the prediction of future BG concentrations. An additional aspect that will be addressed is to find the most suitable strategy to exploit the so-obtained BG predictions to forecast an upcoming hypoglycemic event. To reach the aim, our analysis has been performed by resorting to datasets recorded in real-life conditions. In particular, data derive from: the Dexcom (Dexcom Inc., San Diego, CA, USA) pivotal study for the assessment of the G6 CGM sensor (NCT02880267); the Ohio Type 1 Diabetes Mellitus dataset [1] and from the outpatient control-to-range pilot study (CTR3): a closed-loop 5-month trial [94].

Chapter 2 is dedicated to a head-to-head investigation of linear and nonlinear state-of-art approaches that employ only CGM data as input. It will be shown that individualized linear models are more effective than population ones, while no significant advantages seem to emerge when employing nonlinear methodologies.

Chapter 3 investigates how much adding meal timing and seasonality information to CGM data can improve predictive performance. This is possible thanks to the use of a novel methodology based on clustering and stochastic

seasonal local models. It will be shown that this approach outperforms CGM-only-based predictors and it provides comparable performance to other more complex (linear and nonlinear) prediction methods fed with more information (i.e., CGM, timing and dosing of insulin and CHO). Despite satisfactory performance in predicting future BG values over the glycemic range, the prediction of hypoglycemic events (which involves a comparison of predicted and measured BG values and a classification-like framework) remains a challenge for all algorithms.

Chapter 4 describes the development of a novel algorithm to improve the prediction of hypoglycemic events. In particular, by employing one the most performing model found in Chapter 3, two main innovations are introduced: the use of a cost function to take into account the clinical impact of the prediction error and the use of confidence intervals for multiple prediction horizons. This approach outperforms conventional methods for hypoglycemia forecasting and, with adequate adjustments can be translated also to machine and deep learning methodologies.

So far, previous chapters have focused on black-box approaches. For this reason, *Chapter 5* aims to investigate whether the use of a white-box model taking into account the physiological information of the glucose-insulin system can improve predictive performance. In detail, the proposed physiological white-box model is identified from patient data (i.e., CGM, meal, and insulin) and used within the particle filter framework to predict BG ahead in time. Unfortunately, the results do not support our hypothesis since data-driven techniques significantly outperform the physiological white-box model.

Despite black-box models can grant accurate performance, their results can be difficult for the users to explain. In addition, when competing models achieve similar performance, the model to be used in practice should provide the most straightforward physiological interpretation. For these reasons, *Chapter 6* addresses one of the main issues when using black-box models: the lack of interpretability of the outcome. This can be done by: i) resorting to Shapley addictive explanations: a novel game-theoretic approach to explain the output of any machine learning model and ii) developing a case-of-study in which the interpretation should be preferred over prediction accuracy when choosing between two data-driven models achieving similar predictive performance.

Chapter 7 summarizes the main findings, draws some concluding remarks about the research carried out in the present thesis, discusses the limitations of the studies and proposes some future works.

Chapter 2

Forecasting of glucose levels and hypoglycemic events employing CGM data only

[1] The accurate forecast of blood glucose levels and/or hypoglycemic episodes can play a key role in improving T1D management by triggering proactive therapeutic actions to mitigate or to avoid impending critical events. At the present time, predictive algorithms based on CGM data only remain a very valuable option, as the acquisition and synchronization of datastreams from other data sources (e.g., meal and insulin information, physical activity, etc.) is not always straightforward in a real-time setting. Several contributions in the literature have tackled this problem, but comparing their findings is not trivial due to different data collection conditions (highly controlled set-ups, such as inpatient trials, as opposed to real-life recordings), preprocessing methods, and evaluation metrics.

This chapter offers a head-to-head comparison by systematically comparing several linear and nonlinear prediction algorithms and examining a number of degrees of freedom in their design on a same dataset, acquired in daily life conditions with one of the latest CGM sensors available on the market.

[1] This chapter contains material published in *Prendin et al., Sensors, 2021*, [95].

2.1 Chapter introduction and content

2.1.1 Rationale for the investigation of predictive algorithms fed only by CGM data

As described in the previous chapter, CGM devices have proved to be useful in improving insulin therapy and, in general, T1D management [19, 20, 21], and they are currently accepted as standard tools for glucose monitoring. Most of these devices usually provide alerts that warn the subject when the CGM values exceed the normal glucose range. Furthermore, the employment of CGM to provide short-term predictions of future glucose values or to forecast forthcoming hypo-/hyperglycemic episodes could lead to a further improvement, since targeted preventive measures -such as preventive hypotreatments (fast-acting carbohydrate consumption [27]) or correction insulin boluses [28]- could be taken to reduce the occurrence and impact of these critical episodes. Therefore, the availability of an effective BG predictive algorithm becomes of primary importance for present and future standard therapies.

In the last two decades, several algorithms for the short-term prediction of future glucose levels have been developed, using both CGM data only (to mention but a few representative examples, see [58, 70, 96, 97, 98]) and CGM data plus other available information such as the amount of ingested CHO, injected insulin, and physical activity (see, for example [50, 79, 99, 100, 101]). While the use of these additional datastreams is expected to enhance prediction performance compared to algorithms based on CGM data only [101], a nonnegligible drawback is that their application in real-world scenarios requires supplementary wearable devices (e.g., insulin pumps, mobile applications, and physical activity trackers) and actions (e.g., the safe and reliable exchange of information from one device to the other, and interactions with the user). Indeed, at present, these systems are not extensively used by individuals with diabetes [102, 103]. Consequently, the possibility of efficiently performing the real-time prediction of future glucose levels with CGM data only remains, at the present time, a practically valuable option. This is the reason why investigating the performance of predictive algorithms fed by CGM data only is of primary importance.

2.1.2 Chapter contribution

In the last 15 years, many real-time predictive algorithms based on CGM data only have been proposed in the literature [55, 82, 104, 105, 106, 107]. However, it is very difficult to establish which of them is the best performing one. Indeed, the mere comparison of performance indices extracted from different published papers could be unfair or misleading, because differences in datasets, implementation, preprocessing, and evaluation can make it difficult to claim that one prediction method is the most effective. The attempts to compare state-of-the-art methods and literature contributions on the same dataset are, to the best of our knowledge, very limited. A systematic review of glucose prediction methods has been proposed by Oviedo et al., in 2017 [50]. Nonetheless, the focus of [50] is on a methodological review rather than on performing a head-to-head comparison on the same dataset. A recent comparison of different prediction algorithms on the same dataset has been proposed by McShinsky et al. in [108]. A difference with the present contribution is that McShinsky et al. includes both CGM-only prediction methods and algorithms relying on other signals and involves a small population (12 subjects).

To fill this gap and to offer a performance baseline for future work, in this chapter, we present a head-to-head comparison of thirty different real-time glucose prediction algorithms fed by CGM data only on the same dataset, which consists of 124 CGM traces of 10-day duration collected with the Dexcom G6 CGM sensor. Notably, this sensor is one of the most recently marketed, and its employment allow us to also assess if some previous literature findings still hold with more modern, accurate CGM sensors. Specifically, we test linear black-box models (i.e., AR, ARMA, and ARIMA), nonlinear machine-learning (ML) methods (i.e., SVR, regRF, and NN), and a deep-learning (DL) model (i.e., LSTM). For the linear and ML methods, we consider both population and individualized algorithms. The former are one-fits-all algorithms, designed to work on the entire population; the latter are algorithms customized for each single patient based on their previously collected data, in order to deal with the large variability in glucose profiles among individuals with diabetes. Moreover, given the different nature of glucose fluctuations during the day and night (larger in the former case due to meal ingestion and less pronounced in the latter case) [58, 101], we design specific versions for these two time periods. With regard to model training, we opportunely divide the dataset into training and test sets, also performing a Monte Carlo simulation to avoid the

possibility of the numerical results being related to a specific training-test partitioning. The performance of all the algorithms are evaluated on a 30 min prediction horizon (PH) focusing on both prediction accuracy and the capability of detecting hypoglycemic events.

2.1.3 Chapter outline

This chapter describes the different modeling strategies (Section 2.2) that have been considered to develop linear black-box models (Section 2.3), nonlinear ML methods, and the DL model (Section 2.4). With regard to model assessment, we opportunely split the dataset into training and test sets, also performing a Monte Carlo simulation (Section 2.6) to avoid the possibility of the numerical results being related to a specific training-test partitioning. The results (Section 2.7) show that, in terms of BG prediction, the best-performing linear and nonlinear methods are comparable, while the first slightly outperforms the second in terms of hypoglycemic prediction. In addition, the results support the importance of individualization of the model, while no significant advantages emerge when employing nonlinear strategies.

2.2 Modeling strategies for developing predictive algorithms

Several options for creating the different variants of the considered classes of prediction algorithms are investigated. In order of increasing complexity, the first option is to consider a population algorithm that computes the prediction of the future CGM value by using the same model (i.e., structure and/or order) and the same parameters for all the individuals, i.e., without any personalization. This has the practical advantage that the model training can be performed only once, e.g., when the algorithm is designed, and the model learning procedure can leverage large datasets of CGM traces. The downside of this approach is that the prediction algorithm is not customized according to individual data [55]. Another option, with complexity higher than that of the previous one, is to develop subject-specific algorithms, which allows taking into account the large interindividual variability characterizing T1D individuals. The drawback of this approach is that the model training must be repeated for each individual in order to enable personalized glucose predic-

tions. A further level of complexity is to consider multiple models for each individual, e.g., one for day time and one for night time. The key idea behind this choice is that the "day-time" model should be able to learn the glucose dynamics perturbed by all the external events (e.g., meals, insulin injections, and physical activity), whereas the "night-time" model should be able to learn the smoother dynamics present at night time [101]. Since no information on sleep time is available in our dataset, we decide to define day time as the interval from 6:00 up to 23:00 and night time as that from 23:05 up to 5:55. According to the rationale discussed above, the resulting categories of prediction algorithms tested in this work are summarized in the tree diagram reported in Figure 2.1. For each category, several different model classes are considered, for a total of 30 different prediction algorithms. A detailed description of the prediction algorithms tested is provided in the following two subsections.

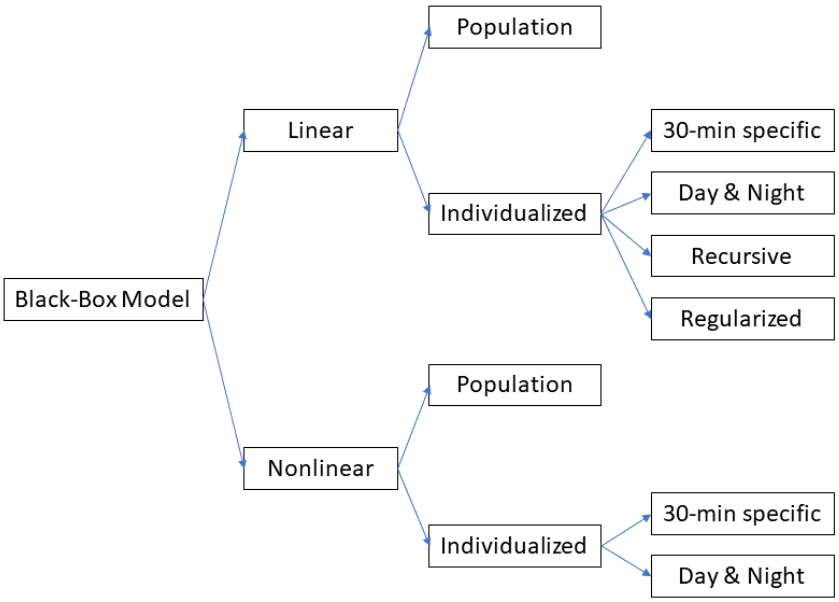


Figure 2.1: Schematic tree diagram of the main approaches tested in this chapter.

2.3 Linear black-box models

Linear predictive algorithms are based on a model of the CGM time series. Such a model is derived by applying the standard pipeline described in [109]. The first three steps, i.e., the choice of the model class, model complexity, and parameter estimation, are related to model learning. The last step is model prediction, which deals with the computation of the predicted value, employing the model and the past CGM data. These steps are described below.

2.3.1 Choice of the model class

Three linear model classes are considered: autoregressive (AR), autoregressive moving average (ARMA), and autoregressive integrated moving average (ARIMA) models. In the following sections, we use the notation AR(p), ARMA(p,m), and ARIMA(p,m,d), indicating with p, m, and d the order for the AR, MA, and integrated (I) part, respectively.

2.3.2 Model complexity

Once the model class is fixed, the model complexity, i.e., the number of parameters to be estimated, has to be chosen. Common techniques used for this purpose are the Akaike information criterion (AIC), the Bayesian information criterion (BIC), and cross validation (CV) [109, 110]. The model orders p and m are, respectively, searched in the sets P = 1,2,...,30 and M = 0,1,...,15. After a preliminary analysis, showing that no significant differences can be seen between these methods (not shown), the BIC is chosen as the method for selecting the best model orders. Concerning the individualized linear models, we investigated a partial personalization: the model complexity of the population algorithms is maintained, but the parameter values are subject-specific (a model with individualized parameters and population orders). Then, a complete personalization is achieved by learning both the model complexity and the parameter values from patient data (a model with individualized parameters and individualized orders).

2.3.3 Parameter estimation

The first approach we use to estimate model parameters is the state-of-the-art prediction error method (PEM) [109], based on the minimization of the one-step prediction error. Furthermore, since we focus on 30-min-ahead prediction, we also consider the possibility of identifying the model parameters that minimize the 30-min-ahead prediction error (30 min-specific) rather than the 5-min-ahead error as prescribed by the standard pipeline.

With these estimation techniques, CGM time series are described by models with fixed structures and time-invariant parameters. To better follow intrapatient variability, we also investigate recursive least-squares (RLS) parameter estimation [53], which is applied, without any loss of generality, only to the AR(1) model, since previous work demonstrated the effectiveness of the AR-

RLS(1) configuration [111]. Note that the RLS estimation requires setting an additional parameter, the forgetting factor, which represents a memory term for past input data [112]. This AR-RLS(1) falls into the category of a model with a fixed structure but time-varying parameters. Another option we consider, is the regularized PEM approach, which considers AR models of elevated order (e.g., p = 100) and adds to the standard PEM cost function a regularization term representing a suitable prior on the unknown coefficients, which allows avoiding overfitting [110]. A suitable prior, known as stable spline kernel, is adopted in this work [61]. To avoid unstable models are used for the forecasting, the choice of the model complexity and the parameter-estimation steps are repeated until a stable model is identified.

2.3.4 Model prediction

Once a linear model is available from the previous steps, the k-step-ahead prediction can be derived for any value of k. This is performed by applying a standard Kalman filter framework [109]. We use this approach to derive the 30-min (k = 6)-ahead prediction. We decided to focus on PH = 30 min only for two main reasons. First, the literature work [27, 104], and [113] has shown that efficient corrective actions (e.g., hypotreatments or pump suspension [104, 113]) triggered 20–30 min before hypoglycemia are effective in avoiding/mitigating the episodes. Second, it has been shown that PH = 30 min is a good trade-off between limiting the error of the prediction outcome (the higher the PH, the higher the error) and the effectiveness of the prediction [48].

2.4 Nonlinear black-box models

A learning pipeline similar to that adopted for the linear models is employed for ML and DL predictive algorithms. The main steps in the learning phase are the choice of the model class, the tuning of hyperparameters (the counterpart of the model complexity), and model training (i.e., parameter estimation). The last step consists of computing the 30-min-ahead glucose prediction once the nonlinear model is obtained.

2.4.1 Choice of the model class

Three ML models, successfully used in a wide range of regression problems, are considered: support vector regression (SVR) [71, 114], regression random forest (RegRF) [115], and feed forward neural network (NN) [65]. In addition, we considered a DL model, namely, long short-term memory (LSTM) network, which has shown promising results in glucose prediction [69, 116]. The key idea of the SVR model is to map the CGM data into a higher-dimensional feature space via a nonlinear mapping and, then, to perform a linear regression in such space [117]. The goal of SVR is to find a function that has, at most, ϵ deviation from the target in the training data. Moreover, the use of adequate kernels allows dealing with linearities and nonlinearities in data [118].

RegRF is an ensemble learning method based on aggregated regression trees. A regression tree is built by recursively top–down-partitioning the feature space (composed of CGM values) into smaller sets until a stopping criterion is met. For each terminal node of the tree, a simple model (e.g., a constant model) is fitted [119]. The prediction of RegRF is obtained by combining the output of each tree.

The NN model allows learning complex nonlinear relationships between input and output values [120]. It is composed of a set of neurons organized in layers (input, hidden, and output layers). Each neuron is characterized by a nonlinear function, e.g., sigmoid, which provides the input for the next layer, and by weights and biases. These parameters are learned from the data and are determined in order to achieve the minimum value of the cost function during the training phase. The output layer is a linear combination of the output of the previous layers.

LSTM is a useful model when maintaining long-term information over time is relevant to learn dependency and dynamics from data [121]. The key element of the LSTM model is the memory cell composed of four gates (forget, input, control, and output gates) that decide whether the information must be kept or removed from this cell at each time step. Note that, given the large number of parameters needed by LSTM and the relatively short CGM time series available for each subject in the dataset, in this work, it was not possible to apply the individualized approach for LSTM. Thus, for the LSTM model, we limited the analysis to the population approach only. In addition, since the focus of the paper is on a predictive algorithm fed by CGM data, the LSTM features were lagged CGM samples only. Further details on LSTM are avail-

able in Appendix B. A detailed review of these methods is beyond the scope of this work, and we defer the interested reader to the original work or to [122].

2.4.2 Input size and hyperparameter tuning

For each ML model, the optimal input size (i.e., the number of consecutive CGM readings) and other model-specific hyperparameters are chosen by using a grid search approach combined with hold-out-set CV [109] to avoid overfitting. A list of the model-specific hyperparameters and their values are reported in Table 2.1.

Concerning LSTM, given the dimensions of our dataset and the elevated number of hyperparameters to be tuned, we decided to manually set some of them, such as the number of layers, learning rate, and decay factor, on the basis of literature studies to avoid the risk of overfitting [93, 116]. This approach proved to be efficient in reducing such a risk in even more complex and deep neural networks [70, 79, 98]. Moreover, to further strengthen the learning phase, we added a dropout layer to the LSTM, which randomly ignored neurons during the training. Finally, based on the results of the hold-out-set CV, we found that the optimal LSTM structure consisted of a network composed of a single LSTM layer, 30 hidden nodes, and 10 lagged CGM values as input.

As for the individualized linear models, we also investigated a partial personalization for nonlinear ones: the hyperparameters and optimal input size of the population algorithms are maintained, but the parameter values are subject specific (a model with individualized parameters and population hyperparameters). Then, a complete personalization is achieved by determining the model-specific hyperparameters, the optimal input size, and the parameters based on individual data (a model with individualized parameters and individual hyperparameters).

2.4.3 Model training

Independently of the modeling strategy considered (i.e., population, individualized, or day/night specific), the CGM data are standardized using z-score standardization [122]. Then, parameter estimation is performed by minimizing the model-specific loss function by using specific optimized versions of the stochastic gradient descent algorithm.

Table 2.1: Nonlinear models hyperparameters

Model	Hyperparameter	Range
SVR	error penalty term, kernel scale factor	[10^{-3}-10^{3}] (logarithmic scaled)
RegRF	number of trees	[10-500]
	number of leaves, max number of splits	[1-max(2,training samples)] (logarithmic scaled)
NN	number of layers	[1-3]
	number of neurons	[5-20]
	activation function	Hyperbolic tangent, sigmoidal
	max training epochs	[500-1500]
LSTM	number of hidden units	[20-100]
	max training epochs	[50-1000]
	dropout rate	[0.01-0.7]

2.4.4 Model prediction

The three previous phases allow learning a model that can directly produce the 30-min-ahead-in-time prediction, once fed by a sequence of standardized CGM data.

2.5 Criteria and metrics for the assessment of the algorithms

The algorithms are compared considering both the accuracy of the glucose value prediction and the hypoglycemia event detection capability.

2.5.1 Glucose value prediction

The predicted glucose profiles are evaluated with three commonly used metrics. First, we considered the root mean square error (*RMSE*) between the predicted glucose values and measured CGM data:

$$RMSE = \frac{1}{\sqrt{N}} ||(y(t) - \hat{y}(t|t-PH))||_2 = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(y(t) - \hat{y}(t|t-PH))^2} \quad (2.1)$$

where PH is the prediction horizon, N is the length of the subject CGM data portion in the test set, $y(t)$ is the current CGM value, and $\hat{y}(t|t-PH)$ is its PH-step-ahead prediction. By $||x(t)||_2$, we denote the Euclidean norm of the signal *x(t)*, namely: $||x(t)||_2 = \sqrt{\sum_{t=1}^{N}(x(t))^2}$.

RMSE takes positive values, with RMSE = 0 corresponding to the perfect prediction, and increasing RMSE values corresponding to larger prediction errors.

Furthermore, we also considered the coefficient of determination (COD):

$$COD = 100 \cdot (1 - \frac{||(y(t) - \hat{y}(t|t-PH))||_2^2}{||(y(t) - \bar{y}(t))||_2^2}) \tag{2.2}$$

where $\bar{y}$ is the mean of the CGM data. The COD presents the maximum value (i.e., 100%) if the predicted profile exactly matches the target CGM signal. If the variance of the prediction error is equal to the variance of the signal or, equivalently, if the prediction is equal to the mean of the signal, the COD is 0%. There is no lower bound for COD values (they may also be negative).

Finally, the delay existing between the CGM signal and the predicted profile is defined as the temporal shift that minimizes the square of the mean quadratic error between the target and the prediction:

$$delay = \underset{j \in [0,PH]}{\arg\min} \left[\frac{1}{N} \sum_{t=1}^{N-PH} (\hat{y}((t|t-PH)+j) - y(t))^2 \right] \tag{2.3}$$

Of course, the lower the delay, the prompter and more useful the prediction. A delay equal to the PH means that the model prediction is not better than looking at the current glucose level. Finally, in order to investigate if significant differences exist among the compared algorithms, a one-way analysis of variance (ANOVA) is used to compare the RMSE values. A significance level of 5% (p-value < 0.05) is considered in all cases. The adjustment for multiple comparisons is performed by using the Bonferroni correction.

2.5.2 Hypoglycemia prediction framework

Concerning the assessment of the ability to predict hypoglycemic events, following [48], we defined the occurrence of a new hypoglycemic event when a CGM value below 70 mg/dL is observed and the previous six CGM readings are above 70 mg/dl. An example of a hypoglycemic event is shown in Fig-

ure 2.2. Hypoglycemic alarms are defined for the predicted CGM signal with exactly the same criteria used for hypoglycemic event definition.

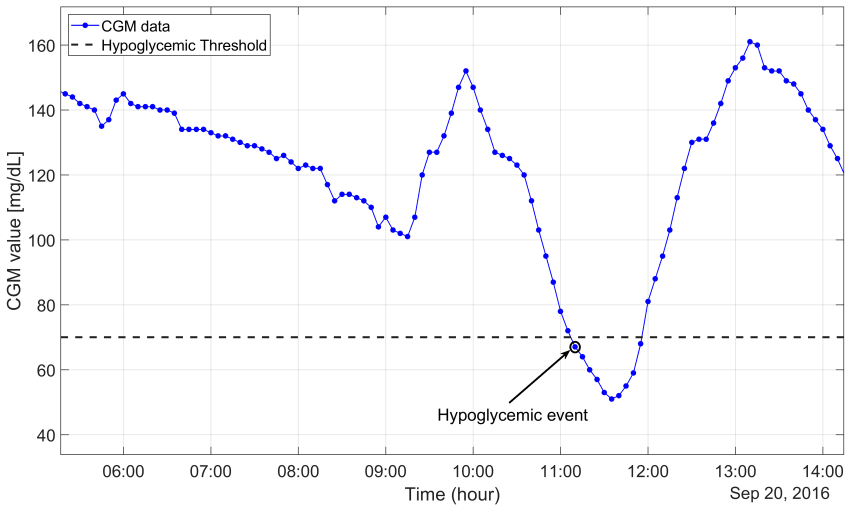


Figure 2.2: Example of hypoglycemic event onset, CGM data (blue dotted line).

Hypoglycemia prediction metrics

Considering a PH = 30 min and detection window (DW) of 40 min, we assign:

- True positive (TP): if an alarm is raised at least 5 min before the hypoglycemic event and at most DW+5 min before that episode, as shown in Figure 2.3 (top left panel). According to this definition, alarms raised with a time anticipation larger than DW+5 min are not counted as TPs, because it is difficult to claim a match between the alarm and the hypoglycemic event;

- False positive (FP): if an alarm is raised, but no event occurred in the following DW minutes, as shown Figure 2.3 (top-right panel);

- False negative (FN): if no alarm is raised at least 5 min before the event and at most DW+5 min before the event, as shown in Figure 2.3 (bottom-left panel);

Finally, we define as late alarms the alarms raised within DW minutes after the hypoglycemic event, as shown in Figure 2.3 (bottom-right panel). Late alarms are considered neither TPs nor FPs, i.e., the events corresponding to late alarms are not counted (NC) in the computation of the event prediction metrics. The calculation of true negatives (TNs) is of limited interest [123],

since we are dealing with an unbalanced dataset (only a few hypoglycemic events in 10 days of monitoring).

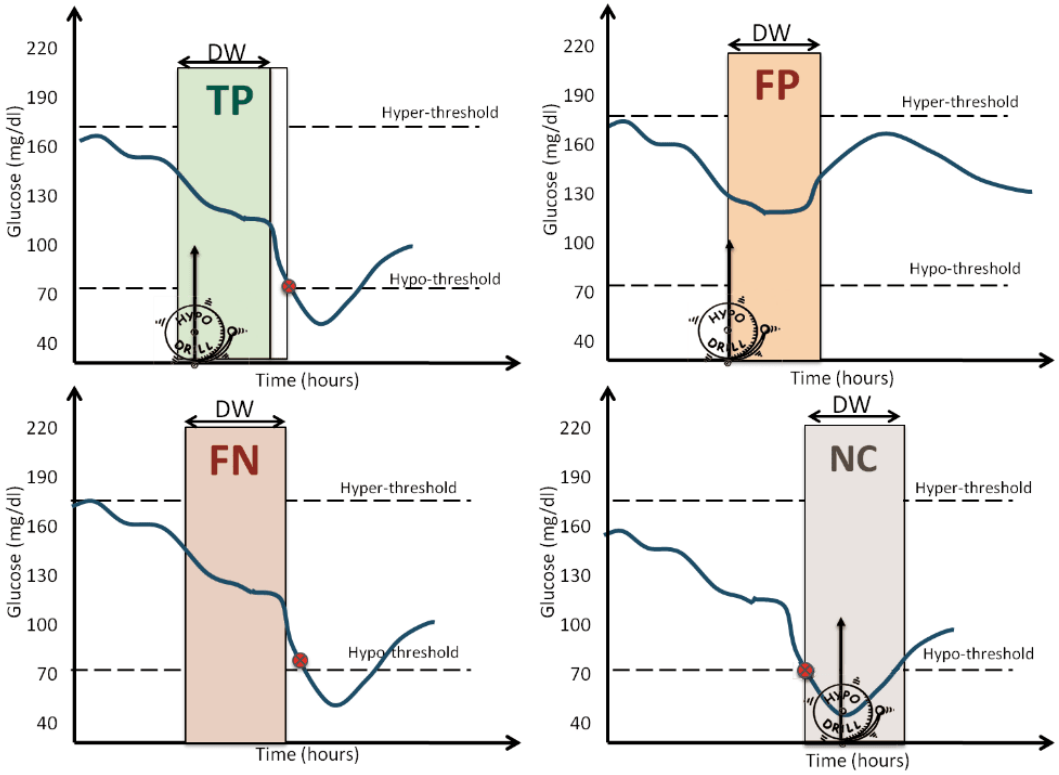


Figure 2.3: Examples of true positive (**top-left** corner), false positive (**top-right** corner), false negative (**bottom-left** corner), and not countable (**bottom-right** corner).

Once the TPs, FPs, and FNs are found, the following metrics are used to evaluate the different models:

$$precision = \frac{TP}{TP + FP} \tag{2.4}$$

$$recall = \frac{TP}{TP + FN} \tag{2.5}$$

$$F1 - score = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{2.6}$$

The precision (2.4) is the fraction of the correct alarms over the total number of alarms generated. The recall (2.5), also called the sensitivity, is the fraction of correctly detected events over the total number of events. The F1-score (2.6) is the harmonic mean of the two previous metrics. Since the dataset is strongly unbalanced, we also evaluated the daily number of FPs generated by the algorithm (FPs per day). We also evaluated the time gain (TG) of the hypoglycemic

alert as the time between the alert and the real hypoglycemic event.

Unlike the glucose prediction metrics, for which a different metric value is calculated for each subject, the values of the hypoglycemia prediction metrics are obtained by considering all the hypoglycemic events of the different subjects, as they belong to a unique time series.

2.6 The dataset and its partitioning

The data are kindly provided by Dexcom (Dexcom Inc., San Diego, CA, USA) and taken from the pivotal study of one of their last commercial sensor (Dexcom G6 CGM sensor), described at ClinicalTrials.gov (NCT02880267). This was a multicenter study, involving 11 centers. Each center obtained approval from the local IRB/ethical committee, as reported in the main publication associated with the study [124]. The original dataset included 177 CGM traces collected in 141 T1D adults (aged 18+) by the Dexcom G6 sensor (36 subjects wore two sensors in parallel). For the purposes of this work, we selected 124 CGM traces, keeping only one CGM datastream for each subject and discarding subjects who wore the CGM devices for less than 10 consecutive days. The sampling time was 5 min. In summary, the dataset grant us 1240 days of CGM data, ~350000 samples and more than 19200 CGM samples below 70 mg/dL (i.e., 5.4% of the total samples), with ~1600 hypoglycemic episodes. It should be noted that, even though hypoglycemia is rather rare in the real data, the large dataset adopted and the consequent abundant number of hypoglycemic episodes allows a solid assessment of the algorithm's ability to predict a hypoglycemic episode. Moreover, the number of hypoglycemic episodes present in our dataset is significantly larger than those of other papers having the same aim [58, 125].

2.6.1 Training and test set

A comparison of the proposed prediction algorithms is obtained by evaluating the performance of each method on a same test set. A total of 20% of all the CGM traces (i.e., 25 CGM time series) are randomly chosen from the original dataset and are candidates as a test set for evaluating all the predictive algorithms. The remaining time series (i.e., 99 CGM traces) are used to train the population algorithms. Concerning the training of the individualized algorithms, the 25 CGM time series, the candidates as a test set, are split into

training and test sets. In a preliminary examination, we found that the dimension of the training set should be approximately 7 days for nonlinear models. However, the linear algorithms require 33 h of CGM data for the training phase only. Therefore, the test set, identical for all the algorithms, is composed of the last 3 days (out of 10 days) of the 25 CGM time series initially chosen. By doing so, the CGM data of the training and test set are completely independent.

Since during data acquisition, failures and missed data may occur, the CGM traces, in the training set only, are preprocessed as follows: first, they are realigned to a uniform temporal grid, and if there is a data gap and it is smaller than 15 min, missed values are imputed via third-order spline interpolation. If the gap is longer than 15 min, the CGM trace is split into different segments.

2.6.2 Monte Carlo simulations

Splitting the dataset as described in the foregoing subsection has the advantage of providing a test set that is the same for all the algorithms but has the issue that the test set is small (about 75 days over the total 1240), thus containing a limited number of hypoglycemic episodes (~90 over about 1600 total hypoglycemic events). Both the glucose and hypoglycemic prediction performance are randomly affected by the choice of the test set. In fact, one test set extraction might turn out to be particular advantageous for algorithm A and penalizing for algorithm B, while another can result in the opposite. This issue can be overcome by performing a Monte Carlo simulation: the procedure of randomly splitting the dataset into training and test sets is iterated several times (in this chapter we employed 100 runs). For each iteration, a new training and test set is obtained, and then, the glucose prediction analysis described in this work is performed.

2.7 Results

2.7.1 Illustration of a representative training-test partitioning example

Glucose prediction and hypoglycemic event detection performance of a representative training–test partition, chosen among the 100 Monte Carlo simulations, are shown in Table 2.2 for linear models and in Table 2.3 for nonlinear models. In particular, in Table 2.2 and Table 2.3, the glucose prediction metrics are reported as median value [interquartile range] over the 25 CGM time series used as the test set. Finally, statistical analysis of the test set of this representative training–test set extraction is performed.

Linear black-box models

The population algorithms underestimate in hyperglycemia and overestimate in hypoglycemia, as illustrated for a representative subject in Figure 2.4. In particular, the CGM data (blue line) show a hypoglycemic episode before 18:00, an elevated blood glucose peak (210 mg/dL) at 22:00, and another hypoglycemic event before 00:00.
In these three situations, the population ARMA(4,1) model (green dash-dotted line), for example, provides glucose prediction values quite distant from the target CGM data. In fact, the RMSE achieved by the population ARMA and ARIMA are, respectively, about 23.75 and 23.78 mg/dL. The early detection of hypoglycemic episodes is unsatisfactory even for the population ARIMA algorithm, the best performing among the population approaches: both the precision and recall are low, respectively, at around 63% and 48%. The median TG is only 5 min.

Looking at the results in Table 2.2, we can note that the individualized models outperform the population ones: the RMSE provided by the population AR and by the individualized AR are, respectively, around 23.63 and 22.76 mg/dL. The detection of hypoglycemic events is also increased with the AR individualized models. Indeed, the recall and precision are around 40% and 58%, respectively, with the individualized models and around 48% and 46%, respectively, with the population models. The median TG improved from 5 min with the population AR to 10 min with the individualized AR. In particular, individualized ARIMA models allow mitigating the impact of slow changes in glucose mean concentrations. Thus, the corresponding predicted profiles turn out to

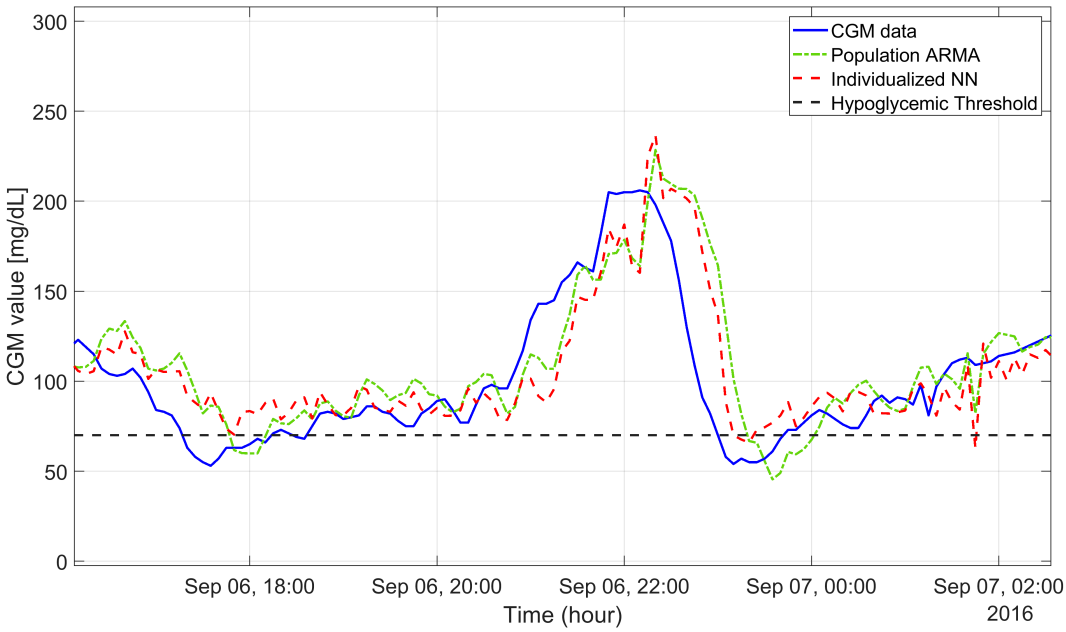


Figure 2.4: CGM data (blue line), 30-min-ahead prediction obtained with population ARMA(4,1) (green dash-dotted line) and individualized neural network (red dashed line), hypoglycemic threshold (black dashed line).

be more adherent to the target signal, as visible in the representative subject of Figure 2.5 (individualized ARIMA(2,1,1), whose prediction is reported by the red dash-dotted line, provided accurate predictions when the CGM data fall below the hypoglycemic threshold, i.e., from 8:00 to 10:00). These features make individualized ARIMA the best-performing linear algorithm both for glucose value prediction, granting a median RMSE of 22.15 mg/dL, and for hypoglycemic event prediction, with a recall of 82% and precision of 64%. One might expect that the model derived by minimizing the 30-min-ahead prediction error would achieve better performance than the model obtained following the standard PEM pipeline, i.e., by minimizing the 5-min prediction error and then deriving the predictor.
However, this is not the case, and it can be seen that the 30-min AR model provides similar performance (RMSE: 22.79 mg/dL, COD: 83.89%, recall: 21%, and precision: 42%) to the individual models identified with the standard PEM approach (RMSE: 22.76 mg/dL, COD: 84.53%, recall: 40%, and precision: 58%). This is in line with the theory in [109, 110].

The day-and-night-specific algorithms provide higher RMSE (24.22, 24.37, and 23.1 mg/dL for AR, ARMA, and ARIMA, respectively) than the algorithms described previously. The hypoglycemic detection is comparable to that of individualized models. The extra complexity of the day-and-night-

specific models does not appear to be justified by better performance. The regularized models perform very similarly to the individualized models (RMSE: 23.23 mg/dL, while the recall and precision are, respectively, 50% and 60%) but require a more complicated identification procedure. Finally, concerning AR-RLS(1), it allows to rapidly track the changes in glucose trends (Figure 2.5, black dash-dotted line), but it can be very sensitive to noisy CGM readings, and the resulting RMSE is higher than those for the other algorithms investigated (27.43 mg/dL). This feature is also reflected in an increased number of false alarms generated (about one/day). However, both the recall and precision are high: 86% and 55%, respectively. The median TG is 15 min. In summary, the best linear model was given by individualized ARIMA. Finally, statistically significant differences between the RMSE results obtained with the population algorithm and the results obtained by the individualized algorithm are indicated in Table 2.2 by asterisks.

Nonlinear black-box models

Considering the population models, the best ML method for the detection of hypoglycemic events is a SVR fed by 50 min of CGM data with a Gaussian kernel, which present TG = 10 min, recall = 69%, precision = 63%, and one false alarm every 2 days. Despite the good results in terms of event detection, it should be noted that the RMSE is around 22.85 mg/dL. The RegRF achieves the highest RMSE among the population nonlinear models considered: 23.42 mg/dL. Furthermore, by visual inspection, we observed that the predicted profiles obtained by RegRF suffer from large delays, especially when the target signal shows an upward trend. Moreover, RegRF tends to overestimate in hypoglycemia, generating a recall around 20% and a precision of 36% only.

The minimum RMSE is achieved by an NN fed by 50 min of CGM data, composed of two hidden layers, each of them with 10 neurons, similar to what is described in [65]. Despite the RMSE is the lowest among the nonlinear population methods (21.81 mg/dL), all the hypoglycemic detection metrics are not satisfactory: the recall is 27%, the precision is 39%, and the TG is 5 min. The LSTM-predicted profile (the green dash-dotted line in Figure 2.5) is similar to the one obtained by a NN: it exhibited a RMSE around 23 mg/dL, recall around 26%, and precision around 46%. Generally, the individualization of the model hyperparameters allows reducing the RMSE, e.g., the individual-

ized SVR and NN with individual hyperparameters achieves median RMSE of 22.16 and 21.52 mg/dL, respectively. In addition, the result obtained by the individualized NN outperforms all the 30 algorithms tested in this work. However, the slight improvement in terms of the prediction of glucose values does not imply an important improvement in hypoglycemic event prediction. In fact, the best individualized ML model for hypoglycemia forecasting is the individualized SVR, whose performance is similar to that of the population SVR model: the recall is about 59% vs. 63%, the precision is 72% vs. 69%, and the median TG is 10 min in both cases (individualized vs. population, respectively). The individual NN provides a predicted profile that tends to underestimate in hyperglycemia and overestimate in hypoglycemia as shown in Figure 2.4 (the prediction of the NN with individual hyperparameters, the red dashed line, is more adherent to the target when the CGM is inside the range 80–120 mg/dL). Individualized RegRF provides the worst performance in terms of both glucose and hypoglycemic event prediction: the RMSE is 26.16 mg/dL, the recall is 39%, and the precision is 60%. The individualized day-and-night-specific ML algorithms provides, in general, RMSE higher (around 30 mg/dL) than those of the algorithms described previously. The ability to detect hypoglycemic events is lower than that of the individualized ML models.

