UNIVERSITÀ
DEGLI STUDI
DI PADOVA

DIPARTIMENTO
DI INGEGNERIA
DELL'INFORMAZIONE

Ph.D. School in Information Engineering
Section of Bioengineering
XXVI Series

Online Glucose Prediction in Type 1 Diabetes by Neural Network Models

School Director
Prof. Matteo Bertocco

Bioengineering Coordinator
Prof. Giovanni Sparacino

Advisor
Prof. Giovanni Sparacino

Ph.D. Candidate
Chiara Zecchin

A thesis submitted for the degree of
philosopiæ doctor (PhD)
January 2014

Summary

Diabetes mellitus is a chronic disease characterized by dysfunctions of the normal regulation of glucose concentration in the blood. In Type 1 diabetes the pancreas is unable to produce insulin, while in Type 2 diabetes derangements in insulin secretion and action occur. As a consequence, glucose concentration often exceeds the normal range (70-180 mg/dL), with short- and long-term complications. Hypoglycemia (glycemia below 70 mg/dL) can progress from measurable cognition impairment to aberrant behaviour, seizure and coma. Hyperglycemia (glycemia above 180 mg/dL) predisposes to invalidating pathologies, such as neuropathy, nephropathy, retinopathy and diabetic foot ulcers. Conventional diabetes therapy aims at maintaining glycemia in the normal range by tuning diet, insulin infusion and physical activity on the basis of 4-5 daily self-monitoring of blood glucose (SMBG) measurements, obtained by the patient using portable minimally-invasive lancing sensor devices. New scenarios in diabetes treatment have been opened in the last 15 years, when minimally invasive continuous glucose monitoring (CGM) sensors, able to monitor glucose concentration in the subcutis continuously (i.e. with a reading every 1 to 5 min) over several days (7-10 consecutive days), entered clinical research. CGM allows tracking glucose dynamics much more effectively than SMBG and glycemic time-series can be used both retrospectively, e.g. to optimize metabolic control therapy, and in real-time applications, e.g. to generate alerts when glucose concentration exceeds the normal range thresholds or in the so-called "artificial pancreas", as inputs of the closed loop control algorithm. For what concerns real time applications, the possibility of preventing critical events is, clearly, even more appealing than just detecting them as they occur. This would be doable if glucose concentration were known in advance, approximately 30-45 min ahead in time. The quasi continuous nature of the CGM signal renders feasible the use of prediction algorithms which could allow the patient to take therapeutic decisions on the basis of future instead of current glycemia, possibly mitigating/ avoiding imminent critical events. Since the introduction of CGM devices, various methods for short-time prediction of glucose concentration have been proposed in the literature. They are mainly based on black box time series models and the majority of them uses only the history of the CGM signal as input. However, glucose dynamics are influenced by many factors, e.g. quantity of ingested carbohydrates, administration of drugs including insulin, physical activity, stress, emotions and inter- and intra-individual variability is high. For these reasons, prediction of glucose time course is a challenging topic and results obtained so far may be improved.

The aim of this thesis is to investigate the possibility of predicting future glucose concentration, in the short term, using new models based on neural networks (NN)

exploiting, apart from CGM history, other available information. In particular, we first develop an original model which uses, as inputs, the CGM signal and information on timing and carbohydrate content of ingested meals. The prediction algorithm is based on a feedforward NN in parallel with a linear predictor. Results are promising: the predictor outperforms widely used state of art techniques and forecasts are accurate and allow obtaining a satisfactory time anticipation. Then we propose a second model, which exploits a different NN architecture, a jump NN, which combines benefits of both feedforward NN and linear algorithm obtaining performance similar to the previously developed predictor, although the simpler structure. To conclude the analysis, information on doses of injected bolus of insulin are added as input of the jump NN and the relative importance of every input signal in determining the NN output is investigated by developing an original sensitivity analysis. All the proposed predictors are assessed on real data of Type 1 diabetics, collected during the European FP7 project DIAdvisor$^{\text{TM}}$. To evaluate the clinical usefulness of prediction in improving diabetes management we also propose a new strategy to quantify, using an *in silico* environment, the reduction of hypoglycemia when alerts and relative therapy are triggered on the basis of prediction, obtained with our NN algorithm, instead of CGM. Finally, possible inclusion of additional pieces of information such as physical activity is investigated, though at a preliminary level.

The thesis is organized as follows. Chapter 1 gives an introduction to the diabetes disease and the current technologies for CGM, presents state of art techniques for short-time prediction of glucose concentration of diabetics and states the aim and the novelty of the thesis. Chapter 2 discusses NN paradigms from a theoretical point of view and specifies technical details common to the design and implementation of all the NN algorithms proposed in the following. Chapter 3 describes the first prediction model we propose, based on a NN in parallel with a linear algorithm. Chapter 4 presents an alternative simpler architecture, based on a jump NN, and demonstrates its equivalence, in terms of performance, with the previously proposed algorithm. Chapter 5 further improves the jump NN, by adding new inputs and investigating their effective utility by a sensitivity analysis. Chapter 6 points out possible future developments, as the possibility of exploiting information on physical activity, reporting also a preliminary analysis. Finally, Chapter 7 describes the application of NN for generation of preventive hypoglycemic alerts and evaluates improvement of diabetes management in a simulated environment. Some concluding remarks end the thesis.

Sommario

Il diabete mellito è una patologia cronica caratterizzata da disfunzioni della regolazione della concentrazione di glucosio nel sangue. Nel diabete di Tipo 1 il pancreas non produce l'ormone insulina, mentre nel diabete di Tipo 2 si verificano squilibri nella secrezione e nell'azione dell'insulina. Di conseguenza, spesso la concentrazione glicemica eccede le soglie di normalità (70-180 mg/dL), con complicazioni a breve e lungo termine. L'ipoglicemia (glicemia inferiore a 70 mg/dL) può risultare in alterazione delle capacità cognitive, cambiamenti d'umore, convulsioni e coma. L'iperglicemia (glicemia superiore a 180 mg/dL) predispone, nel lungo termine, a patologie invalidanti, come neuropatie, nefropatie, retinopatie e piede diabetico. L'obiettivo della terapia convenzionale del diabete è il mantenimento della glicemia nell'intervallo di normalità regolando la dieta, la terapia insulinica e l'esercizio fisico in base a 4-5 monitoraggi giornalieri della glicemia, (Self-Monitoring of Blood Glucose, SMBG), effettuati dal paziente stesso usando un dispositivo pungidito, portabile e minimamente invasivo. Negli ultimi 15 anni si sono aperti nuovi orizzonti nel trattamento del diabete, grazie all'introduzione, nella ricerca clinica, di sensori minimamente invasivi (Continuous Glucose Monitoring, CGM) capaci di misurare la glicemia nel sottocute in modo quasi continuo (ovvero con una misurazione ogni 1-5 min) per parecchi giorni consecutivi (dai 7 ai 10 giorni). I sensori CGM permettono di monitorare le dinamiche glicemiche in modo più fine delle misurazioni SMBG e le serie temporali di concentrazione glicemica possono essere utilizzate sia retrospettivamente, per esempio per ottimizzare la terapia di controllo metabolico, sia prospettivamente in tempo reale, per esempio per generare segnali di allarme quando la concentrazione glicemica oltrepassa le soglie di normalità o nel "pancreas artificiale". Per quanto concerne le applicazioni in tempo reale, poter prevenire gli eventi critici sarebbe chiaramente più attraente che semplicemente individuarli, contestualmente al loro verificarsi. Ciò sarebbe fattibile se si conoscesse la concentrazione glicemia futura con circa 30-45 min di anticipo. La natura quasi continua del segnale CGM rende possibile l'uso di algoritmi predittivi che possono, potenzialmente, permettere ai pazienti diabetici di ottimizzare le decisioni terapeutiche sulla base della glicemia futura, invece che attuale, dando loro l'oppurtunità di limitare l'impatto di eventi pericolosi per la salute, se non di evitarli. Dopo l'introduzione nella pratica clinica dei dispositivi CGM, in letteratura, sono stati proposti vari metodi per la predizione a breve termine della glicemia. Si tratta principalmente di algoritmi basati su modelli di serie temporali e la maggior parte di essi utilizza solamente la storia del segnale CGM come ingresso. Tuttavia, le dinamiche glicemiche sono determinate da molti fattori, come la quantità di carboidrati ingeriti durante i pasti, la somministrazione di farmaci, compresa l'insulina, l'attività fisica, lo

stress, le emozioni. Inoltre, la variabilità inter- e intra- individuale è elevata. Per questi motivi, predire l'andamento glicemico futuro è difficile e stimolante e c'è margine di miglioramento dei risultati pubblicati finora in letteratura.

Lo scopo di questa tesi è investigare la possibilità di predire la concentrazione glicemica futura, nel breve termine, utilizzando modelli basati su reti neurali (Neural Network, NN) e sfruttando, oltre alla storia del segnale CGM, altre informazioni disponibili. Nel dettaglio, inizialmente svilupperemo un nuovo modello che utilizza, come ingressi, il segnale CGM e informazioni relative ai pasti ingeriti, (istante temporale e quantità di carboidrati). L'algoritmo predittivo sarà basato su una NN di tipo feedforward, in parallelo ad un modello lineare. I risultati sono promettenti: il modello è superiore ad algoritmi stato dell'arte ampiamente utilizzati, la predizione è accurata e il guadagno temporale è soddisfacente. Successivamente proporremo un nuovo modello basato su una differente architettura di NN, ovvero una "jump NN", che fonde i benefici di una NN di tipo feedforward e di un algoritmo lineare, ottenendo risultati simili a quelli del modello precedentemente proposto, nonostante la sua struttura notevolmente più semplice. Per completare l'analisi, valuteremo l'inclusione, tra gli ingressi della jump NN, di segnali ottenuti sfruttando informazioni sulla terapia insulinica (istante temporale e dose dei boli iniettati) e valuteremo l'importanza e l'influenza relativa di ogni ingresso nella determinazione del valore glicemico predetto dalla NN, sviluppando un'originale analisi di sensitività. Tutti i modelli proposti saranno valutati su dati reali di pazienti diabetici di Tipo 1, raccolti durante il progetto Europeo FP7 (7$^{\text{th}}$ Framework Programme, Settimo Programma Quadro) DIAdvisor$^{\text{TM}}$. Per valutare l'utilità clinica della predizione e il miglioramento della gestione della terapia diabetica proporremo una nuova strategia per la quantificazione, in simulazione, della riduzione del numero e della gravità degli eventi ipoglicemici nel caso gli allarmi, e la relativa terapia, siano determinati sulla base della concentrazione glicemica predetta, utilizzando il nostro algoritmo basato su NN, invece che su quella misurata dal sensore CGM. Infine, investigheremo, in modo preliminare, la possibilità di includere, tra gli ingressi della NN, ulteriori informazioni, come l'attività fisica.

La tesi è organizzata come descritto in seguito. Il Capitolo 1 introduce la patologia diabetica e le attuali tecnologie CGM, presenta le tecniche stato dell'arte utilizzate per la predizione a breve termine della glicemia di pazienti diabetici e specifica gli scopi e le innovazioni della presente tesi. Il Capitolo 2 introduce le basi teoriche delle NN e specifica i dettagli tecnici che abbiamo scelto di adottare per lo sviluppo e l'implementazione di tutte le NN proposte in seguito. Il Capitolo 3 descrive il primo modello proposto, basato su una NN in parallelo a un algoritmo lineare. Il Capitolo 4 presenta una struttura alternativa più semplice, basata su una jump NN, e dimostra la sua equivalenza, in termini di prestazioni, con il modello precedentemente proposto. Il Capitolo 5 apporta ulteriori miglioramenti alla jump NN, aggiungendo nuovi ingressi e investigando la loro utilità effettiva attraverso un'analisi di sensitività. Il Capitolo 6 indica possibili sviluppi futuri, come l'inclusione di informazioni sull'attività fisica, presentando anche un'analisi preliminare. Infine, il Capitolo 7 applica la NN per la generazione di allarmi preventivi per l'ipoglicemia, valutando, in simulazione, il miglioramento della gestione del diabete. Alcuni commenti e osservazioni concludono la tesi.

List of Abbreviations

AP Artificial Pancreas

AR Auto-Regressive

ARMA Auto-Regressive with Moving Average

ARMAX Auto-Regressive with Moving Average and eXogenous Inputs

ARX Auto-Regressive with eXogenous Inputs

BG Blood Glucose

CE Conformité Européenne

CG-EGA Continuous Glucose - Error Grid Analysis

CGM Continuous Glucose Monitoring

CHO Carbohydrate

EGA Error Grid Analysis

ESOD Energy of Second Order Derivative

FDA Food and Drug Administration

FFNN FeedForward Neural Network

GA Genetic Algorithm

HBGI High Blood Glucose Index

IDDM Insulin Dependent Diabetes Mellitus

LBGI Low Blood Glucose Index

LS Least Squares

MAE Mean Absolute Error

MSE Mean Square Error

NN Neural Network

NIDDM Non-Insulin Dependent Diabetes Mellitus

PA Physical Activity

PAMS Physical Activity Monitoring System

PH Prediction Horizon

RAD Relative Absolute Difference

RLS Recursive Least Squares

RMSE Root Mean Square Error

SMBG Self-Monitoring Blood Glucose

SSE Sum of Squared Errors

TG Time Gain

T1D Type 1 Diabetes

T2D Type 2 Diabetes

WHO World Health Organization

Contents

1

Diabetes and Continuous Glucose Monitoring (CGM)

According to the World Health Organization (WHO) 347 million people worldwide have diabetes [1]. In 2004, an estimated 3.4 million people died from consequences of high fasting blood sugar (more than 80% in low- and middle-income countries) and WHO projects that diabetes will be the 7th leading cause of death in 2030. From an economic point of view, diabetes costs were estimated in $ 245 billion in 2012 in the US [2], while they ranges from 6 to 14% of the total health expenditure in EU countries [3]. This explains why diabetes is considered one of the most challenging socio-health emergencies of the 3rd millennium [4] and also why the impact of innovative methodologies and technologies for diabetes monitoring and treatment can be extremely high. This chapter gives an overview of the diabetes disease and of its therapy. In this context, the potential clinical importance of the Continuous Glucose Monitoring (CGM) sensors, appeared in the market in the early 2000s, is highlighted, together with a short description of minimally invasive and non invasive CGM devices.

1.1 The diabetes mellitus disease

1.1.1 Glucose-insulin regulatory system

In human beings, glucose represents the basic nutrition factor for the muscles and the only energy source for the brain. Glucose reaches the blood stream via several mechanisms (released by the intestine after a meal, or produced by the liver and, in small part, by the kidneys in fasting conditions) and is then absorbed by tissues either via hormone-mediated mechanisms (e.g. by the muscles) or via non-mediated transportation (e.g. by the brain). Thanks to a complex hormonal regulatory mechanism, glucose concentration in blood of healthy subjects is tightly kept in a limited rage, i.e. 70-180 mg/dL, although it fluctuates due to utilization and production processes. Different hormones are involved in this regulation: the most important is insulin, which is produced by the beta-cells of the pancreas, and is responsible for lowering glucose concentration in blood after a meal by facilitating the uptake of glucose by the muscles, by suppressing the hepatic production of glucose by the liver and by controlling the conversion of glucose into glycogen for internal storage in the liver [5]. If the glycemia decreases and sufficient nutrients delivery to the tissues is not guaranteed, counter-regulatory hormones, such as glucagon, are secreted and stimulate the conversion of glycogen to glucose, allowing to keep the concentration of glucose in the safety range [6].

Figure 1.1 shows a rough description of glucose-insulin regulatory system. Glucose is used by many organs, tissues and cells. Some, like brain or red blood cells, consume glucose continuously and independently of insulin and the interruption of this supplying may cause severe damages. For muscles, fatty tissue and liver the absorption of glucose is proportional to insulin concentration. Glucose in blood derives both from intestinal absorption of Carbohydrate (CHO) (not shown in Figure 1.1) and from internal production. In particular, the latter consists in the conversion to glucose of glycogen stored in the liver or in the so-called gluconeogenesis (the “re-construction” of glucose using substrate derived from glucose degradation). An increase in blood glucose concentration causes an increase in insulin secretion. Glucose and insulin concentration have the same effect on the glucose production and utilization: an increase in insulin (or glucose) concentration causes a decrease of glucose production and an increase of glucose utilization by muscle, while there is no influence on glucose utilization by brain.

1.1.2 Types of diabetes mellitus

The term diabetes mellitus describes a metabolic disorder of multiple aetiology characterized by chronic hyperglycemia with disturbances of CHO, fat and protein metabolism

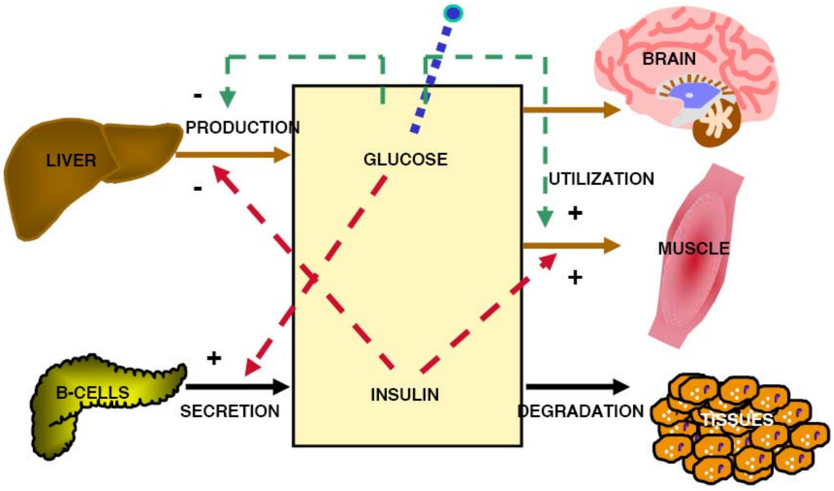


Figure 1.1: Scheme of the glucose-insulin regulatory system. Continuous arrows represent fluxes. In particular, brown ones are referred to glucose, while black ones to insulin. Dashed arrows represent the positive and negative control, indicated with "+" and "-" respectively. The green dotted arrows highlight the self-control employed by a substance, while red dotted arrows indicate the control of a substance over the other one.

resulting from defects in insulin secretion, insulin action, or both. Diabetes mellitus is diagnosed, according to the WHO, by the classic symptoms of polyuria, polydipsia and unexplained weight loss, and/or a hyperglycemia ($\geq$200 mg/dL) in a random sample, or fasting (no caloric intake for 8 h) plasma glucose higher than 126 mg/dL, and/or postprandial value higher than 200 mg/dL. (2 h plasma glucose level during an oral glucose tolerance test) [7]. Two major types of diabetes, requiring distinct therapy, can be distinguished.

1.1.2.1 Type 1 Diabetes (T1D)

Type 1 Diabetes (T1D), or Insulin Dependent Diabetes Mellitus (IDDM), is characterized by loss of insulin production by the pancreatic beta cells, leading to total insulin deficiency. Only approximately 5% of people with diabetes have this form of the disease [8]. In most cases, T1D has an autoimmune origin and various factors may contribute to its onset, including genetics and exposure to certain viruses. T1D typically appears during childhood or adolescence, thus it is also called "juvenile diabetes", however, it also can develop in adults. Despite active research, T1D has no cure, although it can be managed.

The therapy of T1D consists in exogenous injections of insulin to compensate for missing secretion from the pancreas. Before each meal, the patient decides the insulin bolus to be injected to allow the tissues to uptake the glucose that will reach the bloodstream. Such bolus is defined according to tables designed by the physician and tuned on the patient's history. Moreover, either slow-acting insulin or a continuous infusion of insulin are administered to mimic the so called insulin basal rate, which allows the body to continuously absorb the glucose which is produced mostly by the liver.

1.1.2.2 Type 2 Diabetes (T2D)

Type 2 Diabetes (T2D), or Non-Insulin Dependent Diabetes Mellitus (NIDDM), is a chronic condition that affects the way the body metabolizes glucose. In T2D, the organism either resists the effects of insulin or does not produce enough insulin to maintain a normal glucose level. It is frequently associated with obesity and a sedentary lifestyle. T2D is the most common diabetes type, accounting for about 90% to 95% of all diagnosed cases [9] and mostly affects adult people, however, it increasingly affects children as childhood obesity increases [10].

There's no cure for T2D, but it can be managed by tuning appropriately Physical Activity (PA) and diet. In some T2D subjects, after years of overproduction of insulin, the pancreas may cease to secrete insulin and exogenous insulin infusions become necessary [11].

1.1.3 Diabetes-Related Complications

In diabetes, the concentration of glucose in blood, referred in the following as Blood Glucose (BG), often exceeds the euglycemic range. Hypoglycemia and hyperglycemia might lead to short and long term complications. Hypoglycemia affects mostly the brain, given its continuous glucose demand and it can progress from measurable cognition impairment to aberrant behaviour, seizure and coma [12]. Several factors can cause hypoglycemia in people with diabetes, including taking too much insulin or other diabetes medications, skipping a meal, or exercising harder than usual. Hyperglycemia, if left untreated, can become severe and lead to serious complications requiring emergency care, such as diabetic coma. In the long term, persistent hyperglycemia, even if not severe, can lead to several invalidating complications, including micro-vascular complications (involving small blood vessels) and macro-vascular complications (involving large blood vessels) [13]. The former, like neuropathy, nephropathy and retinopathy can lead to nerves damage, renal failure and blindness respectively. The latter to coronary heart disease, strokes and peripheral vascular disease. Several factors can contribute to hyperglycemia in people with diabetes, including food and PA choices, illness, or not taking enough glucose-lowering medication.

In order to prevent the onset of these complications, diabetes therapy attempts to keep BG within the euglycemic range. As said in Subsection 1.1.2, this is usually done tuning diet, PA and use of appropriate medications, like insulin injections before meals and to mimic the basal insulin rate, in T1D. However, insulin dosing is a difficult task and, often, patients are not able to maintain their glucose concentration "in target" because of insulin under/overdosing. It is very important to keep the glycemic concentration in blood monitored in order to effectively tune the insulin bolus and basal rate. Patients with diabetes are thus required to monitor their blood glucose levels frequently, as explained in the following section.

1.2 Technologies for glucose monitoring in diabetes therapy

1.2.1 Self-Monitoring Blood Glucose (SMBG)

The most established and used technique to monitor glucose concentration is SMBG. Devices for SMBG have become available in the early seventies, and have now become a pocket tool that any diabetic uses daily. The most common test for measuring BG involves pricking a finger with a lancet device to obtain a small blood sample, applying a drop of blood onto a reagent test-strip, and determining the glucose concentration by inserting the strip into a measurement device. Different manufacturers use different technologies, but most systems measure an electrical characteristic proportional to the amount of glucose in the blood sample [14]. Examples of commercially available glucometers are shown in Figure 1.2. While the firs two are standalone devices (One Touch®UltraMini® [15] and Accu-Chek®Aviva-Plus [16]) and only need to be fed with a measurement strip, the third device (iBGstar™marketed by sanofi-aventis [17]) can be connected to an Apple iPhone or iPod touch to register all the information that a patient needs, and can be interfaced with pieces of software that run on the smartphone.

In the majority of cases, SMBG time-series are analyzed and interpreted by the

Figure 1.2: Commercial SMBG devices. From lef to right: One Touch®Ultra Mini® [15], Accu-Check®Aviva-Plus [16] and iBGstar™ [17].

physician during periodic visits, e.g. every two/four months and the individual therapeutic plan is revised accordingly. SMBG samples can also be used in real-time by the patient to assess somewhat the current effectiveness of glucose control. However, the sampling procedure cannot be repeated more than 5-6 times a day, indeed, the finger prick is painful for the patient, who needs to collect a drop of blood from the fingertips at each measurement. Thus, due to their sparseness, SMBGs cannot give complete information of glycemic excursions and dynamics and it may happen that glucose subtly exceeds the safe euglycemic range without the patient's awareness [18]. To overcome the limitations of SMBG monitoring, in the last 15 years devices able to measure glucose concentration almost continuously, the so called CGM sensors, have been developed and commercialized and are becoming even more popular and widely adopted by diabetic patients.

1.2.2 CGM

To overcome SMBG limits, during the last 15 years, CGM sensors have been developed. CGM devices measure the glycemic level in the interstitial fluid in place of the blood compartment, as done instead by SMBG, thus their invasiveness is minimal and glucose concentration can be measured in real time with a 1-5 min sampling period and for up to 7-10 consecutive days (with the perspective of increasing the duration of their life up to 14 days in the next few years).

CGM sensors can essentially be divided into two categories: implantable needle-type enzyme sensors, and systems based on the use of a micro-dialysis probe coupled with a glucose biosensor. In the rest of this section we will give a brief overview of some popular commercial sensors. Reviews of how these sensors work at the biochemistry level, with a critical discussion of pros, cons and perspectives and a comprehensive bibliography are reported in [19–22].

1.2.2.1 Subcutaneous needle-based enzyme sensors

Needle-type enzyme sensors exploit glucose-oxidation reaction and measure the current flowing from the working to the counter electrode. The glucose-oxidase measurement principle is based on the generation of hydrogen peroxide via the enzyme glucose oxidase. After this step, a mediator conveys the electrons to the working electrode, where a potential is applied to oxidize the mediator itself. Since the sensor is implanted, enzyme and mediator should be immobilized onto the electrode surface to avoid them to dissolve in the interstitial fluid. A popular mediator is oxygen, because it is available in the interstitial fluid without requiring immobilization [20]. However, since oxygen concentration in interstitial fluid can be several hundred times smaller than the glucose concentration, techniques to limit glucose concentration should be adopted [23, 24]. The reaction, considering oxygen as a mediator, is the following:

$$\begin{aligned} \text{glucose} + O_2 &\xrightarrow{glucose\,oxidase} H_2O_2 + \text{gluconic acid} \\ H_2O_2 &\xrightarrow{\sim 700mV} O_2 + 2H^+ + 2e^- \end{aligned} \tag{1.1}$$

Some examples of commercially available subcutaneous sensors include the FreeStyle Navigator®(Abbott Laboratories, Alameda, CA, USA), the Dexcom SEVEN®PLUS and Dexcom G4®PLATINUM (Dexcom Inc., San Diego, CA, USA), the MiniMed Guardian Real-Time (Medtronic MiniMed, Northridge, CA, USA), to mention a few.

The FreeStyle Navigator®CGM System consists of four components (see Figure 1.3): a miniature electrochemical sensor placed in the subcutaneous adipose tissue, a disposable sensor delivery unit, a radiofrequency transmitter connected to the sensor, and a hand-held receiver to display continuous glucose values [25]. The sensor can be used for 5 days, the glucose data on the receiver are updated once a minute[1] and include a trend arrow to indicate the direction and rate of change averaged over the preceding 15 min. The user interface of the receiver allows the threshold alarms to be set at different glucose levels. The receiver contains a built-in Free-Style blood glucose meter for calibration of the sensor as well as for confirmatory blood glucose measurements. It was approved by Food and Drug Administration (FDA) in 2008 [26, 27].

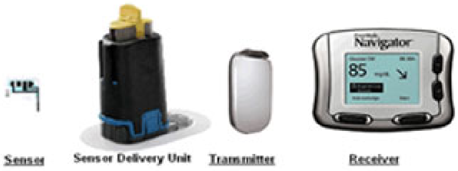


Figure 1.3: FreeStyle Navigator®CGM System [27]. From left to right: miniature electrochemical sensor placed in the subcutaneous adipose tissue, sensor delivery unit, radiofrequency transmitter connected to the sensor and hand-held receiver to display continuous glucose values.

The Dexcom SEVEN®PLUS sensor consists of three parts (see Figure 1.4(a)): a small sensor placed in the subcutaneous adipose tissue, a wireless transmitter, which has approximately the same size of a quarter coin, and a receiver [28]. It performs a new

[1] The FreeStyle Navigator sensor used during the DIAdvisor$^{\text{TM}}$DAQ trial (see Appendix B) returned raw current data with a sampling time of 1 min and glucose concentration data every 10 min. SMBG used for calibration were also rendered available, thus, once the data had been downloaded, the raw current data could be calibrated to obtain glycemic data every minute for testing prediction algorithms. Nevertheless, some of the literature models discussed in Section 1.5 use FreeStyle Navigator glucose data with a sampling period of 10 min.

measure every 5 minutes for 7 days. The receiver displays the sensor glucose value along with a graph showing glucose trend of the last 1, 3 or 9 h. The receiver contains memory up to 30 days of continuous glucose information and has programmable high and low glucose alerts and a non-changeable low glucose alarm set at 55 mg/dL. It was approved by FDA in 2009 [29]. An improvement of this sensor is the recently commercialized Dexcom G4®PLATINUM, approved by FDA in 2012 [30], whose performance are notably better than those of the SEVEN®PLUS, as reported in [30, 31].

(a) Dexcom SEVEN®PLUS sensor. From left to right: the sensor, the transmitter and the receiver [28].

(b) Dexcom G4®PLATINUM sensor. From left to right: the sensor, the receiver and the transmitter [30].

Figure 1.4: Dexcom SEVEN®PLUS and Dexcom G4®PLATINUM sensors.

The Guardian Real-Time device consists of the Guardian®REAL-Time CGM System monitor (Figure 1.5, left), the MiniLink REAL-Time Transmitter and the glucose sensor inserted in the subcutis (Figure 1.5, right). This sensor performs a new measure every 5 minutes for 3 days [32]. The receiver contains memory up to 21 days of continuous glucose information and has alerts if a glucose level falls below or rises above pre-set values. It was approved by FDA in 2005. This sensor is usually integrated with an insulin pump to provide the MiniMed Paradigm Real-Time Insulin Pump and Glucose Monitoring System [33].

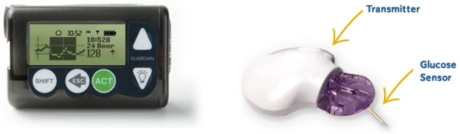


Figure 1.5: The Guardian®REAL-Time [32]. REAL-Time CGM System monitor (left), the MiniLink REAL-Time Transmitter together with the glucose sensor inserted in the subcutis (right).

1.2.2.2 Microdialysis sensors

Another type of minimally invasive subcutaneous CGM sensor is based on a microdialysis system, which exploits a hollow fiber, permeable to glucose and other small molecules and impermeable to larger molecular species, inserted subcutaneously. A fluid isotonic to the interstitial fluid, but containing no glucose, is pumped through the membrane fibers, so that the glucose in the interstitial fluid, driven by osmotic forces, diffuses through the membrane into the fluid stream, and the glucose concentration in the pumped fluid reaches an equilibrium with the glucose concentration in the interstitial fluid. The fluid flowing through the microdialysis membrane is then pumped to a glucose detector, which usually measure glucose with the amperometric approach, exploiting glucose oxidase and oxygen. The major advantages of microdialysis are the possibility of exposing the detector to atmospheric oxygen, (avoiding the deficit that characterizes glucose oxidase electrochemical sensors using O_2/H_2O_2 as mediator), and the fact that the measurement is not affected by biofouling mechanisms, since the sensor is outside the body. However, new issues are represented by the necessity of a biocompatible microdialysis membrane, and by the time lag due to the pipe between the microdialysis membrane and the glucose sensor.

The GlucoDay®by Menarini Diagnostics (Florence, Italy) is a microdialysis-based glucose monitoring system [34, 35], based on enzymatic-amperometric measurement analyzing the fluid coming from the subcutis of the abdominal region. The system, shown in Figure 1.6, comprises an apparatus with dimension comparable to a walkman, a sensor fibre no thicker than a hair as well as two plastic bags (one for the buffer solution, one for the waste products) as disposables. The apparatus contains also a measurement cell and a peristaltic pump. The buffer solution is pumped from a bag into the subcutaneous tissue through the microfibre and rinses the interstitial fluid, from which the measurements are obtained every 3 min and stored in memory. Data are downloaded after monitoring (maximum monitoring time, 48 h). It incorporates safety alarms for hypo- or hyperglycemia events. Recently, the same company launched the GlucoMen®Day (currently waiting the Conformité Européenne (CE) mark), which overcomes various shortcomings of its predecessor [36]. It is smaller and more compact, and has a longer lifetime (100 h), is more stable and embeds different algorithms for signal processing and data management [37,38].

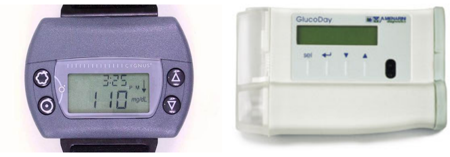


Figure 1.6: GlucoDay®S device: receiver (left) and transmitter (right) attached to the inserted sensor [35].

Another sensor based on a microdialyses principle is the SCGM 1 sensor (Roche Diagnostics, Mannheim, Germany) [39,40].

1.2.2.3 Other techniques for CGM

As alternative to subcutaneous sensors based on the glucose-oxydase enzyme and to sensors based on microdialysis, other systems and prototypes have been proposed for CGM monitoring. We cite some of them for sake of completeness.

- **Iontophoresis and Sonophoresis.** These techniques require the stimulation of the skin from outside, in order to extract glucose from the skin for its direct measure. The *iontophoresis* is based on the extraction of glucose associated with the application of an electrical potential, causing the migration of ions from beneath the skin. In particular, sodium and chloride are pulled towards the cathode and anode respectively. The ion flow also causes neutral molecules like glucose to migrate across the skin along with the water hydrating the ions. Glucose is then detected with the enzymatic reaction reported in eq 1.1. The GlucoWatch G2 Biographer (Cygnus Inc., Redwood City, CA, not on the market any more because it caused skin irritation in users), is an example of device which used the reverse iontophoresis. *Sonophoresis* uses low-frequency ultrasounds to create an array of microscopic holes on human skin, which increase its permeability, allowing glucose to trespass the skin to be directly measured. The SonoPrep (Echo Therapeutics Inc., Philadelphia, PA [41]) is a device which exploits this technology.

- **Micropores and Microneedles Techniques.** Micropores techniques perforate the stratum corneum without perforating the full thickness of the skin with the aid of pulsed laser or local heat. Interstitial fluid is then collected applying vacuum and a direct measure of glucose is obtained.

- **Noninvasive CGM.** Non-invasive CGM sensors measure glucose concentration through the skin without extracting blood or interstitial fluid or without a needle penetrating the skin for reaching these fluids. Hence, these sensors are more comfortable for the patient than the previously described sensors and do not cause adverse physiological reactions. These sensors measure different physical properties of the skin and underlying tissues which are modulated by glucose concentration changes. Among the physical principles exploited for this scope, we can list optical techniques, e.g. based on absorption phenomena (Near InfraRed Spectroscopy, Mid InfraRed Spectroscopy), on scattering (Raman Spectroscopy, Occlusion Spectroscopy), on Optical Coherence Tomography, on Fluorescence Technologies; Photoacoustic Spectroscopy; Impedance Spectroscopy; Electromagnetic Sensing; Thermal Emission Spectroscopy. A general idea is to combine several of these techniques to obtain signals which are correlated to the concentration of glucose in blood (multisensor concept). Although non-invasive CGM are attractive from a user's point of view, they do not offer the same accuracy of subcutaneous sensors yet. In particular they are difficult to calibrate, and they are not yet usable to extract reliable information on glucose dynamics [42].

In this thesis, only subcutaneous minimally invasive CGM will be considered, hence the acronym CGM will be always referred to these kind of devices.

1.2.3 Offline and online use of CGM time series

Diabetic patients who monitor themselves via SMBGs and CGM can gather a lot of information regarding their pathology. In particular, patients can exploit such information in several ways, e.g. to tune their insulin therapy. Moreover, the advent of CGM devices offers a richer insight of glucose dynamics and their relation with exogenous events.

CGM time series can be analyzed retrospectively to evaluate glucose variability, e.g. [43] and to suggest the refinement of the patient individual therapy. In clinical practice, it is today largely accepted that CGM sensors can significantly improve diabetes control and reduce HbA1c [44–46]. Furthermore, CGM has been recommended, possibly integrated in the so called sensor-augmented pump, for the treatment of subjects prone to hypoglycemia, e.g. [47, 48].

Apart from offline analysis of quasi continuous glucose recordings, CGM sensors allow interesting online applications, as the generation of hypo and hyperglycemic alerts as soon as interstitial glucose exceeds those critical thresholds, with the possibility for the patient of timely treating/mitigating the event (by a sugar ingestion to compensate a hypo or an insulin administration to tackle a hyper), see [49] for a review. Moreover, a research community of applied mathematicians and biomedical engineers is active to improve CGM sensor outcomes and strengthen the impact of applications by developing real-time algorithms for denoising, signal enhancement, prediction of glucose time course and alert generation, see e.g. [50, 51] for reviews and [52] for the recently proposed "algorithmically smart sensor" concept. In addition, the CGM sensor is crucial in the development of the Artificial Pancreas (AP), a minimally-invasive pump which subcutaneously administers insulin according to a temporal profile determined in real-time by a sophisticated closed-loop control algorithm that has CGM measurements as one of its key inputs, see e.g [53,54] for recent perspectives.

Of particular interest in the present thesis is the possibility of predicting glucose concentration in the short term (approximately 30-60 min in advance), allowing the patient to take therapeutic decisions on the basis of future instead of current glycemia, possibly mitigating/ avoiding imminent critical events.

1.3 Short-term prediction of glucose concentration from CGM sensor data

Glucose dynamics are influenced by many factors, e.g. CHO intake, administration of drugs including insulin, PA, stress, emotions. Furthermore, inter- and intra-individual variability is high. For these reasons, prediction of future glucose levels poses several challenges. To better highlight glucose prediction issues, in Figure 1.7 we show the time-course of glucose concentration (black dots linearly interpolated to facilitate the visualization of the time series) measured during the day of a T1D subject by the Dexcom SEVEN PLUS CGM sensor, together with information on insulin injections (green stems) and on CHO ingested during meals (blue stems). Hypo- and hyperglycemic thresholds are also reported (thin horizontal lines). As we can note, during the first night glucose concentration was in the euglycemic range, but fell below 70 mg/dL around time 07:30. The subject had breakfast around 8:00 but did not inject any insulin bolus in concomitance to the meal. During the morning, around 09:00, glucose concentration reached hyperglycemic values and the subject injected a correction bolus of insulin around time 10:00 and re-entered the euglycemic range around time 12:00. At time 13:00 and 19:00 the subject ate and injected insulin to counterbalance the effects of CHO. Notably, around time 17:00 the CGM signal fell in the hypoglycemic range and the subject promptly ingested 10 g of sugar to increase his glycemia and re-enter the safe range. After dinner, around time 20:00, glycemia crossed the hyperglycemic threshold

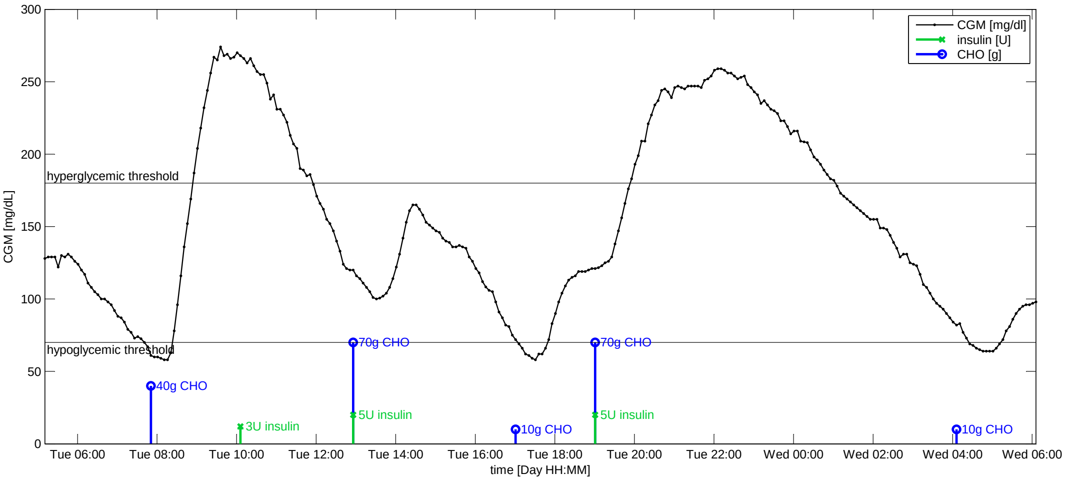


Figure 1.7: Representative CGM signal (black dots linearly interpolated to facilitate the visualization of the time series) measured by the SEVEN PLUS device and information on insulin doses (green stems) and CHO content of meals (blue stems) of a T1D.

and re-entered the safe range only around time 01:00. The subject also experienced a hypoglycemic event during the second night, at time 04:00 and he medicated it by timely ingesting 10 g of sugar. This example confirms that, in principle, forecasting glucose concentration should use several inputs: certainly glucose concentration measured by the CGM sensor, but also ingested CHO and injected insulin play a major role. However, accounting for all these inputs, formalizing them in mathematical terms and extracting useful signals from them is not easy. For these reasons, as better discussed in Section 1.4, the majority of published glucose prediction methods solely use the CGM signal as input.

