ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΣΥΣΤΗΜΑΤΩΝ ΜΕΤΑΔΟΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΛΙΚΩΝ

Development of personalised multilevel glucose prediction model for patients with Type 1 Diabetes Mellitus

Μπαϊρακτάρης Φώτιος- Α.Μ.: 03400023

Μεταπτυχιακή διατριβή του ΔΠΜΣ "Επιστήμη Δεδομένων και Μηχανική Μάθηση"

Επιβλέπουσα Καθηγήτρια:
Νικήτα Κωνσταντίνα, Καθηγήτρια ΣΗΜΜΥ, ΕΜΠ

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τις επιβλέπουσές μου Κ. Ζαρκογιάννη και Κ. Νικήτα για τη συνεργασία στα πλαίσια αυτής της διπλωματικής, σε μία περίεργη χρονιά λόγω της πανδημίας. Θέλω επίσης να ευχαριστήσω την οικογένειά μου και όλους όσους με στήριξαν σε αυτή την δύσκολη χρονιά, και να αφιερώσω αυτή την εργασία στη μνήμη του πατέρα μου.

Acknowledgements

I would like to thank my supervisors K. Zarkogianni and K. Nikita for cooperating towards the completion of this work, in a challenging year because of the pandemic. I want to thank my family and everyone who supported me in this challenging year, and dedicate this work in loving memory of my father.

Περίληψη

Ο σακχαρώδης διαβήτης τύπου 1 (ΣΔΤ 1) είναι μία αυτοάνοση ασθένεια που χαρακτηρίζεται από υψηλά επίπεδα γλυκόζης στο αίμα λόγω μειωμένης έκκρισης ινσουλίνης. Τα άτομα με σακχαρώδη διαβήτη χρειάζεται να κάνουν αρκετές αλλαγές στον τρόπο ζωής τους προκειμένου να αποφεύγονται, κατά το δυνατόν, επεισόδια υψηλής γλυκόζης στο αίμα (υπεργλυκαιμικά) ή χαμηλών επιπέδων γλυκόζης (υπογλυκαιμικά). Οι επιπλοκές του ΣΔ μπορούν να είναι από ήπιες μέχρι πολύ σοβαρές οι οποίες μπορεί να απειλήσουν τη ζωή του ασθενούς. Για αυτό τον λόγο ο γλυκαιμικός έλεγχος είναι ζωτικής σημασίας. Για τη ρύθμιση των επιπέδων της γλυκόζης έχουν αναπτυχθεί διατάξεις συνεχούς καταγραφής γλυκόζης, οι οποίες σε συνδυασμό με αντλίες συνεχόμενης έγχυσης ινσουλίνης μπορούν να βελτιώσουν την ποιότητα ζωής των ατόμων με ΣΔΤ1.
Στην παρούσα εργασία αναπτύχθηκε υβριδικό μοντέλο πρόβλεψης των επιπέδων γλυκόζης βασισμένο στη συνδυασμένη χρήση διαμερισματικών μοντέλων και βαθιάς μάθησης. Το μοντέλο λαμβάνει είσοδο καταγραφές γλυκόζης, ρυθμούς έγχυσης ινσουλίνης και την περιεχόμενη ποσότητα υδατανθράκων στα λαμβανόμενα γεύματα και εξάγει προβλέψεις των επιπέδων γλυκόζης στο πλάσμα. Για την ανάπτυξη και αξιολόγηση του μοντέλου χρησιμοποιήθηκαν δεδομένα 12 ασθενών με ΣΔΤ1 καθώς και in silico δεδομένα από τον UVa T1DM simulator.

Λέξεις κλειδιά: Διαβήτης, Γλυκόζη, Ινσουλίνη, Γλυκαιμικός έλεγχος, Νευρωνικά Δίκτυα, Μηχανική μάθηση, LSTM, RNN , GRU

Abstract

Type 1 diabetes mellitus (T1DM) is an autoimmune disease that is characterized by increased blood glucose levels due to impaired insulin response. Patients suffreting from diabetes mellitus need to make several lifestyle changes in order to avoid, at the level that this is possible events of elevated (hyperglycaemic) or depleted blood glucose levels (hypoglycaemic). Complications of DM can range from mild to severe and can be life-threatening. Therefore, glycemic control is of critical importance for T1DM patients. To that end, continuous glucose measuring devices have been developed, which when used in conjuction with continuous insulin infusion pumps can greatly improve quality of life in T1DM patients.
In this work a model to predict future blood glucose values has been developed, based on the combined usage of compartment models and artificial neural networks (ANNs). The model accepts records of glucose, insulin injection rates and the amount of carbohydrates contained in each meal as input data and outputs predicted values of future blood glucose levels. For the development and evaluation of the model data from 12 T1DM patients, as well as in silico data from UVa T1DM simulator were used.

Keywords: Diabetes, Glucose, Insulin, Glycemic control, Neural Networks,Machine Learning, LSTM, RNN , GRU

Table of Contents

1 Introduction **1**
- 1.1 Motivation . . . 1
- 1.2 Objective . . . 2
- 1.3 Structure . . . 2

2 Glucose metabolism **3**
- 2.1 Glucose homeostasis . . . 3
- 2.2 Pancreatic hormones . . . 5
  - 2.2.1 Insulin . . . 5
  - 2.2.2 Glucagon . . . 5
- 2.3 Diabetes Mellitus . . . 6
  - 2.3.1 Types . . . 6
  - 2.3.2 Diagnosis . . . 6
  - 2.3.3 Statistics . . . 7
  - 2.3.4 Complications . . . 9

3 Glycemic Control **11**
- 3.1 Glucose monitoring . . . 11
  - 3.1.1 Conventional methods . . . 11
  - 3.1.2 Continuous glucose monitors . . . 12
- 3.2 Glucose control via exogenous insulin injection . . . 14
  - 3.2.1 Daily multiple injections method . . . 14
  - 3.2.2 Insulin infusion pumps . . . 14

4 Glucose prediction models **17**
- 4.1 Compartmental models . . . 17
  - 4.1.1 Hovorka model . . . 17
  - 4.1.2 Dalla Man model . . . 18
  - 4.1.3 Sorensen model . . . 18
- 4.2 Machine learning models . . . 19
  - 4.2.1 Gaussian processes . . . 19
  - 4.2.2 Adaptive weighted average method . . . 19

4.2.3 SVR with Differential Evolution . . . 20
4.2.4 Feed forward ANN . . . 20
4.2.5 Online adaptive ANN . . . 20
4.2.6 Jump Neural Network . . . 21
4.2.7 Self organizing map . . . 21
4.2.8 ANN with last 20 minutes as input . . . 21
4.2.9 Adaptive ANN strategy . . . 22
4.2.10 LSTM with 1 hour of data as input . . . 22
4.2.11 Fully connected ANN . . . 22
4.3 Hybrid models . . . 22
4.3.1 SVR . . . 22
4.3.2 Ensemble models with CM . . . 23
4.3.3 Insulin Infusion Advisory System . . . 23

5 Machine Learning and Neural Networks 27
5.1 Tasks and Experience . . . 28
5.1.1 Tasks . . . 28
5.1.2 ML methods-Experience . . . 29
5.2 Artificial Neural Networks-ANNs . . . 30
5.2.1 Single layer perceptron . . . 30
5.2.2 Multi layer perceptron-MLP . . . 31
5.3 Training process . . . 32
5.3.1 Data scaling . . . 32
5.3.2 Loss functions . . . 34
5.3.3 Activation functions . . . 35
5.3.4 Optimization . . . 35
5.3.5 Problems in training procedure . . . 37
5.4 Deep learning-Neural Network types . . . 38
5.4.1 Convolutional Neural Networks-CNNs . . . 39
5.4.2 Recurrent Neural Networks-RNNs . . . 39

6 Methodology 43
6.1 The data and conceptual framework . . . 43
6.1.1 In silico data . . . 48
6.2 Data cleaning and multilevel modelling . . . 48
6.2.1 Savitzky-Golay filter . . . 49
6.2.2 Compartment Models . . . 49
6.2.3 Glucose gut absorption compartmental model . . . 50
6.2.4 Subcutaneous insulin compartmental model . . . 51
6.2.5 Insulin signaling pathway CM . . . 51
6.3 Neural network architecture . . . 52

7 Results and Discussion **55**
7.1 Evaluation criteria 55
7.1.1 Regression metric - RMSE 56
7.1.2 Correlation metric - CC 57
7.1.3 Continuous Glucose Error Grid Analysis - CG-EGA 58
7.2 Results and discussion 60
7.2.1 GRU model results 61
7.2.2 MLP model results 62
7.2.3 Discussion 63

8 Conclusion and future work **101**

List of figures

1 Glucose homeostasis illustrated 4
2 Diabetes cases by country. Credit:[Fed19] 7
3 Diabetes related health expenditure. Credit:[Fed19] 8
4 Diabetes deaths in 2019 by age. Credit:[Fed19] 8

5 Venn diagram of ML,AI and Deep Learning. Credits:[GBC16] . . . 28
6 Single layer perceptron model 31
7 Multi layer perceptron model 32
8 ADAM description in pseudocode. Credit:[KB14] 37
9 LSTM illustrated. Credit:[Lst] 40
10 GRU illustrated. Credit:[Lst] 40

11 Patients' age, gender and diabetes duration information 44
12 Information regarding individual patients' sets of data 44
13 Conceptual diagram of the hybrid model constructed in the present work 45
14 Descriptive statistics of sample dataset (patient 1) 46
15 Descriptive statistics of the sample dataset after imputation 48
16 In silico dataset example 48
17 Insulin pathway CM. Credit:[GGJ13] 52

18 Clarke's Point Error Grid Analysis (P-EGA) zones illustrated. Credit:[CAK08] 59
19 Clarke's Rate Error Grid Analysis (R-EGA) zones illustrated. Credit:[CAK08] 60
20 GRU model results for patients 1 to 3. Prediction Horizon: 30 minutes 67
21 GRU model results for patients 4 to 6. Prediction Horizon: 30 minutes 68
22 GRU model results for patients 7 to 9. Prediction Horizon: 30 minutes 69
23 GRU model results for patients 10 to 12. Prediction Horizon: 30 minutes 70
24 GRU model results for one patient from each in silico dataset age group. Prediction Horizon: 30 minutes 71
25 GRU model results for patients 1 to 3. Prediction Horizon: 60 minutes 72
26 GRU model results for patients 4 to 6. Prediction Horizon: 60 minutes 73
27 GRU model results for patients 7 to 9. Prediction Horizon: 60 minutes 74

28 GRU model results for patients 10 to 12. Prediction Horizon: 60 minutes 75
29 GRU model results for one patient from each in silico dataset age group. Prediction Horizon: 60 minutes 76
30 GRU model results for patients 1 to 3. Prediction Horizon: 120 minutes 77
31 GRU model results for patients 4 to 6. Prediction Horizon: 120 minutes 78
32 GRU model results for patients 7 to 9. Prediction Horizon: 120 minutes 79
33 GRU model results for patients 10 to 12. Prediction Horizon: 120 minutes 80
34 GRU model results for one patient from each in silico dataset age group. Prediction Horizon: 120 minutes 81
35 MLP model results for patients 1 to 3. Prediction Horizon: 30 minutes 82
36 MLP model results for patients 4 to 6. Prediction Horizon: 30 minutes 83
37 MLP model results for patients 7 to 9. Prediction Horizon: 30 minutes 84
38 MLP model results for patients 10 to 12. Prediction Horizon: 30 minutes 85
39 MLP model results for one patient from each in silico dataset age group. Prediction Horizon: 30 minutes 86
40 MLP model results for patients 1 to 3. Prediction Horizon: 60 minutes 87
41 MLP model results for patients 4 to 6. Prediction Horizon: 60 minutes 88
42 MLP model results for patients 7 to 9. Prediction Horizon: 60 minutes 89
43 MLP model results for patients 10 to 12. Prediction Horizon: 60 minutes 90
44 MLP model results for one patient from each in silico dataset age group. Prediction Horizon: 60 minutes 91
45 MLP model results for patients 1 to 3. Prediction Horizon: 120 minutes 92
46 MLP model results for patients 4 to 6. Prediction Horizon: 120 minutes 93
47 MLP model results for patients 7 to 9. Prediction Horizon: 120 minutes 94
48 MLP model results for patients 10 to 12. Prediction Horizon: 120 minutes 95
49 Collective results of GRU and MLP models for real patients' data . 96
50 Collective results of GRU and MLP models for in silico children data 97
51 Collective results of GRU and MLP models for in silico adolescents data 98
52 Collective results of GRU and MLP models for in silico adults data 99

Chapter 1

Introduction

Diabetes Mellitus (DM), most widely known as diabetes, refers to a group of metabolic disorders characterized by increased blood glucose levels over a prolonged period of time. DM is closely linked to pancreatic function and insulin response, and can cause life-threatening complications, both long term and acute. A total of 463 million people are estimated to be living with some form of diabetes, and approximately 4.2 million died from related complications in 2019 [Fed19]. Scope of the present thesis is to develop a personalized glucose prediction model for T1DM patients, utilizing a hybrid architecture of neural networks and multilevel modelling via mathematical compartment models. An accurate predictive model has the potential to be used in automated controllers that moderate insulin dosage in order to prevent hypoglycemic and hyperglycemic events.

1.1 Motivation

DM is on the rise globally, making the improvement of control systems increasingly important. Glucose control systems aim to keep glucose levels in check for T1DM patients, allowing them to lead a normal life free of the complications that can arise by the disease.
The most widely used method for measurement of blood glucose levels is also the most inconvenient for the patient, since it involves frequent fingerstick measurements and insulin injections. Automation of this procedure can lead to improved quality of life in T1DM patients. Technological advancements in the field have lead to impressive progress concerning closed loop control systems, which aim to control blood glucose levels so as to prevent acute episodes, while being increasingly less invasive. Continuous glucose monitoring devices comprise one part of such systems, while an accurate predictive model is also needed for prevention of hypo - and hyperglycemic events. The present work revolves around the construction

of a predictive model to be incorporated into such systems.

1.2 Objective

Machine learning and artificial intelligence is a rapidly growing field with the potential to produce scientific breakthroughs and drastic quality of life changes for the general population. These tools can be used to aid in combating a wide range of diseases, and diabetes is a characteristic example of them. Glucose prediction models can be improved by application of ML and Artificial Neural Networks (ANNs). Ample data is available for patients with T1DM, facilitating the training process of an ML model. A hybrid method consisting of mathematical models for glucose and insulin kinetics, combined with ANNs will be utilized to construct predictive models. Comparison of the hybrid framework of the present thesis to the current state of art ANN will take place.

1.3 Structure

The outline of this thesis is as follows. The first part (chapters 1 to 5) contains theoretical and bibliographic references, and the second part (chapters 6 to 8) contains methodology, model presentation and discussion of results.
As for the outline of the first part, the second chapter presents theoretical aspects of the glucose homeostasis mechanism and diabetes as a disease that arises from disruption of said mechanism. The third chapter contains information about various glycemic control systems. Existing predictive models will be discussed in the fourth chapter. Theoretical aspects of machine learning and neural networks shall be presented in chapter 5.
Moving on to the second part, chapter 6 contains methodology. Data preprocessing, theoretical models used and types of ANNs that comprise the predictive models will be discussed. Chapter 7 contains evaluation metrics and presentation of results. Chapter 8 concludes the present thesis by discussing future possible work in this area.

Chapter references

[Fed19] International Diabetes Federation. *IDF Diabetes Atlas-Ninth Edition 2019*. 2019.

Chapter 2

Glucose metabolism

2.1 Glucose homeostasis

Every movement and function of the human body requires expenditure of energy, which needs to be restored. The process of energy replenishment involves chemical reactions which take pace at the intracellular level, with sugars and fatty acids as fuel, and oxygen and various enzymes as catalysts. Glucose is then produced, which acts as the main source of fuel for human cells. Brain and central nervous system cells rely exclusively on glucose for fuel, barring long periods of absence of food intake. In addition to its core function as an energy source, glucose is also stored in the liver and muscle tissue in the form of glycogen. Circulation of blood is responsible for transferring glucose throughout organs, after it is absorbed in the gut via the process of digestion.
Blood glucose concentration is dependent on the rate of glucose insertion and exit to and from the bloodstream. It originates from the following sources:

- Glucose released from the gut, after being absorbed though meal digestion.
- Glycogen breakdown in the liver.
- The process of glycogenesis, in which glucose is synthesized from other organic molecules.

If an individual is fasted for 8-12 hours, glycogen breakdown in the liver takes over as the main source of glucose. Longer fasting periods enable the process of glycogenesis in the liver. Various hormones play important roles in the glucose homeostasis mechanism in the human body, among the most important of which are insulin and glucagon.