It is interesting to note that all these nonlinear methods do not provide satisfactory results in terms of hypoglycemia detection. It is worth noting that no statistically significant differences between the RMSE results obtained with the individualized nonlinear algorithms with individual hyperparameters (SVR and fNN) and the individual linear ones with individual orders (AR, ARMA, and ARIMA) can be observed.

2.7.2 Monte Carlo analysis

The results for the glucose prediction and hypoglycemic event detection performance of the 100 Monte Carlo simulations are shown in Table 2.4. For each metric, we report the mean and standard deviation of all the simulations. It is worth noting that the numerical results described in the foregoing subsection are confirmed by this further analysis. Finally, the statistical analysis performed for the Monte Carlo iterations shows that no significant differences between the RMSE results obtained with the best-performing nonlinear and the best-performing linear algorithms can be observed.

Table 2.2: Performance of linear algorithms on a representative dataset partitioning (30-min PH). The asterisks indicate p-values<0.05

Model Type		Model Class	Glucose Prediction Metric			Hypo Event detection				
			delay (min)	RMSE [mg/dl]	COD(%)	F1(%)	P(%)	R(%)	FP/day	TG(min)
POPULATION		AR*	25 [23.75-25]	23.63 [20.91-32.24]	80.89 [72.39-86.59]	47	46	48	0.41	5 [5-10]
		ARMA*	25 [20-25]	23.75 [20.75-32.15]	81.23 [72.47-86.65]	46	45	47	0.31	5 [5-10]
		ARIMA*	25 [20-25]	23.78 [20.75-32.15]	81.21 [72.44-86.62]	55	63	48	0.33	5 [5-10]
INDIVIDUAL	population order	AR	20 [20-25]	22.73 [19.02-30.36]	84.63 [80.98-87.9]	55	63	48	0.47	10 [5-15]
		ARMA	20 [20-25]	22.83 [19.31-30.91]	84.64 [77.01-88.36]	51	50	52	0.85	10 [5-15]
		ARIMA	25 [20-25]	23.12 [20.22-28.65]	83.36 [78.68-87.99]	67	64	71	0.67	10 [10-15]
	individual order	AR	25 [20-25]	22.76 [18.76-29.47]	84.53 [80.79-88.1]	48	58	40	0.47	10 [5-10]
		ARMA*	25 [23.75-25]	22.55 [20.16-30.46]	83.71 [76.99-87.91]	36	48	29	0.51	10 [5-15]
		ARIMA*	25 [25-25]	22.15 [19.8-28.87]	84.64 [78.71-87.59]	72	64	82	0.76	10 [5-15]
	individual order 30-min specific	AR	25 [20-25]	22.79 [19.75-28.84]	83.89 [76.7-88.36]	28	42	21	0.43	5 [5-15]
		ARMA	25 [25-30]	22.89 [20.54-29.81]	83.37 [75.8-87.93]	24	39	17	0.44	5 [5-15]
		ARIMA*	25 [25-25]	22.39 [19.97-29.31]	84.47 [76.28-88.23]	64	56	75	0.57	10 [5-10]
	individual order day&night	AR	25 [25-25]	24.22 [20.74-30.16]	80.72 [76.37-84.87]	26	41	20	0.29	5 [5-15]
		ARMA	25 [25-26.25]	24.37 [21.31-30.25]	77.31 [75.49-84.72]	24	39	17	0.29	10 [5-15]
		ARIMA	25 [25-26.25]	23.1 [20.47-29.76]	82.2 [76.95-86.74]	67	70	64	0.44	10 [5-15]
	regularized	AR	20 [20-25]	23.23 [19.85-31.01]	82.52 [77.22-87.74]	54	60	50	0.55	10 [5-20]
	RLS	AR	30 [25-30]	27.43 [24.63-33.88]	75.66 [67.77-81.16]	68	55	86	0.88	15 [10-25]

Table 2.3: Performance of nonlinear algorithms of a representative training-test partitioning (30-min PH)

Model Type		Model Class	Glucose Prediction Metric			Hypo Event detection				
			delay (min)	RMSE [mg/dl]	COD(%)	F1(%)	P(%)	R(%)	FP/day	TG(min)
INDIVIDUAL	POPULATION	SVR	25 [25-25]	22.85 [18.81-28.61]	85.14 [79.35-88.15]	65	63	69	0.53	10 [5-15]
		RegRF	30 [30-30]	23.42 [21.29-30.86]	80.65 [72.83-84.91]	25	36	20	0.3	5 [5-10]
		NN	20 [20-25]	21.81 [18.65-27.86]	86.19 [81.1-89.41]	31	39	27	0.36	5 [5-11.25]
		LSTM	25 [20-25]	23.1 [20.26-28.75]	82.31 [77.54-87.33]	33	46	26	0.3	5 [5-10]
	population hyperparameters	SVR	25 [25-25]	21.97 [19.68-28.98]	84.22 [78.78-87.39]	64	72	59	0.31	10 [5-15]
		RegRF	30 [30-30]	23.81 [21.35-30.47]	72.73 [67.85-79.93]	25	33	21	0.03	5 [5-5]
		NN	20 [20-25]	21.76 [18.89-28.97]	83.98 [79.37-88.7]	47	59	40	0.45	10 [5-18.75]
	individual hyperparameters	SVR	20 [20-25]	22.16 [20.62-28.79]	81.97 [65.89-87.45]	54	57	52	0.62	10 [10-20]
		RegRF	25 [25-25]	26.16 [22.49-33.97]	77.14 [69.79-82.47]	47	60	39	0.42	12.5 [5-20]
		NN	20 [20-25]	21.52 [19.12-28.29]	85.37 [78.78-88.11]	47	57	40	0.47	10 [5-18.75]
	individual hyperparameters day&night	SVR	25 [20-25]	30.13 [25.17-40.9]	67.75 [57-76.34]	48	61	40	0.41	10 [5-20]
		RegRF	25 [25-25]	33.34 [26.84-37.71]	68.47 [62.71-74.49]	39	53	31	0.43	10 [10-20]
		NN	20 [20-25]	24.4 [20.88-29.89]	82.11 [74.84-86.19]	33	53	24	0.34	10 [5-17.5]

Table 2.4: Performance of linear and nonlinear algorithms on 100 Monte Carlo iterations (30-min PH)

Model Type			Model Class	Glucose Prediction Metric		Hypo Event detection				
				delay (min)	RMSE [mg/dl]	COD(%)	F1(%)	P(%)	R(%)	FP/day	TG(min)
POPULATION			AR	25(0)	23.86(2.44)	79.8(3.18)	46.97(6.04)	54.33(6.8)	41.64(6.45)	0.48(0.12)	8.32(2.21)
			ARMA	25(0)	23.75(2.43)	79.86(3.17)	47.17(5.81)	54.77(6.8)	41.69(6.16)	0.47(0.12)	8.09(2.25)
			ARIMA	25(0)	23.96(2.42)	80.06(3.16)	50.27(5.18)	58.24(6.32)	44.51(5.7)	0.44(0.13)	9.18(1.8)
INDIVIDUAL											
population order			AR	21.45(2.29)	22.79(1.57)	84.83(1.82)	44.12(7.03)	51.99(6.63)	38.58(7.57)	0.48(0.1)	9.59(1.42)
			ARMA	21.55(2.33)	22.89(1.59)	84.05(1.84)	41.47(6.31)	48.38(6.03)	36.66(7.22)	0.53(0.15)	9.64(1.89)
			ARIMA	24.73(1.15)	22.74(1.8)	83.74(1.64)	62.83(5.6)	56.23(6.12)	71.67(6.8)	0.77(0.19)	11.73(2.4)
individual order			AR	24.45(1.57)	22.78(1.67)	84.56(1.86)	49.73(7.45)	58.77(7)	43.11(7.78)	0,48(0.11)	9.64(1.31)
			ARMA	25(0)	22.83(1.57)	83.79(1.67)	32.5(7.45)	44.25(7.33)	25.98(7.15)	0.44(0.1)	9.95(2.28)
			ARIMA	25(0)	22.13(1.58)	84.36(1.77)	70.5(3.69)	61.04(4.33)	83.64(3.89)	0.73(0.13)	10.18(0.94)
individual order	30-min		AR	25(0)	22.97(2.37)	83.4(3.17)	28.96(12.78)	42.07(11.36)	23.05(12.16)	0.39(0.12)	8.64(4.93)
	specific		ARMA	25(0)	23.04(2.22)	82.7(3.3)	24.25(11.17)	37.49(11.5)	18.85(10.4)	0.4(0.13)	9.55(4.77)
			ARIMA	25(0)	22.45(1.29)	84.26(1.81)	66.63(5.43)	60.54(6.63)	74.08(6.94)	0.58(0.17)	10(0)
individual order	day&night		AR	25(0)	24.15(1.53)	78.87(1.83)	27.12(4.38)	37.31(7.97)	21.31(3.14)	0.32(0.07)	10(4.11)
			ARMA	25(0)	24.44(1.59)	78.75(2.18)	25.57(4.96)	36.95(10.63)	19.55(3.39)	0.28(0.08)	9(4.04)
			ARIMA	25(0)	22.93(1.31)	83.68(1.56)	66.37(5.09)	68.36(5.64)	64.98(7.33)	0.41(0.13)	9.86(0.75)
regularized			AR	21.82(2.43)	22.87(1.63)	83.1(2.11)	43.53(6.27)	48.5(5.59)	39.72(7.2)	0.57(0.1)	11.36(2.54)
RLS			AR	29.82(0.94)	27.67(1.6)	76.12(2.11)	63.89(4.46)	51.43(5)	84.32(5.16)	1.01(0.15)	16.36(2.4)
POPULATION			SVR	24.45(2.99)	22.72(2.75)	81.69(8.39)	50.81(13.21)	47.59(11.73)	44.15(12.83)	0.56(0.33)	9.79(3.03)
			RegRF	25.09(0.67)	23.35(1.77)	80.91(1.89)	19.57(11.74)	23.99(16.56)	12.24(11.07)	0.43(0.18)	8.47(4.22)
			NN	21.36(2.25)	21.74(1.45)	85.93(1.7)	26.15(10.79)	37.6(11.79)	20.58(9.98)	0.43(0.14)	6.91(3.57)
			LSTM	24.55(1.45)	22.97(1.99)	83.25(2.28)	20.52(13.13)	40.6(15.03)	15.03(12.46)	0.28(0.15)	8.32(6.16)
INDIVIDUAL											
population			SVR	24.27(3.52)	22.6(4.62)	82.89(11.77)	53.82(12.75)	59.91(11.02)	49.54(12.67)	0.47(0.35)	11.15(3.26)
hyperparameters			RegRF	25.55(1.57)	23.38(2.02)	78.23(2.06)	31.37(9.65)	47.11(8.59)	24.42(9.5)	0.36(0.12)	11.91(3.04)
			fNN	20.18(0.94)	21.78(1.78)	84.78(1.49)	38.58(7.56)	47.95(7.54)	32.89(8.47)	0.48(0.14)	10.59(2.15)
individual			SVR	23.64(2.25)	22.21(2.09)	81.32(2.36)	53.63(7.86)	58.21(7.16)	49.71(9.69)	0.55(0.13)	12.36(2.82)
hyperparameters			RegRF	25(0)	26.06(2.02)	77(2.31)	40.34(6.68)	50.81(7.04)	33.93(7.24)	0.44(0.11)	15.36(3.38)
			fNN	20.09(0.67)	21.63(1.69)	85.1(1.45)	37.54(7.59)	47.02(8)	31.77(8.14)	0.49(0.13)	10.14(2.12)
individual	day&night		SVR	24(2.02)	29.22(2.33)	71.35(4.33)	45.85(6.85)	53.92(7.98)	39.89(7.04)	0.52(0.12)	12.55(3.25)
			RegRF	25(0)	29.72(2.06)	69.69(3.2)	34.49(6.11)	45.47(6.39)	28.28(6.7)	0.45(0.1)	14.41(3.91)
hyperparameters			NN	20.73(1.78)	23.54(1.91)	82.11(1.92)	32.25(6.88)	48.69(7.4)	24.5(6.56)	0.35(0.1)	11.18(3.22)

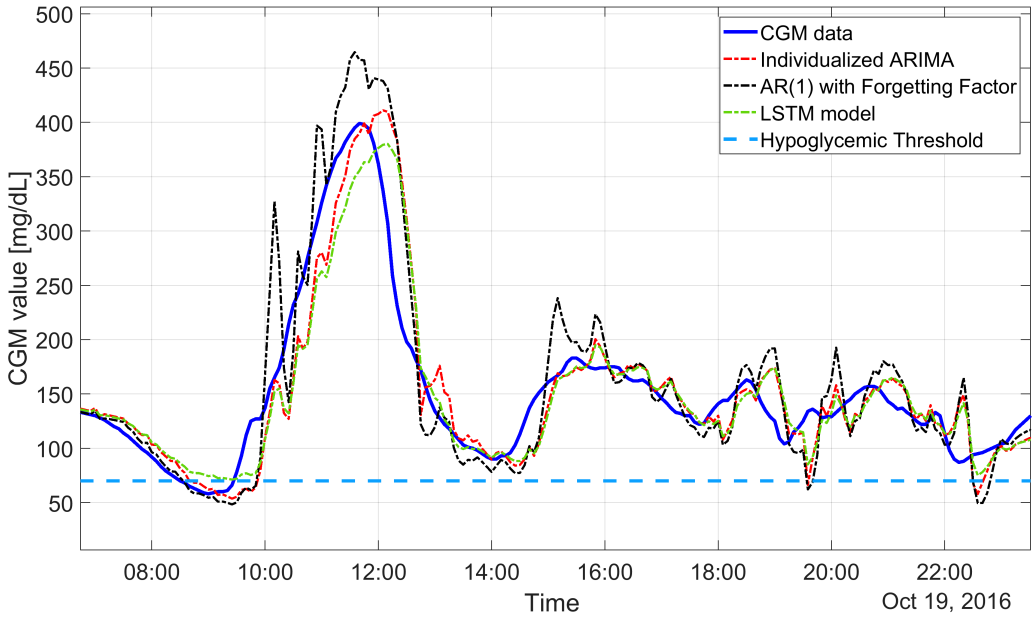


Figure 2.5: CGM data (blue line) and 30-min-ahead prediction obtained by AR-RLS(1) (black dash-dotted line), individualized ARIMA(2,1,1) (red dash-dotted line), and LSTM model (green dash-dotted line). Hypoglycemic threshold (light blue dashed line).

2.7.3 Exploratory analysis for different PH

All the algorithms described in this work focus on short-term prediction (i.e., 30 min), which enables patients to take proactive/corrective measures to mitigate or to avoid critical events. As a further exploratory analysis, we evaluate the prediction performance of the best linear and nonlinear algorithms for several PHs. As shown in Figure 2.6, the prediction error considerably increases for long-term prediction for both the linear and nonlinear algorithms. This is expected: the larger the temporal distance, the larger the number of factors that can influence blood glucose concentration. This result further strengthens our motivation to limit the head-to-head comparison of glucose predictive algorithms fed by CGM data to only a 30 min prediction horizon.

2.8 Summary of the main findings

Among the 30 glucose predictive algorithms tested in this head-to-head comparison, the linear algorithm granting the best future glucose prediction is the individualized ARIMA (median RMSE of 22.15 mg/dL). The best nonlinear algorithm is individualized NN (median RMSE of 21.52 mg/dL). While the

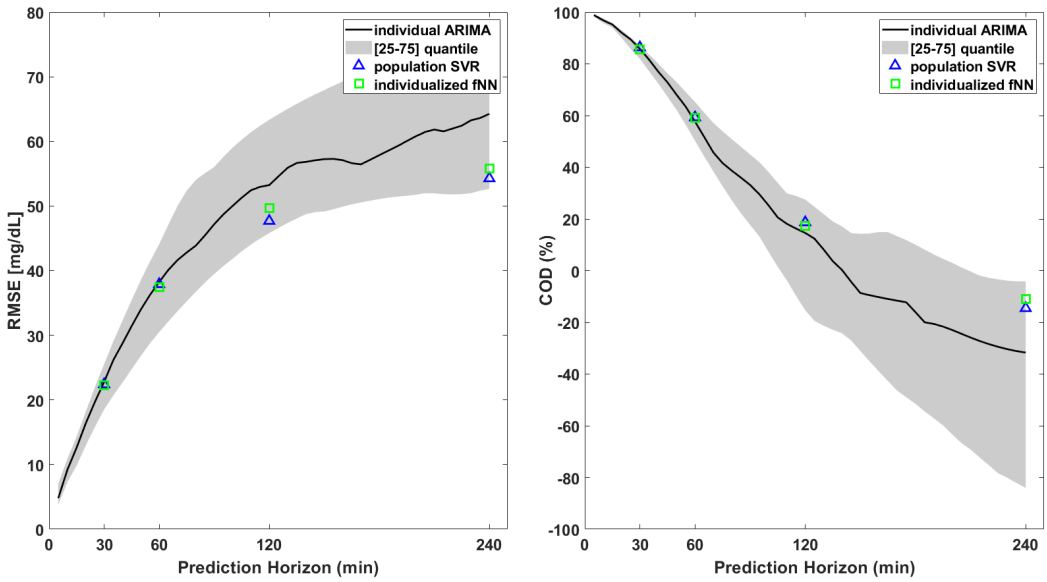


Figure 2.6: RMSE (**left**) and COD (**right**) for the 3 best-performing algorithms out of the 30 tested in this work. The black lines are the median RMSE and COD (left and right, respectively) obtained using individual ARIMA with different prediction horizons. Blue triangles and green squares indicate the same metrics for PH = 30, 60, 120, and 240 min for population SVR and individualized NN, respectively.

median RMSE of the individualized NN is slightly smaller than the median RMSE obtained using an individualized ARIMA, the difference among the two is not found to be statistically significant. When hypoglycemic event detection is considered, individualized ARIMA achieves the best F1-score (72%), outperforming SVR (F1-score = 65%), the best nonlinear method based on this metric. All the algorithms exhibit TG (i.e., the temporal distances between the hypoglycemic events and the predictive alarms) that span from 5 up to 15 min, with the best results for individualized ARIMA and SVR. The generation of preventive hypoglycemic alerts 5–15 min before the event could be clinically relevant. In fact, in the best-case scenario in which a preventive hypotreatment is ingested 15 min before the hypoglycemic episode, the rescue CHO will likely reach the blood before the hypoglycemic event, preventing or drastically mitigating it. Even a 5-min anticipation, while probably insufficient to prevent hypoglycemia, would still contribute to reducing both its duration and its amplitude. The practical benefit of taking preventive actions before hypoglycemia with TG similar to those reported here has been shown in [104].

Two main findings are worth being highlighted. First, the individualized methods slightly outperform their population counterparts, confirming the positive impact of model parameter individualization, which allows customiz-

ing models for each single patient and dealing with the large variability in glucose profiles among individuals with diabetes. Second, the use of advanced nonlinear techniques, substantially more complex than their linear counterparts, does not majorly benefit the prediction performance. Clearly, this last finding does not exclude that other nonlinear ML or DL techniques can change the picture (an exhaustive exploration of nonlinear techniques is practically impossible, also considering the number of new contributions constantly proposed in these fields), but proves that linear methods are still highly valuable options that offer a good trade-off between complexity and performance. It is worth noting that both the numerical and statistical findings of this analysis seem to be in line with most of the literature [48, 53, 58, 65, 70, 93, 98, 115]. Nonetheless, we report a clear contrast with the findings in some other contributions [69, 126]. As a final observation, Figure 2.3 and Figure 2.4 show that all the predictive algorithms exhibit a large prediction error when, after a meal, glucose is rapidly increasing. This suggests that the use of CGM-only algorithms has an intrinsic drawback: these models are not able to describe changing in dynamics due to external factors.

Chapter 3

Incorporating meal timing information in predictive algorithms

[1] As discussed in Chapter 2, one of the main limitations of all the predictive algorithms (linear and nonlinear) employing only CGM data as input is that any metabolic disturbance, e.g. a meal, would deteriorate the accuracy of the predicted BG levels by increasing the delay between the target and the predicted profile. Therefore, the use of additional sources of information should be considered to improve the accuracy of prediction algorithms. However, recording accurate exogenous information has nonegligible drawbacks. For example, getting accurate meal information can be burdensome and prone to errors in long-term management: indeed, the individual with T1D has to manually input (using an electronic diary or a dedicated mobile application) a correct estimate of the amount of CHO ingested for each meal.
To overcome this issue, this chapter assesses a novel BG forecasting methodology that combines CGM data and meal timing. Unlike conventional approaches that require both timing and dosing of CHO and insulin, the proposed method has the practical advantage of requiring minimal input data: CGM readings, which are automatically recorded by the CGM sensor, and mealtime, that can easily forwarded to the prediction algorithm by e.g., smart insulin pens or CSII. For such a scope, the dataset exploited for the first analysis (Chapter 2) cannot be used here, as it contains only CGM traces and no meal information. Consequently, the validation of the methodology has been

[1] This chapter contains material published in *Prendin et al., Sensors, 2022*, [127]

performed on two datasets acquired under free-living conditions monitored either in open-loop or closed-loop control, in which the information on meal timing has been accurately recorded.

3.1 Stochastic seasonal local models for glucose prediction

3.1.1 Chapter contribution

Many time series, especially in the financial and economic fields, exhibit seasonal behavior, i.e., regular patterns of changes and fluctuations that periodically repeat [128]. However, seasonality is not a priori characteristic of glucose data, but can be artificially induced by developing appropriate strategies that link the postprandial response to periodic meal consumption. In this context, the standard regression models evaluated in Chapter 2 are not flexible enough to accurately capture these patterns in the data, and new models have to be introduced.

To this end, this chapter assesses a novel methodology for glucose forecasting based on the combined use of seasonal stochastic local models and fuzzy C-means clustering. In particular, seasonal models are introduced for the first time in [129], and the combined use of seasonal models along with clustering techniques was presented in [130] and [131]. In these works, the methodology was developed and validated only in well controlled datasets: the first [130] was recorded during in-hospital clinical trials, while the second [131] was obtained by exploiting the educational version of the Uva/Padova simulator [132]. In both cases, the results were encouraging since the proposed approach based on seasonal models and clustering outperformed all the state-of-art techniques for BG prediction. However, a real-time assessment on data recorded under free-living conditions has not been performed yet. In fact, dealing with real data poses some problems about the completeness and reliability of stored information that can degrade the ability of algorithms to accurately forecast BG levels [51, 62]. Moreover, glucose dynamics recorded in free-living conditions can be much more complex to describe than those obtained by simulations or others recorded during in-hospital trial sessions, since in the first case the patient is exposed to substantially larger disturbances to glucose homeostasis. In this chapter we fill this gap by assessing the clustering and seasonal local mod-

eling methodology for glucose prediction proposed in [130] [131] on two real datasets of different size (11 and 13 subjects monitored for 8 weeks and about 5 months, respectively) and obtained with different insulin dosing strategies (manual open-loop and closed-loop control).

3.1.2 Chapter outline

This chapter describes the new datasets exploited for this work (Section 3.2), the main steps of the proposed methodology (Section 3.3). Also, we considered several approaches (Section 3.4): an individualized ARIMA model and a NN based on CGM data only; an individualized ARIMA with exogenous inputs (ARIMAX) model and a variant of NN, namely NN-X, fed by CGM, insulin and CHO information (timing and amount). Of note, in the previous chapter we have shown that ARIMA and NN are the best performing linear algorithm for blood glucose forecasting using CGM data only. As extensively studied by our research team, see [62], ARIMAX is one of the most suitable options when additional information, such as insulin and CHO information, are available. It is worth noting that, both ARIMA and ARIMAX models allow achieving accurate prediction performance even if compared to other nonlinear and more complex algorithms [62, 68, 95]. Predictive performance (see Section 3.5) are evaluated for different prediction horizons (PH) and the results on both dataset are consistent: for PH > 45 minutes, the proposed approach based on clustering and seasonal local models outperforms individualized ARIMA models and NN for PH > 60 minutes. Remarkably, there is no statistically significant difference when compared to individualized ARIMAX with the practical advantage of the minimal input information needed (i.e. meal timing). Furthermore, we perform a preliminary investigation about the hypoglycemic predictive capabilities of these algorithms (see 3.5.4).

3.2 The new datasets

The first dataset used in this study is the Ohio Type 1 Diabetes Mellitus dataset [1], from now on referred as the OhioT1DM. The OhioT1DM dataset was updated on the 2020 release and it comprises 12 subjects with T1D monitored for 8 weeks. The subjects wore a Medtronic Enlite CGM device (sampling time is 5 minutes) along with an insulin pump (Medtronic 530G or 630G) and a wearable system (Basis Peak fitness or Empatica Embrace) to measure phys-

iological variables, for instance: skin temperature, skin conduction and heart rate. Moreover, the dataset provides subjects self-reported information about meals: timing, amount and type (i.e. breakfast, lunch, dinner, snack, hypoglycemia treatment). Since self-reported mealtime is a crucial information for the real-time validation purposes of this work, subject ID 567 which did not record any meal during the last 10 days of monitoring was discarded.

Subj ID	**Missing values (%)**	**CV(%)**	**TIR(%)**	**TAR(%)**	**TBR(%)**
540	8	41	72	22	6
544	15	36	70	29	1
552	23	37	80	18	3
559	11	42	61	36	4
563	7	33	73	25	2
570	5	33	43	56	2
575	7	42	70	23	7
584	8	35	53	46	1
588	3	30	63	37	1
591	12	37	68	28	4
596	18	34	78	20	2
Mean(SD)	11 (6)	36.4 (4)	66.4 (11)	31 (12)	3 (2)

Table 3.1: Background information for OhioT1DM dataset. Numerical values are rounded to the nearest integer.

Each subject comprising the OhioT1DM dataset was split into training set (about the 82% of the entire monitoring period) consisting of the initial 6 weeks of monitoring, and into a test set (about the 18%) composed by the last 10 days. The second dataset was collected in a multicenter clinical trial (NCT02137512) aimed at assessing the long-term use of a hybrid closed loop insulin delivery system developed at the University of Virginia [94]. From now on, it will be referred as CTR3 dataset. The CTR3 dataset comprises 14 individuals with T1D monitored for about 4-5 months using the Dexcom G4 sensor, which sampling time is 5 minutes. Basal insulin was automatically recorded by the insulin pump (Roche Accu-Check Spirit Combo). Meal amount and timing were manually inputted in the system for all the meals. Based on this information the system computed a suitable bolus of insulin. The data of each subject is split in a test set (about the 10% of the dataset), consisting of the last 10 monitoring days while the remaining part is used as training set (about the remaining

90%). In this dataset an individual was discarded since more than the 50% of the CGM trace was composed by missing values.

Table 3.1 and Table 3.2 report, for the OhioT1DM and CTR3 dataset respectively, the percentage of missing values, the percentage of time spent in hypoglycemia (TBR), in target (TIR), in hyperglycemia (TAR), and the glycemic variability index [133] computed as $CV = 100 \cdot \frac{\sigma}{\mu}$ where, CV is the coefficient of variation, σ is the standard deviation and μ is the mean of glucose levels.

Original dataset	Missing value (%)	CV(%)	TIR(%)	TAR(%)	TBR(%)
1	4	29	80	19	1
2	23	32	79	20	1
3	3	30	80	18	2
4	8	39	75	22	3
5	18	35	78	20	2
6	21	31	84	15	1
7	25	32	70	30	1
8	12	31	83	15	2
9	35	36	83	16	1
10	25	38	70	27	3
11	15	31	85	13	2
12	22	37	72	26	2
13	19	33	80	19	1
Mean(SD)	17.6 (9)	33.3 (3.4)	78.4 (5.1)	20 (5)	1.6 (0.8)

Table 3.2: Background information for CTR3 dataset. Numerical values are rounded to the nearest integer.

Both datasets are acquired in free-living conditions, and they show a real-life scenario characterized by complex glucose dynamics, making the prediction of future glucose levels a challenging task. In the training set of both datasets, CGM gaps smaller than 30 minutes were filled using linear interpolation, while no imputation was performed on the test set. Looking at Table 3.1 and Table 3.2, a main difference among the two analyzed datasets can be found in the mean TIR: 66.4% vs 78.4%, and in the mean TAR: 31% vs 20% for the OhioT1DM and the CTR3, respectively. This was partially expected since the CTR3 dataset is a closed-loop dataset, however the mean CV, which is used to quantify the glycemic variability, is quite similar: 36.4% vs 33%. In the following sections, the main steps of the proposed approach are described. Of note, C-SARIMA, as described in [131], is designed to be tailored to individu-

als. Consequently, the following steps are computed for each individual of the dataset.

3.3 The fuzzy clustering and local modeling methodology

3.3.1 Time series segmentation

The first step of the methodology requires to partition CGM time series into a set of periods. To do so, exploiting the mealtime information, the post-prandial period (PP) is defined as the CGM measurements:

- from mealtime up to 4 hours after meal intake, or
- from mealtime up to the following meal intake (if this happens before 4 hours).

PPs containing more than one hour and a half (18 CGM samples) of missing glucose concentrations are discarded.

Partitioning CGM time series in such a way, leads to PPs having different lengths. To deal with this issue, PPs smaller than 4 hours of monitoring data, are expanded with blank values, i.e. NaN (Not-a-Number) values, to reach the maximum length. As a result, each CGM time series in segments had the same length. This is crucial for enforcing the seasonality and applying the methodology. After the NaN-padding step, a large number of PPs show blank values in the final positions and that should be adequately treated as missing data in the following steps.

3.3.2 Time series clustering

This step aims to group PPs which show a similar glycemic pattern. Following previous works, the Partial Distance Strategy Fuzzy C-Means Clustering (PDSFCM) was applied, since it can handle missing data, thus proving adequate for dealing with NaN-padded PPs and with uncomplete data acquisitions. This clustering method is a modified version of Fuzzy C-Means (FCM) [134] which allows each PP to be included into several clusters with different

degrees of membership. In particular, w_{ij} denotes the degrees of membership to the $i-th$ cluster of the $j-th$ PP. The degree of membership is a number in range [0,1] and the sum of the degrees of membership of each PP is 1:

$$0 \leq w_{ij} \leq 1 \;\; and \;\; \sum_{i=1}^{nC} w_{ij} = 1 \forall j \tag{3.1}$$

PDSFCM finds the degree of membership for each PP in the clusters [134] by minimizing the following objective function:

$$\sum_{i=1}^{nC} \sum_{j=1}^{N} w_{ij}^{m} d^2 \left(x_j, v_i\right) \tag{3.2}$$

where $x_1, x_2, \ldots, x_N$ denotes the vector of the PPs glucose profiles; N is the total number of PPs; nC is the number of clusters ($nC > 1$); m is the fuzzy exponent, i.e. a real number greater than 1; $v_1, v_2, \ldots, v_{nC}$ are the cluster centroids defined as:

$$v_i = \frac{\sum_{j=1}^{N} w_{ij}^{m} x_j}{\sum_{j=1}^{N} w_{ij}^{m}}, \quad 1 \leq i \leq nC \tag{3.3}$$

From now on, the center of the cluster (or cluster centroid) will be referred as cluster prototype.

Finally, $d(x_j, v_i)$ is the partial distance (i.e., a modified version of the Euclidean distance for dealing with missing values [135]) between any PP, (x_j) and the cluster prototype i, (v_i).

Given a set of centroids, w_{ij} is computed using the following equation:

$$w_{ij} = \frac{1}{\sum_{k=1}^{nC} \left(\frac{d^2(x_j, v_i)}{d^2(x_j, v_k)} \right)^{\frac{1}{m-1}}}, \quad 1 \leq i \leq nC, \quad 1 \leq j \leq N \tag{3.4}$$

To compute the w_{ij} minimizing (3.2), the centroid definition (3.3) and the membership equation (3.4) are iteratively updated until no further improvement in the cost function is achieved [134].

Finding the right number of clusters is a critical task: a small number may result in clusters that are not completely separated, on the contrary a large number may deteriorate the compactness of one or more clusters. For such a scope, many validation criteria have been proposed [134]. In this work, the optimal number of clusters nC as well as the fuzzy exponent m have been automatically chosen by minimizing the Fukuyama-Sugeno index [134, 136] on

the training set using an exhaustive grid search approach (ranges for $nC = \{2, \ldots, 30\}$ and for $m = \{1, \ldots, 3\}$). Such an index measures both the compactness and the separation between each cluster and the prototypes.

3.3.3 Model identification

Once the clustering step has been performed, several sets of "similar" glycemic profiles, having the same length, have been obtained. Then, for each cluster, PPs are concatenated to obtain an artificial glucose time series which shows an artificially induced seasonal pattern associated to the periodic meal consumption. By doing so, the seasonality, which is not originally present in raw CGM time series, is now enforced.

Capturing the dynamics and the seasonality of the artificial concatenated time series, can be done by identifying a Seasonal Autoregressive Integrated Moving-average (SARIMA) model for each cluster. A SARIMA model is a generalization of an Autoregressive Integrated Moving-average (ARIMA) model which is able to take into account for the seasonality.

In fact, an ARIMA model can be described as follows:

$$y(t) = \alpha + \omega(t) \tag{3.5}$$

$$\phi_p\left(z^{-1}\right) \nabla^d \omega(t) = \theta_q\left(z^{-1}\right) \epsilon(t) \tag{3.6}$$

where, $y(t)$ is the CGM value at time t, α is the intercept, $\omega(t)$ is the disturbance series, ∇ is the backward differencing operator such that $\nabla\omega(t) = \omega(t) - \omega(t-1)$ and d is the order of the differencing step. $\epsilon(t)$ is a white noise process driving the model, and $\phi_p(z^{-1})$ and $\theta_q(z^{-1})$ are the polynomials of order p and q for the autoregressive and moving-average part of the model.

Similarly, a SARIMA model can be described by adding the seasonal terms to Equation 3.6:

$$y(t) = \alpha + \omega(t) \tag{3.7}$$

$$\phi_p\left(z^{-1}\right) \Phi_P\left(z^{-S}\right) \nabla_s^D \nabla^d \omega(t) = \theta_q\left(z^{-1}\right) \Theta_Q\left(z^{-S}\right) \epsilon(t) \tag{3.8}$$

where, S indicates the seasonality, $\nabla_S \omega(t) = \omega(t) - \omega(t-S)$, and D is the order of the seasonal differencing step. $\Phi_P(z^{-S})$ and $\Theta_Q(z^{-S})$ are the polynomials of order P and Q for the seasonal autoregressive and seasonal moving-average part of the model.

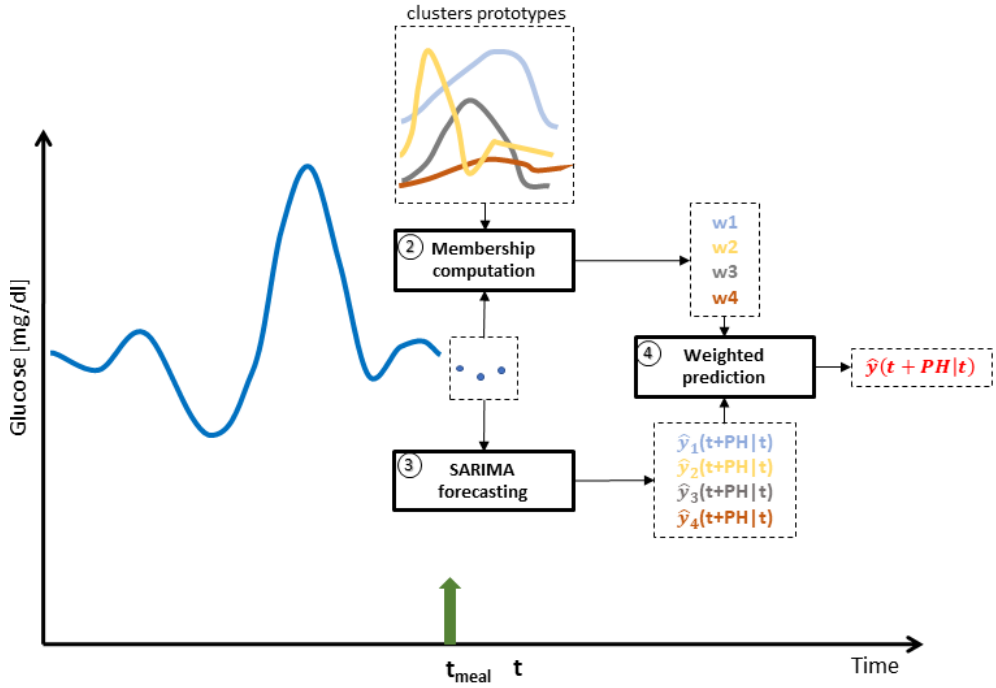


Figure 3.1: Schematic overview of the real time prediction process

The SARIMA degrees of freedom, i.e. the order of the Autoregressive (AR), Moving-average (MA), Integrated (I) seasonal and nonseasonal parts are chosen by minimizing the Bayesian Information Criterion (BIC) using an exhaustive grid search approach. In particular, the ranges for $p = \{1,\ldots,4\}$, $q = \{0,\ldots,4\}$, $d = \{0,1\}$, $P = \{1,\ldots,3\}$, $Q = \{0,\ldots,3\}$, and $D = \{0,1\}$ were considered. Following [131], the seasonality term (S) equals to 53 samples: 48 samples which are the length of the PP plus 5 CGM samples which preceded mealtime, the so-called pre-samples, introduced for a proper model initialization.

3.3.4 Real-time glucose forecasting

Finally, once SARIMA models are identified for each cluster, glucose can be predicted ahead in time by weighting the predictions of all SARIMA models. Figure 3.1 provides an overview of the forecasting process. As depicted in Figure 3.1, let suppose that:

- the optimal number of clusters found in the training set is four (hence: four prototypes and four SARIMA models are available);
- it is mealtime ($\mathbf{t_{meal}}$ in Figure 3.1, indicated by a green vertical arrow).

The real time glucose forecasting procedure is triggered at mealtime. The output is the predicted glucose level, indicated in Figure 3.1 as $\hat{y}(t+PH|t)$, and it can be computed by applying the following pipeline:

1. wait for collecting 3 CGM samples (i.e., wait for 15 minutes, if the sampling time is 5 minutes);
2. compute the membership values, i.e. the weights (w_1, w_2, w_3, w_4), between the collected CGM samples and the clusters prototypes using equation 3.4;
3. compute the glucose predictions exploiting the four identified SARIMA models (i.e., $\hat{y}_1(t+PH|t)$, $\hat{y}_2(t+PH|t)$, $\hat{y}_3(t+PH|t)$, $\hat{y}_4(t+PH|t)$);
4. compute the output $\hat{y}(t+PH|t)$ as the weighted sum of computed predictions in step 3 using weights computed in step 2;
5. repeat steps from 2 to 4 each time a new sample is recorded.

Remark. To calculate the membership values (Step 2) during the real-time prediction process, equation (3.4) was modified to compute the distances between the collected CGM samples and the corresponding segment of the cluster prototypes. In fact, each time a new CGM sample is recorded, the membership is calculated with an increased number of samples (i.e., CGM data are accumulated until the maximum length of the postprandial period is reached).

3.3.5 Computational effort

The computationally demanding parts of C-SARIMA are related to the clustering optimization procedure (i.e., determining the number of clusters and the fuzzy exponent) and to the local models identification process (i.e., SARIMA model order selection and parameters identification). However, these steps are computed only once and offline, leveraging training data. On the contrary, the online steps (described in Figure 3.1) are computationally cheap. In fact, each time a new CGM sample is recorded, the average time required to compute the PH-step ahead prediction is about 0.38 seconds. In details: 32 μsec for membership computation, 0.37 sec for SARIMA models forecasting and 9 μsec for the weighted sum. The computation time has been evaluated on an ASUS laptop equipped with an Intel(R) Core(TM) i7-8565U CPU @1.80GHz 1.99 GHz.