While we refer the reader to [50, 51] for comprehensive reviews on algorithms for prediction of glucose concentration, in the rest of this chapter we will shortly describe some class of widely used prediction models, paying particular attention to Neural Network (NN)-based algorithms. Section 1.4 reviews approaches based only on past CGM data. Section 1.5 presents algorithms proposed in the last five years, able to exploit not only CGM, but also information on insulin therapy, ingestion of CHO and PA, which are known from physiology to influence glucose concentration dynamics. Section 1.6 summarizes contributions demonstrating the clinical utility of prediction for reducing hypoglycemia. Section 1.7 states the aim of the present thesis and, finally, Section 1.8 gives an outline of the thesis.

1.4 Prediction methods based only on CGM information

1.4.1 AR and ARMA models

Two popular time-series modelling approaches adopted for short-time prediction are based on Auto-Regressive (AR) and Auto-Regressive with Moving Average (ARMA) models. These techniques assume that future glucose concentration can be expressed as a linear function of previous glucose measurements and do not use neither prior information nor meal or insulin information.

In [55] a time invariant AR model of order 10 was proposed. The model was identified on data of 9 T1D subjects, monitored for approximately 5 consecutive days with the iSense CGM device [56], with a sampling time of 1 min. Parameters were optimized using regularized Least Squares (LS) and the models were assessed in terms of Root Mean Square Error (RMSE) and Error Grid Analysis (EGA) [57], considering Prediction Horizon (PH) of 30, 60 and 120 min. Both subject specific and subject invariant models were evaluated obtaining comparable results. In [58] Gani and colleagues proposed an AR(30) time invariant subject specific model. The models were optimized and assessed on data of 9 T1D subjects, monitored for approximately 5 consecutive days with the iSense CGM device (sampling time of 1 min). The first 2000 min of every time series were used for optimizing the AR model parameters and the remaining 2000 min were used as test data. Three cases were considered: scenario 1, in which raw glucose data were used; scenario 2, in which glucose data where smoothed before computing AR coefficients and scenario 3, in which smoothing and regularization were used. Parameters were determined via LS and the models were assessed on PH of 30, 60 and 90 min in terms of RMSE and time anticipation. Only scenario 3 guaranteed accurate predictions and a clinically acceptable time lag for PHs of 30 and 60 min.

In several contributions, to cope with the non-stationarity due to intra-subject variability characterizing glucose dynamics, the authors adopt time variant AR and ARMA models, identified recursively every time a new glucose measurement becomes available, using a forgetting factor to assign a relative weight to past data and a finite memory to the system. In [59] Sparacino and colleagues proposed a first order AR model with time-varying parameter. The model was identified on CGM data of 28 T1D volunteers monitored for 48 consecutive hours by the GlucoDay CGM system (sampling time of 3 min), in normal daily life conditions. Parameters were estimated at each time step using Recursive Least Squares (RLS). Various values of the *forgetting factor* were tested with PH of 30 and 45 min. Prediction was assessed computing Mean Square Error (MSE), Energy of Second Order Derivative (ESOD) and time anticipation. Results were accurate and time anticipation was sufficient to potentially avoid or mitigate several critical hypo- and hyperglycemic events. In [60] an ARMA(2,1) model with time-varying parameters was investigated. The model parameters were estimated with RLS at each time step, using a change detection method to enable dynamic adaptation of the model to intra-subject variability and dynamic disturbances. The models were identified and tested on denoised data monitored with the Gold™CGMS®system (Medtronic MiniMed) for 48 consecutive hours, with a sampling time of 5 min. Two distinct databases were used: one formed by 22 healthy hospitalized individuals, the other one constituted by 14 T2D subjects in free daily life conditions. Models were evaluated in terms of Sum of Squared Errors (SSE), Relative Absolute Difference (RAD) and EGA for PHs up to 30 min. Results proved that recursive identification of the model parameters allowed improving accuracy, with respect to time-invariant models.

1.4.2 Polynomial models

In [59] a time-varying first order polynomial (i.e. linear) model was identified with weighted RLS on CGM data of 28 T1D subjects, monitored for 48 consecutive hours by the GlucoDay CGM system (sampling time of 3 min). The quality of prediction was quantified in terms of MSE, ESOD and time anticipation, considering PHs of 30 and 45 min. Results were comparable to those obtained with the AR(1) [59] predictor on the same data.

1.4.3 Kalman filter

A Kalman filtering methodology was proposed in [61, 62], which only uses information on past CGM readings by assuming a double integrated random walk as prior for glucose dynamics. In [61] the authors used simulated data to demonstrate the effects of measurement sampling frequency, prediction threshold level for hypoglycemia detection and PH on sensitivity and specificity of hypoglycemia prediction. In [62] the approach was used on 13 time series relative to hypoglycemic clamps, in which glucose concentration was measured with the Medtronic CGMS sensor (sampling period of 5 min). Over all the dataset, the sensitivity and specificity of hypoglycemia (defined as glucose lower than 70 mg/dL) prediction were calculated, for different PHs ranging from 5 to 30 min and different prediction thresholds, from 60 to 90 mg/dL, for hypoglycemia detection.

1.4.4 Kernel-based regularization strategies

In [63] a kernel-based regularization learning algorithm was proposed. The authors adopted a meta-learning approach, in which the kernel and the regularization parameter are adaptively chosen on the basis of previous similar learning tasks, using past glucose information. The algorithm was trained on data of one diabetic patient, monitored for 24 h with an Abbott FreeStyle Navigator CGM sensor (sampling time 10 min). The predictor was then tested, without any re-adjustment, on other 10 diabetic patients, monitored with the Abbott CGM sensor and on 6 diabetic subjects, monitored with the SEVEN PLUS CGM system (sampling time 5 min). Results were computed in terms of EGA and Prediction EGA (PEGA) [64] for PHs of 30 and 60 min.

1.4.5 Hybrid strategies

In [65] the authors proposed a combination of multiple models for hypoglycemia prediction. Going into details, their system consisted in a linear projection based on the trend on the previous 15 min, a Kalman filtering in line with that presented in [62], a hybrid infinite impulse response filter, statistical models and numerical logical algorithms. A voting system processed the output of the five algorithms and determined if a hypoglycemic alert was generated. The method was developed using data of 21 T1D children, monitored with the FreeStyle Navigator CGM sensor (sampling time 1 min) and tested using a separate dataset of 18 subjects, monitored with the same sensor. Low glucose concentration was induced by gradual increases in basal insulin infusion rate up to 180% from the subject's own baseline infusion rate. The algorithm was assessed, retrospectively, on the basis of the number of hypo events correctly forecasted, evaluating and comparing performance obtained with different voting thresholds, PHs ranging from 35 to 55 min and alarm thresholds equal to 70, 80 and 90 mg/dL.

1.4.6 Neural Networks (NN)

In the last few years, the possibility of using NNs for short-time glucose prediction has been investigated. In particular, in [66] Pérez-Gandía et al. proposed a NN whose inputs were CGM samples in the previous 20 min and the current time instant, and whose output was glucose concentration after PH ranging from 15 up to 45 min. The proposed NN is feedforward with 2 hidden layers with 10 and 5 neurons, respectively. The model was trained with Levenberg-Marqardt backpropagation and tested on two distinct datasets: one constituted by 9 subjects monitored using the Guardian CGM sensor (sampling period of 5 min) and the other one formed by 6 subjects monitored with the Navigator CGM system (sampling time of 1 min). Results, quantified in terms of RMSE and time anticipation, were comparable to those obtained with the AR(1) and the linear model of [59] on the same dataset.

1.5 Prediction methods based on CGM and other available information

1.5.1 ARX and ARMAX models

A natural approach to exploit CGM and other available information is to extend AR and ARMA models by adding a term related to exogenous signals among their inputs.

A first attempt of exploiting information on CHO and insulin therapy by Auto-Regressive with eXogenous Inputs (ARX) models was made in [67] by Finan and colleagues. Both a time-invariant and a time variant approach were assessed in terms of FIT and RMSE for PHs ranging from 30 to 90 min. The models were tested on two datasets, collected on the same patients under different conditions. The dataset consisted in several time series measured in normal ambulatory conditions in 9 T1D adults monitored with the CGMS (sampling period of 5 min). Insulin pump records of basal rates, bolus amounts and time, and subject recorded estimates of time and CHO quantity of meals were also collected. Each dataset spanned 2 to 8 days. Third order batch ARX models were identified from the first half of the dataset and were used to predict the second half of the dataset. In a second portion of the study, 6 of the 9 subjects were administered prednisone, (an insulin sensitivity lowering drug), for 3 consecutive days. For these datasets batch ARX models identified from normal data were used to predict prednisone data. In addition, time variant ARX models were identified recursively to predict prednisone data. PHs of 30, 45, 60 and 90 min were investigated. The batch ARX method produced prediction as accurate or slightly more accurate than the recursive ARX method.

In [68] a time-varying ARX model using meal and insulin information preprocessed to generate, respectively, CHO rate of appearance in the blood and plasma insulin was proposed. Model parameters were estimated recursively using the normalized least mean square algorithm and exploiting a physiological gain adaptation rule. The model was used for prediction of future glucose concentration up to 50 min ahead in time and tested on data of 15 hospitalized T1D subjects, monitored for 76 h with the FreeStyle Navigator CGM device (sampling time 10 min). Results were quantified in terms of FIT and Continuous Glucose - Error Grid Analysis (CG-EGA) [69] and defined satisfactory.

In [70] a time-varying multivariate subject specific Auto-Regressive with Moving Average and eXogenous Inputs (ARMAX) model, with exogenous inputs including food intake, PA, emotional stimuli and lifestyle was investigated. Parameters were identified online using the weighted RLS method coupled with a change detection strategy for a faster adaptation in case of drastic glycemic disturbance. Data used in this study were relative to 5 T2D subjects under free living conditions, monitored for about 24 days. Glucose concentration was monitored with the MMT-7012 Medtronic CGM sensor (sampling time 5 min) and physiological signals were measured with the SenseWear Pro3 (BodyMedia Inc., Pittsburgh, PA) armband body monitoring system. Prediction performance was numerically evaluated computing the RAD and the SSE of Glucose Prediction, investigating a PH of 30 min. Results showed that prediction accuracy was improved and error metrics were reduced using the multivariate model, with respect to an univariate model based solely on CGM data. The authors also preliminarily evaluated the ability of the multivariate ARMAX algorithm of predicting hypoglycemia, reporting acceptable sensitivity and false alarm rate.

Recently, Turskoy and colleagues [71] proposed a subject specific recursive ARMAX model for prediction of glucose concentration exploiting insulin on board and PA information. In their implementation, the ARMAX model was converted to its state-space form, to develop a simpler stability criterion and simplify the set of equations. Moreover, a constraint to guarantee that insulin had a negative effect on the predicted signal was introduced. The model was tested on data of 14 T1D subjects, smoothed both, non-casually using a Savitzky-Golais filter and casually, with a Kalman filter. Glucose concentration was measured with the iPro CGM device (sampling period 5 min), while metabolic, PA and emotional state were monitored with the SenseWear Pro3 armband system. PHs ranging from 5 to 60 min were considered. Assessment criteria included the accuracy of the forecasted profile, measured with RMSE and SSE and the model ability of predicting hypoglycemia, quantified in terms of sensitivity, false alarm ratio and average detection time. Results suggested that when PA information was added to the ARMAX model the prediction error decreased significantly. Moreover, the model was able to predict accurately almost all the hypoglycemic events with an average anticipation of 28 min.

1.5.2 Machine learning strategies

In [72] Daskalaki and colleagues compared, on a simulated dataset, the performance of an AR, an ARX and a NN model exploiting, respectively, only CGM history, CGM and insulin and CGM, insulin and meal information. The AR and ARX models have time-varying parameters updated at each time step using RLS algorithms. The NN model has feedback connections for better learning glucose nonlinear time-varying dynamics. The three models were optimized and evaluated on a virtual population of 30 T1D subjects, extracted from [73], simulated for 8 consecutive days with a sampling period of 5 min. PHs of 30 and 45 min were considered and goodnes of prediction was quantified computing RMSE, RAD, time lag and correlation coefficient. The NN resulted to be the most appropriate algorithm for prediction of glucose profile based on glucose, insulin and meal data and outperformed the other models for all patients and for both PHs. However, no comparison with an ARX model using both insulin and meal information was reported.

In [74] Zaho et al. used CGM, meal and insulin information as inputs of a latent variable based predictor, optimized via partial least square and canonical correlation analysis. Impulsive information on insulin therapy and meal were preprocessed with a second order transfer function model to obtain time-smoothed inputs. The proposed approach was compared with time-invariant AR and ARX models and with a latent variable algorithm based solely on CGM data. The algorithm was assessed on 10 virtual datasets [73] simulated for 7 days with a sampling time of 5 min. The first 2 days were used for optimizing and validating the algorithm, while the following 5 days were used for testing it. Furthermore, the strategy was also applied to clinical data of 7 ambulatory T1D subjects, monitored with the SEVEN PLUS CGM sensor (5 min sampling time). In this case, the first day of data was used for model identification and the rest of the time series was used for testing the algorithm. Results were quantified in terms of RMSE and CG-EGA. On simulated data, the latent variable algorithm using CGM, meal and insulin outperformed the other reference methods. However, on real data performance was comparable.

In [75, 76] Georga and colleagues proposed, respectively, a random forest and a support vector based algorithm to predict glucose concentration. In both contributions the authors analyzed PHs of 15, 30, 60 and 120 min. The inputs of both predictors include CGM, the rate of appearance of meal and the cumulative amount of glucose appeared in the blood, plasma insulin concentration generated with a model of the absorption of exogenous insulin, the hour of the day and the cumulative amount of energy expended during PA. Both algorithms were optimized and tested on data of 27 T1D patients, monitored for 5 to 22 days with the Guardian Real-Time CGM system (sampling time 5 min). Information on PA was registered with the SenseWear armband system, while information on food intake and insulin therapy was recorded manually by the patients. Results, computed in terms of RMSE and correlation coefficient, suggested that the best accuracy was obtained when all the exogenous inputs were used. No comparison with other predictors was reported.

In [77] Pappada and colleagues developed a feedforward NN incorporating, in addition to CGM data, other inputs such as SMBG readings, impulsive information on insulin and meal, information on hypo- and hyperglycemic symptoms, lifestyle, activity and emotions. Their NN model predicts a complete vector of future glucose values across the model PH of 75 min. The database used for optimizing and testing the NN was constituted by 27 insulin dependent diabetic patients; glucose concentration was measured by the CGMS Gold device (sampling time of 5 min) and documentation of other information was done by the patients using an electronic diary. The predictor was trained on data of 17 patients and assessed on data of the remaining 10 subjects in terms of RMSE, percentage Mean Absolute Error (MAE) and EGA. The model accuracy was satisfactory but hypoglycemia resulted routinely overestimated.

Recently, Wang and colleagues [78] combined several prediction algorithms (AR, extreme learning machine and support vector regression) using adaptive weights inversely proportional to each model's prediction error. The models were optimized and tested on data of 10 T1D subjects monitored either with the SEVEN PLUS, either with a Medtronic CGM device (both with a sampling time of 5 min). The first half of each time series was used for optimizing the models and the second half for testing the algorithm. Prediction quality was assessed computing the RMSE, the relative error, the EGA and the J index [79]. Results showed that the model ensemble performed better than the singular algorithms and was more robust with respect to variations on data characteristics.

1.6 Quantification of the clinical usefulness of glucose prediction for hypoglycemia reduction

One of the most appealing application of glucose prediction is the generation of hypoglycemic alerts based on the predicted glucose value, potentially allowing the patient to take adequate therapeutic decisions in advance, possibly avoiding or at least mitigating the risky event. So far, some proof of concept applications on the possibility of limiting induced hypoglycemia have been described in the literature.

In Buckingham et al. [80,81], nocturnal hypoglycemia was induced in 15 hospitalized subjects increasing basal insulin infusion. Five different literature glucose prediction techniques (all based solely on past glycemic data measured by CGM) were simultaneously used to forecast future hypoglycemia, and hypo-alerts were generated on the basis of a predefined voting scheme. The triggering of a hypoglycemic alarm involved the suspension of basal insulin infusion, until recovering of blood glucose concentration to a safe value, either for a maximum of 90 min. This strategy allowed preventing the majority of the nocturnal induced hypo crisis. In [82] the same paradigm was tested outpatients, however, the authors' objective was the assessment only of the safety, not of the effectiveness, of the system.

A different strategy was employed by Hughes et al. in [83]. In this work, the authors presented a method to detect/predict the risk of hypoglycemia and to perform a gradual attenuation of insulin delivery on the basis of risk factors, instead of immediate pump shutoff. The aim was to create a safety module to be used especially in AP applications. The method was assessed on data simulated through the FDA approved T1D simulator [73] in presence of hypoglycemic conditions induced by elevated basal rate and overbolus of insulin. Results indicated that attenuating insulin delivery reduced, or at least delayed the onset of hypoglycemia, especially if the rescue CHO dose was delivered sufficiently ahead-in-time.

1.7 Aim of the thesis

As described in Section 1.4 the majority of glucose prediction strategies proposed in the literature does not extensively exploit all the relevant available information. In particular, besides CGM history, information on ingested CHO, insulin therapy and PA could improve prediction accuracy. Indeed, from physiology, meal, insulin and exercise are known to act like disturbances for glucose homeostasis, influencing its dynamics. The first aim of this thesis is thus to develop a NN based prediction model exploiting, apart from CGM, other available information. This task will be tackled in Chapters 3, 4 and 5, where two NN algorithms will be proposed and the benefits of adding inputs other than CGM will be evaluated. NN algorithms can easily exploits, as inputs, signals with different nature and characteristics and are naturally able to learn complex nonlinear functions, thus they are promising models for glucose concentration prediction, potentially more powerful than linear ARX and ARMAX algorithms. In the glucose prediction literature, only a few NN strategies have been proposed so far and they did not significantly outperformed linear time series models. We will overcome this limitation by optimizing the NN structure to better exploit its ability of learning complicated nonlinear functions. Furthermore, we will preprocess exogenous information to obtain input signals that result more informative for prediction purposes. The proposed NN models will be optimized and tested, considering several merit indexes, using simulated CGM profiles and real data of T1D subjects in free life conditions and results will be compared with those obtained by state of art algorithms, implemented on the same data.

As described in Section 1.6 a few literature contributions assessed the benefit deriving from prediction methods to prevent/mitigate hypoglycemia on real data. However, state of art analysis are mainly qualitative and a comprehensive objective assessment is missing. Indeed, a drawback of clinical studies is that, once a patient takes an action (e.g., eating sugar at time t_1), there is no way of coming back to time t_1 and evaluating what would have happened if different actions were adopted (e.g., administration of sugar at time $t_1+\tau$ or no administration at all). Thus, the second aim of this thesis is the comprehensive conceptual investigation of benefits in diabetes management that could be obtained if hypoglycemic alerts were generated on the basis of prediction. This will be addressed in Chapter 7 on a simulated database. Using the simulation environment will allow us to overcome limitations of literature approaches because, for the same subject and the same hypoglycemic event, we will be able to run different simulations (corresponding to parallel alternative and mutually exclusive scenarios) and fairly compare the effect of different actions, starting from the very same initial patient conditions. Moreover, the simulation analysis is also a powerful tool to optimize the design of expensive clinical trials for quantifying, as objectively as possible, the clinical usefulness of prediction.

Both simulated and real data will be used in this thesis. Simulated T1D glycemic profiles are obtained using the UVA/Padova diabetic simulator, a system shown to represent adequate glucose fluctuations in T1D observed during meal challenges and accepted by FDA as a substitute to animal trials in preclinical testing of closed-loop control strategies [73, 84]. Real data consists in signals collected during the FP7 EU project DIAdvisor$^{\text{TM}}$ [85], which involved our research group in the past.

1.8 Thesis outline

The thesis is organized as follows. Chapter 2 introduces the fundamental theory of NN and describes our implementation choices, common to all the proposed models. Chapter 3 describes the first predictor we developed, which is based on a NN in parallel with a time varying linear model and uses information on CGM and CHO content and timing of meals. Results obtained on simulated and real data and comparison with state of the art methods are also reported. Chapter 4 presents a new prediction model based on a jump NN that overcomes some limitations of the previous structure and obtains statistically comparable results. Chapter 5 further investigates the jump NN predictor by adding information relative to insulin therapy as input of the model and quantifying the relative contribution of each input signal in determining the predicted time series. Chapter 6 presents a quantification of short-term effects of mild PA on glucose concentration dynamics and discusses future perspectives, as the inclusion of signals related to PA as additional inputs of the NN. Chapter 7 describes a practical application of prediction for generation of hypoglycemic alerts. The reduction of hypoglycemia obtained using prediction is extensively evaluated on a simulated dataset and the design of a clinical trial, to confirm results obtained in simulation, is also briefly discussed. Finally, Chapter 8 concludes the thesis summarizing the original results obtained in our research and discussing possible future works.

2

Fundamentals of Neural Network (NN) modelling

2.1 General features of NN

A NN is a mathematical model that aims to mimic the functioning of the brain and is motivated by the recognition that the human brain computes entirely differently from conventional digital computers. Indeed the brain can be thought as a highly complex, nonlinear and parallel computer able to organize its structural units, called neurons, so as to perform certain computations (e.g. perception, pattern recognition, etc) faster than modern digital computers. Analogously, an artificial NN is a massively parallel distributed model constituted by interconnection of simple processing units, called neurons, which has a natural propensity for storing knowledge and making it available for use. It resembles the brain because:

1. knowledge is acquired by the network environment through a learning process;
2. synaptic weights, i.e. inter-neuron connection strengths, store the acquired knowledge.

NN are characterized by the following useful properties:

1. *Nonlinearity.* Artificial neurons can be both linear or nonlinear and the network resulting from their interconnection is itself nonlinear if nonlinear units exist.

Furthermore this nonlinearity is distributed through the system. Nonlinearity is an important property if the process to learn is itself nonlinear.

2. *Input-output mapping.* The most popular learning paradigm is called *supervised learning* and consists in the optimization of synaptic weights by applying a set of labelled training samples, each one consisting in a input signal and relative target. The samples are presented to the network and the weights are modified so as to minimize the distance between the target and the actual NN output. Thus the network, from examples, constructs an input-output map of the given problem. Such an approach belongs to the field of *nonparametric statistical inference*, since no prior assumptions are made on statistical models for the data.

3. *Adaptivity.* A NN trained to work in a specific environment can be easily retrained to deal with minor changes in the environmental conditions. In addition, when operating in a nonstationary context, the network could be designed to change its synaptic weights in real time. However, it should be emphasized that adaptivity might lead to non-robustness: indeed an adaptive system with short time constants may change rapidly and, therefore, respond to spurious disturbances with a degradation of performance. An adaptive system should be chosen only if the principal time constant of the process is long enough for the system to ignore spurious disturbances, but short enough to respond to meaningful environmental changes. This problem is referred to as the *stability-plasticity dilemma* [86].

2.2 NN architecture

2.2.1 Artificial neuron model

The neuron is the information processing unit fundamental to the operation of a NN. The block diagram of Figure 2.1 represents the model of a neuron, which is constituted by three basic elements:

1. A set of synapses, characterized by their own *weight*. Going into details, a signal x_j at the input of synapses j connected to neuron k is multiplied by the weight w_{kj}. It is worth making a note on how the subscripts of synaptic weight w_{kj} are written: the first subscript refers to the neuron to which the synapses leads, while the second subscript refers to the neuron from which the synapses originates. Weights can be positive as well as negative. The neuronal model also includes a *bias* $b_k = w_{k0}$ (associated to the fixed input $x_0 = 1$), whose effect is increasing (if positive) or decreasing (if negative) the net input of the activation function.

2. An *adder* that sums the weighted input signals.

3. An *activation function* for processing the input of the neuron and, usually, limiting the amplitude of the output.

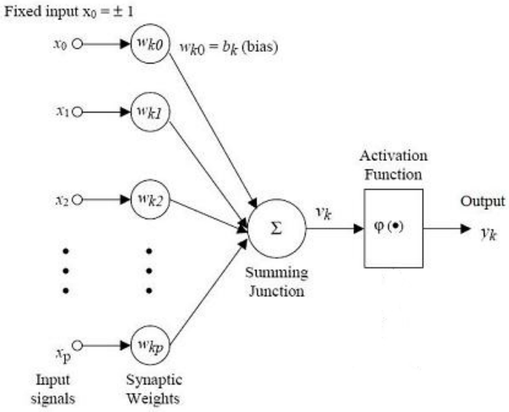


Figure 2.1: Model of an artificial neuron [87].

The neuron can be described mathematically by

$$y_k = \varphi\left(\sum_{j=0}^{p} x_j w_{kj}\right) \tag{2.1}$$

where $x_1, \ldots, x_p$ are the input signals, $w_{k1}, \ldots, w_{kp}$ are the synaptic weights of neuron k, w_{k0} is the weight connected with the input $x_0 = 1$, thus it represents the bias, $\varphi(\cdot)$ is the activation function and y_k is the output of neuron k.

2.2.1.1 Neuron activation function

The activation function $\varphi(v)$ defines the neuron output in terms of the weighted sum of its inputs. Figure 2.2 shows the three basic types of activation function:

1. Threshold function, mainly used for classification, which can be the Heaviside function (Figure 2.2(a))

$$\varphi(v) = \begin{cases} 1 & \text{if } v \geq 0 \\ 0 & \text{if } v < 0 \end{cases} \tag{2.2}$$

   or the sign function (Figure 2.2(b)), if an antisymmetric function with respect to the origin is desirable

$$\varphi(v) = \begin{cases} 1 & \text{if } v > 0 \\ 0 & \text{if } v = 0 \\ -1 & \text{if } v < 0 \end{cases} \tag{2.3}$$

2. Linear function (Figure 2.2(c))

$$\varphi(v) = v \tag{2.4}$$

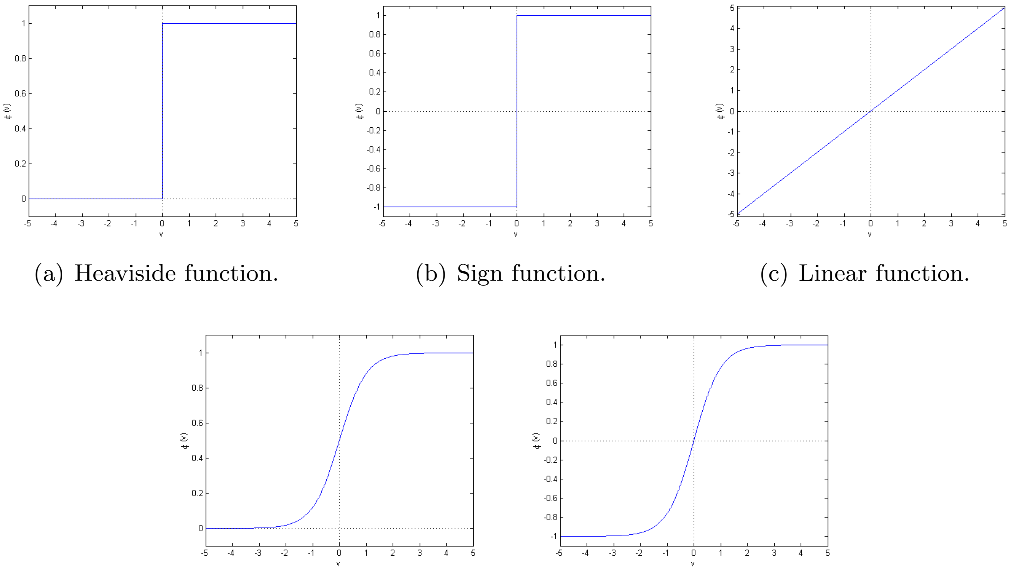
commonly used as activation function of the output neuron when the network is used for function approximation and prediction.

3. Sigmoid function, which is the most common form of activation function used for hidden layers of NN. This is usually a sigmoid logistic function (Figure 2.2(d))

$$\varphi(v) = \frac{1}{1 + e^{-2v}} \tag{2.5}$$

or a sigmoid tangent hyperbolic function (Figure 2.2(e)), if we prefer a function antisymmetric with respect to the origin

$$\varphi(v) = \frac{1 - e^{-2v}}{1 + e^{-2v}} \tag{2.6}$$



(a) Heaviside function. (b) Sign function. (c) Linear function.

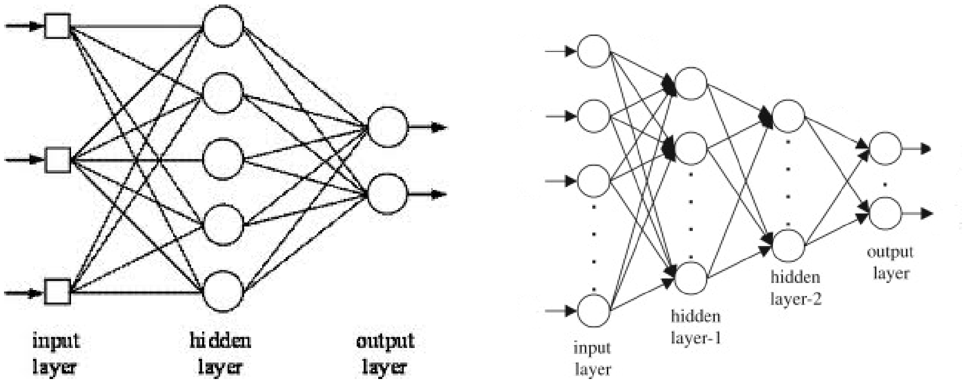
(d) Sigmoidal logistic function. (e) Tangent hyperbolic function.

Figure 2.2: Neural activation functions.

2.2.2 Multilayer FeedForward Neural Network (FFNN)

In a *multilayer FeedForward Neural Network (FFNN)* (commonly referred also as *multilayer perceptron*) the neurons are organized in layers and each layer projects only onto the following layer, but not vice versa, thus no feedback loops exist. The function of hidden neurons is to intervene between external inputs and network output, enabling the network to extract high order statistics [87].

The source nodes in the input layer supply the elements of the input vector, which constitute the inputs of the first hidden layer. The output signals of the first hidden layer are used as inputs of the second hidden layer and so on for the rest of the network. Typically, the neurons in each layer of the network have as inputs the output signals of the preceding layers only. The set of output signals of the neurons in the final layer of the network constitutes the NN response to the input activation pattern. The graph in Figure 2.3(a) illustrates a FFNN with 3 inputs, one hidden layer of 5 neurons and an output layer with 2 neurons. For brevity, such a structure is shortly referred as 3-5-2 and, generally, a FFNN with m input nodes, h_1 neurons in the first hidden layer, h_2 neurons in the secon hidden layer and q neurons in the output layer is referred as $m - h_1 - h_2 - q$ network. The architecture in Figure 2.3(b) represents a FFNN with two hidden layers and an unspecified number of neurons in each layer. The NNs of Figure 2.3 are fully connected since each node in every layer is connected to every node of the adjacent forward layer. If some links were missing, the network would have been partially connected.



(a) Fully connected FFNN with one hidden layer.

(b) Fully connected FFNN with two hidden layers.

Figure 2.3: Fully connected FFNNs.

A multilayer perceptron has three characteristics:

1. Each hidden neuron is characterized by a nonlinear differentiable activation function,

commonly the logistic function or the tangent hyperbolic function. The presence of nonlinearities is essential for the NN ability of learning nonlinear relationships between inputs and target. The output neurons are usually characterized by a linear function.

2. The NN contains one or more hidden layers, which enable it to learn complex task by extracting, progressively, features from the input patterns.

3. The NN has a high degree of connectivity.

The computing power of multilayer FFNN derives from these characteristics and from the ability of learning from experience through training.

Let us define: y_k the k^{th} output of the NN; m_l the number of neurons of the l^{th} layer, with $l = 0, \dots, L$, thus m_0 is the number of inputs, m_1 the size of the first hidden layer and m_L the number of outputs; w_{ji}^{n-m} the weight connecting the i^{th} neuron of the m^{th} layer to the j^{th} neuron of the n^{th} layer, with $n, m = 0, \dots, O$, thus $n = 0$ is the input layer, $n = 1$ the first hidden layer and $n = O$ the output layer, and the same for m; finally, w_{k0}^{l} the bias of the k^{th} neuron of the l^{th} layer and, with an abuse of notation, for simplifying equations, w_{k0}^{l-m} also represents the bias of the k^{th} neuron of the l^{th} layer. The mathematical representation of the FFNN of Figure 2.3(a) is

$$y_k = w_{k0}^2 + \sum_{j=1}^{m_1} w_{kj}^{2-1} \varphi \left(\sum_{i=0}^{m_0} w_{ji}^{1-0} x_i \right) \tag{2.7}$$

and the mathematical representation of the FFNN of Figure 2.3(b) is

$$y_l = w_{l0}^3 + \sum_{k=1}^{m_2} w_{lk}^{3-2} \varphi \left(w_{k0}^2 + \sum_{j=1}^{m_1} w_{kj}^{2-1} \varphi \left(\sum_{i=0}^{m_0} w_{ji}^{0-1} x_i \right) \right) \tag{2.8}$$

Comparing equations (2.7) and (2.8) it is clear that adding one hidden layer increases the number of parameters to be estimated by the factor (m_1+1)(m_2-1)+(m_2+1). Adding hidden layers adds complexity, but also increases the number of parameters to estimate, thus increasing the training time, the number of training examples necessary for robustly training the NN and, last but not least important, the risk that the optimization procedure may converge to a local, rather than a global, optimum.

2.2.3 Jump NN

An alternative to a pure FFNN is a *jump NN*, i.e. a FFNN in which the inputs x_i have direct linear links, called *jump connections*, to the output, as well as to the output through the hidden layers [88]. Figure 2.4 shows a FFNN with jump connections (in dark red) with 4 inputs, one hidden layer with 5 neurons and one output neuron. Using the

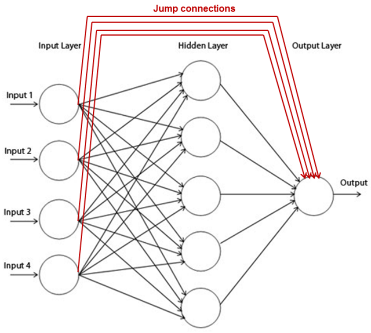


Figure 2.4: Block scheme of a jump NN (jump connections are in dark red).

same symbols introduced in Subsection 2.2.2 the jump NN of Figure 2.4 is represented, mathematically, as

$$y_k = \sum_{j=1}^{m_1} w_{kj}^{2-1} \varphi \left(\sum_{i=0}^{m_0} w_{ji}^{1-0} x_i \right) + \sum_{i=0}^{m_0} w_{ki}^{2-0} x_i \tag{2.9}$$

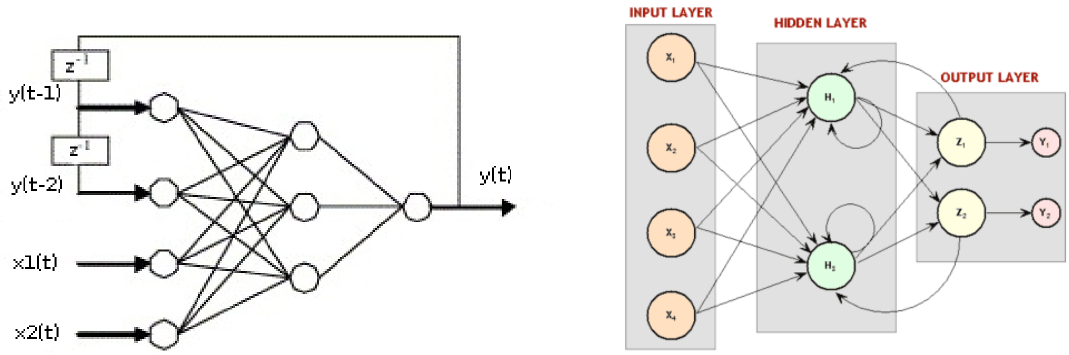
whose first term coincides with equation (2.7) and the second term is relative to the direct connections between inputs and output.

An advantage of a jump NN is that it nests the pure linear model and the FFNN model and allows the possibility that a function may have both, a linear and a nonlinear component. As a consequence, such an architecture is particularly appropriate when the function to be learned has linear and nonlinear relationships with the regressors.

2.2.4 Recurrent NN

A *recurrent NN*, differently from a FFNN, has at least one feedback loop. There might be both, *self-feedback loops*, if the output of a neuron is fed back into its own input and *non self-feedback loops*, if the output of a neuron is fed back as input of neurons of the previous layers. Figure 2.5(a) shows a recurrent NN with one hidden layer and feedback, with two delay elements, from the output to the inputs. Figure 2.5(b) shows a recurrent NN with self feedback loops in the neurons of the hidden layer and feedback connections from the output to the hidden layer.

Let us define $y(t-1)$ the output of the recurrent NN of Figure 2.5(a) at time $t-1$,



(a) Recurrent NN with one hidden layer and one output neuron, with recursion (with two delay elements z^{-1} from output to inputs).

(b) Recurrent NN with one hidden layer, with self feedback connections and two output neurons, whose output is fed back into hidden neurons [89].

Figure 2.5: Two examples of recurrent NNs.

the following equation represents the system schematized in Figure 2.5(a)

$$y(t) = w_0^2 + \sum_{j=1}^{m_1} w_j^{2-1} \varphi \left(\sum_{i=0}^{N_I} w_{ji}^{1-0} x_i + w_j^{z^{-1}} y(t-1) + w_j^{z^{-2}} y(t-2) \right) \tag{2.10}$$

where w_0^2 represents the bias relative to the output neuron and $w_j^{z^{-k}}$ is the weight connecting the output, delayed of k-steps, to the j^{th} hidden neuron.

As we can deduce from equation (2.10), the presence of feedback loops introduces a memory in the evolution of the neurons, profoundly changing the learning ability and the performance of the NN and requiring the use of delay elements, which result in a dynamical nonlinear behavior.

2.3 NN training

One of the most important properties of a NN is its ability of learning from the environment, adjusting properly its synaptic weights. In line with a definition of Mendel and McClaren [90], learning may be defined as [87]

> *Learning is a process by which the free parameters of a neural network are adapted through a process of stimulation by the environment in which the network is embedded. The type of learning is determined by the manner in which the parameters changes take place.*

This definition implies three fundamental steps:

1. The NN is stimulated by an environment.
2. The NN undergoes changes in its weights as a result of this stimulation.
3. The NN responds in a new way to the environment because of the changes occurred in its free parameters.