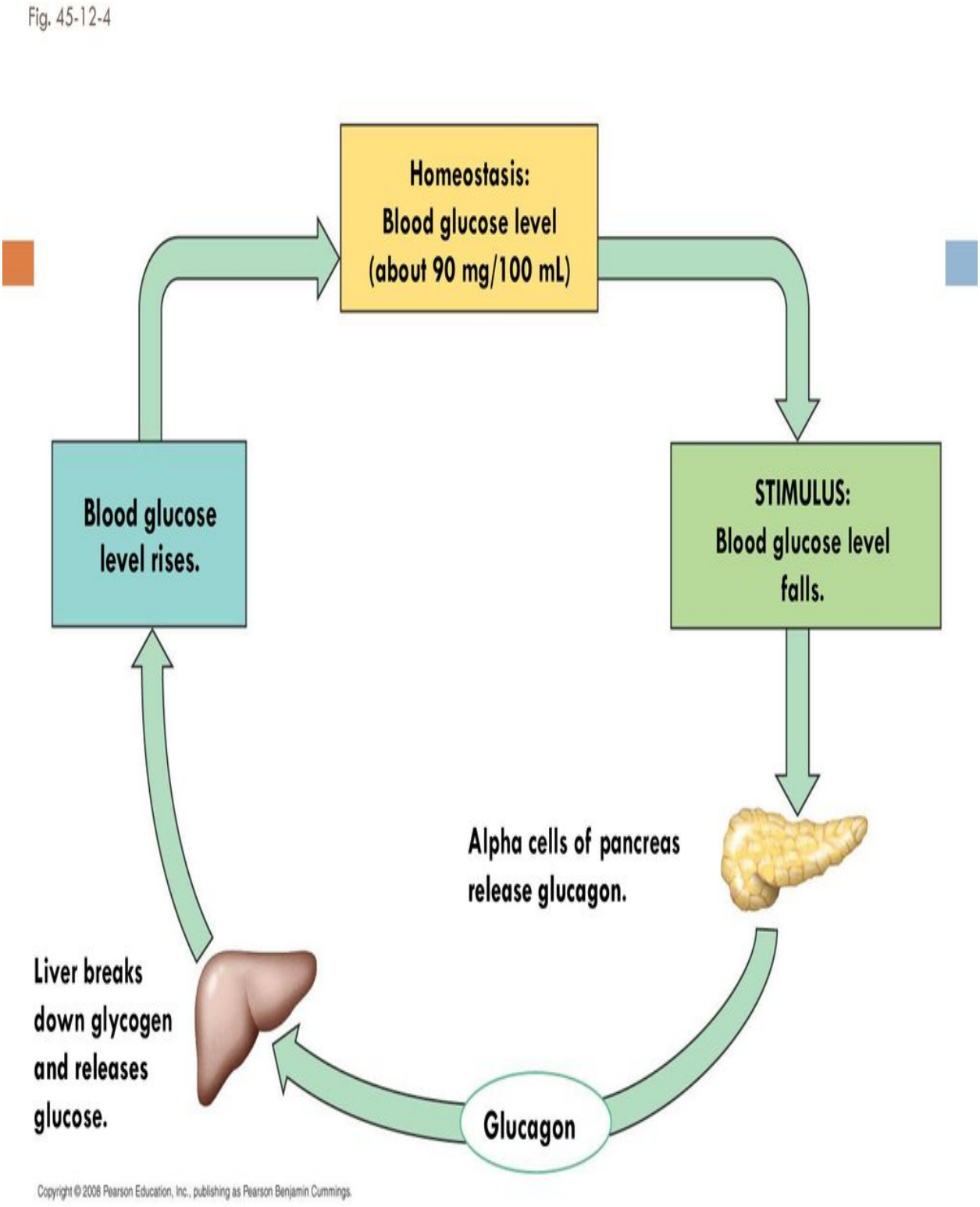

Fig. 1: Glucose homeostasis illustrated

2.2 Pancreatic hormones

2.2.1 Insulin

Insulin is a peptide hormone produced by beta cells of the pancreas, and is considered as the main anabolic hormone in the body, serving a regulatory purpose in the process of macronutrient (carbohydrates, protein and fatty acids) metabolism, by promoting the absorption of glucose from the blood into liver, fat and muscle tissue. It is a protein consisting of two polypeptide chains, numbering a total of 51 aminoacids. Insulin is secreted in response to high level blood glucose levels and aminoacid intake, both of which take place after meals. Similar to other hormones, the function of insulin is linked to several receptors in fat, liver and muscle cells. Its primary purpose is to remove glucose from the bloodstream to the aforementioned destinations. Specifically:

- Insulin signals increased glucose absorption from peripheral tissues (mainly skeletal muscles).
- Following the above action, it signals glycogenesis in liver.
- Almost concurrent to the above action it inhibits glucagon secretion from pancreatic alpha cells, temporarily stopping glycogenesis and glycolysis procedures.

Insulin is only secreted when blood glucose concentration is greater than 3.3 $\frac{mmol}{lt}$. After a meal, rapid insulin secretion takes place, followed by increased pancreatic insulin production in response to elevated blood glucose levels.

2.2.2 Glucagon

Glucagon is a hormone consisting of 29 aminoacids and is produced by pancreatic alpha cells. While insulin moderates glucose levels during meals, glucagon does the same for fasting periods (it promotes liver glucose production during fasting). Sherwood, Parris and Unger [SPU71] were the first to describe DM as a two-hormone imbalance disease, rather than solely insulin dependent. They also theorized that treatments aiming to reduce glucagon levels (when higher than normal) will be of high importance to combat DM. While in a healthy individual glucagon is inhibited during meals and insulin acts fast on stopping liver glucose production, in patients with DM, post meal glucagon inhibition is inadequate to stop the liver from producing glucose.

2.3 Diabetes Mellitus

Decreased insulin response of an individual's person's cells (meaning they need a higher amount of insulin than normal to absorb glucose) leads to a condition known as insulin resistance. The pancreas has to compensate for this situation, producing more insulin to help glucose enter the cells. As long as the insulin is sufficient to keep blood glucose levels in the healthy range, the person will not develop DM. However, excess weight and physical inactivity are factors that contribute to insulin resistance, subsequently increasing the risk of diabetes over time.

2.3.1 Types

Type I Diabetes Mellitus (T1DM)

T1DM, also called insulin dependant diabetes, is an autoimmune disease, in which the body attacks and destroys pancreatic beta cells. This in turn leads to reduced insulin secretion, or none at all. As a result of that, blood glucose levels can deviate from the healthy ranges, making the patient prone to either very high (hyperglycemia) or very low blood glucose concentration (hypoglycemia). T1DM has genetic predisposition and is mainly diagnosed at a young age. However, it can also appear in adults, if the autoimmune disease was not aggressive enough to destroy a large number of beta cells in the early years of a person's life.

Type II Diabetes Mellitus (T2DM)

T2DM, also called non insulin dependant diabetes, is an acquired disease. It is caused by reduced insulin secretion, increased insulin resistance, or both. Insulin resistance is the main complication in the early stages of the disease. T2DM is the most common form of diabetes in adults. Factors contributing to its appearance are obesity, age and family history of T2DM.

2.3.2 Diagnosis

Diagnosis of diabetes requires one of the following:

- Fasted blood sugar levels $> 126\frac{mg}{dl}$, measured before 9 AM and with at least 9 hours of fasting. This person, even if asymptomatic, is immediately classified as diabetic.
- If key symptoms such as frequent thirst and urination are present, and at a random blood sugar measurement (any time of the day) the value is greater than $200\frac{mg}{dl}$.

- Glucose tolerance test: If two hours after drinking a specific fluid, blood sugar levels are above $200\frac{mg}{dl}$.

2.3.3 Statistics

Diabetes is on the rise worldwide with 463 million cases in 2019, estimated to reach 578 million by 2030 and 700 million by 2045 [Fed19]. Figure 2 shows diabetes cases by country.

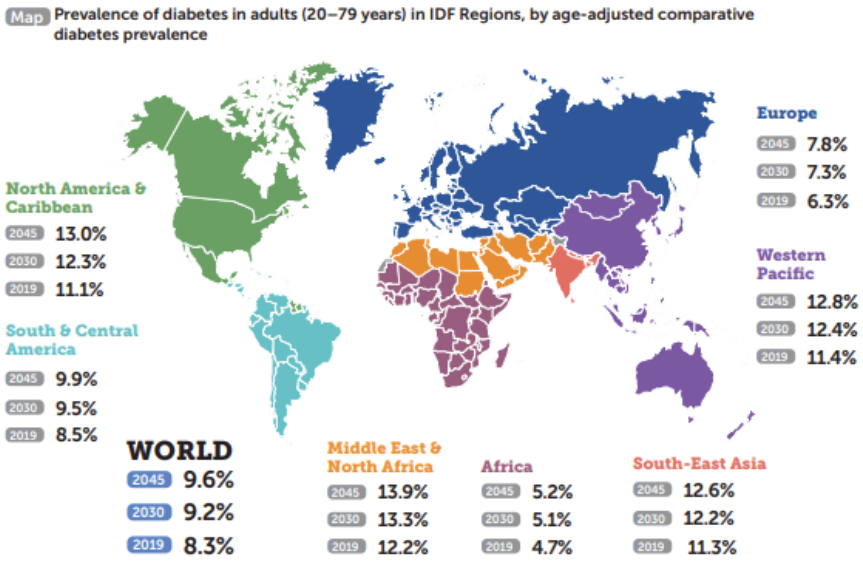

Fig. 2: Diabetes cases by country. Credit:[Fed19]

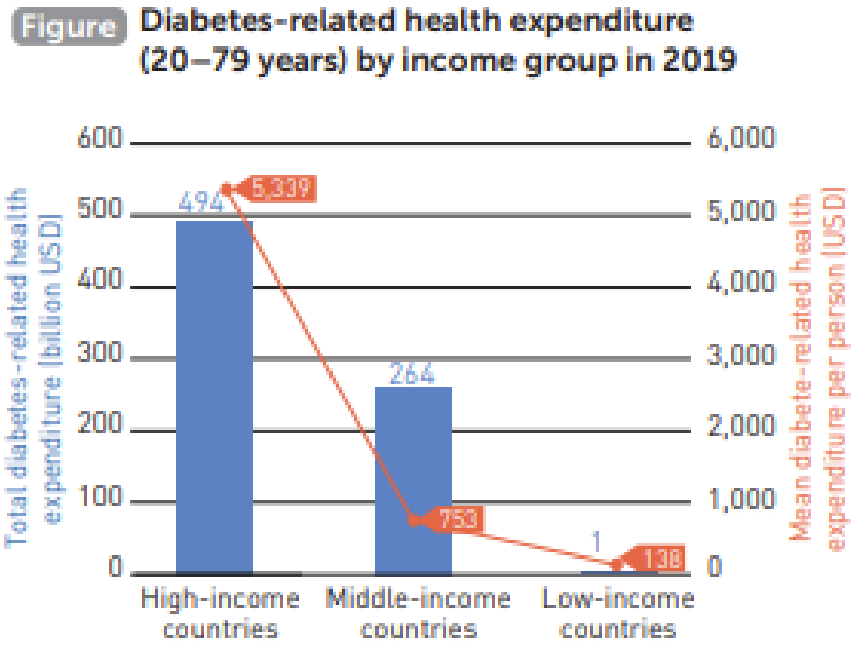

Fig. 3: Diabetes related health expenditure. Credit:[Fed19]

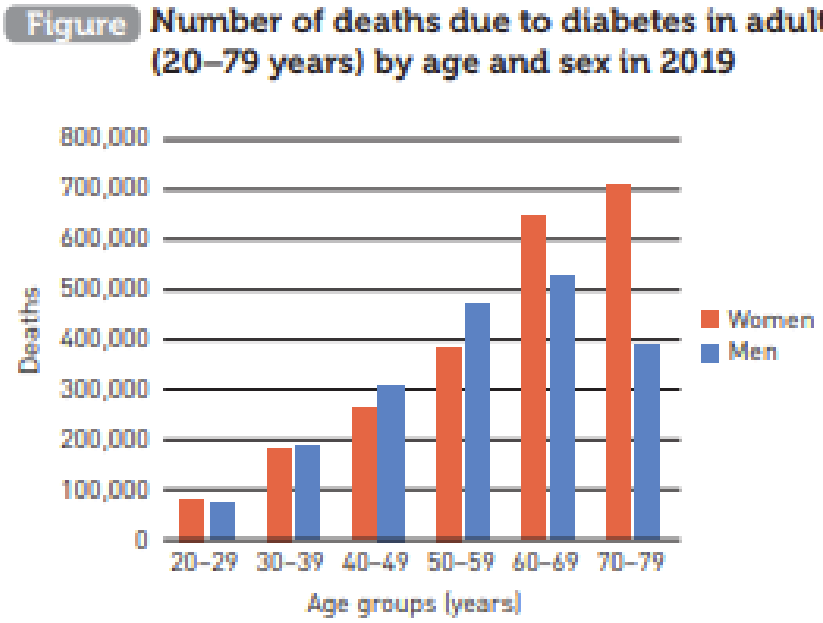

Fig. 4: Diabetes deaths in 2019 by age. Credit:[Fed19]

Apart from medical issues, the economic and social impact of DM is significant as well. Figure 3 shows health expenditure in billions of USD in 2019 for groups

of countries, classified by their average income. Direct costs were 760 billion USD in 2019, expected to reach 825 billion by 2030 and 2045 billion by 2045.

2.3.4 Complications

Common complications of diabetes, some of which can be life threatening, are given below:

- Eye disease: Diabetic retinopathy is a leading cause of blindness in many countries. It is caused by both T1DM and T2DM, and is associated with deteriorating glycaemic control and the presence of hypertension. Diabetic retinopathy impairs quality of life for people living with diabetes significantly.
- Cardiovascular diseases account for 30 to 50% of all diabetes-related deaths. Elevated blood glucose with diabetes is associated with almost double the cardiovascular disease risk.
- Diabetic kidney disease: Diabetes and hypertension account for 80 percent of end-stage kidney disease globally. Also associated with cardiovascular diseases.
- Nerve/vascular damage and diabetic foot complications: This complication affects 40 to 60 million people globally. Diabetes-related peripheral neuropathy is often misdiagnosed and this leads to high rates of morbidity and mortality.

Various technological advances have been made in order to prevent diabetes related complications. One of the biggest breakthroughs are systems which can continuously monitor glucose in patients, which are called Continuous Glucose Monitors (CGMs).

Chapter references

[Fed19] International Diabetes Federation. *IDF Diabetes Atlas-Ninth Edition 2019*. 2019.

[SPU71] Louis M. Sherwood, Edith E. Parris, and Roger H. Unger. "Glucagon Physiology and Pathophysiology". In: *New England Journal of Medicine* 285.8 (1971), pp. 443–449. DOI: 10.1056/nejm197108192850806.

[Who] *Diabetes Fact sheet Number 312*. https://web.archive.org/web/20130826174444/http://www.who.int/mediacentre/factsheets/fs312/en/.

[Ζ11] Κωνσταντία Χ. Ζαρκογιάννη. "Ευφυή Συστήματα Υποστήριξης Εξατομικευμένων Ιατρικών Αποφάσεων για τη Διαχείριση του Σακχαρώδους Διαβήτη". PhD thesis. ΕΜΠ, Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, 2011.

Chapter 3

Glycemic Control

Complications of diabetes are a consequence of high or low blood glucose levels. If those levels stay in the normal ranges, a person with DM can live a normal life. Therefore, any lifestyle modifications for diabetes should take place with the aim to keep glucose levels in the clinically acceptable range. With that purpose in mind, glycemic control is what must be achieved. This chapter is dedicated to presenting the two elements of glycemic control: Glucose monitoring and insulin administration.

3.1 Glucose monitoring

Glucose monitoring takes place with two methods. The traditional blood measurement and more novel methods of continuous monitoring.

3.1.1 Conventional methods

Conventional fingerstick methods involve taking blood sample from one finger of the patient with the method of piercing the fingertip. The sample is then dropped on a single use measuring strip containing chemical compounds that react to blood glucose (such compounds are glucose oxidase, dehydrogenase and hexokinase). The strip is then inserted to the measuring device [TLT02]. Blood glucose concentration is calculated either by measuring either electrical current flow through the sample or percentage of reflected radiation. These devices have a range of PC connecting capabilities, both wired and wireless, and results can then be statistically analyzed. Accuracy of measurements is a very important advantage of this method, since blood measurements are taken as the gold standard of glucose monitoring. However, pain and discomfort associated with finger piercing leads to such methods being impractical for continuous monitoring. Even hourly measurements would be nearly impossible to incorporate in a patient's lifestyle.

This lead to the development of a different type of devices with the purpose of enhanced glycemic control.

3.1.2 Continuous glucose monitors

Continuous Glucose Monitors (CGMs) [Cgm] are state of the art devices that measure a person's blood glucose levels in real time (usually in 5 minute intervals). They are the preferred method to incorporate in real time control scenarios, and are divided into categories with respect to the biochemical procedure used for glucose measurement [Gir+09].