3.4 Benchmark glucose predictive algorithms

Based on previous chapter, the effectiveness of the proposed approach based on clustering and SARIMA modeling is assessed by comparing the predicted PPs with those obtained by an individualized ARIMA model based on CGM data only and an individualized ARIMAX model fed by CGM, insulin and CHO information. For each subject, an ARIMA and an ARIMAX model is identified. Similarly to SARIMA models, the order of AR, MA, I and exogenous (X) parts of the model are fixed for all subjects and chosen by minimizing BIC (among all the individuals) using an exhaustive grid search approach. In particular, the grid of explored order for AR $= \{1, \ldots, 20\}$, MA $= \{0, \ldots, 20\}$, I $= \{0,1\}$, X $= \{1, \ldots, 20\}$. Note that, while the model complexity is fixed, the model is individualized by estimating subject-specific models' parameters.

Finally, it could be of interest to investigate whether nonlinear models grant different performances as compared to the proposed methodology. For such a scope, two feed forward neural networks are considered as comparators. The first the network (NN) [65] employs CGM measurements up to 25 min before the current time as input information. The second network (NN-X) employs as input: CGM readings, insulin and CHO information up to 25 min before the current time. In both cases, the output is the glucose prediction PH minute ahead in time. In details, NN and NN-X are composed by two hidden layers equipped with 10 and 5 neurons (with sigmoidal transfer function) and an output layer equipped with a single neuron (with linear transfer function).

For what it concerns parameters learning (weights and bias), they are randomly initialized and updated according to a standard backpropagation training algorithm (Levenberg-Marquardt) which is applied in a batch mode: weights and biases are updated when all the inputs and targets are presented. It is worth remarking that the training process must be performed for each PH.

3.5 Predictive performance on post-prandial periods

In this section the performance of the proposed approach is presented. The accuracy of the predicted PPs is evaluated for different PH, i.e. PH $= \{30, 45, 60, 75\}$ minutes. The RMSE (2.1) between the predicted and the target CGM PP has been considered as metric for the assessment. The novel approach is

indicated as C-SARIMA in Table 3.3 and Table 3.4, with respect to the benchmark algorithms. All the algorithms are evaluated both on the OhioT1DM and CTR3 dataset.

Table 3.3, shows the results for the OhioT1DM. Statistical significance is determined using a paired t-test if normality is accepted, a Wilcoxon signed-rank test if normality is rejected. The cross (+) indicates that there is statistically significant difference (ssd) between the C-SARIMA and ARIMA. The asterisk ($*$) indicates that there is ssd between C-SARIMA and ARIMAX. The circumflex (ˆ) indicates that there is ssd between C-SARIMA and NN. The (") indicates that there is ssd between C-SARIMA and NN-X.

models	**RMSE [mg/dL]**			
	PH = 30 min	**PH = 45 min**	**PH = 60 min**	**PH = 75 min**
ARIMA	19.64 [18.42-20.54]	26.91 [23.86-28.59]	33.67 [29.82-35.11]	38.82 [32.48-41.59]
NN	20.11 [17.58-20.99]	26.41 [25.10-28.31]	32.11 [30.94-33.26]	35.18 [32.55-37.74]
C-SARIMA	20.13(*,") [18.63-21.38]	27.23(") [24.63-28.74]	31.96(+) [29.55-33.95]	33.91(+,ˆ) [31.97-37.29]
ARIMAX	18.73 [17.31-20.06]	26.46 [22.96-27.03]	30.82 [29.30-31.92]	34.73 [31.31-39.09]
NN-X	17.78 [16.79-21.04]	25.68 [24.85-27.62]	30.67 [28.98-34.93]	34.06 [32.71-35.54]

Table 3.3: Comparison of the performance of the C-SARIMA against individualized ARIMA and ARIMAX model, NN and NN-X on the OhioT1DM dataset

At the short-term prediction horizon (i.e., PH $\leq$ 45 minutes), the proposed approach achieves similar performance to the individualized ARIMA model: there is no statistically significant difference among the two techniques. In particular, the RMSE provided by the proposed methodology is slightly higher (20.13 mg/dL vs 19.64 mg/dL and 27.23 mg/dL vs 26.91 mg/dL, for PH $=$ 30, 45, respectively). However, for the long-term prediction horizon (i.e., PH $\geq$ 60 minutes), the performance of the C-SARIMA outperforms ARIMA models (RMSE$=$ 31.96 mg/dL vs 33.67 mg/dL and 38.82 mg/dL vs 33.91 mg/dL). In particular, for PH $=$ 60 and 75 minutes, the difference is found to be statistically significant (p-values $<$ 0.05).

The NN performed similarly to C-SARIMA (median RMSE of 20.11 mg/dL, 26.41 mg/dL and 32.11 mg/dL) and no statistically significant difference in the RMSE is found for PH $\leq$ 60 minutes. On the contrary, C-SARIMA outperforms

the NN for PH=75 minutes by granting RMSE = 33.91 mg/dL vs 35.18 mg/dL (p-value<0.05).

Comparing the C-SARIMA with individualized ARIMAX models, it can be found that for PH $\leq$ 45 min, the best results are obtained by individualized ARIMAX models (RMSE 18.73 mg/dL vs 20.13 mg/dL and 26.46 mg/dL vs 27.23 mg/dL). However, for PH = 60, 75 minutes the C-SARIMA models provides results that do not differ in a statistically significant manner to ARIMAX.

Finally, NN-X provides the better results with respect to C-SARIMA for PH$\leq$ 60 minutes: RMSE is 17.78 mg/dL, 25.68 mg/dL and 30.67 mg/dL, while no significant improvement is found for PH = 75 minutes (RMSE = 33.91 mg/dL vs 34.06 mg/dL).

models	**RMSE [mg/dL]**			
	PH = 30 min	**PH = 45 min**	**PH = 60 min**	**PH = 75 min**
ARIMA	21.02 [20.03-24.86]	29.42 [27.40-33.24]	35.38 [34.63-40.48]	44.01 [39.50-45.86]
NN	21.78 [19.35-24.23]	30.64 [26.88-34.11]	34.21 [29.92-38.68]	42.60 [35.97-44.42]
C-SARIMA	21.63 [20.00-25.90]	29.67(") [25.83-34.07]	33.47(+) [29.59-39.62]	40.18(+,^,") [32.92-42.42]
ARIMAX	20.83 [17.80-23.40]	28.13 [24.22-32.65]	33.57 [28.54-40.44]	39.99 [31.36-43.40]
NN-X	21.12 [17.49-23.89]	27.98 [23.52-34.63]	33.37 [27.36-34.63]	38.41 [30.38-41.71]

Table 3.4: Comparison of the performance of the C-SARIMA against individualized ARIMA and ARIMAX model, NN and NN-X on the CTR3 dataset

Table 3.4 shows the results for the CTR3 dataset. As for the OhioT1DM, for short-term PH the C-SARIMA provides similar performance to an individualized ARIMA, i.e. there is no significant improvement if compared to an individualized ARIMA models: median RMSE is 21.63 mg/dL vs 21.02 mg/dL and 29.67 mg/dL vs 29.42 mg/dL, for PH = 30 and PH = 45 minutes. However, for PH = 60 minutes and PH = 75 minutes the proposed methodology outperforms the competitor providing a statistically significant difference (median RMSE = 33.47 mg/dL vs 35.38 mg/dL and 40.18 mg/dL vs 44.01 mg/dL, respectively).

NN grants performance comparable to C-SARIMA for all the PH$\leq$ 60 (median RMSE of 21.78 mg/dL, 30.64 mg/dL, 34.21 mg/dL) and inferior prediction for PH = 75 minutes (42.60 mg/dL vs 40.18, p-value<0.05). Similarly,

when comparing SARIMA with respect to ARIMAX, one can see that for PH $\leq$ 45 min, the best results are obtained by ARIMAX model (RMSE = 20.83 mg/dL and 28.13 mg/dL). However, the two results are similar for all the PH considered: there is no statistically significant difference. Median RMSE produced by NN-X is inferior to the one of C-SARIMA for PH = 45, 60 and 75 minutes (RMSE = 27.98 mg/dL, 33.37 mg/dL and 38.41 mg/dL) but statistically significant difference is found only for PH = 45 and 75 minutes.

3.5.1 Algorithms employing the same amount of information

It could be interesting to investigate the performance of predictive models that employs same information of C-SARIMA. For such a scope, we proposed ARIMAX+mealtime and NN+mealtime: two variants of ARIMAX and NN-X fed by CGM and mealtime information only. Results are reported in Table 3.5 and Table 3.6, for OhioT1DM and CTR3 dataset, respectively. The asterisk ($*$) indicates if ssd is found between C-SARIMA and ARIMAX+mealtime, (+) indicates if ssd is found between C-SARIMA and NN+mealtime.

models	**RMSE [mg/dL]**			
	PH = 30 min	**PH = 45 min**	**PH = 60 min**	**PH = 75 min**
ARIMAX+mealtime	18.93 [17.42-20.52]	27.88 [22.90-28.93]	34.28 [28.26-35.78]	38.39 [32.47-41.68]
NN+mealtime	20.16 [18.04-21.88]	26.53 [23.06-28.28]	32.78 [30.55-33.88]	34.22 [31.36-37.81]
C-SARIMA	20.13 [18.63-21.38]	27.23 [24.63-28.74]	31.96(*,+) [29.55-33.95]	33.91(*,+) [31.97-37.29]

Table 3.5: Table Comparison of performance between C-SARIMA vs. individualized ARIMAX + mealtime and NN + mealtime model fed by CGM and meal time on OhioT1DM data set.

Numerical results are consistent between the two datasets: for PH = 30 minutes, the best results are achieved by ARIMAX+mealtime (the median improvement is about 1 mg/dL in the OhioT1DM dataset and about 0.5 mg/dL in the CTR3 dataset with respect to C-SARIMA). For PH = 45 minutes, the three methods achieved similar performance (about 27 mg/dL and 29 mg/dL, on the OhioT1DM and CTR3 dataset, respectively). As shown in Table 3.5 and Table 3.6, for PH > 45 minutes the best results are achieved by C-SARIMA

models	RMSE [mg/dL]			
	PH = 30 min	**PH = 45 min**	**PH = 60 min**	**PH = 75 min**
ARIMAX+mealtime	20.97 [17.83-24.63]	29.40 [23.36-33.36]	36.75 [29.90-41.74]	42.95 [36.35-44.44]
NN+mealtime	21.57 [18.13-24.50]	29.24 [24.37-33.55]	34.55 [29.43-38.28]	41.29 [32.79-42.47]
C-SARIMA	21.63 [20.00-25.90]	29.67 [25.83-34.07]	33.47(*,+) [29.59-39.62]	40.18(*,+) [32.92-42.42]

Table 3.6: Table Comparison of performance between C-SARIMA vs. individualized ARIMAX + mealtime and NN + mealtime model fed by CGM and meal time on CTR3 dataset.

which outperforms its comparators (the improvement is statistically significant, p-value < 0.05). In fact, as detailed in Table 3.5, compared to ARIMAX+mealtime and NN+mealtime, C-SARIMA grants RMSE $= 31.96$ mg/dL vs 34.28 mg/dL vs 32.78 mg/dL, for PH $= 60$ minutes. And RMSE $= 33.91$ mg/dL vs 38.39 mg/dL vs 34.22 mg/dL, for PH $= 75$ minutes. Similarly, as shown in Table 3.6, compared to ARIMAX+mealtime and NN+mealtime, C-SARIMA grants RMSE $= 33.47$ mg/dL vs 36.75 mg/dL vs 34.55 mg/dL, for PH $= 60$ minutes. And RMSE $= 40.18$ mg/dL vs 42.95 mg/dL vs 41.29 mg/dL, for PH $= 75$ minutes.

3.5.2 Discussion of the results

The results among the two datasets are consistent: the proposed methodology based on clustering and SARIMA models has comparable or superior performance with respect to one of the most performing linear algorithms based on CGM data only, i.e. individualized ARIMA model. In particular, C-SARIMA outperforms ARIMA for PH $= 60$ and 75 minutes. Furthermore, results show that C-SARIMA is able to provide similar performance or slightly superior to a state-of-the-art nonlinear method for glucose prediction (NN). In particular, such a difference is found to be statistically significant for PH $= 75$ minutes.

The second linear comparator is an individualized ARIMAX model, which is expected to enhance prediction performance due to the use of additional information carried by insulin and CHO. In this comparison, the proposed approach provides performance that are not significantly different from ARIMAX for PH $= 45$, 60 and 75 minutes. This is remarkable since the SARIMA and clustering-based approach use less information, CGM and mealtime only,

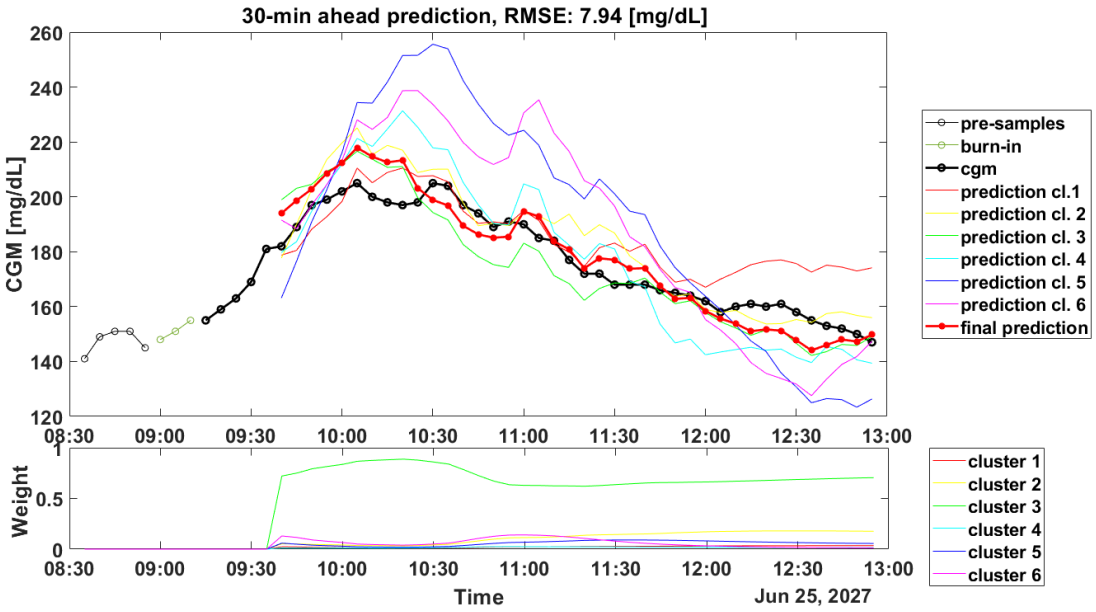


Figure 3.2: Illustration of an accurate forecasting of BG levels, PH = 30 minutes. OhioT1DM dataset.

while ARIMAX also requires information about the CHO ingested and on the amount of insulin administered, which represents a non-negligible drawback, since the estimation of the correct amount of CHO and insulin is critical for subjects with T1D [137]. In the OhioT1DM dataset a similar finding seems to hold also for the nonlinear comparator with inputs (NN-X). In the CTR3 dataset, no significant difference is found for PH= 60 minutes, whereas a significant (albeit hardly practically relevant) improvement is achieved by NN-X with respect to C-SARIMA for PH = 45 and 75 minutes. However, it is worth noting that on the OhioT1DM dataset such an improvement is usually larger for short-term PH, but it becomes minor for long-term predictions. When dealing with real data acquired in free-living conditions, the glucose response after meal intakes exhibits a wide range of variability. This variability forced the clustering step to use an increased number of clusters if compared to the results obtained on simulated datasets [131]. In fact, after the cluster optimization procedure the mean number of clusters per subject was 16 while in [131] it was about 10. Being the first step of the pipeline, a successful clustering of the PPs is crucial for the success of the entire proposed methodology. In fact, if it provides several sets of "similar" glycemic response, the resulting artificial seasonal CGM time series will show regular patterns periodically repeated. If this condition is satisfied, this leads to a better identification of SARIMA models and to an increased prediction accuracy.

Another critical aspect linked to the clustering step is about the computation of the weights during the real-time glucose forecasting. Such computation is crucial for obtaining accurate predicted profiles: in Figure 3.2 and Figure 3.3 prediction results for a representative subject of the OhioT1DM dataset are shown, (ID:544) and it can be seen how weights computation can lead to good and poor accuracy in the prediction of the PPs.

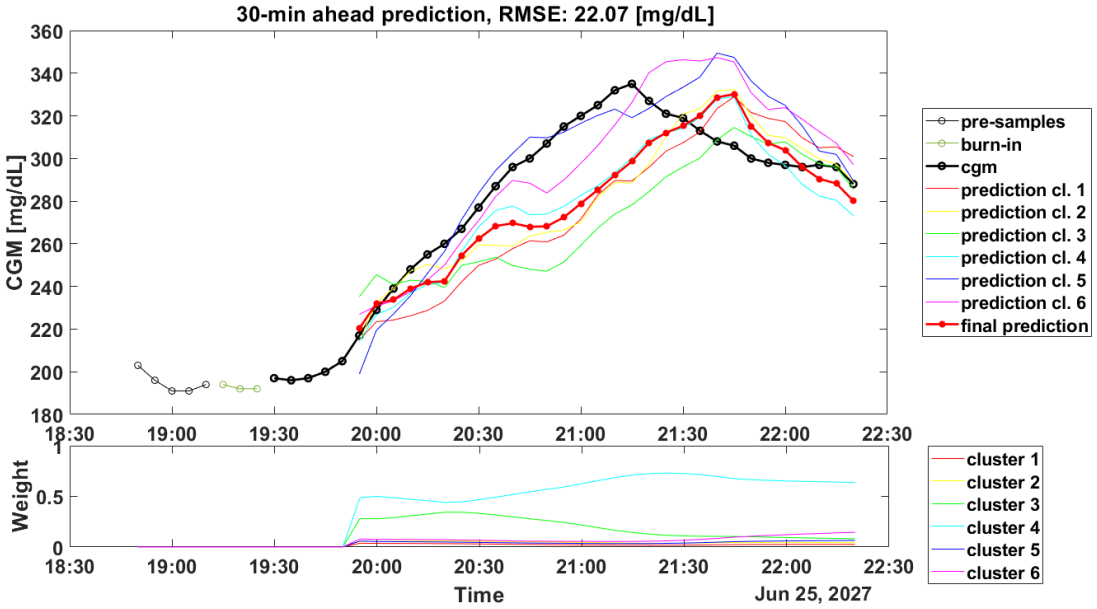


Figure 3.3: Illustration of real-time forecasting of BG, PH = 30 minutes. OhioT1DM dataset.

Figure 3.2 shows on the top panel the PP trace (black line) and the final prediction (red bold line). For a better visualization 6 out of 12 predicted profiles (colored lines) were discarded since their weights (visible in the bottom panel) were almost equal to zero. In addition, in the top panel are reported the 5 CGM samples (black thin line) before the meal (in this case there is a breakfast at 8.55) and the 3 CGM samples (indicated as burn-in in the legend) after the meal intake which are used to compute the initial weights as described in the schematic overview of the forecasting process in Figure 3.1. In Figure 3.2, the computed weights grant an accurate final prediction, since they assign the CGM data points to the most similar cluster, in this case cluster 3. On the contrary in Figure 3.3, which shows the CGM periods after dinner, the weights computation leads to an incorrect assignment. Looking at the predicted profiles, it seems that the most accurate predicted profile is the one obtained with the SARIMA model identified on cluster 5 or on cluster 6 (blue and violet line, respectively). However, the highest weight is related to cluster 4 which accu-

rately forecasts the initial samples (from 19.55 to 20.05) but then it is not able to follow the target signal. Likely, the incorrect computation of weights could be due to the fact that the prototypes in the training set are not completely able to describe the current PP, thus suggesting that a larger training set is required. Unfortunately, as shown in Table 3.4, similar results can be found even if a larger dataset, like the CTR3, is considered.

3.5.3 Comparison with previous works

Although the comparison with literature contributions is not straightforward due to the fact that in this chapter only PPs (and not the whole CGM traces) are considered, the numerical results seem to be in line with the results reported in [62, 79, 80, 98, 108]. Furthermore, the proposed methodology provides a performance similar to that obtained by more complex deep learning methodologies exploiting additional information, as described in [70]. Moreover, comparing the main findings with respect to previous works about this methodology shows quite different results in terms of performance metrics. In [130], the forecasting accuracy of the proposed methodology is measured by computing the RMSE for several PH. In [130], the proposed methodology grants a RMSE $=$ 9.99 mg/dL, 15.70 mg/dL and 19.29 mg/dL for PH $=$ 30, 45 and 60 minutes. However, the authors focused on evaluating how successfully the predicted trajectory fits actual CGM data, which is different from evaluating the accuracy of the predicted glucose levels at a certain PH ahead in time as described in [131], and in this work. Another limitation of [130] is related to the dataset. In [130] data are acquired during a clinical trial that comprises 18 closed loop experiments of 60 hours based on scheduled meal intakes and exercise sessions. Due to the limited dataset, the reported results are related only to the validation set. In the last work, [131], the RMSE is computed as described in Equation 2.1, making a fair comparison between this work and [131] possible. In particular, the RMSE achieved by predicting postprandial periods is approximately 15 mg/dL and 25 mg/dL for PH $=$ 30 and 60 minutes, respectively. In this work, as shown in Table 3.3 and Table 3.4, the RMSE for PH $=$ 30 and PH $=$ 60 is about 21 mg/dL and 32 mg/dL. The main difference among these results can be found in the dataset: in [131] authors exploited simulated datasets. These in-silico simulations have been performed by exploiting a modified set-up of the educational version of the UVA/Padova simulator [132]. In simulated datasets, glucose responses are quite similar and well de-

fined: after meal intake, BG increases and it returns to the euglycemic range within 2.5 hours after meal.

3.5.4 Exploratory analysis on hypoglycemia prediction performance

As described in the previous chapter, focusing on event prediction is a slightly different task than BG levels forecasting. However, connecting these two different problems is possible by designing a methodology that can be used in cascade to the BG prediction algorithm in order to convert predicted BG values into hypoglycemic alerts. For such a scope, we investigated hypoglycemia prediction performance of the algorithms tested in this chapter by using a standard (and simple) hypo-alert approach, based on a threshold crossing approach, that is, the algorithm raises a hypoglycemic alert if the predicted BG level is below the hypoglycemia threshold.

	Precision (%)	Recall (%)	F1-score (%)	Time Gain (min)
C-SARIMA	52	67	59	10 [5-25]
ARIMA	55	69	61	10 [6.25-20]
ARIMAX	59	81	68	15 [10-25]
NN	44	78	56	10 [5-15]
NN-X	56	69	62	15[5-20]

Table 3.7: post-prandial hypoglycemia performance for OhioT1DM dataset (24 hypoglycemic episodes), PH = 30 minutes

	Precision (%)	Recall (%)	F1-score (%)	Time Gain (min)
C-SARIMA	45	65	54	12.5 [5-27.5]
ARIMA	41	73	53	10 [5-20]
ARIMAX	47	85	61	15 [5-25]
NN	30	68	42	10 [5-15]
NN-X	32	64	42	5 [5-25]

Table 3.8: post-prandial hypoglycemia performance for CTR3 dataset (37 hypoglycemic episodes), PH = 30 minutes

Following the framework proposed in Chapter 2, and in [48, 95], we evaluated precision, recall and F-score as well as the time gain for a PH = 30 minutes. In both datasets, the best results are achieved by individualized ARIMAX (F1-score = 68% and F1-score=61%, for OhioT1DM and CTR3 respectively) while C-SARIMA grants F1-score=59% and F1-score=54% for OhioT1DM and CTR3, respectively. Considering the overall performance detailed in Table 3.7 and Table 3.8, it seems that an effective prediction of hypoglycemia poses a difficult challenge to C-SARIMA as well as to any other competitor methods (ARIMA or ARIMAX, but also the nonlinear prediction algorithms).

3.6 Summary of the main findings

Summarizing, this chapter has shown:

- the combined use of CGM data and mealtime events to develop an artificially induced seasonal pattern associated with periodic meal consumption;
- the assessment of a novel methodology (C-SARIMA) based on fuzzy clustering and seasonal stochastic local models for the forecasting of BG based on CGM and mealtime information;
- the performance of linear (ARIMAX) and nonlinear (NN-X) BG forecasting techniques that employ as input: CGM, CHO intakes, and insulin injections (time and dosing information).

In previous proof-of-concept works, C-SARIMA was shown to outperform other literature methodologies, especially if long-term PH are considered. However, the assessment of the methodology was limited to well-controlled or simulated datasets and a more robust validation on real and challenging dataset acquired in free-living condition was needed. For such a scope, the validation of the methodology has been done by exploiting two datasets to take into account a different size of the datasets (i.e., the number of available monitoring weeks/months) and insulin administration regiments (manual control versus hybrid closed loop). The results found on both datasets are consistent each other: the proposed C-SARIMA methodology outperforms individualized ARIMA model for PH>45 minutes and NN for PH>60 minutes. Remarkably, there is no statistically significant difference between the results provided by C-SARIMA and the ones provided by individualized ARIMAX model fed

by the CGM, CHO, and insulin information.
It is also interesting to note that individual ARIMAX models represent a viable alternative to the more complex nonlinear methodologies tested in this chapter, as also demonstrated in [62, 68]. In addition, it has been pointed out that the prediction of hypoglycemia is a critical task for all the algorithms presented in this chapter and requires the development of novel dedicated approaches.

Chapter 4

Designing a predictive algorithm to forecast hypoglycemic events

[1] In Chapter 3, we focused on assessing a methodology based on clustering and stochastic seasonal local models, which employs minimal input information. For longer PH, C-SARIMA outperforms predictive algorithms fed by CGM data only, but it is not able to provide a statistical significant improvement with respect to individual ARIMAX or NN-X models fed by CGM, CHO and insulin. These results suggest that the use of both timing and dosing information for CHO and insulin is required to achieve better accuracy in predicting BG levels. Also, we showed that an effective prediction of hypoglycemia poses a difficult challenge to C-SARIMA as well as to any other comparator method. For this reason, the purpose of this chapter is to improve real-time prediction of impending hypoglycemic events using individualized ARIMAX models that have been shown to be effective in BG prediction [62] and able to capture the essential dynamics of glucose-insulin [56, 58, 139, 140, 141, 142]. More specifically, this chapter aims to show that the conventional approach for real-time hypoglycemia forecasting can be improved by leveraging: a glucose-specific cost function (named *gMSE*) in model parameters identification, and a "prediction funnel", that is, confidence intervals (CI) for multiple PH, within the hypo-alarm raising strategy. To this end, we employed a subset of the CTR3 dataset (11 T1D individuals, 2 monitoring weeks) characterized by reliable meal and insulin dosing information which was already used in a previously published work [62].

[1] This chapter contains material published in *Faccioli S.*, Prendin F.*, et al., Journal of Diabetes Science and Technology, 2022*, [138].

4.1 Regression-based approaches for hypoglycemia prediction in T1D

4.1.1 Chapter contribution

As pointed out in a recent report on AI applications for diabetes management [41], the combined use of CGM devices, insulin pumps and dedicated mobile applications [42] brought the possibility of recording different type of information, for instance, CGM data, insulin, meal, physical activity, and self-reported life events. This enables the development of advanced AI-enabled DSS, which are composite tools that implement multiple software modules to support the patient in the decision-making process. One of the key elements that can be embedded in an advanced DSS is the prediction module. In fact, knowing ahead in time, e.g., with 20 minutes of time anticipation, if blood glucose (BG) is getting close to possibly harmful values allows patients to take proactive actions to mitigate or avoid critical episodes like hypoglycemia (i.e., BG below 70 mg/dL), considerably improving T1D management [19, 20, 27, 143, 144, 145].

A large number of literature studies focused on the challenge of forecasting hypoglycemic episodes in the short-term [146]. More specifically, hypoglycemia prediction was addressed by either classification-based or regression-based approaches. Classification-based approaches consist in developing a binary classifier [147], i.e., an algorithm producing only two types of possible output, "impending hypoglycemia" or "no hypoglycemia predicted". On the contrary, regression-based approaches are two-steps procedures that (as a first step) predict the future glucose concentration after a certain prediction horizon (usually chosen in the range 30-60 min), and then (as second step) raise an alarm if the predicted value falls below a suitable threshold (usually, but not necessarily, 70 mg/dL). Predicted glucose levels in the first step can be obtained by using either linear [56, 58, 139, 140] or non-linear methodologies [48, 114, 125, 126, 148, 149, 150]. The purpose of this chapter is to improve the real-time forecasting of impending hypoglycemic events in T1D when CGM data, injected insulin and meal intake information are available. For such a purpose, this chapter exploits individualized ARIMAX models which, as already mentioned, were shown to be effective in BG prediction [62] and capable of capturing the essential dynamics of glucose-insulin regulation [56, 58, 139, 140, 141, 142]. Moreover, they present two important characteristics: a) the model parameters individualization (a key aspect to deal with

the large inter- and intra-subject variability of glucose-insulin dynamics) has been deeply studied and powerful convergence results as well as statistical properties analysis are available in the literature [109]; b) BG prediction can be computed using well-established and computationally convenient algorithms, such as those tracing back to Kalman filtering [151].

Specifically, this chapter demonstrates that there are two margins of improvement within conventional approaches employed in the literature for the real-time forecasting of hypoglycemic events. The first one is about the model identification procedure. In particular, instead of using the classical Mean Square Error cost function, we consider a glucose specific cost function, named glucose Mean Square Error (*gMSE*) proposed in [152]. This cost function, that applies an extra penalty to overestimation of low BG and to underestimation of high BG, enables the identification of more effective subject-tailored models for predicting hypoglycemic episodes. The second margin of improvement regards the alarm-raising strategy. Instead of focusing on a single PH (as conventionally done in the literature), we consider the "prediction funnel", i.e., confidence intervals for simultaneously multiple PH, in order to take into account the expected decrease of accuracy in the prediction as the PH increases.

4.1.2 Chapter outline

In Section 4.2 we described the dataset reduction and the preprocessing steps required to deal with data of different nature, then we introduced the conventional regression-based approach to forecasting hypoglycemic events (Section 4.3). Finally, we detailed the new approach that focuses on: the use of the glucose specific cost function for models' parameters identification and the use of the prediction-funnel for the hypoglycemic alarm strategy (Section 4.4). Also, we designed two different hyperparameters tuning strategies to further improve the forecasting capabilities of the algorithm. Finally, a systematic evaluation of the contribution given by the proposed novelties is assessed by measuring precision, recall, F1-score, false-positive per day and time gain (Section 4.5). Sections 4.6 and 4.7 present the main results and preliminary conclusions of the analysis conducted so far.

4.2 Dataset and preprocessing steps

As for the previous chapter, data were taken from a previously published paper [94], to which we refer the reader for any protocol detail. Briefly, a group of 14 T1D individuals participated in the 5-month test of the 24/7 use of a hybrid closed-loop insulin delivery system. Among the collected data, those of interest for the purpose of this work are: CGM readings, recorded using a Dexcom G4 sensor (Dexcom, Inc., San Diego, CA, USA) with a sampling period time of 5 min; time-course of the insulin infusion delivered by the Roche Accu-Check Spirit Combo device (Roche Diabetes Care, Inc., Indianapolis, IN, USA); amounts of carbohydrates ingested at meals (CHO), as estimated and manually annotated by the participants.

Dealing with experimental data of different nature (CGM, insulin information and CHO intake) poses some technical issues related to: a) device synchronization; b) completeness of stored data. These problems are faced as follows:

- all signals were aligned to the same time grid uniformly sampled at $T_S = 5$ min;

- if one or more signals presented gaps longer than 30 minutes, the entire data portion was discarded, as data off-line inference is deemed unreliable.

Then, for each patient, two consecutive portions of 7 days without discarded data were selected. The two portions contained only a few hours of missing CGM values due to sensor replacement. The so selected 14-day long dataset was then split in training-set (first 7 days) and test-set (last 7 days). Finally, the remaining short data gaps were reconstructed. The reconstruction strategy depends on whether the gap occurs in the training or in the test set. Since the training-set is entirely available during the model training phase, non-causal techniques can be used to reconstruct missing training data. In this case, we adopted a third-order spline interpolation. On the contrary, on the test-set glucose prediction has to be performed in real-time, so missing data (about 1% on average per subject) was filled by exploiting the last few available CGM samples. In particular, a first order polynomial is fitted by exploiting the last 15 minutes of recorded CGM data and then, such a model is used to infer missing data. Notably, 3 out of 14 subjects lacked a sufficiently long portion of consecutive data.

In summary, the analysis is conducted on data of 11 subjects. For each subject 7 days of training data and 7 days of test are available. In this population, 39 hypoglycemic episodes were observed in the training set and 42 episodes occurred in the test set. This amount to an average of 1 hypoglycemic episode in every 2 days per patient.

4.3 Conventional regression-based approach to the prediction of impending hypoglycemic events

In this chapter we considered, as baseline for the assessment, a regression-based hypoglycemia prediction algorithm which consists of two steps: 1) identification of a model suited to predict future glucose levels in real-time using CGM, insulin and CHO information; 2) threshold-crossing alarm-raising strategy.

For step 1, based on previous findings [62, 95], we employ an ARIMAX model that can be described by the following equation:

$$A(q)\,y(t) = B_{ins}(q)\,ins(t - nk_{ins}) + B_{cho}(q)\,cho(t - nk_{cho}) + \frac{C(q)}{(1-q)^{-1}}e(t) \tag{4.1}$$

where $y(t)$ is the current CGM reading, $ins(t)$ and $cho(t)$ are the model inputs: insulin and carbohydrates intake, respectively and $e(t)$ is the noise term assumed to be identically, independently distributed (i.i.d). Finally, q is the backward shift operator such that $q^{-1}u(t) = u(t-1)$.

$$A(q) = 1 + a_1 q^{-1} + \ldots + a_{n_a} q^{-n_a} \tag{4.2}$$

$$B_{ins}(q) = b_{1ins} + b_{2ins} q^{-1} + \ldots + b_{n_{b\_ins}} q^{-n_{b\_ins}+1} \tag{4.3}$$

$$B_{cho}(q) = b_{1cho} + b_{2cho} q^{-1} + \ldots + b_{n_{b\_cho}} q^{-n_{b\_cho}+1} \tag{4.4}$$

$$C(q) = 1 + c_1 q^{-1} + \ldots + c_{n_c} q^{-n_c} \tag{4.5}$$

The order of the autoregressive part (i.e. the number of past CGM readings required by the model) is defined by n_a, n_{b_ins} and n_{b_cho} indicates the order of the inputs (insulin and cho, respectively). n_c is the order of the moving average part of the model. The inputs' delays are indicated by nk_{ins} and nk_{cho}.

n_a, n_{b_ins}, n_{b_cho} and n_c represent the degrees of freedom of the model and they are selected by using the bayesian information criterion (BIC).

To estimate the unknown model parameters

$\theta = [a_1, \ldots, a_{n_a} | \, b_{1ins}, \ldots, b_{n_{b_{ins}}} | b_{1cho}, \ldots, b_{n_{b_{cho}}} | c_1, \ldots, c_{n_c}]$, the classic prediction error method (PEM) approach of minimizing the standard mean square error (MSE) cost function:

$$\hat{\theta} = argmin_\theta MSE\left(g(k), \hat{g}(k|k-1,\theta)\right), \tag{4.6}$$

is considered, with

$$MSE\left(g(k), \hat{g}(k|k-1,\theta)\right) = \frac{1}{N}\sum_1^N \left(g(k) - \hat{g}(k|k-1,\theta)\right)^2 \tag{4.7}$$

where N is the number of available data samples, $g(k)$ is the CGM-measured glucose at time k, and $\hat{g}(k|k-1,\theta)$ is the 1-step ahead prediction, i.e., the glucose value at k predicted at time $k-1$ by the model (and thus dependent also on the model parameters θ). Once the model parameters are identified, the model can be used to predict the future glucose level ahead in time at a certain PH, a quantity denoted as $\hat{g}(k+PH|k,\theta)$.

For step 2, a hypoglycemic alarm is raised if $\hat{g}(k+PH|k,\theta)$ is below the hypoglycemic threshold of 70 mg/dL.

Note that using this method, from now on called single-PH approach, at each step k only one prediction is considered. Hence, PH is a key tuning parameter of this strategy. A typical PH value is 30 min. A schematic representation of the conventional approach is reported in Figure 4.1 (upper panel).

4.4 Proposed novel approach to the prediction of impending hypoglycemic events

As previously described, an innovation is introduced for each of the two steps of the conventional approach described in the previous section. In step 1, the ARIMAX model is identified, at the individual subject level, using a glucose-specific cost function which takes into account the clinical impact of the prediction error. In step 2 the alarm strategy is based on a prediction–funnel approach, which exploits the possibility of handling the ARIMAX model within a Kalman filtering framework. A schematic representation of the proposed approach is reported in Figure 4.1 (bottom panel). Details of the innovations are described below.

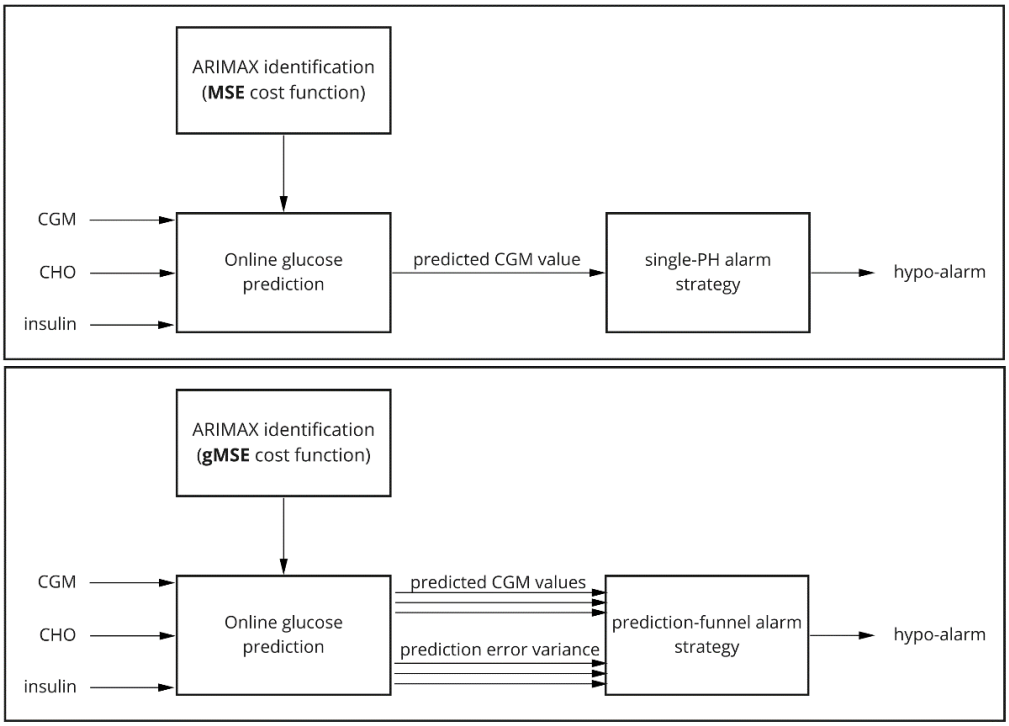


Figure 4.1: Schematic representation of the conventional approach (upper panel: MSE+single-PH alarm strategy) and the proposed approach (bottom panel: gMSE+prediction-funnel alarm strategy).