There are several learning algorithms, mainly differing in the way in which the synaptic weights are adjusted. In Subsection 2.3.3 we will describe in details the backpropagation learning paradigm, which is one of the most widely used in the literature and, furthermore, is the technique adopted for training the NNs presented in this thesis. For more information on learning algorithms we refer the reader to [87].

2.3.1 Learning paradigms

We can distinguish between two learning paradigms: supervised and unsupervised training.

2.3.1.1 Supervised training

In supervised training, the knowledge of the environment is represented by a set of input-output examples, which constitute the training set. The NN is exposed to a training vector and its parameters are adjusted on the basis of the error signal, i.e. the distance between the NN output and the desired response. The adjustment continues iteratively step-by-step, with the final goal of making the NN respond as accurately as possible to all the training examples.

2.3.1.2 Unsupervised training

In unsupervised training there are no labelled examples of the function to be learned. Two paradigms are possible.

Reinforcement learning/ Neurodynamic programming: the input-output mapping is learned through continuous interaction with the environment to minimize a scalar index of performance. The goal is the minimization of a *cost-to-go function*, i.e. the expectation of the cumulative cost of actions, taken over a sequence of steps, instead of the immediate cost. The delay in generating the reinforcement signal implies that the NN must assign credit and blame individually to each action that led to the final outcome, while the primary reinforcement may only evaluate the outcome.

Unsupervised learning: the NN should learn autonomously a representation of the environment, by extracting the statistical regularities of the input data and forming internal representations for encoding features of the input, creating automatically classes.

2.3.2 Learning task

The choice of a learning strategy is determined by the learning task that a NN is required to perform. These tasks might be, for example: pattern association, pattern recognition, function approximation, control, filtering. In this thesis NN will be used for prediction, thus, below, we will shortly introduce the task of *function approximation*.

In function approximation, given an unknown functional relationship

$$\mathbf{y} = \mathbf{f}(\mathbf{x}) \tag{2.11}$$

with $\mathbf{x}$ input and $\mathbf{y}$ output and a set of N labelled examples

$$\mathcal{T} = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{N} \tag{2.12}$$

the task is to design a NN able to approximate the unknown function $\mathbf{f}(\cdot)$ such that the actual network output function $\mathbf{F}(\cdot)$ is close enough to the target over all inputs. Usually, the Euclidean distance is used to measure the goodness of the approximation and the NN weights are chosen to minimize

$$\|\mathbf{F}(\mathbf{x}) - \mathbf{f}(\mathbf{x})\| < \varepsilon \quad \text{for all } \mathbf{x} \tag{2.13}$$

with ε small positive number. If the size N of the training set is large enough and the NN has enough free parameters, the approximation error ε can be small enough for the task. The approximation problem is an ideal candidate for supervised learning.

2.3.3 Learning algorithm

The most commonly used learning algorithm for FFNN (and jump NN), trained for function approximation is the backpropagation algorithm. To ease its derivation, which is rather complex, we summarize the notation used in the following paragraph.

2.3.3.1 Notation

- Indexes i, j and k refers to different neurons; with signals propagating through the NN strictly from left to right, neuron j lies in a layer to the right of neuron i and neuron k lies in a layer to the right of neuron j (following the alphabetical order of the indexes).

- During the n^{th} iteration the n^{th} training example is presented to the network.
- $\mathcal{E}(n)$ is the sum of error squares at iteration n. $\mathcal{E}_{av}$ is the average error over all the training examples.
- $d_j(n)$ is the desired response for neuron j.
- $y_j(n)$ is the output of neuron j.
- $e_j(n)$ is the error at the output of neuron j at iteration n, thus $e_j(n) = d_j(n) - y_j(n)$.
- $w_{ji}(n)$ is the weight connecting neuron i to neuron j at iteration n.
- $\Delta w_{ji}(n)$ is the correction applied to the weight w_{ji} at iteration n.
- $v_j(n)$ is the the weighted sum of all synaptic inputs plus bias of neuron j at iteration n.
- $\varphi_j(\cdot)$ is the activation function describing the input-output nonlinear relationship between input and output of neuron j.
- $w_{j0} = b_j$ is a weight associating a fixed input equal to 1 to neuron j and represents the bias applied to neuron j.
- $x_i(n)$ is the i^{th} element of the n^{th} input vector.
- $o_k(n)$ is the k^{th} element of the n^{th} output vector.
- η is the learning rate.
- m_l is the number of neurons in layer l, with $l = 0, \dots, L$. Thus m_0 is the size of the input layer, m_1 is the size of the first hidden layer and m_L is the number of outputs.

2.3.3.2 Backpropagation algorithm

The backpropagation algorithm is the most common routine for training FFNN. There are many variants of this algorithm, to optimize its speed and to minimize the risk of getting trapped in local minima. There is also a dynamic version, namely *backpropagation through time*, proposed for training recurrent NNs. However, it is less efficient than the original algorithm both, in terms of speed and performance of the trained network.

The algorithm consists in two steps: a *forward pass* and a *backward pass*.

- In the *forward pass*, a training input is presented to the NN and the corresponding output values are computed. Each output is compared with the corresponding target and the error committed by every output neuron is computed.

- In the *backward pass*, the synaptic weights are modified applying the error-correction rule. The error is backpropagated through the network, from the output to the input, trying to infer the error committed by every hidden neuron (of which both, target and actual output are unknown), modifying the weights to minimize the error.

The error relative to the output neuron j at iteration n is

$$e_j(n) = d_j(n) - y_j(n) \tag{2.14}$$

The instantaneous value of the error energy for neuron j is defined as $\frac{1}{2}e_j^2(n)$. The value $\mathcal{E}(n)$ of the total error energy is obtained summing $\frac{1}{2}e_j^2(n)$ over all the neurons of the output layer, i.e. over all the neurons for which the error signal can be calculated directly,

$$\mathcal{E}(n) = \frac{1}{2}\sum_{j\in C} e_j^2(n) \tag{2.15}$$

where C is the set including all the neurons of the output layer. Let N denote the number of examples in the *training set*. The average squared error energy is obtained as

$$\begin{aligned}\mathcal{E}_{av} &= \frac{1}{N}\sum_{n=1}^{N}\mathcal{E}(n) = \frac{1}{2N}\sum_{n=1}^{N}\sum_{j\in C} e_j^2(n)\\ &= \frac{1}{2N}\sum_{n=1}^{N}\sum_{j\in C}(d_j(n) - y_j(n))^2 \end{aligned}\tag{2.16}$$

Both, $\mathcal{E}(n)$ and $\mathcal{E}_{av}$ are functions of the NN weights and biases. For a given training set, $\mathcal{E}_{av}$ represents the cost function that should be minimized by adjusting the free parameters of the NN.

The NN training can be performed:

- *online*, modifying the weights after each example of the training set, i.e. minimizing at each iteration $\mathcal{E}(n)$;

- in *batch* mode, modifying the weights on the basis of the error computed on the entire training set, i.e. minimizing $\mathcal{E}_{av}$.

In the rest of the paragraph we will consider the case in which weights are updated online.

Let us define the input of neuron j as

$$v_j(n) = \sum_{i=0}^{m} w_{ji}(n) y_i(n) \tag{2.17}$$

with m total number of inputs applied to neuron j and w_{j0} (corresponding to the fixed input $y_0 = 1$) weight associated with the bias b_j. The output of neuron j is

$$y_j(n) = \varphi_j(v_j(n)) \tag{2.18}$$

The backpropagation algorithm applies a correction $\Delta w_{ji}(n)$ to the weight w_{ji}, proportional to the partial derivative of the error $\partial \mathcal{E}(n)/\partial w_{ji}(n)$ (or of the average error, in batch mode $\partial \mathcal{E}_{av}/\partial w_{ji}$). Using the chain rule the gradient can be expressed as

$$\frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} = \frac{\partial \mathcal{E}(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \frac{\partial v_j(n)}{\partial w_{ji}(n)} \tag{2.19}$$

Differentiating both sides of (2.15) we obtain

$$\frac{\partial \mathcal{E}(n)}{\partial e_j(n)} = e_j(n) \tag{2.20}$$

Differentiating both sides of (2.14) we obtain

$$\frac{\partial e_j(n)}{\partial y_j(n)} = -1 \tag{2.21}$$

Differentiating (2.18) with respect to $v_j(n)$ we obtain

$$\frac{\partial y_i(n)}{\partial v_j(n)} = \varphi_j'(v_j(n)) \tag{2.22}$$

where $\varphi_j'(\cdot)$ is the derivative of $\varphi_j(\cdot)$ with respect to its argument. Finally, differentiating (2.17) with respect to $w_{ji}(n)$ we have

$$\frac{\partial v_j(n)}{\partial w_{ji}(n)} = y_i(n). \tag{2.23}$$

Substituting equations from (2.20) to (2.23) in (2.19) yields

$$\frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} = -e_j(n)\varphi_j'(v_j(n))y_i(n) \tag{2.24}$$

The correction $\Delta w_{ji}(n)$ applied to $w_{ji}(n)$, is defined as

$$\begin{aligned}\Delta w_{ji}(n) &= -\eta \frac{\partial \mathcal{E}(n)}{\partial w_{ji}(n)} \\ &= \frac{\partial \mathcal{E}(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \frac{\partial v_j(n)}{\partial w_j i(n)} \\ &= \eta \delta_j(n) y_i(n) \end{aligned} \tag{2.25}$$

with η learning rate and $\delta_j(n)$ local gradient defined as

$$\begin{aligned}\delta_j(n) &= -\frac{\partial \mathcal{E}(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \\ &= e_j(n)\varphi_j'(v_j(n)) \end{aligned} \tag{2.26}$$

From equations (2.25) and (2.26) we note that $e_j(n)$ is key factor in computing the weight adjustment $\Delta w_{ji}(n)$. Depending on the position of neuron j in the network we may distinguish two cases: if j is an output neuron computing its error is trivial, if j is a hidden neuron its output and its error are not directly accessible. The problem is to know how to penalize hidden neurons for their responsibility in determining the output errors. This issue is solved by backpropagating the error signals through the NN.

Case 1: neuron j is an output node. In this case we know the target signal, thus we can use equation (2.14) to compute the error $e_j(n)$ and then compute the local gradient $\delta_j(n)$ using equation (2.26).

Case 2: neuron j is a hidden node.

From (2.26) we may redefine the local gradient of a hidden neuron j as

$$\begin{aligned}\delta_j(n) &= -\frac{\partial \mathcal{E}(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \\ &= -\frac{\partial \mathcal{E}(n)}{\partial y_j(n)} \varphi_j'(v_j(n)) \end{aligned} \tag{2.27}$$

where we used (2.22) in the second line. As seen in (2.15), if k is an output node its error is

$$\mathcal{E}(n) = \frac{1}{2} \sum_{k \in C} e_k^2(n) \tag{2.28}$$

which corresponds to (2.15), with k used instead of j for avoiding confusion, since j refers to a hidden neuron under *case 2*. Differentiating (2.28) with respect to $y_j(n)$ we get

$$\begin{aligned} \frac{\partial \mathcal{E}(n)}{\partial y_j(n)} &= \sum_{k \in C} e_k \frac{\partial e_k(n)}{\partial y_j(n)} \\ &= \sum_{k \in C} e_k \frac{\partial e_k(n)}{\partial v_k(n)} \frac{\partial v_k(n)}{\partial y_j(n)} \end{aligned} \tag{2.29}$$

where we applied the chain rule in the second row. From (2.14) and (2.18) we know that

$$\begin{aligned} e_k(n) &= d_k(n) - y_k(n) \\ &= d_k(n) - \varphi_k(v_k(n)) \end{aligned} \tag{2.30}$$

thus

$$\frac{\partial e_k(n)}{\partial v_k(n)} = -\varphi_k'(v_k(n)) \tag{2.31}$$

From (2.17) we also note that the input of neuron k is

$$v_k(n) = \sum_{j=0}^{m} w_{kj}(n) y_j(n) \tag{2.32}$$

and differentiating with respect to $y_j(n)$

$$\frac{\partial v_k(n)}{\partial y_j(n)} = w_{kj}(n) \tag{2.33}$$

Using (2.31) and (2.33) in (2.29) we obtain the desired partial derivative

$$\begin{aligned} \frac{\partial \mathcal{E}(n)}{\partial y_j(n)} &= -\sum_{k \in C} e_k(n) \varphi_k'(v_k(n)) w_{kj}(n) \\ &= -\sum_{k \in C} \delta_k(n) w_{kj}(n) \end{aligned} \tag{2.34}$$

Finally, using (2.34) in (2.27) we get the backpropagation formula for the local gradient $\delta_j(n)$ relative to a hidden neuron

$$\delta_j(n) = \varphi_j'(v_j(n)) \sum_{k \in C} \delta_k(n) w_{kj}(n) \tag{2.35}$$

For what concerns the factors involved in the computation of the local gradient:

- $\varphi'_j(v_j(n))$ depends solely on the activation function of neuron j;
- $\delta_k(n)$, with $k \in C$, require knowledge of the error $e_k(n)$ for all neurons that lie in the layer to the immediate right of hidden neuron j and that are directly connected to neuron j;
- $w_{kj}(n)$ are the synaptic weights associated with these connections, i.e. the connections between neuron j and the neurons in the layer to its immediate right.

Summarizing, in the backpropagation algorithm:

1. Input n is presented to the NN and the error committed by the network is computed.
2. The correction $\Delta w_{ji}(n)$ is computed as

$$\begin{pmatrix} \text{Weight} \\ \text{correction} \\ \Delta w_{ji}(n) \end{pmatrix} = \begin{pmatrix} \text{learning} \\ \text{rate} \\ \eta \end{pmatrix} \cdot \begin{pmatrix} \text{local} \\ \text{gradient} \\ \delta_j(n) \end{pmatrix} \cdot \begin{pmatrix} \text{input signal} \\ \text{of neuron } j \\ y_i(n) \end{pmatrix}$$

   The local gradient $\delta_j(n)$ depends on whether neuron j is an output node or a hidden node. If neuron j is an output node $\delta_j(n) = \varphi'_j(v_j(n))e_j(n)$. If neuron j is a hidden node $\delta_j(n)$ is obtained by the product of $\varphi'_j(v_j(n))$ and the weighted sum of the δ_s computed for the neurons in the next layer that are connected with neuron j.

3. Iterations continue until a minimum of $\mathcal{E}(n)$ is reached.

In the *batch learning* case $\mathcal{E}_{av}$ is minimized instead of $\mathcal{E}(n)$. Thus, equation (2.19) becomes

$$\frac{\partial \mathcal{E}_{av}}{\partial w_{ji}} = \frac{1}{N} \sum_{n=1}^{N} \frac{\partial \mathcal{E}(n)}{\partial e_j(n)} \frac{\partial e_j(n)}{\partial y_j(n)} \frac{\partial y_j(n)}{\partial v_j(n)} \frac{\partial v_j(n)}{\partial w_j i(n)} \tag{2.36}$$

Apart from the introduction of the sum over all the elements of the training set, the procedure corresponds to that previously presented and Δw_{ji} is given by

$$\Delta w_{ji} = \eta \frac{1}{N} \sum_{n=1}^{N} \delta_j(n) y_i(n) \tag{2.37}$$

It is worth stressing that with *batch learning* the NN weights remain constant and all the elements of the training set are presented to the network. Then the average error is computed and the weights are updated along the delta-rule, to minimize the error.

Generally, the backpropagation algorithm cannot be shown to converge and there are no well defined criteria to stop it. Reasonable criteria derive from the characterization of a global or local minimum in the error surface. Let the weight vector $\mathbf{w}*$ denote a minimum. A necessary condition is that the gradient vector of the error surface with respect to the weight $\mathbf{w}$ is zero at $\mathbf{w} = \mathbf{w}*$. Thus a convergence criterion might be a sufficiently small Euclidean norm of the gradient vector. Alternatively, we might define a criterion exploiting the fact that the cost function $\mathcal{E}_{av}$ is stationary at $\mathbf{w} = \mathbf{w}*$. Thus the backpropagation algorithm may be considered to have converged when the absolute rate of change in the average square error per epoch is sufficiently close to zero. Another criterion usually adopted is based on the generalization ability of the trained NN and is discussed in Subsection 2.3.4.

2.3.4 Generalization in NN

In backpropagation the synaptic weights of the FFNN are computed by learning the training examples as accurately as possible. The hope is that the trained NN will generalize well on test data, similar to those seen during the training procedure, but never used to optimize the weights. In fact, one of the problems that may occur during NN training is *overfitting*, i.e. the network memorizes the training data and finds feature that are due to noise, but not informative of the function to be modelled.

One method for improving network generalization is to keep the complexity of the NN low and using a number of neurons just large enough to adequately fit the target function [91]. In addition, increasing the size of the training set is another good option to prevent overtraining. However, often a huge training set is not available and a NN with a few neurons might not be adequate for learning the function of interest. Thus, two techniques commonly used are *early stopping* and *regularization*.

2.3.4.1 Early stopping

Ordinarily, a FFNN learns in stages, increasing the performance in the training set as the training session progresses, towards a local minimum of the error surface. However, the NN might end up overfitting the training data and generalizing poorly. The onset of overfitting can be identified using *cross-validation*: the training data are split into an effective training set, used for computing the error and its gradient and updating the network weights and a validation set, used for monitoring the error during training. The training session is stopped periodically and the error on the validation set is computed. The validation error normally decreases during the initial phase of training, however, when the network begins to overfit the data, the error on the validation set begins to rise.

When the error increases for a specified number of consecutive iterations the training is stopped and the weights at the minimum of the validation error are returned. This procedure is referred to as *early stopping* and was presented in [92].

2.3.4.2 Regularization

Another method for improving generalization is using *regularization.* This involves modifying the performance function and minimizing

$$R(\mathbf{w}) = \mathcal{E}_s(\mathbf{w}) + \lambda \mathcal{E}_c(\mathbf{w}) \tag{2.38}$$

The first term, $\mathcal{E}_s(\mathbf{w})$, is the standard performance measure, depending on the network weights and the input data. The second term, $\mathcal{E}_c(\mathbf{w})$, is the model complexity penalty and λ is a regularization parameter, representing the relative importance of the complexity penalty term with respect to the performance measure term.

A popular choice for regularization is the *weight decay* procedure, proposed in [93]. The complexity penalty term is defined as

$$\begin{aligned} \mathcal{E}_c(\mathbf{w}) &= \|\mathbf{w}\|^2 \\ &= \sum_{i \in \mathcal{C}_{total}} w_i^2 \end{aligned} \tag{2.39}$$

with $\mathcal{C}_{total}$ set of all the synaptic weights in the network. This procedure forces some of the weights to take values close to zero. Accordingly, the network weights are grouped into two categories: those having a great influence in the model performance and those having almost no influence on it. The latter are likely to take completely arbitrary values and might lead to poor generalization performance.

2.4 NN structure optimization

The NN architecture defines its structure, including the number of hidden layers and of neurons in each layer. The number of input and output neurons is easy to determine, since it corresponds to the the number of input and output variables, respectively. On the contrary, the determination of the appropriate number of hidden layers and hidden neurons is a critical task, since no prior knowledge are usually available, nor statistic or theoretical rules. A NN with one hidden layer and an appropriate number of hidden nodes should be able of approximating any function (see Section 2.6 for details). In practice, NN with one or two hidden layers are commonly used with satisfactory performance.

Analogously, there is no formula to select the number of hidden neurons, thus this choice involves experimentation and simulation. A network with too few hidden nodes would not be able to learn accurately enough the training data. On the other side, a network with too many hidden nodes is likely to overfit the data. Three methods are commonly used for optimizing the number of hidden neurons: *fixed* approach, *network growing or constructive* approach, and *network pruning or destructive* approach [94, 95].

In the fixed approach several NNs are trained and each is evaluated. Usually, N-fold-cross-validation is adopted: the training set is divided into N subsets, each network is trained on $N-1$ subsets and tested on the remaining subset and the procedure is repeated N times, leaving out each time a different subset for testing the models. The performance of each NN is the average obtained on the N experiments. The increment in the number of hidden neurons might be one, two or more (or logarithmic). The network with the smallest error is selected because it is able to generalize best.

The constructive and destructive approaches involve changing the number of hidden neurons during training and this functionality is not supported by the majority of commercial NN software packages. The constructive approach starts with the smallest possible architecture and continues adding hidden neurons until the network performance is stable or begins deteriorating. The destructive approach starts with a big network and continues removing neurons until the performance begins deteriorating.

Regardless of the method chosen for optimizing the NN architecture, the rule is to choose the NN that performs best on a validation set, with the smallest number of neurons.

2.5 Data preprocessing

Several preprocessing techniques are commonly applied before the data are used for training the NN to accelerate convergence and to ease the problem to be learned [95]. The most common are noise removal, input dimensionality reduction and feature extraction, data transformation, data inspection with outliers deletion.

An essential operation is scaling data so that all the regressors have similar variance and span the same range of values. This is usually done by applying a linear mapping of the training set in the range $[-1,\ 1]$, i.e. the interval of input values in which the sigmoid functions (both hyperbolic tangent and logistic) have a linear behaviour. Thus a signal x assuming values in the range $[x_{min},\ x_{max}]$ is mapped in the set $[y_{min},\ y_{max}]$ by applying

$$y = (y_{max} - y_{min}) \frac{x - x_{min}}{x_{max} - x_{min}} + y_{min} \tag{2.40}$$

Alternatively, the signal is mapped to have zero mean and unitary standard deviation, by applying

$$y = \frac{x - x_{mean}}{x_{sd}} \tag{2.41}$$

where x_{mean} is the average of the values assumed by the signal x, on the training set and x_{sd} is its standard deviation. Data normalization is essential to prevent larger numbers from overriding smaller ones and to prevent premature saturation of hidden nodes, which would deteriorate the learning process.

In practice data preprocessing involves trial and error: one method to select appropriate input variables is to test various combinations, however, it might be not possible to do it exhaustively. Other alternative methods include [96]:

1. Methods that use *a priori* knowledge of the system being modelled. A priori knowledge and a good understanding of the system to be modelled are essential for selecting a set of candidate inputs, however, this method should, possibly, be combined with other analytical approaches.

2. Methods based on *cross-correlation*. This is one of the most popular analytical techniques for inputs selection. The major disadvantage of this technique is the fact that it captures only linear dependence between two variables, while it is unable to detect any nonlinear dependence.

3. Methods using *heuristic approaches*. These methods comprise step-wise addition of variable to the set of inputs, backward elimination of inputs and comparison of different networks trained with different subsets of inputs. Since these approaches are based on trial-and-error, there is no guarantee that they will find the globally best subset. Moreover, they are computationally expensive.

4. Methods extracting knowledge from trained NN. These methods rely on the computation of *sensitivity* of the output with respect to each input to choose which inputs should be removed. The difficulty of this approach is selecting the appropriate cut-off point for input significance and choosing an appropriate method for computing sensitivity.

5. Methods combining the above four approaches.

2.6 NN for function approximation

A multilayer perceptron trained with backpropagation (see Subsection 2.3.3) can perform a *nonlinear input-output mapping*, as stated in the

Theorem 2.6.1. Universal Approximation Theorem
Let $\varphi(\cdot)$ be a non-constant, bounded and monotone-increasing function. Let I_{m_0} denote the m_0 dimensional hypercube $[0,\ 1]^{m_0}$. The space of continuous functions on I_{m_0} is denoted by $C(I_{m_0})$. Then, given any function $f \in C(I_{m_0})$ and $\varepsilon > 0$, there exist an integer M and sets of real constants α_i, θ_i, e w_{ij}, where $i = 1, \ldots, m_1$ and $j = 1, \ldots, m_0$ so that we may define

$$F(x_1, \ldots, x_{m_0}) = \sum_{i=1}^{m_1} \alpha_i \varphi \left(\sum_{j=1}^{m_0} w_{ij} x_j + b_i \right) \tag{2.42}$$

as an approximate realization of the function $f(\cdot)$, that is

$$\mid F(x_1, \ldots, x_{m_0}) - f(x_1, \ldots, x_{m_0}) \mid < \varepsilon$$

for all $\{x_1, \ldots, x_{m_0}\} \in I_{m_0}$.

The universal approximation theorem is directly applicable to FFNN: the logistic and the tangent hyperbolic function ($\varphi(v) = 1/(1 + e^{-2v})$ and $\varphi(v) = (1 + e^{-2v})/(1 + e^{-2v})$ respectively), used as activation function of hidden neurons of a multilayer perceptron are both non-constant, bounded and monotone increasing function and, thus, satisfy the requirements for $\varphi(\cdot)$. Furthermore, equation (2.42) represents the output of a FFNN with m_0 inputs, denoted as $x_1, \ldots, x_{m_0}$, and a unique hidden layer with m_1 neurons. Hidden neuron i has synaptic weights $w_{i_1}, \ldots, w_{i_{m_0}}$ and bias b_i and the network output is a linear combination of the hidden layer outputs weighted with $\alpha_1, \ldots, \alpha_{m_1}$. The universal approximation theorem guarantees that a FFNN with a single hidden layer can approximate a given training set represented by the set of inputs $x_1, \ldots, x_{m_0}$ and a target output $f(x_1, \ldots, x_{m_0})$.

This theorem is important from a theoretical point of view, however, it does not specify how to determine the multilayer perceptron architecture. Furthermore, it assumes that the target function is known (without errors) and that a hidden layer, potentially of unlimited size, could be used. Both these assumptions are usually violated and in several applications more than one hidden layer is used.

In the context of function approximation, the use of backpropagation offers another useful property. A FFNN with smooth activation functions should have output function derivatives that can approximate the derivative of the unknown input-output mapping. A theoretical proof of this statement is presented in [97] where it is shown that multilayer perceptrons can approximate functions not differentiable in the classical sense, but with generalized derivatives, as in the case of piecewise differentiable functions. This theorem justifies the use of FFNN in applications that require the approximation of both, a function end its derivative.

2.7 NN models for glucose prediction: the chosen design and implementation strategy

In NN theory there is no clear mathematically proven formula for successful network modelling. Thus several popular rule of thumbs are normally adopted. In this section we will describe our design choices, adopted for implementing and optimizing the NN models described in Chapters 3, 4 and 5.

2.7.1 Input signals selection

The inputs of all the NN models were chosen using a mixed approach, exploiting *a priori* knowledge, *cross-correlation* results and *N-fold-cross-validation* analysis. Indeed we used prior knowledge from physiology to select available information which are known to influence glucose concentration dynamics and generate (preprocessing this information) a set of possible input signals. Afterwards we performed a cross-correlation analysis, on the training set, between the target glucose concentration and every candidate input and we repeated this analysis for various time lags of the candidate input. In this way we determined if, effectively, a significant correlation between target and candidate input exists and we also established whether a time lag should be applied to the candidate input signal. Finally, in case there were multiple signals, relative to the same piece of information, significantly correlated with future glycemia and in case the same signal was significantly correlated for different time lags we used N-fold-cross-validation on the training set, to test all the possible combination of inputs and select the best one.

It is worth noting that, before training the NN, all the selected input signals were mapped to assume values merely in the range $[-1, 1]$.

2.7.2 Structure optimization

The method we adopted for optimizing the NN structure is based on N-fold-cross-validation, applied on the training set. For every developed NN model, we started with a unique hidden layer with one neuron and we increased the number of neurons one by one, testing the performance of every model using the three metrics described in Appendix C. Increasing the number of hidden neurons initially improves the NN performance, however, from a certain point inwards, results begin to worsen or the improvement is close to zero.

Thus we selected the model that obtained acceptable performance with as few neurons as possible. The selection was done primarily on the basis of the RMSE, however Time Gain (TG) and ESOD were also used in case multiple models could be selected on the basis of the RMSE.

A preliminary analysis showed that NN models with two hidden layers did not perform better than NNs with a unique hidden layer. For this reason we concentrated on structures with only one hidden layer of neurons.

2.7.3 NN training

All the proposed NN models were trained with *backpropagation* implemented in the *Levenberg-Marquardt* variant, applied in batch mode [98, 99]. The Levenberg-Marquardt algorithm, like the quasi-Newton methods, was designed to approach second-order training speed without having to compute the Hessian matrix. When the performance function has the form of a sum of squares (as is typical in training FFNN), then the Hessian matrix can be approximated as

$$\mathbf{H} = \mathbf{J}^T \mathbf{J} \tag{2.43}$$

and the gradient can be computed as

$$\mathbf{g} = \mathbf{J}^T \mathbf{e} \tag{2.44}$$

where $\mathbf{J}$ is the Jacobian matrix that contains first derivative of the network errors with respect to the weights and biases, and $\mathbf{e}$ is a vector of network errors. The Jacobian matrix can be computed through a standard backpropagation technique [99] that is much less complex than computing the Hessian matrix.

The Levenberg-Marquardt algorithm uses this approximation to the Hessian matrix in the following Newton-like update:

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \left(\mathbf{J}^T \mathbf{J} + \mu \mathbf{I}\right)^{-1} \mathbf{J}^T \mathbf{e} \tag{2.45}$$

When the scalar μ is zero, this is just Newton's method, using the approximate Hessian matrix. When μ is large, this becomes gradient descent with a small step size. Newton's method is faster and more accurate near an error minimum, so the aim is to shift toward Newton's method as quickly as possible. Thus, μ is decreased after each successful step (reduction in performance function) and is increased only when a tentative step would increase the performance function. In this way, the performance function is always reduced at each iteration of the algorithm.

The original description of the Levenberg-Marquardt algorithm is given in [98] and its application to neural network training is described in [99]. This algorithm appears to be the fastest method for training moderate-size FFNNs (up to several hundred weights) and has an efficient implementation in Matlab®.

Early stopping was used for terminating the training routine. Thus, for every NN model developed the training and validation set was randomly split into the effective training set constituted by 70% of the data and the validation set formed by the remaining 30% of data. After every iteration on the whole training set the NN weights were updated and the algorithm was tested on the validation set, to check if over-fitting was occurring. Training was stopped when, for 100 consecutive times, the validation performance had not increased and the weights of the last successful validation test were kept.

We also tested the backpropagation with *Bayesian regularization* training algorithm. This technique updates the weight and bias values according to Levenberg-Marquardt optimization, minimizing a combination of squared errors and squared weight values so that, at the end of training, the resulting network has good generalization without using early stopping. In addition, the unnecessary weights should assume values close or equal to zero at the end of the training and should, potentially, be eliminated by the NN without compromising its performance. This training procedure gave results comparable to those obtained with the classical backpropagation, however it was considerably more time consuming, thus the classical Levenberg-Marquardt algorithm was adopted. Furthermore, using Bayesian regularization all the weights resulted significant at the end of the training, confirming also that the chosen NN architecture was parsimonious.

One of the limitations of Levenberg-Marquardt backpropagation derives from the use of the Jacobian for calculations, which assumes that performance is a mean or sum of squared errors. Therefore the objective function minimized during training must be the MSE or the SSE. Despite MSE and its variants (e.g. SSE, RMSE, etc) are widely used for assessing the performance of glucose concentration prediction algorithms, these metrics are suboptimal, as discussed in Appendix C. Indeed during training we might want to take into account also the time anticipation of prediction, the adherence of the derivative of the predicted time series to the derivative of the target signal and we might aim to assign a higher penalty to overestimation of hypoglycemia and underestimation of hyperglycemia, than to underestimation of hypoglycemia and overestimation of hyperglycemia. This is not possible if the NN is trained using functions implemented in the Matlab Neural Network toolbox.

We performed a preliminary analysis training the NN using a Genetic Algorithm (GA) followed by a gradient descent method with initial parameters equal to the best solution found by the GA. As possible objective function we considered:

- A regularized MSE for limiting spurious oscillations due to noise amplification in the predicted time series. Thus the objective function minimized was:

$$J = \|\hat{\mathbf{y}} - \mathbf{y}\|^2 + \gamma \|\ddot{\hat{\mathbf{y}}}\|^2 \tag{2.46}$$

  where $\ddot{\hat{\mathbf{y}}}$ represents the second order time derivative of $\hat{\mathbf{y}}$.

- A function penalizing both, deviation of prediction and of prediction derivative from target and target derivative, respectively

$$J = \|\hat{\mathbf{y}} - \mathbf{y}\|^2 + \gamma \|\dot{\hat{\mathbf{y}}} - \dot{\mathbf{y}}\|^2 \tag{2.47}$$

  where $\dot{\mathbf{y}}$ represents the first order time derivative of $\mathbf{y}$.

- The gluco-specific MSE proposed in [100], which modifies MSE with a Clark error grid inspired penalty function, which penalizes overestimation in hypoglycemia and underestimation in hyperglycemia.

This training routine required a considerably higher time than Levenberg-Marqardt backpropagation and gave no global improvement of prediction performance. For these reasons all the proposed NN models described in the next Chapters will be trained with the standard Levenberg-Marqardt backpropagation algorithm implemented in the Matlab Neural Network toolbox.

However, as future work it might be worth investigating objective functions more adequate for quantifying the goodness of glucose prediction.

2.8 Concluding remarks

As discussed in Section 1.4, the majority of algorithms for glucose concentration prediction uses past CGM readings only as input and does not exploit available information on meal and insulin therapy. One of the reasons is the difficulty of formalizing such information in mathematical terms and of incorporating, among the inputs of the predictor, signals with different characteristics, e.g. glucose concentration, meal and insulin. As we have seen in this chapter, NNs allow the creation of empirical models using heterogeneous information and are thus promising candidates for forecasting CGM utilizing, potentially, all the available information. Moreover, their intrinsically non linear behaviour is an appealing feature for accomplishing the task of learning a complex function as glucose concentration time course.

Our first aim will be the development of a short time (PH = 30 min) NN-based predictor able to exploit information on CGM as well as on time and dose of CHO ingested during meals. This will be accomplished in Chapters 3 and 4.

3

New glucose prediction method by NN plus linear prediction algorithm (NN-LPA)

3.1 Rationale

Rather surprisingly, complex prediction techniques based on NNs, as [66, 77] did not significantly outperform the much simpler strategies based on time-series modelling. For instance, in [66] results obtained, for the same dataset, with the NN strategy are similar to those obtained with the AR(1) algorithm of [59]. Results of [77] indicate that the NN described therein does not outperform the NN of [66], even if the first embeds also information on meal intake, insulin medications, emotions and physical exercise.

A possible justification of this disappointing performance of NNs lies in the way NNs have been used in [66, 77]. The following example motivates this assertion. Figure 3.1 displays a CGM time series (black dotted line) of a representative real subject. The plot also shows the profile predicted by a simple linear strategy, the first order polynomial algorithm of [59] (referred as poly(1) hereafter), with PH = 30 min (gray line). The plot is restricted to a 8 h time interval to allow to better capture, visually, differences between the different profiles. The prediction error of poly(1) is particularly low in the time interval 11:00-14:30 h, where the target time series exhibits limited variability. Conversely, the poly(1) prediction shows an evident loss of accuracy after meals. In fact, CHO intake can be thought as an exogenous disturbance that introduce a new component in glucose dynamics that the linear poly(1) algorithm is not able to track promptly. Since FFNN with nonlinear activation functions in their hidden layers have an intrinsically nonlinear behavior, it would be natural to expect them to significantly improve on the simple poly(1) prediction strategy. On the contrary, as shown in Figure 3.1, the NN (cyan line) prediction of [66] behaves similarly to poly(1) and results inaccurate in correspondence of the meal.

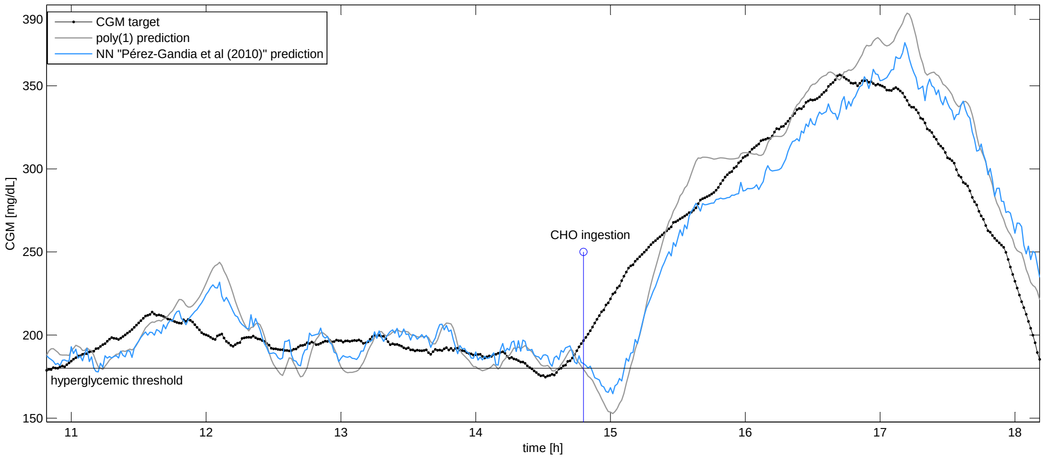


Figure 3.1: Real CGM profile (black dotted line), the prediction with PH=30 min obtained with poly(1) (gray line), and with the NN of [66] (cyan line). Plot taken from [66], (Fig.4). The blue stem denotes CHO intake.

The theoretical potentialities of FFNNs in learning nonlinear relationships appear to be not fully exploited when they have to model both linear and nonlinear components of glucose dynamics. In [101] it has been suggested that when data show a relevant linear pattern, in addition to a minor, but essential nonlinear component, the network could be used in parallel with a linear model. The advantage of this approach is that the linear model extrapolates the slope of the signal, while the NN learns only nonlinear dynamics. Two alternative strategies can be used for identifying the complete model:

- The linear model parameters can be estimated in a first step and, successively, the NN can be trained on the error of the linear model, keeping the linear model fixed.
- The linear model and the NN can be trained together.

The second strategy is more flexible, but the linear model is identified only in combination with the nonlinear NN, thus it might not be a good representation of the process on its own and may result unstable on its own.

We adopt a similar approach for determining the glucose predictor: the NN we design is trained to describe the nonlinear components in glucose dynamics that poly(1) is not able to predict [102]. Indeed the NN model is in parallel with the linear prediction algorithm. For this reason this architecture will be referred as NN-LPA from now inward. This is a first major novelty of this approach, with respect to NNs proposed so far in the literature. A second novelty is that the NN embeds, among its inputs, information on ingested CHO, preprocessed with the physiological model proposed in [103], using population parameters estimated in [104].

3.2 Architecture of the prediction algorithm

In order to ease the explanation of the methodology, in Figure 3.2 we report a block diagram of the glucose predictor.