Invasive CGMs

This category of CGMs require an invasive approach, typically subcutaneous insertion of a sensor. They function by collecting various measurements from the interstitial fluid. Some categories are described below.
Subcutaneous sensors using glucose oxidase enzyme: These CGM devices consist of two parts. The subcutaneous sensor and a portable device which monitors glucose levels. The sensor contains an electrode with glucose oxidase enzyme at one of its ends. This enzyme acts as a catalyst for the oxidosis process of glucose into gluconolactone and hydrogen peroxide. Hydrogen peroxide is further broken down (via electrical current) into two hydrogen cations, an oxygen molecule and two free electrons. Free electrons modify the electric signal proportional to glucose concentration, which is calculated by this procedure. Important advantages of such devices include wireless connection to the monitor, real time monitoring capabilities, high frequency of measurements (at most every 5 minutes), and ease of installation (in some cases, it can be self-installed by the patient). There are some drawbacks to using devices that fall into this category. There is increased chance of local inflammation, due to proteins being trapped in the sensor. Moreover, such systems need frequent scaling of measurements, typically requiring at least two fingerstick measurements per day. Finally, inaccuracies in fingerstick measurements are propagated into the readings of the device.
Microdialysis based sensors: This system consists of a subcutaneous catheter which constantly collects interstitial fluid samples, and an external device which contains glucose oxidase. While more accurate and needing less fingerstick measurements than the previous method, it is impractical to use due to its dimensions. It also has limited usage in real time control scenarios, since measurements lag about 35 minutes.
Fully implantable sensors: This category shows much promise due to their one-time intervention necessary for the life duration of each sensor. They can also be more readily implemented in closed loop control systems. Issues include tissue reactions and general biointerface problems that reduce sensor lifespan. However,

glycemic control with implantable insulin infusion pump has been shown to be safe and with acceptable performance. After biointerface and cost issues are resolved, such systems could be the method of choice concerning practicability and improving patients' quality of life.

Non invasive CGMs

Non invasive CGM devices rely mainly on spectroscopy methods to measure glucose. Some types of devices are describes below.
Optical sensors use some form of spectroscopy to measure glucose concentration. These include the following.
Devices utilizing near infrared(IR) spectroscopy. Near IR radiation can penetrate tissue up to 10 cm, increasing with wavelength. Experiments have shown good correlation between blood glucose levels and diffuse reflectivity of near IR radiation in fingertips, making this method acceptable. Disadvantages include high susceptibility of measurements to changes in pressure, temperature, humidity and atmospheric pressure.
IR spectroscopy devices can also be used, but its usage is much more limited than the previous method. Its main function is detection of glucose related infection in mucus membrane.
Raman spectroscopy is based on measurement of scattered light intensity, which is mainly influenced by oscillation and rotation of scattering. However, measurement errors and instabilities depending on various chemical compounds in skin tissue make this an unreliable method.
Photoacoustic spectroscopy method uses an optical beam to cause rapid thermal increase in the sample, producing a sound wave which is then measured by a microphone. Although correlation between photoacoustic signal from index finger and blood glucose levels is excellent, the device is costly and sensitive to environmental factors.
Measurement of scattered light diffusion: This method involves measurement of changes in the refractive index of skin tissue. Raise in blood glucose levels is proportional to the raise of refractive index, showing excellent correlation especially for measurements in the abdominal area. However, there are a number of factors which can affect the scattered signal, and they must be taken into account.
Measuring changes in polarization: Glucose is a good optical rotator, and this can be used in extracting in vitro glucose measurements. However, skin is not a good site for such measurements, due to high optical scattering. Aqueous humour of the eye is a good site for this technique to be used. Environmental factors affect this technique not unlike the previous ones, making it a challenge to be implemented.

Skin wearable sensors are another form of non invasive CGMs. Such devices

make use of either reverse ionization procedure, or spectroscopy of tissue resistance. The former involves constant low power current flow which causes glucose motion in the interstitial fluid. Glucose concentration is then measured by the sensor. In practice, results have been poor for this method. The latter technique measures tissue impedance and has shown promise in lab tests, but much lower accuracy at home measurements.

3.2 Glucose control via exogenous insulin injection

Combined with a CGM, insulin dosage is crucial to maintaining healthy glucose ranges. It can be administrated either via multiple injections or continuously via devices that are called insulin pumps.

3.2.1 Daily multiple injections method

Devices with shapes resembling that of a pen, called ”insulin pens”, are gradually replacing syringes and facilitate injection of multiple doses of insulin. They consist of a vial containing anywhere from 1.5 to 3 ml of U 100/ml insulin, and a dosage indicator. One time use needles are used for administration.

3.2.2 Insulin infusion pumps

Insulin infusion pumps secure continuous flow of insulin 24/7, and are the state of the art method of exogenous insulin secretion. They consist of an insulin vial and a micro-computer, which enables the user to programme basal insulin rate as well as insulin bolus dosage. They can be readily implemented into closed loop control and real time control systems, leading to improvement of HbA1c values, reducing danger of severe hypoglycemia and normalizing insulin deviations. Systems involving CGMs and insulin pumps have the advantages of:

- Much less frequent fingerstick measurements.
- Wireless communication between CGM and insulin pump, to prevent extreme events.
- Ability to create graphs in real time, with resolution time of 5 minutes.

Insulin therapy with such closed loop control systems lead to reduced danger of hypoglycemic and hyperglycemic episodes and improved quality of life for patients with DM. The continuous need for better prediction of extreme events is crucial for the perfection of such systems.

Chapter references

[Cgm] *Healthline: Everything You Need to Know About Insulin*. https://www.healthline.com/diabetesmine/what-is-continuous-glucose-monitor-and-choosing-one.

[Gir+09] Céline M. Girardin et al. "Continuous glucose monitoring: A review of biochemical perspectives and clinical use in type 1 diabetes". In: *Clinical Biochemistry* 42.3 (2009), pp. 136–142. DOI: 10.1016/j.clinbiochem.2008.09.112.

[TLT02] J.A. Tamada, M. Lesho, and M.J. Tierney. "Keeping watch on glucose". In: *IEEE Spectrum* 39.4 (2002), pp. 52–57. DOI: 10.1109/6.993789.

[Z11] Κωνσταντία Χ. Ζαρκογιάννη. "Ευφυή Συστήματα Υποστήριξης Εξατομικευμένων Ιατρικών Αποφάσεων για τη Διαχείριση του Σακχαρώδους Διαβήτη". PhD thesis. ΕΜΠ, Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών, 2011.

Chapter 4

Glucose prediction models

The present work revolves around constructing a glucose prediction model that produces satisfactory results for the patients' datasets, after the proper augmentation is done. In this chapter, a bibliographic review of existing models will take place. Models are divided between mathematical (compartmental models), machine learning and hybrid (compartmental models combined with machine learning). The reader must keep in mind that a machine learning model is not only defined by its ''building blocks'', but also the data that are used to train it. To compare such models, one must do so with the same dataset for the comparison to be valid. Even if none of the models presented in this chapter are trained on the same dataset, there is value in presenting similarities and differences in their construction. For each model presented, the respective train/test dataset characteristics and preprocessing will be mentioned.

4.1 Compartmental models

Compartmental models were the first to be used for modelling glucose kinetics, long before ANNs became popular. Nowadays, pure compartmental models are rarely used without combination of some form of data-driven approach, but some models will be presented for completeness.

4.1.1 Hovorka model

Hovorka et al [Hov+04] developed a simulation model with compartmental models of glucose and subcutaneous insulin kinetics, as well as a two-compartment model of glucose gut absorption. This model has the ability to simulate the constantly variable behavior of glucose metabolism system of T1DM patients. Model parameters were either determined from clinical trials or probability distributions. The gut absorption model is too simplistic in this context, which results in less

accurate simulation accuracy when compared to more complex gut absorption CMs. Hovorka used this model to construct a novel glucose control algorithm for the Artificial Pancreas Consortium.

4.1.2 Dalla Man model

Dalla Man et at [MRC07] developed a model to simulate glucose absorption after meals. Data was collected from 204 healthy subjects, following certain meal protocols. Using glucose trackers, glucose and insulin flow during meals were calculated. There are two distinct subsystems in this model, namely the glucose and insulin compartmental models. Parameters of the model describing healthy patients' meal glucose absorption were determined. In a smaller dataset of 14 T2DM patients, relevant parameters were calculated for this population. The authors also used this model to simulate T1DM patients, replacing the insulin secretion compartmental model with one of subcutaneous insulin kinetics. Endogenous glucose production was raised in this simulation, due to higher glucose levels in the T1DM population.

4.1.3 Sorensen model

Sorensen's compartmental model is one of the more complex physiological models, treating the human body as a collection of six physiological compartments:

- The central nervous system.
- Heart and lungs.
- Sceletal muscle and fatty tissue.
- The gut.
- The liver.
- The kidneys.

Glucose and insulin subsystems are considered independent of each other, and are coupled via control flows. 22 differential equations are used for a healthy individual, and 19 DEs with 44 parameters for a T1DM patient. However, this model only serves the purpose of describing the average T1DM patient, because of constant parameters. It was later improved by Parker et al [PGD00] to evaluate glucose controllers.

4.2 Machine learning models

Three of the models that will be described in this section do not make use of neural networks. They are autoregressive (AR) models utilizing support vector regression (SVR) [Ham+18], Gaussian processes (GP) [Geo+15] and adaptive weighted average framework [WWM13].

4.2.1 Gaussian processes

Georga et al [Geo+15] used the Gaussian process technique along with feature ranking algorithms to select the most informative features of glucose models.
Data was obtained from 15 T1DM patients undergoing multiple dose insulin therapy as part of the METABO project. CGM device and armband were the same as in the reference above.
Preprocessing was an important part in this work, as Random Forest (RF) feature ranking algorithms were used which made usage of the bootstrap technique. RReliefF was also used, as a standard feature ranking algorithm for regression problems. After defining feature importance cutoff, feature selection process took place. SVR technique with 10-fold cross validation was used as in the previous publication from the same author, along with Gaussian processes. Results demonstrated superiority of the GP method over plain SVR, as well as improvement of performance with the addition of the feature ranking algorithm as a preprocessing step.

4.2.2 Adaptive weighted average method

Wang et al [WWM13] used an adaptive weighted average framework based on AR, extreme learning machine (ELM) and SVR.
CGM features of 10 T1DM patients were used, part of the JDRF CGM Study Group randomized clinical trial. Only past blood glucose data were used in this framework, without information on insulin or other biomarkers.
The algorithm of the predictive model combined standard AR, ELM and SVR predictions, and utilized all of them. Detailed description of the scheme is as follows. For each prediction algorithm, time-varying weights were assigned, which were proportional to the accuracy of the performance of each one. This leads to different degrees of trust for each individual algorithm, and it follows that the resulting model is superior to each of its component parts (since it optimizes the ”trust” for each individual method). Such a scheme has no limits to its usage, and could even be done with different ANN algorithms.

4.2.3 SVR with Differential Evolution

Hamdi et al [Ham+18] used the SVR technique along with Differential Evolution (DE) for SVR parameter optimization.
Real CGM data of 12 patients were used. No other preprocessing took place, as past blood glucose values were the sole feature used.
The Differential Evolution algorithm was used to select parameters for SVR. This algorithm belongs in the family of genetic algorithms, where a fitness function is selected to measure performance. With each step, the best performing weight vectors carry on to the next generation. After reaching the optimal parameters, forecasting values are obtained. This method demonstrated better performance than plain SVR.
The next models in this section make use of some form of ANNs to predict glucose values, as do most in the recent literature.

4.2.4 Feed forward ANN

Pappada et al [Pap+11] used a feed-forward ANN to predict blood glucose in real time with a prediction horizon of 75 minutes.
Data was obtained from 27 T1DM patients, with a 17 patient train-10 patient test split. CGM data was not the only feature used, as insulin, nutritional intake, lifestyle and emotional factors data were utilized as well.
A feed-forward ANN with a three-layer design was implemented. The input layer was configured with 11 neurons, one for each input (feature). The hidden layer was designed with nine neurons and hyperbolic tangent as the activation function. Results were regarded as clinically acceptable and not leading to adverse therapeutic direction.

4.2.5 Online adaptive ANN

Daskalaki et al [Das+12] used an online adaptive ANN and compared to an AR model, and an AR with external insulin input (ARX).
Glucose and insulin data of 8 days were derived from a virtual population of 30 patients (divided equally into adults, adolescents and children). Meal scheme was simulated and standardized for all patients. Data were normalized in the range $[-1, 1]$ before being fed into the model.
A fully connected ANN was constructed with one input layer, a variable number of hidden layers and neurons per layer, and an output layer of one neuron. Hyperbolic sigmoid and linear were used as transfer functions for the hidden and output layer respectively. A real time recurrent learning algorithm was used for training. Results demonstrated superiority of the ANN model to other AR methods.

4.2.6 Jump Neural Network

Zecchin et al [Zec+14] proposed what is called a jump NN prediction algorithm for prediction horizon of 30 minutes.
Glucose and meal data were available for 20 T1DM patients. Normalization and Bayesian smoothing were performed in the preprocessing stage. The train/test scheme involved tuning the NN parameters (training) on the first 10 patients and evaluating on the remaining 10.
The feed-forward ANN proposed is called ”jump NN”, because the inputs are connected not only to the first hidden layer, but the output layer as well. Nonlinear activation functions were used in hidden neurons and linear ones for the output. Ten-fold cross validation was used for parameter optimization. Results were satisfactory compared to the reference NNs at the time of publish of the paper.

4.2.7 Self organizing map

Zarkogianni et al [Zar+15] proposed a self organizing map (SOM) to predict future glucose values.
Data from 10 T1DM patients from a six day observation period were used. Apart from CGM sensor data, physical activity data was also extracted. Normalization occurred before feeding data into the model.
SOMs can be used for dynamic modelling, creating a two dimensional grid of N neurons and associating each one to a weight vector with dimensions equal to those of the input vector. During each training step, the winning neuron is the one with the least euclidean distance to the input vector. Weights are then updated and the next training step begins. SOMs were compared to feedforward NNs, wavelet NNs and a linear regression model on this dataset, and provided superior predictive performance both in mathematical and clinical criteria.

4.2.8 ANN with last 20 minutes as input

Perez-Gandia et al [PG+10] proposed an ANN with the last 20 minutes of blood glucose levels as input.
Data from 15 T1DM patients over a four week period were used, without any form of normalization, as only one feature was used.
This feedforward ANN comprised of three layers, with 10 neurons in the first and five in the second. Sigmoid function is used to transfer weights between neurons. The output layer has only one neuron with linear transfer function. In each training step, the last 20 minutes of blood glucose data were used. In comparison to a simple AR model, this ANN proved superior.

4.2.9 Adaptive ANN strategy

Ben Ali et al [Ali+18] proposed an ANN and optimized the number of features for each patient.
Blood glucose data from 13 T1DM patients were used, without normalization. The feedforward ANN utilized grid search for each individual, constructing different feed-forward ANNs. Benchmarking against SVR, AR and ELM demonstrated better performance for this model.

4.2.10 LSTM with 1 hour of data as input

Martinsson et al [Mar+19] proposed an RNN with the last 60 minutes of CGM as input.
Data from 6 T1DM patients were used, without normalization. The model utilized LSTM cells, with a 60/20/20 train/validation(with early stopping)/test split. Adam optimizer was used, and parameter selection took place with the grid search method. The performance of this method was satisfactory, but seemed to be less accurate than recurrent models with fewer input data (fewer look-back steps).

4.2.11 Fully connected ANN

Alfian et al [Alf+20] proposed a fully connected ANN that is taken as the current state of the art.
CGM data of 12 real T1DM patients were used. Savitzky Golay filter was used to clean noise from the dataset.
Two 100 layer hidden layers with ReLU as activation were used, along with Adam as optimizer and parameters defined by grid search. The results of this paper are taken as state of the art (concerning the ANN architecture) because it is recent and demonstrates better accuracy than compared models on this dataset.

4.3 Hybrid models

Hybrid models aim to bridge the gap between pure mathematical hypotheses and black box ANN models. It is theorized that feeding relevant data to the problem (in this case, with the combination of compartmental models with ML and ANNs) can increase the predictive performance of the model.

4.3.1 SVR

Georga et al [Geo+13] used the SVR technique along with compartmental models of the glucose-insulin regulatory system and a patient specific model

of subcutaneous glucose concentration.

Data was taken from 27 T1DM patients being treated with multiple dose insulin therapy in the framework of a EU research project called METABO. Observation period for each subject ranged from 5 to 22 days. Each patient wore the Guardian Real-Time CGM system which records an average glucose value every five minutes. The subjects also wore an armband monitoring body physiological signals from multiple sensors, such as energy expenditure of physical activities or exercise. Food intake information was also collected, with the carbohydrate intake of each meal analyzed.
The predictive model proposed was a combination of compartmental models of the glucose-insulin regulatory system (consisting of insulin models and meal carbohydrate absorption models) and an individual model (for each patient) to predict SC glucose concentration. The SVR technique was employed to learn the glucose metabolism of each patient, given the main dataset features as well as the output of compartmental models as principal input variables. Ten-fold cross validation was implemented to assess the predictive performance of the technique, with RMSE as the error metric. Results were satisfactory , with less than three percent of events in the danger zone.

4.3.2 Ensemble models with CM

Saiti et al [Sai+20] used ensemble methods in combination with compartmental models for blood glucose level prediction in T1DM.
Blood glucose measurements of two T1DM patients monitored for 30 days with a CGM device, along with meal and activity data were provided. Data of four patients from the D1NAMO dataset were also used.
The predictive model proposed was a combination of compartmental models for fast acting insulin absorption simulation along with an ensemble of multiple machine learning models. Data from linear, bagging and boost metaregressor were combined. Results showed that ensembling methods provided better results than individual methods, and that CM increased performance of the model.

4.3.3 Insulin Infusion Advisory System

Zarkogianni et al [Zar+11] developed a personalized insulin infusion advisory system in combination with 2 CMs for the predictive model.
The work referenced above used a CM for insulin kinetics, as well as a CM for glucose absorption from the gut. This data was fed into an RNN with the real time recurrent learning (RTRL) algorithm, which allows the RNN to update weights

while operating. Evaluation was performed on the average T1DM patient's behavior obtained from the UVa T1DM simulator. Results demonstrated superior performance of this hybrid model to other glucose controllers.