4.4.1 Glucose specific model identification

In step 1, the parameter estimation strategy relies on the minimization of an ad-hoc cost function, called glucose-specific mean square error (*gMSE*), [152]. This metric, inspired by the well-known Clarke error grid (CEG) [153], modifies MSE to account for the clinical impact of the prediction error. This is done by increasing MSE values up to 250% in case of glucose over-estimation during hypoglycemia and up to 200% in case of glucose under-estimation in hyperglycemia. By doing so, over-estimation in hypoglycemia is penalized more than under-estimation in the same region. In fact, the first situation is clinically more dangerous: it could prevent the detection of an hypoglycemic episode or induce an optimistic evaluation of its severity, leading to inadequate treatment. A symmetric reasoning holds for the case of hyperglycemia but, since hypoglycemia is deemed more dangerous than hyperglycemia, in the first case MSE is increased more (up to 250%) than in the second case (only up to 200%). The MSE is increased so that *gMSE* retains two fundamental mathematical properties of the original metric, smoothness and convexity, as they simplify the optimization involved in the parameter estimation. The estimated model pa-

rameters $\hat{\theta}$ were obtained as follows:

$$\hat{\theta} = argmin_{\theta} gMSE\left(g(k), \hat{g}(k|k-1,\theta)\right). \tag{4.8}$$

By doing so, the identified model provides more accurate and clinically relevant prediction in the hyper- and hypoglycemic range than in the normal range, thus permitting more effective hypoglycemia prediction.

4.4.2 Derivation of the Kalman predictor

Once the model is identified (step 1), the PH-step ahead prediction can be derived from that model for any value of PH. This is done by applying a standard Kalman predictor framework [109, 151].

Let us define a discrete state-space model in innovation form as in [109]:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + Ke(k) \\ g(k) &= Cx(k) + e(k). \end{aligned} \tag{4.9}$$

$x(k)$ is the $n \times 1$ state vector; $u(k)$ is the $p \times 1$ vector of input (in our case $p = 2$, $u(k) = [i(k), m(k)]^T$, where $i(k)$ is the insulin-related time series, while $m(k)$ accounts for CHO assumptions); $g(k)$ is the CGM concentration; $e(k)$ is the innovation noise (white zero-mean noise with variance σ_e^2). Its value can be estimated from the data using the 1-step ahead prediction residual; A is the $n \times n$ state matrix, B is the $n \times p$ input matrix, K the $n \times 1$ noise-state vector, and C is the $1 \times n$ output vector. A, B, C, and K, as well as σ_e^2, are obtained by first identifying an ARIMAX model on the patient data and transforming the obtained model in state-space form [109].

The Kalman predictor [151] associated to the identified model is:

$$\begin{aligned} \hat{x}(k+1|k) &= A\hat{x}(k|k-1) + Bu(k) + K(g(k) - \hat{g}(k|k-1)) \\ \hat{g}(k|k-1) &= C\hat{x}(k|k-1), \end{aligned} \tag{4.10}$$

where $\hat{x}(k+1|k)$ and $\hat{g}(k|k-1)$ are the 1-step ahead predicted states and glucose, respectively.

It can be shown that the PH-step ahead prediction can be obtained as [151]:

$$\begin{aligned} \hat{g}(k+PH|k) &= CA^{PH-1}\hat{x}(k+1|k) + \\ &+ C\textstyle\sum_{i=1}^{PH-1} A^{PH-1-i}Bu(k+i) \end{aligned} \tag{4.11}$$

starting from $\hat{x}(k+1|k)$, i.e., the prediction provided by the Kalman predictor.

The Kalman predictor just described also provides an estimate of the variance of the state prediction error, $\Sigma(k|k-1)$, and of the variance of the glucose prediction error:

$$\sigma^2(k|k-1) = C\Sigma(k|k-1)C^T + \sigma_e^2. \tag{4.12}$$

If we consider a PH-step ahead prediction (equation), it is possible to compute the variance of the state prediction error [109, 151] , $\Sigma(k+PH|k)$, and thus the variance of the glucose prediction:

$$\sigma^2(k+PH|k) = C\Sigma(k+PH|k)C^T + \sigma_e^2. \tag{4.13}$$

Please note that the model identification phase estimates both the deterministic and the stochastic model describing the system. No further tuning of the Kalman predictor is then requested.

4.4.3 Prediction-funnel alarm strategy

In step 2, the novelty consists in the use of the prediction-funnel within the alarm raising strategy. In devising the new strategy, the starting point was noting that several approaches proposed in literature focused on one single prediction and they seldom account for prediction accuracy in detecting the crossing of the hypoglycemic threshold [56, 58]. So, the proposed alarm strategy considers simultaneously multiple predictions at different PH, accounting also for their uncertainty. As previously discussed, when employing the Kalman predictor, it is now possible to obtain multiple predictions in a computationally efficient manner:

$$\hat{g}(k+1|k), \hat{g}(k+2|k), \ldots, \hat{g}(k+PH|k), \tag{4.14}$$

and the corresponding variance of the prediction error, for each PH:

$$\sigma^2(k+1|k), \sigma^2(k+2|k), \ldots, \sigma^2(k+PH|k). \tag{4.15}$$

This information is used to equip each prediction with a confidence interval,

$$\hat{g}(k+i|k) \pm m\sigma(k+i|k), \text{ for } i = 1, \ldots, PH_{\max} \tag{4.16}$$

thus, obtaining a prediction–funnel, shown in Figure 4.2.
The parameter m is a tuning parameter that increases or decreases the width of the funnel.

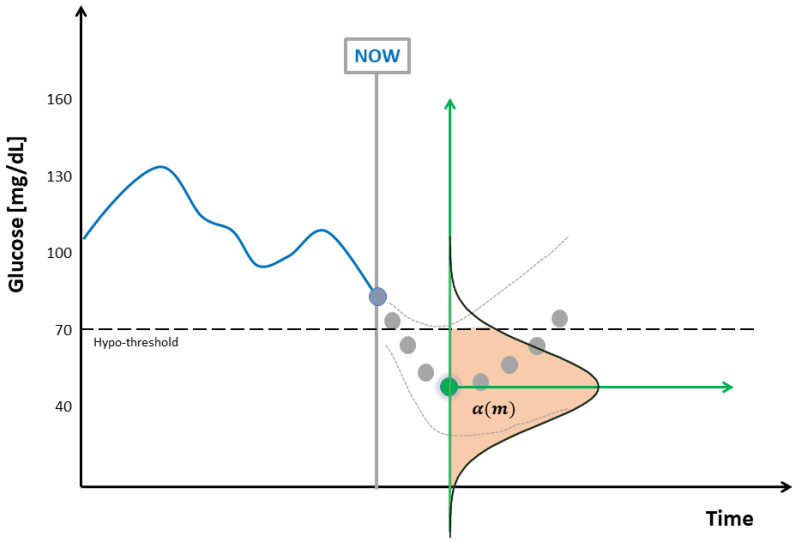


Figure 4.2: Illustration of the prediction-funnel and the role of $\alpha(m)$. The blue dot is the current BG values ($g(k)$) the grey dots are the predicted glucose levels $\hat{g}(k+1|k), \hat{g}(k+2|k), \ldots, \hat{g}(k+PH_{max}|k)$, the dashed lines are the confidence intervals. In this illustrative example, the orange area, $\alpha(m)$ is the probability that $\hat{g}(k+4\,|\,k)$ (green dot) is below the hypoglycemic threshold.

Once the prediction–funnel is obtained, the upper bound of the prediction's confidence intervals is monitored to detect if it goes below the hypoglycemic threshold $tr_{\text{h}} = 70$ mg/dL for some of the considered prediction horizons. In other words, it is checked if

$$\hat{g}(k+PH|k) + m\sigma(k+PH|k) \leq tr_{\text{h}} = 70 \text{ mg/dL} \tag{4.17}$$

for any $PH = 1, \ldots, PH_{\text{max}}$.

An alarm is then raised if this happens for at least N_{pred} samples of the considered prediction horizons.

For instance, if $N_{pred} = 1$, to raise an alarm it is required that one sample of the prediction-funnel falls below the hypoglycemic threshold. On the contrary, if $N_{pred} = PH_{\text{max}}$, an alarm is raised if the entire prediction-funnel is found to be below the hypoglycemic threshold. A preliminary investigation to tune N_{pred}, reported in the following section, shows that the best performances are achieved with $N_{pred} = 1$ and $PH_{\text{max}} = 60$ minutes.

The tuning of the remaining degree of freedom, the parameter m, will be briefly described in the following, proposing two different strategies. In fact, as described in equation 4.17, increasing m will make the alarm more conserva-

tive. To gain a more intuitive feeling on the role of this parameter, it might be useful to interpret m as the tuning knob associated with the probability $\alpha(m)$ that the predicted BG level falls below 70 mg/dL, see Figure 4.2. While the exact relationship between m and the probability $\alpha(m)$ depends on the true distribution of the prediction error, a common approximation is to assume it as a normal distribution. The accuracy of this approximation is not of critical important since the only purpose of $\alpha(m)$ is interpretability and the actual value of this probability is not used with the algorithm, based only on m.

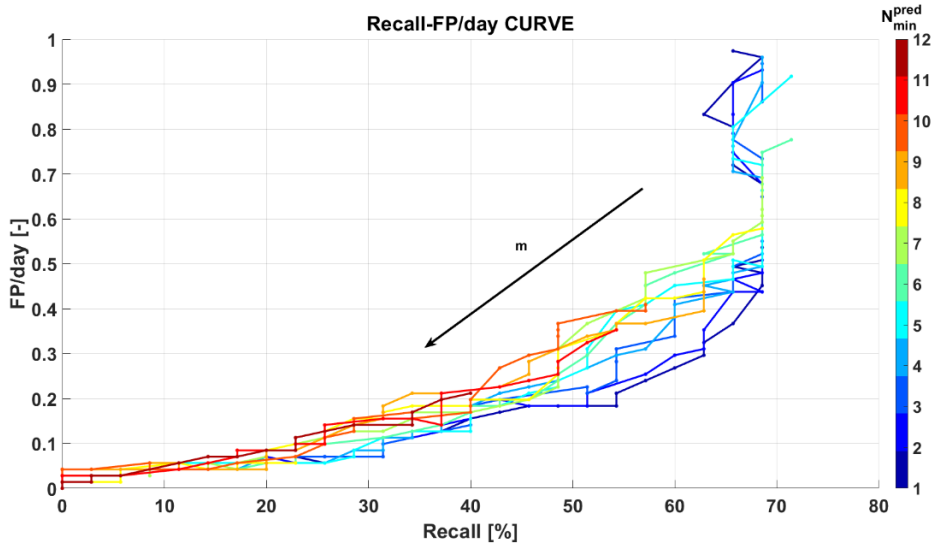


Figure 4.3: Recall vs FP/day analysis: each curve is obtained using different values of N_{pred}, each point is obtained for different values of m.

4.4.4 Hyperparameters tuning

The prediction-funnel alarm strategy requires to accurately tune two parameters: m and N_{pred}. As a first step, we investigated population parameters, that is, same parameters' values for all the individuals. To do so, the performance of the proposed approach are computed for each couple [m, N_{pred}] and they are reported as a point in the space [R, FP/day]. Grid ranges for $m = \{0.01, \ldots, 4\}$ and N_{pred}= $\{1,\ldots,12\}$. Each curve of Figure 4.3 is obtained for different values of N_{pred} and colored depending on it, while different points of the same curve are obtained using different values of m. The ideal performance can be found in the bottom right corner (i.e., R=100%, FP/day=0). As shown in Figure 4.3, increasing N_{pred} leads to performance deterioration: the larger the number of consecutive samples satisfying equation 4.17, the lower the number of hypoglycemic alarms raised by the algorithm, thus leading to an increased number

of *FN*. Considering m, the smaller its value, the more aggressive the algorithm, thus leading to a better R but also to a large number of FP/day. Considering Figure 4.3, m seems to be the most impacting parameter. Therefore, we decided to keep $N_{pred} = 1$ for all the subjects but to optimize m for each subject. In particular, we explored two different personalized strategies to tune m by using the training set. In *tuning 1*, m is set in order to provide a similar recall as the one achieved by the state-of-art (in Table 4.1, MSE + single-PH, PH = 30 min). As a second alternative approach (named *tuning 2*), m was chosen to maximize *F1-score*.

4.5 Assessment of the proposed algorithm

4.5.1 Criteria for the assessment

Following the definition proposed in the consensus paper [154], we say that an hypoglycemic episode (*HE*) has occurred at time k^* if CGM is below 70 mg/dL for a period of at least 15 minutes. Then, similarly to Chapter 2 but with some modifications to take into account for the use of multiple PH, we define:

- True Positive (*TP*) if an *HE* occurred at time k^* and an alarm is activated before the event, precisely in a window from 60 and 5 minutes before k^*. *Remark:* According to this definition, only the alarms which are relatively close to the *HE* are considered correct, while alarms too far apart in the past are not counted as *TP*;
- False Positive (*FP*) if an alarm is raised at time $\bar{k}$ but no hypoglycemia occurred in the following 60 minutes;
- False Negative (*FN*) if an *HE* occurred at time k^* but no alarms are triggered by the algorithm in the previous 60 minutes.

Finally, special attention is given to late alarms, defined as alarms at time k^* or up to 15 minutes after. Clearly, these episodes are not counted as *TP*, as they were not timely triggered. On the contrary, they increase the count of *FN*. However, they cannot be considered as erroneous, so they do not increase the count of *FP*.

Once the events were labeled as *TP*, *FP*, and *FN*, we computed Precision, Recall and F1-score to evaluate the state-of-art and the proposed approaches. As already discussed, precision (*P*) is the fraction of the correct alarms over

the total number of raised alarms. Recall (R), also known as sensitivity, is the fraction of correctly detected hypoglycemic events over the total number of events. F1-score ($F1$) is the harmonic mean of the two previous metrics.

Since the dataset is strongly unbalanced, we also evaluated the average number of FPs generated by the algorithm in one day (FP/day). Finally, we calculated the time gain (TG) of the hypoglycemic alarms as the time between when the alarm was raised by the algorithm and the start of the HE. According to the definition of TP, the maximum achievable TG is 60 min, while the lowest is 5 min.

We reported only the overall recall on the population (fraction of detected hypoglycemic episodes with respect to all the events occurred in the population) and the overall precision, as well as the overall F1-score. To assess the statistical significance of the differences observed in recall and precision a chi-square test of independence with 1 degree of freedom was applied, as suggested in [155]. Moreover, the TG that is reported as mean and standard deviation (SD) of the time gain of every detection (regardless the patient in which the detection occurred).

Whilst the overall test set (7 monitoring days for each subject) includes 42 hypoglycemic episodes, single patient data presents only a small number of hypoglycemic episodes (on average, 1 every 2 day for each subject) and thus single-patient level analysis is omitted. In fact, single patient recall would be strongly quantized: for instance, in a patient with 2 hypoglycemic episodes, recall can take only three values: 100%, 50% or 0%.

4.5.2 Hypoglycemia forecasting performance

Table 4.1 shows the hypoglycemia prediction performances of several configurations of the described prediction algorithms. In particular, the first row, highlighted with grey background, reports the baseline performance achieved by the conventional algorithm using individualized models, identified minimizing MSE, and considering only one prediction horizon, PH = 30 min. The last row of the table, also highlighted with grey background, reports the performance achieved by the new algorithm proposed in this work that includes both the improvements induced (the use of $gMSE$ for model identification and the prediction–funnel-based strategy). To elucidate the contribution of each proposed innovations to the final performance, Table 4.1 reports also the performance achieved with inclusion of one modification at the time ($gMSE$ +

ALARM STRATEGY	PH	COST FUNCTION	P [%]	R [%]	F1 [%]	FP/day [-]	TG [min] mean (SD)	Notes
SINGLE-PH	30 min	MSE	43	95	59	0.77	15 (10)	*Conventional approach*
		gMSE	*44*	*100*	*61*	*0.77*	*15 (10)*	
	45 min	MSE	37	83	51	0.86	15 (10)	
		gMSE	*45*	*98*	*61*	*0.73*	*15 (10)*	
	60 min	MSE	36	76	49	0.82	20 (15)	
		gMSE	*42*	*90*	*58*	*0.75*	*15 (15)*	
PREDICTION-FUNNEL **[tuning 1]**	from 5 to 60 min	MSE	51	91	65	0.59	15 (15)	
		gMSE	*51*	*96*	*66*	*0.59*	*15 (15)*	
PREDICTION-FUNNEL **[tuning 2]**	from 5 to 60 min	MSE	62	76	68	0.33	15 (15)	
		gMSE	*65*	*88*	*75*	*0.29*	*15 (10)*	*Proposed approach*

Table 4.1: Comparison of hypoglycemia prediction performance. The individualized ARIMAX models, identified using different cost functions, are exploited both by applying the single-PH alarm strategy for different PH and the prediction-funnel. Results are reported in terms of precision (P), recall (R), F1-score (F1), False Positive per day (FP/day) and Time Gain (TG), reported as mean (standard deviation).

single-PH and MSE + prediction–funnel). Moreover, different values of the hyper-parameters are investigated.

Focusing first on the single-PH strategy, we investigated the impact of PH on the prediction performance of the state-of-art approach, by evaluating three possible PH: PH = 30, 45, and 60 minutes. The best results are achieved with the PH = 30 min, that is in fact commonly adopted in literature. In particular, the larger the PH, the higher the *TG*, but at the expenses of a worse *P*, *R*, and FP/day. For instance, comparing the state-of-art approach with PH = 30 min and with PH = 60 min we can see that *TG* increases from 15 to 20 minutes (mean values), but *P* falls from 43% to 36% and *R* from 95% to 76%, while FP/day increases from 0.77 to 0.82.

The introduction of the *gMSE* in place of the MSE, improves both the precision and the recall with respect to the state-of-art approach. This holds true for all considered values of PH. Considering for instance PH = 30 min, with the use of *gMSE*, *P* increases from 43% to 44%, *R* from 95% to 100%, while FP/day and *TG* are almost the same.

We then investigated the impact of the improved alarm strategy (prediction–funnel instead of using a single-PH). In particular, two different approaches to the tuning of the parameter *m* were considered. In both cases, we considered a patient-specific *m*, but in the first approach *m* was set in order to get a similar recall as the one achieved by the state-of-art (MSE + single-PH, PH = 30 min). As a second alternative approach *m* was chosen in each patient to maximize the *F1* in the training-set of that patient. The first approach is presented in Table 4.1 as *tuning 1*, whereas the second is denoted *tuning 2*.

The improvement granted by the prediction–funnel is clearly visible with *tuning 1*, that offers higher precision, higher *F1* and less FP/day with respect to the state-of-art: *P* increase from 43% to 51%, *F1* from 59% to 65%, and FP/day decreased from 0.77 to 0.59. This improvement is achieved while retaining similar recall (*R* from 95% to 91%). The performances of *tuning 2* are more difficult to interpret, since it selects a different trade-off between precision and recall. Specifically, it renounces to some recall in favor of a better precision and less FP/day, leading to an overall improvement in the F1-score.

Finally, combining the two improvements, the algorithm proposed in this work outperforms the state-of-art. Once again, this is clearly visible with *tuning 1*, that grants similar recall and *TG* of the state-of-art (above 95%) but with higher precision, F1-score, and less false positives: *P* from 43% to 51%, *F1* from 59% to 69%, and false positives-per-day decreased from 0.77 to 0.59. By adopt-

ing a slightly more conservative approach, proposed in *tuning 2*, precision and false positives can be further improved ($P = 65\%$, $FP/day = 0.29$, i.e., about one every 3 days), at the expenses of a deterioration of the recall ($R = 88\%$). This new trade-off offers a better F1-score that reaches 75%. The chi-square test proposed in [155] shows that the proposed approach has statistically significant better precision and FP/day, with respect to the conventional approach (p-value< 0.01 and <0.0001, respectively). No significant difference is found on in the recall (p-value= 0.22).

Remark. According to HE definition formulated in [154], an event in which CGM data fall below the hypoglycemic threshold only for one- or two-time samples should not be considered an hypoglycemia. However, if an alarm is raised in these situations (that we will call in the following quasi-hypoglycemic episodes, *qHE*), such an alarm will count as a *FP*. *FP* error caused by a *qHE* could be considered less clinically relevant than other *FPs*, therefore we also report how many FPs are of this kind. For the state-of-art method (MSE + single-PH approach, PH = 30 min), 26% of the recorded *FP*s were associated to *qHE*, while this percentage raises to 34% with the newly proposed algorithm (*gMSE* + prediction–funnel), further supporting the superiority of the proposed algorithm. Discarding these events from the *FP* count, as frequently done in literature, would reduce the FP/day from 0.77 to 0.62 and increase the precision from 43% to 54% for the state-of-art approach while, for the proposed approach, would decrease FP/day from 0.29 to 0.21 and increase P from 65% to 73%. Another consequence of the *HE* definition adopted, is that even a non-predictive hypoglycemia detection algorithm, simply based on the CGM trace crossing the 70 mg/dL threshold, may raise false positive alarms ($FP/day = 0.2$). In fact, according to our definition of TP, the CGM reading produces only late hypo-alarms (events that will be detected exactly whenever they started: TG = 0 min). As a consequence, TP count is bound to be 0, and both recall and precision are necessarily 0, (i.e., P=R=0).

4.5.3 Comparison with state-of-art

While the presented results showed the benefit of including the two proposed novelties in the conventional, regression-based, linear hypoglycemia prediction algorithm, it might be of interest to investigate how the prediction performance of the improved algorithm positions with respect to other contributions. To this aim, we propose a tentative comparison based on the results re-

ported in the previously mentioned works [48, 58, 114, 125, 139, 145, 149, 150]. However, it is difficult to define a fair comparison: different definition of the events might significantly impact on the final metrics, as shown by the previous section, where a seemingly minor modification in the definition of *HE* has non-negligible impact on FP/day and precision. Moreover, different dataset might be collected in very different conditions (highly controlled clinical trials *vs.* real-life) introducing a further confounding factor. Authors in [48, 58, 149] reached a recall, respectively, of about 93%, 93%, and 86%, comparable or slightly superior to the recall of our algorithm, with $R = 88\%$, at the expense of lower precision: about 24%, 38%, and 62%, while the proposed approach in this work achieved $P = 65\%$. Similarly, [58, 125] achieved the remarkable recall of $R = 100\%$, but at the expense of a very high number of *FP*s (more than 1 *FP*s per day). Authors in [114, 139, 140] showed a similar recall to the one obtained in this work (89%, 94%, and 94%) with a better precision (78%, 90%, and 77%). However, the authors adopted a more permissive *HE* definition. For instance, [114] considered as *TP*s also alarms raised after the CGM crossed the hypoglycemic threshold, whereas we consider them as *FN*s. In [139], performance was assessed using controlled inpatient data. Authors in [150] showed a slightly inferior recall (86%) and did not report any metrics related to false alarms. So, to overcome the above-mentioned limitations in comparing literature contributions with the proposed algorithm, we provided the performance granted by a baseline (we named conventional approach). Remarkably, such a baseline is found to be a challenging competitor: its hypoglycemia predictive performance are in line -or even outperform- some literature findings, for instance [48, 58, 149]. However, the proposed solution largely improves the conventional approach in terms of precision (65% vs 43%, p-value < 0.01) false positive per day (0.29 vs 0.77, p-value <0.0001) and F1-score (75% vs 59%) at the expenses of a slightly but not statistically significant (p-value $= 0.22$) deterioration of the recall (88% vs 95%).

4.6 Summary of the main findings

In this chapter we explored a new approach to predict hypoglycemic events that is based on an individualized ARIMAX models, identified by using a cost function specifically designed to account for the clinical impact of prediction error, and on a novel alarm strategy that considers the entire prediction–funnel. The results show that models identified via *gMSE* minimization

provide better hypoglycemia prediction performances than models based on MSE. Furthermore, results show that the new alarm-raising strategy based on the prediction–funnel improves hypoglycemia forecasting, thanks to the possibility of exploiting multiple PH. The adoption of both the proposed improvements grants the best performances.

As a final remark, we focused on personalized models in order to deal with the large inter-individual variability characterizing T1D population. The slow changes in patient physiology occurring over the weeks (intra-patient variability) are not evaluated as this contribution focuses on 1 week of data. A natural option to deal with intra-patient variability is to resort to recursive model identification techniques, well-established methods for tracking the changes in patient dynamic by updating the model every time a new measurement becomes available [59, 156, 157]. Nevertheless, given the relatively slow time scale of intra-patient changes, daily or weekly updates of the model, simply obtained by periodically repeating the proposed model identification procedure on the most recent 7 days of data, are expected to be sufficient.

4.7 Preliminary conclusions on the use of different input information and algorithms

The results presented in Chapters 2, 3 and 4, can define a preliminary picture about the importance of different input information and algorithms for glucose prediction and hypoglycemia forecasting. As a general trend, predictive performance can be classified on the basis of the PH:

- for short-term prediction (PH$\leq$30 min), CGM-only predictive algorithms based on linear models (in particular, ARIMA-based predictors) represent a practically valuable option to achieve accurate BG predictions, even if compared to their nonlinear counterpart (i.e., NN). However, an intrinsic drawback of all these models (posing some issues about their performance in long-term forecasting) is that any metabolic disturbance, e.g., a meal, would deteriorate the accuracy of the predicted BG levels;

- for PH $>$ 30 min, the addition of insulin and CHO data to CGM in prediction models can improve the performance of algorithms. Also in this scenario, the use of nonlinear techniques tested so far, substantially more complex than their linear counterparts, does not seem to improve the

prediction performance significantly. This suggests that linear methods are still valuable options that offer a good trade-off between complexity and performance. However, the real-world application of these methods is limited to the use of supplementary portable devices and / or dedicated mobile applications that, at present, are not widely used by most individuals with diabetes;

- for PH>45 min, the use of CGM data and mealtime (to enforce an artifical seasonality in glucose data) allows C-SARIMA to outperform CGM-only models. This approach has the practical advantage of minimal input information required (reducing patient's burden in recording meal information). However, this method does not provide a significant improvement with respect to algorithms that employ both CHO and insulin (dosing and timing information).

Focusing on the prediction of hypoglycemia is a slightly different task from the forecasting of glucose levels. However, as described in Chapter 2, connecting these two different problems is possible by using specific algorithms or strategies that should be used in cascade to the punctual BG prediction to raise hypoglycemic alerts. As shown in Chapters 2 and 3, this task is challenging for all algorithms (linear and nonlinear) and by using any combination of the inputs considered (CGM only, CGM and meal time, CGM and CHO and insulin). Therefore, we modified the standard approach by introducing two innovations to improve the overall performance. Of note, the proposed novel approach, with adequate adjustment, can be employed in any nonlinear BG predictive models.

So far, we focused on intensively evaluating BG forecasting algorithms based on traditional black-box methods, such as ARIMA, ARIMAX, NN and other variants. Since the use of CGM, CHO and insulin data is found to be the best input information, we move a step forward by investigating a white-box physiological model within the particle filter to forecast BG levels, and by proposing other more complex machine and deep learning methods for BG forecasting.

Chapter 5

White-box and advanced black-box models for BG forecasting

[1] So far, we have modeled the complex glucose-insulin system using black-box models that aim to learn the input-output relationship from patient-recorded data. We have shown that the best predictive performance can be achieved by adding CHO and insulin (timing and dosing) information to CGM data in predictive models. Moreover, we found that there are no large differences in the forecasting accuracy among the evaluated linear and nonlinear state-of-art algorithms once fed by same input information. Another option to describe glucose dynamics is to resort to physiological white-box approaches: mathematical models characterized by a physiologically consistent mathematical structure (usually defined by several differential equations) and a set of model parameters. The development of a personalized algorithm based on a large scale physiological model is a challenging task that requires advanced tools for its identification and dedicated approaches to employ it for the prediction of BG levels. In this chapter, we explore the potential of using a physiological model by introducing: i) a reduced version of the Uva/Padova maximal model of glucose-insulin dynamics; ii) a Markov Chain Monte Carlo (MCMC) identification strategy for model individualization, and iii) a particle filter framework for the forecasting of BG levels. The derived physiological model is compared to three deep learning algorithms and to an advanced linear non-parametric approach that recently proved effective in glucose prediction [62]. Algorithms' assessment has been performed on the OhioT1DM dataset.

[1] This chapter contains material to be submitted for publication as *Cappon G.*, Prendin F.*, et al., "Individualized Models for Glucose Prediction in Type 1 Diabetes: comparing black-box approaches to a physiological white-box one", IEEE Transaction on Biomedical Engineering, 2022.*

5.1 Physiological-based and data-driven models

5.1.1 Chapter contribution

In the recent years, the large availability of data coupled with the technological advancement in diabetes management have led the community to investigate more complex and advanced methodologies for BG forecasting, such as novel machine/deep learning strategies [41, 46, 80] and non-parametric models [62]. These data-driven models enable the description of complex, nonlinear interactions between input and output data disregarding any prior knowledge about the underlying physiological process and they do not need any detailed description of the internal, metabolic dynamics of the glucose-insulin systems. However, unknown disturbances, large inter-/intrapatient variability in glucose physiology make accurate BG predictions a challenging and still open problem. In this context, physiological white-box models represent an alternative to black-box approaches. In particular, among the nonlinear physiological models available in the T1D literature, there are the so-called minimal models [158], that proposed simplified descriptions of the physiology with a few equations and model parameters. This parsimonious parametrization grants identifiability in pre-defined experimental conditions but, unfortunately, these models have proved too rigid and simplistic to allow accurate prediction [50]. A possible white-box alternative are maximal models, commonly used in computer simulations [85, 86, 159, 160]. They provide a more realistic physiological description by using several equations with many parameters. A key challenge that prevented the use of these models for glucose prediction is that their many parameters are hard to be estimated form easily accessible patient data (i.e. CGM, meal and insulin data), making them hard to personalize and thus limiting their predictive effectiveness. Moreover, they have a nonlinear structure, requiring sophisticated tools both for parameters estimation and for the computation of glucose prediction.

In this chapter, we face the challenges mentioned above and explore the potential of using a white-box maximal-model based methodology for glucose prediction, comparing it to black-box alternatives. The adopted white-box model is inspired by the Uva/Padova T1D Simulator (T1DS), [161], accepted by the US Food and Drug Administration (FDA) as a replacement of animal preclinical testing of closed-loop drug delivery systems. Also, T1DS provides a population of 100 virtual adult subjects, each characterized by a different vec-

tor of physiological parameters to capture the inter-/intra-subject variability of T1D population. A Bayesian approach, implemented by Markov Chain Monte Carlo (MCMC) [162], is used to estimate the large number of parameters in the presence of complex nonlinear dynamics. The obtained personalized model is then used within a nonlinear prediction scheme based on a particle filter methodology [163]. For what it concerns the deep learning models, we developed two different recurrent neural network architectures that implement feedback connections to learn the long and short-term dependencies in time series data: a multi input LSTM and a Gated Recurrent Unit (GRU). Also, we implemented a Temporal Convolutional Network (TCN) that uses convolutional operations to capture local and temporal information. Finally, predictive performance are assessed on the OhioT1DM dataset. These open-loop data, which are recorded under free-living conditions, represent a suitable choice for our scope as: i) they comprise large disturbances to glucose homeostasis (making the prediction of BG levels very challenging) and ii) they avoid the introduction of identification artifacts that usually occur when dealing with data from complex dynamical systems under closed-loop conditions [109, 151, 164].

5.1.2 Chapter outline

This chapter is organized as follows. Section 5.2 describes the proposed nonlinear physiological model of glucose-insulin regulation used for BG prediction (as well as the a priori information available from the literature on its parameters). Section 5.2.4 presents its identification through a Markov Chain Monte Carlo (MCMC) Bayesian estimator capable of personalizing model parameters. Section 5.2.5 describes the particle filter framework to compute glucose predition ahead in time. Section 5.3 introduces the black-box deep learning models and the advanced nonparametric approach. Finally, Section 5.5 illustrates the predictive performance and Section 5.6 summarizes the main findings.
Of note, Appendix A provides a detailed description of the identification procedure of the physiological model and the particle filter framework to predict BG levels; Appendix B detailed the deep learning methodologies, and Appendix C illustrates the nonparametric approach.

5.2 Physiological model-based algorithm

The maximal-physiological model of glucose-insulin dynamics described in [161], consists of 18 differential equations and 39 parameters. In particular, we can identify three main systems: the glucose, insulin and glucagon ones. Briefly, the model relates the measured plasma concentrations (i.e., glucose, insulin, and glucagon) with:

- the internal glucose fluxes (meal rate of appearance, endogenous production, utilization, and renal extraction);
- the insulin fluxes (rate of appearance of subcutaneous insulin and degradation);
- the glucagon fluxes (secretion and degradation).

Given the complexity of the model and the large number of parameters to be estimated for individualization, we decided to use a simplified version of the model composed by 9 differential equations and 10 unknown parameters, as described in [165]. The proposed model comprises the subcutaneous insulin absorption, the oral glucose absorption, and the glucose-insulin kinetics subsystems and it has two inputs: insulin infusion $I(t)$, and carbohydrate intake $CHO(t)$, and one output, the interstitial glucose concentration $IG(t)$.

5.2.1 Subcutaneous insulin absorption subsystem

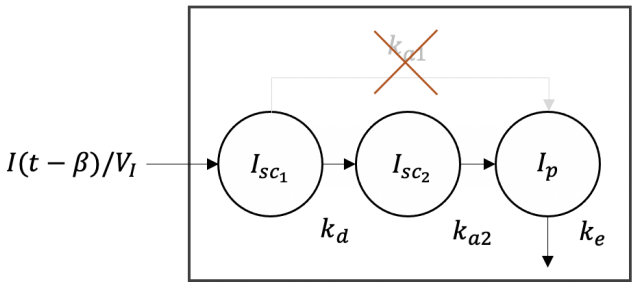


Figure 5.1: Subcutaneous insulin absorption subsystem scheme.

The subcutaneous insulin absorption model is a slightly simplified version of the one described in [166] and illustrated in Figure 5.1. The model is made up of three compartments. Exogenous insulin (I) is infused into the first compartment, where it appears after a delay β. In the first compartment, representing insulin in a non-monomeric state, insulin is transformed in a monomeric

state and then diffused to the plasma. The model equations are:

$$\begin{cases} \dot{I}_{sc1}(t) = -k_d \cdot I_{sc1}(t) + I(t-\beta)/V_I \\ \dot{I}_{sc2}(t) = k_d \cdot I_{sc1}(t) - k_{a2} \cdot I_{sc2}(t) \\ \dot{I}_p(t) = k_{a2} \cdot I_{sc2} - k_e \cdot I_p(t) \end{cases} \tag{5.1}$$

where I_{sc1} (mU/kg) and I_{sc2} (mU/kg) represent the insulin in a non-monomeric and monomeric state, respectively; I_p (mU/l) is the plasma insulin concentration; k_d (min^{-1}) is the rate constant of diffusion from the first to the second compartment; k_{a2} (min^{-1}) is the rate constant of subcutaneous insulin absorption from the second compartment to the plasma; k_e (min^{-1}) is the fractional clearance rate; V_I (l/kg) is the volume of insulin distribution; β (min) is the delay in the appearance of insulin in the first compartment. The simplification with respect to the maximal-model [166] consists in neglecting the fact that a fraction of non-monomeric insulin can reach the plasma directly. This absorption route is depicted in gray in Fig. 5.1 and associated to the rate constant k_{a1}. The simplification is due to the fact that both the insulin fraction and the number of patients where this happens is small [166]. A priori information on model parameters, reported in Table 5.1, have been obtained from the literature [166]. Specifically, V_I and β have been set to population values, i.e. 0.126 l/kg and 8 min, respectively. Furthermore, k_d has been constrained to $k_d \geq k_{a2}$ since the two combinations are interchangeable. Unknown model parameters result $\boldsymbol{\theta}_{ins} = [k_{a2}, k_d]$.

5.2.2 Oral glucose absorption subsystem

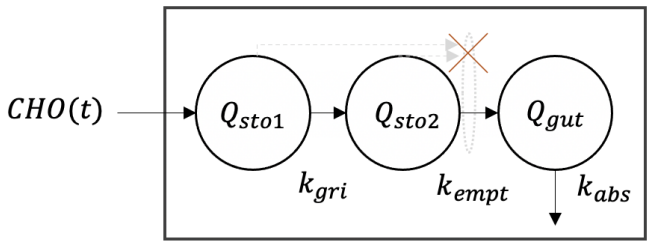


Figure 5.2: Oral glucose absorption subsystem scheme.

The oral glucose absorption subsystem model, taken from [167], illustrated in Figure 5.2, describes the gastro-intestinal tract as three-compartment system: the first two compartments account for food in the stomach (solid and

grinded state), while the third compartment models the upper small intestine where CHO is absorbed. Model equations are:

$$\begin{cases} \dot{Q}_{sto1}(t) = -k_{gri} \cdot Q_{sto1}(t) + CHO(t) \\ \dot{Q}_{sto2}(t) = k_{gri} \cdot Q_{sto1}(t) - k_{empt} \cdot Q_{sto2}(t) \\ \dot{Q}_{gut}(t) = k_{empt} \cdot Q_{sto2}(t) - k_{abs} \cdot Q_{gut}(t) \end{cases} \tag{5.2}$$

where Q_{sto1} (mg/kg) and Q_{sto2} (mg/kg) are the glucose amount in the stomach in a solid and liquid state, respectively; Q_{gut} (mg/kg) is the glucose concentration in the intestine; k_{gri} (min^{-1}) is the rate constant of grinding; k_{empt} (min^{-1}) is the rate constant of gastric emptying; k_{abs} (min^{-1}) is the rate constant of intestinal absorption; CHO (mg/kg/min) is the ingested carbohydrate rate. Model (5.2) allows to estimate the rate of glucose appearance in plasma Ra (mg/kg/min) as:

$$Ra(t) = f \cdot k_{abs} \cdot Q_{gut}(t) \tag{5.3}$$

where f (dimensionless) is the fraction of the intestinal content absorbed in the plasma. The simplification consists in assuming a constant gastric emptying rate, thus neglecting its dependence on stomach content (depicted in gray in Fig. 5.2). A priori information on model (5.2) has been obtained from the literature [167] and has been detailed in Table 5.1. In particular, we set f equal to 0.9 and we constrained $k_{gri} = k_{empt}$. Furthermore, k_{abs} has been constrained to $k_{abs} \leq k_{empt}$ since the two combinations are interchangeable. As such, the unknown model parameters are $\boldsymbol{\theta}_{oral} = [k_{abs}, k_{empt}]$.

5.2.3 Glucose-insulin kinetics subsystem

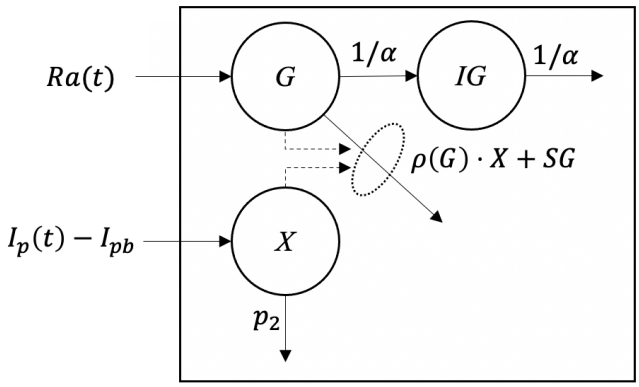


Figure 5.3: Glucose-insulin kinetics subsystem scheme.

This subsystem is based on a well-known two-compartment model that describes the impact of the plasmatic insulin action and glucose rate of appearance in plasma glucose concentration introduced in [84]. The model is further equipped with a third compartment to describe the transport of glucose from plasma to the interstitium where it is measured by the sensor. The model is illustrated in Fig. 5.3. Model equations are:

$$\begin{cases} \dot{G}(t) = -[SG + \rho(G)X(t)] \cdot G(t) + SG \cdot G_b + Ra(t)/V_G \\ \dot{X}(t) = -p_2 \cdot [X(t) - SI \cdot (I_p(t) - I_{pb})] \\ \dot{IG}(t) = -\frac{1}{\alpha}(IG(t) - G(t)) \end{cases} \tag{5.4}$$

where G (mg/dL) is the plasma glucose concentration, X (min^{-1}) is the insulin action on glucose disposal and production; SG (min^{-1}) is the glucose effectiveness describing glucose ability, per se, to promote glucose disposal and inhibit glucose production; G_b (mg/dL) is the basal glucose concentration in the plasma; V_G (dL/kg) is the volume of glucose distribution; p_2 (min^{-1}) is the rate constant of insulin action dynamics; SI (mL/μU·min) is the insulin sensitivity; I_{pb} (mU/L) is the basal insulin concentration in the plasma; IG (mg/dL) is the interstitial glucose concentration; α (min) is the delay between the plasmatic and interstitial glucose concentration compartments.