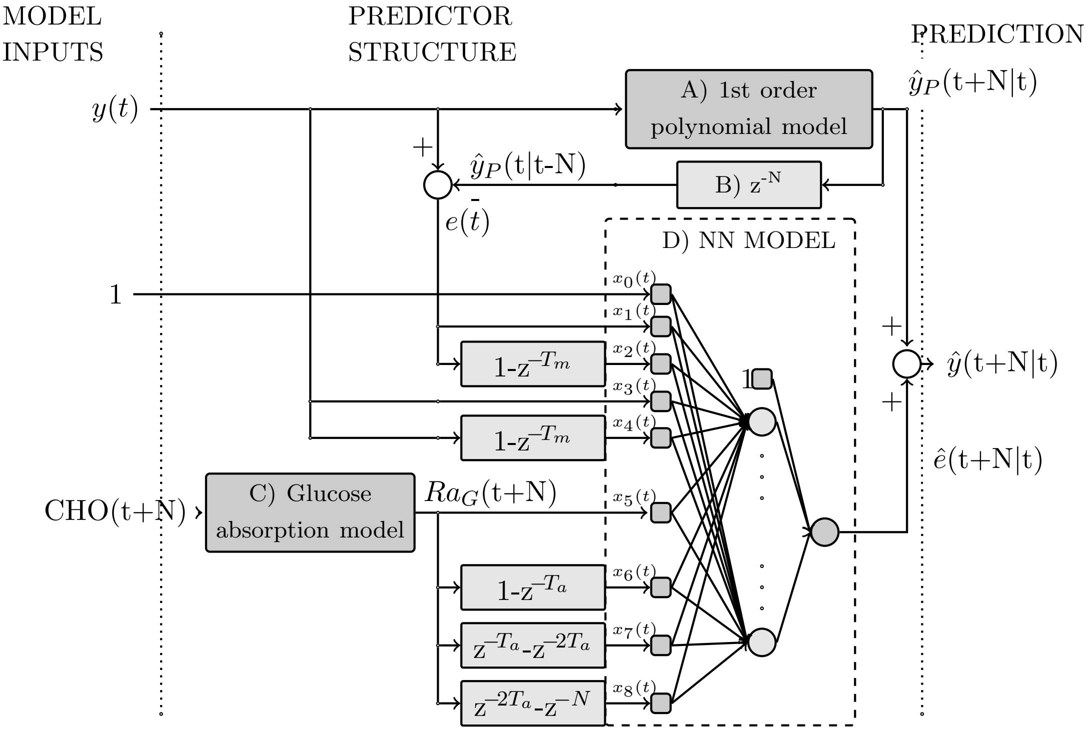


Figure 3.2: Block scheme of the glucose predictor architecture. The model is composed by a NN in parallel with a linear prediction algorithm and is called, for this reason, NN-LPA. In our implementation T_m=15 time steps (i.e. 15 min), and T_a=10 time steps (i.e. 10 min).

Let us introduce the symbol $x(t)$ to indicate the signal x measured at time step t; the symbol $\hat{x}(t_2|t_1)$ to indicate the signal x at time step t_2, predicted using data until time step t_1, N is the PH in number of steps (thus, if the sampling period is of T_s min,

$N = PH/T_s$), while $z^{-k}x(t) = x(t-k)$, i.e. z^{-k} indicates the k step delay operator.

As anticipated in the previous paragraph, $\hat{y}(t+N|t)$, i.e. glucose concentration at time step $t+N$, predicted from data available until time step t, results from the sum of two components, $\hat{y}_P(t+N|t)$ and $\hat{e}(t+N|t)$. The first term $\hat{y}_P(t+N|t)$ is the glucose prediction obtained through a first order polynomial (thus, linear) algorithm (block labelled as “A” in Figure 3.2), on the basis of the past CGM readings. Here the poly(1) method of [59] is used. The calculation of the second term, $\hat{e}(t+N|t)$, which is the estimation of the error committed by the linear predictor, is more complex. A memory block (denoted by “B” in Figure 3.2) stores $\hat{y}_P(t+N|t)$ for N steps and, every time a new glucose level $y(t)$ is provided by the CGM sensor, the error $e(t) = y(t) - \hat{y}_P(t|t-N)$ is computed. The error $e(t)$ and other inputs, which will be described in detail in Subsection 3.2.1, feed a NN which is trained to predict $e(t+N)$, i.e. the error affecting $\hat{y}_P(t+N|t)$ (block “D” in Figure 3.2, details reported in Section 3.3). Finally $\hat{e}(t+N|t)$ is summed to $\hat{y}_P(t+N|t)$, to obtain a better estimate of $y(t+N)$.

3.2.1 Description of the neural network model

The architecture of the network is schematized in block “D” in Figure 3.2. Inputs and outputs are described below. Regarding the NN structure, it presents one hidden layer with 8 neurons, each one with tangent hyperbolic activation function, and an output layer with one neuron with linear transfer function. The network is totally connected and feedforward.

The output of the NN is $\hat{e}(t+N|t)$, i.e. the unknown error affecting $\hat{y}_P(t+N|t)$ (the present poly(1) prediction of $y(t+N)$).

As shown in block “D” of Figure 3.2, the first four inputs are:

- the current prediction error $e(t) = y(t) - \hat{y}_P(t|t-N)$, where $\hat{y}_P(t|t-N)$ is the glycemia predicted N steps before by the linear model, and $y(t)$ is the current glycemia measured by the sensor;
- the trend of the prediction error, in the last T_m steps, $(1 - z^{-T_m})e(t)$.
- the current glucose concentration measured by the CGM sensor $y(t)$;
- the glycemic trend in the last T_m steps $(1 - z^{-T_m})y(t)$, (with $T_m = 15$ steps, i.e. 15 min in our implementation).

Four other inputs are present in block “D” of Figure 3.2. They all depend on the amount of ingested CHO. Information on ingested CHO provided by the patients is impulsive, however, CHO effects on glycemia are neither impulsive, nor instantaneous, nor constant over time. For this reason, to exploit at best the available meal information we preprocessed this input with a physiological model of oral glucose absorption (block labelled as “C” in Figure 3.2). In particular, we used the model proposed in [103], completed with the population parameters obtained in [104] (some details are reported in Appendix A.1). Precisely, the NN uses:

- the glucose rate of appearance, i.e. the output of the glucose absorption simulation model, predicted at time $t + N$, $Ra_G(t + N)$;

- three differences of the predicted rate of appearance of ingested CHO:

  1. $(1 - z^{-T_a})Ra_G(t + N)$,
  2. $(z^{-T_a} - z^{-2T_a})Ra_G(t + N)$,
  3. $(z^{-2T_a} - z^{-PH})Ra_G(t + N)$.

In our implementation on the data later described in Section 3.4, we will consider T_a=$N/3$=10 steps (i.e. 10 min). This value of 10 steps was chosen because it captures adequately the future dynamics of Ra_G in the time interval $[t,\, t + N]$. Anyway, it should be re-adjusted if different PHs or different sampling rates were considered.

The above network structure and inputs were determined, using the Matlab R2010a Neural Networks Toolbox [91], exploiting a priori physiological knowledge and through a 10-fold-cross-validation strategy applied on the training set.

Remark: to correctly compute the future rate of appearance of ingested CHO, the patient should announce the meal PH minutes in advance. However, in the absence of meal announcement, the effect of ingested CHO could be computed retroactively when the meal occurs, the only observed effect being a limited loss of prediction accuracy during the PH minutes preceding the meal.

3.2.2 Mathematical representation of the NN model

Predicted glucose concentration is obtained as

$$\hat{y}(t + N|t) = \hat{y}_P(t + N|t) + \hat{e}(t + N|t) \tag{3.1}$$

In particular, the first term in the right side of (3.1) is

$$\hat{y}_P(t + N|t) = \theta_1\, N + \theta_0 \tag{3.2}$$

where the parameters θ_0 and θ_1 are updated at each time step, (using a forgetting factor μ chosen in (0,1)), by the equations

$$\theta_0 = y(t) \tag{3.3}$$

$$\theta_1 = \arg\min_{\theta_1} \frac{1}{2} \sum_{i=1}^{t} \mu^{t-i} (\tilde{y}(i) - \theta_1(i-t))^2 \tag{3.4}$$

with

$$\tilde{\mathbf{y}} = \mathbf{y} - y(t) \tag{3.5}$$

For what concerns the NN prediction

$$\begin{aligned} \hat{e}(t+N|t) &= \mathbf{\Psi} \cdot \Phi(\mathbf{\Gamma} \cdot \mathbf{X}(t)) && (3.6) \\ &= \psi_0 + \sum_{j=1}^{N_{hn}} \psi_j \varphi \left(\sum_{i=0}^{N_{in}} \lambda_{ji} x_i(t) \right) && (3.7) \end{aligned}$$

where $\mathbf{X}(t)$ indicates the $[N_{in}+1]$ column vector of N_{in} input signals plus the input equal to 1 associated with the weights representing the bias terms, i.e.

$$\begin{aligned} \mathbf{X}(t) = \big[&1,\ y(t),\ (1 - z^{-T_m})y(t),\ e(t),\ (1 - z^{-T_m})e(t),\ Ra_G(t+N), \ldots \\ &\ldots (1 - z^{-T_a})Ra_G(t+N),\ (z^{-T_a} - z^{-2T_a})Ra_G(t+N), (z^{-2T_a} - z^{-N})Ra_G(t+N)\big]^T \end{aligned} \tag{3.8}$$

where the inputs correspond to the signals described in Section 3.2.1. $\mathbf{\Psi}$ represents the $[N_{hn}+1]$ row vector of weights connecting the L hidden neurons to the output neuron, including also the bias term, ($\Psi(k) = \psi_k$ is the weight connecting the k^{th} hidden neuron to the output). $\mathbf{\Gamma}$ is the $[N_{hn}$ x $N_{in}+1]$ matrix of weights connecting inputs and hidden neurons ($\Gamma(ji) = \gamma_{ji}$ represents the weight connecting the i^{th} input to the j^{th} hidden neuron). Φ is the tangent-sigmoid function, computed element-wise on the values of the vector $\mathbf{\Gamma} \cdot \mathbf{X}(t)$. By substituting (3.2), (3.3), (3.4) and (3.7) into (3.1) we obtain

$$\begin{aligned} \hat{y}(t+N|t) = &\left(\arg\min_{\theta_1} \frac{1}{2} \sum_{i=1}^{t} \mu^{t-i} ((y(i) - y(t)) - \theta_1(i-t))^2 \right) N + y(t) + \cdots \\ &\cdots + \psi_0 + \sum_{j=1}^{N_{hn}} \psi_j \varphi \left(\sum_{i=0}^{N_{in}} \lambda_{ji} x_i(t) \right) \end{aligned} \tag{3.9}$$

which is the explicit formula of the prediction schematized in Figure 3.2.

3.3 NN training

3.3.1 Inputs and output preprocessing

The NN inputs and output were scaled, so that, at the beginning of the training procedure, all the signals had potentially the same weight, and they all belonged to the linear range of the tangent sigmoid activation function of the neurons of the hidden layer.

In particular, $e(t)$ and its difference $(1 - z^{-T_m})e(t)$, $y(t)$ and its difference $(1 - z^{-T_m})y(t)$, and the target $e(t + N)$ were mapped so that they had zero mean and standard deviation equal to 0.5.

The signal $Ra_G(t + N)$ was scaled in the range [0, 3] and its differences were mapped so that they had zero mean and standard deviation equal to 0.25.

The rationale was obtaining mapped values mainly distributed in the range $[-1, 1]$, apart from $Ra_G(t + N)$, whose mapped values were mainly concentrated in [0, 1], (in fact Ra_G is a non-negative biological signal whose mean value is close to 0 and whose statistical distribution is not symmetric).

3.3.2 Structure and weights optimization

The number of hidden neurons was chosen with 10-fold-cross-validation on the training set and results equal to 8, thus, since the NN has 8 inputs, the number of free parameters to be optimized during training is equal to 81. Network weights were randomly initialized and optimized through a backpropagation Levenberg-Marquardt training algorithm, applied in a batch mode. The training procedure was stopped, using cross-validation, after 100 consecutive worsenings of the NN performance on the validation set, to avoid overfitting.

3.4 Test-bed

3.4.1 Simulated data

Twenty virtual patients were extracted from the UVA/Padova T1D Simulator [73, 84]. For each subject the simulation scenario consisted of 11 consecutive days of monitoring, with 3 meals per day. Breakfast was randomly located in the time interval 06:00-08:00 h, and consisted of 45+u g of CHO, where u is a random variable drawn from the uniform distribution $u \sim U(-10,\ 10)$g which is used to have more realistic simulations and to account for variability in CHO intake. Lunch was randomly located in the time interval 12:00-14:00 h, and consisted of 75+u g of CHO; finally, dinner was randomly located in the time interval 19:00-21:00 h, and consisted of 85+u g of CHO, with u defined as specified above. In order to obtain a significant number of hypo and hyperglycemic events, in 50% of cases the nominal insulin dosage in correspondence to meals was randomly modified by adding a quantity sampled from a uniform distribution between -3 and +3 U. Finally, realistic CGM time series were obtained by adding a noise sequence generated by an AR first order model (with pole in 0.95) driven by white Gaussian noise with zero mean and variance equal to 2. Such a noise sequence proved more realistic than that obtained with the noise model embedded in the simulator, which had already been demonstrated to be suboptimal [105].

Each of the 20 simulated profiles was divided in three subseries of 3 days, obtaining 60 CGM profiles, that were randomly divided into a training and validation set (40 profiles) and a test set (20 series). 70% of the data in the training set was used to optimize the NN weights' values, while the remaining 30% of the data was used to stop the training algorithm by cross-validation (see Subsection 3.3.2). Profiles in the test set did not take part in the NN architecture optimization, neither in its training nor validation.

3.4.2 Real data

The real data available when we implemented this algorithm were those collected during the first year of the DIAdvisor project [85], during the DAQ trial (see Appendix B for details). 15 T1D patients were monitored for 7 consecutive days with the FreeStyle Navigator®CGM system, that returns a glucose value every minute. The patients were not hospitalized, and neither insulin nor meal were programmed and thus assumed not to have fixed schedule. Since the NN requires, among its inputs, the estimation of the future Ra_G, PH minutes in advance, it is assumed that patients announce meal information (i.e. time and amount of CHO) at least PH minutes ahead in time.

The NN was trained and validated on 25 time series, each one of 3 days, selected so as to ensure a wide variety of glycemic dynamics. Nine daily profiles, containing several hypo and hyperglycemic events, were used to test the NN.

3.5 Results

The proposed NN was assessed in term of RMSE, TG and ESOD$_{\text{norm}}$ (see details in Appendix C) and compared with the NN developed by Pérez Gandía et al. [66] and the AR(1) model of Sparacino et al. [59], implemented on the same dataset.

3.5.1 Simulated data

NN-LPA and the NN of Pérez Gandía et al. [66], (hereafter referred as NNPG), were both trained on the training set and tested on the independent test set. Consistently, the forgetting factor of the AR(1) model was tuned on the training set, and the predictor performance was assessed on the test set. Data did not undergo any preprocessing (e.g. digital filtering).

An example of the application of the three prediction algorithms with PH = 30 min is displayed in Figure 3.3. A short time window (06:30-16:00 h) is reported, in order to allow the reader to better capture differences among the behavior of the three predictions. Anyway we would like to stress that numerical results were calculated on the whole 3-days test series. The figure compares the prediction obtained with NN-LPA (orange line) with those of NNPG (cyan line) and AR(1) (gray line). The target signal, i.e. the original CGM recording, is also reported (black dotted line). It is worth clarifying that the predicted signal is plotted at the time instant to which it refers, i.e. the value plotted at a certain time is obtained N time steps earlier, using only data available until N time steps earlier.

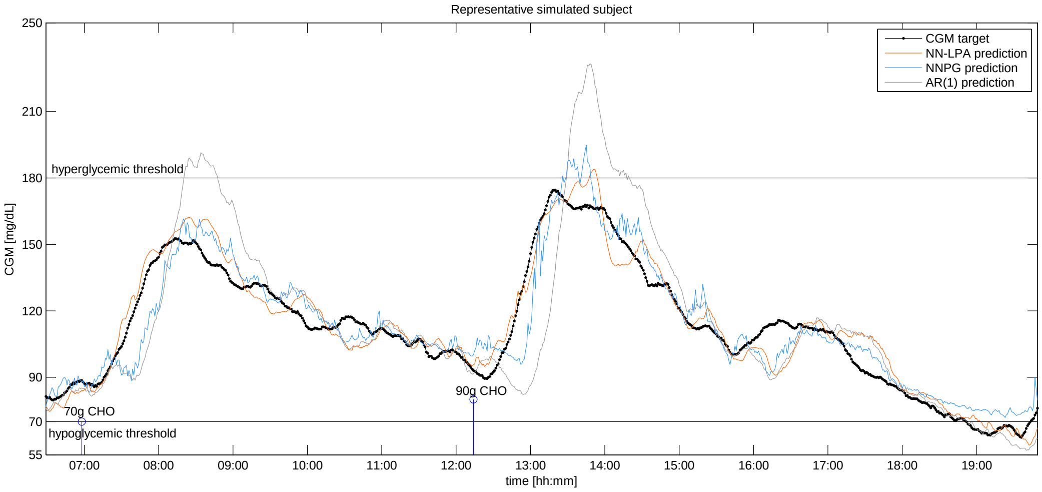


Figure 3.3: A synthetic CGM profile (black dotted line), and the predictions (PH=30 min) obtained with NN-LPA (orange line), NNPG (cyan line), and AR(1) (gray line). CHO ingestion is evidenced by blue stems.

As seen by inspection, NN-LPA performs better than NNPG and AR(1). The prediction obtained by NN-LPA is more adherent to the target profile than NNPG and AR(1), as confirmed by the lower RMSE equal to 9.0 mg/dL for NN-LPA and 11.1 mg/dL and 20.45 mg/dL for NNPG and AR(1), respectively. Furthermore, prediction obtained with NN-LPA has less spurious oscillations than NNPG prediction, as confirmed by

ESOD$_{\text{norm}}$ equal to 2.12 for NN-LPA, 3.5 for AR(1) and 37.9 for NNPG. The most significant improvement can be found after CHO ingestion, i.e. when the performance of NNPG was already observed to be suboptimal (see Figure 3.1). Indeed, in these intervals NN-LPA detects the changes in the sign of the CGM derivative more quickly. This is confirmed also by the higher TG, equal to 27.0 min for NN-LPA, 17.0 min for NNPG and 21.0 min for AR(1). Performance obtained for the other subjects are similar.

Table 3.1 reports a summary of the average results obtained by the three prediction algorithms on all the 20 simulated CGM time series of the test set, and p-values returned by the non-parametric Mann-Whitney U test[1] [106].

Table 3.1: Summary of performance indexes (Mean ± SD) on the 20 simulated datasets (with PH=30min). Asterisk (*) indicates statistically significant difference at the 5% confidence level. p-values are also reported. The lower the RMSE, the higher the TG, the closer to 1 ESOD$_{\text{norm}}$ the better the quality of the predicted profiles.

	NN-LPA	**NNPG**	**AR(1)**
RMSE [mg/dL]	9.4 ± 1.5	$10.7 \pm 1.9*$	$17.5 \pm 6.4*$
p-value		0.0275	$1.6 \cdot 10^{-5}$
TG [min]	24.9 ± 4.4	$16.5 \pm 3.6*$	$21.5 \pm 2.9*$
p-value		0	0.0156
ESOD$_{\text{norm}}$ [-]	1.9 ± 0.2	$39.3 \pm 4.7*$	$3.39 \pm 0.2*$
p-value		$6.8 \cdot 10^{-8}$	$6.8 \cdot 10^{-8}$

The RMSE is satisfactory for both NNs, and significantly lower than for AR(1). Moreover NN-LPA is slightly but significantly more accurate than NNPG in predicting the future glycemia, with a PH of 30 min. As far as TG is concerned, NN-LPA ensures almost 25 minutes of net anticipation. This would be a major improvement over NNPG (+8.3 min greater), and over AR(1) (+4.5 min greater) since such a large margin of time would allow patients to take more effective countermeasures to e.g. avoid (or at least mitigate the effect of) dangerous hypoglycemic events. ESOD$_{\text{norm}}$ is significantly lower for NN-LPA (1.9) than for NNPG (39.3), and for AR(1) (3.4) indicating that NN-LPA predicted profiles exhibit fewer spurious oscillations. From a patient perspective, the smoothness of the predicted time series is crucial, since oscillations can facilitate the generation of false hypo and hyper-alerts, lowering the predictor reliability. Remarkably,

[1] The Mann-Whitney U test is a statistic non-parametric test of the null hypothesis that two populations are the same against an alternative hypothesis. It has greater efficiency than the t-test on non-normal distributions, and it is nearly as efficient as the t-test on normal distributions.

NN-LPA significantly outperforms AR(1), in addition, even though the RMSE appears similar for NN-LPA and for NNPG, the profiles predicted by NN-LPA are definitely more “usable”, than the time series predicted by NNPG, as confirmed by the other indexes.

The non-parametric Mann-Whitney U test confirms that all the differences observed between the numeric values of the indexes are significant (see p-values in Table 3.1).

3.5.1.1 Robustness to errors in meal information

A robustness analysis to assess the impact of errors in meal timing and CHO size estimates was also performed. Two major scenarios, each one with four different subcases, were created. In the first, all meal timings were anticipated or delayed by -10, -5, +5, and +10 minutes, respectively. In the second, errors of -20%, -10%, +10%, and +20% on all meal sizes were introduced. Note that all these scenarios correspond to a worst-case evaluation of NN-LPA behavior in the presence of inaccurate meal data, since, in each subcase, all meals were shifted/ wrongly estimated by the considered time/ amount. Average results are reported in Table 3.2, where p-values refer to the comparison to the reference case, (no errors on meal information). As we can observe, NN-LPA is robust on both errors. In fact, all indexes do not significantly change from the reference results, except RMSE when meal timing is delayed by 10 min, TG when a 20% reduction of CHO amount is applied, and $\text{ESOD}_{\text{norm}}$ in the 20% CHO amount increase scenario. The Mann-Whitney U test confirms that results obtained with slightly inaccurate meal data are, in the majority of cases, not statistically different from those obtained with perfect meal data.

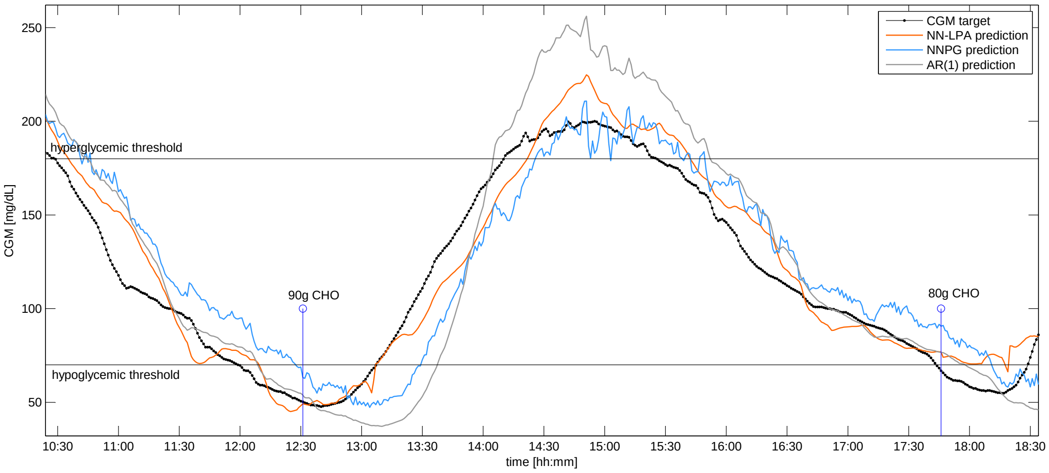
3.5.2 Real data

Figure 3.4 shows the result of the application of the three prediction algorithms to two different real CGM (Abbott FreeStyle Navigator$^{®}$) profiles. To better appreciate differences among predictions the displayed interval is restricted to a 8 h time window. As for simulated data, numerical results refer to the whole 1 day time series, and we chose to plot time windows containing at least one meal, to better show the advantage given by NN-LPA over reference methods. In fact, glucose dynamics are faster and more difficult to predict after a meal than during the night, when glucose concentration behavior is more regular because of the absence of exogenous inputs.

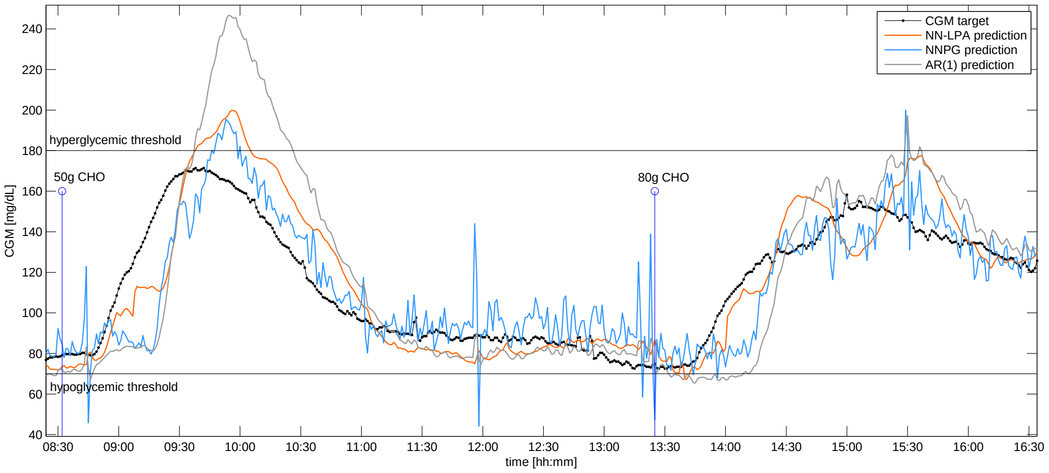
As already observed in simulation, NN-LPA outperforms NNPG and AR(1). In particular, NN-LPA detects changes in the sign of CGM derivative more quickly. This results evident in Figure 3.4(a) around 13:00 h and 18:00 h and in Figure 3.4(b) around 09:00 h and 14:00 h. The greater “usability” of the forecasted profile obtained by

Table 3.2: Summary of performance indexes (Mean ± SD) on all 20 simulated datasets (with PH=30min), for the two scenarios tested by the robustness analysis. Asterisk (*) indicates statistically different results at the 5% confidence level.

	REFERENCE CASE	**SCENARIO 1: error in meal timing**				**SCENARIO 2: error in CHO size estimates**			
		-10 min	-5 min	+5 min	+10 min	-20%	-10%	+10%	+20%
RMSE [mg/dL]	9.4 ± 1.5	10.7 ± 2.1	9.7 ± 1.7	9.9 ± 1.6	$^{*}11.2 \pm 2.1$	9.4 ± 1.9	9.3 ± 1.7	9.5 ± 1.3	9.6 ± 1.3
p-value		0.0531	0.5792	0.1895	0.006	0.675	0.7557	0.5792	0.4903
TG [min]	24.9 ± 4.4	23.3 ± 6.3	24.6 ± 5.1	24.0 ± 4.0	22.4 ± 3.6	$^{*}21.8 \pm 4.2$	23.6 ± 4.4	25.6 ± 4.1	26.1 ± 3.9
p-value		0.6133	0.9672	0.4224	0.0749	0.0431	0.2891	0.5198	0.2762
ESOD$_{\mathbf{norm}}$ [-]	1.9 ± 0.2	2.0 ± 0.2	2.0 ± 0.2	2.0 ± 0.2	2.0 ± 0.2	1.8 ± 0.1	1.9 ± 0.2	2.1 ± 0.3	$^{*}2.3 \pm 0.5$
p-value		0.5428	0.818	0.409	0.12	0.0531	0.229	0.209	0.0084



(a) Subject 1.



(b) Subject 2.

Figure 3.4: Two representative real CGM profiles (black dotted line), and the predictions (PH=30 min) obtained with NN-LPA (orange line), NNPG (cyan line) and AR(1) (gray line). CHO ingestion is evidenced by blue stems.

NN-LPA, with respect to the time series predicted by NNPG, can be appreciated in Figure 3.4(b), where due to the noise affecting the CGM values measured by the sensor, NNPG predictions exhibit non-physiological oscillations, and, occasionally, cross the hypo and hyperglycemic thresholds, even when the true glucose stays in the euglycemic range, potentially generating three false hypo-alerts at 12:00 h, 13:20 h, and 13:25 h. Regarding quantitative indexes relative to Figure 3.4(a):

- RMSE is 20.7 mg/dL for NN-LPA, 23.5 mg/dL for NNPG and 31.9 mg/dL for

AR(1);

- TG is 16.0 min for NN-LPA and 13.0 min for both NNPG and AR(1);
- $\text{ESOD}_{\text{norm}}$ is 4.4 for NN-LPA, 62.6 for NNPG and 5.5 for AR(1).

Regarding quantitative indexes relative to Figure 3.4(b):

- RMSE is 13.5 mg/dL for NN-LPA, 15.5 mg/dL for NNPG and 23.8 mg/dL for AR(1);
- TG is 16.0 min for NN-LPA, 13.0 min for NNPG and 17.0 min for AR(1);
- $\text{ESOD}_{\text{norm}}$ is 1.0 for NN-LPA, 99.9 for NNPG and 3.2 for AR(1).

Table 3.3 reports the average results for the indexes obtained in the 9 real CGM test series, and the p-values obtained with the non-parametric Mann-Whitney U test.

Table 3.3: Summary of performance indexes (Mean ± SD) on the 9 real datasets (with PH=30min). Asterisk (*) indicates statistically significant difference.

	NN-LPA	**NNPG**	**AR(1)**
RMSE [mg/dL]	14.0 ± 4.1	14.2 ± 4.5	19.6 ± 7.2*
p-value		1	0.0625
TG [min]	16.2 ± 3.7	$12.8 \pm 1.6*$	16.7 ± 4.2
p-value		0.0153	0.776
$\text{ESOD}_{\text{norm}}$ [-]	2.7 ± 1.6	$105.3 \pm 52.8*$	3.9 ± 0.8
p-value		$4.11 \cdot 10^{-5}$	0.077

In accordance with what is observed on the simulated data, the RMSE is almost identical for the two NNs, and better than for AR(1), indicating that the accuracy of the predictions is comparable or improved. The TG achieved by NN-LPA is better than the one obtained with NNPG (+3.5 min), and is comparable with the TG of AR(1). It is worth noting that a TG of 16 min is sufficient to mitigate the effects of a hypo or hyperglycemic event, increasing the utility of the proposed prediction algorithm in a therapeutic perspective. As far as $\text{ESOD}_{\text{norm}}$ is concerned, the value obtained by NN-LPA is markedly lower than the value achieved by NNPG, and slightly lower than the value obtained by AR(1). This means that NN-LPA forecasts are much more regular than those of NNPG, without spikes and with far fewer spurious oscillations, possibly leading to false crossings of the euglycemic thresholds.

The results obtained on real test data support quantitatively what already observed on simulated data. Not only NN-LPA predicts the future glycemia with a high accuracy, especially during and after meals, but it also achieves a TG large enough to mitigate, or even totally avoid, future glucose excursions out of the euglycemic range. An exhaustive quantification of the potential reduction of hypoglycemia that could be obtained using NN-LPA's predicted profiles is reported in Chapter 7.

3.6 Conclusions and margins for further improvement

NN-LPA combines a NN model with a first-order polynomial predictor and uses them in parallel to forecast, respectively, the nonlinear and linear components of glucose dynamics. In this way the prediction algorithm takes advantage of the ability of poly(1) to predict linear components of glucose dynamics and of the ability of NN to track the nonlinear components (e.g. after meals). The NN also uses available information on CHO intake, preprocessed with a literature physiological model [103, 104].

Results both on 20 simulated and 9 FreeStyle Navigator®real datasets demonstrated that this approach improves on the recent NNPG approach [66], and over the AR(1) model proposed in [59]. In particular, on simulated data the RMSE achieved by NN-LPA is significantly lower than NNPG, (-2.09 mg/dL), and than AR(1) (-8.1 mg/dL) with a simultaneous increase of the average TG (+10 min and +3.5 min with respect to NNPG and AR(1) respectively). Similar results were obtained also on the 9 real datasets. Thanks to the information on CHO intake, the proposed algorithm is highly accurate especially during and after meals. In particular, it is faster than the reference NNPG and AR(1) methods in detecting upward changes in the glycemic trend due to CHO ingestion. These results have been published in [102].

The proposed method is totally causal and, once trained, the parameters of the NN are fixed and its real-time implementation is computationally light; however, the poly(1) model is time-varying, thus its parameter should be re-optimized at each time step using RLS and past CGM values, weighted by a forgetting factor μ. The algorithm is almost completely autonomous in computing its inputs, the only burden for the patient would be to provide information on CHO content of meals PH minutes in advance. This requirement might be an issue, since in real life it is not always possible to estimate roughly, with an anticipation of 30 min, the amount of CHO of the future meal.

These limitations, i.e. the necessity of updating the parameters of the linear model at each time step and the requirement of announcing the meal at least 30 min in advance will be overcome by the NN predictor presented in Chapter 4.

4

Further development of glucose prediction methods by jump NN

4.1 Rationale

In this chapter we will describe a novel NN architecture we proposed to overcome the three major limitations of NN-LPA, i.e.:

1. the need of a time-varying linear model in parallel with the NN structure;
2. the necessity of announcing the meal PH min in advance;
3. the fact that the NN models the nonlinear relation between past and future CGM and both, linear and nonlinear effects of glucose rate of appearance in the blood, on future glucose concentration.

As pointed out in Subsection 2.2.3 of this thesis, jump NNs are particularly suitable for fitting and predicting time series characterized by the presence of both linear and nonlinear dynamics. This is the case of glucose signals, where we may assume that each input is able to influence future blood glucose concentration with both linear and nonlinear effects.

Thus our aim is to implement a predictor based on a jump NN and evaluate if it is able to obtain results comparable to, or better of, those of NN-LPA, despite the simpler structure and without announcing the meal ahead in time [107].

4.2 Architecture of the Jump NN

The architecture of the proposed jump NN is schematized in Figure 4.1.

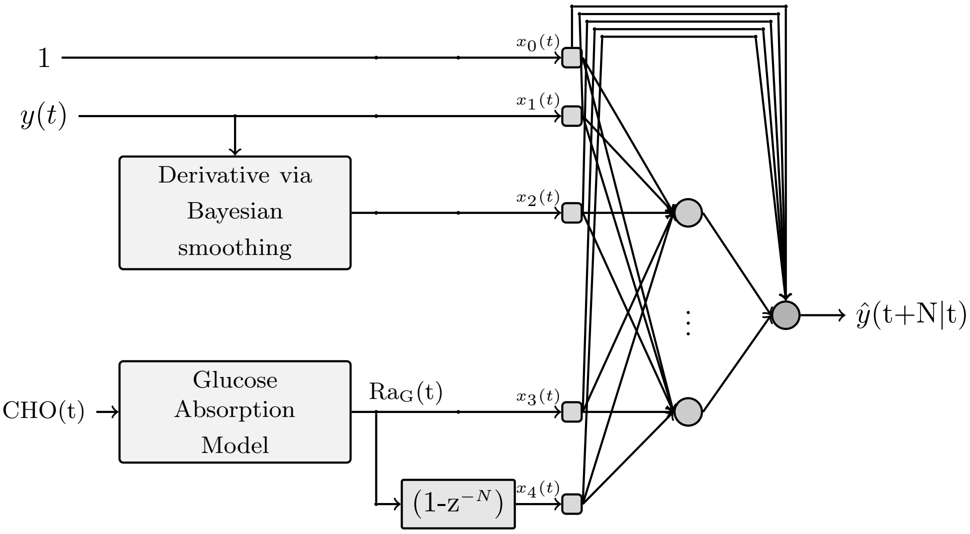


Figure 4.1: Block scheme of the proposed jump NN model for glucose prediction.

As we can note, the main differences with respect to Figure 3.2 are the absence of the polynomial model and the presence of weights connecting each input also to the target. Inputs and output are similar to those of NN-LPA: the NN output is the future CGM signal, with a PH of 30 min, and the 4 NN inputs are the glycemic concentration value, currently measured by the CGM sensor, the current trend of the CGM signal, the glucose rate of appearance, simulated with the model of [103], using as parameters the population values estimated in [104] and its derivative. To face ill-conditioning, the trend of the CGM signal is computed using a Bayesian smoothing approach [108]. For what concerns meal-related information, no meal announcement is required. Indeed, a cross-correlation analysis between the estimated Ra_G obtained with [103] and glucose concentration, considering various time shifts, confirmed that correlation between the current Ra_G and glucose concentration observed 30 min in the future assumes a value comparable to that of the correlation between future Ra_G and future glycemia.

The jump NN predicts a signal that may be expressed as:

$$\hat{y}(t+N|t) = \mathbf{\Omega} \cdot \mathbf{X}(t) + \mathbf{\Psi} \cdot \Phi(\mathbf{\Gamma} \cdot \mathbf{X}(t)) \tag{4.1}$$

$$= \sum_{i=0}^{N_{in}} \omega_i x_i(t) + \sum_{j=1}^{N_{hn}} \psi_j \varphi \left(\sum_{i=0}^{N_{in}} \lambda_{ji} x_i(t) \right) \tag{4.2}$$

where $\mathbf{X}(t)$ indicates the $[N_{in}+1]$ column vector of N_{in} input signals at time step t, plus an input equal to 1 associated with the weights accounting for the bias terms,

$$\mathbf{X}(t) = \left[1,\ y(t),\ \Delta_{BS} y(t),\ Ra_G(t),\ (1-z^{-N})Ra_G(t)\right]^T, \tag{4.3}$$

$\mathbf{\Omega}$ is the $[N_{in}+1]$ row vector of weights connecting every input directly to the output neuron, thus $\Omega(i) = \omega_i$ indicates the weight connecting the i^{th} input to the output neuron. Δ_{BS} indicate the Bayesian smoothing paradigm adopted for computing the signal derivative. $\mathbf{\Psi}$ is the $[N_{hn}]$ row vector of weights connecting every hidden neuron to the output neuron, thus $\Psi(j) = \psi_j$ is the weight connecting the j^{th} hidden neuron to the output neuron. $\mathbf{\Gamma}$ is the $[N_{hn}$ x $N_{in}]$ matrix of weights connecting every input to every hidden neuron, thus $\Gamma(ji) = \gamma_{ji}$ indicates the weight connecting the i^{th} input to the j^{th} hidden neuron. Notably, the second addendum of the right hand side of equation (4.2) coincides with the last term of equation (3.9), (apart from differences in some input signals of the NN). The first addendum of equation (4.2) can be developed as

$$\mathbf{\Omega} \cdot \mathbf{X}(t) = \sum_{i=0}^{N_{in}} \omega_i x_i(t) =$$

$$= \omega_0 + \omega_1 y(t) + \omega_2 \left(\Delta_{BS} y(t)\right) + \omega_3 Ra_G(t) + \omega_4 \left((1-z^{-N})Ra_G(t)\right) \tag{4.4}$$

By comparing equation (4.4) and the first two terms of equation (3.9) we note that the term $\omega_1 y(t) + \omega_2 \left(\Delta_{BS} y(t)\right)$ gives a contribution similar to that of the polynomial model (i.e. $\theta_1^N N + y(t)$). Moreover, the term $\omega_3 Ra_G(t) + \omega_4 \left((1-z^{-N})Ra_G(t)\right)$ accounting for the linear effect of ingested CHO on future glycemia has no counterpart in (3.9). This comparison confirms that the absence of the polynomial model is fulfilled by the weights directly connecting $y(t)$ and $\Delta_{BS} y(t)$ to the output and that, in addition, the jump NN also takes into account the linear effects of CHO on future glycemia.

4.3 Jump NN training

4.3.1 Inputs and output preprocessing

Before training the jump NN, inputs and outputs were normalized, so that they had zero mean and standard deviation equal to 1 so that, at the beginning of the training procedure, all the signals had, potentially, the same importance and they all belonged to the input range in which the tangent sigmoidal activation functions have a linear behavior.

4.3.2 Structure and weights optimization

The jump NN structure was optimized and trained following a procedure analogous to that used for NN-LPA (see Chapter 3, Subsection 3.3.2). The number of hidden neurons was chosen with 10-fold-cross-validation on the training set and is equal to 5 neurons, with tangent sigmoidal activation function. Thus, since the number of input signals is equal to 4, plus the bias-related term, there are 35 free parameters to tune during training. This is considerably lower than the number of weights of NN-LPA, which, after having re-optimized its structure on data sampled every 5 min, is equal to 65 (see *Remark* in Section 4.4 for details).