In the present work, a hybrid model utilizing three different CMs will be used in combination with state of the art ANNs to combine the best of both worlds. Multilevel modelling has not been used in conjunction with ANNs previously for such purposes.

Chapter references

[Alf+20] Ganjar Alfian et al. "Blood glucose prediction model for type 1 diabetes based on artificial neural network with time-domain features". In: *Biocybernetics and Biomedical Engineering* 40.4 (Oct. 2020), pp. 1586–1599. DOI: 10.1016/j.bbe.2020.10.004.

[Ali+18] Jaouher Ben Ali et al. "Continuous blood glucose level prediction of Type 1 Diabetes based on Artificial Neural Network". In: *Biocybernetics and Biomedical Engineering* 38.4 (2018), pp. 828–840. DOI: 10.1016/j.bbe.2018.06.005.

[Das+12] Elena Daskalaki et al. "Real-Time Adaptive Models for the Personalized Prediction of Glycemic Profile in Type 1 Diabetes Patients". In: *Diabetes Technology & Therapeutics* 14.2 (2012), pp. 168–174. DOI: 10.1089/dia.2011.0093.

[Geo+13] E. I. Georga et al. "Multivariate Prediction of Subcutaneous Glucose Concentration in Type 1 Diabetes Patients Based on Support Vector Regression". In: *IEEE Journal of Biomedical and Health Informatics* 17.1 (2013), pp. 71–81. DOI: 10.1109/titb.2012.2219876.

[Geo+15] Eleni I. Georga et al. "Evaluation of short-term predictors of glucose concentration in type 1 diabetes combining feature ranking with regression models". In: *Medical & Biological Engineering & Computing* 53.12 (2015), pp. 1305–1318. DOI: 10.1007/s11517-015-1263-1.

[Ham+18] Takoua Hamdi et al. "Accurate prediction of continuous blood glucose based on support vector regression and differential evolution algorithm". In: *Biocybernetics and Biomedical Engineering* 38.2 (2018), pp. 362–372. DOI: 10.1016/j.bbe.2018.02.005.

[Hov+04] Roman Hovorka et al. "Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes". In: *Physiological Measurement* 25.4 (2004), pp. 905–920. DOI: 10.1088/0967-3334/25/4/010.

[Mar+19] John Martinsson et al. "Blood Glucose Prediction with Variance Estimation Using Recurrent Neural Networks". In: *Journal of Healthcare Informatics Research* 4.1 (2019), pp. 1–18. DOI: 10.1007/s41666-019-00059-y.

[MRC07] Chiara Dalla Man, Robert A. Rizza, and Claudio Cobelli. "Meal Simulation Model of the Glucose-Insulin System". In: *IEEE Transactions on Biomedical Engineering* 54.10 (Oct. 2007), pp. 1740–1749. DOI: 10.1109/tbme.2007.893506.

[Pap+11] Scott M. Pappada et al. "Neural Network-Based Real-Time Prediction of Glucose in Patients with Insulin-Dependent Diabetes". In: *Diabetes Technology & Therapeutics* 13.2 (2011), pp. 135–141. DOI: 10.1089/dia.2010.0104.

[PG+10] C. Pérez-Gandía et al. "Artificial Neural Network Algorithm for Online Glucose Prediction from Continuous Glucose Monitoring". In: *Diabetes Technology & Therapeutics* 12.1 (2010), pp. 81–88. DOI: 10.1089/dia.2009.0076.

[PGD00] R. S. Parker, E. P. Gatzke, and F. J. Doye. "Advanced model predictive control (MPC) for type I diabetic patient blood glucose control". In: *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No.00CH36334)*. Vol. 5. 2000, 3483–3487 vol.5. DOI: 10.1109/ACC.2000.879216.

[Sai+20] Kyriaki Saiti et al. "Ensemble methods in combination with compartment models for blood glucose level prediction in type 1 diabetes mellitus". In: *Computer Methods and Programs in Biomedicine* 196 (2020), p. 105628. DOI: 10.1016/j.cmpb.2020.105628.

[WWM13] Youqing Wang, Xiangwei Wu, and Xue Mo. "A Novel Adaptive-Weighted-Average Framework for Blood Glucose Prediction". In: *Diabetes Technology & Therapeutics* 15.10 (2013), pp. 792–801. DOI: 10.1089/dia.2013.0104.

[Zar+11] K. Zarkogianni et al. "An Insulin Infusion Advisory System Based on Autotuning Nonlinear Model-Predictive Control". In: *IEEE Transactions on Biomedical Engineering* 58.9 (2011), pp. 2467–2477. DOI: 10.1109/TBME.2011.2157823.

[Zar+15] K. Zarkogianni et al. "Comparative assessment of glucose prediction models for patients with type 1 diabetes mellitus applying sensors for glucose and physical activity monitoring". In: *Medical & Biological Engineering & Computing* 53.12 (2015), pp. 1333–1343. DOI: 10.1007/s11517-015-1320-9.

[Zec+14] C. Zecchin et al. “Jump neural network for online short-time prediction of blood glucose from continuous monitoring sensors and meal information”. In: *Computer Methods and Programs in Biomedicine* 113.1 (2014), pp. 144–152. DOI: 10.1016/j.cmpb.2013.09.016.

Chapter 5

Machine Learning and Neural Networks

In this chapter some general information about machine learning (ML) and neural networks (NNs) will be presented. After introducing the common definitions in this area, learning methods and types of networks will be discussed. We will delve deeper into optimization algorithms, and the common problems of training a model. Finally, justification of the models we used for the present work will be made.

Machine learning, deep learning and artificial intelligence are notions that are closely related. Artificial intelligence (AI) is the term widely used to describe ”clever” software or appliances/machinery. AI (in one of its applications) is used to automate tasks and require the least amount of human input as possible. For the above to happen, the software must learn how to distinguish between tasks and how to execute them more efficiently. Machine learning is the area which studies the training process of models (which are the part of the software that is supposed to ”learn”) and how to improve it. Deep learning is a subset of machine learning which involves more complex models. An illustrative Venn diagram of the above is presented in [GBC16].

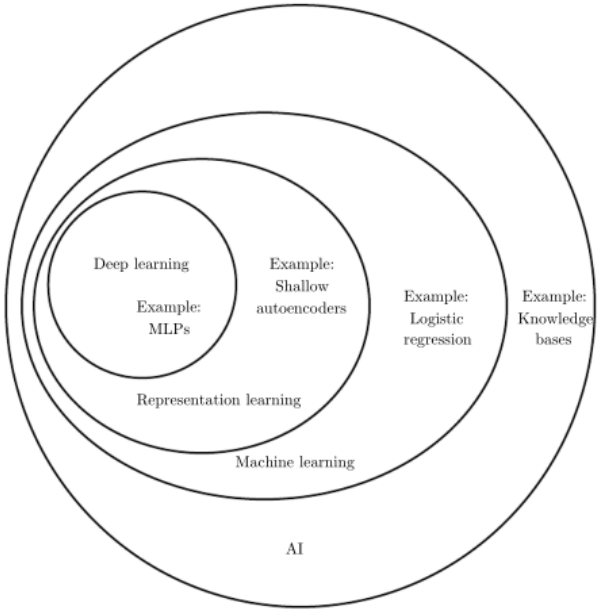

Fig. 5: Venn diagram of ML,AI and Deep Learning. Credits:[GBC16]

5.1 Tasks and Experience

A very useful definition of learning process for a computer program is given by [Mit97].

Definition 1. *A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.*

At this point the notions of *learning* and *training* have to be distinguished from each other. The definition of learning is given above. Training happens simply by feeding data to the program. Learning happens if (and only if) the training leads to higher performance. For example, training can continue until the end of data input, but learning could have stopped at 70% of the data.

5.1.1 Tasks

The desired function of the computer program is called task. Some examples of tasks which can be tackled with ML are the following:

- Classification: This task needs the algorithm to classify input data to a certain category. For example, an image can be classified as an image of a person, a car, an animal etc.

- Regression: The task of prediction future values of numerical data given past values is called regression. Time series forecasting is a characteristic example of a regression task.

- Machine translation: Given an input of strings in a certain language, translate them to a target language.

- Anomaly detection: Scan a collection of data and classify extreme values (eg a value of 2500 in human age variable).

5.1.2 ML methods-Experience

With the training process a program accumulates experience in a given task (irrespective to performance increase or decrease). Four main categories of learning processes are described below.

Supervised Learning

The learning process is called *supervised* if the input data are split into *features* (data to learn from) and *labels* (data to classify into). Features are treated as independent variables and labels as dependent. The purpose of the program if to find the process that maps features to labels. Supervised learning algorithms focus on classification and regression problems. With the increase in input data of recent years (mainly because of higher speed broadband and social media platforms) supervised learning algorithms have gained much popularity. The method is called supervised exactly because the labels are predefined, and this leads to the user ”pointing” the program the direction it must work towards.

Unsupervised Learning

Unsupervised learning involves feeding the program data which can have different types of correlations between them. There are no features and labels in this procedure, the program must learn on its own several properties of the dataset, as well as latent (or hidden) patterns. Unsupervised learning problem categories include clustering, association and dimensionality reduction.

Semi-supervised Learning

Semi-supervised learning uses both labeled and unlabeled data for training (as opposed to supervised, where all training data are labeled). Most frequently, labeled data are in the minority in this learning procedure.

Reinforcement Learning

In reinforcement learning, the program interacts with a dynamic environment, without a clearly defined goal. However, there is a reward/penalty system that measures performance, and the goal is to maximize it. Through repeated trial and error, the program (”agent”) learns the best course of action. Reinforcement learning has been used to create AIs that can defeat the world’s top players in Go, chess and RTS (real-time strategy) games. The last one was extremely challenging, since the number of possible actions in each step are around 10^{26}.

The role of data

Up till now, we have not discussed the quality of data. If one thinks about how the human mind learns, it is useful to try and move the analogy to machine learning. Human mind learns through trial and error, reading and other methods. What would happen if all the books a human reads have wrong methods, or false information? It could not lead to quality learning. Instead, it would ”create” an individual with a false perception of how the world functions. Much like the above, a program cannot learn if not given good data. Data with only one label (in supervised training) for example, would serve no purpose in learning, because there is no way the program would learn to generalize. Data that is too simple and has few patterns would be useless for unsupervised learning problems. We can conclude that the quality of data is of paramount importance to the learning process. Or, as it is summarised in the ML community: ”garbage in = garbage out”. In that respect, data has to be viewed as part of the model and the training process, not as an added part (at least that holds true for the data used in the initial training).

5.2 Artificial Neural Networks-ANNs

Artificial Neural Networks (ANNs) [GBC16], inspired by the function of the human brain, are an algorithmic construct that is at the core of many ML and all Deep Learning applications. They are constructed with building blocks that are called neurons. Neurons can work in parallel and execute calculations in order to solve a problem.

5.2.1 Single layer perceptron

A single layer perceptron (SLP) is a construct which mimics the biological neuron, in an abstract way. Based on McCulloch-Pitts model [ZZ99] an SLP takes an n-dimensional input vector $\vec{x}$ as input and produces a real number y

as a result. The result is calculated by first computing the internal product of $\vec{x}$ with a weight vector $\vec{w}$ (weights are calculated in the training process) and adding (optionally) a bias b. This quantity is then fed into an activation function f, giving the final result y. This can be mathematically illustrated by the following relation

$$\sum_{i=1}^{n}(w_i x_i + b) = \sum_{i=0}^{n} w_i x_i = \vec{w} \cdot \vec{x}$$

$$y = f(\vec{w} \cdot \vec{x})$$

And can be graphically illustrated as in the figure below 6.

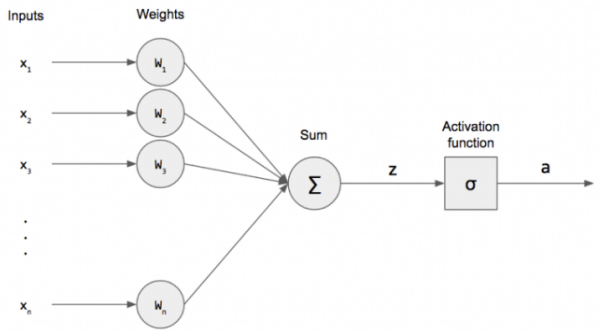

Fig. 6: Single layer perceptron model

Weights of each synapse (n synapses for our example) define how much each variable affects the output value. The activation function decides whether the neuron will be activated or not. In some applications, ANNs have better performance without activation function (every neuron is activated).

5.2.2 Multi layer perceptron-MLP

Deep neural networks are based on Multi Layer Perceptrons (MLPs). Contrary to SLPs which have an input layer and an output layer, MLPs have layers in between, which are called hidden layers. Each neuron of the hidden layer works in the exact same way described above. It treats the output of previous neurons as input and executes the same calculation. Activation functions can be applied to each layer independently. 7 illustrates an MLP with two hidden layers.

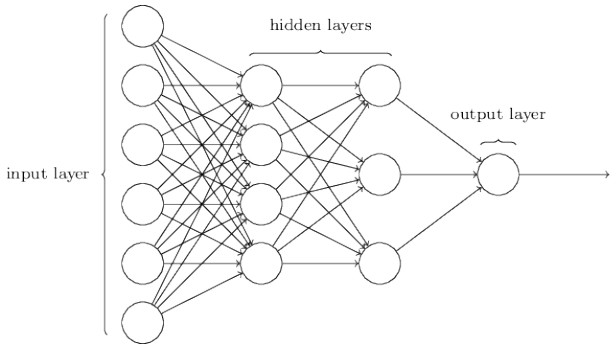

Fig. 7: Multi layer perceptron model

Increasing number of layers and neurons increases the model's number of parameters. There are three ways with which a deep NN can be connected:

- Fully connected: Each neuron of a layer is connected to all neurons of the previous one. Information flows both forwards and backwards in this type.
- Feedforward: In this type, neurons are only connected to the next layer neurons, hence the name (information flow is forward only).
- Feedback or recurrent: In this type of network, exit results can be fed back to the input, creating a dynamical system with oscillatory (even chaotic at times) behavior.

5.3 Training process

In this section several factors that affect training will be discussed. Initially, a reference to data scaling will be made, and then functions that affect training will be presented. Finally, different ways to feed data to the model and optimizing the process will be discussed.

5.3.1 Data scaling

Before constructing the model, one must take into account how the model "sees" the data. All numerical values are just numbers for our model, without it knowing the range for each variable. That would lead to imbalanced training. If, for example, a variable ranges from 0.01 to 0.1, and another one ranges from 100 to 1000, the second variable will have a much larger impact on training, whereas this should not happen. It follows that the data scientist must find a way to make all variables hold the same "weight" (not to be confused with training weights) initially. This can happen by normalization of the data in such a way that they all have similar range of values. There are some ways to do that.

Standard scaling

Standard scaling for a variable x can be done by scaling each value by the following rule:

- Substract from each value the mean value of x.
- Divide the result by the standard deviation.

Or mathematically:

$$x' = \frac{x - \bar{x}}{sd(x)}$$

This method will make all values have 0 as the "center", and there will not be any extreme values (values diverging more than 10 standard deviations from the mean are very rare) and is a solid candidate (one of the first used) for many data analysis problems.

Min-max scaling

Min-max scaling for a variable x can be done by scaling each value by the following rule:

- Substract from each value the minimum value of x.
- Divide the result by the difference of the maximum value minus the minimum value.

Or mathematically:

$$x' = \frac{x - min(x)}{max(x) - min(x)}$$

This method will make all values have the same range (from 0 to 1) and is a very good candidate, because it guarantees same range for every variable. It can be problematic if the minimum or the maximum value are outliers, but works well in general.

Normalizing from -1 to 1

Min-max scaling can be altered to normalize to the range of $[-1, 1]$ with the following rule.

$$x' = 2\frac{x - min(x)}{max(x) - min(x)} - 1$$

This method holds the same advantages of min-max scaling, and it could be used in similar problems.

5.3.2 Loss functions

Loss functions are a very important part of the training process, because they define the penalty applied to each training steps according to deviation from the real values. There are some standard functions that are used in problems such as the one at hand, such as MAE and MSE.

Mean averaged error (MAE)

Mean averaged error (MAE) is given by the following equation:

$$MAE(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)$$

It penalizes each error with a rule of equality (linear to the deviation from the actual value). It is a standard loss function and widely used. However, when we want to penalize larger errors even more, we turn to Mean Squared Error.

Mean squared error (MSE) and root mean squared error (RMSE)

Mean Squared Error (MSE) is given by the following equation:

$$MSE(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$$

While RMSE is given by $RMSE(y, \hat{y}) = \sqrt{MSE(y, \hat{y})}$. These function penalize larger error more than MAE, which is very useful for medical purposes (especially when a very large prediction error can be fatal) and are the method of choice for time series forecasting problems.

Custom loss functions

Custom loss functions can be used if some knowledge of the problem can be implemented into the training process. In [Kar+17], physical laws involving heat transfer are added as a penalty term to the loss function. This showed improvement in both training time and performance, and it is the go-to method when one wants to ”escape” from black-box models and train the model with some rule that is experimentally proven. It is not recommended to use very complex or very specific laws, because that can lead the network to become little more than a theoretical simulator.