The above model, originally introduced in [84] with a constant unitary ρ, $\rho(G) \equiv 1 \; \forall G$, is known to struggle in capturing hypoglycemia, likely due to an inadequate description of insulin action, that was proved to increase when glucose decreases below a certain threshold [168]. For this reason, following the rationale proposed by Dalla Man et al. [132], we introduce in the above model the term $\rho(G)$:

$$\rho(G) = \begin{cases} 1 & \text{if } G \geq G_b \\ 1+10r_1\{[ln(G)]^{r_2} - [ln(G_b)]^{r_2}\}^2 & \text{if } G_{th} < G < G_b \\ 1+10r_1\{[ln(G_{th})]^{r_2} - [ln(G_b)]^{r_2}\}^2 & \text{if } G \leq G_{th} \end{cases} \tag{5.5}$$

where we have $G_{th} < G_b$, with G_{th} is the hypoglycemic threshold (set to 60 mg/dL), and r_1 (dimensionless) and r_2 (dimensionless) are two model parameters with no direct physiological interpretation.

To account for patient-specific intraday insulin sensitivity variability [169] and to model the so-called dawn phenomenon, the parameters SI is considered

time-varying over the day:

$$SI = \begin{cases} SI_B & \text{if} \quad 4\text{ AM} < t \leq 11\text{ AM} \\ SI_L & \text{if} \quad 11\text{ AM} < t \leq 5\text{ PM} \\ SI_D & \text{otherwise} \end{cases} \tag{5.6}$$

Of note, these fixed time intervals are used in the Uva/Padova T1DS and were defined according to [169]. We acknowledge that the choice of fixed time intervals for all subjects may be considered as a limitation of the proposed model. Future studies will address this issue by considering time intervals as uniform random variables.
A priori information on parameter distributions, reported in detail in Table 5.1, has been obtained from the literature [170]. In detail, r_1, r_2, and V_G have been fixed to population values, i.e. 1.44, 0.81, and 1.45 dL/kg respectively. Unknown model parameters of glucose-insulin subsystem are $\boldsymbol{\theta}_{glu}$ = [SG, SI_B, SI_L, SI_D, G_b, p_2].

In summary, the overall physiological model is obtained combining the three submodels introduced so far:

$$\begin{cases} \dot{\boldsymbol{x}}_{phy}(t) = \boldsymbol{f}_{phy}(\boldsymbol{x}_{phy}, \boldsymbol{u}_{phy}, t, \boldsymbol{\theta}_{phy}) \\ y(t) = IG(t) \end{cases} \tag{5.7}$$

where $\boldsymbol{x}_{phy}(t)$ is the state vector defined as

$$\begin{aligned} \boldsymbol{x}_{phy}(t) &:= [\boldsymbol{x}_{ins}, \mid \boldsymbol{x}_{oral}, \mid \boldsymbol{x}_{glu}]^T \\ &= [I_{sc1}, I_{sc2}, I_p, \mid Q_{sto1}, Q_{sto2}, Q_{gut}, \mid G, X, IG]^T; \end{aligned}$$

$\boldsymbol{u}_{phy}(t) := [I(t), CHO(t)]$ is the input vector;
$\boldsymbol{f}_{phy}(\cdot)$ is the state update function obtained combining (5.1)-(5.5).
$\boldsymbol{f}_{phy}$ depends on the set of unknown parameters

$$\boldsymbol{\theta}_{phy} := [\boldsymbol{\theta}_{ins}, \boldsymbol{\theta}_{oral}, \boldsymbol{\theta}_{glu}]$$

whose estimation will be discussed in the next section.

5.2.4 Identification of the proposed physiological model

Model personalization has been performed by identifying for each patient the unknown model parameters $\boldsymbol{\theta}_{phy}$ using the training data $Y := \{CGM(t_k), t_k =$

Table 5.1: A Priori Information on Model Parameters

Parameter	Distribution	Reference
SG	LOGN(-3.8,0.5)	[170]
SI_B	GAMMA(3.3,4.5e-4)	[170]
SI_L	GAMMA(3.3,4.5e-4)	[170]
SI_D	GAMMA(3.3,4.5e-4)	[170]
G_b	N(120,5)	[170]
$sqrt(p_2)$	N(0.11,0.004)	[170]
k_{a2}	LOGN(-4.29,0.43) and $k_d \geq k_{a2}$	[166]
k_d	LOGN(-3.51,0.62) and $k_d \geq k_{a2}$	[166]
k_{abs}	LOGN(-5.46,1.44) and $k_{abs} \leq k_{empt}$	[167]
k_{empt}	LOGN(-1.96,0.71) and $k_{abs} \leq k_{empt}$	[167]

[2]N(μ,σ) stands for a normal distribution of mean μ and standard deviation σ. LOGN(μ,σ) stands for a log-normal distribution of mean μ and standard deviation σ. GAMMA(a,b) stands for a gamma distribution with shape parameter a and scale parameter b.

$k \cdot T_s, k = 1, \ldots, D\}$ and $U := \{\boldsymbol{u}_{phy}(t_k), t_k = k \cdot T_s, k = 1, \ldots, D\}$ where D is the number of data points available.

The identification has been performed by adopting a Bayesian approach, implemented by MCMC [162], and specifically, in this work $\boldsymbol{\theta}_{phy}$ is estimated through its posterior mean defined as

$$\hat{\boldsymbol{\theta}}_{phy} = E[\boldsymbol{\theta}_{phy}|Y, U] = \int \boldsymbol{\theta}\, p_{\boldsymbol{\theta}|Y,U}(\boldsymbol{\theta}|Y, U)d\boldsymbol{\theta} \tag{5.8}$$

where the posterior mean is known to be the minimum-variance unbiased estimator of $\boldsymbol{\theta}_{phy}$. The Bayes theorem allows to obtain the a posteriori density function $p_{\boldsymbol{\theta}|Y,U}(\boldsymbol{\theta}|Y, U)$ as:

$$p_{\boldsymbol{\theta}|Y,U}(\boldsymbol{\theta}|Y, U) = \frac{p_{Y|\boldsymbol{\theta},U}(Y|\boldsymbol{\theta}, U)p_{\boldsymbol{\theta}}(\boldsymbol{\theta})}{\int p_{Y|\boldsymbol{\theta},U}(Y|\boldsymbol{\theta}, U)p_{\boldsymbol{\theta}}(\boldsymbol{\theta})d\boldsymbol{\theta}} \tag{5.9}$$

where $p_{Y|\boldsymbol{\theta},U}(Y|\boldsymbol{\theta}, U)$ is the likelihood function, i.e., the probability of observing a certain Y given the parameter vector $\boldsymbol{\theta}$ and the input U.

Even using (5.9), the integral in (5.8) is analytically intractable, therefore it has to be approximated by resorting to MCMC [162]. In particular, we generate N samples $\boldsymbol{\theta}_i, i = 1, \ldots, N$ from the posterior distribution $p_{\boldsymbol{\theta}|Y,U}(\boldsymbol{\theta}|Y, U)$, by creating a Markov Chain whose stationary distribution is exactly this posterior (target distribution). Then, these samples $\boldsymbol{\theta}_i$ are used to perform Monte Carlo

integration to obtain a point estimate of $\boldsymbol{\theta}_{phy}$:

$$\hat{\boldsymbol{\theta}}_{phy} = \frac{1}{N}\sum_{i}^{N}\boldsymbol{\theta}_i. \tag{5.10}$$

To build such a chain, the Single Component Metropolis-Hastings (SCMH) algorithm has been used [162]. Implementation details are reported in Appendix A.

5.2.5 Physiological model-based prediction

Real-time glucose prediction can be performed by resorting to a sequential algorithm that at each time t_k, when a new measurement $y(t_k) = CGM(t_k)$ becomes available, updates the current estimate of the model state $\boldsymbol{x}(t_k)$ and uses it to infer future glucose concentration. In particular, we employ the particle filter (PF) [163], the state-of-the-art sequential Bayesian prediction technique capable of handling the nonlinear structure of the model. PF is based on the recursive update of the posterior probability function $p(\boldsymbol{x}(t_k)|y(t_{1:k}), \boldsymbol{u}(t_{1:k}))$ where $y(t_{1:k})$ is a shorthand for the variables $y(t_1), \ldots, y(t_k)$ and $\boldsymbol{u}(t_{1:k})$ indicates $\boldsymbol{u}(t_1), \ldots, \boldsymbol{u}(t_k)$.

The recursive update of $p(\boldsymbol{x}(t_{k-1})|y(t_{1:k-1}), \boldsymbol{u}(t_{1:k-1}))$ is performed through two fundamental steps, i.e., one step-ahead prediction and measurement update.

The one step-ahead prediction step assumes that the posterior probability $p(\boldsymbol{x}(t_{k-1})|y(t_{1:k-1}), \boldsymbol{u}(t_{1:k-1}))$ is available at time t_{k-1} and uses such a posterior probability to infer

$$p(\boldsymbol{x}(t_k)|y(t_{1:k-1}), \boldsymbol{u}(t_{1:k}))$$

Then, when at time t_k a new measurement, $y(t_k)$, becomes available, in the measurement update step such a measurement is used to compute the posterior probability

$$p(\boldsymbol{x}(t_k)|y(t_{1:k}), \boldsymbol{u}(t_{1:k})).$$

The two steps are then repeated for each available measurement in the dataset. PF performs these steps using a sampled approximation of the probability

functions at play:

$$p(\boldsymbol{x}(t_{k-1})|y(t_{1:k-1}),\boldsymbol{u}(t_{1:k-1})) \approx \sum_{p=1}^{P} w^p(t_{k-1})\delta(\boldsymbol{x}(t_{k-1}) - \boldsymbol{x}^p(t_{k-1})).$$

where $\{\boldsymbol{x}^p(t_{k-1})\}_{p=1}^{P}$ is a set of P points, called "particles", in the support of $p(\boldsymbol{x}(t_{k-1})|y(t_{1:k-1}), \boldsymbol{u}(t_{1:k-1}))$.
Each particle is associated to a weight $\{w^p(t_{k-1})\}_{p=1}^{P}, \sum_p w^p(t_{k-1}) = 1$, and

$$p(\boldsymbol{x}(t_k)|y(t_{1:k-1}),\boldsymbol{u}(t_{1:k})) \approx \sum_{p=1}^{P} w^{*p}(t_k)\delta(\boldsymbol{x}(t_k) - \boldsymbol{x}^p(t_k)).$$

where $\{\boldsymbol{x}^p(t_k)\}_{p=1}^{P}$ are the P particles representing $p(\boldsymbol{x}(t_k)|y(t_{1:k-1}), \boldsymbol{u}(t_{1:k}))$, each associated to a weight $\{w^{*p}(t_k)\}_{p=1}^{P}, \sum_p w^{*p}(t_k) = 1$.

Regarding the measurement update step, it is possible to demonstrate that it holds

$$p(\boldsymbol{x}(t_k)|y(t_{1:k}), \boldsymbol{u}(t_{1:k})) \propto p(y(t_k)|\boldsymbol{x}(t_k), \boldsymbol{u}(t_{1:k}))p(\boldsymbol{x}(t_k)|y(t_{1:k-1}), \boldsymbol{u}(t_{1:k}))$$

where $p(y(t_k)|\boldsymbol{x}(t_k), \boldsymbol{u}(t_{1:k}))$ is the likelihood function that is fully specified by (5.7).

As an additional result, the posterior probability $p(\boldsymbol{x}(t_k)|y(t_{1:k}), \boldsymbol{u}(t_{1:k}))$ is further used by the PF to compute the posterior probabilities

$$p(\boldsymbol{x}(t_{k+i})|y(t_{1:k}), \boldsymbol{u}(t_{1:k+i})), \forall i = 1, \ldots, PH$$

describing the state distribution predicted i steps-ahead in time up to PH steps ahead.

Finally, a point estimate of future CGM values at time $t_{k+i}, i = 1, \ldots, PH$ can be derived using the expectation of the posterior:

$$\hat{y}(t_{k+i}|t_k) = E[p(y(t_{k+i})|\boldsymbol{x}(t_{k+i}))], \forall i = 1, \ldots, PH.$$

Implementation details are reported in in Appendix A.

5.3 Advanced black-box methodologies

5.3.1 Deep learning models

As discussed in Chapter 1, there are a growing number of deep learning applications in various research areas of T1D management. In particular, due to their ability to handle time-series and sequential data, there is an increasing trend to develop both recurrent and convolutional neural networks (RNN and CNN, respectively) for BG forecasting. Unlike traditional feed-forward neural networks in which the information moves from the input towards the output layer, RNN are characterized by recurrent units with loops allowing the information to propagate back to the same unit. So, each learning step takes into account not only the current input, but also what was learnt from the previous inputs [171]. Although CNN have been originally developed for image classification, it has been demonstrated that these models can automatically discover and extract deep features from time-series data by employing convolution and pooling operations to input data [172].

Some literature contributions, such as [69, 173], have assessed that the use of deep learning algorithms for the prediction of BG levels allows to achieve better performance than traditional methods. However, in general, the improvement in terms of RMSE with respect to baseline methods is usually about 2-3 mg/dL over longer prediction horizons (see for instance, [48, 78, 80]). Given this context, we investigated three multi-input deep learning models: LSTM, GRU and TCN. As shown in Figure 5.4, all the deep learning algorithms considered are fed by three input channels (past history of glucose data, meal intake, and insulin injections) and then directly output a sequence of 12 future consecutive glucose samples as described in [174].

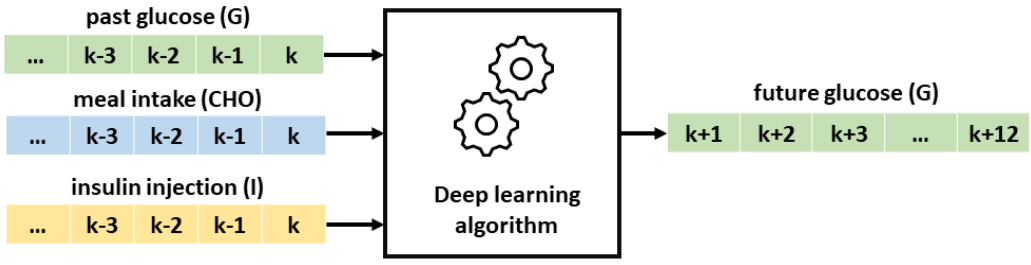


Figure 5.4: Schematic representation of BG forecasting as a sequence prediction task.

LSTM

Thanks to the use of memory cells, this model is able to learn and maintain the long and short term dependencies among the data and it overcomes the vanishing-exploding gradient problem that typically affects traditional RNN structures. The main difference between RNN and LSTM is related to the activation function used for computing the hidden state. In fact, RNN is composed by repeating modules consisting of a single layer with a tangetial activation function. On the contrary, the memory in LSTM is implemented as a cell in which a gate function decide whether the information should be kept or removed from memory at each time step [175]. Usually, this gate is a sigmoidal activation function coupled with pointwise multipliers. Following [69], we designed a network with a single LSTM layer composed by 30 LSTM nodes and a single output layer, with a number of neurons equal to the future glucose samples to be predicted. In this work the dense layer comprises 12 nodes, corresponding to a PH = 60 minutes (sampling time is 5 minutes).

GRU

GRU is a variant of LSTM which is designed to capture dependencies of different time scales. GRU has two gates (named reset and update gates) that control the flow of information without using the memory unit. In fact, the information stored in the internal cell state in an LSTM unit is incorporated into the hidden state of the GRU [176]. In particular, the update gate helps the model to determine which is the important past information (from previous time steps) that needs to be passed. This is the analogous of the output gate in an LSTM. The reset gate is then used to decide how much of the past information to forget. This is the combination of the input and forget gate in a LSTM. Finally, the current memory gate is incorporated into the reset gate, reducing the effect that previous information has on current information that is being passed into the future [176]. Summarizing, we can conclude that both LSTM and GRU have a very similar workflow but the main difference is in the internal working recurrent unit. Also for GRU, we designed a network with a single layer composed by 30 GRU nodes and a single output layer, with a number of neurons equal to the future glucose samples to be predicted, in this case 12 nodes.

TCN

CNN are commonly associated with image classification tasks. However, they can also be exploited for sequence modeling and forecasting. It has been shown that CNN can achieve better performance than RNN in many tasks while avoiding common drawbacks of RNN, such as the exploding/vanishing gradient problem. In particular, we focused on a TCN structure. A TCN is composed of three main structures that include causal convolutions, dilated convolutions, and residual connections [177]. Unlike simple causal convolutions, the dilated convolutions enable an exponentially large receptive field. In other words, this is similar to pooling or stride convolutions: the filter is applied over a region larger than its size by skipping input values with a given step. Finally, TCN employs a residual module (residual connections) containing a branch leading out to a series of transformations, whose output are added to the input of the block [177].

A detailed description of these deep learning methods is reported in Appendix B.

5.3.2 Linear non-parametric models

Although the metabolic physiology is nonlinear, the linear models tested in this work have demonstrated to achieve accurate performance in BG levels forecasting. For this reason, our analysis moved toward the use of advanced and innovative identification methods for linear models. A linear model of glucose-insulin dynamics can be written as:

$$\begin{gathered}\hat{g}(k|k-1,\theta) = \\ h_1 * i(k) + h_2 * m(k) + h_3 * g(k)\ ,\end{gathered} \tag{5.11}$$

where $*$ represents the convolution between two signals, whereas h_1, h_2, and h_3 are impulse responses related to the insulin, meal, and glucose signal, respectively. This is the equivalent formulation of the 1-step ahead predictor, as shown in 4.4.2. In this chapter, we focus on the so-called Stable Spline nonparametric (NP) identification approach which aims to estimate the unknown impulse response related to insulin, meal and glucose, denoted as h_1, h_2, and h_3 in 5.11, from noisy measurements. Unlike the traditional approach that constraints the unknown functions to a parametric structure, the nonparametric approach searches the unknown impulse responses over a infinite-

dimensional space given by a Reproducing Kernel Hilbert Space (RKHS). Such a space is completely specified by the choice of the kernel which incorporates prior knowledge, such as smoothness and stability of the predictor impulse responses to be estimated.

A detailed description of the nonparametric identification method is reported in Appendix C.

5.4 Dataset

As in chapter 3, the dataset used in this analysis is the OhioT1DM dataset [1], which comprises 12 subjects with T1D monitored with a Medtronic Enlite CGM system for 8 weeks. Participants wore an insulin pump (Medtronic 530G or 630G) and a wearable system (Basis Peak fitness or Empatica Embrace). In addition, subjects reported information on meals: timing, amount, and type (i.e., breakfast, lunch, dinner, snack, hypoglycemia treatment). Each subject in the OhioT1DM data set is split into a training set (about the intial 6 monitoring weeks) and into a test set (roughly the last 10 days). Handling data recorded under free-living conditions raises some technical issues. In particular, the OhioT1DM dataset presents long portion of missing CGM readings and the sampling time is not homogeneous. Therefore, all signals were aligned into a uniform time grid with a sampling period of $T_s = 5$ minutes. Any CGM gap in the training set shorter than 30 minutes was interpolated with a first order polynomial while a simple and causal zero-order-hold imputation was performed on the test set.

5.5 Predictive performance

Table 5.2 details the predictive performance for PH = 30, 45 and 60 minutes (in terms of RMSE and TG) achieved by the white-box physiological model (hereafter labeled as PHY), the deep learning models (i.e., LSTM, GRU and TCN), and the linear NP approach. As described in Chapter 4, statistical significant differences (pvalue < 0.05) between model performances, has been assessed by performing preliminary test for normality using the Lilliefors test, then by using a paired t-test if normal-distributed data, or a Wilcoxon signed-rank test when normality is rejected. From Table 5.2, three main outcomes can be observed: i) black box algorithms outperform the PHY in terms of RMSE, for

all prediction horizons, ii) all methodologies achieve similar TG, and iii) that there are no large differences in terms of RMSE between the considered black-box approaches. Particularly, comparing the RMSE results achieved by PHY vs. the other competing methodologies, RMSE roughly increases by 5 mg/dL, 6 mg/dL, and 9 mg/dL for PH = 30, 45, and 60 min, respectively, indicating that PHY appears to be a consistently worse candidate to be adopted for glucose prediction. As long as deep learning approaches are considered, for PH = 30 minutes, LSTM grants a median RMSE = 19.75 mg/dL similar to GRU (median RMSE = 19.81 mg/dL) and slightly inferior to TCN which provides a median RMSE = 20.11 mg/dL. Similar results also hold when evaluating performance for longer PH: median RMSE is about 27 mg/dL for PH = 45 minutes, and about 32 mg/dL for PH = 60 minutes. Overall, the NP approach allows to achieve the lowest RMSE results. In details, NP provides a median RMSE = 18.99 mg/dL, RMSE = 25.72 mg/dL and RMSE = 31.60 mg/dL for PH = 30, 45 and 60 minutes, respectively. Remarkably, the improvement in performance is found to be statistically significant with respect to LSTM and GRU (p-value = 0.04 and p-value = 0.03) for PH = 30 minutes, while no significant difference is found for longer prediction horizons. Furthermore, NP model showed to be significantly different to TCN, with p-value = 0.001, p-value = 0.006 and p-value = 0.009, for PH = 45 and PH = 60 minutes, respectively. Also, the NP approach achieves the largest median improvement with respect to PHY: 26.5%, 26.2% and 24.9%, for PH = 30, 45 and 60, respectively (p-values $< 10^{-4}$, for all the considered PH). The numerical results achieved by LSTM, GRU and TCN are inline (or even slightly better) with what has been obtained in other literature contributions dealing with the assessment of individualized deep learning algorithms for BG forecasting in the OhioT1DM dataset as in [68, 80, 178]. As a matter of fact, in [80] a recurrent convolutional network granted a mean RMSE = 20.6 mg/dL, 26.8 mg/dL and 33.9 mg/dL for PH = 30, 45 and 60 minutes, respectively. Similarly, the TCN and the LSTM tested in [178] achieved a mean RMSE = 20.23 mg/dL and RMSE = 20.11 mg/dL for PH = 30 minutes and RMSE = 34.21 mg/dL and 33.10 mg/dL for PH = 60 minutes. To better understand the reason of such a consistent difference between PHY and black-box models, we analyzed in details their "behaviour" when predicting the glucose time-course in two specific, but representative, one-day-long time windows extracted from the dataset at hand.

This is shown in Figure 5.5, where, in the top panel CGM data (grey dashed line) of a subject of the OhioT1DM dataset (ID:570), is overposed to the 30-

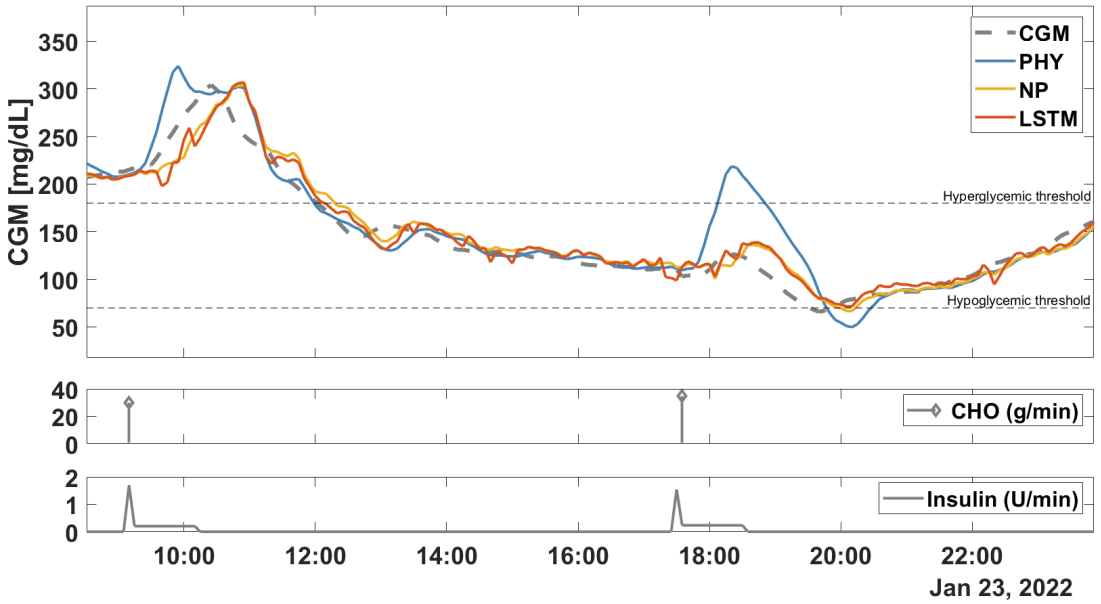


Figure 5.5: Representative subject (ID:570) of the OhioT1DM dataset. The upper panel shows CGM data (grey dashed line) and the 30-min ahead prediction obtained by: PHY (blue line), NP approach (yellow line) and LSTM (red line). Middle panel shows the CHO content of the meal, expressed as g/min. Bottom panel shows injected insulin boluses, expressed as U/min

minute ahead-in-time prediction of PHY, NP, and LSTM models, in blue, yellow and red, respectively. Middle and bottom panels show meal and insulin data, respectively. Analyzing the extracted portion of data, there are two meals with similar amount (30 g/min at 9:10, 35 g/min at 17:35) and two similar corresponding insulin boluses delivered with the so-called dual-wave mode [179], that consists of "splitting" the total amount of insulin bolus necessary to control the postprandial glucose excursion in a combination of two components (spike + square-wave). It is interesting to note that the two corresponding postprandial responses are very different. In the first case, glucose increases from 200 mg/dL to 300 mg/dL within an hour after the meal, whereas, in the second case, glucose remains almost flat (about 20 mg/dL excursion) and then decreases after an hour, reaching hypoglycemia. So, the first postprandial excursion is aligned with the physiological expectation that a meal should be followed by a glucose increase, while in the second postprandial excursion this does not happen. The (unavoidably simplified) physiological model structure has not the flexibility to cover both types of responses and the model imposes a similar shape to the two predicted postprandial glucose excursions leading, as a consequence, to an extremely large prediction error observed during the second meal. On the contrary, the black-box approaches prove more flexible

and are able to produce two different postprandial shapes despite the similar inputs. In details, considering the prediction results obtained with NP and LSTM during the first postprandial window (from 9:10 to 12:10), it can be observed that there is a huge delay, almost equal to the considered PH, between the predicted glucose traces and the target glucose data during the upward trend. In contrast, during the downward trend, good predictions are obtained for both methodologies considered. Instead, considering the second postprandial period (from 17:35 to 20:35), good results are achieved in the first phase (just after meal intake), while the predicted glucose traces result delayed, up to PH minutes, in the second phase (during the downward trend).
The example reported in Figure 5.5 sheds light on what seems to be a limitation of the proposed white-box methodology when used for glucose prediction: the modeled input-output relationship, defined by the underneath glucose-insulin description, largely constraints the impact of carbohydrate and insulin inputs on the predicted trace. Roughly speaking, this means that it is very difficult (or even impossible) for the physiological model to predict very different glucose responses given similar input data, which can occur in real life due to unmodeled phenomena, for example, stress, illness, or physical exercise. On the other hand, the fact that black-box models are fully data-driven, thus not necessarily representing the actual input-output physiological relationship between insulin/carbohydrate and glucose, allows them to be a more flexible approach to handle unexpected, real-world dynamics.
Stressing this point even further from the perspective of data quality, having missing/unannounced meals/boluses, or large carb-counting errors, would potentially represent a further non-negligible problem for white-box prediction approaches.

This is shown in Figure 5.6, representing subjects 552 of the OhioT1DM dataset. As before, the grey dashed line in the top panel is the CGM trace. The blue, yellow and red lines are the 30-minute ahead-in-time predictions of the physiological, the nonparametric and the LSTM model, respectively. Finally, middle and bottom panel show meal and insulin information, respectively. Analyzing this portion of data, although there are three recorded insulin boluses (at 8:45, 9:10, and 12:55), the subject did not report any information about meals. As obvious, this issue constitutes a challenge for all the algorithms that must rely on insulin data only in order to predict postprandial BG concentrations. Considering the prediction results obtained with NP and LSTM, such a missing information lead to a huge delay (almost equal to the considered PH)

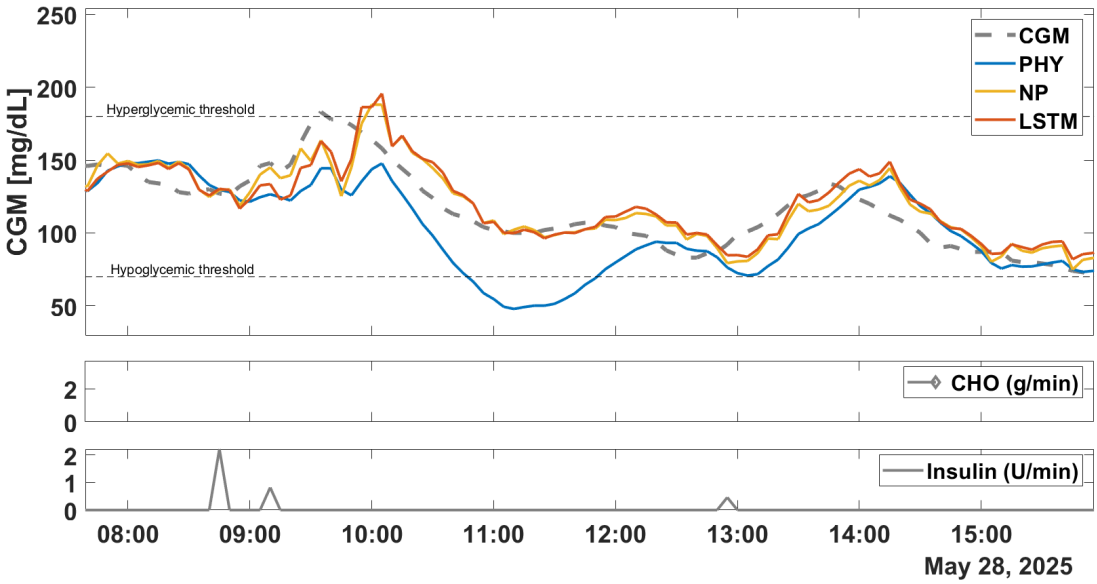


Figure 5.6: Representative subject (ID:552) of the OhioT1DM dataset. The upper panel shows CGM data (grey dashed line) and the 30-min ahead prediction obtained by: PHY (blue line), NP approach (yellow line) and LSTM (red line). Middle panel shows the CHO content of the meal, expressed as g/min. Bottom panel shows injected insulin boluses, expressed as U/min

between the predicted glucose traces and the target glucose data, which, as expected, is more pronounced when meal information is presumably missing and during the postprandial time window (from 9:00 to 11:00, and from 13:00 to 15:00). Observing the results obtained with PHY, as visible in Figure 5.6, the predicted glucose trace deviates significantly both from the target glucose data and from the other predicted traces in output of NP, and LSTM. In particular, it is clear how insulin inputs causes a drop in BG predicted levels, which is not what really happened in the actual glucose concentration, that, on the contrary, rose probably because of some carbohydrates have been ingested by the subject in correspondence of the insulin boluses recorded in this portion of data.

Methods	RMSE [mg/dL]			TG (min)		
	PH = 30 min	PH = 45 min	PH = 60 min	PH = 30 min	PH = 45 min	PH = 60 min
PHY	24.00 [20.77-27.14]	33.03 [29.17-36.06]	41.14 [36.85-44.28]	15 [12.5-20]	22.5 [15-35]	30 [22.5-50]
LSTM	19.75 [17.82-22.01]	26.80 [24.55-30.30]	32.54 [30.08-37.07]	15 [12.5-22.5]	20 [15-25]	25 [22.5-32.5]
GRU	19.81 [17.75-21.93]	26.77 [24.62-29.83]	32.78 [30.31-36.67]	17.5 [12.5-20]	20 [20-25]	25 [25-32.5]
TCN	20.11 [17.97-22.39]	27.28 [25.46-30.47]	33.54 [30.83-37.48]	15 [10-20]	20 [17.5-25]	25 [20-32.5]
NP	18.99 [16.90-21.65]	25.72 [23.44-30.52]	31.60 [28.99-36.82]	15 [10-20]	20 [10-22.5]	20 [15-27.5]

Table 5.2: Comparison between performance metrics (median [25$\hat{\text{th}}$-75$\hat{\text{th}}$]) obtained using the proposed physiological model (PHY), LSTM, GRU, TCN and NP models in the OhioT1DM dataset for PH = 30, 45, 60 min.

5.6 Summary of the main findings

This chapter detailed a comparison between advanced black-box algorithms and a white-box approach for BG forecasting, for different PH. To this end, starting from the maximal nonlinear model of glucose-insulin dynamics implemented in the Uva/Padova T1DS, we derived a novel simplified variant. The proposed physiological model is handled in a Bayesian framework consisting of two main phases. In the first phase, we use MCMC to identify patient-specific parameters from past collected data (i.e., CGM, CHO and insulin). Then, in the second phase, the personalized model is used within a PF to predict BG levels. For what it concerns the black-box methods, we implemented three promising deep learning approaches for time series forecasting: LSTM, GRU and TCN. Also, we considered the use of advanced linear NP models that have shown encouraging results in previous works. As for the physiological models, the data-driven methodologies are fed by CGM, CHO and insulin information. All the algorithms are trained and tested by exploiting the OhioT1DM data, a challenging dataset recorded in free-living conditions in an open-loop setup, where meal information is self-reported by participants. Table 5.2 showed that the considered black-box methodologies significantly outperform the representative white-box approach for all the PH under study. Moreover, among the data-driven algorithms, the best performance are achieved by the linear NP approach, by granting statistically significance difference to LSTM and GRU for PH = 30 minutes, and to TCN for all the considered PH. One possible reason for the differences in performance between white-box and black-box models might reside in the fact that the first are less flexible in accommodating the large variety of patterns observed in the data and that might be caused by multiple unmodeled factors, including variability in meal absorption, different meal compositions, stress, illnesses, physical activity, inaccuracy in estimating carbohydrate content of a meal. Future work will attempt to increase the flexibility of white-box models. This will include considering time-varying model parameters estimated in real-time by the PF, to track patient-specific intraday variability and meal-to-meal differences in CHO absorption in order to capture input-output relationships such as the ones presented in Figure 5.5 and Figure 5.6. Furthermore, to better represent real-world data, we will explore the potential of expanding the proposed model structure by integrating new subsystems describing the impact of carb-counting error [137] and physical activity [180] on glucose concentrations.

One of the most appealing application of predictive algorithms concerns the possibility of integrating them into closed-loop algorithms to automate insulin delivery and/or into decision support systems to suggest preventive corrective actions. However, when handling black-box predictive models (as the ones tested in this chapter), it is important to verify that they comply to the underlying rules of glucose-insulin regulation to enable safe and reliable therapeutic interventions. Indeed, verifying that such a requirement holds, is necessary to guarantee that unwanted and potentially dangerous, control actions are suggested to the patients due to "non-physiological" predicted glucose responses. For this reason, the next chapter will focus on the crucial role of interpretation in the usability of black-box predictive algorithms integrated into decision support systems for preventive suggestions of corrective insulin boluses.

Chapter 6

The importance of interpretability in BG prediction algorithms: an analysis using Shapley addictive explanation

[1] In the previous chapter we detailed a comparison between a nonlinear physiological model (obtained as a simplified version of the Uva/Padova glucose-insulin model) and advanced black-box algorithms for BG prediction. Results have shown that: i) black-box models outperform the proposed white-box models and ii) no large difference in performance can be noticed among the deep learning models, especially if RNN are considered. Black-box models present a useful tool for learning the complex glucose-insulin dynamics from input-output relationship, and are increasingly being considered as components of advanced tools for T1D management such as DSS or closed loop systems. However, in view of employing them to suggest safe therapeutic actions (such as the administration of rescue carbs or insulin boluses), black-box algorithms should be in line with the physiological laws that underlie the glucose-insulin metabolism. As a matter of fact, imagine a model that under some conditions predicts that insulin increases blood glucose levels: such a model could result in a controller increasing insulin infusion when low BG levels are predicted. Of course, this is potentially dangerous for the patient. In this chapter, we introduce Shapley addictive explanations (SHAP), a new tool

[1] This chapter contains material to be submitted for publication as *Prendin et al., "The importance of interpretability in machine-learning models for the real-time prediction of future glucose concentration in diabetes: an analysis using SHAP", Scientific Report.*

for interpreting black box models, and we design a case of study in which two LSTM, one with a non-physiological and the other with a correct physiological interpretation of the input, are used to suggest preventive insulin boluses. SHAP reveals that only one of these algorithms is in line with the physiological response of glucose-insulin interactions, leading to suggest safe therapeutic actions. Of note, given the proof-of-concept nature of this work, we focused on a single subject of the OhioT1DM dataset.

6.1 Rationale for the investigation of interpretable black-box models for glucose prediction and chapter content

6.1.1 Chapter contribution

As for application of black-box modeling, a key problem with the use of machine and deep learning approaches concerns the interpretability of the outcome. Interpretability represents the degree to which humans can understand the logic beneath a model decision [181]. Whilst machine/deep-learning models can grant accurate performance, as shown in previous chapters, their results can be difficult for users to explain and it is often impossible to unveil hidden biases in the datasets or to identify model weaknesses without understanding the decision-making process [182, 183]. Moreover, the lack of transparency in their inner logic may hamper the use of these models by arising (legitimate) questions on their trustability and safety.

Modeling BG dynamics makes no exception. Most of the available datasets collected on individuals with T1D present a strong collinearity between insulin administration and carbohydrates intake. In fact, insulin boluses are commonly administered when meals are consumed (the so-called prandial insulin boluses) and the dose of prandial insulin boluses is almost proportional to the quantity of ingested CHO. As a result, the learning process may sometimes fail and completely misunderstand the effect of these inputs on BG concentration. In other words, the model is not able to discriminate the effect that insulin and CHO have on glucose concentrations. In the worst case, a model could learn that the effect of insulin is to increase BG levels, while the effect of CHO is to decrease BG concentrations. If an incorrect model is then "actively" used within closed-loop systems or DSS, it could suggest rescue carbs to lower high

BG levels or insulin injections to higher low BG levels, by leading to potentially harmful situations. In this context, it is important to check whether the trained model is inline with the physiolgical glucose-insulin dynamics before embedding into tools for T1D management.
In recent years, many tools have been developed for providing an interpretation to black-box models and possibly unveiling problems in the learning process. Some examples are SHapley Additive exPlanation (SHAP) [184], Local Interpretable Model-agnostic Explanation (LIME) [185], Deep Learning Important FeaTures (DeepLIFT) [186], Model Agnostic Concept Extractor (MACE) [187] and Generative Adversarial Network (GAN) based methods [188]. These methodologies make the algorithms' predictions individually comprehensible by providing a description of how much each input contributed to the models' output. Interestingly, despite the large number of contributions investigating the use of machine learning and deep learning for BG prediction [50], [78], only a few of them deal with the interpretability of the models [189, 190, 191] and with their conformity to the physiological glucose-insulin response [67]. This work aims to demonstrate the crucial role of interpretation for the usability of black-box models for BG prediction, especially when these models are employed to suggest corrective actions to the user. In this study, model interpretation will be provided by SHAP, a game theoretic approach to explain the output of any machine learning model.
The case study we designed consists in training two LSTM models for BG prediction, such that they: i) are fed by the same features; ii) are based on a similar structure, and iii) provide similar prediction accuracy. By doing so, it is difficult to claim the superiority of one model with respect to the other and it is not possible to understand whether or not an algorithm can provide safe suggestions once embedded into a DSS. However, SHAP reveals that one of these models is unable to discriminate the correct effect of insulin on BG levels, and could potentially lead to unsafe or clinically harmful suggestions. Then, the interpretation provided by SHAP is retrospectively validated using the two LSTM within a simple DSS that suggests corrective insulin boluses. This further assessment shows that only the model with a correct, physiological interpretation of the features is able to suggest the adequate amount of corrective insulin boluses. Such an insight can only be gained via tools like SHAP, which can therefore guide researchers towards the choice of the best algorithm for what it concerns therapy and safety.