The jump NN parameters were randomly initialized and optimized with the back-propagation Levenberg-Marquardt training algorithm, applied in batch mode. The training procedure was stopped using cross-validation, after 100 consecutive worsening of the performance of the algorithm on the validation set. The above NN structure was optimized with the Matlab 2011b Neural Network toolbox [91].

4.4 Test-bed

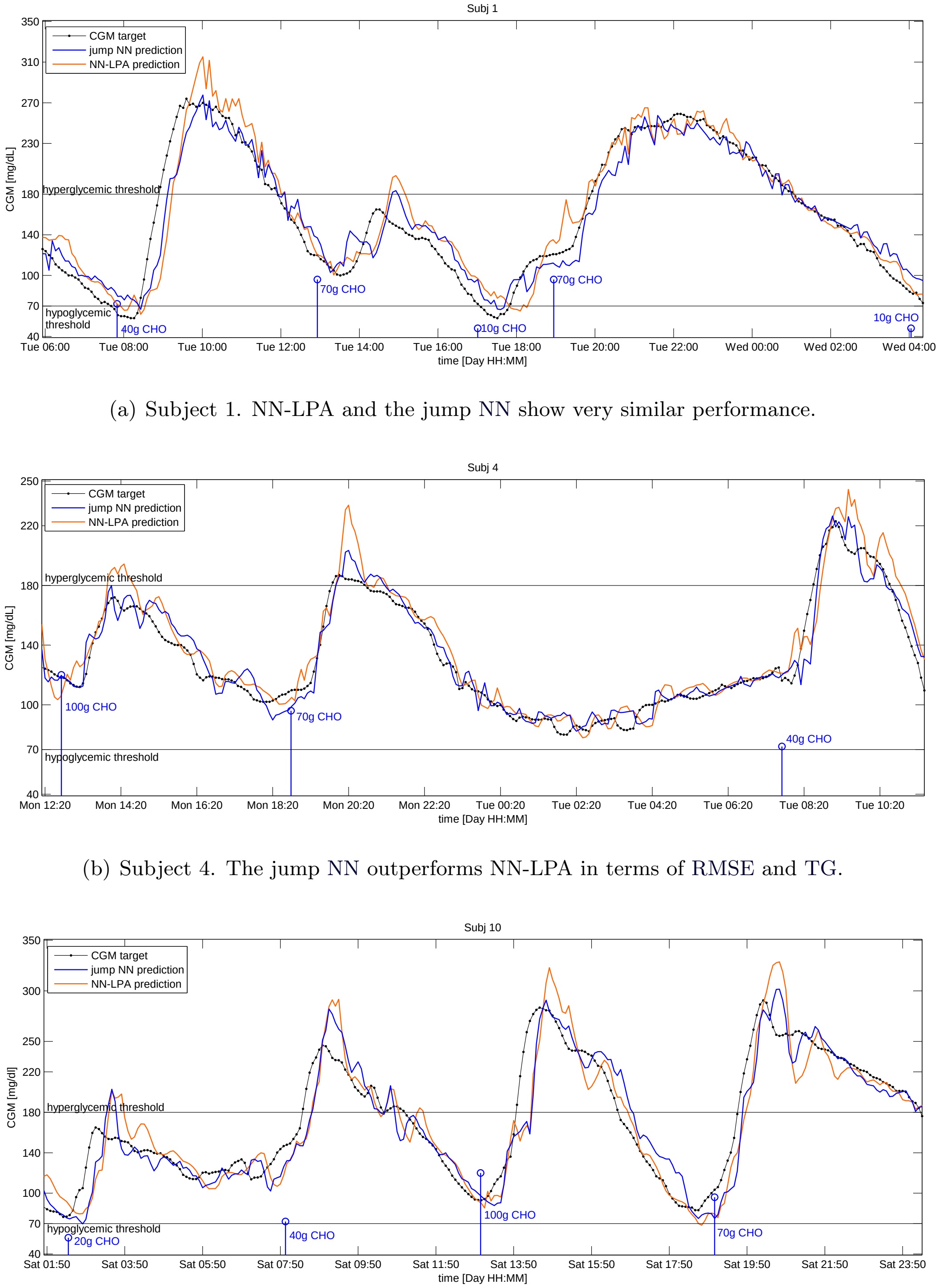
Data collected during the second session of the DIAdvisor [85] project were used to optimize and train the jump NN. In particular data of 20 type 1 diabetic patients, monitored for 2 or 3 consecutive days in real-life conditions were considered. CGM was measured by the Dexcom SEVEN PLUS CGM sensor, which has a sampling time of 5 min. We chose to implement the algorithm on the new available data, instead of using the same database of NN-LPA, because SEVEN PLUS data were less noisy and information on meals and insulin therapy was more reliable. Furthermore, developing the algorithm on data of a widely used commercial device, as the SEVEN PLUS CGM sensor, characterized by a sampling period of 5 min, as the vast majority of CGM devices, rendered results potentially more appealing in light of possible effective implementation in commercial sensors.

The database was divided into a *training and validation set*, constituted by 10 time series and a *test set*, formed by the other 10 time series. During training, the training and validation set was further randomly divided into the training set constituted by 70% of the data, used for minimizing the prediction error and the validation set constituted by the remaining 30% of data, used for stopping the training procedure.

Remark: The NN-LPA of the previous chapter was developed not on SEVEN PLUS data, (sampling time 5 min), but on FreeStyle Navigator®data (sampling time 1 min), which was the CGM sensor used during the DAQ trial of the DIAdvisor project. For a fair comparison, NN-LPA inputs and structure were re-optimized on the new dataset. The updated NN-LPA has 6 inputs instead of 8, since only one derivative of Ra_G was selected (at time t the selected signal was $(1 - z^{-N})Ra_G(t + N)$), while the number of hidden neurons was unchanged, thus equal to 8, leading to 65 free parameters.

4.5 Results

The 3 panels of Figure 4.2 show approximately one day of monitoring of 3 representative subjects. The black dotted line is the target signal, measured by the sensor, the orange line is the prediction obtained with NN-LPA, described in Chapter 3, used as reference, and the blue line is the signal predicted by the jump NN. The two horizontal thin lines correspond to the hypo- and the hyper-glycemic threshold, while the stems represent CHO ingestion. Figure 4.2(a) shows a representative example where the performance of the jump NN and of NN-LPA are very similar, as confirmed also by numerical results reported in the first row of Table 4.1. Indeed, even from visual inspection, we can note that both predictions are close to target (RMSE=17.6 mg/dL for the jump NN, and RMSE=22.9 mg/dL for NN-LPA) and both NNs forecast with a minimum time lag changes in the trend and in the dynamics of the CGM signal (TG=15 min for both models). Furthermore, spurious oscillations are limited (ESOD$_{\text{norm}}$ equals 9.9 for the jump NN and 13.5 for NN-LPA). This suggests that the predicted profile could be useful for generating preventive alerts in case of impending hypo- and hyper-glycemia, with a low risk of generating false alerts. Figure 4.2(b) shows an example where the jump NN outperforms NN-LPA in terms of RMSE and TG (see fourth row of Table 4.1). The lower RMSE corresponds to the lower over-estimation of the target in correspondence of hyperglycemic picks, around time 14:20 h and 20:20 h, while for what concerns the TG, the jump NN predicts the downward trends from time 21:00 h to 00:00 h and from



(a) Subject 1. NN-LPA and the jump NN show very similar performance.

(b) Subject 4. The jump NN outperforms NN-LPA in terms of RMSE and TG.

(c) Subject 10. The jump NN has a TG slightly worse than NN-LPA.

Figure 4.2: CGM profile (black dotted line) and prediction obtained with NN-LPA (orange line) and with the new jump NN (blue line). Stems indicate CHO ingestion, horizontal thin lines represent the hypo- and the hyperglycemic threshold.

09:00 h to 10:30 h better than NN-LPA. Finally, Figure 4.2(c) shows a case where the jump NN has a TG slightly worse than NN-LPA, as confirmed by its visible greater delay in forecasting the signal downward trend in the time interval 16:00-18:00. Results obtained on the 10 time series are shown in Table 4.1, where also average results and the p-values obtained with the non-parametric Mann-Whitney U test are reported.

Table 4.1: Results obtained on the 10 test subjects (with PH=30 min), average (mean±sd) values and p-values computed with the non-parametric Mann-Whitney U test.

	RMSE [mg/dL]		**TG** [min]		$\textbf{ESOD}_{\textbf{norm}}$ [-]	
	Jump NN	**NN-LPA**	**Jump NN**	**NN-LPA**	**Jump NN**	**NN-LPA**
subj 1	17.6	22.9	15	15	9.9	13.5
subj 2	20.3	21.2	15	20	10.0	7.4
subj 3	13.1	15.9	20	20	8.7	7.1
subj 4	12.0	15.9	25	20	7.4	7.5
subj 5	15.6	21.2	20	20	8.5	8.5
subj 6	15.9	20.1	20	20	12.6	23.7
subj 7	13.7	15.1	20	20	9.4	10.0
subj 8	17.8	18.7	15	20	8.1	7.4
subj 9	18.2	22.8	20	20	12.0	9.9
subj 10	21.3	23.2	15	20	9.8	9.6
mean±sd	16.6±3.1	19.7±3.1	18.5 ± 3.4	19.5±1.6	9.6±1.6	10.5±5.0
p-value		p=0.08		p=0.35		p=0.38

Average results confirm what observed on the 3 subjects plotted on the 3 panels of Figure 4.2: predicted CGM profiles are close to the target time series measured by the CGM sensor, as we can infer from the RMSE that, in every subject, is lower for the jump NN than for NN-LPA. In addition, the jump NN predictions are characterized by a TG ranging from 15 min to 25 min, with an average value of 18.5 min. Furthermore, the presence of spurious oscillations, due to measurement noise, is limited, as confirmed by the low values of $\text{ESOD}_{\text{norm}}$ obtained in all subjects. p-values confirm that no statistically significant difference exists between the two NNs.

4.6 Conclusions and margins for further improvement

Results reported in Section 4.5 allow us to conclude that the jump NN predicts satisfactorily future glycemia, giving results statistically comparable to those of NN-LPA. It is worth stressing that the jump NN has a simpler structure, indeed once trained, it is time-invariant and, differently from NN-LPA, does not need a time-varying polynomial model in parallel with it. Remarkably, while the reduction of operations needed for predicting future glucose concentration is irrelevant in a personal computer, it can be of great impact if implemented in the chip of a CGM sensor, where computational power and memory are limited and shared between various simultaneous processes and algorithms. Moreover, the jump NN does not need meal announcement, since it uses information on quantity of ingested CHO until the current time instant, thus the subject simply has to enter this information at the same time of the meal. Differently, NN-LPA needs information on future ingestion of CHO, with an anticipation of PH min, thus the subject should announce the correct quantity of CHO he/she will ingest PH min in advance, which is often unlikely to be doable in every-day life conditions. These results have been published in [107].

A further improvement of the jump NN model will be the inclusion of information on insulin therapy, which we will investigate in Chapter 5.

5

Inclusion of insulin information

5.1 Rationale

As discussed in Chapter 4 the jump NN using information on past CGM and on timing and CHO content of meals resulted equivalent, in terms of performance, to NN-LPA, which is constituted by a feedforward NN in parallel with a time-varying first order polynomial model, whose parameters need to be re-adjusted at each time step and requires meal announcement PH minutes in advance. Given the simpler structure of the jump NN predictor, we decided to adopt this model and try to further improve its performance, by adding, to CGM and CHO related inputs, signals derived from information on timing and dose of insulin therapy [109, 110]. In particular, we analyzed PHs of 15, 30, 45 and 60 min and we compared the performance of four NN predictors using different input combinations:

1. NN CGM using CGM;
2. NN I using CGM and insulin (timing and dose of bolus);
3. NN M using CGM and meal (timing and CHO content);
4. NN I+M using CGM, insulin (timing and dose of bolus) and meal (timing and CHO content).

5.2 Architecture of the jump NN-based predictors

The structure of the chosen predictor is similar to that of the jump NN described in Section 4.2 of this thesis and is schematized in Figure 5.1.

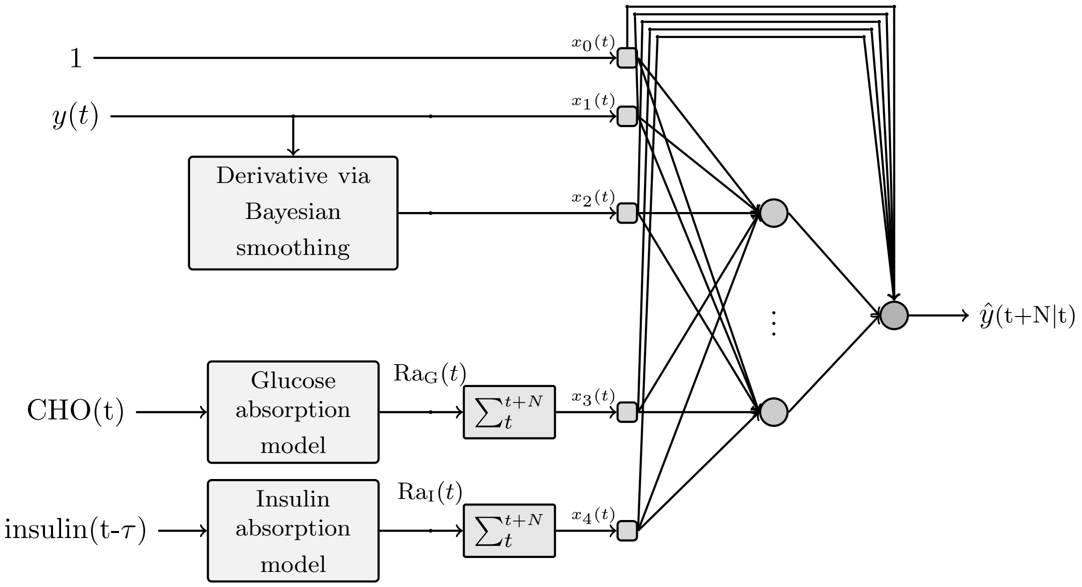


Figure 5.1: Block scheme of the jump NN prediction model.

For what concerns the mathematical representation of the predictor, it is analogous to equation (4.1), apart from the input vector $\mathbf{X}$. For easing the reader, we report the equations and the meaning of symbols. The predicted signal at time t is

$$\hat{y}(t+N|t) = \mathbf{\Omega X}(t)^T + \mathbf{\Psi \Phi}\left(\mathbf{\Gamma X}(t)\right) \tag{5.1}$$

where $\mathbf{X}(t)$ is the $[N_{in}+1]$-size column vector of inputs at time instant t, including an entry equal to 1 accounting for the bias term; $\mathbf{\Omega}$ is the row vector of length $N_{in}+1$ of weights connecting the inputs directly to the output neuron; $\mathbf{\Psi}$ is the row vector of size N_{hn} of weights connecting the hidden neurons to the output neuron; $\mathbf{\Gamma}$ is the matrix of size $[N_{hn} \times N_{in}+1]$ of weights connecting the inputs to the hidden neurons and $\mathbf{\Phi}$ is the hyperbolic tangent activation function of the hidden neurons, computed element-wise on the elements of the matrix $\mathbf{\Gamma X}(t)$; N_{in} is the number of inputs and N_{hn} indicates the number of hidden neurons. Thus, $\hat{y}(t+N|t)$, i.e. prediction obtained at time instant t, and relative to $t+N$, can be expressed, explicitly, as

$$\hat{y}(t+N|t) = \sum_{i=0}^{N_{in}} \omega_i x_i(t) + \sum_{j=1}^{N_{hn}} \psi_j \varphi \left(\sum_{i=0}^{N_{in}} \gamma_{ji} x_i(t) \right) \tag{5.2}$$

where $x_i(t)$ is the i^{th} input at time t; ω_i is the weight connecting the i^{th} input to the output neuron; ψ_j is the weight connecting the j^{th} hidden neuron to the output neuron; γ_{ji} is the weight connecting the i^{th} input to the j^{th} hidden neuron and $\varphi(\cdot)$ is the tangent hyperbolic activation function. The vector of inputs, for model 4, (i.e. NN I+M), is

$$\mathbf{X}(t) = \left[1,\, y(t),\, \Delta_{BS} y(t),\, \sum_{i=t}^{t+N} Ra_G(i),\, \sum_{i=t}^{t+N} Ra_I(i) \right]^T , \tag{5.3}$$

with Δ_{BS} indicating the Bayesian smoothing approach for computing glucose concentration first-order time derivative. Details on the procedure adopted for choosing the input signals are reported in the next section.

5.3 NN inputs

When dealing with insulin information, we had to face three major issues:

1. insulin information is impulsive, while insulin effects last several hours and are not constant over time;
2. insulin injection and CHO ingestion are almost always concomitant and proportional to each other, thus the signals are highly correlated;
3. insulin action is affected by physiological delays and inter- and intra-subject variability is high.

To cope with the first problem we adopted a solution analogous to that used for CHO information. Indeed insulin was preprocessed with a state-of-art physiological model [103], completed with population parameters estimated in [104] to generate *insulin rate of appearance (Ra_I)* in the blood. This signal is an estimate of the velocity with which insulin enters the blood stream after injection. Details are reported in Appendix A.2

Since insulin injection and CHO ingestion are usually concomitant and proportional to each other, Ra_I and Ra_G signals are highly correlated. Thus, to solve issues 2 and 3 we delayed the input related to insulin of 60 min, in line with results obtained in [111], where the average physiological delays in insulin action was estimated to be equal to 60 min.

It is worth noting that we used only information relative to insulin bolus therapy (for patients using insulin pumps) or to fast-acting insulin bolus therapy (for patients using fast and slow insulin). The rationale for discarding information on basal or slow insulin is that those inputs have slow effects, quasi constant over the whole day, thus they do not relevantly affect glucose dynamics during the PHs we considered in our analysis.

For choosing the effective NN inputs, we adopted a mixed strategy based on a priori physiological knowledge, correlation analysis and 10-fold-cross-validation results.

For what concerns the input signals relative to CGM history, in line with the jump NN described in Chapter 4 and the NN-LPA described in Chapter 3, we used the current glucose concentration, measured by the sensor and its first-order time derivative, computed using a Bayesian smoothing approach [112]. Parameters of the Bayesian filter were fixed to render it computationally light and potentially implementable in real time, even on a CGM device.

For what concerns meal and insulin related inputs, we considered various signals related to Ra_G and Ra_I, e.g. their past, current and future (predicted using only current information) values, their first-order time derivatives and their cumulative sum calculated on a sliding window. We computed the correlation between these signals and the target glucose concentration, for every PH we wanted to predict (i.e. 15, 30, 45 and 60 min) and we choose the signals whose correlation with future glucose concentration was higher, possibly for all the PHs. Two signals relative to meal and two signals relative to insulin had a pretty high correlation with future glucose, for every PHs: the current rate of appearance and the cumulative amount of insulin/ glucose, computed summing, respectively, the values of Ra_I and Ra_G between the current and the predicted time instant. However, a 10-fold-cross-validation analysis on the training set showed that if both the inputs relative to CHO and both the inputs relative to insulin were used, the NN converged prematurely and had poor performance. The best results were obtained when the cumulative amount of insulin and glucose were used as inputs.

5.4 NN training

Each NN structure was optimized, for each PH, via 10-fold-cross-validation on the training set. All the NNs have a single hidden layer with a number of neurons ranging from 4 to 5 and one output neuron.

Before training the NN, inputs and output were normalized, so that they had zero mean and standard deviation equal to 1. Network parameters were randomly initialized and optimized through a backpropagation Levenberg-Marquardt training algorithm, applied in a batch mode. The training procedure was stopped using cross-validation, after 100 consecutive worsening of the performance of the algorithm on the validation set.

From a preliminary analysis we noted that the NN was particularly inaccurate in predicting hypoglycemia, especially for PHs longer than 30 min. This is likely due to the fact that low glucose concentration values are a small percentage of the data, thus their impact on the MSE objective function minimized during training is minimal. To improve the NN performance in the hypoglycemic range, weights proportional to the risk of hypoglycemia [113] were used during training to increase the weight of prediction errors when the target glucose concentration is below 100 mg/dL.

5.5 Test-bed

The algorithms are optimized and tested on data collected during the project DIAdvisor [85]. In particular data of 15 type 1 diabetic patients, monitored for 3 consecutive real-life days are considered. Part of this dataset coincides with that used for the jump NN described in Chapter 4. Some new time series, rendered available only at the end of the project, were included while some of the time series used previously had to be discarded, because insulin information was missing. CGM was measured by the SEVEN PLUS CGM sensor, (T_S=5 min).

The dataset was divided into a training and validation set (including the first day of monitoring of every subject) and a test set (containing the following two days of monitoring of every subject). The training and validation set was further randomly divided into a training set (containing the 70% of data) and a validation set (formed by the remaining 30% of data).

Remark: Since the NN predictors we consider will be intended as “population” models, every NN is optimized on the whole training and validation set and then assessed on every profile of the test set.

5.6 Results

5.6.1 Assessment on the entire time window

Figure 5.2 shows glucose concentration during a 7 h time window of a representative subject together with the prediction obtained by the four NNs for PH=15 min (upper panel), PH=30 min (second panel), PH=45 min (third panel) and PH=60 min (bottom panel). The black dotted line is the target signal, as measured by the CGM sensor, the gray line is prediction obtained with NN CGM (using only CGM information), the green

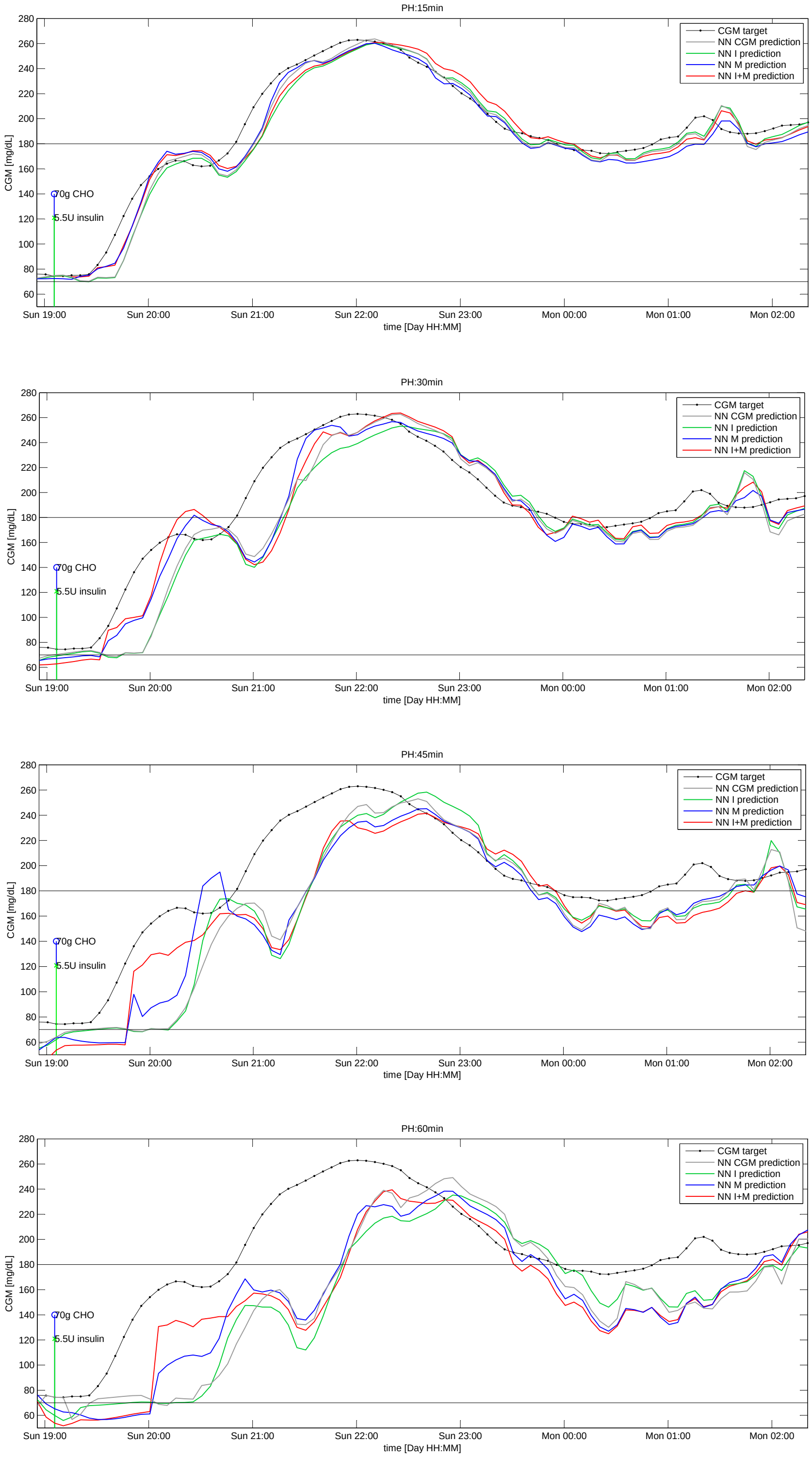


Figure 5.2: Representative CGM profile (black dotted line) and prediction obtained with the four NNs for PH=15, 30, 45 and 60 min (from top to bottom).

line is prediction obtained with NN I (using CGM and insulin information), the blue line is prediction obtained with NN M (using CGM and CHO information) and the red line is prediction obtained with NN I+M (using CGM, insulin and CHO information). The green and red stems represent, respectively, insulin injection and CHO ingestion. Adding to CGM information on CHO and insulin (red line) or information on CHO only (blue line) visually improves the prediction over the 2 h time window following CHO ingestion and insulin injection. If we concentrate on the time frame 19:00-21:00, we note that for all PH but 15 min NN I+M and NN M forecast with a minimum delay the upward trend following the ingestion of CHO, while NN I and NN CGM have a delay almost comparable to PH. On the contrary, in the rest of the profile all the NNs show similar performance and the predicted profiles almost coincide.

From numerical results computed on the entire monitoring, reported in Table 5.1 and in the boxplots of Figure 5.3, we can note that there is no evident difference among the four NNs. This is expected since ingestion of CHO and injection of insulin largely

Table 5.1: Average results (mean±sd) for the 15 test time series computed on the entire test time series.

	PH	NN CGM	NN I	NN M	NN I+M
RMSE [mg/dL]	15 min	13.6±3.5	13.6±3.3	13.6±3.6	13.9±3.7
	30 min	26.0±4.9	25.8±4.6	26.0±5.0	26.4±5.8
	45 min	37.0±5.6	37.1±5.8	37.2±5.5	35.1±6.4
	60 min	47.4±7.2	48.0±7.2	46.1±7.5	44.2±8.2
TG$_{\text{norm}}$ [-]	15 min	0.44±0.1	0.36±0.09	0.42±0.15	0.42±0.15 6
	30 min	0.30±0.07	0.30±0.07	0.34±0.08	0.42±0.11
	45 min	0.26±0.07	0.24±0.06	0.27±0.07	0.31±0.10
	60 min	0.19±0.04	0.17±0.04	0.26±0.07	0.26±0.10
ESOD$_{\text{norm}}$ [-]	15 min	4.2±1.0	3.7±0.7	3.4±0.9	3.1±0.6
	30 min	8.0±2.5	6.5±1.8	7.0±2.1	7.0±2.0
	45 min	11.2±4.9	9.4±4.2	10.7±3.7	7.6±2.4
	60 min	13.5±6.0	7.6±3.2	13.9±5.4	10.5±4.6

influence glucose time course mostly during the 2 h following the events. Therefore, we expect insulin and/ or CHO information to improve prediction during those limited time intervals, which constitute approximately the 25% of the test time series. For this reason, in Subsection 5.6.2 we evaluate the four predictors separately, on the 2 h time window following CHO ingestion and insulin injection and during the night.

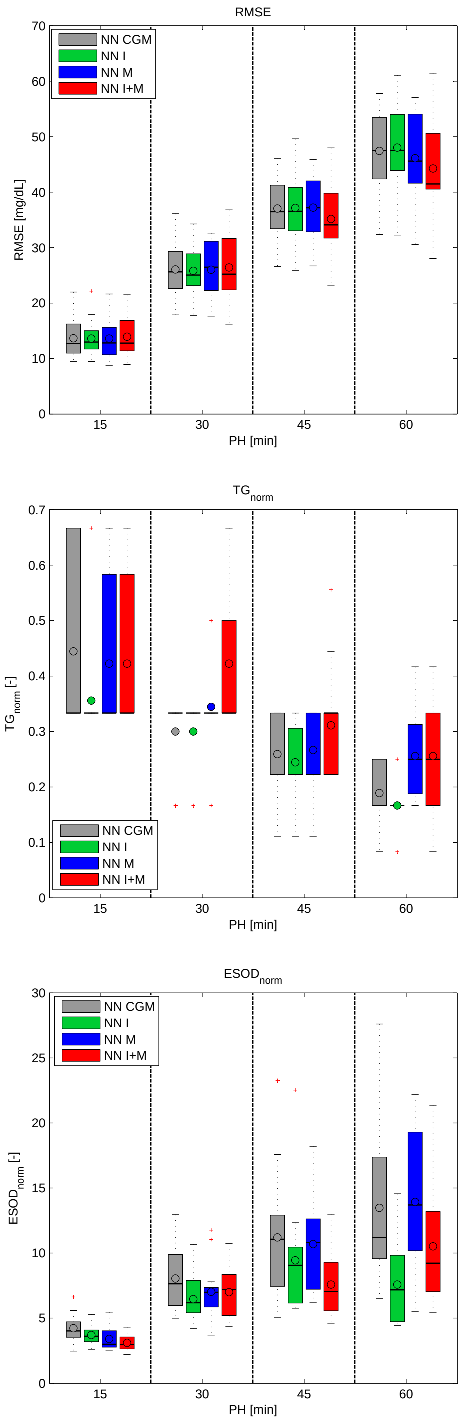


Figure 5.3: Boxplots summarizing the performance of the proposed models in terms of RMSE, average TG_{norm} and $ESOD_{norm}$ on the entire test time series. For each box the horizontal lines represent, from bottom to top, the 25th, the 50th and the 75th percentile respectively, the whiskers extend until the most extreme values, the red crosses represent outliers and the circle corresponds to the average.

5.6.2 Assessment on specific time windows

Figure 5.4 shows in a representative test time series the prediction obtained with the compared models during the 2 h following CHO ingestion and insulin injection (left column) and during the night (i.e. from 23:00 to 06:00), when no CHO are ingested and no insulin is injected (right column). Focusing on the left column, adding to CGM inputs relative to insulin and CHO, or adding to CGM inputs relative to ingested CHO only, improves the accuracy of prediction during the 2 h following the injection of insulin and ingestion of CHO. Both NN I+M (red) and NN M (blue) forecast glucose concentration more accurately than NN I (green) and NN CGM (gray) and with a lower delay. On the contrary, plots in the right column clearly show that during night, (when glycemia is stable and, usually, no CHO is ingested and no insulin is injected), all the models have similar performance.

Figure 5.4 allows us also to discuss the usefulness of exogenous inputs for different PHs. Taking into account exogenous signals does not improve prediction with a PH of 15 min (top panel of Figure 5.4). This is reasonable, since, due to physiological delays and to the relatively slow dynamics of the glucose insulin system, injected insulin and ingested CHO do not affect glycemia instantaneously, thus their effects are not significant after 15 min. Differently, with PHs of 30, 45 and 60 min, adding to CGM information also inputs relative to injected insulin and ingested CHO, or relative, at least, to ingested CHO, visibly improves prediction adherence to the target and time anticipation. However, with a PH of 60 min all the models perform quite poorly, suggesting that inferring relationships between the current inputs and future glucose concentration 60 min ahead in time is too challenging with the models we adopted and information used.

Figure 5.5 shows graphically the performance of the compared algorithms in terms of RMSE, TG_{norm} and $\text{ESOD}_{\text{norm}}$, computed both in the 2 h window following the ingestion of CHO and injection of insulin and during the night. For the 2 h window following CHO and insulin, for PHs greater than 15 min NN I+M and NN M have a RMSE visibly lower than the other models (top left panel). Also TG_{norm} is visibly higher for NN I+M and for NN M, compared to NN I and NN CGM (central left panel). Finally, the value of $\text{ESOD}_{\text{norm}}$ is comparable for all the NNs (bottom left panel). During the night, differences are not so evident and all the models obtain similar RMSE, TG_{norm} and $\text{ESOD}_{\text{norm}}$ values.

Table 5.2 summarizes average results obtained for the compared models for the analyzed PHs. Performance are computed separately, in the 2 h time window following CHO ingestion and insulin injection and during the night. Statistically significant differences between results obtained with NN I+M and results obtained by the other

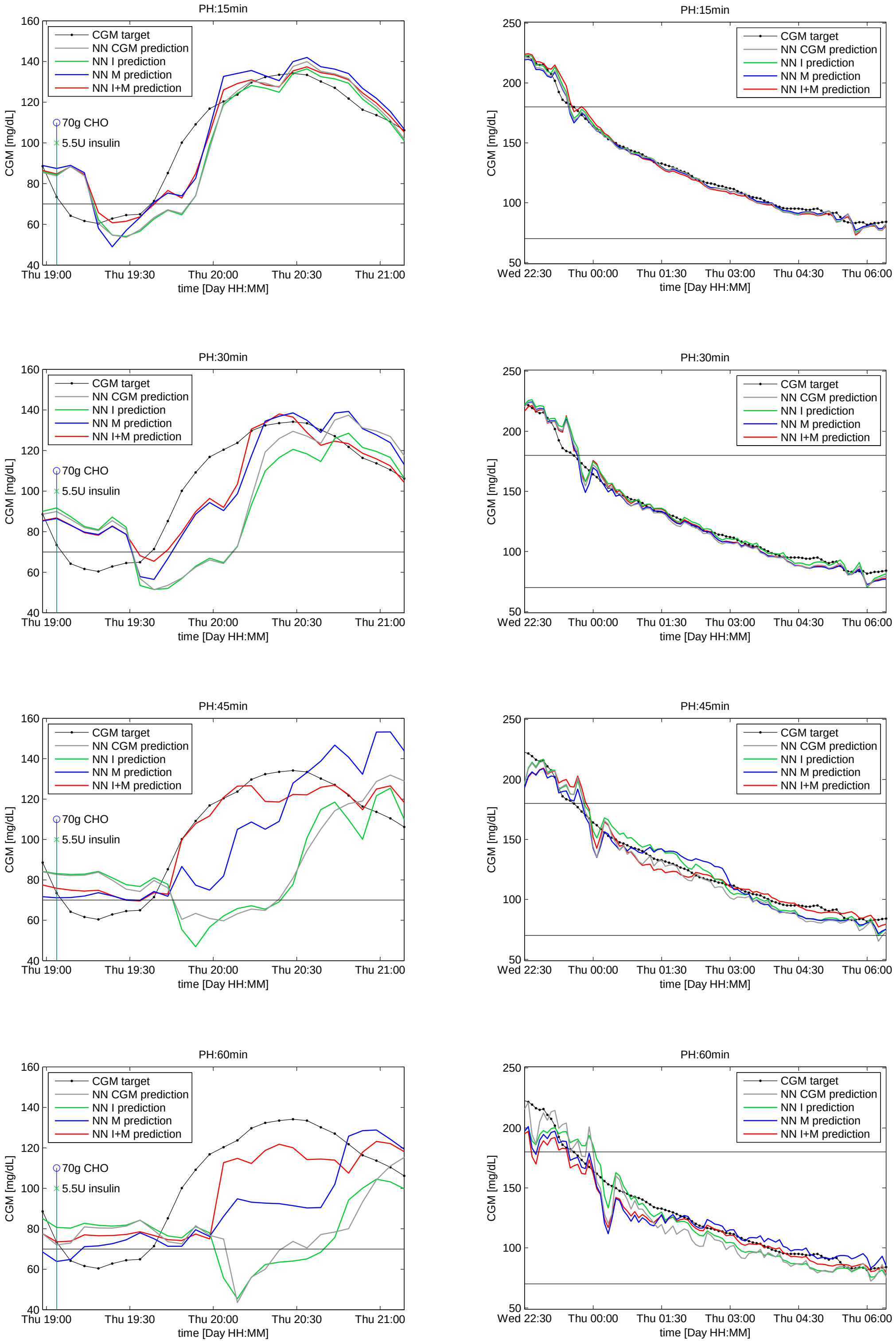


Figure 5.4: Representative subject. Prediction performance in the 2 h time window following CHO ingestion and insulin injection (left column) and during the night (right column). Vertical stems represent insulin injection (green) and CHO ingestion (blue).

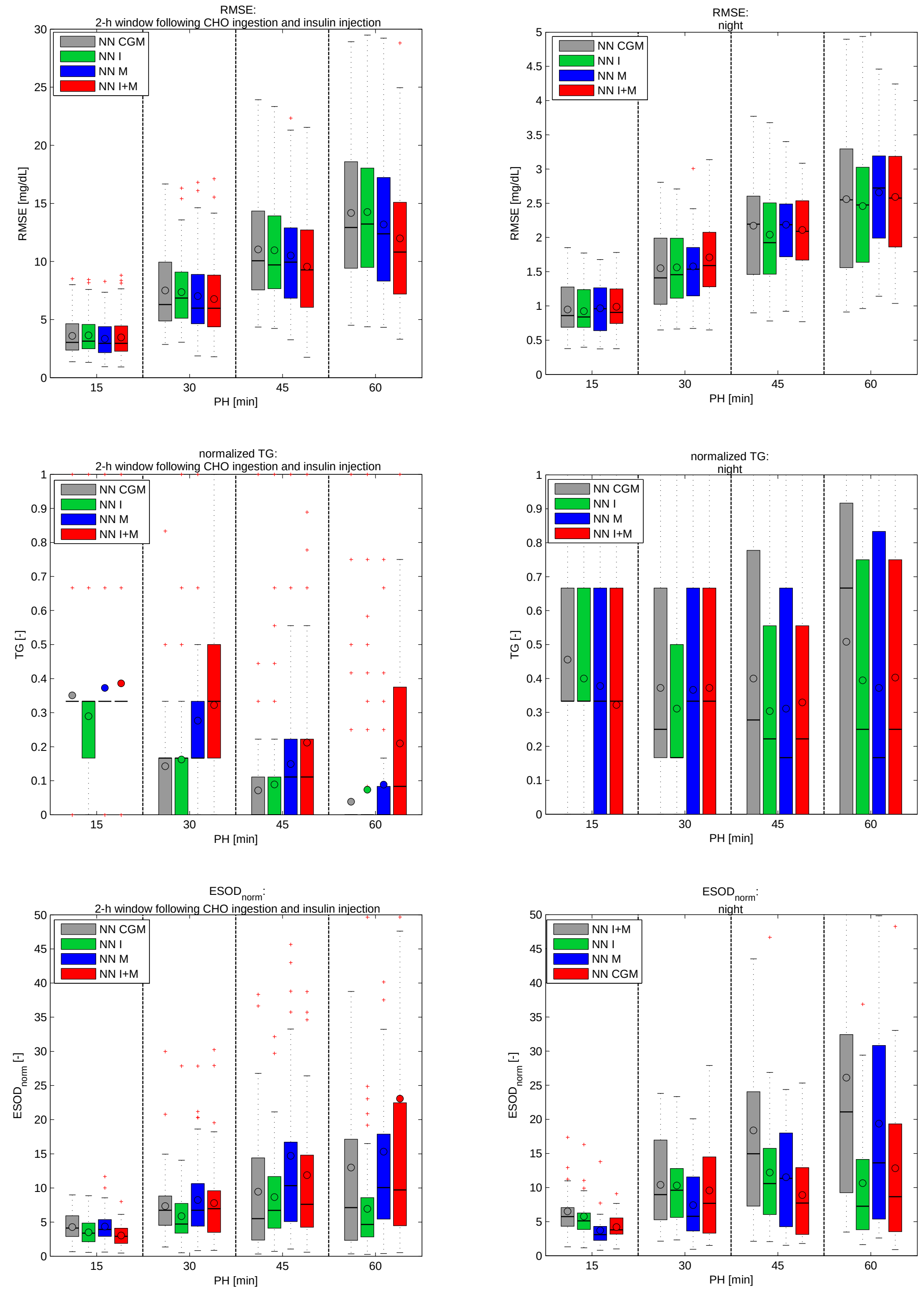


Figure 5.5: Boxplots summarizing the performance of the models in terms of RMSE, average $\mathrm{TG_{norm}}$ and $\mathrm{ESOD_{norm}}$ during the 2 h time window following CHO ingestion and insulin injection (left column) and during the night (from 23:00 to 06:00) (right column).