5.3.3 Activation functions

Activation functions are used to "control" the output of a layer, and are very useful when one knows the range that the variables must lie into. Most commonly used functions include sigmoid, hyperbolic tangent, ReLU and softmax.

- Sigmoid function: It is given by the equation $\sigma(x) = \frac{1}{1+e^{-x}}$ and maps values to the $[0, 1]$ range. Sigmoid function has a property which is undesirable in many cases, since it causes a collection of smaller values closer to 0 and larger values closer to 1. This leads to vanishing gradient problem, thus not making it the ideal case in many applications.

- Hyperbolic tangent: It is given by the equation $tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$ and maps values to the $[-1, 1]$ range. It has an almost linear behavior for values close to 0, whereas very large and small (algebraically) values are mapped to 1 and -1 respectively. This can also cause vanishing gradients.

- Rectified Linear Unit (ReLU): ReLU is a simple function which is given by the equation $f(x) = max(0, x)$. It maps negative values to 0 and leaves positive ones unaffected. Very useful when negative values are undesirable, and in this case it does not cause the vanishing gradient problem.

- Softmax: Softmax function is commonly used in the output layer to normalize output values in the $[0, 1]$ range. If there are m neurons in the output layer, it is given by $Softmax(y_i) = \frac{e^{y_i}}{\sum_{i=0}^{m} e^{y_i}}$.

At this point it should be noted that the choice of activation function has to take into account the type of scaling that has been done to the data. For example, min-max scaling and ReLU bode well, while the same scaling with $tanh$ can lead to erroneous results. If our data is guaranteed to be positive, then the choice of ReLU and no activation is virtually the same.

5.3.4 Optimization

Mathematically, the goal of the neural network is the minimization of the loss function. Let $J(\theta)$ be the loss function where θ is a vector of parameters of the network. By updating these parameters to the direction of $-\nabla_\theta J(\theta)$ one can expect that the algorithm will end up at a local stable minimum. This method is called gradient descent. In an ideal world, nothing else would be needed. From the point of mathematics, the model is solved. However, gradient descent can be very computationally expensive, especially if the dataset is very large.

Stochastic gradient descent (SGD)

Stochastic gradient descent attempts to resolve the problem of computational expense by approximating the gradient of the loss function with the gradient at

a simple example, updating weights with the method below (η is the step with which the weights change, also called learning rate):

$$w = w - \eta_i(w)$$

The update is done many times as the whole training set is fed into the algorithm. If the dataset ends before convergence, it can be shuffled randomly and fed again to the model. Algorithmically, it can be illustrated as follows:

- Choose a random starting weight vector w and a (not random) learning rate η.
- While NOT approximate minimum obtained:
  - Randomly shuffle samples.
  - For each sample, DO:
    $w = w - \eta_i(w)$

Changing the learning rate (adaptive learning rate method) has been shown to yield results as good as gradient descent, but in less time. In practice, SGD is used much more frequently than classic batch gradient descent.

Mini-batch gradient descent

Mini-batch gradient descent is a compromise between batch gradient descent and stochastic gradient descent. It is essentially batch gradient descent but the parameter update is not made after a pass of the whole dataset, but incrementally on small parts (mini-batches). In practice, it is the one most frequently used.

Online training

Online training is mini-batch gradient descent with batch size equal to one. It is a computationally expensive method, however it has the advantage of updating parameters in real time. In our testing, since the datasets were not prohibitively large, it was also used, but proved to yield inferior results to mini-batch with batch size of 16.

Optimizers

Optimization algorithms, or optimizers for short, are the algorithms which define the rule of weight update. Gradient descent and SGD are optimization algorithms, but not the only ones. Small modifications can be made to them, to find ones better suited to each problem. We will briefly describe SGD with momentum, and the optimizer of choice in our problem, ADAM.

SGD with momentum

SGD with momentum (inspired from physical equations) can be described with the following equations:

$$u_t = \gamma u_{t-1} + \eta \nabla_w J_i(w_n; x^{i:i+n}, y^{i:i+n})$$

$$w = w - u_t$$

u is the velocity vector (normalized to same units as θ) and $\gamma \in (0, 1]$ is the momentum coefficient which decides the percentage with which previous gradients contribute to weight updates. Optimizing the momentum has been shown to yield better results than plain SGD.

ADAM (ADAptive Moment Estimation)

ADAM was first presented by Diederik P. Kingma and Jimmy Lei Ba [KB14], and is another adaptive learning rate method. It is described by the system of equations:

$$g_{t,i} = \nabla_{w_t} J(w_{t,i})$$

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$$

$$u_t = \beta_2 u_{t-1} + (1 - \beta_2) g_t^2$$

$$w_{t+1} = w_t - \frac{\eta}{\sqrt{u_t} + \varepsilon} m_t$$

Illustrated in pseudocode:

Require: α: Stepsize
Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates for the moment estimates
Require: $f(\theta)$: Stochastic objective function with parameters θ
Require: θ_0: Initial parameter vector
  $m_0 \leftarrow 0$ (Initialize 1st moment vector)
  $v_0 \leftarrow 0$ (Initialize 2nd moment vector)
  $t \leftarrow 0$ (Initialize timestep)
  while θ_t not converged **do**
    $t \leftarrow t + 1$
    $g_t \leftarrow \nabla_\theta f_t(\theta_{t-1})$ (Get gradients w.r.t. stochastic objective at timestep t)
    $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Update biased first moment estimate)
    $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Update biased second raw moment estimate)
    $\widehat{m}_t \leftarrow m_t / (1 - \beta_1^t)$ (Compute bias-corrected first moment estimate)
    $\widehat{v}_t \leftarrow v_t / (1 - \beta_2^t)$ (Compute bias-corrected second raw moment estimate)
    $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \widehat{m}_t / (\sqrt{\widehat{v}_t} + \epsilon)$ (Update parameters)
  end while
  return θ_t (Resulting parameters)

Fig. 8: ADAM description in pseudocode. Credit:[KB14]

5.3.5 Problems in training procedure

Apart from the vanishing gradient problem which was discussed above and leads to very slow training, there are problems regarding model complexity and data size. Finishing this section, underfitting and overfitting will be discussed.

Underfitting

Underfitting is a problem that simply means the training process has not led to convergence. Statistically speaking, it is a problem of low variance and high bias. Reasons of underfitting include:

- Not a good set of features chosen to train.
- Noise in the data.
- Too simple (low complexity) model.
- Too little training time (insufficient number of epochs for convergence).

Techniques to combat underfitting are straightforward. Increasing model complexity and number of epochs, cleaning data and better feature engineering should combat the problem.

Overfitting

Overfitting is the opposite of underfitting, thus can be described as high variance and low bias.

- Low quality or too few data.
- Higher model complexity than required.
- High number of epochs.

Techniques to combat overfitting are trickier than underfitting (especially if there is a lack of data). Reducing model complexity and number of epochs can help, or use an early stopping criterion. Dropout layers can also help (randomly drop a percent of inputs from previous hidden layer). If data are missing, then they can be augmented via oversampling methods (SMOTE, for example), but such methods are a last resort and should be used with caution.

5.4 Deep learning-Neural Network types

Apart from the fully connected MLP, there are two importan categories of Neural Networks to discuss, one of which is used in one of our models. The Convolutional Neural Network (CNN) and the Recurrent Neural Network (RNN).

5.4.1 Convolutional Neural Networks-CNNs

Convolutional Neural Networks (CNNs) are a class of networks that use the mathematical process of convolution instead of simple vector/matrix multiplication in one or more layers [GBC16]. Choosing a kernel (or filter) convolution is used on the input of the convolutional layer to extract what is called a feature map. CNNs are the gold standard for image classification and computer vision, because they excel at recognizing characteristics such as contours of objects, given the right filter. They are also used in time series prediction tasks, but our testing showed recurrent and fully connected networks to perform better in the given datasets.

5.4.2 Recurrent Neural Networks-RNNs

Recurrent Neural Networks (RNNs) are used widely in time series forecasting and Natural Language Processing (NLP) problems. RNNs feed output data back to the input and hidden layers, leading to parameter updates. They have a kind of ”memory” that saves information of executed calculations, and is accessible to all layers. The two types of networks that are of interest are LSTM and GRU.

Long Short-Term Memory networks-LSTM

Long Short-Term Memory networks, or LSTMs for short, are a type of RNN that has longer memory than simple RNN. It was first proposed in [HS97]. The illustration of an LSTM cell is given in figure 9. The cell state stores information that must be fed back to the model. LSTM has three types of gates. Gates add or remove information to the cell state. The function of the forget gate is to delete unimportant information (eg information for which the sigmoid function returns values close to zero). The input gate passes the hidden state and current input into a sigmoid function and a tanh function. Multiplying both results ”decides” which information will be kept. The new cell state is the multiplication of the old cell state to the forget vector added to the input gate. Finally, the output gate decides what the next hidden state will be. The previous hidden state and the current input is passed into a sigmoid function, and multiply that result with the new cell state passed to a tanh function. After the output gates makes the decision, the new hidden state is fed back to the model.

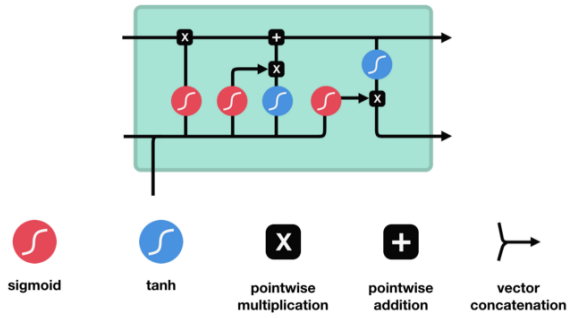

Fig. 9: LSTM illustrated. Credit:[Lst]

Gated Recurrent Unit-GRU

Gated Recurrent Units, or GRUs for short, is a newer generation of RNNs compared to LSTM, proposed by Cho et al [Cho+14]. It is illustrated in figure 10 It has one less gate compared to LSTM and has no cell state, as information is transferred with usage of the hidden state only. Its gates are called reset gate and update gate. The update gate uses a sigmoid function to decide which information to keep and what new information to add, essentially combining functions of input and forget gates of LSTM. The reset gate uses sigmoid to decide which past information to drop. GRU has fewer complexity, leading to fewer calculations than LSTM. However, there is no clear better choice. In our problem, LSTM was prone to overfitting, and GRU was the clear winner. This could change with larger datasets.

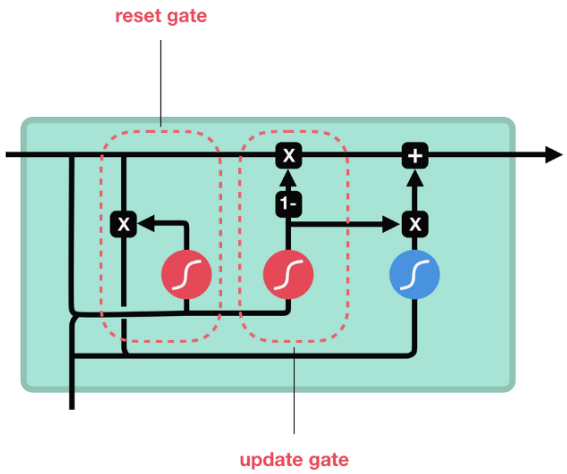

Fig. 10: GRU illustrated. Credit:[Lst]

Chapter references

[Cho+14] Kyunghyun Cho et al. "On the Properties of Neural Machine Translation: Encoder–Decoder Approaches". In: *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 103–111. DOI: 10.3115/v1/W14-4012. URL: https://www.aclweb.org/anthology/W14-4012.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. The MIT Press, 2016. ISBN: 0262035618.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-term Memory". In: *Neural computation* 9 (Dec. 1997), pp. 1735–80. DOI: 10.1162/neco.1997.9.8.1735.

[Kar+17] Anuj Karpatne et al. "Physics-guided Neural Networks (PGNN): An Application in Lake Temperature Modeling". In: (Oct. 2017).

[KB14] Diederik Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *International Conference on Learning Representations* (Dec. 2014).

[Lst] *Illustrated Guide to LSTM's and GRU's: A step by step explanation*. https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21.

[Mit97] Tom Mitchell. *Machine Learning*. New York: McGraw-Hill, 1997. ISBN: 9780071154673.

[ZZ99] Ling Zhang and Bo Zhang. "A geometrical representation of McCulloch-Pitts neural model and its applications". In: *IEEE Transactions on Neural Networks* 10.4 (1999), pp. 925–929. DOI: 10.1109/72.774263.

Chapter 6

Methodology

The current chapter includes presentation of the developed methodology. The first section involves presentation of data and patient characteristics. The second section includes the conceptual framework of the developed methodology. Building blocks of the hybrid model developed will be discussed, along with the preprocessing framework. The third section involves mention of the neural network models combined with the compartment models. Both the recurrent and the fully-connected model architectures (along with optimizers, error and evaluation metrics used) will be presented. The fully connected model [Alf+20] is regarded as state of the art because of being recent and presenting superior performance to previous ANN formalisms in this context. Its inclusion in a hybrid framework aims to go beyond the current state of art.

6.1 The data and conceptual framework

For the development and evaluation of glucose prediction models two datasets have been used.
Real data from twelve T1DM patients, collected over a 9-day observation period. Patient characteristics, as well as information for each patient's dataset are given in figures 11 and 12. In silico data produced with the UVA/PADOVA Type 1 Diabetes Simulator which was submitted to FDA in 2013 [Man+14]. This dataset comprises of 10 virtual patients for each age category (children, adolescents and adults), totalling 30 patients.

	Patient	Age (years)	Gender	Diabetes duration (years)	BMI(kg/m2)B	HbA1c
0	1	3	M	2	18.9	6.3
1	2	3	M	1	16.69	8
2	3	9	F	5	16.47	6.6
3	4	13	F	12	24.29	7.5
4	5	18	F	12	18.25	7.2
5	6	18	M	10	18.56	5.7
6	7	18	M	16	19.83	7.8
7	8	20	F	10	24.41	6.2
8	9	25	F	17	29.24	8.3
9	10	>35	M	-	-	-
10	11	35	M	22	19.84	5.7
11	12	38	M	19	27.44	6
12	Mean ± SD	19.83 ±12.28	-	12.67 ± 7.74	22.00 ± 4.88	6.78 ± 0.94

Fig. 11: Patients' age, gender and diabetes duration information

	Patient	Data rows	min glucose (mg/dl)	max glucose (mg/dl)	Mean glucose (mg/dl)	SD	% Hypoglycemic events (<70mg/dl)	% Hyperglycemic events
1	1.0	2730.0	40.0	400.0	161.31	61.60	4.100	33.91
2	2.0	2653.0	46.0	400.0	171.89	63.43	2.290	40.33
3	3.0	2798.0	54.0	338.0	129.21	46.97	2.930	14.43
4	4.0	2718.0	76.0	342.0	163.34	42.40	0.000	31.12
5	5.0	2674.0	46.0	288.0	143.93	47.71	4.930	21.31
6	6.0	2821.0	46.0	196.0	104.29	27.23	9.640	0.31
7	7.0	2449.0	44.0	260.0	131.61	44.80	8.860	15.43
8	8.0	2819.0	54.0	214.0	117.50	28.47	3.610	1.52
9	9.0	2805.0	68.0	372.0	183.18	59.35	0.392	48.44
10	10.0	2818.0	40.0	310.0	150.04	57.39	4.750	30.34
11	11.0	2664.0	52.0	280.0	129.34	37.34	2.850	8.89
12	12.0	2802.0	40.0	330.0	149.21	51.92	3.310	26.15

Fig. 12: Information regarding individual patients' sets of data

The conceptual framework used in the present work is a hybrid model comprising of compartment models (CMs) and Neural Networks (NNs). Three CMs are being utilized with the purpose of feeding data to the model that contain information about certain parts of the glucose homeostasis procedure.