6.1.2 Chapter outline

This chapter is organized as follows. Section 6.2 illustrates the novel tool for the interpretation of black box algorithms and it provides a practical example by resorting to a simple linear model. Section 6.3 presents the core of this Chapter: the case of study aiming at demonstrating the key role of interpretability when using BG prediction algorithms to suggest corrective actions. Section 6.3.3 describes the DSS we developed to suggest preventive insulin boluses based on LSTM predictions. Section 6.3.4, briefly describes the tool we used to retrospectively assess our algorithms on real data. Finally, Section 6.4.1 details the predictive performance of the two proposed LSTM, Section 6.4.2 illustrates the interpretation provided by the two networks and Section 6.4.3 shows the results in terms of glycemic control metrics obtained by using the two LSTM.

6.2 SHapley Addictive exPlanation (SHAP)

Because of their complexity, LSTM are not inherently interpretable. Nevertheless, several techniques are available for interpreting machine and deep learning models. The interpretation tool we adopted in this work is SHapley Additive exPlanation (SHAP), an approach based on Shapley values that can potentially explain the output of any machine learning model [184].

The Shapley value method comes from the game theory field. It considers a cooperative game with M players and assumes to have the contribution function $v(S)$ that describes the total expected sum of payoff obtained by a subset of players (S). Shapley values fairly distribute the total gain of the game between players. The amount of gain a player j receives is given by

$$\phi(v)_j = \phi_j = \sum_{S \subseteq M \setminus j} \frac{|S|!(|M| - |S| - 1)!}{|M|!} [v(S \cup j) - v(S)] \tag{6.1}$$

where $|\cdot|$ is the cardinality of the set and the difference $v(S \cup j) - v(S)$ indicates the additional contribution that player j gave to the subset S. Equation 6.1 defines the Shapley value ϕ_j assigned to player j as a weighted mean of its additional contributions $v(S \cup j) - v(S)$ to each subset S not containing j ($S \subseteq M \setminus j$).

This game theoretical concept can be translated into the context of glucose prediction by understanding the parallelism between players/features and outcome of a game/model prediction.

For a better understanding, let us consider a trivial linear model for predicting blood glucose levels.

$$\hat{g}(k+PH) = f(g(k), i(k)) = \beta_0 + \beta_1 g(k) + \beta_2 i(k) \tag{6.2}$$

where $f(g(k), i(k))$ is the predictive model that outputs the predicted glucose level -i.e. $\hat{g}(k+PH)$- at a certain PH ahead in time, once it is fed by CGM, $g(k)$, and insulin, $i(k)$, at time k. As we are dealing with a linear model, β_j, with $j = 0, 1, 2$, represents the weight related to $j-th$ feature (i.e. the model coefficients).

In order to explain the model prediction $f(x^*)$, when the model is fed by a particular instance (denoted by *) of the feature vector $x = x^* = [g(k^*), i(k^*)]$, we have to define a contribution function $v(S)$, for all possible subset of feature. To quantify this, Lundberg & Lee suggest to use the conditional expectation of the predictive model given a set of feature, i.e $E[f(x)|x_S = x_S^*]$. Following [192], one can found that the contribution of glucose and insulin at time k to the model $f(g(k), i(k)) = \hat{g}(k+PH)$ is given by:

$$\phi(g(k)) = \beta_1 g(k) - E[\beta_1 g] = \beta_1 g(k) - \beta_1 E[g] = \beta_1 (g(k) - E[g]) \tag{6.3}$$

$$\phi(i(k)) = \beta_2 i(k) - E[\beta_2 i] = \beta_2 i(k) - \beta_2 E[i] = \beta_2 (i(k) - E[i]) \tag{6.4}$$

where $E[g]$ and $E[i]$ are the expected value of glucose and insulin. Considering this result, a Shapley value -for a given feature value- can be seen as the deviation of the feature from its mean value, multiply by its weight.

SHAP is a model-agnostic tool and it can be applied to any machine and deep learning method ranging from tree-based models to recurrent neural networks: SHAP unifies several different explanation methods (LIME and Shapley values in the KernelSHAP, DeepLift and Shapley values in DeepSHAP) [184].

In this work, we resort to SHAP values to obtain an insight about how (and how much) each feature (i.e. CGM, insulin and CHO) contributes to the predicted glucose level. To compare the two model interpretations, SHAP values are visualized through the summary plots which provide a global overview of the explanations for a set of data, see for instance Figure 6.2. In particular, a summary plot details the SHAP value of each individual feature for every sample in the dataset. Each row represents a feature, and all the features of the predictive models are ranked according to their importance on the y-axis.

The dots in each row represent the samples for a specific instance. The color of the point represents the feature value, so it indicates whether that point is associated with a high (red) or low (blues) value of the feature, with respect to its mean value, i.e. how each feature contributes to model output. A positive SHAP value indicates that the feature positively impacts in model output. Finally, for each feature, the width of the plot in a specific position of the x-axis represents the density of the samples associated with that SHAP value.

6.3 The case of study: BG prediction algorithms to preventive insulin boluses

The aim of this proof-of-concept study is to verify that, given two competitor algorithms with similar performance, SHAP allows to correctly identify the one with the physiological interpretation, that should be used for decision-making and control aims. Therefore, in addition to SHAP, the following key elements are required.

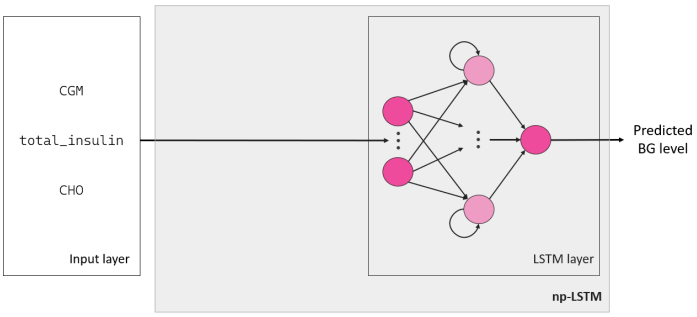
6.3.1 Dataset and preprocessing

This case-of-study is focused on a single subject (ID 588, female, age 40-60) selected from the OhioT1DM dataset [1].

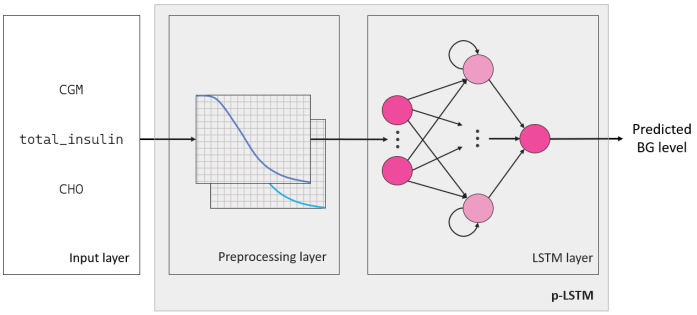
Patient data are split into a training set, consisting of the first 6 weeks of data, and a test set, which includes the last 10 days of data. The training set is used to train the two LSTM, while the test set is used to compute the prediction accuracy of the models. As a final step, to retrospectively evaluate the insulin corrective actions suggested by the model-based DSS, we selected a subset of the test set consisting of 8-hour postprandial windows starting with a meal consumption. Furthermore, only periods that met the following conditions were selected: i) no other CHO were consumed in the following 8 hours after meal intake; ii) no correction boluses were present in the original dataset in the following 8 hours after the prandial insulin bolus.

6.3.2 The black-box predictive algorithms

As described in Chapter 4, LSTM represents a suitable choice for time series prediction since they can learn and maintain long and short-term dependencies from data. This kind of network falls within the category of RNN, but it



(a) np-LSTM



(b) p-LSTM

Figure 6.1: Schematic overview of np-LSTM (a) and p-LSTM (b). The only difference between the two structures is the preprocessing layer in (b), which is used to enforce a physiological interpretation in the LSTM from insulin and CHO.

overcomes the issue of vanishing-exploding gradient which affects deep RNNs during the training phase. Core elements of an LSTM are the gates (forget, input and output) which compose the so called memory cell. At each time step, these gates decide whether the incoming information is useful or if it must be erased from the cell. Further details are available in Appendix B.

We designed two ad-hoc black-box glucose predictive algorithms based on LSTM with the same structure (a single layer of 64 LSTM units). In the former, called non-physiological LSTM (np-LSTM), features are straightforwardly fed into the network. Hence, no specific measure is taken to enforce a physiological interpretation of the inputs. In the latter, called physiological LSTM (p-LSTM), a pre-processing layer is interposed between the input and LSTM layer to help the model understanding the correct effect of insulin and CHO. The features employed in the models are: current CGM measurements, `CGM` (mg/dL); current insulin administration, `total_insulin` (U/min); and current CHO consumption, `CHO` (g/min).

As shown in Figure 6.1, the main difference between the two algorithms is represented by the preprocessing layer embedded within p-LSTM. This pre-

processing non-learnable layer, which is placed between the input and the LSTM layer, consists of two filters applied to the `total_insulin` and `CHO` features. These masks resemble the physiological absorption curves characteristics of insulin and CHO, similarly to what described in [193], [194]. With these filters, we aim at uncoupling the effects of `total_insulin` and `CHO` by shifting their time-action profile on future BG.

The two models are trained to predict the BG levels ahead in time for two different PH, i.e., 30 and 60 minutes. Once trained, these models are i) applied on the test set to evaluate their prediction accuracy; ii) interpreted with SHAP to assess whether or not they correctly explain the output; iii) embedded into a DSS and tested in simulated decision-making scenario.

The performance of the prediction algorithms are evaluated by using two standard metrics. The first is the mean absolute error (MAE), which is defined as

$$MAE = \frac{1}{N}\sum_{t=1}^{N} |y(t+PH) - \hat{y}(t+PH|t)|, \tag{6.5}$$

with N being the number of evaluated samples, y being the true output and $\hat{y}$ being the model prediction. The second metric we used is the RMSE. In general, the smaller MAE and RMSE, the better the algorithms capabilities to forecast glucose levels. To evaluate the quality of the LSTM-based DSS we considered the percentage of CGM samples in different glycemic ranges: within the normoglycemic interval 70-180 mg/dl, called time-in-range (TIR); below 70 mg/dl, called time-below-range (TBR), above 180 mg/dl, called time-above-range (TAR). While the TIR should be maximized, TBR and TAR should be minimized to avoid short- and long-term consequences of T1D, respectively. We also considered the amount of suggested boluses and their amount. We would like to minimize these numbers in order to reduce the burden on patients and to avoid possibly risky insulin overload in the organism.

6.3.3 Preventive correction insulin boluses

As the core of the DSS, the model predictions are exploited to find the optimal corrective insulin bolus. In details, the DSS reasoning works as follows. The algorithm is triggered two hours after a meal intakes and suggests a corrective insulin bolus if BG>180 mg/dL. The bolus dose is chosen as the $i_n \in \{i_1, ..., i_N\}$

that minimizes the following cost function:

$$J = (\hat{g}_n(k + PH|k) - g_0)^2 + 10 \cdot i_n^2 \tag{6.6}$$

where k is current time, $\hat{g}_n(k + PH|k)$ is the PH-step ahead prediction provided by the LSTM model when fed by the CGM and CHO values at instant k and by the insulin dose i_n; g_0 is a target glucose value (set at 120 mg/dL). The first term of (6.6) penalizes the distance of the predicted glucose value from its basal value; the second term penalizes the amount of candidate insulin boluses. The minimization problem is solved via grid search in the finite grid of solutions $\{i_1, ..., i_N\} \in \mathbb{Q}$.

6.3.4 Retrospective assessment on real data

To retrospectively evaluate the goodness of the corrective actions suggested by the LSTM models, we resort to a novel in-silico methodology recently developed by our research team, named ReplayBG [165]. The in-silico framework is based on the physiological model proposed in Chapter 5, here used for simulation aims. Briefly, it consists of two main phases: in the first step, the nonlinear physiological model of glucose-insulin dynamics is identified using a MCMC approach for each selected portion of data, as described in Chapter 5 and in Appendix A. Then, the identified model is used to simulate the postprandial glucose concentration that would have been obtained by adopting the corrective insulin boluses suggested by the predictive algorithms. As a remark, the core of this simulation tool is the minimal model described by Bergman et al. [84], which described the action of plasma insulin on plasma glucose. Such a model has been generalized by adding a model of subcutaneous insulin infusion (describing how exogenous insulin diffuses through plasma [195]) and a model of oral glucose assumption (describing how carbohydrates influence BG [167]).

6.4 Results

The following sub-paragraphs describe point-to-point the results in terms of predictive performance, interpretation and glycemic control.

Table 6.1: Mean (± standard deviation) of MAE and RMSE, computed over 10 different initialization and evaluated on the test set for np-LSTM and p-LSTM with PH of 30 and 60 minutes.

Model	MAE (mg/dl)		RMSE (mg/dl)	
	PH=30 min	PH=60 min	PH=30 min	PH=60 min
np-LSTM	15.20 (±0.05)	23.68 (±0.02)	21.43 (±0.06)	33.16 (±0.06)
p-LSTM	15.44 (±0.1)	23.88 (±0.03)	21.67 (±0.1)	33.45 (±0.06)

6.4.1 Predictive performance

Table 6.1 reports the results obtained with the two LSTM-based models for PH = 30 and 60 min. Because the random initialization of the two LSTM could potentially affect the results, the metrics in Table 6.1 are reported as mean (± standard deviation) computed over 10 different evaluation runs.

Considering the RMSE for a 30-minute prediction horizon, the two algorithms provide similar performance, although a slight improvement in RMSE is granted by the np-LSTM (21.43 mg/dL) with respect to the p-LSTM (21.67 mg/dL). Similarly, for PH = 60 minutes the np-LSTM performs slightly better than p-LSTM (RMSE = 33.16 mg/dL and 33.45 mg/dL, respectively). In both cases, the difference in the performance is less than 1 mg/dL. This was expected, as the only difference between the two architecture consists in the preprocessing layer. Similar considerations can be drawn by looking at the MAE. The np-LSTM provides the best results both for 30-minute and 60-minute PH, even though the performance gap is still small. These metrics are in line with the most of the literature works [68, 98, 196].

6.4.2 Interpretability of the models

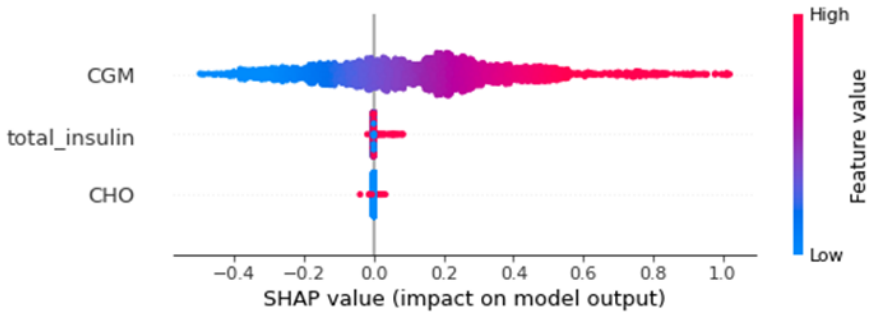
Figure 6.2 shows the summary plots corresponding to np-LSTM and p-LSTM with PH of 30 and 60 minutes. Figure 6.2a shows the summary plot for the np-LSTM for 30-minute PH. Such graphical visualization of the SHAP values reveals that the most important feature is `CGM`. Moreover, it returns that the `total_insulin` feature positively contributes to model prediction, even if it is a weak contribution; indeed, most of its SHAP values are positive. Finally, the less important feature is `CHO`: some values positively affect and some others negatively impact on model's predictions. For what it concerns the 60-minute np-LSTM (Figure 6.2c), the most important feature is `CGM`. Unlike the 30-minute np-LSTM the `CHO` feature is more important than `total_insulin`

and its contribution to model's output is completely positive, as well as that of total_insulin feature. The interpretation of both np-LSTM suggests that BG levels increase with insulin administration, which is the exact opposite of what happens from a physiological point of view. On the other hand, p-LSTM shows a correct, physiological interpretation of the model output, both for PH=30 (Figure 6.2b) and PH=60 min (Figure 6.2d). Again, the most important feature is found to be CGM. Both the summary plots then place CHO as the second most important feature and total_insulin as the last one. For both PH, CHO has mostly an increasing effect on the prediciton (i.e., SHAP values are positive). Most of SHAP values for total_insulin are negative, which means that insulin action decreases future BG values.

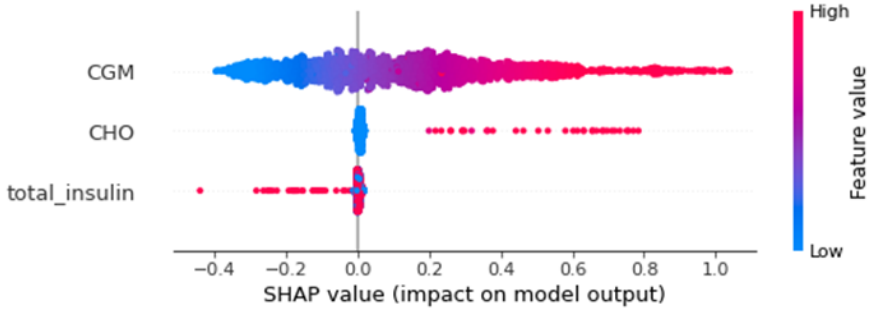
6.4.3 Corrective insulin boluses

For a given PH, the two LSTM structures (p-LSTM and np-LSTM) achieve similar prediction results. Therefore, one would expect these models to also achieve very similar results when used in a decision-making application. On the other hand, the interpretation provided by SHAP highlights significant differences in the learning process.
In the np-LSTM, positive values of total_insulin are associated with an increase of CGM levels. This behavior is non-physiological and, most likely, it happens because the model learnt the combined effect of insulin and CHO, instead of understanding their individual contribution.
The p-LSTM does instead learn the correct signs of these contributions: positive values of total_insulin lead to a decrease in CGM, whereas positive values of CHO have the opposite effect.

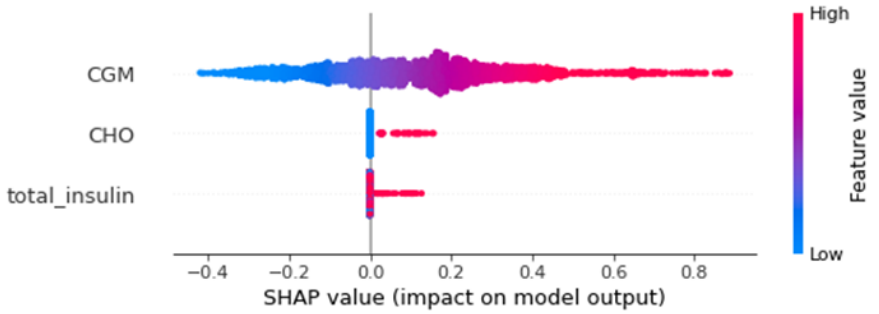
Figure 6.3 reports the results obtained with the two DSS in one representative postprandial window. In the top panel, we can see that the original CGM values (red dotted line) are outside the normoglycemic range (grey dashed lines). In particular, the postprandial hyperglycemic episode lasts for 6 hours at least. The blue and black dotted lines represent the simulated glucose profile which would have been obtained if the patient had followed the corrective actions suggested by the DSS (red arrows in the other panels). In the middle and bottom panel we can see in green the insulin bolus computed by the subject at meal time and the basal insulin (black line). In red, the suggested corrective insulin boluses. As we can see, there are no suggestions with the np-LSTM DSS (middle panel), while the p-LSTM DSS suggests two corrective boluses,



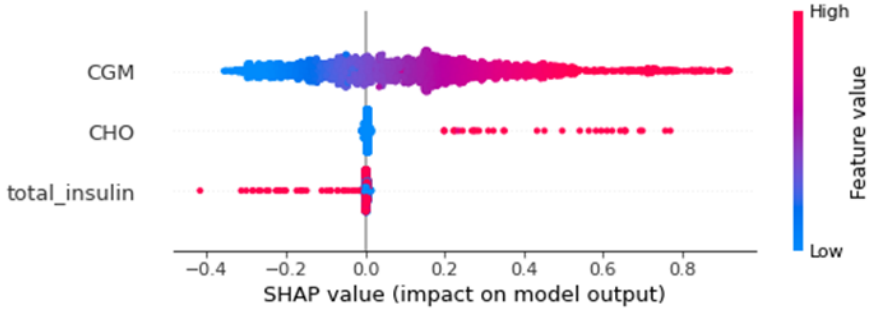
(a) np-LSTM, PH=30 min



(b) p-LSTM, PH=30 min



(c) np-LSTM, PH=60 min



(d) p-LSTM, PH=60 min

Figure 6.2: Summary plots of np-LSTM and p-LSTM for different PH.

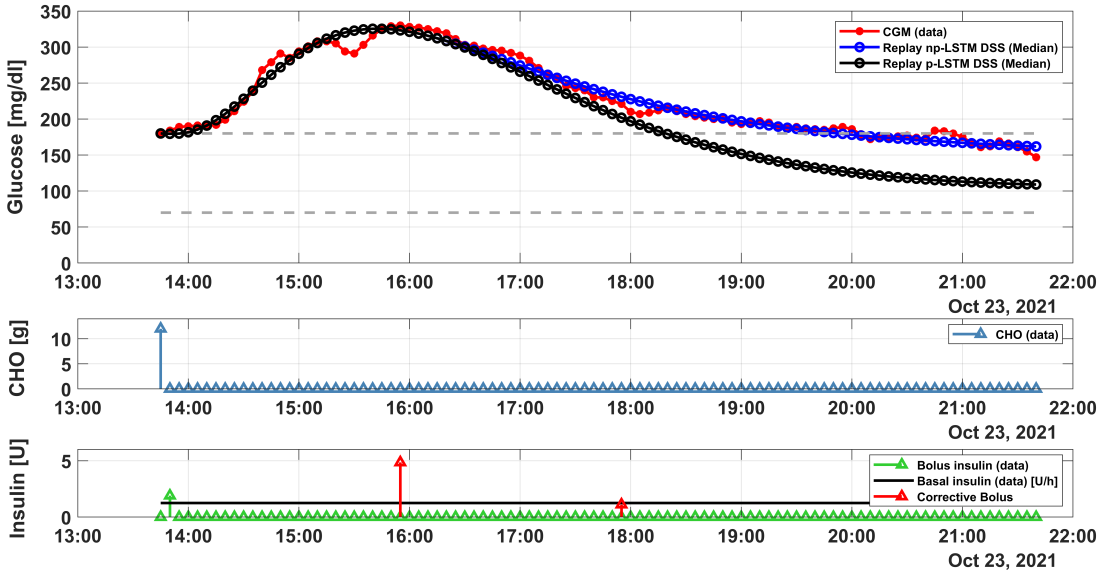


Figure 6.3: p-LSTM raises two corrective boluses that reduce the time spent in hyperglycemia, while np-LSTM does not suggest any corrective action.

one at 16:00 and one at 18:00 (bottom panel). This leads to different traces on the top panel, black dotted line for the p-LSTM and blue dotted line for the np-LSTM. Since no boluses are triggered by the np-LSTM, the blue dotted line is overlapped to the CGM readings. Differently, the black dotted line shows that the administration of the two corrective actions leads to a decreasing of the time spent in hyperglycemia (4 hours vs 6 hours) and a better glycemic control: from 18:00 to 21:00, the black dotted line is completely inside the target range while the blue line enters in the same region only after 20:00.

As shown in Table 6.2, the DSS based on np-LSTM does not suggest the administration of any insulin bolus both for 30- and 60-min PH, thus resulting in TBR, TIR and TAR equal to the ones obtained by the baseline (0%, 60.9% and 39.1%, respectively). Remarkably, the DSS based on p-LSTM suggests to the patient some corrective actions (1 insulin bolus in median for both 30- and 60-minute PH). All the evaluation metrics, for all the PH considered, benefit from the adoption of the DSS based on p-LSTM. Indeed, considering a 30-minute PH the TIR is increased (from 60.9% to 80.7%) and the TAR decreases (from 39.1% to 19.3%). Analogously, considering 60-minute PH, the TIR increases (from 60.9% to 79.2%) and the TAR decreases (from 39.2% to 20.8%).

Table 6.2: Results obtained without decision support (No DS) and with two CIB algorithms: one based on np-LSTM (np-LSTM DSS), the other based p-LSTM (p-LSTM DSS). Results are reported for a PH of 30 and 60 minutes. The results refer to the data windows satisfying the requirements described in Section 2.3, which are simulated using ReplayBG to perform this retrospective analysis. Results are reported as median [25^{th}-75^{th} percentiles] computed over these time windows.

PH	Therapy	TBR (%)	TIR (%)	TAR (%)	Insulin amount (U)	Number of boluses
30 min	No DS	0 [0-0]	60.9 [42.2-72.4]	39.1 [27.6-57.8]	-	-
	np-LSTM DSS	0 [0-0]	60.9 [42.2-72.4]	39.1 [27.6-57.8]	0 [0-0]	0 [0-0]
	p-LSTM DSS	0 [0-0]	80.7 [63.0-84.9]	19.3 [15.1-36.9]	4.3 [2.1-5.2]	1 [0.5-1.5]
60 min	No DS	0 [0-0]	60.9 [42.2-72.4]	39.1 [27.6-57.8]	-	-
	np-LSTM DSS	0 [0-0]	60.9 [42.2-72.4]	39.1 [27.6-57.8]	0 [0-0]	0 [0-0]
	p-LSTM DSS	0 [0-0]	79.2 [57.8-82.8]	20.8 [17.2-42.2]	2.4 [1.2-2.5]	1 [0.5-1.5]

6.5 Summary of the main findings

In this chapter we designed a case-of-study about the development of a DSS that suggests corrective insulin boluses based on predicted BG levels. Specifically, we considered two predictive algorithms based on LSTM, np-LSTM and p-LSTM, which rely on the same input features and the same structure. The only difference between the two is a non-learnable, pre-processing layer in p-LSTM, which is placed between the input layer and the hidden LSTM layer. Commonly, the key parameter for choosing one among many competing predictive models is prediction accuracy. Therefore, as described in Table 6.1, we evaluated the models ability to accurately forecast glucose ahead in time in terms of RMSE and MAE. Considering these numbers, it is not completely clear which model should be used in practice. Both networks provide similar results in terms of RMSE and MAE, with np-LSTM performing slightly better than p-LSTM. For this reason, we proposed an analysis with SHAP, a novel tool for the interpretability of black-box models. SHAP highlighted that only p-LSTM has learned a correct physiological explanation of the output. In contrast, np-LSTM has learned an incorrect interpretation of insulin and CHO as an effect of the collinearity between these two features. When looking at the summary plots, it is clear that the two LSTM work differently, even though they are fed by the same input features. In particular, in Figure 6.2, the `total_insulin` feature positively contributes to the model's output in np-LSTM, both for PH=30 min and PH=60 min. It means that np-LSTM will forecast an increase in BG levels after any insulin bolus. On the contrary, for the p-LSTM, both the `CHO` and `total_insulin` feature show a physiological behaviour: after a CHO intakes, it will forecast an increase in BG levels and, viceversa, after an insulin bolus the model will forecast a decrease in BG concentrations. In conclusion, by visual inspection of the summary plots, p-LSTM seems to be the most suitable model for decision-making aims.

To validate such a thesis, thanks to a simulation tool (ReplayBG) we applied both LSTM models to suggest corrective insulin boluses based on LSTM predictions. Table 6.2 confirms the conclusions drawn from the summary plots: the most suitable model for a DSS is the one which is in line with the physiological meaning, although the p-LSTM grants slightly lower prediction performance than np-LSTM.

Chapter 7

Conclusion and future work

Blood glucose forecasting is a relatively mature field that has received vast attention in the last 20 years within T1D research community for its potential to revolutionize diabetes care. In fact, the accurate forecasting of BG levels represents a key element for the development of next-generation tools for the management of T1D therapy, such as improved decision support and advanced closed-loop control systems. However, none of the literature works published so far have systematically studied how and/or how much different input information as well as simple and/or complex algorithms contribute to improve the performance of predictive algorithms on datasets recorded in daily-life conditions. From a practical perspective, understanding which signals positively (and how much) contribute to model performance can provide an insight about the information needs to be stored/memorized into devices for T1D management and which input is not necessary. In this work we faced the above-mentioned literature limitations by exploring both several input combination (CGM, CGM & seasonality/mealtime, CGM & meal & insulin) and a broad spectrum of black-box approaches that ranges from linear techniques, typically used in time-series analysis and system identification, to the non-linear approaches commonly adopted in machine learning and deep learning techniques. In addition, we developed a dedicated approach to address the problem of predicting hypoglycemic events. Moving from black-box towards the white-box approach, we investigated whether the use of a personalized physiological model can improve the predictive performance with respect to black-box strategies. Finally, we addressed the problem of lack of transparency in black-box algorithms, by introducing a state-of-art tool for model interpretability and by developing a case-of-study where predictive black-box models are used within a DSS to suggest preventive insulin boluses.

7.1 Summary of the thesis contributions

7.1.1 Chapter 2

Among the 30 glucose predictive algorithms tested in this chapter, the best results in terms of RMSE are achieved by individualized ARIMA and NN, no statistically significant differences are found. Of note, considering hypoglycemia prediction, individualized ARIMA achieved the best F1-score (72%). Results indicated that:

- individualized methods slightly outperform their population counterparts, confirming the positive impact of model parameter individualization, which allows customizing models for each single patient and dealing with the large variability in glucose profiles among individuals;
- linear methods are valuable options that offer a trade-off between complexity and performance, especially if PH $\leq$ 30 minutes are considered;
- the main limitations of all the CGM-only algorithms is that any metabolic disturbance, e.g. a meal, would deteriorate the accuracy of the predicted BG levels.

7.1.2 Chapter 3

In this chapter, we introduce the combined use of CGM and mealtime information through the assessment of C-SARIMA, a novel methodology that combines fuzzy C-Means clustering and seasonal local models. Results obtained on two different datasets indicated that:

- C-SARIMA methodology outperforms individualized ARIMA model for PH>45 minutes and NN for PH>60 minutes;
- no statistically significant difference between the results provided by C-SARIMA and the ones provided by individualized ARIMAX model (fed by CGM, CHO and insulin);
- the prediction of hypoglycemia poses a challenging task for all the algorithms presented so far and requires the development of dedicated approaches.

7.1.3 Chapter 4

In this chapter, exploiting an individualized ARIMAX model, we developed a novel approach to address the problem of an accurate prediction of hypoglycemic events. To this end, we presented two improvements:

- the use of novel cost function for model identification (*gMSE*), specifically designed to account for the clinical impact of prediction error;
- a novel alarm strategy that simultaneously considers multiple PH with their confidence intervals (prediction funnel).

The results showed that models identified through *gMSE* minimization provide better hypoglycemia prediction performances than models based on MSE. Furthermore, the new alarm strategy based on the prediction–funnel improves hypoglycemia forecasting, thanks to the possibility of exploiting multiple PHs. The adoption of both the proposed improvements grants the best performances.

7.1.4 Chapter 5

This chapter detailed a comparison between individualized black-box algorithms and a white-box one for BG forecasting, for different PH. To this end, starting from the maximal nonlinear model of glucose-insulin dynamics implemented in the Uva/Padova T1DS, we derived a novel simplified variant. For what it concerns the black-box methods, we implemented three promising deep learning approaches for time series forecasting: LSTM, GRU and TCN. Also, we considered the use of advanced linear nonparametric models that have shown encouraging results in previous works. Results showed that:

- black-box methods significantly outperform the proposed physiological model for all the PH;
- among the data-driven algorithms, the best performance is given by the linear nonparametric approach (statistically significant difference is found with respect to LSTM and GRU for PH = 30 minutes and with respect to TCN for all PH).

7.1.5 Chapter 6

In this chapter we proposed an analysis with SHAP, a novel tool for the interpretability of black-box models. We designed a case-of-study about the

development of a DSS that suggests corrective insulin boluses based on predicted BG levels. Specifically, we considered two predictive algorithms based on LSTM, np-LSTM and p-LSTM, which rely on the same input features and the same structure. SHAP reveals that np-LSTM is not able to discriminate the correct effect of insulin and CHO, while p-LSTM is in line with the physiological laws. As a matter of fact, once applied in simulations, only p-LSTM is able to provide safe and reliable corrective actions that reduces the time spent in hyperglycemia and increases the time spent in target.

7.2 Conclusion

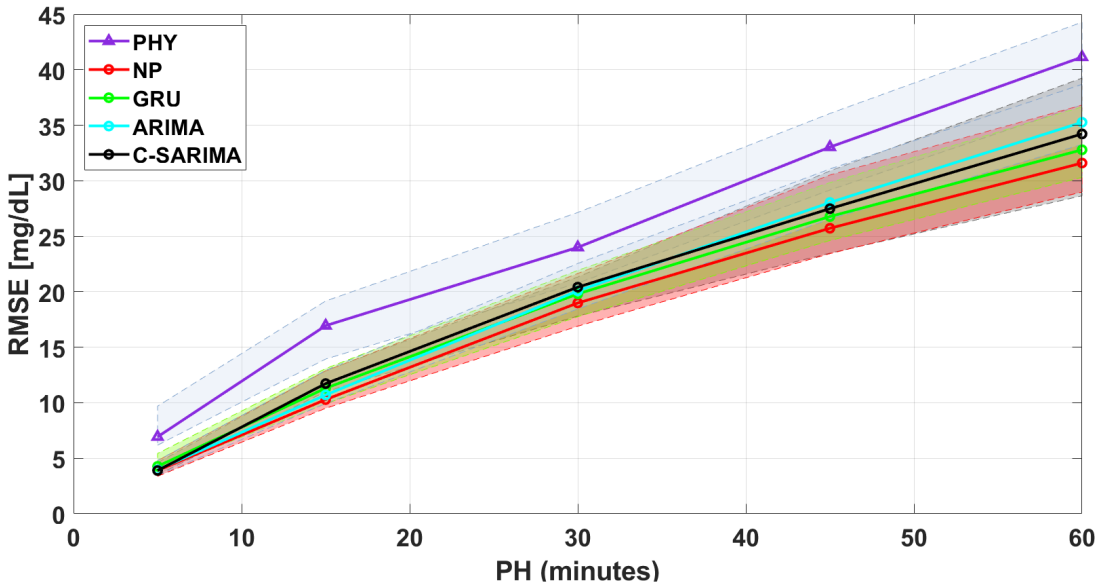


Figure 7.1: Results achieved by the main methodologies explored in this thesis on the OhioT1DM dataset. Performance are expressed as median RMSE (dashed lines) and [25^{th}-75^{th} percentiles] (shaded areas) for different prediction horizon.

As a final contribution, we would like to present some concluding statements about the role of different input information and the contribution brought by models of increasing complexity that may be useful for those who will develop algorithms for glucose prediction in T1D.
For such a scope, we have re-implemented the main algorithms presented in this thesis and assessed them on the same dataset (OhioT1DM) aiming to exclude possible confounding factors and to provide a fair and straightforward comparison. This is shown in Figure 7.1 which details the performance of: i) the best linear CGM-only algorithm found in Chapter 2 (individualized

ARIMA, cyan dotted line); ii) the novel methodology based on Fuzzy C-Means clustering and SARIMA models proposed in Chapter 3 (C-SARIMA [1], black dotted line); iii) the advanced linear non-parametric approach fed by CGM, meal and insulin information proposed in Chapter 5 (NP, red dotted line), as well as the deep learning network (GRU, green dotted line); finally, iv) the physiological white-box model inspired by the Uva/Padova T1DS (PHY, violet triangle line). Based on Figure 7.1, and supported by the main findings of each chapter, we can conclude that:

- as expected, the larger the PH, the larger the prediction error provided by all the algorithms;

- data-driven strategies outperform the proposed physiological white-box model, for all the PH;

- for PH $\leq$ 30 minutes, CGM only information employed as input of linear strategies (like individualized ARIMA), represent a practical valuable solution;

- the use of seasonality information (i.e., CGM + mealtime) improves significantly the performance for PH > 45 minutes if compared to CGM only algorithms;

- for PH $\geq$ 30 minutes, the use of meal and insulin information (timing and dosing) is required to enhance the overall performance;

- employing deep learning strategies allows to improve the performance with respect to state-of-art linear techniques, but it does not drastically change the overall picture;

- advanced linear non-parametric approach provides the best results, by demonstrating that linear assumption is a viable alternative to more complex nonlinear models;

[1] As described in Chapter 3, the proposed C-SARIMA methodology has been developed to predict only the postprandial periods. Therefore, for this conclusive analysis, which aims to compare the main methodologies on the entire glucose traces, C-SARIMA has been modified by adding to the seasonal local models an ARIMA predictor that forecasts glucose from the "end" of a postprandial period to the "beginning" of the next period.

7.3 Limitation of the study and future works

Among the limitations of this research, we can identify three main areas for future improvements. Concerning the forecasting of BG levels, we acknowledge that we have not addressed all the possible machine/deep learning approaches, but we mainly focused on the well-established nonlinear black-box models. For this reason, future work will deal with the assessment of novel powerful methodologies and with the exploration of the promising field of transfer learning. As far as the physiological white box model is concerned, we acknowledge that only a reduced version of the Uva/Padova T1DS has been proposed in this thesis. Future work will deal with the assessment of the complete physiological model integrated into the Uva/Padova T1DS. Regarding the prediction of hypoglycemic events, we applied the two proposed novelties only on linear ARIMAX models. Future work will deal with an investigation of the proposed novelties on nonlinear strategies. For what it concerns the different input data, we acknowledge that we are not taking into account the physical activity data, but we mainly focused on meals and insulin infusions. To this end, future work will deal with signals related to physical activity, such as heart rate variability data, recorded by modern dedicated devices. Finally, for what it concerns the black-box interpretability tools, we plan to extend our analysis to other approaches to investigate their potential in explain black-box model predictions.

Appendix A

Physiological model individualization and prediction

A.1 Bayesian identification approach: implementation details

We partitioned $\boldsymbol{\theta}_{phy}$ into five sets $\boldsymbol{\theta}_{phy} := [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \boldsymbol{\theta}_3, \boldsymbol{\theta}_4, \boldsymbol{\theta}_5]$, namely:
$\boldsymbol{\theta}_1 := [SG, SI_B, G_{b_{dawn}}]$, $\boldsymbol{\theta}_2 := [SI_L, G_{b_{day}}]$, $\boldsymbol{\theta}_3 := [SI_D]$, $\boldsymbol{\theta}_4 := [p_2, k_{a2}, k_d]$, $\boldsymbol{\theta}_5 := [k_{empt}, k_{abs}]$.
This partitioning scheme has been chosen since it improves MC mixing and allows to break the correlation between SI and p_2, known to be critical from the literature [197].

An iteration i of the algorithm consists of five steps $p = 1, \ldots, 5$ and each step updates the p-th partition of $\boldsymbol{\theta}_{phy}$, $\boldsymbol{\theta}_p$, by approval/rejection of a sample $\boldsymbol{\phi}_p$ extracted from the proposal density function $q_p(\cdot|\cdot)$. Specifically, as prescribed by the SCMH procedure, approval occurs with probability α

$$\alpha = \min(1, \frac{\pi(\boldsymbol{\phi}_p|\boldsymbol{\theta}_{i,-p})q_p(\boldsymbol{\theta}_{i-1,p}|\boldsymbol{\phi}_p, \boldsymbol{\theta}_{i,-p})}{\pi(\boldsymbol{\theta}_{i-1,p}|\boldsymbol{\theta}_{i,-p})q_p(\boldsymbol{\phi}_p|\boldsymbol{\theta}_{i-1,p}, \boldsymbol{\theta}_{i,-p})})$$

with $\pi(\boldsymbol{\theta}_p|\boldsymbol{\theta}_{i,-p})$ proportional to the posterior of $\boldsymbol{\theta}_p$ given that the other components $\boldsymbol{\theta}_{-p}$ assume the value $\boldsymbol{\theta}_{-p} = \boldsymbol{\theta}_{i,-p}$:

$$\pi(\boldsymbol{\theta}_p|\boldsymbol{\theta}_{i,-p}) = p_{Y|\boldsymbol{\theta},U}(Y|\boldsymbol{\theta}_p, \boldsymbol{\theta}_{i,-p}, U)p_{\boldsymbol{\theta}}(\boldsymbol{\theta}_p|\boldsymbol{\theta}_{i,-p}, U)$$

$\boldsymbol{\theta}_{i,-p}$ comprises all the other components of $\boldsymbol{\theta}_{phy}$ except for $\boldsymbol{\theta}_p$. Precisely, $\boldsymbol{\theta}_{i,-p}$ contains the most updated version of each component as available at the cur-

rent stage of the algorithm: $\boldsymbol{\theta}_{i,-p} = [\boldsymbol{\theta}_{i,1}, \ldots, \boldsymbol{\theta}_{i,p-1}, \boldsymbol{\theta}_{i-1,p+1}, \boldsymbol{\theta}_{i-1,5}]$. Components up to $p-1$ have already been updated when processing the p-th components at iteration i, while other components, from $p+1$ to 5, have not been updated yet, so their value computed in the previous iteration $i-1$ is used.