NNs are indicated by an asterisk and are computed using the sign test[1] [106].

For what concerns RMSE computed on the time window following CHO ingestion and insulin injections, with a PH of 15 min NN I performs significantly worse than all the other models; in addition, NN M performs significantly better than NN CGM. With a PH of 30 min NN I+M significantly improves on NN M, NN I and NN CGM; NN M significantly improves on NN I and NN CGM. With a PH of 45 min NN I+M significantly outperforms all the other NNs and NN M significantly improves on NN I. Finally, with a PH of 60 min NN I+M significantly improves on all the other predictors; while NN M improves on NN I and NN CGM. Differently, during the night, NN I+M has a RMSE significantly worse than the other models for a PH of 15 min and significantly worse than NN I and NN CGM for PH of 30 min. For longer PHs the differences are no more significant. For what concerns the average TG_{norm} relative to time-intervals following CHO ingestion and insulin injection, for a PH of 15 min the models show similar performance, apart from NN I whose TG_{norm} is significantly worse than those of the other NNs. For a PH of 30 min both NN I+M and NN M significantly improve on NN I and NN CGM. For a PH of 45 min NN I+M significantly outperforms all the other NNs; while NN M significantly improves on NN I and NN CGM. Finally, for a PH of 60 min NN I+M again significantly improves on all the other models; NN M is significantly better than NN I and NN CGM; in addition NN I performs significantly better than NN CGM. During the night, for PH of 15 min, NN I and NN CGM have a TG_{norm} significantly higher than NN I+M, while, for longer PHs, no statistically significant difference is present. For what concerns $\text{ESOD}_{\text{norm}}$, results seem to not depend on ingestion of CHO and injection of insulin and are acceptable for all the NNs.

From the above results we can conclude that when inputs relative to ingested CHO and injected insulin are added to CGM information, the NN ability of predicting glucose concentration after CHO ingestion and relative insulin injections is significantly improved for PHs longer than, or equal to, 30 min. Adding only injected insulin to CGM information is not beneficial for the NN. However, when we add to CGM both, injected insulin and ingested CHO, the forecasted signals obtained with PHs of 45 and 60 min are more accurate and have a higher TG than those obtained when we add to CGM only ingested CHO information.

Difficulties of the NN in taking advantage of the input relative to injected insulin may be due to many factors including the intra- and inter-individual variability of delay in insulin action and absorption [111, 114]. Interestingly, during the night, when effects of CHO ingestion and insulin injection are negligible, (only a quasi-constant basal insulin is

[1] The sign test is a paired, two-sided test of the hypothesis that the difference between the matched samples in the two vectors of results comes from a distribution whose median is zero.

Table 5.2: Average results (mean±sd) for the 15 test time series computed separately, during the 2 h time window following CHO ingestion and insulin injection and during night. Asterisk (*) indicates statistical difference, computed with the sign test, between NN I+M and the considered NN.

		2h time window following CHO ingestion and insulin injection				Night time window			
	PH	NN CGM	NN I	NN M	NN I+M	NN CGM	NN I	NN M	NN I+M
RMSE [mg/dL]	15 min	3.6±1.6*	3.6±1.6*	3.3±1.6	3.5±1.7	1.0±0.4*	1.0±0.4*	1.0±0.4*	1.0±0.3
	30 min	7.5±3.4*	7.4±3.0*	7.0±3.3*	6.8±3.2	1.5±0.6*	1.6±0.6*	1.6±0.5	1.7±0.6
	45 min	11.0±4.6*	11.0±4.5*	10.5±4.7*	9.5±4.4	2.2±0.8	2.0±0.7	2.2±0.6	2.1±0.6
	60 min	14.2±6.0*	14.2±5.9*	13.2±5.9*	12.0±5.6	2.6±1.0	2.5±0.9	2.6±0.8	2.6±0.8
TG_{norm} [-]	15 min	0.35±0.13	0.29±0.21*	0.37±0.18	0.39±0.22	0.46±0.31*	0.40±0.31*	0.38±0.38	0.32±0.32
	30 min	0.14±0.14*	0.16±0.17*	0.28±0.21	0.32±0.27	0.37±0.30	0.31±0.25	0.37±0.35	0.37±0.35
	45 min	0.07±0.18*	0.09±0.17*	0.15±0.21*	0.21±0.27	0.40±0.39	0.31±0.29	0.31±0.34	0.33±0.34
	60 min	0.04±0.16*	0.07±0.21*	0.09±0.16*	0.21±0.26	0.51±0.43	0.39±0.39	0.37±0.40	0.40±0.41
$ESOD_{norm}$ [-]	15 min	4.3±1.9*	3.5±1.7*	4.3±2.1*	3.0±1.5	6.5±3.4*	5.8±3.0*	3.7±2.4*	4.2±1.9
	30 min	7.3±4.4	5.9±4.1*	8.2±5.3	7.8±5.6	10.4±6.3	10.3±5.8	7.4±5.0*	9.6±7.2
	45 min	9.4±10.7	8.7±6.0	14.7±17.9	11.9±14.9	18.4±15.1*	12.2±9.5*	11.5±7.4	8.9±6.3
	60 min	13.0±15.0	6.9±7.4*	15.3±18.6	23.1±56.9	26.1±23.5*	10.6±9.2	19.4±14.9*	12.8±11.4

present), the NN using only CGM information is the most accurate. In fact NN CGM has less parameters to tune during training and learns more accurately the relationship between current and future glycemia, when no other disturbance influences glucose time course.

5.6.3 Results interpretation in terms of prediction sensitivity to inputs

Results shown and commented above suggest that the information on ingested CHO is the most useful for improving prediction results, while the information relative to injected insulin only slightly helps when added to the information on ingested CHO and is not sufficient to ameliorate prediction when used alone. In addition, the difference between NNs using, in addition to CGM, information on insulin injection and CHO ingestion, or on CHO ingestion only and the other two NN models becomes more evident when PHs equal or longer than 30 min are considered.

To quantify the individual usefulness of the various input signals in determining the NN output we performed a sensitivity analysis by Partial Derivative (PaD) method [115,116]. This method starts by computing, analytically, the partial derivative of the NN output with respect to each input

$$\begin{aligned} d_i(t) &= \frac{\partial \hat{y}(t+PH|t)}{\partial x_i(t)} \\ &= \omega_i + \sum_{j=1}^{N_{hn}} \psi_j \varphi' \left(\sum_{k=0}^{N_{in}} \omega_{jk} x_k(t) \right) \omega_{ji} \end{aligned} \tag{5.4}$$

with φ' derivative of the tangent hyperbolic function. d_i is a time series showing the time course of the output derivative for small changes of the i^{th} input. Then the relative contribution of each input variable on the specific output is determined by computing the sum of the squares of the partial derivatives

$$SS_i = \sum_{j=1}^{N} d_i(j)^2 \tag{5.5}$$

with N length of the time series. Finally, the relative contribution of each input variable is given by

$$S_i = \frac{SS_i}{\sum_{k=1}^{N_{in}} SS_k} \tag{5.6}$$

The variable with the highest S has the most effect on the output, with respect to the other variables. S allows to rank the relative influence of each input on the output, with respect to the other input signals and we can also observe how this influence changes when different PHs are considered. Figure 5.6 shows, for every model, the relative output sensitivity to the various inputs for the considered PHs. For what concerns NN CGM,

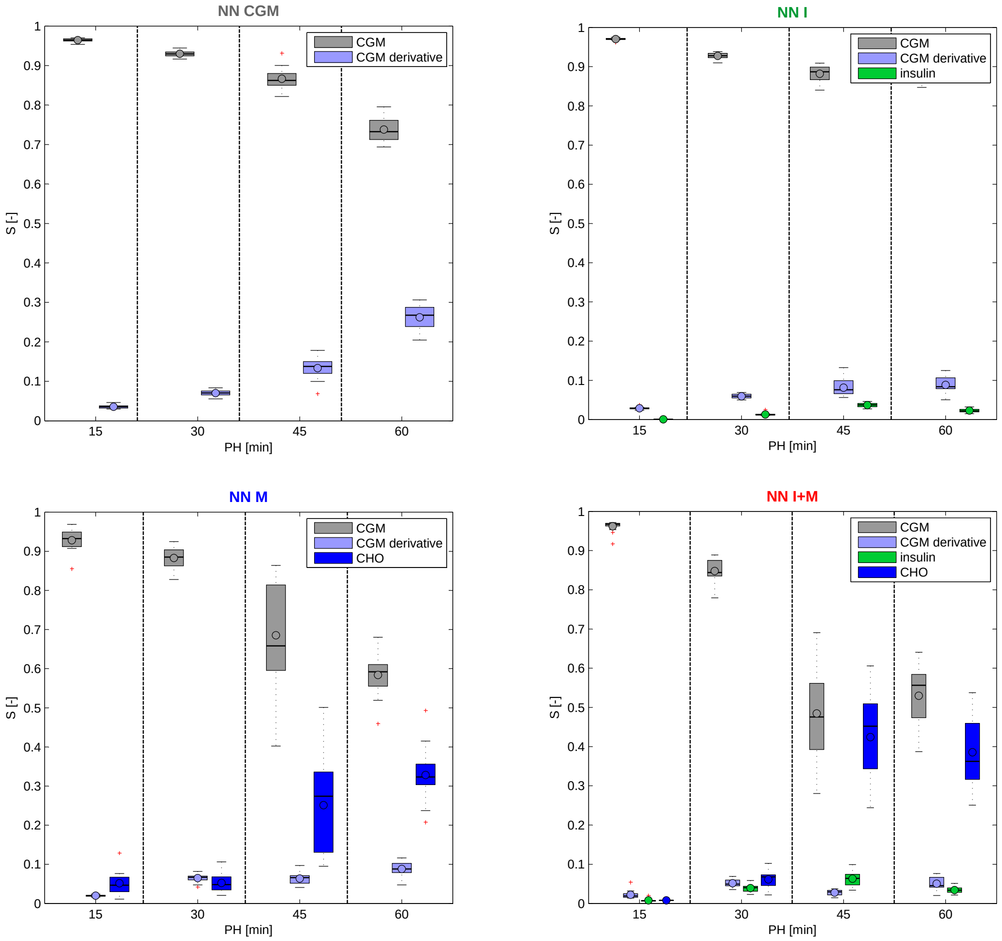


Figure 5.6: Boxplots of relative output sensitivity to inputs for the various models.

(Figure 5.6, top left panel), it relies mainly on CGM information for short PHs, but the relevance of CGM derivative increases when longer PHs are considered. For NN I (Figure 5.6, top right panel) CGM is by far the most informative input for all PHs. The output sensitivity to CGM derivative increases as the PH increases, while the sensitivity to the input relative to insulin is very low, even if it slightly increases as the PH increases. For NN M, (Figure 5.6, bottom left panel), CGM is far the most significant input for PHs of 15 and 30 min, while for PHs of 45 and 60 min the output sensitivity to the input relative to CHO becomes non-negligible. For NN I+M, (Figure 5.6, bottom right panel), for a PH of 15 min current CGM is the most significant input; for a PH of 30 min CGM is still the most informative input, however, the importance of the other signals slightly increases; for PHs of 45 and 60 min the importance of CGM and of CHO is comparable and visibly higher than that of insulin and CGM derivative.

This analysis confirms that glucose concentration history is the most informative signal for predicting glycemia, especially in the short- and in the mid-term (15-30 min). However, the future glucose concentration is sensitive to information on ingested CHO for a PH longer than 30 min. For what concerns information relative to injected insulin, it is more difficult to use them adequately and, in our analysis, it improves prediction only for PHs longer than 30 min if added to information on ingested CHO.

It is of interest to note that inputs relative to injected insulin and ingested CHO can influence glucose prediction only after CHO ingestion and insulin injections, i.e. approximately for 25% of time considering 3 meals and associated injections of insulin per day. This might justify the lower sensitivity of prediction to signals relative to ingestion of CHO and injection of insulin, with respect to the CGM signal.

5.7 Conclusions and margins for future work

In this Chapter we investigated if adding information relative to insulin therapy as additional input of the jump NN presented in Chapter 4, which uses CGM and CHO related inputs, could improve prediction, evaluating PHs in the range 15-60 min. A major limitation of using both, CHO and insulin information, comes from their high correlation, since the injection of an insulin bolus in usually concomitant with the ingestion of CHO and they are proportional. Moreover, even their simulated rate of appearance in the blood are similar. To overcome this problem and take also into account delays in insulin action we delayed the input relative to insulin of 60 min, as estimated in [111]. Results suggest that adding insulin and CHO to CGM information improves prediction performance when PHs longer than, or equal to, 30 min are considered, but only if we restrict our attention to the 2 h time window following the ingestion of CHO and relative insulin injection. This can be justified by the fact that effects of CHO and insulin are evident for approximately 2 h. Indeed if we compute the results on the entire monitoring, or during night, when exogenous disturbances should be absent or quasi-constant, all the NNs perform similarly. Surprisingly, when insulin alone was added to CGM information, no improvement was obtained. However, a possible justification of this result could be a non-adequate preprocessing of insulin information, due to difficulties in modelling its effects because of high variability of delay in its action [111] and absorption, determined by many, often not measurable, concurrent factors (e.g. insulin on board, injection site, skin temperature, etc [114]). To better interpret the obtained results we performed an analysis of prediction sensitivity to inputs and results confirmed that future glucose concentration is mainly sensitive to past CGM history and CHO information becomes visually relevant for PHs longer than 30 min.

In light of the finding that adding information on quantity of ingested CHO and injected insulin improves prediction accuracy only during the limited time window that follows the ingestion of CHO and injection of insulin, a possible future analysis could include the implementation of several NN-based models, using different combinations of input signals. The final prediction could be obtained as a weighted sum of the output of all the considered models, with weights proportional to the performance of each model and to its expected validity, in the considered time instant.

Furthermore, an additional improvement of prediction accuracy could be obtained by incorporating, among the inputs of the NN, also signals relative to PA, as preliminarily discussed in Chapter 6.

6

Use of Physical Activity (PA) on glucose prediction algorithms: preliminary analysis

6.1 Rationale

In Chapters 3 and 4 we demonstrated that adding information on time and quantity of ingested CHO to CGM history as inputs of a NN predictor improves results, with respect to models using only information on glucose concentration. Moreover, in Chapter 5 we investigated the possibility of incorporating also information relative to insulin therapy as input of the predictor. We pointed out that CHO quantity and insulin dose are highly correlated, thus using both signals does not guarantee the improvement of prediction results. Moreover, it is difficult to exploit adequately inputs relative to insulin therapy due to physiological delays and inter- and intra-individual variability in insulin action and absorption. In Chapter 5 we also demonstrated that CHO and insulin information effectively improve prediction performance only in a short time frame (approximately 2 h) following CHO ingestion and insulin injection. The improvement is no more appreciable if performance are computed during night, when exogenous disturbances should be absent or quasi-constant.

Additional promising inputs that could be consider are signals relative to PA. PA is uncorrelated from meal and insulin signals and is known to have short and long term effects on glucose dynamics. However, although effects of PA on glucose metabolism are qualitatively quite well developed, their quantification and incorporation into mathematical models, for scopes including e.g. glucose prediction and T1D simulation, is still an existing problem.

As a preliminary analysis, we investigated [117], quantitatively, the short-term correlation between variations of glucose concentration dynamics and the PA related signal returned by Physical Activity Monitoring System (PAMS), a system comprising accelerometers and inclinometers, able to detect and quantify PA, even at low intensity that mimic activities of daily living [118].

6.2 Database and protocol

Data used for this analysis were collected in the Clinical Research Unit at Mayo Clinic, (Rochester, MN) as part of an in-patient study designed to detect glycemic patterns and postprandial insulin sensitivity in control and T1D subjects, in presence of mild PA [119]. 20 control and 19 T1D individuals were studied for 88 hours. Each day they were fed with 3 meals, each one containing 80 grams of CHO, similar macronutrient and calories compositions, without differences between meals or between days. C-peptide negative T1D subjects were on insulin pump and administered an insulin bolus with meals. Each day subjects took part to 4 to 6 consecutive sessions of low intensity PA in which they alternated 26.5 min of walking on a treadmill at 1.2 mph with 33.5 min of sitting. The distance covered daily varied from 3.5 to 4.2 miles. It is worth noting that the walking velocity was chosen to be consistent with median free living walking velocity, since the protocol wanted to mimic activities of daily living.

PA data were collected using PAMS, a system that captures data on body posture and movements continuously every half second for up to 10 consecutive days [118,120,121]. As shown in Figure 6.1, PAMS comprises 2 tri-axial accelerometers (each captures motion along three orthogonal axis) and 4 inclinometers, (each captures two axis of acceleration against the gravitational field) for recording body posture and movements. The 2 accelerometers were placed over the base of the spine; the inclinometers were attached to the left and right outer aspect of the trunk, and left and right outer aspect of the thigh. Specially designed underwear was used to attach the sensors. The accelerometers measure PA data along three orthogonal axis, (x, y, z), with the dynamic range to ±2g (with g gravitational acceleration). The outcome PAMS signal, expressed in activity units (AU), is obtained summing the instantaneous acceleration over epochs of 1 min [121,122].

Glucose concentration was monitored continuously with the Dexcom SEVEN PLUS

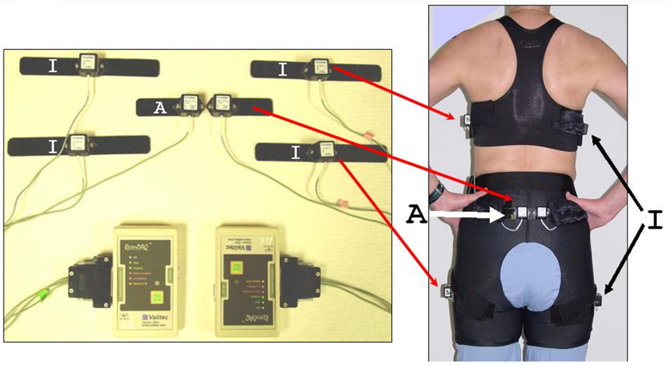


Figure 6.1: PAMS comprises 4 inclinometers (I); 2 tri axial accelerometers (A) and 2 data loggers. The system is worn as shown in the right panel.

CGM device. Figure 6.2 shows two typical piece of data, measured in a T1D subject, walking sessions are highlighted in gray. The top panels represent the CGM time course, the bottom panels show the PAMS signal.

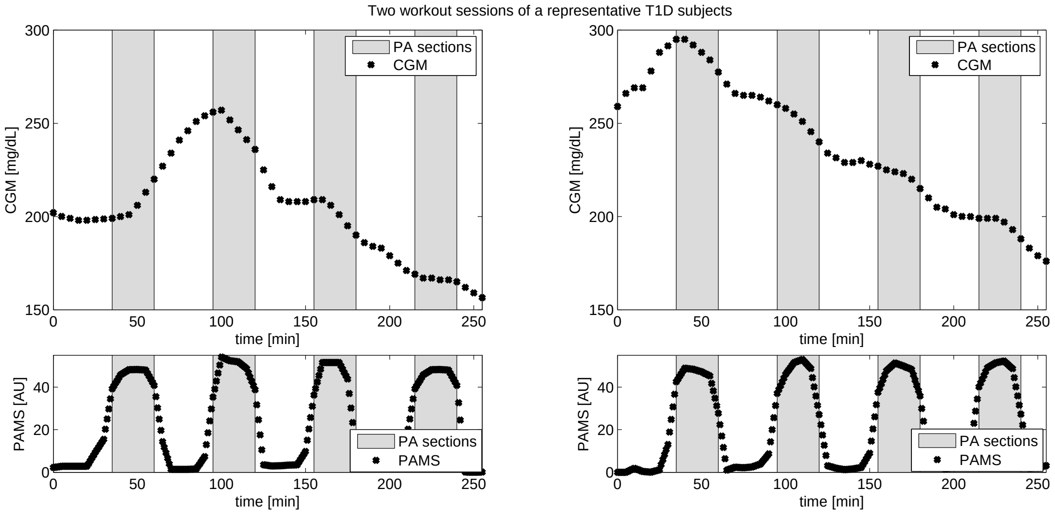


Figure 6.2: CGM time series (top panels) and PAMS measurements (bottom panels) during two workout sessions of a representative T1D subject. Walking sessions are highlighted in gray.

Since we wish to assess the effects of PA on variations of glucose dynamics, quantified via first- and second-order glucose time-derivatives, only piece of data relative to consecutive PA sessions (i.e. repetitions of active and resting time) are considered in our analysis, without including any long sedentary period. In the rest of the chapter, we will refer these portions of data as *workout sessions.* According to the protocol, for each patient 3 to 4 workout sessions were recorded. In addition, a time alignment of the signals is performed: this procedure simply consists in down-sampling PAMS, whose original sampling time was of 1 min, considering only those values measured at the same time instants at which CGM signal, whose sampling time was 5 min, is also available.

6.3 Computation of glucose concentration time-derivatives

Changes in glucose dynamics were quantified by computing the first- and second-order time-derivatives of glucose concentration. In particular, a Bayesian smoothing approach, similar to that already employed in [108, 112] to denoise CGM data, was used to face ill-conditioning of derivatives calculation and limit artifacts due to measurement noise affecting CGM readings.

Briefly, in a matrix-vector embedding, the N-size vector $\mathbf{y}$ containing the (uniformly spaced) CGM samples is modelled as

$$\mathbf{y} = \mathbf{G}\mathbf{u} + \mathbf{v} \tag{6.1}$$

where $\mathbf{u}$ is the N-size vector containing the samples of the (unknown) time derivative, $\mathbf{v}$ is the random vector of the measurement errors (assumed uncorrelated, with zero mean and constant unknown variance), and $\mathbf{G}$ is an N-size lower triangular square Toeplitz matrix having as its first column $T_s[1, 1, \ldots, 1]$ or $T_s^2[1, 2, \ldots, N]$, respectively, if the vector of the first-order or second-order time derivative is estimated (T_s is the sensor sampling period). Because of ill-conditioning, LS estimation of $\mathbf{u}$ given $\mathbf{y}$ in equation (6.1) is unreliable, and a Bayesian regularization approach [108, 112] similar to that applied by Guerra et al. [123] for glucose trend estimation from CGM data, is used. According to this approach, the estimated $\hat{\mathbf{u}}$ is computed as

$$\hat{\mathbf{u}} = (\mathbf{G}^T\mathbf{G} + \gamma\mathbf{F}^T\mathbf{F})^{-1}\mathbf{G}^T\mathbf{y} \tag{6.2}$$

where γ is the regularization parameter, whose value is determined according to a maximum likelihood/consistency criterion, and $\mathbf{F}$ is a squared N-size lower triangular Toeplitz matrix that, according to considerations on CGM data explained in detail by Facchinetti et al. [108, 112] has a first column equal to $[1, 2, 1, 0, \ldots, 0]$.

6.4 Partial correlation analysis

For each workout session, the relationship between PAMS and glucose concentration time derivatives was quantified by partial correlation computed at various time shifts (τ) in the range 0-60 min. We could not choose time shifts greater than 60 min because of constraints of our protocol: indeed the subject starts a PA session (walking on treadmill plus resting) every hour, thus restricting our analysis to time shifts shorter than or equal to 60 min is essential to avoid superimposition of effects of consecutive PA sessions. Partial correlation measures the degree of association between two signals, removing the effect of a set of controlling signals. Specifically, the controlling signals were CGM, meal and insulin (the last one only for T1D patients) related information. In particular, meal intakes were preprocessed to generate glucose rate of appearance in the blood [103], while insulin dosages were used to calculate the so-called insulin on board with the formulas described in [124]. Using partial instead of conventional correlation guarantees that results are not affected by any collateral effects of either glucose concentration value, CHO ingestion or insulin injections and they quantify exclusively the correlation between PAMS and glucose derivatives.

Mathematically, the partial correlation between $\mathbf{X}$ (in our case PAMS) and $\mathbf{Y}$ (in our case first- or second-order glucose time derivative), given a set of n controlling variables $\mathbf{Z} = \{\mathbf{Z_1}, \mathbf{Z_2}, ..., \mathbf{Z_n}\}$ (in our case CGM, glucose rate of appearance and insulin on board), indicated as $\rho_{\mathbf{XY},\mathbf{Z}}$, is the correlation between the residuals $\mathbf{r_X}$ and $\mathbf{r_Y}$ resulting from the linear regression of $\mathbf{X}$ with $\mathbf{Z}$ and of $\mathbf{Y}$ with $\mathbf{Z}$, respectively. Solving the linear regression problem requires to find the n-dimensional weight vectors

$$\mathbf{w}^*_{\mathbf{X}} = \arg\min_{w} \left\{ \sum_{i=1}^{N} (x_i - \langle \mathbf{w}, \mathbf{z}_i \rangle)^2 \right\} \tag{6.3}$$

$$\mathbf{w}^*_{\mathbf{Y}} = \arg\min_{w} \left\{ \sum_{i=1}^{N} (y_i - \langle \mathbf{w}, \mathbf{z}_i \rangle)^2 \right\} \tag{6.4}$$

where N is the length of the time series, and $\langle \mathbf{w}, \mathbf{z}_i \rangle$ represents the internal product between vector $\mathbf{w}$ and vector $\mathbf{z}_i$. Given the weight vectors of (6.3) and (6.4), the residuals can then be computed, respectively, as

$$\mathbf{r}_{\mathbf{X},i} = x_i - \langle \mathbf{w_X}^*, \mathbf{z}_i \rangle \tag{6.5}$$

$$\mathbf{r}_{\mathbf{Y},i} = y_i - \langle \mathbf{w_Y}^*, \mathbf{z}_i \rangle \tag{6.6}$$

and the partial correlation is given by the formula

$$\hat{\rho}_{\mathbf{X}\mathbf{Y},\mathbf{Z}} = \frac{N\sum_{i=1}^{N} \mathbf{r}_{\mathbf{X},i}\mathbf{r}_{\mathbf{Y},i} - \sum_{i=1}^{N} \mathbf{r}_{\mathbf{X},i} \sum_{i=1}^{N} \mathbf{r}_{\mathbf{Y},i}}{\sqrt{N\sum_{i=1}^{N} \mathbf{r}_{\mathbf{X},i}^2 - \left(\sum_{i=1}^{N} \mathbf{r}_{\mathbf{X},i}\right)^2}\sqrt{N\sum_{i=1}^{N} \mathbf{r}_{\mathbf{Y},i}^2 - \left(\sum_{i=1}^{N} \mathbf{r}_{\mathbf{Y},i}\right)^2}} \tag{6.7}$$

6.5 Results

6.5.1 Correlation between PAMS and first order glucose time derivative

Median results computed on the 19 T1D and on the 20 control subjects are graphically shown in Figure 6.3. In diabetic subjects (left), there is a negative correlation between first-order glucose concentration derivative and PAMS for τ lower than 30 min, with a maximum, in absolute terms, for τ equals 15 min. For τ in the range (30, 60) min the correlation becomes positive, and is maximal with τ equals 40-45 min. Results relative to control subjects (right) are similar, however the degree of correlation is smaller (in absolute terms), and correlation peaks are anticipated of 5-10 min compared to what we observed on T1D. These results suggest that low intensity PA decreases glucose concentration in the short term, with a decrease particularly evident after 10-15 min and as exercise stops glucose tends to increase, with a maximal increase after 10-15 min.

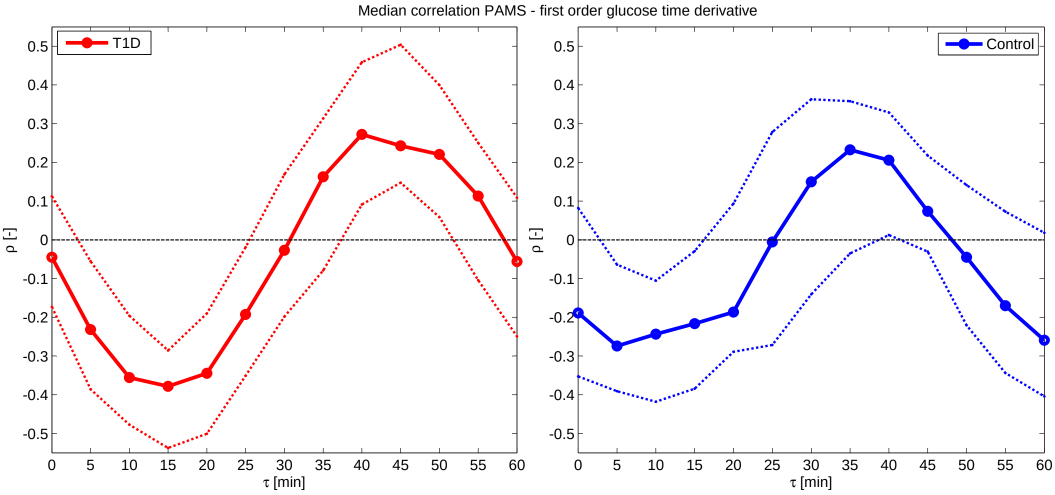


Figure 6.3: Median correlation curves (and 25th and 75th percentiles, dotted lines) between PAMS and first-order CGM derivatives, computed on the 19 T1D subjects (left) and on the 20 control subjects (right).

6.5.2 Correlation between PAMS and second order glucose time derivative

Median results are plotted in Figure 6.4. In diabetic subjects (left) there is a negative correlation between PAMS and second-order glucose time derivative for τ lower than 20 min, and this correlation is maximal (in absolute terms) for τ equals 5 min, while there is a positive correlation for τ in the interval (25, 45) min and this correlation is maximal for τ equals 35 min. In control subjects (right) the degree of correlation is slightly lower (in absolute value), and correlation peaks are anticipated with respect to T1D patients. Results relative to correlation between PAMS and second-order glucose concentration derivative suggest that, even in case glucose does not decrease during walking, at least it increases at a lower rate; furthermore, in case glucose concentration does not increase during rest, at least it decreases less rapidly.

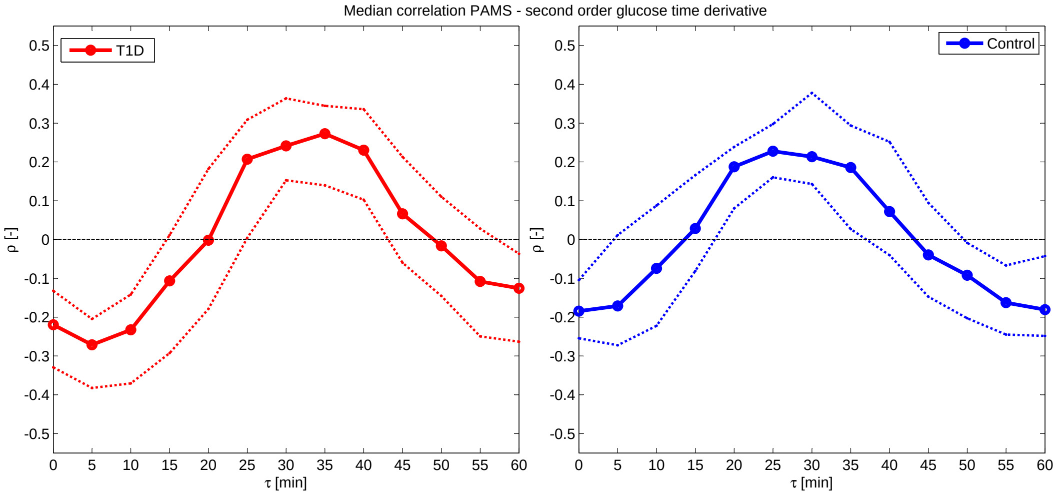


Figure 6.4: Median correlation curves (and 25th and 75th percentiles, dotted lines) between PAMS and second-order CGM derivatives, computed on the 19 T1D subjects (left) and on the 20 control subjects (right).

6.6 Conclusions and margins for further investigations

The aim of this preliminary analysis was quantitatively assessing if PA causes measurable variations of glucose dynamics in the short term ($\leq$ 60 min) and if those variations follow a typical pattern. This constitutes a necessary step before building quantitative models of PA effects on glucose concentration, e.g. for prediction or closed-loop control purposes. We quantified correlation between mild PA that reproduces every-day life activity, measured using the PAMS signal, and glucose trends, quantified estimating first and second order glucose time-derivatives from the CGM signal.

Results obtained on 19 T1D and 20 control subjects confirm a tendency of glucose concentration to decrease during exercise and to increase during rest periods. Interestingly, correlation is higher, in absolute value, in T1D than in control subjects, suggesting that in diabetic patients PA causes greater excursions of glucose concentration, as suggested in [119]. Moreover, in diabetic subjects response to PA in terms of glucose dynamics modification is slower than in control subjects.

The presence of short-term correlation between changes in glucose dynamics and mild exercise suggests the potential utility of including PA information in short-time prediction models, to infer more precisely future glucose concentration in presence of PA. The ability of predicting exercise effects on glucose concentration could be very helpful, since it would allow adapting insulin infusion during and after PA and it could forewarn against hypoglycemic events, by alerting the patient before their occurrence. The effective quantitative incorporation of PA information within glucose predictors could be matter of in depth future investigations. For instance, PA information could be exploited to dynamically modulate the forgetting factor typically used in low-order time varying AR/polynomial models. Furthermore, signals relative to PA, possibly preprocessed in line with the results of this correlation analysis, could be included as inputs of NN prediction models.

7

Clinical usefulness of prediction for generation of hypoglycemia alerts: a comprehensive *in silico* study

7.1 Rationale

One of the major issues in diabetes management is to limit hypoglycemic events. Indeed hypoglycemia has threatening short-term consequences, since, if not quickly detected and treated, it could progress from measurable cognition impairment, to aberrant behavior, seizure, coma and even death [12]. Commercial CGM devices generate visual/acoustic alarms in real time when measured glycemia crosses critical thresholds (e.g. 70 mg/dL for hypoglycemia) [125]. Preventing rather than simply detecting critical events when they occur would be preferable and, to do so, short-term (30-45 min) glucose prediction methods could be exploited [126, 127].

In the literature, the benefit deriving from the exploitation of prediction methods to prevent/ mitigate hypoglycemia by soliciting appropriate treatments (e.g. sugar intake and/or pump basal suspension) has been assessed from real data in [65, 80, 81, 128, 129], as described in Section 1.6 of the present thesis. However, it would be of interest to compare different scenarios occurring for the same patient and starting from the very same patient conditions, which is not possible in clinical studies, where every action has an effect on glycemia and, unavoidably, exclude the possibility of seeing what would have happened if different decisions were made, as explained in Section 1.7. Thus, the aim of this analysis is to use an *in silico* environment to quantify the potential benefits, in terms of number and duration of hypo-events, coming from the use of predicted, rather than measured, glucose for hypoglycemic alert generation in 50 synthetic subjects [130]. Virtual patients were created by the UVA/Padova type 1 diabetic simulator [73, 84], described briefly in Appendix A. The synthetic patients were virtually monitored in horizons of 54 h (including 2 lunch, 2 dinner and 3 breakfast events per patient), in presence of additive white noise with realistic variance corrupting CGM data, and of sources of uncertainty on the quantity of ingested CHO and injected insulin. Three parallel scenarios were considered:

1. the subject was unaware of hypoglycemia, no alert was generated and no counter-measures were taken when blood glucose concentration fell below the hypoglycemic threshold (worst case);
2. a hypo-alarm was triggered based on CGM measurements and 15 g of CHO were ingested by the patient;
3. a hypoglycemic alert was given on the basis of the 30-min ahead-of-time predicted glycemia, obtained via the NN-based algorithm described in Chapter 3, and 15 g of CHO were ingested by the patient, as in Scenario 2.

7.2 Creation of simulated realistic data

The database consists of 50 type 1 diabetic virtual patients, extracted from the UVA/Padova simulator [73, 84]. For each subject, one CGM time series has been simulated, consisting of about 2 days and a half of monitoring and sampling time of 5 minutes. The choice of this specific sampling time is due to the fact that it coincides with that of the majority of currently used CGM devices. Each simulated time series consists of 54 h of monitoring, from 03:00 of day 1 to 09:00 of day 3. The monitoring interval was chosen in order to be long enough to observe at least one hypoglycemic event for each subject. Moreover, since breakfast is administered between 06:00 and 08:00, termination of the monitoring interval at 09:00 allows patients to complete the recover from an eventual nocturnal hypoglycemia. Ten of the patients were further simulated for 4 additional days, and the relative profiles were used to train the NN prediction algorithm. Three meals per day were considered in the simulated scenario. To render the profiles more realistic, CHO intake quantities and meal timings were differentiated from meal to meal and from day to day. Breakfast was randomly located in the time interval 06:00-08:00 h and consisted of 35-55 g of CHO, lunch was in the interval 12:00-14:00 h and consisted of 60-90 g of CHO, finally dinner was in the interval 19:00-21:00 h and consisted of 70-100 g of CHO. For what concerns insulin, a basal-bolus infusion scheme was adopted, with boluses computed to counterbalance the effect of the concomitant meals. To obtain additional hypoglycemic events in the simulated profiles, overdosed insulin was administered. In particular, every day basal insulin was increased twice for 30 min of a random amount sampled from a uniform distribution in (0-3) U/h. This action has also an effect on glucose similar to an increase of insulin sensitivity or to a mild PA. Furthermore, for half of the patients randomly chosen, one insulin bolus was augmented once a day of a random percentage sampled from a uniform distribution in (0-30)%. For the other half of the patients, the size of one of the meals was simulated to be wrongly estimated and the amount of CHO effectively ingested was decreased of a percentage randomly chosen in the interval (0-30)%. Finally, in order to mimic the random measurement error affecting CGM, a white noise sequence whose samples were extracted from a Gaussian distribution with zero mean, and variance equal to 4 (in line with [131, 132]) was added to each time series.

To quantify the benefits coming from the exploitation of prediction-based hypo alerts, we compared the three scenarios described in the introduction of the present chapter. In Scenario 1, no hypo-alerts were generated and hypoglycemia was thought to be not recognized and dealt with. This corresponds to a sort of worst case situation for the diabetic patient, though possible especially during the night [133]. In Scenario 2, the alert was triggered on the basis of the measured CGM readings. In Scenario 3, the alarm was generated on the basis of predicted glycemia, obtained through the algorithm described in Chapter 3. Alert generation obeyed the simple strategy explained in Section 7.3. In both scenarios 2 and 3, a bolus of 15 g of CHO was ingested in the 5 min following the alert. Scenario 2 and Scenario 3 were assessed also in presence of randomly delayed/absent ingestion of CHO. Results are quantified in terms of number of hypoglycemic events, their duration and total time in hypoglycemic range. In addition we computed also the distribution of glucose concentration and the Low Blood Glucose Index (LBGI) and High Blood Glucose Index (HBGI) [113], two commonly adopted indicators of the risk of hypoglycemia and hyperglycemia, respectively. The highest the value of these indexes, the highest the associated risk.