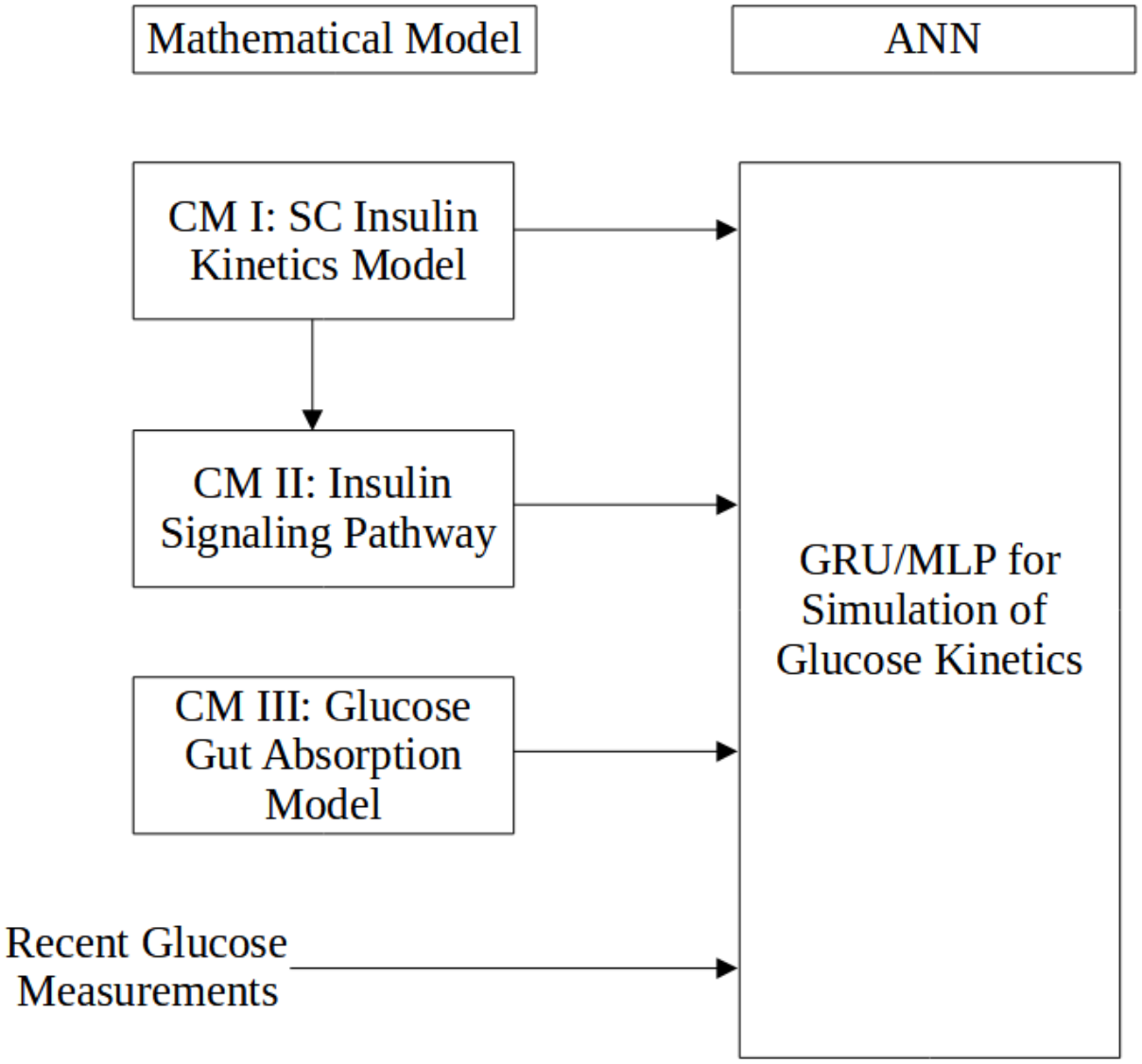

Fig. 13: Conceptual diagram of the hybrid model constructed in the present work

One column of the provided data contains information about the amount of carbohydrates in each meal. After digestion, these carbohydrates are converted into glucose and absorbed into the bloodstream. The output of CM I provides postprandial raise of blood glucose levels due to the digestive process, and is fed directly to the ANN. CM III accepts insulin as input and provides the amount of subcutaneous (SC) insulin as output. This output can be fed directly to the ANN, although the method of choice in the present framework is to provide the output of CM III as input to CM II, which returns glucose uptake via multilevel modelling of the insulin signaling pathway. CM outputs are both added to the glucose value (for the insulin signaling pathway and gut absorption model outputs) and used as separate feature columns in the datasets. Feature selection takes place further

down the preprocessing procedure.
Figure 12 demonstrates imbalance in the data between hypoglycemic and hyperglycemic events, and that fact is expected to lead to imbalance in predictive performance between such events, which will be further commented in discussion of the results.

Initially, each real patient's dataset contains eight feature columns, providing the information below.

- Date: The date when the measurement took place.
- time(min): Time in minutes after 12:00 AM on the first day of measurements (eg 27/10/07 for patient 1). Each measurement takes place exactly 5 minutes after the previous one.
- time(hour): Exact time of day.
- Glucose measurements from CGMS: The glucose measurements from the Continuous Glucose Monitoring device. Measured in $\frac{mg}{dl}$.
- Blood Glucose Levels: Glucose levels measured from blood sample. Only available certain times of the day. Same units as the previous column.
- Insulin Bolus(U): Insulin that is given to the patient before meals. Measured in international units of insulin (U).
- Basal Insulin(U/hour): The rate at which insulin is being continuously injected to the patient. Measured in insulin units per hour.
- Time and amount of carbohydrates ingested (gr): The amount of carbohydrates in a meal the patient had. Measured in grams.

Descriptive statistics are initially performed on the dataset to check for missing and/or wrong values.

	time(min)	Glucose Measurements from CGMS	Blood Glucose Levels	Insulin Bolus (U)	Basal Insulin (U/hour)	Time and amount of carbohydrates ingested (gr)
count	2870.000000	2870.000000	2870.000000	2870.000000	2870.000000	2870.000000
mean	7215.500000	153.450174	4.523693	0.016411	0.299930	0.251568
std	4143.209806	69.410304	29.533717	0.112541	0.059567	2.351951
min	43.000000	0.000000	0.000000	0.000000	0.200000	0.000000
25%	3629.250000	112.000000	0.000000	0.000000	0.250000	0.000000
50%	7215.500000	150.000000	0.000000	0.000000	0.300000	0.000000
75%	10801.750000	194.000000	0.000000	0.000000	0.350000	0.000000
max	14388.000000	400.000000	406.000000	2.300000	0.400000	38.000000

Fig. 14: Descriptive statistics of sample dataset (patient 1)

Statistics of the time(min) variable are of no importance. There seem to be no extraordinarily high values in the other columns, and no NaN or None values are

present. However, there is an issue to be resolved with respect to the ”Glucose Measurements from CGMS” column. The fact that measurements are being taken continuously at five minute intervals does not bode well with the presence of events containing values of 0.0. The above issue would be trivialized if blood glucose measurements existed for each such event. However, that is not the case, as this issue persists for 128 rows of the sample dataset. There are two distinct approaches that can be utilized to overcome the issue. The first one being to remove such rows from the datasets entirely, and the second being to procede with a form of missing values imputation.

At first glance, both of these options seem valid. 128 rows is less than 5% of the sample dataset, and the data will be sufficiently smoothed out after removing them. However, the second method will be chosen, for reasons described below.

This is a time series forecasting problem with data that is uniform in time (five minutes distance between each observation). There is value to that uniformity, as the time variable can be removed from the analysis completely and one can focus solely on previous values of certain variables to predict the target variable. Preserving that uniformity will be preferred, thus imputation will become a part of the preprocessing stage.

There is a variety of methods that can be utilized to impute missing values in a dataset, but the choice must be consistent with the nature of the data. Since each observation is only taken five minutes after the last one, the upward or downward trend should be incorporated in the imputing method. In this regard, substituting each missing value with the mean of observations is automatically excluded as an option, since it could lead to irregularities and false information (for example, a hypoglycemic episode could be falsely reported as self-resolved after five minutes), as well as falsely large gradients which can lead to training errors.

Another method would be to impute each missing value with its previous one plus or minus a small residual value. This could provide a very good solution, if the missing values were isolated. However, an analysis reveals that there are streaks of zero values (which could indicate a temporary fault in the sensor), thus neither this option would suit our needs.

The method of choice is the nearest neighbors imputer. This will ensure that each missing value will be interpolated by its neighbors, sufficiently smoothing the data. With this option, a hypoglycemic event will not be falsely masked, neither will a hyperglycemic one. Given that one of the most important goals of the model is to predict those episodes in hope of preventing them, this method is a solid choice.

After imputing, descriptive statistics can be produced a second time, to check the current state of irregularities.

	time(min)	Glucose Measurements from CGMS	Blood Glucose Levels	Insulin Bolus (U)	Basal Insulin (U/hour)	Time and amount of carbohydrates ingested (gr)
count	2870.000000	2870.000000	2870.000000	2870.000000	2870.000000	2870.000000
mean	7215.500000	161.319414	4.523693	0.016411	0.299930	0.251568
std	4143.209806	60.081888	29.533717	0.112541	0.059567	2.351951
min	43.000000	40.000000	0.000000	0.000000	0.200000	0.000000
25%	3629.250000	120.000000	0.000000	0.000000	0.250000	0.000000
50%	7215.500000	158.000000	0.000000	0.000000	0.300000	0.000000
75%	10801.750000	194.000000	0.000000	0.000000	0.350000	0.000000
max	14388.000000	400.000000	406.000000	2.300000	0.400000	38.000000

Fig. 15: Descriptive statistics of the sample dataset after imputation

The mean value has increased and standard deviation has been reduced, which is to be expected since the imputed values are greater than 40 $\frac{mg}{dl}$, and zero values created greater divergence from the mean. The statistics of other columns remain unchanged. The next steps will utilize the transformed columns.

6.1.1 In silico data

In silico data is a collection of 30 in patients, divided equally between age groups (children, adolescents and adults). Resolution time is one minute instead of five, and the insulin units are not in international units, but rather in $\frac{pg}{ml}$. After converting the resolution time to 5 minutes, and the units of insulin units to IU, the datasets are of the following form.

	Time (min)	Blood Glucose	CGMS	Insulin	Carbohydrates intake
0	1	141.204660	141.177617	0.00576	0.0
1	6	141.204653	140.668986	0.02880	0.0
2	11	141.204535	140.117665	0.02880	0.0
3	16	141.204013	139.395560	0.02880	0.0
4	21	141.202634	138.495398	0.02880	0.0

Fig. 16: In silico dataset example

6.2 Data cleaning and multilevel modelling

One of the most important factors in the training process of a neural network is the data provided. Raw data tends to have two main issues, one of which has already been addressed above. These issues consist of missing, incomplete or erroneous values in one or more variables (addressed via the process of

imputation), and noise in the data, requiring a filter for proper smoothing.

Data noise is an important factor, for reasons similar to the issues arising from the problem of missing data. A neural network will converge faster and with better performance if the gradient values of the variables are not very large (ie if the data is ”smoother”). There are various techniques to smooth the data, and the one of choice is the same one used in [Alf+20], the Savitzky-Golay filter.

6.2.1 Savitzky-Golay filter

In their paper [SG64], Savitzky and Golay proposed a method of data smoothing based on local least-squares polynomial approximation. The method they used is as follows:

- Fit a polynomial to a collection of input samples.
- Evaluate it at a single point in the approximation interval.

The above method proved to be equivalent to discrete convolution of fixed impulse response. Filters derived by this method are widely used in various scientific sectors. The above filters were initially created with the purpose of smoothing noisy data obtained from chemical spectrum analyzers. It was demonstrated that least squares smoothing lead to noise reduction while maintaining the shape and height of waveform peaks (in their case the peaks were Gaussian, but the results can be generalized). The Savitzky-Golay filter is widely used in digital data.
The implementation of Savitzky-Golay filter is trivial in Python, as the SciPy package includes a function which filters data by this method. Following [Alf+20] and [Tur+13], a filter with window size of 17 (testing proved this to produce better results with our data than the size of 15 in those papers) and fitting a first order polynomial was used.

6.2.2 Compartment Models

The columns that provide the information of the amount of insulin that is injected into the patient and the carbohydrates received with each meal are very useful parameters in the prediction of glucose measurements. However, there are scientific methods which can be used to provide additional information regarding the effect those parameters have on insulin levels. Namely, there are two important questions:

- How can the effect of carbohydrates ingested on the rise of blood glucose levels be quantified?

- What is the rule that gives the absorption of the injected insulin into the bloodstream?

The mathematical models that answer the equations above are known as compartment models (CMs). Three models will be used. Two will be straightforward, to answer the above equations. The third one, a multilevel model involving intracellular response to insulin, shall be explained in more detail.

6.2.3 Glucose gut absorption compartmental model

The compartmental model of glucose gut absorption is the same as in [MRC07]. A system of differential equations to be solved gives the gut absorption of glucose to the bloodstream. These equations are:

$$Q_{sto}(t) = Q_{sto,1}(t) + Q_{sto,2}(t), Q_{sto}(0) = 0 \tag{6.2.1}$$

$$\frac{dQ_{sto,1}(t)}{dt} = -k_{gri} \cdot Q_{sto,1}(t) + D \cdot \delta(t), Q_{sto,1}(0) = 0 \tag{6.2.2}$$

$$\frac{dQ_{sto,2}(t)}{dt} = -k_{empt}(Q_{sto}) \cdot Q_{sto,2}(t) + k_{gri} \cdot Q_{sto,2}(0) = 0 \tag{6.2.3}$$

$$\frac{dQ_{gut}(t)}{dt} = -k_{abs} \cdot Q_{gut}(t) + k_{empt}(Q_{sto}) \cdot Q_{gut}(0) = 0 \tag{6.2.4}$$

$$R\alpha(t) = \frac{f \cdot k_{abs} \cdot Q_{gut}(t)}{BW}, R\alpha(0) = 0 \tag{6.2.5}$$

$$\tag{6.2.6}$$

Q_{sto} is the amount of glucose in the stomach, divided into solid (index "1") and liquid phase (index "2"). Q_{gut} is the glucose mass in the intestine. All the above quantities are measured in mg. $k_{gri}(min^{-1})$ is the rate of grinding. $k_{empt}(Q_{sto})(min^{-1})$ is the gastric rate of emptying, which depends on the stored glucose in the gut. $k_{abs}(min^{-1})$ is the rate with which the istentines absorb glucose. f signifies the fraction of absorbed glucose in gut that actually appears in plasma. D is the amount of carbohydrates in mg, BW is the body weight in kilograms, and $R\alpha$ (mg/kg/min) is the appearance rate of glucose in the plasma.
The way the above model is used is as follows. An ODE solver is constructed, which is designed to give 0 as an exit if the amount of carbs in the previous times step is zero. Then, calculating the amount of glucose in the gut, also taking into account the amount which was present in the previous time step, we solve for $R\alpha$. The glucose extracted from this procedure is added to the value of the next time step in our dataset.

6.2.4 Subcutaneous insulin compartmental model

The compartmental model of subcutaneous insulin is one of those presented in [NC00]. The system of differential equations describing the model is the following:

$$\dot{I}_{sc1}(t) = -(k_d + k_{a1}) \cdot I_{sc1}(t) + I_{SC}(t), I_{sc1}(0) = I_{sc1ss} \tag{6.2.7}$$
$$\dot{I}_{sc2}(t) = k_d \cdot I_{sc2}(t), I_{sc2}(0) = I_{sc2ss} \tag{6.2.8}$$
$$I_u(t) = k_{a1} \cdot I_{sc1}(t) + k_{a2} \cdot I_{sc2}(t) \tag{6.2.9}$$
$$\tag{6.2.10}$$

I_{sc1}, I_{sc2} are the subcutaneous insulin quantities that exist in monomer and polymer form respectively. $I_{SC}(t)$ (pmol/kg/min) is the subcutaneous insulin injection rate. $k_d = 0.0164 min^{-1}$ is the constant rate of insulin decomposition, and $k_{a1} = 0.0018 min^{-1}$, $k_{a2} = 0.0182 min^{-1}$ represent the absorption rates of monomer and polymer insulin respectively. Finally, I_u is the insulin that enters the bloodstream at a given time step.
The above model takes the insulin that is injected in the patient at a given moment in time, and converts it into bloodstream insulin. The way is it used in preprocessing is similar to that of the previous model. Constructing an ODE solver, the bloodstream insulin is calculated and entered at the next time step in a new data column.

6.2.5 Insulin signaling pathway CM

The previous CM gives the amount of insulin in the bloodstream at a given moment. However, we can go one step further to translate this insulin into glucose uptake, given the amount of blood glucose present. That exact purpose is served by an insulin signaling pathway CM. The model used here will be from [GGJ13]. The multilevel model's process is illustrated below:

The whole body model for T1DM is expanded to include intracellular level information of insulin signaling to enhance glucose uptake via GLUT4 translocation, relating insulin in the interstitial fluid with glucose uptake in adipose tissue. The resulting glucose uptake is calculated as the sum of glucose uptake in muscle and adipose tissue.
With this final model, every available information of the data can be converted into glucose, aiding to the predictive capabilities of the models to be used. The system of ODEs describing this CM is the following:

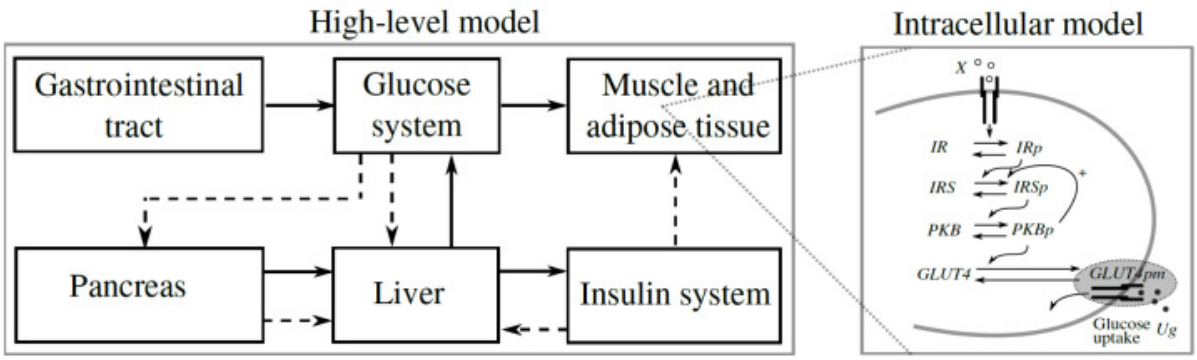

Fig. 17: Insulin pathway CM. Credit:[GGJ13]

$$\frac{dIR}{dt} = k_{1,b}IR_P - (k_{1,f}Ins + k_{1a,Basic}IR)$$
$$\frac{dIR_P}{dt} = -k_{1,b}IR_P + (k_{1,f}Ins + k_{1a,Basic}IR)$$
$$\frac{dIRS}{dt} = k_{2,b}IRS_P - k_{2,f}IR_PIRS$$
$$\frac{dIRS_P}{dt} = -k_{2,b}IRS_P + k_{2,f}IR_PIRS$$
$$\frac{dPKB}{dt} = k_{4,b}PKB_P - k_{4,f}IRS_PPKB$$
$$\frac{dPKB_P}{dt} = -k_{4,b}PKB_P + k_{4,f}IRS_PPKB$$
$$\frac{dGLUT4_C}{dt} = k_{5,b}GLUT4_M - (k_{5,f}PKB_PGLUT4_C + k_{5,Basic}GLUT4_C)$$
$$\frac{dGLUT4_M}{dt} = -k_{5,b}GLUT4_M + (k_{5,f}PKB_PGLUT4_C + k_{5,Basic}GLUT4_C)$$

6.3 Neural network architecture

After preprocessing and extracting extra features from the output of CMs, two different ANNs were used, each with different data columns. All data were normalized using min-max scaling, taking values from 0 to 1.