For what it concerns the proposal distribution, we used a Gaussian centered in the value assumed by $\boldsymbol{\theta}_p$ in the previous chain iteraction

$$q_p(\cdot|\cdot) = N(\boldsymbol{\theta}_{i-1,p}, \Sigma_p)$$

where Σ_p is a tuning parameter that regulates the acceptance rate of the chain. We set Σ_p to a diagonal matrix whose components are an estimate of the conditional standard deviation of each element of partition p, $\mathrm{sd}(\boldsymbol{\theta}_{phy_p}|Y, U)$, multiplied by a scaling factor $2.4/\sqrt{d}$ as suggested in [198]. This estimates is computed by running two exploratory MCMCs for $nIter = 600$ iterations and updated every 1500 iterations of the algorithm, thus implementing an adaptive SCMH.

Finally, the convergence of the MCMC has been verified through the well-known Raftery-Lewis criterion [162], which provides the number of iterations necessary to ensure the Markov Chain to represent the target posterior distribution.

The Adaptive Single Component Metropolis Hasting is summarized in Algorithm 1.

$i \leftarrow 0$;
initialize $\boldsymbol{\theta}_{phy_0}$, $nIter$;
repeat
  for $p \leftarrow 1$ **to** 5 **do**
    set $\boldsymbol{\theta}_{phy_{i,-p}} = [\boldsymbol{\theta}_{phy_{i,1}}, \ldots, \boldsymbol{\theta}_{phy_{i,p-1}}, \boldsymbol{\theta}_{phy_{i-1,p+1}}, \boldsymbol{\theta}_{phy_{i-1,5}}]$;
    sample $\boldsymbol{\phi}_p \sim q_p(\cdot|\cdot)$;
    set

$$\alpha = \min(1, \frac{\pi(\boldsymbol{\phi}_p|\boldsymbol{\theta}_{i,-p})q_p(\boldsymbol{\theta}_{i-1,p}|\boldsymbol{\phi}_p, \boldsymbol{\theta}_{i,-p})}{\pi(\boldsymbol{\theta}_{i-1,p}|\boldsymbol{\theta}_{i,-p})q_p(\boldsymbol{\phi}_p|\boldsymbol{\theta}_{i-1,p}, \boldsymbol{\theta}_{i,-p})})$$

    sample U $\sim$ Uniform(0,1);
    if $U \leq \alpha$ **then**
      set $\boldsymbol{\theta}_{phy_{i,p}} = \boldsymbol{\phi}_p$;
    else
      set $\boldsymbol{\theta}_{phy_{i,p}} = \boldsymbol{\theta}_{phy_{i-1,p}}$;
    end
  end
  $i \leftarrow i + 1$
until $n < nIter$;

Algorithm 1: Adaptive Single Component Metropolis Hastings

A.2 Particle filter for BG prediction: implementation details

In the following, we present the numerical scheme implemented by PF to perform the one step-ahead prediction, measurement update, and multiple step-ahead prediction.

One step-ahead prediction step.
Recalling that, at time t_{k-1}, $p(\boldsymbol{x}(t_{k-1})|y(t_{1:k-1}), \boldsymbol{u}(t_{1:k-1}))$ is available in a sampled form defined by set of P particles $\{\boldsymbol{x}^p(t_{k-1})\}_{p=1}^P$ with associated weights $\{w(t_{k-1})^p\}_{p=1}^P, \sum_p w(t_{k-1})^p = 1$ such that

$$p(\boldsymbol{x}(t_{k-1})|y(t_{1:k-1}), \boldsymbol{u}(t_{1:k-1})) \approx \sum_{p=1}^{P} w^p(t_{k-1})\delta(\boldsymbol{x}(t_{k-1}) - \boldsymbol{x}^p(t_{k-1})),$$

PF performs the one step-ahead prediction step by drawing a new set of particles $\{\boldsymbol{x}^p(t_k)\}_{p=1}^P$ from $p(\boldsymbol{x}(t_k)|\boldsymbol{x}(t_{k-1}))$:

$$\boldsymbol{x}^p(t_k) \sim p(\boldsymbol{x}^p(t_k)|\boldsymbol{x}^p(t_{k-1})). \tag{A.1}$$

This probability is specified by (5.7):

$$p(\boldsymbol{x}^p(t_k)|\boldsymbol{x}^p(t_{k-1})) = N(\boldsymbol{f}(\boldsymbol{x}^p(t_{k-1}), \boldsymbol{u}, t_{k-1}, \boldsymbol{\theta}), \Sigma_v).$$

In view of this, to draw the new set of particles it is sufficient to let each particle $\boldsymbol{x}^p(t_{k-1})$ evolve according to model (5.7), and corrupt it with a realization of the noise $\boldsymbol{v}$.

Measurement update step. The PF algorithm sets the weight $w^{*p}(t_k)$ of each p-th particle $\boldsymbol{x}^p(t_k)$ to the likelihood function evaluated on $\boldsymbol{x}^p(t_k)$

$$w^{*p}(t_k) = p(y(t_k)|\boldsymbol{x}^p(t_k), \boldsymbol{u}(t_{1:k})). \tag{A.2}$$

In particular, from equation (5.7) and the statistics of the stochastic modelling error e, $p(y(t_k)|\boldsymbol{x}^p(t_k), \boldsymbol{u}(t_{1:k}))$ is defined as:

$$p(y(t_k)|\boldsymbol{x}(t_k), \boldsymbol{u}(t_{1:k})) = N(y(t_k) - y^p(t_k), SD_e). \tag{A.3}$$

where $y^p(t_k)$ is obtained using (5.7) and SD_e is the constant standard deviation of the error.

Weights are then normalized such that $\sum_p w^{*p}(t_k) = 1$.

This provides a sampled form representation of the posterior density

$$p(\boldsymbol{x}(t_k)|y(t_{1:k-1}),\boldsymbol{u}(t_{1:k})) \approx \sum_{p=1}^{P} w^{*p}(t_k)\delta(\boldsymbol{x}(t_k) - \boldsymbol{x}^p(t_k)).$$

Resampling step. To improve the accuracy of PF, the measurement update step is completed by updating the set of particles. Specifically, $\{\boldsymbol{x}^p(t_k)\}_{p=1}^P$ are substituted with a new set of P particles, $\{\boldsymbol{x}^{*p}(t_k)\}_{p=1}^P$ generated from the sampled representation of $p(\boldsymbol{x}(t_k)|y(t_{1:k-1}), \boldsymbol{u}(t_{1:k}))$ such that $\Pr(\boldsymbol{x}^{*p}(t_k) = \boldsymbol{x}^p(t_k)) = w^{*p}(t_k)$. This step is performed by a well-established resampling algorithm [199].

As a result, all new particles $\{\boldsymbol{x}^p(t_k)\}_{p=1}^P$ are associated to the same weight $w^{*p}(t_k) = 1/P$, thus the approximation of $p(\boldsymbol{x}(t_k)|y(t_{1:k-1}), \boldsymbol{u}(t_{1:k}))$ simplifies to

$$p(\boldsymbol{x}(t_k)|y(t_{1:k-1}), \boldsymbol{u}(t_{1:k})) \approx \frac{1}{P}\sum_{p=1}^{P} \delta(\boldsymbol{x}(t_k) - \boldsymbol{x}^p(t_k)).$$

Multiple steps-ahead prediction.

Multiple steps ahead predictions can be obtained as follows. First, the probabilities $p(x(t_{k+i})|y(t_{1:k}), \boldsymbol{u}(t_{1:k+i})), \forall, i = 1, \ldots, PH$ are obtained in sampled form starting from $p(x(t_k)|y(t_{1:k}), \boldsymbol{u}(t_{1:k}))$ and propagating the P particles $\{\boldsymbol{x}^p(t_k)\}_{p=1}^P$ i steps ahead as we did in the one step-ahead prediction step.

Then, the set of particles $\{y^p(t_{k+i}|t_k)\}_{p=1}^P$ is computed and used to obtain $p(y(t_{k+i})|y(t_{1:k}), \boldsymbol{u}(t_{1:k+i}))\forall, i = 1, \ldots, PH$ in sampled form.

Finally, a point estimate of glucose i steps-ahead, $y(t_{k+i})$, is obtained as the average computed over the sampled form of $p(y(t_{k+i})|y(t_{1:k}), \boldsymbol{u}(t_{1:k+i}))$, i.e.:

$$\hat{y}(t_{k+i}|t_k) = \frac{1}{P}\sum_{p=1}^{P} y^p(t_{k+i}|t_k), \forall i = 1, \ldots, PH.$$

The implemented PF is summarized in Algorithm 2.

$k \leftarrow 1$;
$\boldsymbol{x}_{t_0} = N(\boldsymbol{x}_{ss}, \Sigma_v))$;
repeat
  Step 1: One step-ahead prediction;
  for $p \leftarrow 1$ **to** P **do**
    sample $\boldsymbol{x}^p(t_k)$ from $N(\boldsymbol{f}(\boldsymbol{x}^p(t_{k-1}), \boldsymbol{u}, t_{k-1}, \boldsymbol{\theta}), \Sigma_v))$;
  end
  Step 2: Measurement update;
  for $p \leftarrow 1$ **to** P **do**
    compute $w^{*p}(t_k) = N(y(t_k) - y^p(t_k), SD_\epsilon)$;
  end
  normalize $w^{*p}(t_k) = w^{*p}(t_k) / \sum_p w^{*p}(t_k)$;
  Step 3: Resampling;
  sample $\{\boldsymbol{x}^{*p}(t_k)\}_{p=1}^P$ s.t. $\Pr(\boldsymbol{x}^{*p}(t_k) = \boldsymbol{x}^p(t_k)) = w^{*p}(t_k)$;
  set $\{\boldsymbol{x}^p(t_k)\}_{p=1}^P = \{\boldsymbol{x}^{p*}(t_k)\}_{p=1}^P$;
  Step 4: Multiple steps-ahead prediction;
  for $p \leftarrow 1$ **to** P **do**
    compute $y^p(t_{k+i}|t_k), \forall i = 1, \ldots, PH$;
  end
  compute $\hat{y}(t_{k+i}|t_k) = \frac{1}{P}\sum_{p=1}^{P} y^p(t_{k+i}|t_k), \forall i = 1, \ldots, PH$;
  $k \leftarrow k + 1$
until $k \leq D$;

Algorithm 2: Particle Filter

Appendix B

Deep learning models

B.1 LSTM

Long short-term memory (LSTM) networks are a type of recurrent neural network (RNN) capable of learning and maintaining time dependencies in sequence prediction problems [200]. At a high-level, LSTM consists of three parts, each dedicated to a specific individual function. The first part is responsible to decide whether the incoming information (from the previous timestamp) has to be remembered or it can be forgotten. In the second phase, the cell learns new information from the input. Finally, the cell passes the updated information from the current time step to the next time step. These three phases of an LSTM cell are known as: the *Forget gate*, the *Input gate*, and the *Output gate*.

As RNN, LSTM is equipped by a hidden state for previous and current timestamps, namely $H(t-1)$ and $H(t)$, respectively. In addition to that, a cell state, namely $C(t-1)$ and $C(t)$ for previous and current time step respectively, are available to the LSTM. Of note, the hidden state is known as *Short-term* memory and the cell state is known as *Long-term* memory. This state represents the core of an LSTM. In fact, the cell state transfers relative information along the sequence chain. This can be interpret as the memory of the network which can carry on relevant information throughout the processing of the sequence. As a result, even information about previous time steps can have an impact on subsequent timestamps, reducing the effects on short-term memory. In such a context, the information gets added or removed to the cell state through the gates, which decide which information has to be translated to the cell state.

The gates of an LSTM contain sigmoid activations (σ), that flattens values

between 0 and 1. This helps the network to update or forget the data. In fact, if the multiplication results in 0, the information is considered forgotten. Similarly, the information is kept if the value is 1. By applying this strategy, the LSTM learns which data is not important, therefore can be forgotten, or which is important to be kept.

Forget gate, f_t: it decides which information has to be kept and which can be ignored. The information from the current input X_t and the previous hidden state H_{t-1} are passed through the sigmoid function, that generates values between 0 and 1. As already mentioned, values close to 0 will be forgotten, values close to 1 will be kept. f_t will be used later by the cell for point-wise multiplication.

Input gate, i_t: it updates the cell status. First, the current state X_t and previously hidden state H_{t-1} are passed into a second sigmoid function. The values are transformed between 0 (not-important) and 1 (important). Next, X_t and H_{t-1} will be passed through a tanh function. To regulate the network, the tanh operator will create a vector (n_t) with all the possible values between -1 and 1. The output values generated from the activation functions are ready for point-by-point multiplication.

Now, the network has enough information form the forget gate and input gate. The next step is to decide and store the information from the new state in the cell state. The previous cell state C_{t-1} is multiplied by the forget vector f_t. If the outcome is 0, then values will get dropped in the cell state. Next, the network takes the output value of the input vector i_t and performs point-by-point addition, which updates the cell state giving the network a new cell state C_t.

Output gate, o_t: it determines the value of the next hidden state. First, the values of the current state X_t and previous hidden state H_{t-1} are passed into the third sigmoid function. Finally, the current hidden state H_t is obtained by (point-wise) multiplication with the new cell state C_t, which is passed through the tanh function, to the o_t. It is worth noting that this hidden state H_t is used for prediction.

Finally, here below the equations related to an LSTM network:

$$\begin{aligned} f_t &= \sigma\left(U_f \cdot X_t + W_f \cdot H_{t-1} + b_f\right) , \\ i_t &= \sigma\left(U_i \cdot X_t + W_i \cdot H_{t-1} + b_i\right) , \\ n_t &= tanh\left(U_n \cdot X_t + W_n \cdot H_{t-1} + b_n\right) , \\ o_t &= \sigma\left(U_o \cdot X_t + W_o \cdot H_{t-1} + b_o\right) , \\ C_t &= f_t \bigodot C_{t-1} + i_t \cdot n_t , \\ H_t &= o_t \bigodot tanh(C_t) . \end{aligned} \tag{B.1}$$

where X_t is the input to current time step, and H_{t-1} is the hidden state of the previous time step; U_j and W_j are the weight associated with input and with hidden state, respectively, and b_j is a bias vector, for $j \in \{f, i, n, o\}$.

B.2 GRU

Gated recurrent unit (GRU) is a popular variant of LSTM network introduced by [201]. It uses gating mechanisms to control and manage the flow of information between cells in the network. Unlike LSTM, GRU has only two gates: the *Update* and *Reset* gate.

Update gate, u_t: it decides which information is to retain and which is the new information to be added. As for the LSTM, the current input X_t and the previous hidden state H_{t-1} are passed into a sigmoid functions.

Reset gate, r_t: it controls which is information that has to be passed to the GRU in the next instance of time. As for the u_t, the current input X_t and the previous state H_{t-1} are passed into the sigmoid function.

Once r_t is available, the new information n_t can be computed by passing through the tanh, the sum of: the current input X_t, the point-wise multiplication between r_t and the previous hidden state H_{t-1}. As a final step, the new hidden state H_t is computed by pointwise multiplication of n_t with the $(1 - u_t)$ and u_t with H_{t-1}

The equations related to a GRU network are reported here below:

$$\begin{aligned} u_t &= \sigma\left(U_u \cdot X_t + W_u \cdot H_{t-1} + b_u\right) , \\ r_t &= \sigma\left(U_r \cdot X_t + W_r \cdot H_{t-1} + b_r\right) , \\ n_t &= tanh\left(U_n \cdot X_t + W_n \cdot \left(r_t \bigodot H_{t-1}\right) + b_n\right) , \\ H_t &= (1 - u_t) \bigodot n_t + u_t \bigodot H_{t-1} . \end{aligned} \tag{B.2}$$

where X_t is the input to current time step, and H_{t-1} is the hidden state of the previous time step; U_j and W_j are the weight associated with input and with hidden state, respectively, and b_j is a bias vector, for $j \in \{u, r, n\}$.

B.3 TCN

Temporal convolutional network (TCN) is inspired by recent convolutional architectures for sequential data and combines simplicity, autoregressive prediction, and very long memory [202]. The TCN is designed from two basic principles:

1. the convolutions are causal, meaning that there is no information leakage from future to past;

2. the architecture can take a sequence of any length and map it to an output sequence of the same length.

To achieve the first point, TCN exploits causal convolutions, i.e., convolutions leveraging data available up to time t. To accomplish the second point, the TCN uses a one-dimensional (1D) fully-convolutional network architecture, where each hidden layer is the same length as the input layer.

Dilated convolution: simple causal convolutions have the disadvantage to only look back at history with size linear in the depth of the network, i.e., the receptive field grows linearly with every additional layer. To overcome this point, the architecture of TCN employs dilated convolutions that enable an exponentially large receptive field. More formally, for an input sequence $x \in \mathbb{R}^T$ and a filter $h : \{0, ..., k-1\} \rightarrow \mathbb{R}$, the dilated convolution operation H on element x of the sequence is defined as

$$H(x) = (x * h)(x) = \sum_{i=0}^{k-1} f(i) x_{s-d \cdot i} , \tag{B.3}$$

where $d = 2^{\nu}$ is the dilation factor, with ν the level of the network, and k is the filter size. The term $s - d \cdot i$ accounts for the direction of the past. Dilation is equivalent to introducing a fixed step between every two adjacent filters. Using larger dilation enables an output at the top level to represent a wider range of inputs, thus effectively expanding the receptive field of a Convolutional Neural Network. There are two ways to increase the receptive field of a TCN: choosing lager filter sizes k and increasing the dilation factor d, since the effective history of one layer is $(k-1)d$.

Residual block: in place of a convolutional layer, TCN employs a generic residual module. Each residual block contains a branch leading out to a series of transformations $\mathcal{F}$, whose outputs are added to the input x of the block,

$$o = Activation\left(x + \mathcal{F}(x)\right) \ . \tag{B.4}$$

A residual block comprises two layers of dilated causal convolutions and rectified linear units (ReLU) as non-linearities as shown in Figure B.1.

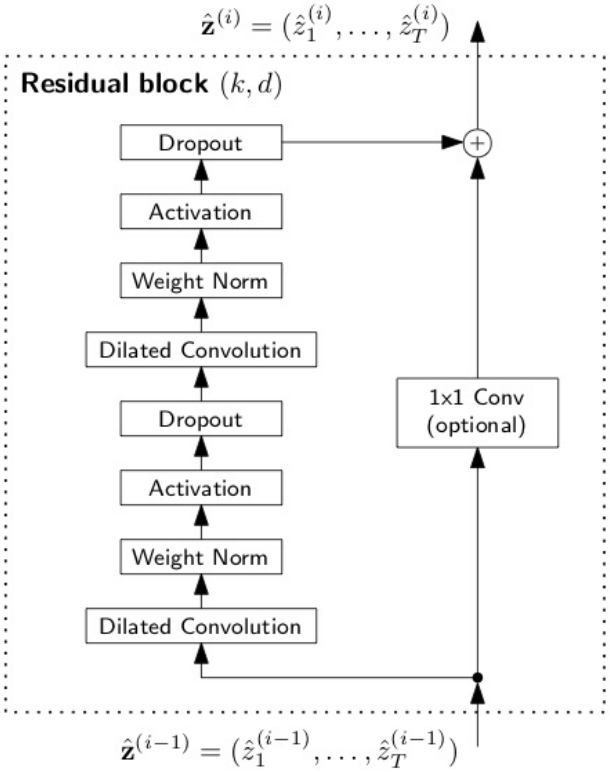


Figure B.1: Diagram of the Residual Block

Weight normalization is applied to the convolutional filters and a spatial dropout

is added after each dilated convolution for regularization, meaning that at each training step a whole channel is zeroed out.

The TCN model is deliberately kept simple, combining some of the best practices of modern convolutional architectures. TCNs can be build to have very long effective history sizes, which means they have the ability to look very far into the past to make a prediction. To this end, a combination of very deep networks augmented with residual layers and dilated convolutions are deployed. The TCN architecture contains the following properties:

- *Parallelism*: convolutions can be calculated in parallel because the same filter is used in each layer. Therefore, in both training and evaluation, a long input sequence can be processed as a whole, instead of sequentially as done in RNNs.
- *Flexible receptive field size*: stacking more dilated convolutional layers, using larger dilation factors, or increasing the filter size allow to extend the receptive field size. Thus, TCNs afford better control of the model's memory size, and are easy to adapt to different domains.
- *Low memory requirement for training*: the filters are shared across a layer, with the back-propagation path depending only on the network depth.

Appendix C

Non-parametric linear models

C.1 Non-parametric approach for model identification

[1]

Let us now consider that the 1-step ahead predictor can be written as:

$$\begin{gathered}\hat{g}(k|k-1,\theta) = \\ h_1 * i(k) + h_2 * m(k) + h_3 * g(k)\ ,\end{gathered} \tag{C.1}$$

where $*$ represents the convolution between two signals, whereas h_1, h_2, and h_3 are impulse responses related to the insulin, meal, and glucose signal, respectively.
These functions are unknown, and they have to be estimated from noisy measurements. The estimation of these unknown responses can be performed by solving an optimization problem in an infinite-dimensional functional space given by a reproducing kernel Hilbert space (RKHS) [203, 204, 205]. The kernel of the RKHS should reflect the properties of the functions to be estimated and its choice is a key point in the non-parametric approaches. Indeed, it can incorporate useful prior knowledge, e.g., smoothness; moreover, the trade-off between data fit and regularity of the estimate can be properly handled by the estimate of the kernel's unknown parameters. In order to incorporate within the kernel the stability of the predictor's impulse responses, in this study it was used the Stable-Spline kernel (SSK), proposed in [206]. Using this approach, a

[1] This appendix contains material published in Faccioli et al., *IEEE Transaction on Biomedical Engineering, 2021*, [62]

generic Stable-Spline impulse response f_{SSK} is seen as a realization of a zero-mean Gaussian random process, whose covariance specify the SSK kernel, and can be written as

$$\begin{aligned} Cov(f_{SSK}(k), f_{SSK}(l)) = \lambda^2 K(k,l) = \\ \lambda^2 \left(\frac{e^{-\beta(k+l)} e^{-\beta max(k,l)}}{2} - \frac{e^{-3\beta max(k,l)}}{6} \right) , \end{aligned} \tag{C.2}$$

where $k, l = 1, 2, ..., \infty$, $\beta > 0$, and $\lambda > 0$. Let now define K_{h_1}, K_{h_2}, and K_{h_3} as the SSK of h_1, h_2, and h_3 respectively, and let $\mathscr{H}_{h_1}$, $\mathscr{H}_{h_2}$, and $\mathscr{H}_{h_3}$ denote the RKHS of deterministic functions on $\mathbb{N}$ associated with K_{h_1}, K_{h_2}, and K_{h_3} (with norms denoted by $\| \cdot \|_{\mathscr{H}_{h_1}}$, $\| \cdot \|_{\mathscr{H}_{h_2}}$, and $\| \cdot \|_{\mathscr{H}_{h_3}}$), [207]. The Stable-Spline estimators $\hat{h_1}$, $\hat{h_2}$, and $\hat{h_3}$ are obtained from the solution of the following Tikhonov-type problem:

$$\begin{aligned} (\hat{h_1}, \hat{h_2}, \hat{h_3}) = \\ \underset{h_1 \in \mathscr{H}_{h_1}, h_2 \in \mathscr{H}_{h_2}, h_3 \in \mathscr{H}_{h_3}}{\operatorname{argmin}} \{ \| g^+ - Ah_1 - Bh_2 - Ch_3 \|^2 + \\ + \gamma_{h_1} \|h_1\|^2_{\mathscr{H}_{h_1}} + \gamma_{h_2} \|h_2\|^2_{\mathscr{H}_{h_2}} + \gamma_{h_3} \|h_3\|^2_{\mathscr{H}_{h_3}} \} , \end{aligned} \tag{C.3}$$

$$[A]_{kl} = i(k-l) , \quad [B]_{kl} = m(k-l) , \quad [C]_{kl} = g(k-l) ,$$

$$k = 1, 2, ..., \infty , \quad l = 1, 2, ..., n ,$$

$$g^+ = [g_1 \;\; g_2 \;\; \cdots \;\; g_n]^T ,$$

where $\| \cdot \|$ is the Euclidian norm, $\gamma_{h_1} = \sigma^2 / \lambda^2_{h_1}$, $\gamma_{h_2} = \sigma^2 / \lambda^2_{h_2}$, $\gamma_{h_3} = \sigma^2 / \lambda^2_{h_3}$, and g^+ is a vector containing n CGM data with n being the number of future samples considered in the identification procedure. Note that, to make practically implementable the above strategy, the infinitely long impulse responses are approximated by truncation after N_p samples.

Note also that from (C.2) the covariances of the impulse responses h_1, h_2, and h_3 include the parameters β_{h_1}, β_{h_2}, and β_{h_3}, respectively. It can be therefore defined the hyperparameters of our problem (β_{h_1}, β_{h_2}, β_{h_3}, λ_{h_1}, λ_{h_2}, λ_{h_3}, and σ), that have to be properly tuned before to the solution of the Tikhonov problem (C.3). By assuming known the hyperparameters, the solution of (C.3) is given

by

$$\begin{aligned} \hat{h_1} &= \lambda_{h_1}^2 K_{h_1} A^T \phi \,, \\ \hat{h_2} &= \lambda_{h_2}^2 K_{h_2} B^T \phi \,, \\ \hat{h_3} &= \lambda_{h_3}^2 K_{h_3} C^T \phi \,, \\ \phi &= (\lambda_{h_1}^2 A K_{h_1} A^T + \lambda_{h_2}^2 B K_{h_2} B^T + \\ &\quad + \lambda_{h_3}^2 C K_{h_3} C^T + \sigma^2 I_n)^{-1} g^+ \,, \end{aligned} \tag{C.4}$$

where I_n is the $n \times n$ identity matrix.

Regarding the hyperparameters (denoted by ζ), they can be estimated via maximum marginal likelihood [207]:

$$\begin{aligned} \hat{\zeta} &= \underset{\zeta}{\operatorname{argmin}}(J(g^+, \zeta)) \,, \\ J(g^+, \zeta) &= \frac{1}{2} ln(det[2\pi V[g^+]]) + \frac{1}{2} (g^+)^T (V[g^+])^{-1} g^+ \,, \\ V[g^+] &= \lambda_{h_1}^2 A K_{h_1} A^T + \lambda_{h_2}^2 B K_{h_2} B^T + \\ &\quad + \lambda_{h_3}^2 C K_{h_3} C^T + \sigma^2 I_n \,, \end{aligned} \tag{C.5}$$

where J is the opposite log-marginal likelihood of g^+. For further details, please refer to [207].

Bibliography

[1] C. Marling and R. Bunescu, "The ohiot1dm dataset for blood glucose level prediction: Update 2020," in *CEUR workshop proceedings*, vol. 2675, p. 71, NIH Public Access, 2020.

[2] D. Daneman, "Type 1 diabetes," *Lancet*, vol. 367, no. 9513, pp. 847–858, 2006.

[3] K. Alberti and P. Zimmet, "Definition, diagnosis and classification of diabetes mellitus and its complications. Part 1: Diagnosis and classification of diabetes mellitus. Provisional report of a WHO Consultation," *Diabetic Medicine*, vol. 15, pp. 539–553, 7 1998.

[4] M. I. Schmidt, A. Hadji-Georgopoulos, M. Rendell, S. Margolis, and A. Kowarski, "The dawn phenomenon, an early morning glucose rise: implications for diabetic intraday blood glucose variation," *Diabetes care*, vol. 4, no. 6, pp. 579–585, 1981.

[5] American Diabetes Association, "Introduction: standard of medical care in diabetes," *Diabetes Care*, vol. 42, no. S1, pp. S61–S70, 2019.

[6] J. Walsh and R. Roberts, *Pumping insulin: everything you need for success on a smart insulin pump*, vol. 4. Torrey Pines Press San Diego, CA, 2006.

[7] K. K. Trout, C. Homko, and N. C. Tkacs, "Methods of measuring insulin sensitivity," *Biological Research for Nursing*, vol. 8, no. 4, pp. 305–318, 2007.

[8] P. Choudhary and S. A. Amiel, "Hypoglycaemia in type 1 diabetes: technological treatments, their limitations and the place of psychology," *Diabetologia*, vol. 61, no. 4, pp. 761–769, 2018.

[9] R. S. Weinstock, D. Xing, D. M. Maahs, A. Michels, M. R. Rickels, A. L. Peters, R. M. Bergenstal, B. Harris, S. N. DuBose, K. M. Miller, R. W. Beck, and for the T1D Exchange Clinic Network, "Severe Hypoglycemia

and Diabetic Ketoacidosis in Adults With Type 1 Diabetes: Results From the T1D Exchange Clinic Registry," *The Journal of Clinical Endocrinology & Metabolism*, vol. 98, pp. 3411–3419, 08 2013.

[10] J. K. Snell-Bergeon and R. P. Wadwa, "Hypoglycemia, diabetes, and cardiovascular disease," *Diabetes Technology & Therapeutics*, vol. 14, no. S1, pp. S–51, 2012.

[11] B. M. Frier, G. Schernthaner, and S. R. Heller, "Hypoglycemia and cardiovascular risks," *Diabetes care*, vol. 34, no. Supplement_2, pp. S132–S137, 2011.

[12] R. E. Davis, M. Morrissey, J. R. Peters, K. Wittrup-Jensen, T. Kennedy-Martin, and C. J. Currie, "Impact of hypoglycaemia on quality of life and productivity in type 1 and type 2 diabetes," *Current Medical Research and Opinion*, vol. 21, no. 9, pp. 1477–1483, 2005.

[13] L. Gonder-Frederick, K. Vajda, K. Schmidt, D. Cox, J. Devries, O. Erol, K. Kanc, H. Schächinger, and F. Snoek, "Examining the behaviour subscale of the hypoglycaemia fear survey: an international study," *Diabetic Medicine*, vol. 30, no. 5, pp. 603–609, 2013.

[14] R. W. Beck, R. M. Bergenstal, L. M. Laffel, and J. C. Pickup, "Advances in technology for management of type 1 diabetes," *The Lancet*, vol. 394, no. 10205, pp. 1265–1273, 2019.

[15] S. Clarke and J. Foster, "A history of blood glucose meters and their role in self-monitoring of diabetes mellitus," *British Journal of Biomedical Science*, vol. 69, no. 2, pp. 83–93, 2012.

[16] J. K. Kirk and J. Stegner, "Self-monitoring of blood glucose: practical aspects," *Journal of Diabetes Science and Technology*, vol. 4, no. 2, pp. 435–439, 2010.

[17] J. Kravarusic and G. Aleppo, "Diabetes technology use in adults with type 1 and type 2 diabetes," *Endocrinology and Metabolism Clinics*, vol. 49, no. 1, pp. 37–55, 2020.

[18] K. Dovc and T. Battelino, "Evolution of diabetes technology," *Endocrinology and Metabolism Clinics*, vol. 49, no. 1, pp. 1–18, 2020.

[19] J. P. Shivers, L. Mackowiak, H. Anhalt, and H. Zisser, ""turn it off!": diabetes device alarm fatigue considerations for the present and the future," *Journal of Diabetes Science and Technology*, vol. 7, no. 3, pp. 789–794, 2013.

[20] G. McGarraugh, "Alarm characterization for continuous glucose monitors used as adjuncts to self-monitoring of blood glucose," *Journal of Diabetes Science and Technology*, vol. 4, no. 1, pp. 41–48, 2010. PMID: 20167166.

[21] G. Cappon, M. Vettoretti, G. Sparacino, and A. Facchinetti, "Continuous glucose monitoring sensors for diabetes management: a review of technologies and applications," *Diabetes Metabolism*, vol. 43, no. 4, pp. 383–397, 2019.

[22] S. B. Abraham, S. Arunachalam, A. Zhong, P. Agrawal, O. Cohen, and C. M. McMahon, "Improved real-world glycemic control with continuous glucose monitoring system predictive alerts," *Journal of Diabetes Science and Technology*, vol. 15, no. 1, pp. 91–97, 2021.

[23] J. R. Castle and P. G. Jacobs, "Nonadjunctive use of continuous glucose monitoring for diabetes treatment decisions," *Journal of Diabetes Science Technology*, vol. 10, no. 5, 2016.

[24] M. Vettoretti, A. Facchinetti, G. Sparacino, and C. Cobelli, "Type-1 diabetes patient decision simulator for in silico testing safety and effectiveness of insulin treatments," *IEEE Transaction Biomedical Engineering*, vol. 65, no. 6, pp. 1281–1290, 2017.

[25] T. Battelino, I. Conget, B. Olsen, I. Schütz-Fuhrmann, E. Hommel, R. Hoogma, U. Schierloh, N. Sulli, J. Bolinder, and SWITCH Study Group, "The use and efficacy of continuous glucose monitoring in type 1 diabetes treated with insulin pump therapy: A randomised controlled trial," *Diabetologia*, vol. 55, no. 12, pp. 3155–3162, 2012.

[26] G. Aleppo, K. J. Ruedy, T. D. Riddlesworth, D. F. Kruger, A. L. Peters, I. Hirsch, R. M. Bergenstal, E. Toschi, A. J. Ahmann, V. N. Shah, *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," *Diabetes care*, vol. 40, no. 4, pp. 538–545, 2017.

[27] N. Camerlingo, M. Vettoretti, S. Del Favero, G. Cappon, G. Sparacino, and A. Facchinetti, "A real-time continuous glucose monitoring–based algorithm to trigger hypotreatments to prevent/mitigate hypoglycemic events," *Diabetes Technology & Therapeutics*, vol. 21, no. 11, pp. 644–655, 2019.

[28] Q. Sun, M. V. Jankovic, J. Budzinski, B. Moore, P. Diem, C. Stettler, and S. G. Mougiakakou, "A dual mode adaptive basal-bolus advisor based on reinforcement learning," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 6, pp. 2633–2641, 2018.

[29] G. P. Forlenza, B. Buckingham, and D. M. Maahs, "Progress in diabetes technology: developments in insulin pumps, continuous glucose monitors, and progress towards the artificial pancreas," *The Journal of Pediatrics*, vol. 169, pp. 13–20, 2016.

[30] C. Berget, L. H. Messer, and G. P. Forlenza, "A clinical overview of insulin pump therapy for the management of diabetes: past, present, and future of intensive therapy," *Diabetes Spectrum*, vol. 32, no. 3, pp. 194–204, 2019.

[31] K. Evans, "Insulin pumps in hospital: a guide for the generalist physician," *Clinical Medicine*, vol. 13, no. 3, p. 244, 2013.

[32] G. E. Umpierrez and D. C. Klonoff, "Diabetes technology update: use of insulin pumps and continuous glucose monitoring in the hospital," *Diabetes care*, vol. 41, no. 8, pp. 1579–1589, 2018.

[33] D. Naranjo, M. L. Tanenbaum, E. Iturralde, and K. K. Hood, "Diabetes technology: uptake, outcomes, barriers, and the intersection with distress," *Journal of Diabetes Science and Technology*, vol. 10, no. 4, pp. 852–858, 2016.

[34] N. C. Foster, R. W. Beck, K. M. Miller, M. A. Clements, M. R. Rickels, L. A. DiMeglio, D. M. Maahs, W. V. Tamborlane, R. Bergenstal, E. Smith, *et al.*, "State of type 1 diabetes management and outcomes from the t1d exchange in 2016–2018," *Diabetes Technology & Therapeutics*, vol. 21, no. 2, pp. 66–72, 2019.

[35] S. L. Sy, M. M. Munshi, and E. Toschi, "Can smart pens help improve diabetes management?," *Journal of Diabetes Science and Technology*, vol. 16, no. 3, pp. 628–634, 2022.

[36] E. Bekiari, K. Kitsios, H. Thabit, M. Tauschmann, E. Athanasiadou, T. Karagiannis, A.-B. Haidich, R. Hovorka, and A. Tsapas, "Artificial pancreas treatment for outpatients with type 1 diabetes: systematic review and meta-analysis," *British Medical Journal*, vol. 361, 2018.

[37] T. Peyser, E. Dassau, M. Breton, and J. S. Skyler, "The artificial pancreas: current status and future prospects in the management of diabetes," *Annals of the New York Academy of Sciences*, vol. 1311, no. 1, pp. 102–123, 2014.

[38] J. Pavan, D. Salvagnin, A. Facchinetti, G. Sparacino, and S. Del Favero, "Incorporating sparse and quantized carbohydrates suggestions in model predictive control for artificial pancreas in type 1 diabetes," *IEEE Transactions on Control Systems Technology*, 2022.

[39] C. Cobelli, C. Dalla Man, G. Sparacino, L. Magni, G. De Nicolao, and B. P. Kovatchev, "Diabetes: models, signals, and control," *IEEE Reviews in Biomedical Engineering*, vol. 2, pp. 54–96, 2009.

[40] M. Messori, J. Kropff, S. Del Favero, J. Place, R. Visentin, R. Calore, C. Toffanin, F. Di Palma, G. Lanzola, A. Farret, *et al.*, "Individually adaptive artificial pancreas in subjects with type 1 diabetes: a one-month proof-of-concept trial in free-living conditions," *Diabetes Technology & Therapeutics*, vol. 19, no. 10, pp. 560–571, 2017.

[41] I. Contreras, J. Vehi, *et al.*, "Artificial intelligence for diabetes management and decision support: literature review," *Journal of Medical Internet Research*, vol. 20, no. 5, p. e10775, 2018.

[42] L. Cossu, G. Cappon, G. Sparacino, and A. Facchinetti, "A new integrated platform for gathering and managing multivariable and multisensor data in diabetes clinical studies," in *2020 International Conference on e-Health and Bioengineering (EHB)*, pp. 1–4, IEEE, 2020.

[43] G. Noaro, G. Cappon, M. Vettoretti, G. Sparacino, S. Del Favero, and A. Facchinetti, "Machine-learning based model to improve insulin bolus

calculation in type 1 diabetes therapy," *IEEE Transaction on Biomedical Engineering*, vol. 68, no. 1, pp. 247–255, 2020.

[44] N. S. Tyler, C. M. Mosquera-Lopez, L. M. Wilson, R. H. Dodier, D. L. Branigan, V. B. Gabo, F. H. Guillot, W. W. Hilts, J. El Youssef, J. R. Castle, *et al.*, "An artificial intelligence decision support system for the management of type 1 diabetes," *Nature Metabolism*, vol. 2, no. 7, pp. 612–619, 2020.

[45] M. Anthimopoulos, J. Dehais, S. Shevchik, B. H. Ransford, D. Duke, P. Diem, and S. Mougiakakou, "Computer vision-based carbohydrate estimation for type 1 patients with diabetes using smartphones," *Journal of Diabetes Science and Technology*, vol. 9, no. 3, pp. 507–515, 2015.

[46] J. Daniels, P. Herrero, and P. Georgiou, "A deep learning framework for automatic meal detection and estimation in artificial pancreas systems," *Sensors*, vol. 22, no. 2, p. 466, 2022.

[47] S. Faccioli, I. Sala-Mira, J. Díez, A. Facchinetti, G. Sparacino, S. Del Favero, and J. Bondia, "Super–twisting-based meal detector for type 1 diabetes management: Improvement and assessment in a real-life scenario," *Computer Methods and Programs in Biomedicine*, vol. 219, p. 106736, 2022.