Remark. In the spirit of keeping the protocol as simple as possible, the action associated with hypo alert was standardized to the ingestion of 15 g of CHO. According to [134] such a measure is commonly adopted by diabetic patients and has the effect of raising glycemia of about 50 mg/dL in approximately 15 min. In addition, basal insulin infusion was neither suspended, nor attenuated (differently from [65,80,81]), also because this would be expected to have delayed effect.

7.3 Hypoglycemic alert generation strategy

The prediction strategy adopted is the one described in Chapter 3. For what concerns the generation of hypo alerts, we consider a basic procedure which generates an alert when the glucose profile (measured by the CGM sensor in Scenario 2, forecasted by the prediction algorithm in Scenario 3) crosses 70 mg/dL, and is lower than this threshold for at least 2 consecutive sampling times (checking for the presence of two consecutive samples in the hypoglycemic range delays the alarm by 5 min but limits the problem of dealing with false alerts). After 30 min from the first hypo-alert, if the subject is still in the hypoglycemic range, a second alarm is generated and other 15 g of CHO are ingested by the patient. The adopted strategy for alert generation is elementary, since the focus of the present conceptual work is on benefits of considering predicted rather than measured glucose for triggering hypo alarms.

7.4 Results

Figure 7.1 shows graphically results relative to two simulated subjects. For what concerns the upper panel of Figure 7.1 in Scenario 3 (glucose concentration denoted by dashed red line), the nocturnal hypoglycemia is avoided (the lowest glucose concentration results 72 mg/dL) thanks to the generation of the alert (followed by CHO ingestion) at time 03:25. In Scenario 2, the alarm is given at time 04:05 and the hypoglycemic event can be only mitigated: in fact, the subject spends 60 min in the hypoglycemic range, reaching a lowest glycemia of 63 mg/dL (glucose concentration denoted by dotted blue line). Without hypo-alert generation (Scenario 1), the virtual subject experiences a threatening nocturnal hypoglycemia, with a minimum glucose concentration value of 53 mg/dL (glucose concentration denoted by dotted green line), which lasts for 255 min (approximately more than 4 h). The bottom panel of Figure 7.1 shows another representative test subject. In Scenario 3 (glucose concentration denoted by dashed red line), the prediction based alert (followed by CHO ingestion) is generated 20 min ahead in time, however hypoglycemia in this case is not totally avoided, but mitigated: the subject spends 20 min in the hypoglycemic range, (lowest glycemia equals 67 mg/dL). In Scenario 2 and 1 the time spent in the hypoglycemic range is 50 and 55 min, respectively,

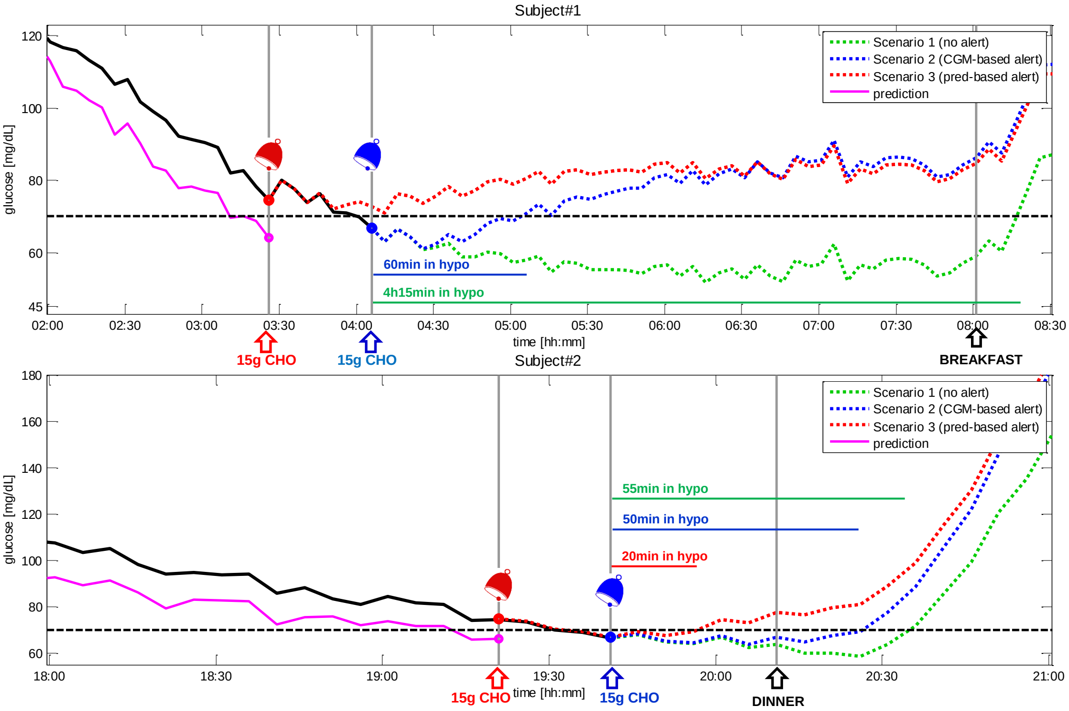


Figure 7.1: Two representative subjects. Continuous black line represents glucose concentration till CGM crosses the hypoglycemic threshold, continuous magenta line identifies the 30 min ahead-of-time glucose prediction till a prediction alert is generated). Scenario 1: (dotted green line), no hypo-alert; Scenario 2: (dotted blue line), CGM-based hypo alert (blue alarm bell). Scenario 3: (dashed red line), prediction-based hypo alert (red alarm bell). Note that the value reported for the prediction at time t is the estimate of the glycemic concentration at time t+PH obtained, at time t itself, by using data available until time t. PH is 30 min.

(lowest glycemia equals 64 mg/dL and 59 mg/dL, respectively), with recovers from the event only after dinner.

Results computed on the 50 virtual subjects dataset considering the entire period of monitoring (54 h) are given in terms of median and 5$^{\text{th}}$ and 95$^{\text{th}}$ percentiles. Table 7.1 and Figure 7.2 summarize number of hypoglycemic events, their duration and total time in hypoglycemic range. In Scenario 1 (unawareness of hypoglycemia, no alerts), patients experience, in median 4, (5$^{\text{th}}$ and 95$^{\text{th}}$ percentiles equal 2-7) hypoglycemic episodes, of median duration of 120 (10-330) min. The total time spent in hypoglycemic range is 9$^{\text{h}}$30$^{\text{min}}$ (4$^{\text{h}}$05$^{\text{min}}$-20$^{\text{h}}$30$^{\text{min}}$) over 54 h of monitoring, which corresponds to 17.7% (7.6%-38.0%) of the total time of monitoring. In Scenario 2, the number of hypoglycemic events is similar to that of Scenario 1. This is expected, since, in Scenario 2 the alarm is CGM-based, thus it is triggered when the subject is, de facto, already in hypoglycemia. However, the severity of hypo-events is significantly mitigated (p<0.01), with a median

Table 7.1: Median results and 5th and 95th percentiles for number of hypoglycemic events and average length of hypo events (min), and time spent in hypo (h and %) during the total period of monitoring (54 h). p-values are computed with the non-parametric Mann-Whitney U test. In each row, the top p-value refers to the comparison with Scenario 1, while the bottom p-value is relative to the comparison with Scenario 2

	number of hypo events				hypo duration [min]				total time in hypo-range [hh:mm]			
	5th	50th	95th	p-val	5th	50th	95th	p-val	5th	50th	95th	p-val
Scenario 1 **(no alert)**	2	4	7		10	120	330		$4^{h}05^{min}$ 7.6%	$9^{h}30^{min}$ 17.7%	$20^{h}30^{min}$ 38.0%	
Scenario 2 **(CGM-based alert)**	2	4	7	p= 0.29	10	40	70	p< 0.01	$1^{h}35^{min}$ 2.9%	$2^{h}35^{min}$ 4.7%	$5^{h}00^{min}$ 9.2%	p< 0.01
Scenario 3 **(pred-based alert)**	0	1	4	p< 0.01 p< 0.01	10	15	45	p< 0.01 p< 0.01	$0^{h}00^{min}$ 0.0%	$0^{h}35^{min}$ 1.2%	$1^{h}35^{min}$ 2.9%	p< 0.01 p< 0.01

Table 7.2: Glucose concentration distributions, LBGI and HBGI calculated in the 3 scenarios (median and 5th and 95th percentiles). p-values computed with the non-parametric Mann-Whitney U test are also reported. In each row, the top p-value refers to the comparison with Scenario 1, while the bottom p-value is relative to the comparison with Scenario 2.

	glucose concentration [mg/dL]				LBGI				HBGI			
	5th	50th	95th	p-val	5th	50th	95th	p-val	5th	50th	95th	p-val
Scenario 1 **(no alert)**	57	110	219		4.0	5.9	13.5		1.9	4.6	12.4	
Scenario 2 **(CGM-based alert)**	70	119	231	p< 0.01	2.6	3.6	4.5	p< 0.01	1.9	5.1	12.8	p= 0.17
Scenario 3 **(pred-based alert)**	74	119	230	p< 0.01 p< 0.01	2.0	2.9	3.4	p< 0.01 p< 0.01	2.2	5.4	12.7	p= 0.19 p= 0.87

duration of 40 (10-70) min, for a total time in hypoglycemia of $2^{h}35^{min}$ ($1^{h}35^{min}$-$5^{h}00^{min}$), corresponding to 4.7% (2.9-9.2%) of the total time. In Scenario 3 patients could potentially avoid, or at least mitigate, many hypoglycemic events by assuming CHO in advance. In fact, the number of hypoglycemic events is 1 (0-4), 75% lower than Scenario 2 and Scenario 1, ($p<0.01$). In addition, in Scenario 3 the median duration of hypo-events is 15 (10-45) min, significantly shorter than in Scenario 2 (-62.5%, $p<0.01$) and Scenario 1 (-87.5%, $p<0.01$). Furthermore, in Scenario 3, the percentage of time spent in hypoglycemic range is 1.2% (0.0-2.9%), corresponding to $0^{h}35^{min}$ ($0^{h}0^{min}$-$1^{h}35^{min}$), with a reduction of 74.5% and 93.2% with respect to Scenario 2 and Scenario 1, respectively.

Figure 7.2 graphically summarizes the results of Table 7.1. In the top panel the histogram of the count of number of hypoglycemic events per patient, during the monitoring period, clearly shows that in Scenario 3 the majority of patients experience only from 0 to 3 hypoglycemic events, while in Scenario 2 and in Scenario 1 the majority of patients experience from 2 to 4 hypoglycemic events. In addition, as shown in the boxplots in the bottom panels of Figure 7.2, hypoglycemia duration (left panel) and total time in

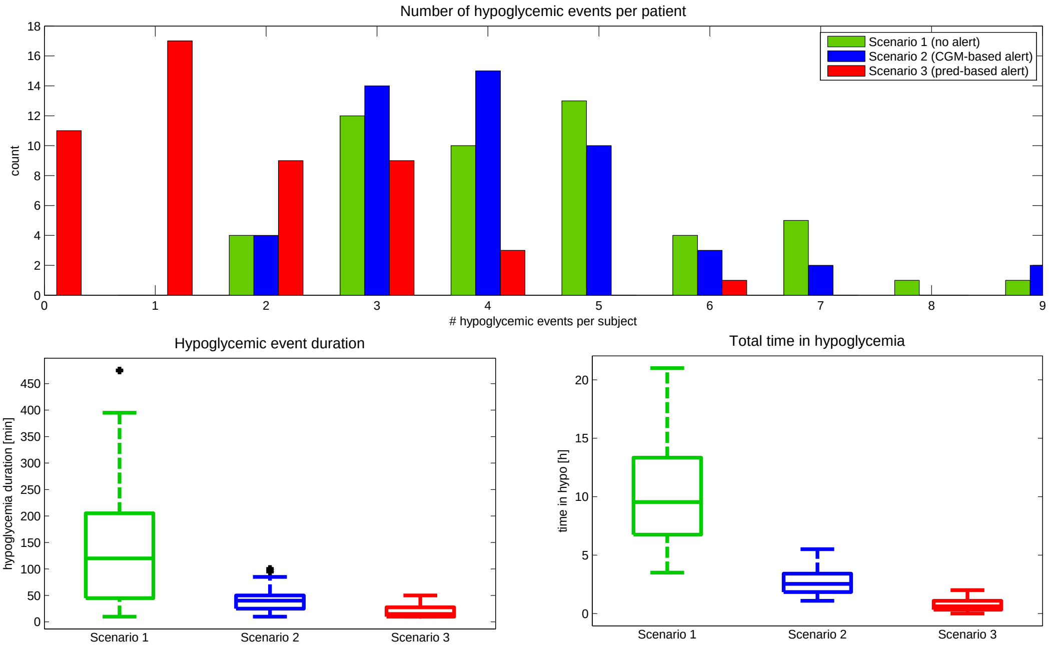


Figure 7.2: Top panel shows the histogram of the number of hypoglycemic episodes per subject, observed during the period of monitoring in the three scenarios. Bottom panels show the boxplot of duration (in min) of hypoglycemic events (left panel) and of total time (in h) spent in hypoglycemic range (right panel) in the three scenarios (green, blue, red for Scenario 1, 2, and 3 respectively).

hypoglycemic range (right panel) considerably decrease moving from Scenario 1 and Scenario 2 to Scenario 3.

Table 7.2 reports, for each scenario, the median and 5th and 95th percentiles of the distribution of glucose concentration values and of LBGI and HBGI in the 50 virtual patients. As expected, the 5th percentile of the distribution of glucose concentration gradually increases in moving from Scenario 1 to Scenario 3. At the same time, the 95th percentile of glucose concentration distribution does not significantly change between Scenario 1 and Scenario 3, indicating that hypo treatments do not significantly increase the highest hyperglycemic value. This is confirmed also by the estimated distribution of glycemic values, in the three scenarios, plotted in the top panel of Figure 7.3. In

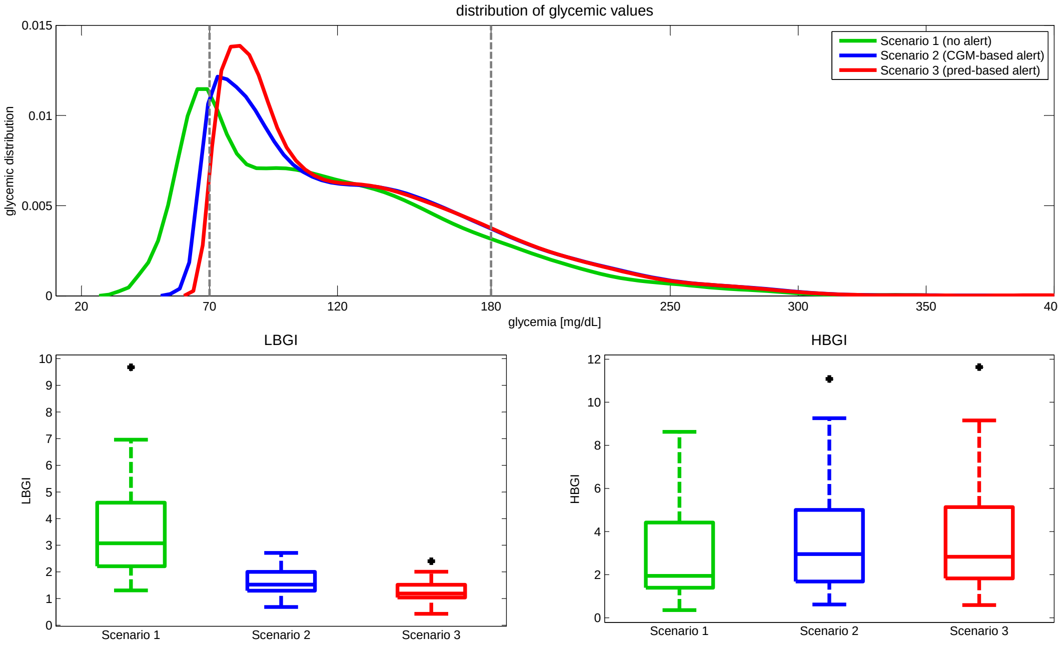


Figure 7.3: Top panel shows the distribution of glycemia in the three scenarios. Bottom panels show the boxplot of LBGI (left) and HBGI (right) in the three scenarios (green, blue, red for Scenario 1, 2, and 3 respectively).

fact the percentage of glycemic values in hypoglycemic range is 19% in Scenario 1, and decreases to 5% in Scenario 2, and to 1% in Scenario 3. The percentage of glycemic values in hyperglycemic range is 14% in Scenario 1, and 17% in both Scenario 2 and Scenario 3. Also the analysis of LBGI and of HBGI, summarized in Table 7.2, confirm that the risk of hypoglycemia is significantly reduced ($p<0.01$), without any increased risk of hyperglycemia ($p>0.5$), in moving from Scenario 1 to Scenario 2 and to Scenario

3. This can be deduced also by visual inspection of the boxplots of the distribution of LBGI and HBGI values in all the 50 subjects (bottom panels of Figure 7.3).

7.5 Robustness: delayed/ absent patient's response to alerts

In the previous Section we simulated the virtual patients responding to alerts in no more than 5 min in both Scenario 2 (CGM-based alerts) and Scenario 3 (prediction-based alerts). However, in real life conditions, subjects could be unable to promptly ingest CHO, or to hear the alarm. For example, in the real case studies documented in [127] young patients did not respond to 34% of the alerts. In [135] patients did not respond to hypoalerts in 4.2% of the cases, and it took them on average 17 min during day-time, and 60 min during night-time, to take countermeasures in case of hypoglycemia.

To assess the effect of delayed/absent responses to alerts, we did additional simulations introducing delays in CHO ingestion and the possibility of no CHO ingestion at all, both in Scenarios 2 and 3. In particular, every time an alarm is triggered (either on the basis of CGM either of prediction), with probability 0.85 the subject ingests 15 g of CHO after a delay uniformly distributed in the time interval (0-30) min, while with probability 0.15 no ingestion of CHO at all occurs. In the case of absent response, if the subject is still in hypoglycemia, a new alert is triggered after 30 min. In the case of delayed response, the new alert is generated 30 min after the subject has effectively ingested CHO, if he/she is still in hypoglycemia. Every time an alert is given, the same procedure just described is repeated (i.e. the patient ignores the alert, either ingests CHO with a certain delay).

Figure 7.4 and Table 7.3 summarize the results in terms of number of hypoglycemic events per patient, their duration and total time in hypoglycemic range during the monitoring period. By comparing results with those of the best case scenario of Figure 7.2 and Table 7.1, we can clearly note a deterioration of the benefits of CGM-based and prediction-based alerts coupled with CHO ingestion. There is still a visible and significant reduction of number of hypoglycemic events and of their duration passing from Scenario 2 to Scenario 3 ($p<0.01$). In Scenario 2, as expected, the number of hypoglycemic events in Table 7.3 cannot worsen with respect to Table 7.1 because in both cases the CGM-based alerts are generated when the subject is already in hypoglycemia. In Scenario 3 the number of hypoglycemic events in Table 7.3 increases with respect to Table 7.1 and a median of 3 (1-5) hypoglycemic events per subject was observed. In particular, moving from Scenario 2 to Scenario 3, the number of hypoglycemic events significantly decreases of the 25% ($p<0.01$). Hypoglycemia duration in Table 7.3 is longer

Table 7.3: As in Table 7.1, but in presence of delays in answering to alerts

	number of hypo events				hypo duration [min]				total time in hypo-range [hh:mm]			
	5th	50th	95th	p-val	5th	50th	95th	p-val	5th	50th	95th	p-val
Scenario 1 **(no alert)**	2	4	7		10	120	330		4^{h}05^{min} 7.6%	9^{h}30^{min} 17.7%	20^{h}30^{min} 38.0%	
Scenario 2 **(CGM-based alert)**	2	4	7	p= 0.57	10	50	149	p< 0.01	1^{h}55^{min} 3.5%	3^{h}35^{min} 6.6%	10^{h}35^{min} 19.6%	p< 0.01
Scenario 3 **(pred-based alert)**	1	3	5	p< 0.01 p< 0.01	10	25	112	p< 0.01 p< 0.01	0^{h}20^{min} 0.6%	1^{h}30^{min} 2.7%	5^{h}45^{min} 10.6%	p< 0.01 p< 0.01

Table 7.4: As in Table 7.2, but in presence of delays in answering to alerts.

	glucose concentration [mg/dL]				LBGI				HBGI			
	5th	50th	95th	p-val	5th	50th	95th	p-val	5th	50th	95th	p-val
Scenario 1 **(no alert)**	57	110	219		4.0	5.9	13.5		1.9	4.6	12.4	
Scenario 2 **(CGM-based alert)**	70	119	231	p< 0.01	2.6	3.6	4.5	p< 0.01	1.9	5.1	12.8	p= 0.17
Scenario 3 **(pred-based alert)**	74	119	230	p< 0.01 p< 0.01	2.0	2.9	3.4	p< 0.01 p< 0.01	2.2	5.4	12.7	p= 0.19 p= 0.87

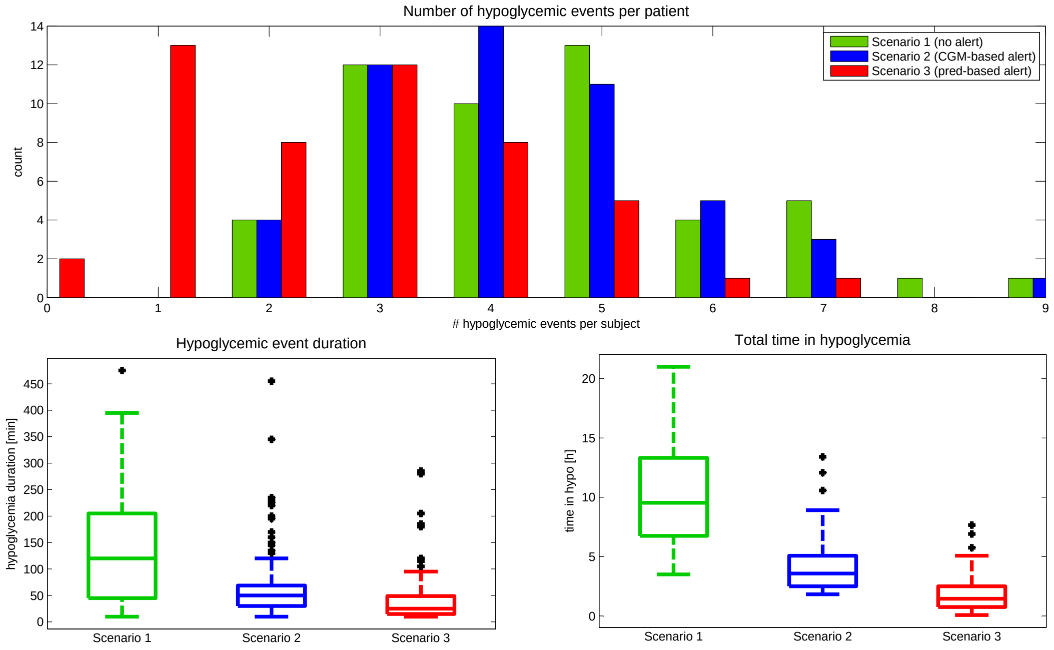


Figure 7.4: As in Figure 7.2, but in presence of delays in answering to alerts.

than in Table 7.1 and is equal to 50 (10-149) min in Scenario 2, and to 25 (10-112) min in Scenario 3. In fact in Scenario 3 the duration of hypoglycemic events decreases of the 50% with respect to Scenario 2 ($p<0.01$). The total time in hypoglycemic range is equal to $3^{h}35^{min}$ ($1^{h}55^{min}$-$10^{h}35^{min}$) in Scenario 2, and to $1^{h}30^{min}$ ($0^{h}20^{min}$-$5^{h}45^{min}$) in Scenario 3, (significant reduction of 59%, $p<0.01$).

The distribution of glycemic values is reported in Table 7.4 and in Figure 7.5, top panel. The shape of the distribution of glucose shows a mild increase in the percentage of values in hypoglycemic range in Scenario 2 and 3, with respect to Table 7.2 and to Figure 7.3, top panel. However, as confirmed by p-values, ingestion of CHO on the basis of predicted glycemia, even if delayed or sometimes ignored, still significantly increases the lowest glycemic concentration experienced by patients. In fact the percentage of glycemic values in hypoglycemic range is equal to 8% in Scenario 2 and to 4% in Scenario 3. Analysis of LBGI and HBGI values, reported in Table 7.4 and in Figure 7.5 (bottom panels), confirms that, moving from Scenario 2 to Scenario 3, a significant decrease of the risk of hypoglycemia occurs, without any parallel increase of the hyperglycemia risk.

In conclusion, delays in responding to hypo alerts, or absence of response, obviously worsen the results obtained in scenarios 2 and 3 in Section 7.4. However, since delayed/absent ingestion of CHO affects in a similar way Scenario 2 and Scenario 3, the

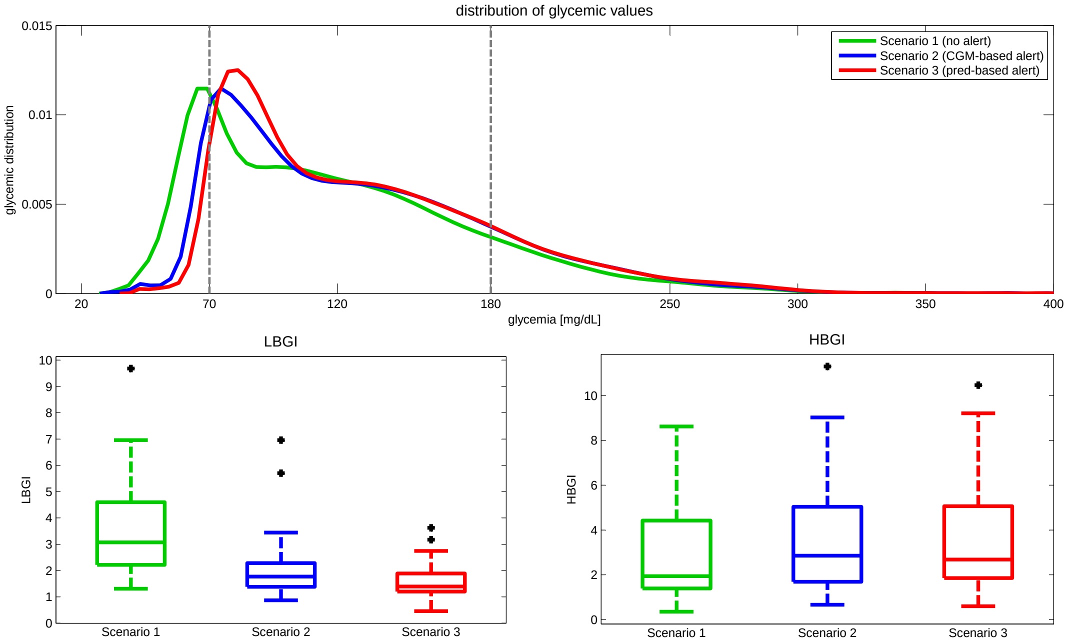


Figure 7.5: As in Figure 7.3, but in presence of delays in answering to alerts.

relative difference between these two scenarios remains significant: in fact, passing from Scenario 2 to Scenario 3, the total time in hypoglycemic range decreases of the 59%, the hypoglycemia duration decreases of the 50% and the number of hypoglycemic events decreases of the 25%.

To conclude, we remark that this simulated analysis cannot capture all the aspects of reality, but any bias equally affects results observed in Scenario 2 (CGM-based alert) as well as in Scenario 3 (prediction-based alert). Thus, on one hand, the absolute results presented in this manuscript could be considered an upper bound of what could be observed in real life. On the other hand, the relative difference between the results obtained in Scenario 2 (CGM-based) and Scenario 3 (prediction-based alert) would probably not change significantly.

7.6 Conclusions and margins for future works

CGM-based short-term glucose prediction algorithms could allow the patient to take appropriate countermeasures to avoid/mitigate hypo-events before their occurrence. By generating data for 50 virtual subjects, in this work we compared occurrence and duration of hypoglycemic events in three scenarios occurring in the same patient, i.e. hypoglycemia unawareness and no countermeasure (Scenario 1), ingestion of 15 g of CHO as glucose concentration measured by CGM sensor crosses the hypoglycemic threshold (Scenario 2) and ingestion of 15 g of CHO as glucose concentration predicted 30-min ahead-of-time crosses the hypoglycemic threshold (Scenario 3). Results show that, by generating hypo-alerts based on prediction, hypoglycemia occurrence could be mitigated and almost totally avoided (in median 1 hypoglycemic event in 54 h of monitoring), and time spent in hypoglycemia could be reduced to 1.2% of the period of monitoring, corresponding to 35 min in 54 h. For what concerns the generation of false alerts, in Scenario 3 we had, on average, 1 false alert every 39 h. However, in this analysis we have not considered the problem of how to generate alerts and have limited ourselves to using a simple threshold comparison strategy. In fact, generating alerts from CGM profiles is a critical issue because of data noise and should be matter of in depth investigation.

The *in silico* environment, although realistic and widely used to preliminarily test new algorithms, has some limitations. For example, diurnal variation of model parameters is not yet taken into account in the model due to lack of quantitative knowledge of this phenomena. Moreover, there is no model for the various factors that influence glycemia in real life, as, for example, stress and illness. These issues have been partially dealt with by simulating a large number of patients (50 synthetic subjects) for a short period of time (54 h), rather than simulating a few patients for longer periods. Furthermore, it is worth underlying that any bias due to simplifications of the *in silico* environment equally affects results observed in Scenario 2 (CGM-based alert) as well as in Scenario 3 (prediction-based alert). Thus, the relative difference between the results obtained in the two scenarios would probably not change significantly.

To complete the analysis of the effective clinical usefulness of the use of prediction for generation of hypoglycemic alerts, our promising results obtained *in silico* should be confirmed in vivo. To this purpose, a clinical trial should start in the first quarter of 2014, in collaboration with Dexcom Inc (San Diego, CA). The protocol design of the study was optimized using both results obtained retrospectively, on data collected by Dexcom during Pivotal trials, and results obtained in simulation, on a population of virtual subjects whose parameters had been optimized to reproduce closely glucose dynamics observed on the real patients participating to the pivotal trials [136]. 30 to 40 T1D should be enrolled in the trial, for a duration of 8 weeks and the primary outcome should be a significant reduction in the number of severe hypoglycemic events when alerts and relative therapy are triggered on the basis of the predicted glucose profile, obtained with the strategy jointly developed by our research group and Dexcom [137].

8
Conclusions

In diabetes management, tight monitoring of glucose concentration is essential for limiting short and long term complications due to hypo- and hyperglycemic events. Short-time prediction (30-60 min ahead in time) of glucose concentration might improve T1D therapy by allowing the patient to tune the therapy on the basis of future, instead of current, glycemia, possibly avoiding, or at least mitigating, critical events. Accurate prediction of glucose concentration, in every glycemic range, is important in closed loop applications based on model predictive control, and also in open loop therapy, to allow diabetic subjects to anticipate therapeutic decisions, based on expected future glycemia and planned daily life activities. Not least important is the use of prediction in open loop therapy for generating preventive alerts, when glucose concentration is expected to cross pre-set risky thresholds in the short term, potentially allowing diabetics to avoid the majority of critical hypo and hyperglycemic events. Most of the prediction methods proposed in the literature in the last decade are based on models that use only CGM history as input. Recently, various attempts of using also insulin, CHO and PA information have been proposed, mainly by incorporating these additional inputs in simple linear ARX and ARMAX models. However, exploiting these supplementary sources of information is not easy since their effects are affected by physiological delays and inter- and intra-individual variability is high. NN based models appear to be suitable candidates to forecast future glucose concentration. Indeed NNs are intrinsically non-linear, can learn complex functions and extract, relatively easily, relevant information from input signals with different characteristics and nature. Despite these appealing features, NNs have been scarcely utilized, so far, for prediction of glucose concentration.

Starting from the observation that feedforward NN described in the literature [66, 77] did not significantly outperform linear time series models, we firstly proposed a paradigm composed by a feedforward NN in parallel with a linear model [102] so that the nonlinear behaviour of the NN could be better exploited. Inputs of this model are signals derived from glucose concentration, measure by the CGM sensor and signals derived from information on timing and quantity of ingested CHO, simulated with a physiological model of oral glucose absorption. The proposed architecture outperformed the NN of [66] and the AR model of [59] both on simulated and real data. Moreover, we proved its robustness against errors in the estimation of timing and CHO content of meals. Afterwards, we demonstrated, using the same input signals, that a different architecture, i.e. a jump NN, which is able to separately deal with linear and nonlinear relationships between inputs and output, had performance statistically comparable with the previously proposed model, despite its simpler structure [107]. This is a major novelty, since, to the best of our knowledge, jump NNs had never been proposed before for glucose concentration prediction. In addition, the simplicity of the chosen structure, once trained, renders it potentially implementable also in a CGM device, where computational power is limited and shared between several algorithms. Finally, we incorporated among the inputs of the jump NN a signal derived from information on timing and quantity of injected insulin boluses, preprocessed with a physiological model of insulin absorption and sensitivity. Our analysis assessed, comparing NN models with different combination of input signals, how much prediction was effectively improved when information on CHO ingestion and/ or insulin injection was added to information on CGM and included among the NN inputs. We showed that exogenous inputs relative to CHO and insulin significantly improve prediction in the 2 h time window that follows the ingestion of CHO and the injection of insulin, while their benefits are no visible, for example, during night periods [109, 110]. This fact, previously unnoticed in the glucose prediction literature, could be justified, physiologically, by the fact that CHO and insulin effects are particularly evident for about 2 h and become scarcely relevant after a longer time interval.

A future development of our prediction paradigm could be the inclusion, among the inputs of the NN, of signals relative to PA and energy expenditure. Indeed we demonstrated [117] that even mild PA is significantly correlated, in the short-term, with changes in glucose dynamics. These results suggest that the NN ability of accurately predicting glucose time course would benefit by the inclusion of this additional source of information. Even if, so far, some attempts of exploiting signals related to PA for prediction of glucose concentration have been made, this field is still rather unexplored and there are no widely accepted models of PA effects on glucose time course. How to adequately preprocess and utilize this information is a challenging issue that is worth future in depth investigation.

One of the natural applications of short-time prediction is the generation of preventive hypoglycemic alerts. In the literature, some contributions assessed, on real data of hospitalized subjects, the reduction and mitigation of induced hypoglycemia obtained when therapeutic actions were triggered on the basis of prediction of glucose concentration. However, these analysis could not be exhaustive, since, on real data, once an action is taken, there is no possibility of knowing what would have happened if different decisions were made. To overcome these limitations we used the *in silico* environment of [73], which is widely accepted to preliminarily test new algorithms, given its high realism. In particular, we quantified how much hypoglycemia could be reduced if hypoglycemic alerts and relative therapy (ingestion of CHO) were triggered based on prediction, instead of CGM [130]. Results showed a significant reduction of hypoglycemia and an improvement of the management of glucose concentration, when alerts were generated based on prediction. Furthermore, we demonstrated that hypoglycemia was reduced even if the T1D virtual subject responded with a certain delay to the alerts, or even ignored some of them. Such a comprehensive analysis and comparison between alternative scenarios had never been performed and, to confirm our promising and encouraging results in vivo, an extensive clinical study should start in the first quarter of 2014, in collaboration with Dexcom Inc. (San Diego, CA). Indeed our research group optimized, using simulation and analysing retrospectively real data, the design and protocol of a clinical trial [136]. The aim of the study is demonstrating, in vivo, that prediction based [137] hypoglycemic alerts, incorporated in a research prototype of the Dexcom G4 PLATINUM CGM sensor [31], allow a significant reduction of the natural occurrence of hypoglycemia in every-day life conditions.

Further possible future works include the investigation of specifically formulated objective functions that quantifies the goodness of glucose concentration prediction, e.g. [100], for optimizing the NN weights and the design of the NN for predicting specifically hypo- and hyperglycemia, instead of the entire range of glucose values, as done e.g. in [65] with different models, transforming the prediction problem in a classification issue.

A
Glucose-insulin meal model

The mathematical model providing the base for the *in silico* subjects of the simulation environment is the glucose-insulin meal model of Dalla Man et al. [103, 104, 138], whose equations are reported in this Appendix. In particular, the model has 26 free parameters, whose joint distribution has been computed from real individuals' data. The simulator allows generating a large cohort of virtual "subjects", characterized by key metabolic parameters spanning the variability observed in the population of people with T1D. The model was shown to represent adequate glucose fluctuations in T1D observed during meal challenges and was thus accepted by FDA as a substitute to animal trials in preclinical testing of closed-loop control strategies [73, 84]. For these reasons, the simulator has been widely used to preliminarily test new algorithms, given its sufficient realism, e.g. [124, 139–141].

A.1 Glucose absorption model

The rate of appearance of glucose in plasma is obtained through the physiological model of glucose intestinal absorption reported in [103] and graphically shown in Figure A.1. The model describes the glucose transit through the stomach and intestine by representing the stomach with two compartments, (one for solid and one for triturated phase), while the gut is described with a single compartment. The differential equations system that

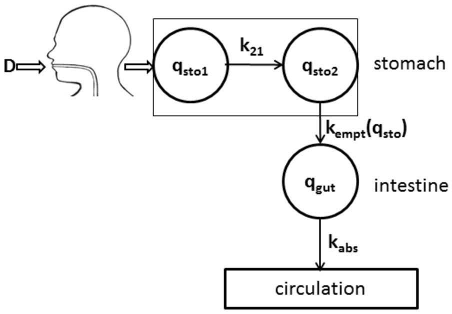


Figure A.1: Glucose absorption model, which assumes two compartments for the stomach (one for the liquid and one for the solid phase) a gastric empting rate (k_{empt}) dependent on the total amount of glucose in the stomach (q_{sto}), a single compartment for the intestine (q_{gut}) and a constant rate of intestinal absorption (k_{abs}).

characterizes the three compartment model is

$$\begin{cases} q_{sto}(t) &= q_{sto1}(t) + q_{sto2}(t) & q_{sto}(0) &= 0; \\ \dot{q}_{sto1}(t) &= -k_{21}\, q_{sto1}(t) + D\,\delta(t) & q_{sto1}(0) &= 0; \\ \dot{q}_{sto2}(t) &= -k_{empt}(q_{sto})\, q_{sto2}(t) + k_{21}\, q_{sto1}(t) & q_{sto2}(0) &= 0; \\ \dot{q}_{gut}(t) &= -k_{abs}\, q_{gut}(t) + k_{empt}(q_{sto})\, q_{sto2}(t) & q_{gut}(0) &= 0; \\ Ra_G(t) &= \frac{f\, k_{abs}\, q_{gut}(t)}{m_{BW}} & Ra_G(0) &= 0. \end{cases} \tag{A.1}$$

where q_{sto1} [mg] and q_{sto2} [mg] represent the amount of CHO in the stomach (solid and liquid phase), D [mg] is the amount of ingested CHO, q_{gut} [mg] is the CHO mass in the intestine, k_{21} [min^{-1}] represents the rate of grinding, $k_{empt}(q_{sto})$ [min^{-1}] is the rate constant of gastric emptying, which is a nonlinear function of q_{sto}, as reported in equation (A.2), k_{abs} [min^{-1}] is the rate constant of intestinal absorption, f the fraction of the intestinal flux that appears in plasma, m_{BW} the body weight, Ra_G [mg/kg/min] is the rate of appearance of glucose in plasma.

$$k_{empt}(q_{sto}) = k_{min} + k\{\tanh[\alpha(q_{sto} - b\,D)] - \tanh[\beta(q_{sto} - c\,D)] + 2\} \tag{A.2}$$

where

$$k = \frac{k_{max} - k_{min}}{2} \tag{A.3}$$

$$\alpha = \frac{5}{2D(1-b)} \tag{A.4}$$

$$\beta = \frac{5}{2Dc} \tag{A.5}$$

with k_{max}, k_{min}, b and c parameters.