Model version 1: Recurrent model (GRU)

The first model used a 16-neuron GRU layer with ReLu as activation, followed by a fully connected 16-neuron Dense layer with ReLu, and a one neuron Dense as output layer. Three feature columns were used, after performing gridsearch:

- Glucose, our target variable (after adding the output of all relevant CMs).
- The output of the glucose gut absorption CM.
- The output of the insulin signaling pathway CM.

Model version 2: Multi layer perceptron (MLP)

The state of the art model [Alf+20]: Two fully connected 100-neuron Dense layers with no activation function, and a one-neuron Dense as output layer. Only the glucose column was used, since addition of features had a negative impact on performance.

Training scheme

For both models, the following training scheme was used (a 5 day train/4 day test split was followed). Training for a maximum of 1000 epochs, stopping prematurely if no improvements were made after 10 epochs. Learning rate of 5e-4 was used for both models, with Adam as optimizer.

After presentation of methodology, the conclusion of the current chapter paves the way for presenting evaluation metrics and results in the next one.

Chapter references

[Alf+20] Ganjar Alfian et al. "Blood glucose prediction model for type 1 diabetes based on artificial neural network with time-domain features". In: *Biocybernetics and Biomedical Engineering* 40.4 (Oct. 2020), pp. 1586–1599. DOI: 10.1016/j.bbe.2020.10.004.

[GGJ13] Winston Garcia-Gabin and Elling W. Jacobsen. "Multilevel Model of Type 1 Diabetes Mellitus Patients for Model-Based Glucose Controllers". In: *Journal of Diabetes Science and Technology* 7.1 (Jan. 2013), pp. 193–201. DOI: 10.1177/193229681300700125.

[Man+14] Chiara Dalla Man et al. "The UVA/PADOVA Type 1 Diabetes Simulator". In: *Journal of Diabetes Science and Technology* 8.1 (2014), pp. 26–34. DOI: 10.1177/1932296813514502.

[MRC07] Chiara Dalla Man, Robert A. Rizza, and Claudio Cobelli. "Meal Simulation Model of the Glucose-Insulin System". In: *IEEE Transactions on Biomedical Engineering* 54.10 (Oct. 2007), pp. 1740–1749. DOI: 10.1109/tbme.2007.893506.

[NC00] Gianluca Nucci and Claudio Cobelli. "Models of subcutaneous insulin kinetics. A critical review". In: *Computer Methods and Programs in Biomedicine* 62.3 (July 2000), pp. 249–257. DOI: 10.1016/s0169-2607(00)00071-7.

[SG64] Abraham. Savitzky and M. J. E. Golay. "Smoothing and Differentiation of Data by Simplified Least Squares Procedures." In: *Analytical Chemistry* 36.8 (July 1964), pp. 1627–1639. DOI: 10.1021/ac60214a047.

[Tur+13] Kamuran Turksoy et al. "Hypoglycemia Early Alarm Systems Based on Multivariable Models". In: *Industrial & Engineering Chemistry Research* 52.35 (May 2013), pp. 12329–12336. DOI: 10.1021/ie3034015.

Chapter 7

Results and Discussion

The present chapter contains evaluation criteria, presentation of results and discussion. As continuation to the previous chapter, where the conceptual framework concerning data preprocessing and the theoretical part of the multilevel modelling approach utilized in the developed hybrid model, evaluation criteria comprise the initial discussion of the current chapter. Namely, root mean square error (RMSE) score, Pearson's correlation coefficient (CC) and Continuous Glucose Error Grid Analysis (CG-EGA) will be discussed, and arguments will be given justifying choice of each one as a valid evaluation criterion for the problem of forecasting future glucose values in T1DM patients. Results will be presented both in graphs and in tables, providing collective results for each patient (both real and in silico) regarding each one of the above criteria. Further into the chapter, discussion on the quality of results for both models will take place, both standalone and in comparison to one another. Predictive performance is expected to degrade as the prediction horizon increases, and this topic of discussion will conclude the chapter.

7.1 Evaluation criteria

Prediction models to be used in health data cannot be evaluated by only one criterion. If that were the case, the sole criterion would have to belong in the statistical category of metrics, or metrics developed solely for the application at hand. However, both types of criteria are necessary for complete evaluation. Statistical/mathematical criteria present metrics for performance evaluation that can also indicate the bias and variance of the method used, and whether overfitting or underfitting is present. Special metrics developed for health datasets are more purpose-specific, and provide an insight to the real world performance and implementation of the model. The ultimate purpose for the problem at hand is avoidance of hypoglycemic and hyperglycemic events. The metric used to assess

performance for that purpose is CG-EGA. Mathematical and statistical metrics utilized are RMSE score and Pearson's Correlation Coefficient (CC). In the present section these metrics will be discussed.

7.1.1 Regression metric - RMSE

RMSE belongs to the category of regression metrics, along with Mean Absolute Error (MAE) and Mean Squared Error (MSE). To justify usage of one metric over the other two, all of them will be briefly discussed in this section. Introductory, let y be the vector containing real values of the variable to be predicted, and $\hat{y}$ the vector containing predicted values. Let them both be of dimension N. Metrics will be presented using this notation.

MAE

MAE, as its name suggests, is calculated as the mean error of the absolute value of the prediction error. With the notation used, it is given by the equation:

$$MAE(\hat{y}, y) = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$$

MAE is a widely used metric in the regression field. Used in the training process, it penalizes deviations from the real values as a linear function, irrespective of whether the predicted value is higher or lower than the actual. MAE produces an error which has the same units as the quantities to be predicted, making it more intuitive to associate a certain score to a certain level of predictive performance. However, there are two main issues preventing usage of MAE in the present work. The first one is the fact that the "perfect" metric should normalize the penalty term according to how close the real value is to the lower or upper bounds of normal glucose levels. If closer to the lower bound (but in the normal range), a prediction lower that the real value should be penalized more severely, because of the potential to lead to a false alarm of hypoglycemic event. If the real value is in the hypoglycemic range and the predicted value in the normal range, the penalty should be even larger, because it does not prevent the hypoglycemic event. Adding to the discussion, the larger the deviation from the actual value, the larger the possibility of the future value being in a dangerous zone, and the predicted value missing that information. A desired property of metrics to be used is to penalize larger errors more severely than smaller ones, and the linear rule does not allow for such behavior.

MSE

MAE, as its name suggests, is calculated as the mean squared difference of actual and predicted values. With the notation used, it is given by the equation:

$$MSE(\hat{y}, y) = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2$$

MSE is another commonly used metric in regression problems. Used in the training process, it penalizes deviations from the real values as a square function, irrespective of whether the predicted value is higher or lower than the actual. The units of MSE are those of the predicted value when squared, making it less intuitive than MAE in this regard.
The main advantage of MSE over MAE for the predictive problem of the current thesis is the fact that it penalizes larger errors more severely than MAE, with the purpose of eliminating them in later stages of the training process. It is still not tailored to the problem (no generalized statistical metric is, without some sort of modification) but is a solid candidate.

RMSE

RMSE is calculated as the square root of MSE:

$$RMSE(\hat{y}, y) = \sqrt{MSE(\hat{y}, y)} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}$$

It is the method of choice in the present work, because it combines advantages of both MAE and MSE. RMSE is intuitive, because its units are the same as those of the predicted quantity, and it penalizes larger errors more severely than MAE. The fact that it is not tailored to the problem will be addressed with the combined usage of a glucose specific evaluation metric.

7.1.2 Correlation metric - CC

Pearson's correlation coefficient (CC) is a measure of linear correlation between two sets of data. Its formula is the normalized covariance of two variables, divided by the product of their standard deviations, as in the equation below:

$$CC(\hat{y}, y) = \frac{\sum_{i=1}^{N}(\hat{y}_i - \bar{\hat{y}})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(\hat{y}_i - \bar{\hat{y}})^2}\sqrt{\sum_{i=1}^{N}(y_i - \bar{y})^2}}$$

As illustrated above, CC is defined in a way that defines its values in the range $[-1, 1]$. The strength of correlation is measured by its absolute value, and

the sign of CC shows positive or negative correlation for positive or negative sign respectively. CC is an important measure complementary to RMSE, and possibly holds even greater importance. Between two models that provide similar RMSE results, the one with higher CC is preferred over the other one. Consider the following example. If a model were to predict future values of a harmonic function which takes values in the range $[-1, 1]$, a predictor which always returns 0 might return similar RMSE value with a predictor returning a harmonic function. However, the value of CC will be significantly higher in the harmonic predictor, deciding in its favor over the simple constant function. A low CC value can also reveal very simple predictors that have no merit in being used in a specific problem.
Since statistical and correlation criteria (complementary to each other) have been presented, the final step of evaluation metric presentation will be one of clinical importance, the Continuous Glucose Error Grid Analysis.

7.1.3 Continuous Glucose Error Grid Analysis - CG-EGA

Continuous Glucose Error Grid Analysis (CG-EGA), also known as Clarke's Error Grid Analysis, was first introduced by Clarke [CAK08] as a clinical measure of performance for glucose predictive models. It combines Point Accuracy (P-EGA) and Rate Accuracy (R-EGA). The grid constructed for P-EGA contains predicted glucose values (mg/dl) in the y-axis and true values in the x-axis.

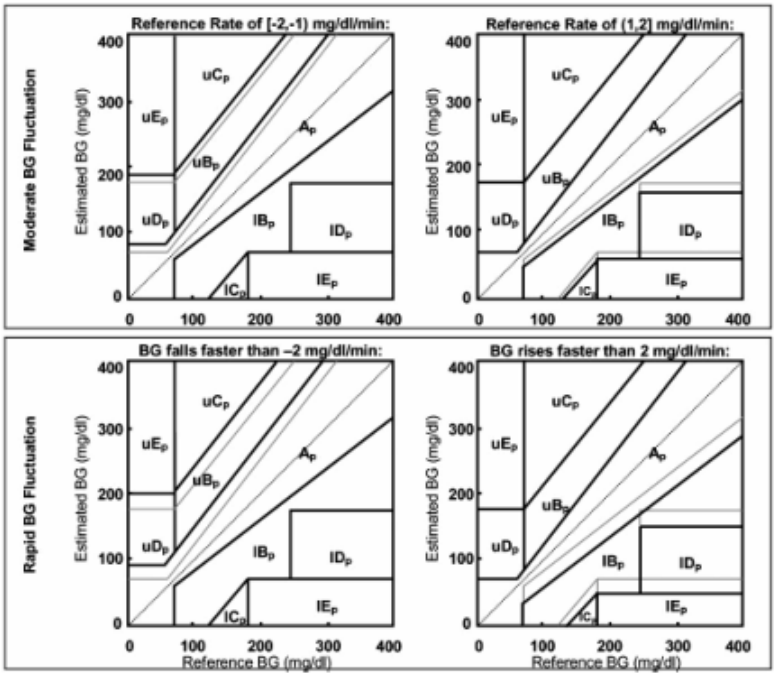

Fig. 18: Clarke's Point Error Grid Analysis (P-EGA) zones illustrated. Credit:[CAK08]

As illustrated in Figure 18, P-EGA grid is divided in five zones with respect to clinical accuracy. Zone A represents ''Clinically Accurate'' determinations, since decisions made based on prediction pairs in this zone would be accurate. Zone B errors are labeled as ''Benign'' since they would not lead to significant hypoglycemia or hyperglycemia. Zone C represents overcorrection errors. Zone D represents ''Failure to treat'' errors, and zone E contains erroneous readings, since it would lead to the exact opposite treatment of the one necessary (treating hypoglycemia as hyperglycemia and vice-versa). The width of each zone is modified in proportion to fluctuations of BG levels.

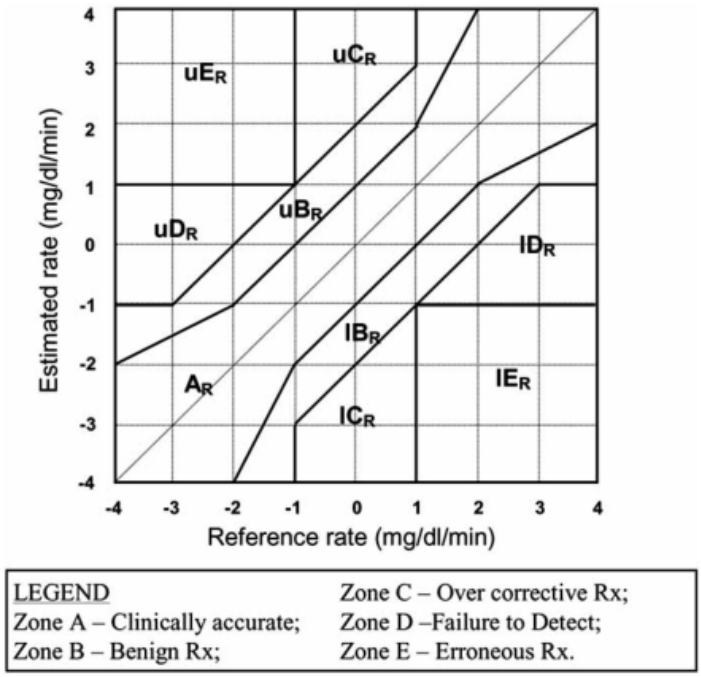

Fig. 19: Clarke's Rate Error Grid Analysis (R-EGA) zones illustrated. Credit:[CAK08]

Figure 19 illustrates the R-EGA method, which constitutes a graph of estimated glucose rate of change to the reference rate. Clinical accuracy zones hold the same meaning as in the previous analysis. However, simple point error grid analysis does not provide information about predicted and real fluctuations in BG levels (albeit change of the zone width in contrast to the simple EGA method). Since closed loop control systems require adequate warning time to prevent hypo- and hyperglycemic events, correct prediction of glucose rate of change is of vital importance to moderate insulin dosage at the required rate.

7.2 Results and discussion

The current section contains graphs and tables presenting results. After initial presentation of results for each model and prediction horizon, discussion will take place. Graphs will be presented for every real patient and a sample of in silico patients (one of each category for children, adolescents and adults). Tables to be presented contain collective results for each model, separately involving the categories of real and in silico patients respectively. A detailed discussion of results will follow their brief presentation, and figures illustrating them will be presented at the end of the chapter.

7.2.1 GRU model results

The GRU model performance is the first to be presented. Recurrent models are widely used for glucose prediction applications, and performance rivalling state of the art in the current datasets exhibits the strengths of such models.

Prediction Horizon 30 minutes

For the real patients' dataset, mean RMSE score is 14.91 mg/dl, with standard deviation of 2.78 mg/dl. Mean CC is 94.5% with standard deviation of 1.7%. Mean $\pm$ SD percentage of predictions in Zone A of CG-EGA is $92.62 \pm 3.09\%$, $4.75 \pm 1.86\%$ in Zone B, and $2.63 \pm 2.46\%$ in Zone E. Concerning in silico datasets, in the children population, performance is as follows. RMSE=$12.35 \pm 5.56 mg/dl$, CC=$95.6 \pm 1.6\%$, Zone A $97.07 \pm 2.4\%$, Zone B $1.7 \pm 1.64\%$, Zone E $1.22 \pm 1.36\%$. For the adolescents population, RMSE=$8.86 \pm 1.98 mg/dl$, CC=$96 \pm 2.7\%$, Zone A $99.26 \pm 0.97\%$, Zone B $0.43 \pm 0.72\%$, Zone E $0.3 \pm 0.74\%$. For the adult population, RMSE=$7.55 \pm 1.36 mg/dl$, CC=$95.4 \pm 1.3\%$, Zone A $99.78 \pm 0.32\%$, Zone B $0.11 \pm 0.14\%$, Zone E $0.11 \pm 0.21\%$.