[48] M. Gadaleta, A. Facchinetti, E. Grisan, and M. Rossi, "Prediction of adverse glycemic events from continuous glucose monitoring signal," *IEEE journal of biomedical and health informatics*, vol. 23, no. 2, pp. 650–659, 2018.

[49] K. Lunze, T. Singh, M. Walter, M. D. Brendel, and S. Leonhardt, "Blood glucose control algorithms for type 1 diabetic patients: A methodological review," *Biomedical Signal Processing and Control*, vol. 8, no. 2, pp. 107–119, 2013.

[50] S. Oviedo, J. Vehí, R. Calm, and J. Armengol, "A review of personalized blood glucose prediction strategies for t1dm patients," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 33, no. 6, p. e2833, 2017.

[51] A. Z. Woldaregay, E. Årsand, S. Walderhaug, D. Albers, L. Mamykina, T. Botsis, and G. Hartvigsen, "Data-driven modeling and prediction of

blood glucose dynamics: Machine learning applications in type 1 diabetes," *Artificial Intelligence in Medicine*, vol. 98, pp. 109–134, 2019.

[52] D. J. Cox, L. Gonder-Frederick, and W. Clarke, "Driving decrements in type i diabetes during moderate hypoglycemia," *Diabetes*, vol. 42, no. 2, pp. 239–243, 1993.

[53] G. Sparacino, F. Zanderigo, S. Corazza, A. Maran, A. Facchinetti, and C. Cobelli, "Glucose concentration can be predicted ahead in time from continuous glucose monitoring sensor time-series," *IEEE Transactions on biomedical engineering*, vol. 54, no. 5, pp. 931–937, 2007.

[54] J. Reifman, S. Rajaraman, A. Gribok, and W. K. Ward, "Predictive monitoring for improved management of glucose levels," *Journal of Diabetes Science and Technology*, vol. 1, no. 4, pp. 478–486, 2007.

[55] A. Gani, A. V. Gribok, Y. Lu, W. K. Ward, R. A. Vigersky, and J. Reifman, "Universal glucose models for predicting subcutaneous glucose concentration in humans," *IEEE Transaction on Information Technology in Biomedicine*, vol. 14, no. 1, pp. 157–165, 2009.

[56] M. Eren-Oruklu, A. Cinar, D. K. Rollins, and L. Quinn, "Adaptive system identification for estimating future glucose concentrations and hypoglycemia alarms," *Automatica*, vol. 48, no. 8, pp. 1892–1897, 2012.

[57] M. Otoom, H. Alshraideh, H. M. Almasaeid, D. López-de Ipiña, and J. Bravo, "Real-time statistical modeling of blood sugar," *Journal of Medical Systems*, vol. 39, no. 10, pp. 1–6, 2015.

[58] J. Yang, L. Li, Y. Shi, and X. Xie, "An arima model with adaptive orders for predicting blood glucose concentrations and hypoglycemia," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 3, pp. 1251–1260, 2018.

[59] D. A. Finan, F. J. Doyle III, C. C. Palerm, W. C. Bevier, H. C. Zisser, L. Jovanovič, and D. E. Seborg, "Experimental evaluation of a recursive model identification technique for type 1 diabetes," *Journal of Diabetes Science and Technology*, vol. 3, no. 5, pp. 1192–1202, 2009.

[60] E. I. Georga, J. C. Príncipe, E. C. Rizos, and D. I. Fotiadis, "Kernel-based adaptive learning improves accuracy of glucose predictive modelling in

type 1 diabetes: a proof-of-concept study," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 2765–2768, IEEE, 2017.

[61] S. Del Favero, G. Pillonetto, C. Cobelli, and G. De Nicolao, "A novel nonparametric approach for the identification of the glucose-insulin system in type 1 diabetic patients," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 8340–8346, 2011.

[62] S. Faccioli, A. Facchinetti, G. Sparacino, G. Pillonetto, and S. Del Favero, "Linear model identification for personalized prediction and control in diabetes," *IEEE Transactions on Biomedical Engineering*, vol. 69, no. 2, pp. 558–568, 2021.

[63] N. Hobbs, I. Hajizadeh, M. Rashid, K. Turksoy, M. Breton, and A. Cinar, "Improving glucose prediction accuracy in physically active adolescents with type 1 diabetes," *Journal of Diabetes Science and Technology*, vol. 13, no. 4, pp. 718–727, 2019.

[64] S. Faccioli, B. Ozaslan, J. F. Garcia-Tirado, M. Breton, and S. Del Favero, "Black-box model identification of physical activity in type-l diabetes patients," in *2018 40th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*, pp. 3910–3913, IEEE, 2018.

[65] C. Pérez-Gandía, A. Facchinetti, G. Sparacino, C. Cobelli, E. Gómez, M. Rigla, A. de Leiva, and M. Hernando, "Artificial neural network algorithm for online glucose prediction from continuous glucose monitoring," *Diabetes Technology & Therapeutics*, vol. 12, no. 1, pp. 81–88, 2010.

[66] C. Zecchin, A. Facchinetti, G. Sparacino, G. De Nicolao, and C. Cobelli, "Neural network incorporating meal information improves accuracy of short-time prediction of glucose concentration," *IEEE Transaction on Biomedical Engineering*, vol. 59, no. 6, pp. 1550–1560, 2012.

[67] T. Kushner, M. D. Breton, and S. Sankaranarayanan, "Multi-hour blood glucose prediction in type 1 diabetes: A patient-specific approach using shallow neural network models," *Diabetes Technology & Therapeutics*, vol. 22, no. 12, pp. 883–891, 2020.

[68] J. Xie and Q. Wang, "Benchmarking machine learning algorithms on blood glucose prediction for type i diabetes in comparison with classical

time-series models," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 11, pp. 3101–3124, 2020.

[69] A. Aliberti, I. Pupillo, S. Terna, E. Macii, S. Di Cataldo, E. Patti, and A. Acquaviva, "A multi-patient data-driven approach to blood glucose prediction," *IEEE Access*, vol. 7, pp. 69311–69325, 2019.

[70] Q. Sun, M. V. Jankovic, L. Bally, and S. G. Mougiakakou, "Predicting blood glucose with an lstm and bi-lstm based deep neural network," in *2018 14th Symposium on Neural Networks and Applications (NEUREL)*, pp. 1–5, IEEE, 2018.

[71] R. Bunescu, N. Struble, C. Marling, J. Shubrook, and F. Schwartz, "Blood glucose level prediction using physiological models and support vector regression," in *2013 12th International Conference on Machine Learning and Applications*, vol. 1, pp. 135–140, IEEE, 2013.

[72] S. Mirshekarian, R. Bunescu, C. Marling, and F. Schwartz, "Using lstms to learn physiological models of blood glucose behavior," in *2017 39th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 2887–2891, IEEE, 2017.

[73] A. Bertachi, L. Biagi, I. Contreras, N. Luo, and J. Vehí, "Prediction of blood glucose levels and nocturnal hypoglycemia using physiological models and artificial neural networks.," in *KHD@ IJCAI*, pp. 85–90, 2018.

[74] S. Contador, J. M. Velasco, O. Garnica, and J. I. Hidalgo, "Glucose forecasting using genetic programming and latent glucose variability features," *Applied Soft Computing*, vol. 110, p. 107609, 2021.

[75] E. I. Georga, V. C. Protopappas, D. Polyzos, and D. I. Fotiadis, "A predictive model of subcutaneous glucose concentration in type 1 diabetes based on random forests," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 2889–2892, IEEE, 2012.

[76] M. Wadghiri, A. Idri, T. El Idrissi, and H. Hakkoum, "Ensemble blood glucose prediction in diabetes mellitus: A review," *Computers in Biology and Medicine*, p. 105674, 2022.

[77] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015.

[78] T. Zhu, K. Li, P. Herrero, and P. Georgiou, "Deep learning for diabetes: a systematic review," *IEEE Journal of Biomedical and Health Informatics*, vol. 25, no. 7, pp. 2744–2757, 2020.

[79] K. Li, J. Daniels, C. Liu, P. Herrero, and P. Georgiou, "Convolutional recurrent neural networks for glucose prediction," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 2, pp. 603–613, 2019.

[80] J. Daniels, P. Herrero, and P. Georgiou, "A multitask learning approach to personalized blood glucose prediction," *IEEE Journal of Biomedical and Health Informatics*, vol. 26, no. 1, pp. 436–445, 2021.

[81] M. De Bois, M. A. El Yacoubi, and M. Ammi, "Adversarial multi-source transfer learning in healthcare: Application to glucose prediction for diabetic people," *Computer Methods and Programs in Biomedicine*, vol. 199, p. 105874, 2021.

[82] K. Zarkogianni, K. Mitsis, E. Litsa, M.-T. Arredondo, G. Fico, A. Fioravanti, and K. S. Nikita, "Comparative assessment of glucose prediction models for patients with type 1 diabetes mellitus applying sensors for glucose and physical activity monitoring," *Medical & Biological Engineering & Computing*, vol. 53, no. 12, pp. 1333–1343, 2015.

[83] W. P. van Doorn, Y. D. Foreman, N. C. Schaper, H. H. Savelberg, A. Koster, C. J. van der Kallen, A. Wesselius, M. T. Schram, R. M. Henry, P. C. Dagnelie, *et al.*, "Machine learning-based glucose prediction with use of continuous glucose and physical activity monitoring data: The maastricht study," *PloS one*, vol. 16, no. 6, p. e0253125, 2021.

[84] R. N. Bergman, Y. Z. Ider, C. R. Bowden, and C. Cobelli, "Quantitative estimation of insulin sensitivity.," *American Journal of Physiology-Endocrinology And Metabolism*, vol. 236, no. 6, p. E667, 1979.

[85] R. Hovorka, V. Canonico, L. J. Chassin, U. Haueter, M. Massi-Benedetti, M. O. Federici, T. R. Pieber, H. C. Schaller, L. Schaupp, T. Vering, *et al.*, "Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes," *Physiological Measurement*, vol. 25, no. 4, p. 905, 2004.

[86] C. Dalla Man, R. A. Rizza, and C. Cobelli, "Meal simulation model of the glucose-insulin system," *IEEE Transactions on Biomedical Engineering*, vol. 54, no. 10, pp. 1740–1749, 2007.

[87] D. De Pereda, S. Romero-Vivo, B. Ricarte, and J. Bondia, "On the prediction of glucose concentration under intra-patient variability in type 1 diabetes: A monotone systems approach," *Computer Methods and Programs in Biomedicine*, vol. 108, no. 3, pp. 993–1001, 2012.

[88] A. J. Laguna, P. Rossetti, F. J. Ampudia-Blasco, J. Vehí, and J. Bondia, "Identification of intra-patient variability in the postprandial response of patients with type 1 diabetes," *Biomedical Signal Processing and Control*, vol. 12, pp. 39–46, 2014.

[89] R. Visentin, C. Dalla Man, and C. Cobelli, "One-day bayesian cloning of type 1 diabetes subjects: toward a single-day uva/padova type 1 diabetes simulator," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 11, pp. 2416–2424, 2016.

[90] C. Liu, J. Vehí, P. Avari, M. Reddy, N. Oliver, P. Georgiou, and P. Herrero, "Long-term glucose forecasting using a physiological model and deconvolution of the continuous glucose monitoring signal," *Sensors*, vol. 19, no. 19, p. 4338, 2019.

[91] A. Zhong, P. Choudhary, C. McMahon, P. Agrawal, J. B. Welsh, T. L. Cordero, and F. R. Kaufman, "Effectiveness of automated insulin management features of the minimed® 640g sensor-augmented insulin pump," *Diabetes Technology & Therapeutics*, vol. 18, no. 10, pp. 657–663, 2016.

[92] G. P. Forlenza, Z. Li, B. A. Buckingham, J. E. Pinsker, E. Cengiz, R. P. Wadwa, L. Ekhlaspour, M. M. Church, S. A. Weinzimer, E. Jost, *et al.*, "Predictive low-glucose suspend reduces hypoglycemia in adults, adolescents, and children with type 1 diabetes in an at-home randomized crossover study: results of the prolog trial," *Diabetes Care*, vol. 41, no. 10, pp. 2155–2161, 2018.

[93] S. Mirshekarian, H. Shen, R. Bunescu, and C. Marling, "Lstms and neural attention models for blood glucose prediction: Comparative experiments on real and synthetic data," in *2019 41st Annual International Con-*

ference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 706–712, IEEE, 2019.

[94] B. Kovatchev, P. Cheng, S. M. Anderson, J. E. Pinsker, F. Boscari, B. A. Buckingham, F. J. Doyle III, K. K. Hood, S. A. Brown, M. D. Breton, *et al.*, "Feasibility of long-term closed-loop control: a multicenter 6-month trial of 24/7 automated insulin delivery," *Diabetes Technology & Therapeutics*, vol. 19, no. 1, pp. 18–24, 2017.

[95] F. Prendin, S. Del Favero, M. Vettoretti, G. Sparacino, and A. Facchinetti, "Forecasting of glucose levels and hypoglycemic events: Head-to-head comparison of linear and nonlinear data-driven algorithms based on continuous glucose monitoring data only," *Sensors*, vol. 21, no. 5, p. 1647, 2021.

[96] C. C. Palerm, J. P. Willis, J. Desemone, and B. W. Bequette, "Hypoglycemia prediction and detection using optimal estimation," *Diabetes Technology & Therapeutics*, vol. 7, no. 1, pp. 3–14, 2005.

[97] C. C. Palerm and B. W. Bequette, "Hypoglycemia detection and prediction using continuous glucose monitoring—a study on hypoglycemic clamp data," *Journal of Diabetes Science and Technology*, vol. 1, no. 5, pp. 624–629, 2007.

[98] T. Zhu, K. Li, P. Herrero, J. Chen, and P. Georgiou, "A deep learning algorithm for personalized blood glucose prediction.," in *KHD@ IJCAI*, pp. 64–78, 2018.

[99] C. Zecchin, A. Facchinetti, G. Sparacino, and C. Cobelli, "Jump neural network for online short-time prediction of blood glucose from continuous monitoring sensors and meal information," *Computer Methods and Programs in Biomedicine*, vol. 113, no. 1, pp. 144–152, 2014.

[100] J. I. Hidalgo, J. M. Colmenar, G. Kronberger, S. M. Winkler, O. Garnica, and J. Lanchares, "Data based prediction of blood glucose concentrations using evolutionary methods," *Journal of Medical Systems*, vol. 41, no. 9, pp. 1–20, 2017.

[101] C. Zecchin, A. Facchinetti, G. Sparacino, and C. Cobelli, "How much is short-term glucose prediction in type 1 diabetes improved by adding

insulin delivery and meal content information to cgm data? a proof-of-concept study," *Journal of Diabetes Science and Technology*, vol. 10, no. 5, pp. 1149–1160, 2016.

[102] N. Allen and A. Gupta, "Current diabetes technology: striving for the artificial pancreas," *Diagnostics*, vol. 9, no. 1, p. 31, 2019.

[103] K. Zarkogianni, E. Litsa, K. Mitsis, P.-Y. Wu, C. D. Kaddi, C.-W. Cheng, M. D. Wang, and K. S. Nikita, "A review of emerging technologies for the management of diabetes mellitus," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 12, pp. 2735–2749, 2015.

[104] B. Buckingham, H. P. Chase, E. Dassau, E. Cobry, P. Clinton, V. Gage, K. Caswell, J. Wilkinson, F. Cameron, H. Lee, *et al.*, "Prevention of nocturnal hypoglycemia using predictive alarm algorithms and insulin pump suspension," *Diabetes Care*, vol. 33, no. 5, pp. 1013–1017, 2010.

[105] E. Dassau, F. Cameron, H. Lee, B. W. Bequette, H. Zisser, L. Jovanovič, H. P. Chase, D. M. Wilson, B. A. Buckingham, and F. J. Doyle, "Real-time hypoglycemia prediction suite using continuous glucose monitoring: a safety net for the artificial pancreas," *Diabetes Care*, vol. 33, no. 6, pp. 1249–1254, 2010.

[106] M. Frandes, B. Timar, R. Timar, and D. Lungeanu, "Chaotic time series prediction for glucose dynamics in type 1 diabetes mellitus using regime-switching models," *Scientific Reports*, vol. 7, no. 1, pp. 1–10, 2017.

[107] D. Joedicke, G. Kronberger, J. M. Colmenar, S. M. Winkler, J. M. Velasco, S. Contador, and J. I. Hidalgo, "Analysis of the performance of genetic programming on the blood glucose level prediction challenge 2020.," in *KDH@ ECAI*, pp. 141–145, 2020.

[108] R. McShinsky and B. Marshall, "Comparison of forecasting algorithms for type 1 diabetic glucose prediction on 30 and 60-minute prediction horizons.," in *KDH@ ECAI*, pp. 12–18, 2020.

[109] L. Ljung, "System identification: theory for the user," *PTR Prentice Hall, Upper Saddle River, NJ*, vol. 28, p. 540, 1999.

[110] G. Pillonetto, F. Dinuzzo, T. Chen, G. De Nicolao, and L. Ljung, "Kernel methods in system identification, machine learning and function estimation: A survey," *Automatica*, vol. 50, no. 3, pp. 657–682, 2014.

[111] A. Facchinetti, G. Sparacino, S. Guerra, Y. M. Luijf, J. H. DeVries, J. K. Mader, M. Ellmerer, C. Benesch, L. Heinemann, D. Bruttomesso, *et al.*, "Real-time improvement of continuous glucose monitoring accuracy: the smart sensor concept," *Diabetes Care*, vol. 36, no. 4, pp. 793–800, 2013.

[112] A. Facchinetti, G. Sparacino, E. Trifoglio, and C. Cobelli, "A new index to optimally design and compare continuous glucose monitoring glucose prediction algorithms," *Diabetes Technology & Therapeutics*, vol. 13, no. 2, pp. 111–119, 2011.

[113] F. Cameron, D. M. Wilson, B. A. Buckingham, H. Arzumanyan, P. Clinton, H. P. Chase, J. Lum, D. M. Maahs, P. M. Calhoun, and B. W. Bequette, "Inpatient studies of a kalman-filter-based predictive pump shutoff algorithm," *Journal of Diabetes Science and Technology*, vol. 6, no. 5, pp. 1142–1147, 2012.

[114] E. I. Georga, V. C. Protopappas, D. Ardigo, D. Polyzos, and D. I. Fotiadis, "A glucose model based on support vector regression for the prediction of hypoglycemic events under free-living conditions," *Diabetes Technology & Therapeutics*, vol. 15, no. 8, pp. 634–643, 2013.

[115] I. Rodríguez-Rodríguez, I. Chatzigiannakis, J.-V. Rodríguez, M. Maranghi, M. Gentili, and M.-Á. Zamora-Izquierdo, "Utility of big data in predicting short-term blood glucose levels in type 1 diabetes mellitus through machine learning techniques," *Sensors*, vol. 19, no. 20, p. 4482, 2019.

[116] T. El Idriss, A. Idri, I. Abnane, and Z. Bakkoury, "Predicting blood glucose using an lstm neural network," in *2019 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 35–41, IEEE, 2019.

[117] W. Gani, H. Taleb, and M. Limam, "Support vector regression based residual control charts," *Journal of Applied Statistics*, vol. 37, no. 2, pp. 309–324, 2010.

[118] W. Wang, Z. Xu, W. Lu, and X. Zhang, "Determination of the spread parameter in the gaussian kernel for classification and regression," *Neurocomputing*, vol. 55, no. 3-4, pp. 643–663, 2003.

[119] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan, and B. P. Feuston, "Random forest: a classification and regression tool for com-

pound classification and qsar modeling," *Journal of Chemical Information and Computer Sciences*, vol. 43, no. 6, pp. 1947–1958, 2003.

[120] P. M. Sathe and J. Venitz, "Comparison of neural network and multiple linear regression as dissolution predictors," *Drug Development and Industrial Pharmacy*, vol. 29, no. 3, pp. 349–355, 2003.

[121] S. Siami-Namini, N. Tavakoli, and A. S. Namin, "A comparison of arima and lstm in forecasting time series," in *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 1394–1401, IEEE, 2018.

[122] T. Hastie, R. Tibshirani, and J. Friedman, *The elements of statistical learning: data mining, inference, and prediction*. Springer Science & Business Media, 2009.

[123] T. Saito and M. Rehmsmeier, "The precision-recall plot is more informative than the roc plot when evaluating binary classifiers on imbalanced datasets," *Plos One*, vol. 10, no. 3, p. e0118432, 2015.

[124] R. P. Wadwa, L. M. Laffel, V. N. Shah, and S. K. Garg, "Accuracy of a factory-calibrated, real-time continuous glucose monitoring system during 10 days of use in youth and adults with diabetes," *Diabetes Technology & Therapeutics*, vol. 20, no. 6, pp. 395–402, 2018.

[125] E. Daskalaki, K. Nørgaard, T. Züger, A. Prountzou, P. Diem, and S. Mougiakakou, "An early warning system for hypoglycemic/hyperglycemic events based on fusion of adaptive prediction models," *Journal of Diabetes Science and Technology*, vol. 7, no. 3, pp. 689–698, 2013.

[126] M. Frandes, B. Timar, and D. Lungeanu, "A risk based neural network approach for predictive modeling of blood glucose dynamics," in *Exploring Complexity in Health: An Interdisciplinary Systems Approach*, pp. 577–581, IOS Press, 2016.

[127] F. Prendin, J.-L. Díez, S. Del Favero, G. Sparacino, A. Facchinetti, and J. Bondia, "Assessment of seasonal stochastic local models for glucose prediction without meal size information under free-living conditions," *Sensors*, vol. 22, no. 22, p. 8682, 2022.

[128] S. Hylleberg, *Modelling seasonality*. Oxford University Press, 1992.

[129] E. Montaser, J.-L. Díez, and J. Bondia, "Stochastic seasonal models for glucose prediction in the artificial pancreas," *Journal of Diabetes Science and Technology*, vol. 11, no. 6, pp. 1124–1131, 2017.

[130] E. Montaser, J.-L. Díez, P. Rossetti, M. Rashid, A. Cinar, and J. Bondia, "Seasonal local models for glucose prediction in type 1 diabetes," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, no. 7, pp. 2064–2072, 2019.

[131] E. Montaser, J.-L. Díez, and J. Bondia, "Glucose prediction under variable-length time-stamped daily events: A seasonal stochastic local modeling framework," *Sensors*, vol. 21, no. 9, p. 3188, 2021.

[132] C. D. Man, F. Micheletto, D. Lv, M. Breton, B. Kovatchev, and C. Cobelli, "The uva/padova type 1 diabetes simulator: new features," *Journal of Diabetes Science and Technology*, vol. 8, no. 1, pp. 26–34, 2014.

[133] A. Ceriello, L. Monnier, and D. Owens, "Glycaemic variability in diabetes: clinical and therapeutic implications," *The Lancet Diabetes & endocrinology*, vol. 7, no. 3, pp. 221–230, 2019.

[134] W. Wang and Y. Zhang, "On fuzzy cluster validity indices," *Fuzzy Sets and Systems*, vol. 158, no. 19, pp. 2095–2117, 2007.

[135] J. K. Dixon, "Pattern recognition with partly missing data," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 9, no. 10, pp. 617–621, 1979.

[136] S. Sengupta, S. De, A. Konar, and R. Janarthanan, "An improved fuzzy clustering method using modified fukuyama-sugeno cluster validity index," in *2011 International Conference on Recent Trends in Information Systems*, pp. 269–274, IEEE, 2011.

[137] C. Roversi, M. Vettoretti, S. Del Favero, A. Facchinetti, G. Sparacino, and H.-R. Consortium, "Modeling carbohydrate counting error in type 1 diabetes management," *Diabetes Technology & Therapeutics*, vol. 22, no. 10, pp. 749–759, 2020.

[138] S. Faccioli*, F. Prendin*, A. Facchinetti, G. Sparacino, and S. Del Favero, "Combined use of glucose-specific model identification and alarm strategy based on prediction-funnel to improve online forecasting

of hypoglycemic events," *Journal of Diabetes Science and Technology*, p. 19322968221093665, 2022, * authors contributed equally.

[139] M. Eren-Oruklu, A. Cinar, and L. Quinn, "Hypoglycemia prediction with subject-specific recursive time-series models," *Journal of Diabetes Science and Technology*, vol. 4, no. 1, pp. 25–33, 2010. PMID: 20167164.

[140] C. Toffanin, S. Del Favero, E. Aiello, M. Messori, C. Cobelli, and L. Magni, "Glucose-insulin model identified in free-living conditions for hypoglycaemia prevention," *Journal of Process Control*, vol. 64, pp. 27–36, 2018.

[141] D. A. Finan, H. Zisser, L. Jovanovic, W. C. Bevier, and D. E. Seborg, "Identification of linear dynamic models for type 1 diabetes: a simulation study," *IFAC Proceedings Volumes*, vol. 39, no. 2, pp. 503–508, 2006.

[142] M. Messori, M. Ellis, C. Cobelli, P. D. Christofides, and L. Magni, "Improved postprandial glucose control with a customized model predictive controller," in *2015 American control conference (ACC)*, pp. 5108–5115, IEEE, 2015.

[143] K. Turksoy, J. Kilkus, I. Hajizadeh, S. Samadi, J. Feng, M. Sevil, C. Lazaro, N. Frantz, E. Littlejohn, and A. Cinar, "Hypoglycemia detection and carbohydrate suggestion in an artificial pancreas," *Journal of Diabetes Science and Technology*, vol. 10, no. 6, pp. 1236–1244, 2016.

[144] X. Mo, Y. Wang, and X. Wu, "Hypoglycemia prediction using extreme learning machine (elm) and regularized elm," in *2013 25th Chinese Control and Decision Conference (CCDC)*, pp. 4405–4409, IEEE, 2013.

[145] K. Turksoy, E. S. Bayrak, L. Quinn, E. Littlejohn, D. Rollins, and A. Cinar, "Hypoglycemia early alarm systems based on multivariable models," *Industrial & Engineering Chemistry Research*, vol. 52, no. 35, pp. 12329–12336, 2013.

[146] O. Mujahid, I. Contreras, and J. Vehi, "Machine learning techniques for hypoglycemia prediction: Trends and challenges," *Sensors*, vol. 21, no. 2, p. 546, 2021.

[147] H. Efendic, H. Kirchsteiger, G. Freckmann, and L. del Re, "Short-term prediction of blood glucose concentration using interval probabilistic models," in *22nd Mediterranean conference on control and automation*, pp. 1494–1499, IEEE, 2014.

[148] J. Vehí, I. Contreras, S. Oviedo, L. Biagi, and A. Bertachi, "Prediction and prevention of hypoglycaemic events in type-1 diabetic patients using machine learning," *Health Informatics Journal*, vol. 26, no. 1, pp. 703–718, 2020.

[149] D. Dave, D. J. DeSalvo, B. Haridas, S. McKay, A. Shenoy, C. J. Koh, M. Lawley, and M. Erraguntla, "Feature-based machine learning model for real-time hypoglycemia prediction," *Journal of Diabetes Science and Technology*, vol. 15, no. 4, pp. 842–855, 2021. PMID: 32476492.

[150] K. S. Eljil, G. Qadah, and M. Pasquier, "Predicting hypoglycemia in diabetic patients using data mining techniques," in *2013 9th International Conference on Innovations in Information Technology (IIT)*, pp. 130–135, IEEE, 2013.

[151] G. F. Franklin, J. D. Powell, M. L. Workman, *et al.*, *Digital control of dynamic systems*, vol. 3. Addison-wesley Reading, MA, 1998.

[152] S. Del Favero, A. Facchinetti, and C. Cobelli, "A glucose-specific metric to assess predictors and identify models," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 5, pp. 1281–1290, 2012.

[153] W. L. Clarke, D. Cox, L. A. Gonder-Frederick, W. Carter, and S. L. Pohl, "Evaluating Clinical Accuracy of Systems for Self-Monitoring of Blood Glucose," *Diabetes Care*, vol. 10, pp. 622–628, 09 1987.

[154] T. Danne, R. Nimri, T. Battelino, R. M. Bergenstal, K. L. Close, J. H. DeVries, S. Garg, L. Heinemann, I. Hirsch, S. A. Amiel, R. Beck, E. Bosi, B. Buckingham, C. Cobelli, E. Dassau, F. J. Doyle, S. Heller, R. Hovorka, W. Jia, T. Jones, O. Kordonouri, B. Kovatchev, A. Kowalski, L. Laffel, D. Maahs, H. R. Murphy, K. Nørgaard, C. G. Parkin, E. Renard, B. Saboo, M. Scharf, W. V. Tamborlane, S. A. Weinzimer, and M. Phillip, "International consensus on use of continuous glucose monitoring," *Diabetes Care*, vol. 40, no. 12, pp. 1631–1640, 2017.

[155] A. Yeh, "More accurate tests for the statistical significance of result differences," in *COLING 2000 Volume 2: The 18th International Conference on Computational Linguistics*, 2000.

[156] I. Hajizadeh, M. Rashid, K. Turksoy, S. Samadi, J. Feng, M. Sevil, N. Frantz, C. Lazaro, Z. Maloney, E. Littlejohn, *et al.*, "Multivariable recursive subspace identification with application to artificial pancreas systems," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 886–891, 2017.

[157] M. Eren-Oruklu, A. Cinar, L. Quinn, and D. Smith, "Estimation of future glucose concentrations with subject-specific recursive linear models," *Diabetes Technology & Therapeutics*, vol. 11, no. 4, pp. 243–253, 2009.

[158] R. N. Bergman, L. S. Phillips, C. Cobelli, *et al.*, "Physiologic evaluation of factors controlling glucose tolerance in man: measurement of insulin sensitivity

and beta-cell glucose sensitivity from the response to intravenous glucose.," *The Journal of Clinical Investigation*, vol. 68, no. 6, pp. 1456–1467, 1981.

[159] M. E. Wilinska, L. J. Chassin, C. L. Acerini, J. M. Allen, D. B. Dunger, and R. Hovorka, "Simulation environment to evaluate closed-loop insulin delivery systems in type 1 diabetes," *Journal of Diabetes Science and Technology*, vol. 4, no. 1, pp. 132–144, 2010.

[160] S. S. Kanderian, S. Weinzimer, G. Voskanyan, and G. M. Steil, "Identification of intraday metabolic profiles during closed-loop glucose control in individuals with type 1 diabetes," *Journal of Diabetes Science and Technology*, vol. 3, no. 5, pp. 1047–1057, 2009. PMID: 20144418.

[161] R. Visentin, E. Campos-Náñez, M. Schiavon, D. Lv, M. Vettoretti, M. Breton, B. P. Kovatchev, C. D. Man, and C. Cobelli, "The uva/padova type 1 diabetes simulator goes from single meal to single day," *Journal of Diabetes Science and Technology*, vol. 12, no. 2, pp. 273–281, 2018. PMID: 29451021.

[162] W. R. Gilks, S. Richardson, and D. Spiegelhalter, *Markov chain Monte Carlo in practice*. CRC press, 1995.

[163] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on signal processing*, vol. 50, no. 2, pp. 174–188, 2002.

[164] P. Van den Hof, "Closed-loop issues in system identification," *Annual Reviews in Control*, vol. 22, pp. 173–186, 1998.

[165] G. Cappon, M. Vettoretti, G. Sparacino, and A. Facchinetti, "Replaybg: a novel simulation methodology to assess algorithms for type 1 diabetes treatment on retrospective data," *Submitted on IEEE Transactions on Biomedical Engineering*, 2021.

[166] M. Schiavon, C. Dalla Man, and C. Cobelli, "Modeling subcutaneous absorption of fast-acting insulin in type 1 diabetes," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 9, pp. 2079–2086, 2017.

[167] C. Dalla Man, M. Camilleri, and C. Cobelli, "A system model of oral glucose absorption: validation on gold standard data," *IEEE Transactions on Biomedical Engineering*, vol. 53, no. 12, pp. 2472–2478, 2006.

[168] B. P. Kovatchev, L. S. Farhy, D. J. Cox, M. Straume, V. I. Yankov, L. A. Gonder-Frederick, and W. L. Clarke, "Modeling insulin-glucose dynamics during insulin induced hypoglycemia. evaluation of glucose counter-regulation," *Computational and Mathematical Methods in Medicine*, vol. 1, no. 4, pp. 313–323, 1999.

[169] R. Visentin, C. Dalla Man, Y. C. Kudva, A. Basu, and C. Cobelli, "Circadian variability of insulin sensitivity: physiological input for in silico artificial pancreas," *Diabetes Technology & Therapeutics*, vol. 17, no. 1, pp. 1–7, 2015.

[170] C. Dalla Man, A. Caumo, R. Basu, R. Rizza, G. Toffolo, and C. Cobelli, "Minimal model estimation of glucose absorption and insulin sensitivity from oral test: Validation with a tracer method," *American Journal of Physiology - Endocrinology and Metabolism*, vol. 287, no. 4 50-4, pp. E637–E643, 2004.

[171] J. T. Connor, R. D. Martin, and L. E. Atlas, "Recurrent neural networks and robust time series prediction," *IEEE Transactions on Neural Networks*, vol. 5, no. 2, pp. 240–254, 1994.

[172] B. Zhao, H. Lu, S. Chen, J. Liu, and D. Wu, "Convolutional neural networks for time series classification," *Journal of Systems Engineering and Electronics*, vol. 28, no. 1, pp. 162–169, 2017.

[173] M. Munoz-Organero, "Deep physiological model for blood glucose prediction in t1dm patients," *Sensors*, vol. 20, no. 14, p. 3896, 2020.

[174] M. Zhang, K. B. Flores, and H. T. Tran, "Deep learning and regression approaches to forecasting blood glucose levels for type 1 diabetes," *Biomedical Signal Processing and Control*, vol. 69, p. 102923, 2021.

[175] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu, and H. Zhang, "Deep learning with long short-term memory for time series prediction," *IEEE Communications Magazine*, vol. 57, no. 6, pp. 114–119, 2019.

[176] P. T. Yamak, L. Yujian, and P. K. Gadosey, "A comparison between arima, lstm, and gru for time series forecasting," in *Proceedings of the 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence*, pp. 49–55, 2019.

[177] Y. Liu, H. Dong, X. Wang, and S. Han, "Time series prediction based on temporal convolutional network," in *2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS)*, pp. 300–305, IEEE, 2019.

[178] T. Zhu, L. Kuang, J. Daniels, P. Herrero, K. Li, and P. Georgiou, "Iomt-enabled real-time blood glucose prediction with deep learning and edge computing," *IEEE Internet of Things Journal*, 2022.

[179] S. W. Lee, M. Cao, S. Sajid, M. Hayes, L. Choi, C. Rother, and R. De León, "The dual-wave bolus feature in continuous subcutaneous insulin infusion pumps controls prolonged post-prandial hyperglycaemia better than standard bolus in

type 1 diabetes.," *Diabetes, Nutrition & Metabolism*, vol. 17, no. 4, pp. 211–216, 2004.

[180] M. Rashid, S. Samadi, M. Sevil, I. Hajizadeh, P. Kolodziej, N. Hobbs, Z. Maloney, R. Brandt, J. Feng, M. Park, *et al.*, "Simulation software for assessment of nonlinear and adaptive multivariable control algorithms: glucose–insulin dynamics in type 1 diabetes," *Computers & Chemical Engineering*, vol. 130, p. 106565, 2019.

[181] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," *Artificial Intelligence*, vol. 267, pp. 1–38, 2019.

[182] K.-H. Yu, A. L. Beam, and I. S. Kohane, "Artificial intelligence in healthcare," *Nature Biomedical Engineering*, vol. 2, no. 10, pp. 719–731, 2018.

[183] H. Hagras, "Toward human-understandable, explainable ai," *Computer*, vol. 51, no. 9, pp. 28–36, 2018.

[184] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[185] D. Alvarez-Melis and T. S. Jaakkola, "On the robustness of interpretability methods," *arXiv preprint arXiv:1806.08049*, 2018.

[186] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *International conference on machine learning*, pp. 3145–3153, PMLR, 2017.

[187] A. Kumar, K. Sehgal, P. Garg, V. Kamakshi, and N. C. Krishnan, "Mace: Model agnostic concept extractor for explaining image classification networks," *IEEE Transactions on Artificial Intelligence*, vol. 2, no. 6, pp. 574–583, 2021.

[188] X. Chen, Y. Duan, R. Houthooft, J. Schulman, I. Sutskever, and P. Abbeel, "Infogan: Interpretable representation learning by information maximizing generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 29, 2016.

[189] G. Cappon, L. Meneghetti, F. Prendin, J. Pavan, G. Sparacino, S. Del Favero, and A. Facchinetti, "A personalized and interpretable deep learning based approach to predict blood glucose concentration in type 1 diabetes.," in *KDH@ ECAI*, pp. 75–79, 2020.

[190] M. De Bois, M. A. El Yacoubi, and M. Ammi, "Interpreting deep glucose predictive models for diabetic people using retain," in *International Conference on Pattern Recognition and Artificial Intelligence*, pp. 685–694, Springer, 2020.

[191] H. Rubin-Falcone, I. Fox, and J. Wiens, "Deep residual time-series forecasting: Application to blood glucose prediction," in *KDH@ ECAI*, 2020.

[192] K. Aas, M. Jullum, and A. Løland, "Explaining individual predictions when features are dependent: More accurate approximations to shapley values," *Artificial Intelligence*, vol. 298, p. 103502, 2021.

[193] C. Ellingsen, E. Dassau, H. Zisser, B. Grosman, M. W. Percival, L. Jovanovič, and F. J. Doyle III, "Safety constraints in an artificial pancreatic β cell: an implementation of model predictive control with insulin on board," *Journal of Diabetes Science and Technology*, vol. 3, no. 3, pp. 536–544, 2009.

[194] M. Schiavon, C. Dalla Man, Y. C. Kudva, A. Basu, and C. Cobelli, "Quantitative estimation of insulin sensitivity in type 1 diabetic subjects wearing a sensor-augmented insulin pump," *Diabetes Care*, vol. 37, no. 5, pp. 1216–1223, 2014.

[195] M. Schiavon, C. Dalla Man, and C. Cobelli, "Insulin sensitivity index-based optimization of insulin to carbohydrate ratio: in silico study shows efficacious protection against hypoglycemic events caused by suboptimal therapy," *Diabetes Technology & Therapeutics*, vol. 20, no. 2, pp. 98–105, 2018.

[196] J. Martinsson, A. Schliep, B. Eliasson, C. Meijner, S. Persson, and O. Mogren, "Automatic blood glucose prediction with confidence using recurrent neural networks," in *KHD@IJCAI*, 2018.

[197] G. Pillonetto, G. Sparacino, and C. Cobelli, "Numerical non-identifiability regions of the minimal model of glucose kinetics: superiority of bayesian estimation," *Mathematical Biosciences*, vol. 184, no. 1, pp. 53–67, 2003.

[198] H. Haario, E. Saksman, and J. Tamminen, "Adaptive proposal distribution for random walk metropolis algorithm," *Computational Statistics*, vol. 14, pp. 375–395, 1999.

[199] G. Kitagawa, "Monte carlo filter and smoother for non-gaussian nonlinear state space models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 1–25, 1996.

[200] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[201] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," *arXiv preprint arXiv:1406.1078*, 2014.

[202] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[203] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*, vol. 2. MIT press Cambridge, MA, 2006.

[204] F. Dinuzzo, M. Neve, G. De Nicolao, and U. P. Gianazza, "On the representer theorem and equivalent degrees of freedom of SVR," *Journal of Machine Learning Research*, vol. 8, pp. 2467–2495, 2007.

[205] F. Dinuzzo and G. De Nicolao, "An algebraic characterization of the optimum of regularized kernel methods," *Machine Learning*, vol. 74, no. 3, pp. 315–345, 2009.

[206] G. Pillonetto, A. Chiuso, and G. De Nicolao, "Predictor estimation via Gaussian regression," *Proceedings of the IEEE Conference on Decision and Control*, pp. 744–749, 2008.

[207] G. Pillonetto, A. Chiuso, and G. De Nicolao, "Prediction error identification of linear systems: A. nonparametric Gaussian regression approach," *Automatica*, vol. 47, no. 2, pp. 291–305, 2011.