In Chapters 3, 4 and 5 Ra_G was simulated using the population parameter values estimated in [104]. The model was validated in [103], and parameters were estimated with coefficients of variation ranging from 6% to 46%. Parameters k_{gri}=0.0558 min^{-1} and c=0.00236 mg^{-1} were fixed, while the remaining parameters were estimated: k_{max}=0.0558 min^{-1}, k_{min}=0.008 min^{-1}, k_{abs}=0.057 min^{-1}, f=0.9 (adim) and b=0.82 (adim).

A.2 Insulin absorption model

The physiological model used for describing subcutaneous insulin kinetics is described in [138] and graphically shown in Figure A.2. Mathematically, it is represented by the following system of equations:

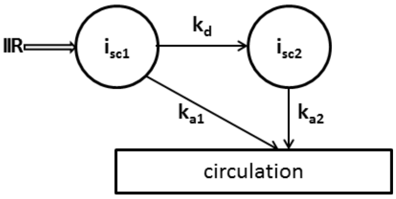


Figure A.2: Insulin absorption model, which assumes a compartment for nonmonomeric insulin and a compartment for monomeric insulin.

$$\begin{cases} \dot{i}_{sc1}(t) &= -(k_d + k_{a1}) i_{sc1}(t) + IIR(t) & i_{sc1}(0) = i_{sc1ss}; \\ \dot{i}_{sc2}(t) &= k_d i_{sc1}(t) - k_{a2} i_{sc2}(t) & i_{sc2}(0) = i_{sc2ss}; \\ Ra_I(t) &= k_{a1} i_{sc1}(t) + k_{a2} i_{sc2}(t) & Ra_I(0) = 0. \end{cases} \tag{A.6}$$

where i_{sc1} is the amount of nonmonomeric insulin in the subcutaneous space, i_{sc2} is the amount of monomeric insulin in the subcutaneous space, $IIR(t)$ (pmol/kg/min) is the exogenous insulin infusion rate, k_d (min^{-1}) is the rate constant of insulin dissociation, and k_{a1} and k_{a2} are the rate constants of non-monomeric and monomeric insulin absorption, respectively. For what concerns parameters value, in analogy with the approach adopted with the glucose absorption model we used the population parameters estimated in [104], whose values are: k_d=0.0164 min^{-1}, k_{a1}=0.0018 min^{-1}, k_{a2} = 0.0182 min^{-1}.

B
Real database (from the DIAdvisor project)

The DIAdvisor was a large-scale integrating project, running in the period 2008-2012, aiming at the development of a device which uses CGM and vital signs to optimize T1D therapy, funded under the Framework Program 7 (FP7) by the European Commission.

Data used in this thesis were collected in two different data acquisition sessions of the DIAdvisor project. Notably, two different CGM sensors were used: during the first data acquisition session, (the DAQ trial), the FreeStyle Navigator device was used, which has a sampling time of 1 min. During the following data acquisition sessions, the Dexcom SEVEN PLUS sensor was utilized, whose sampling time is equal to 5 min. The reasons for adopting a different CGM device were the greater accuracy of the SEVEN PLUS, compared to the FreeStyle Navigator, the easier procedure for downloading data and last but not least important, the fact that the majority of commercial CGM devices had implemented a sampling time of 5 min. This was relevant, since the purpose of the DIAdvisor platform was to be independent from a particular device, thus developing it on CGM data measured every 5 min made it, potentially, more portable.

In total 60 patients, (male or female), diagnosed with T1D or T2D were enrolled into the study. Inclusion criteria comprised being between 18 and 70 years old, having been diagnosed with T1D or T2D (according to WHO criteria [142]) for at least one year prior to study entry and following a basal-bolus insulin therapy using an external pump or multiple-daily injections. A standard GMS mobile phone with camera was used by the study participants for capturing pictures of all the food and beverages they ingested. For each picture, the timing and the estimated amount of CHO were extracted and added to the data relative to meals. Information on insulin therapy (timing, dose and type) was manually recorded by the patients.

C
Assessment metrics

The quality of prediction obtained with the proposed algorithms is quantitatively assessed by computing three metrics commonly used in the glucose prediction literature. As discussed in [79], using different metrics is necessary for a comprehensive evaluation of glucose prediction performance. Let us define $y(t)$ measured signal, $\hat{y}(t|t-T)$ predicted signal relative to time t, obtained using data available until time $t-T$, N length of $y(t)$, T_s sampling time of $y(t)$, PH Prediction Horizon in min, $T = PH/T_s$, i.e. T prediction horizon in number of steps. The indexes we use are:

- The RMSE (mg/dL) between the predicted time-series and the original glucose time-series measured by the CGM sensor, calculated as

$$\text{RMSE} = \sqrt{\left(\frac{1}{N}\sum_{k=1}^{N}\left(\hat{y}(k|k-T) - y(k)\right)^2\right)} \tag{C.1}$$

- The TG

$$\text{TG} = \text{PH} - \text{delay} \tag{C.2}$$

  and the normalized TG

$$\text{TG}_{\text{norm}} = \frac{\text{PH} - \text{delay}}{\text{PH}} \tag{C.3}$$

with the delay quantified as the temporal shift that minimizes the distance between $y(t)$ and $\hat{y}(t|t-T)$

$$\text{delay} = \underset{k\in[0,\,T]}{\arg\min}\left\{\frac{1}{N-T}\sum_{i=1}^{N-T}\left(\hat{y}(i+k|i+k-T)-y(i)\right)^2\right\}T_s \tag{C.4}$$

- The ESOD (i.e. the sum of the squared second order differences) of the predicted time series, normalized by the ESOD of the target time series [59]

$$\text{ESOD}_{\text{norm}} = \frac{\text{ESOD}(\hat{y})}{\text{ESOD}(y)} \tag{C.5}$$

where

$$\text{ESOD}(\hat{y}) = \frac{1}{T_s^4}\sum_{k=3}^{N}\left(\hat{y}(k|k-T)-2\hat{y}(k-1|k-1-T)+\hat{y}(k-2|k-2-T)\right)^2 \tag{C.6}$$

$$\text{ESOD}(y) = \frac{1}{T_s^4}\sum_{k=3}^{N}\left(y(k)-2y(k-1)+y(k-2)\right)^2 \tag{C.7}$$

The RMSE is a widely used metric in the CGM literature, e.g. [59,60,66,67,143], however it has some limitations: it does not penalize spurious oscillations around the target and it is unable to penalize differently under- and over- estimation of the target.

TG is one of the most important indexes from a practical perspective, since it quantifies the average anticipation with which events could be, in theory, detected and can have, thus, a clinical value. The higher the TG and the closer to 1 the TG_{norm}, the better the prediction, since the patient could decide therapeutic actions ahead in time and, likely, avoid critical events. Notably, the definition of the delay given in equation (C.4) is consistent with those of [58,105].

$\text{ESOD}_{\text{norm}}$ reflects how (possibly spurious) oscillations are amplified in the predicted time series. Thus it roughly quantifies the risk of generating false hypo/hyper alerts. The closer to 1, the better the predicted time series.

Bibliography

[1] G. Danaei, M.M. Finucane, Y. Lu, G.M. Singh, M.J. Cowan, C.J. Paciorek, et al. National, regional, and global trends in fasting plasma glucose and diabetes prevalence since 1980: systematic analysis of health examination surveys and epidemiological studies with 370 country-years and 2.7 million participants. *Lancet*, 378(9785):31–40, 2011.

[2] American Diabetes Association. Economic costs of diabetes in the U.S. in 2012. *Diabetes care*, 36(4):1033–1046, 2013.

[3] European Commission. Estimates of cost of diabetes per year in the European Union and in other European countries. http://ec.europa.eu/health/major_chronic_diseases/docs/idf_cost_2011.pdf, April 2012. Accessed 22 January 2014.

[4] P. Zimmet, K.G. Alberti, and J. Shaw. Global and societal implications of the diabetes epidemic. *Nature*, 414(6865):782–787, 2001.

[5] A.R. Saltiel and R. Kahn. Insulin signalling and the regulation of glucose and lipid metabolism. *Nature*, 414:799–806, 2001.

[6] S. Melmed, K.S. Polonsky, Larsen P.R., and H.M. Kronenberg. *Williams Textbook of Endocrinology: Expert Consult*. Elsevier Health Sciences, W B Saunders Co, 12th edition, 2011.

[7] World Health Organization. Diabetes. http://www.who.int/mediacentre/factsheets/fs312/en/index.html, November 2013. Accessed 22 January 2014.

[8] American Diabetes Association. Type 1 diabetes. http://www.diabetes.org/diabetes-basics/type-1/, n.a. Accessed 22 January 2014.

[9] Centers for Disease Control and Prevention. National diabetes fact sheet: national estimates and general information on diabetes and prediabetes in the United States.

http://www.cdc.gov/diabetes/pubs/pdf/ndfs_2011.pdf, 2011. Accessed 22 January 2014.

[10] Mayo Clinic. Type 2 diabetes. http://www.mayoclinic.com/health/type-2-diabetes/DS00585, January 2013. Accessed 22 January 2014.

[11] M.A. Powers. *Handbook of diabetes medical nutrition therapy*. Jones and Bartlett Learning, 1996.

[12] P.E. Cryer. Hypoglycemia, functional brain failure, and brain death. *J Clin Invest*, 117(4):868–870, 2007.

[13] I.M. Stratton, A.I. Adler, H.A.W. Neil, D.R. Matthews, S.E. Manley, C.A. Cull, D. Hadden, R.C. Turner, and R.R. Holman. Association of glycaemia with macrovascular and microvascular complications of type 2 diabetes: prospective observational study. *Brit Med J*, 321(7258):405–412, 2000.

[14] L. Heinemann and D. Boecker. Lancing: Quo vadis? *J Diabetes Sci Technol*, 5(4):966–981, 2011.

[15] OneTouch. The OneTouch®Ultra Mini. http://www.onetouch.com/onetouch-ultramini, January 2014. Accessed 22 January 2014.

[16] Roche Diagnostics. Accu-Chek®Aviva Plus System. https://www.accu-chek.com/us/glucose-meters/aviva.html, November 2013. Accessed 22 January 2014.

[17] Sanofi Diabetes. iBGStar®Blood Glucose Meter. http://www.bgstar.com/web/ibgstar?bp_geordir=true, December 2011. Accessed 22 January 2014.

[18] G.V. McGarraugh, W.L. Clarke, and B.P. Kovatchev. Comparison of the clinical information provided by the FreeStyle Navigator continuous interstitial glucose monitor versus traditional blood glucose readings. *Diabetes Technol Ther*, 12:365–371, 2010.

[19] F. Ricci, D. Moscone, and G. Palleschi. Ex vivo continuous glucose monitoring with microdialysis technique: the example of GlucoDay. *IEEE Sens J*, 8:63–70, 2008.

[20] G. McGarraugh. The chemistry of commercial continuous glucose monitors. *Diabetes Technol Ther*, 11:S17–S24, 2009.

[21] C.M. Girardin, C. Huot, M. Gonthier, and E. Delvin. Continuous glucose monitoring: a review of biochemical perspectives and clinical use in type 1 diabetes. *Clin Biochem*, 42:136–142, 2009.

[22] G. Sparacino, M. Zanon, A. Facchinetti, C. Zecchin, A. Maran, and C. Cobelli. Italian contributions to the development of continuous glucose monitoring sensors for diabetes management. *Sensors*, 12(10):13753–13780, 2012.

[23] J. Wang. Electrochemical glucose biosensors. *Chem Rev*, 108:814–825, 2008.

[24] H.B. Ginsberg. The current environment of CGM technologies. *J Diabetes Sci Technol*, 1:117–121, 2007.

[25] R.L. Weistein, S.L. Schwartz, R.L. Brazg, J.R. Bugler, T.A. Peyser, and G.V. McGarraugh. Accuracy of the 5-day FreeStyle Navigator continuous glucose monitoring system. *Diabetes Care*, 30(5):1125–1130, 2007.

[26] D.M. Wilson, R.W. Beck, W.V. Tamborlane, M.J. Dontchev, C. Kollman, P. Chase, L.A. Fox, K.J. Ruedy, E. Tsalikian, and S.A. Weinzimer. The accuracy of the FreeStyle Navigator continuous glucose monitoring system in children with type 1 diabetes. *Diabetes Care*, 30(1):59–64, 2007.

[27] U.S. Food and Drug Administration. Medical devices: FreeStyle Navigator®continuous glucose monitoring system - p050020. http://www.fda.gov/medicaldevices/productsandmedicalprocedures/deviceapprovalsandclearances/recently-approveddevices/ucm074293.htm, September 2013. Accessed 22 January 2014.

[28] Dexcom. The Dexcom SEVEN®PLUS. http://www.dexcom.com/seven-plus, n.a. Accessed 22 January 2014.

[29] S. Garg, H. Zisser, S. Schwartz, T. Bailey, R. Kaplan, S. Ellis, and L. Jovanvic. Improvement in glycemic excursion with a transcutaneous, real-time continuous glucose sensor. *Diabetes Care*, 29(1):44–50, 2006.

[30] Dexcom. Dexcom G4 PLATINUM. http://www.dexcom.com/dexcom-g4-platinum, n.a. Accessed 22 January 2014.

[31] A. Garcia, A.L. Rack-Gomer, N.C. Bhavaraju, H. Hampapuram, A. Kamath, T. Peyser, A. Facchinetti, C. Zecchin, G. Sparacino, and C. Cobelli. Dexcom G4AP: An advanced continuous glucose monitor for the artificial pancreas. *J Diabetes Sci Technol*, 7(6):1436–1445, 2013.

[32] Medtronic. Guardian®REAL-Time CGM System. http://www.medtronicdiabetes.com/treatment-and-products/guardian-real-time-cgm-system, n.a. Accessed 22 January 2014.

[33] J. Mastrototaro and S. Lee. The integrated MiniMed Paradigm real-time insulin pump and glucose monitoring system: Implications for improved patient outcomes. *Diabetes Technol Ther*, 11(1):S37–S44, 2009.

[34] T. Kubiak, B. Woerle, B. Kuhr, I. Nied, G. Glaesner, N. Hermanns, B. Kulzer, and T. Haak. Microdialysis-based 48-hour continuous glucose monitoring with GlucoDay: clinical performance and patient's acceptance. *Diabetes Technol Ther*, 8(5):570–575, 2006.

[35] A. Menarini Diagnostics. System description. http://www.menarinidiag.co.uk/Products/continuous_glucose_monitoring/system_description, n.a. Accessed 22 January 2014.

[36] F. Valgimigli, F. Lucarelli, C. Scuffi, S. Morandi, and I. Sposato. Evaluating the clinical accuracy of GlucoMen® Day: a novel microdialysis-based continuous glucose monitor. *J Diabetes Sci Technol*, 4(5):1182–1192, 2010.

[37] F. Ricci, F. Caprio, A. Poscia, F. Valgimigli, D. Messeri, E. Lepori, G. Dall'Oglio, G. Palleschi, and D. Moscone. Toward continuous glucose monitoring with planar modified biosensors and microdialysis: Study of temperature, oxygen dependence and in vivo experiment. *Biosens Bioelectron*, 22(9):2032–2039, 2007.

[38] F. Lucarelli, F. Ricci, F. Caprio, F. Valgimigli, C. Scuffi, D. Moscone, and G. Palleschi. GlucoMen Day continuous glucose monitoring system: A screening for enzymatic and electrochemical interferents. *J Diabetes Sci Technol*, 6(5):1172–1181, 2012.

[39] C. Kapitza, V. Lodwig, K. Obermaier, K.J.C. Wientjes, K. Hoogenberg, K. Jungheim, and L. Heinemann. Continuous glucose monitoring: reliable measurements for up to 4 days with the SCGM1 system. *Diabetes Technol Ther*, 5(4):609–614, 2003.

[40] K. Jungheim, K.J. Wientjes, L. Heinemann, V. Lodwig, T. Koschinsky, and A.J. Schoonen. Subcutaneous continuous glucose monitoring: feasibility of a new microdialysis-based glucose sensor system. *Diabetes Care*, 24:1696–1697, 2001.

[41] Echo Therapeutics. Needle-free monitoring and drug delivery. http://www.echotx.com/, n.a. Accessed 22 January 2014.

[42] S. Vaddiraju, D.J. Burgess, I. Tomazos, F.C. Jain, and F. Papadimitrakopoulos. Technologies for continuous glucose monitoring: Current problems and future promises. *J Diabetes Sci Technol*, 4(6):1540–1562, 2010.

[43] D. Rodbard. New and improved methods to characterize glycemic variability using continuous glucose monitoring. *Diabetes Technol Ther*, 11:551–565, 2009.

[44] W.V. Tamborlane, R.W. Beck, B.W. Bode, et al. Continuous glucose monitoring and intensive treatment of type 1 diabetes. *N Engl J Med*, 359(10):1464–1476, 2008.

[45] T. Battelino, M. Phillip, N. Bratina, R. Nimri, P. Oskarsson, and J. Bolinder. Effect of continuous glucose monitoring on hypoglycemia in type 1 diabetes. *Diabetes Care*, 34(4):795–800, 2011.

[46] D. Deiss, J. Bolinder, J.P. Riveline, T. Battellino, E. Bosi, N. Tubiana-Rufi, D. Kerr, and M. Philip. Improved glycemic control in poorly controlled patients with type 1 diabetes using real-time continuous glucose monitoring. *Diabetes Care*, 29(12):2730–2732, 2006.

[47] R.M. Bergenstal, W.V. Tamborlane, A. Ahmann, et al. Effectiveness of sensor-augmented insulin-pump therapy in type 1 diabetes. *N Engl J Med*, 363:311–320, 2010.

[48] E. Cengiz, J.L. Sherr, S.A. Weinzimer, and W.V. Tamborlane. New-generation diabetes management: glucose sensor-augmented insulin pump therapy. *Expert Rev Med Devices*, (8):449–458, 2011.

[49] G. Sparacino, A. Facchinetti, A. Maran, and C. Cobelli. Continuous glucose monitoring time series and hypo/hyperglycemia prevention: requirements, methods, open problems. *Curr Diabetes Rev*, 4:181–192, 2008.

[50] G. Sparacino, A. Facchinetti, and C. Cobelli. “smart” continuous glucose monitoring sensors: On-line signal processing issues. *Sensors*, 10(7):6751–6772, 2010.

[51] B.W. Bequette. Continuous glucose monitoring: real-time algorithms for calibration, filtering, and alarms. *J Diabetes Sci Technol*, 4(2):404–418, 2010.

[52] A. Facchinetti, G. Sparacino, S. Guerra, Y.M. Luijf, J.H. Devries, J.K. Mader, M. Ellmerer, C. Benesch, L. Heinemann, D. Bruttomesso, A. Avogaro, and C. Cobelli. Real-time improvement of continuous glucose-monitoring accuracy: The smart sensor concept. *Diabetes Care*, 36(4):793–800, 2013.

[53] C. Cobelli, E. Renard, and B. Kovatchev. Artificial pancreas: past, present, future. *Diabetes*, 60(11):2672–2682, 2011.

[54] H. Thabit and R. Hovorka. Closed-loop insulin delivery in type 1 diabetes. *Endocrinol Metab Clin North Am*, 41(1):105–117, 2012.

[55] J. Reifman, S. Rajaraman, A. Gribok, and W.K. Ward. Predictive monitoring for improved management of glucose levels. *J Diabetes Sci Technol*, 1(4):478–486, 2007.

[56] AmyT. iSense and their “glycemic signature”. http://www.diabetesmine.com/2008/10/isense-and-their-glycemic-signature.html, October 2008. Accessed 22 January 2014.

[57] 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*, 10(5):622–628, 1987.

[58] A. Gani, A.V. Gribok, J. Rajaraman, and J. Reifman. Predicting subcutaneous glucose concentration in humans: Data-driven glucose modeling. *IEEE Trans Biomed Eng*, 56(2):246–254, 2009.

[59] 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 Trans Biomed Eng*, 54(5):931–937, 2007.

[60] M. Eren-Oruklu, A. Cinar, L. Quinn, and D. Smith. Estimation of the future glucose concentrations with subject specific recursive linear models. *Diabetes Technol Ther*, 11(4):243—253, 2009.

[61] C.C. Palerm, J.P. Willis, J. Desemone, and B.W. Bequette. Hypoglycemia prediction and detection using optimal estimation. *Diabetes Technol Ther*, 7(1):3–14, 2005.

[62] C.C. Palerm and W. Bequette. Hypoglycemia detection and prediction using continuous glucose monitoring - a study on hypoglycemic clamp data. *J Diabetes Sci Technol*, 1:624–629, 2007.

[63] V. Naumova, S.V. Pereverzyev, and S. Sivananthan. A meta-learning approach to the regularized learning-case study: Blood glucose prediction. *Neural Networks*, 33(9):181–193, 2012.

[64] S. Sivananthan, V. Naumova, C. Dalla Man, A. Facchinetti, E. Renard, C. Cobelli, and S.V. Pereverzyev. Assessment of blood glucose predictors: The prediction-error grid analysis. *Diabetes Technol Ther*, 13(8):787–796, 2011.

[65] 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*, 33(6):1249–1254, 2010.

[66] C. Pérez-Gandía, A. Facchinetti, G. Sparacino, C. Cobelli, E.J. Gómez, M. Rigla, A. de Leiva, and M.E. Hernando. Artificial neural network algorithm for on-line glucose prediction from continuous glucose monitoring. *Diabetes Technol Ther*, 12(1):81–88, 2010.

[67] D.A. Finan, F.J. Doyle, 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. *J Diabetes Sci Technol*, 5(3):1192–1202, 2009.

[68] G. Castillo-Estrada, L. del Re, and E. Renard. Nonlinear gain in online prediction of blood glucose profile in type 1 diabetic patients. In *49th IEEE Conference on Decision and Control (CDC)*, pages 1668–1673, Hilton Atlanta Hotel, Atlanta, GA, USA, Dec 15-17, 2010.

[69] W.L. Clarke. The original Clarke error grid analysis (EGA). *Diabetes Technol Ther*, 7(5):776–779, 2005.

[70] M. Eren-Oruklu, A. Cinar, D.K. Rollins, and L. Quinn. Adaptive system identification for estimating future glucose concentrations and hypoglycemia alarms. *Automatica*, 48(8):1892–1897, 2012.

[71] K. Turksoy, E.S. Bayrak, E. Littlejohn, L. Quinn, and A. Cinar. Hypoglycemia early alarm systems based on multivariable models. *Ind Eng Chem Res*, 52:12329–12336, 2013.

[72] E. Daskalaki, A. Prountzou, P. Diem, and S.G. Mougiakakou. Real-time adaptive models for the personalized prediction of glycemic profile in type 1 diabetes patients. *Diabetes Technol Ther*, 14(2):168–174, 2012.

[73] B.P. Kovatchev, M.D. Breton, C. Cobelli, and C. Dalla Man. Method, system and computer simulation environment for testing of monitoring and control strategies in diabetes. Patent WO/2008/157781, 2008.

[74] C. Zhao, E. Dassau, L. Jovanovič, H.C. Zisser, F.J. Doyle III, and D.E. Seborg. Predicting subcutaneous glucose concentration using a latent-variable-based statistical method for type 1 diabetes mellitus. *J Diabetes Sci Technol*, 6(3):617–633, 2012.

[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 *34th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 2889–2892, Hilton San Diego Bayfront, San Diego, CA, USA, Aug 28-Sep 1, 2012.

[76] E.I. Georga, V.C. Protopappas, D. Ardigo, M. Marina, I. Zavaroni, D. Polyzos, and D. Fotiadis. Multivariate prediction of subcutaneous glucose concentration in type 1 diabetes patients based on support vector regression. *IEEE J Biomed Health Inform*, 17(1):71–81, 2013.

[77] S.M. Pappada, B.D. Cameron, P.M. Rosman, R.E. Bourey, T.J. Papadimos, W. Oloruntu, and M.J. Borst. Neural network-based real-time prediction of glucose in patients with insulin-dependent diabetes. *Diabetes Technol Ther*, 13(2):135–141, 2011.

[78] Y. Wang, X. Wu, and X. Mo. A novel adaptive-weighted-average framework for blood glucose prediction. *Diabetes Technol Ther*, 15(10):1–10, 2013.

[79] A. Facchinetti, G. Sparacino, E. Trifoglio, and C. Cobelli. A new index to optimally design and compare CGM glucose prediction algorithms. *Diabetes Tech Ther*, 13(2):111–119, 2011.

[80] B. Buckingham, E. Cobry, P. Clinton, V. Gage, K. Caswell, E. Kunselman, F. Cameron, and H.P. Chase. Preventing hypoglycemia using predictive alarm algorithms and insulin pump suspension. *Diabetes Technol Ther*, 11(2):93–97, 2009.

[81] 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*, 33(5):1013–1017, 2010.

[82] B.A. Buckingham, F. Cameron, P. Calhoun, D.M. Maahs, D.M. Wilson, H.P. Chase, B.W. Bequette, J. Lum, J. Sibayan, R.W. Beck, et al. Outpatient safety assessment of an in-home predictive low-glucose suspend system with type 1 diabetes subjects at elevated risk of nocturnal hypoglycemia. *Diabetes Technol Ther*, 15(8):622–627, 2013.

[83] C.S. Hughes, S.D. Patek, M.D. Breton, and B.P. Kovatchev. Hypoglycemia prevention via pump attenuation and red-yellow-green "traffic" lights using continuous glucose monitoring and insulin pump data. *J Diabetes Sci Technol*, 4(5):1146–1155, 2010.

[84] B.P. Kovatchev, M.D. Breton, C. Dalla Man, and C. Cobelli. Biosimulation modeling for diabetes: in silico preclinical trials: a proof of concept in closed-loop control of type 1 diabetes. *J Diabetes Sci Technol*, 3:1374–1381, 2009.

[85] DIAdvisor. Personal glucose predictive diabetes advisor. http://www.diadvisor.eu/, n.a. Accessed 22 January 2014.

[86] S. Grossberg. How does a brain build a cognitive code? *Psychol Rev*, 87(1):1–51, 1980.

[87] S. Haykin. *Neural Networks: a Comprehensive Foundation*. Macmillan College Puglishing Company, 866 Third Avenue, New York, 10022, 1st edition, 1994.

[88] P.D. McNelis. *Neural Networks in Finance: Gaining Predictive Edge in the Market*. Elsevier Academic Press, 84 Theobald's Road, London WC1X 8RR, UK, 2005.

[89] E.R. Jones. Neural networks' role in predictive analytics. http://www.information-management.com/specialreports/2008_61/-10000704-1.html, 2008. Accessed 22 January 2014.

[90] J.M. Mendel and R.W. McLaren. chapter Reinforcement learning control and pattern recognition systems, pages 287–318. Academic Press, New York, 1970.

[91] M.H. Beale, M.T. Hagan, and H.B. Demuth. Neural network toolbox™ user's guide R2011b. Available at http://www.mathworks.it/help/pdf_doc/nnet/nnet_ug.pdf, 2013. Accessed 22 January 2014.

[92] S. Amari, N. Murata, K.R. Müller, M. Finke, and H. Yang. Statistical theory of overtraining. is cross-validation asymptotically effective? *Adv Neural Inf Process Syst*, pages 176–182, 1996.

[93] G.E Hinton. Connectionist learning procedures. *Artif Intell*, 40(1-3):185–234, 1989.

[94] I. Kaastra and M. Boyd. Designing a neural network for forecasting financial and economic time series. *Neurocomputing*, 10(3):215–236, 1996.

[95] I.A. Basheer and Hajmeer M. Artificial neural networks: fundamentals, computing, design, and application. *J Microbiol Meth*, 43(1):3 – 31, 2000.

[96] G.J. Bowden, G.C. Dandy, and H.R. Maier. Input determination for neural network models in water resources applications. part 1-background and methodology. *J Hydrol*, 301(1-4):75–92, 2005.

[97] K. Hornik, M. Stinchcombe, , and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

[98] D. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *SIAM J Appl Math*, 11(2):431–441, 1963.

[99] M.T. Hagan and M. Menhaj. Training feed-forward networks with the Marquardt algorithm. *IEEE Trans Neural Net*, 5(6):989–993, 1994.

[100] S. Del Favero, A. Facchinetti, and C. Cobelli. A glucose-specific metric to assess predictors and identify models. *IEEE Trans Biomed Eng*, 59(5):1281–1290, 2012.

[101] O. Nelles. *Nonlinear system identification: from classical approaches to neural networks and fuzzy models.* Springer-Verlag, Berlin Heidelberg, Germany, 2011.

[102] 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 Trans Biomed Eng*, 59(6):1550–1560, 2012.

[103] C. Dalla Man, M. Camilleri, and C. Cobelli. A system model of oral glucose absorption: validation on gold standard data. *IEEE Trans Biomed Eng*, 53(12):2472–2478, 2006.

[104] C. Dalla Man, R.A. Rizza, and C. Cobelli. Meal simulation model of the glucose insulin system. *IEEE Trans Biomed Eng*, 54(10):1740–1749, 2007.

[105] A. Facchinetti, G. Sparacino, and C. Cobelli. Modeling the error of continuous glucose monitoring sensor data: critical aspects discussed through simulation studies. *J Diabetes Sci Technol*, 4(1):4–14, 2010.

[106] J. D. Gibbons and S. Chakraborti. *Nonparametric Statistical Inference*, volume 168. CRC press, 2003.

[107] 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. *Comput Meth Prog Biomed*, 113(1):144–152, 2014.

[108] A. Facchinetti, G. Sparacino, and C. Cobelli. Online denoising method to handle intraindividual variability of signal-to-noise ratio in continuous glucose monitoring. *IEEE Trans Biomed Eng*, 58(9):2664–2671, 2011.

[109] C. Zecchin, A. Facchinetti, G. Sparacino, and C. Cobelli. Insulin and meal information improvement of glucose prediction by a neural network. In *Diabetes Technol Ther*, volume 16, 2014. Supplement 1, in press.

[110] C. Zecchin, A. Facchinetti, G. Sparacino, and C. Cobelli. Is glucose prediction in Type 1 diabetes improved by adding insulin and meal information? A neural network quantitative study. Submitted.

[111] 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. *Am J Physiol Endocrinol Metab*, 287(4):E637–E643, 2004.

[112] A. Facchinetti, G. Sparacino, and C. Cobelli. An online self-tunable method to denoise CGM sensor data. *IEEE Trans Biomed Eng*, 57(3):634–641, 2010.

[113] B.P. Kovatchev, D.J. Cox, L.A. Gonder-Frederick, and W. Clarke. Symmetrization of the blood glucose measurement scale and its applications. *Diabetes Care*, 20(11):1655–1658, 1997.

[114] J.W. Chen, J.S. Christiansen, and T. Lauritzen. Limitations to subcutaneous insulin administration in type 1 diabetes. *Diabetes Obes Metab*, 5:223–233, 2003.

[115] M. Gevrey, I. Dimopoulos, and S. Lek. Review and comparison of methods to study the contribution of variables in artificial neural network models. *Ecol Model*, 160(3):249–264, 2003.

[116] M.H. Shojaeefard, M. Akbari, M. Tahani, and F. Farhani. Sensitivity analysis of the artificial neural network outputs in friction stir lap joining of aluminum to brass. *Adv Mater Sci Eng*, 2013:1–7, 2013.

[117] C. Zecchin, A. Facchinetti, G.i Sparacino, C. Dalla Man, C. Manohar, J.A. Levine, A. Basu, Y.C. Kudva, and C. Cobelli. Physical activity measured by physical activity monitoring system correlates with glucose trends reconstructed from continuous glucose monitoring. *Diabetes Technol Ther*, 15(10):836–844, 2013.

[118] A. Levine, L.M. Lanningham-Foster, S.K. McCrady, A.C. Krizan, L.R. Olson, P.H. Kane, M.D. Jensen, and M.M. Clark. Interindividual variation in posture allocation: possible role in human obesity. *Science*, 307(5709):584–586, 2005.

[119] C. Manohar, J.A. Levine, D.K. Nandy, A. Saad, C. Dalla Man, S.K. McCrady-Spitzer, R. Basu, C. Cobelli, R.E. Carter, A. Basu, and Y.C. Kudva. The effect of walking on postprandial glycemic excursion in patients with type 1 diabetes and healthy people. *Diabetes Care*, 35(12):2493–2499, 2012.

[120] J.A. Levine, P.A. Baukol, and K.R. Westerterp. Validation of the Tracmor triaxial accelerometer system for walking. *Med Sci Sports Exerc*, 33:1593–1597, 2001.

[121] J. Levine, E.L. Melanson, K.R. Westerterp, and J.O. Hill. Tracmor system for measuring walking energy expenditure. *Eur J Clin Nutr*, 57:1176–1180, 2003.

[122] C. Manohar, S. McCrady, I.T. Pavlidis, and J.A. Levine. An accelerometer-based earpiece to monitor and quantify physical activity. *J Phys Act Health*, 6(6):781–789, 2009.

[123] S. Guerra, G. Sparacino, A. Facchinetti, M. Schiavon, C. Dalla Man, and C. Cobelli. A dynamic risk measure from continuous glucose monitoring data. *Diabetes Technol Ther*, 13(8):843–852, 2011.

[124] S. Patek, L. Magni, E. Dassau, C. Karvetski, C. Toffanin, G. De Nicolao, S. Del Favero, M. Breton, C. Dalla Man, and E. Renard. Modular closed-loop control of diabetes. *IEEE Trans Biomed Eng*, 59(11):2986–2999, 2012.

[125] B. Bode, K. Gross, N. Rikalo, S. Schwartz, T. Wahl, C. Page, et al. Alarms based on real-time sensor glucose values alert patients to hypo-and hyperglycemia: The Guardian continuous monitoring system. *Diabetes Technol Ther*, 6(2):105–113, 2004.

[126] T. Bremer and D.A. Gough. Is blood glucose predictable from previous values? a solicitation for data. *Diabetes*, 48(3):445–451, 1999.

[127] B. Buckingham. Hypoglycemia detection, and better yet, prevention, in pediatric patients. *Diabetes Technol Ther*, 7(5):792–796, 2005.

[128] M. Eren-Oruklu, A. Cinar, and L. Quinn. Hypoglycemia prediction with subject-specific recursive time-series models. *J Diabetes Sci Technol*, 4(1):25–33, 2010.

[129] R.A. Harvey, E. Dassau, H.C. Zisser, W. Bevier, D.E. Seborg, L. Jovanovič, and F.J. Doyle III. Clinically relevant hypoglycemia prediction metrics for event mitigation. *Diabetes Technol Ther*, 14(8):719–727, 2012.

[130] C. Zecchin, A. Facchinetti, G. Sparacino, and C. Cobelli. Reduction of number and duration of hypoglycemic events by glucose prediction methods: A proof-of-concept in silico study. *Diabetes Technol Ther*, 15(1):66–77, 2013.

[131] S. Guerra, A. Facchinetti, G. Sparacino, G. De Nicolao, and C. Cobelli. Enhancing the accuracy of subcutaneous glucose sensors: A real-time deconvolution-based approach. *IEEE Trans Biomed Eng*, 59(6):1658–1669, 2012.

[132] A. Facchinetti, G. Sparacino, and C. Cobelli. Enhanced accuracy of continuous glucose monitoring by online extended kalman filtering. *Diabetes Technol Ther*, 12(5):353–356, 2010.

[133] P.E. Cryer. Mechanisms of hypoglycemia-associated autonomic failure and its component syndromes in diabetes. *Diabetes*, 54(12):3592–3601, 2005.

[134] V.J. Briscoe and S.N. Davis. Hypoglycemia in type 1 and type 2 diabetes: physiology, pathophysiology, and management. *Clinical Diabetes*, 24(3):115–121, 2006.

[135] P. Choudhary, J. Shin, Y. Wang, M. Evans, P.J. Hammond, D. Kerr, J.A.M. Shaw, J.C. Pickup, and S.A. Amiel. Insulin pump therapy with automated insulin suspension in response to hypoglycemia. *Diabetes Care*, 34(9):2023–2025, 2011.

[136] C. Zecchin, A. Facchinetti, G. Sparacino, A. Kamath, T. Peyser, A.L. Rack-Gomer, Y.C. Kudva, and C. Cobelli. In silico study to assess potential reduction of severe hypoglycemia by Dexcom G4 PLATINUM research prototype implementing prediction-based hypoglycemic alerts. In *Book of Abstracts, 13th DTM, San Francisco (CA, USA)*, Oct 31-Nov 2 2013.

[137] N. Bhavaraju, H. Hampapuram, A. Kamath, A.L. Rack-Gomer, C. Cobelli, A. Facchinetti, G. Sparacino, and C. Zecchin. Systems and methods for providing sensitive and specific alarms. US provisional patent No 61/720,286.

[138] C. Dalla Man, D.M. Raimondo, R.A. Rizza, and C. Cobelli. GIM, simulation software of meal glucose-insulin model. *J Diabetes Sci Technol*, 1(3):323–330, 2007.

[139] L. Magni, M. Forgione, C. Toffanin, C. Dalla Man, B. Kovatchev, G. De Nicolao, and C. Cobelli. Run-to-run tuning of model predictive control for type 1 diabetes subjects: in silico trial. *J Diabetes Sci Technol*, 3(5):1091–1098, 2009.

[140] B. Kovatchev, C. Cobelli, E. Renard, S. Anderson, M. Breton, S.D. Patek, W. Clarke, D. Bruttomesso, A. Maran, S. Costa, A. Avogaro, C. Dalla Man, A. Facchinetti, L. Magni, G. De Nicolao, J. Place, and A. Farrett. Multinational study of subcutaneous model-predictive closed-loop control in type 1 diabetes mellitus: summary of the results. *J Diabetes Sci Technol*, 4(6):1374–1381, 2010.

[141] K. van Heusden, E. Dassau, H.C. Zisser, D.E. Seborg, and F.J. Doyle. Control-relevant models for glucose control using a priori patient characteristics. *IEEE Trans Biomed Eng*, 59(7):1839–1849, 2012.

[142] World Health Organization. Definition and diagnosis of diabetes mellitus and intermediate hyperglycemia. http://whqlibdoc.who.int/publications/2006/9241594934_eng.pdf, 2006. Accessed 22 January 2014.

[143] 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 Trans Inf Technol Biomed*, 14(1):157–165, 2010.

Acknowledgements

I would like to thank all the people that supported me and shared with me inspirational comments and discussions, during my PhD program. In particular my advisor, Professor Giovanni Sparacino, for his precious help and guidance during these years, for the freedom he allowed me while developing my research project, for the trust he always gave me and for the discussions and suggestions about future career perspectives. I also would like to acknowledge my colleagues for our conversations and professional and life advices. Two special acknowledgements: the first to Dr. Andrea Facchinetti, my "second supervisor" and travel-mate, for everything he taught me, for his enormous support and for the working experiences we shared. The second to Luca Cherubin, for the constructive work done together on the jump neural network algorithm.

Thanks to the Department of Clinical and Experimental Medicine, University of Padova, for providing us data collected under the DIAdvisor project. Thanks to Mayo Clinic, (Rochester, MN), for sharing with us data collected during an inpatient study designed to detect glycemic patterns in control and T1D subjects, in the presence of mild PA, in particular to Dr Yogish Kudva and Dr Ananda Basu for their useful advices for setting up our analysis and interpreting results. A special thanks to Dexcom people, for the formative collaboration with our research group, for all the ideas and projects shared and for giving me the wonderfully formative opportunity of working three months with them in San Diego.

Thank you to all my friends, with which I shared many moments outside academia.

Un grazie speciale ai miei genitori e a mio fratello, per il loro supporto, la loro pazienza infinita e per aver accettato (e appoggiato) le mie scelte, anche quando non in linea con le loro aspettative.