Prediction horizon 60 minutes

For the real patients' dataset, mean RMSE score is 31 mg/dl, with standard deviation of 7.68 mg/dl. Mean CC is 74.4% with standard deviation of 4.6%. Mean $\pm$ SD percentage of predictions in Zone A of CG-EGA is $81.34 \pm 6.56\%$, $11.09 \pm 4.15\%$ in Zone B, and $7.57 \pm 3.84\%$ in Zone E. Concerning in silico datasets, in the children population, performance is as follows. RMSE=$15.89 \pm 4.98 mg/dl$, CC=$90.5 \pm 6.7\%$, Zone A $93.19 \pm 3.96\%$, Zone B $2.67 \pm 1.73\%$, Zone E $4.14 \pm 2.84\%$. For the adolescents population, RMSE=$12.6 \pm 1.54 mg/dl$, CC=$90 \pm 8.4\%$, Zone A $98.44 \pm 1.23\%$, Zone B $1.2 \pm 0.85\%$, Zone E $0.36 \pm 0.74\%$. For the adult population, RMSE=$12.69 \pm 2.1 mg/dl$, CC=$85.6 \pm 4.2\%$, Zone A $98.21 \pm 1.7\%$, Zone B $1.02 \pm 0.83\%$, Zone E $0.77 \pm 1.03\%$.

Prediction horizon 120 minutes

For the real patients' dataset, mean RMSE score is 49.21 mg/dl, with standard deviation of 13.86 mg/dl. Mean CC is 30.5% with standard deviation of 11.6%. Mean $\pm$ SD percentage of predictions in Zone A of CG-EGA is $75.16 \pm 8.26\%$, $14.41 \pm 4.73\%$ in Zone B, and $10.43 \pm 4.37\%$ in Zone E. Concerning in silico datasets, in the children population, performance is as follows. RMSE=$32.51 \pm 17.36 mg/dl$, CC=$64.71 \pm 10.16\%$, Zone A $83.81 \pm 7.42\%$, Zone B $5.88 \pm 3.07\%$, Zone E $10.31 \pm 5.06\%$. For the adolescents population, RMSE=$25.42 \pm 7.69 mg/dl$, CC=$66.2 \pm 11.5\%$, Zone A $93.46 \pm 4.38\%$, Zone B $5.01 \pm 3.1\%$, Zone E $1.53 \pm 1.72\%$. For the adult population,

RMSE=$19.71 \pm 4.47 mg/dl$, CC=61.5 ± 11.01%, Zone A 96.19 ± 2.46%, Zone B 2.57 ± 1.69%, Zone E 1.24 ± 1.22%.

7.2.2 MLP model results

Prediction horizon 30 minutes

For the real patients' dataset, mean RMSE score is 12.42 mg/dl, with standard deviation of 2.28 mg/dl. Mean CC is 96.4% with standard deviation of 1.3%. Mean $\pm$ SD percentage of predictions in Zone A of CG-EGA is 93.76 ± 2.49%, 4.44 ± 2.13% in Zone B, and 1.80 ± 0.80% in Zone E. Concerning in silico datasets, in the children population, performance is as follows. RMSE=$11.77 \pm 5.37 mg/dl$, CC=96.5 ± 1.3%, Zone A 97.59 ± 3.06%, Zone B 1.13 ± 1.30%, Zone E 1.28 ± 1.99%. For the adolescents population, RMSE=$8.21 \pm 2.37 mg/dl$, CC=97.1 ± 2%, Zone A 99.82 ± 0.39%, Zone B 0.17 ± 0.36%, Zone E 0.01 ± 0.04%. For the adult population, RMSE=$7.00 \pm 1.03 mg/dl$, CC=96.3 ± 1.3%, Zone A 99.96 ± 0.09%, Zone B 0.00 ± 0.00%, Zone E 0.04 ± 0.09%.

Prediction horizon 60 minutes

For the real patients' dataset, mean RMSE score is 26.91 mg/dl, with standard deviation of 7.08 mg/dl. Mean CC is 81.9% with standard deviation of 4%. Mean $\pm$ SD percentage of predictions in Zone A of CG-EGA is 82.58 ± 6.79%, 10.53 ± 4.25% in Zone B, and 6.89 ± 3.65% in Zone E. Concerning in silico datasets, in the children population, performance is as follows. RMSE=$26.88 \pm 14.87 mg/dl$, CC=77.5 ± 6.3%, Zone A 81.79 ± 12.77%, Zone B 8.86 ± 6.68%, Zone E 9.35 ± 6.85%. For the adolescents population, RMSE=$19.40 \pm 5.84 mg/dl$, CC=81.8 ± 9.4%, Zone A 94.87 ± 3.32%, Zone B 4.12 ± 2.65%, Zone E 1.01 ± 1.03%. For the adult population, RMSE=$15.35 \pm 2.99 mg/dl$, CC=78.4 ± 6.5%, Zone A 96.03 ± 2.62%, Zone B 2.68 ± 2.10%, Zone E 1.29 ± 0.99%.

Prediction horizon 120 minutes

For the real patients' dataset, mean RMSE score is 43.51 mg/dl, with standard deviation of 11.58 mg/dl. Mean CC is 41% with standard deviation of 12.1%. Mean $\pm$ SD percentage of predictions in Zone A of CG-EGA is 82.57 ± 6.99%, 9.80 ± 4.29% in Zone B, and 7.63 ± 4.14% in Zone E. Concerning in silico datasets, in the children population, performance is as follows. RMSE=$42.05 \pm 26.38 mg/dl$, CC=30.3 ± 15.7%, Zone A 78.19 ± 13.30%, Zone B 6.48 ± 4.81%, Zone E 15.33 ± 9.19%. For the adolescents population, RMSE=$32.88 \pm 12.82 mg/dl$, CC=36.8 ± 13.9%, Zone A 92.78 ± 6.60%, Zone B 4.28 ± 2.93%, Zone E 2.94 ± 4.30%. For the adult population, RMSE=$23.67 \pm 5.06 mg/dl$, CC=30.3 ± 14.8%, Zone A 97.09 ± 3.15%, Zone B 1.50 ± 1.50%, Zone E 1.42 ± 2.37%.

7.2.3 Discussion

Results for both models will be discussed for different prediction horizons, and comparison will be made afterwards.

Recurrent model - GRU

Prediction horizon 30 minutes

Regarding real patients' data, mean RMSE value was deemed satisfactory at 14.91 mg/dl, while mean CC of 94.5% shows a very strong correlation between predicted and real values. Total percentage of Zone A and Zone B predictions in the CG-EGA analysis is higher than 97%, contributing to the conclusion that the recurrent model exhibits strong predictive capabilities with real world usage in avoiding hypoglycemic and hyperglycemic events for the majority of patients. Mean Zone E percentage was 2.63%, however there is an outlier value of 9.34%. With 75% of datasets prediction values achieving a result of 2.29% in Zone E, closer inspection is required. The results for patient 1 provide 3.09% in Zone E, patient 10-5.3% and patient 7-9.34%. While results for patient 1 do not show significant deviation from the norm, results for patient 7 demonstrate the fact that the model could miss a significant number of hypoglycemic events for some patients. Pinpointing the weak spot of the current model, future work can be pointed towards the direction of improving predictive performance of hypoglycemic events for prediction horizon of 30 minutes.
Regarding in silico children data, evaluation metrics provide better mean results, as well as significantly lower maximum percentage in the danger zone of CG-EGA analysis. Lower RMSE and higher CC are obtained, as well as less than half the percent of events in Zone E (as a mean value).
Adolescents in silico dataset results exhibit even better performance that that of the children dataset. Significantly lower RMSE of 8.86 mg/dl, slightly higher CC of 96%, and significantly improved results for clinical evaluation criteria. Mean Zone E percentage is 0.3%, maxing at 2.41%. A trend showing better in silico results for adolescents vs children data can be initiated via these observations, and is to be tested following further discussion of results.
Arguments in favor of this trend are enhanced after analysis of evaluation metric results for the adult in silico population. Mean RMSE of 7.55mg/dl is the lowest among populations, with CC of 95.4%. Zone E events are by far the lowest among age groups, with mean value of 0.11% and 50% of patients' predicted values showing zero events in the danger zone.
An argument can be made about expected better performance of the model in data provided from simulations, given less latent variables and correlations. This argument is also to be tested in further discussion of results.

Prediction horizon 60 minutes

Predictive performance is expected to decline with prediction horizon further into the future. This fact is confirmed by results of prediction horizon of 1 hour. Concerning real patients' data, mean RMSE is increased to 31 mg/dl, mean CC reduced to 74.37%, which still provides strong correlation but significantly weaker than the previous prediction horizon. Mean Zone A plus Zone B predictions were higher than 92%, with 7.57% in Zone E. Three out of 12 patients' predictive results had greater than 9% Zone E, maxing out at approximately 15 percent. This enhances the previous observation about the recurrent model's weak spot in avoiding hypoglycemic events, and necessity of improvement in this regard.
In silico predictive performance was consistently higher than that of real data. For the children population, mean RMSE was 15.89 mg/dl, mean CC 90.5% and Zone E predictions percentage was 4.13%, maxing out at 10.62%. Clinical evaluation criteria for this population were similar to real patients' evaluation criteria for prediction horizon of 30 minutes, with statistical criteria exhibiting slightly lesser performance.
Results for the adolescents population were better in terms of RMSE and predictions in the danger zone (0.36% Zone E, maxing out at 2.43%) with similar CC of 90%. Adult population showed a performance drop of about 5% with respect to correlation compared to children and adolescents, with similar RMSE to that of adolescents and slightly higher rate of predictions in the danger zone.

Prediction horizon 120 minutes

Prediction horizon of two hours is rarely used in the literature. While prediction and avoidance of extreme events would prove to be of significant importance in real world applications, current results do not allow for practical application, and models developed in the context of the present thesis do not escape from that norm. High mean RMSE of 49.21 mg/dl and low mean CC of 30% for real patients' data reveal weak correlation between real and predicted values of glucose. While combined Zone A + Zone B results are comparable to that of the previous prediction horizon, weak correlation prevents the model to be used in real world applications with the current levels of performance. This conclusion is illustrated in figures of predicted vs real values for prediction horizon of 120 minutes, exhibiting significantly different behavior of the two graphs.
Predictive performance in the in silico populations was significantly improved in comparison to the real one, with mean RMSE of 32.5 mg/dl and mean CC of 64.7% in the children dataset, 25.42 mg/dl and 66.2% in the adolescents dataset, and 19.71 mg/dl and 61.5% in the adults dataset. Percentage in the danger zone was relatively high for the children dataset, much lower than that of adolescents and adults datasets. However, low CC still prevents the model from deing implemented

for applications for prediction horizon of two hours.

MLP model

Prediction horizon 30 minutes

Regarding real patients' data, mean RMSE value was lower than that of the recurrent model at 12.42 mg/dl, while mean CC of 96.4% shows a very strong correlation between predicted and real values. Total percentage of Zone A and Zone B predictions in the CG-EGA analysis is higher than 98%, contributing to the conclusion that the MLP model also exhibits strong predictive capabilities with real world usage in avoiding hypoglycemic and hyperglycemic events for the majority of patients. Mean Zone E percentage was 1.8%, with max value of 3.09%. For prediction horizon of 30 minutes, both statistical and clinical evaluation criteria show better performance of the MLP model.
Regarding in silico children data, evaluation metrics provide better mean results, as well as lower maximum percentage in the danger zone of CG-EGA analysis. Lower RMSE and similar CC are obtained, as well as less than events in Zone E (as a mean value). However, the maximum value of events at Zone E was higher in the in silico children population, possibly pointing to lesser performance of the MLP model for in silico populations.
Adolescents in silico dataset results exhibit better performance that that of the children dataset, surpassing that of the recurrent model. Significantly lower RMSE of 8.21 mg/dl, slightly higher CC of 97%, and significantly improved results for clinical evaluation criteria. Mean Zone E percentage is 0.013%, maxing at 0.13%. These results enhance the trend initiated by GRU results, regarding increased performance in higher age groups.
As for the adult in silico population, mean RMSE of 7mg/dl is the lowest among populations, with CC of 96.3%. Zone E events are not the lowest among age groups (adolescents results were better), with mean value of 0.04% and maximum value of 0.25%.
Better model performance for in silico patients is also observed for this model version.

Prediction horizon 60 minutes

Performance decline with increasing prediction horizon is also observed for the MLP model, however results are consistently better than those of the recurrent one for real patients' data. Mean RMSE of 26.91 mg/dl with mean CC of 81.88% show stronger correlation between predicted and real values than those of the previous model. Danger zone predictions are also on the lower side, with mean of 6.89%, maxing out at 13.46%.

For in silico data, the situation is reversed for prediction horizon greater than 30 minutes. Error, correlation and clinical evaluation criteria all reveal lesser performance of the MLP model in all of the in silico age groups. The trend that showed performance in the adolescents dataset better than that of the other populations for prediction horizon of 1 hour holds true for the MLP model as well, albeit with significantly lesser performance than that of the recurrent one.

Prediction horizon 120 minutes

Results for prediction horizon of two hours for the present model lead to similar conclusions to those of the former prediction horizon. Better performance in real patients' data regarding all evaluation criteria, lesser performance for each in silico age group. The latter is more pronounced in the current prediction horizon.

Model comparison and concluding remarks

Comparison of the models needs to take into account the fact that the MLP model performed its best when only one feature (glucose values) was utilized, while the recurrent model performed better with a certain set of features, including outputs from CMs. The above fact is an advantage of the GRU model, since it bodes well with being used in hybrid models. Performance in real datasets was slightly higher for MLP, but the situation reversed for in silico datasets. Moreover, performance for in silico datasets was consistently higher for both models than that of real patients' data, which can be attributed to fewer latent variables in data produced by simulation, since certain equations are used. That fact is an indication for the necessity of development of simulation software that imitates the behavior of a real patient's insulin response more closely than that of existing models. Concluding the discussion, recurrent models show promise and better integration in hybrid model scenarios than the current state of the art model, albeit slightly lower performance currently.

Illustration of results

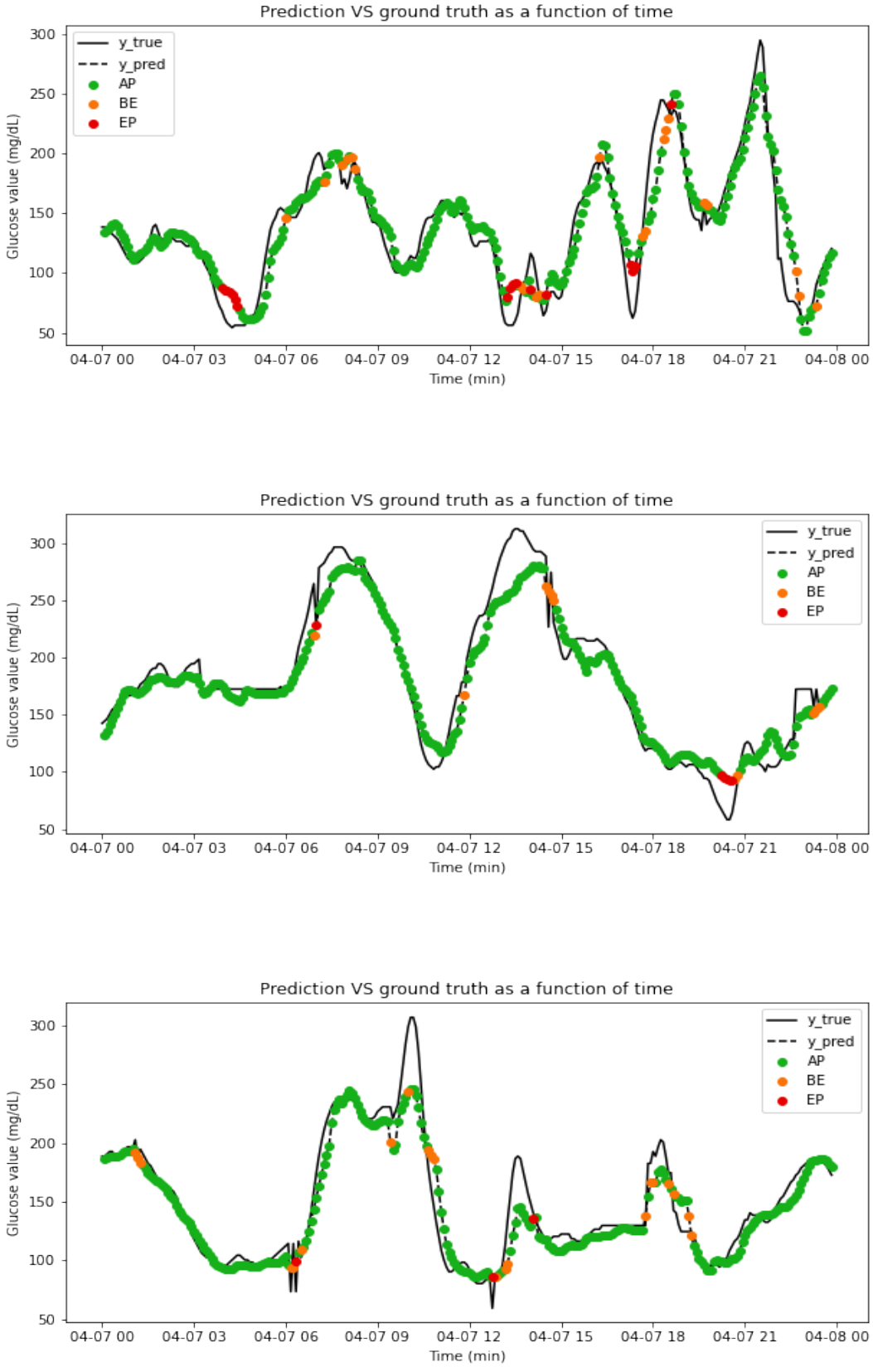

Fig. 20: GRU model results for patients 1 to 3. Prediction Horizon: 30 minutes

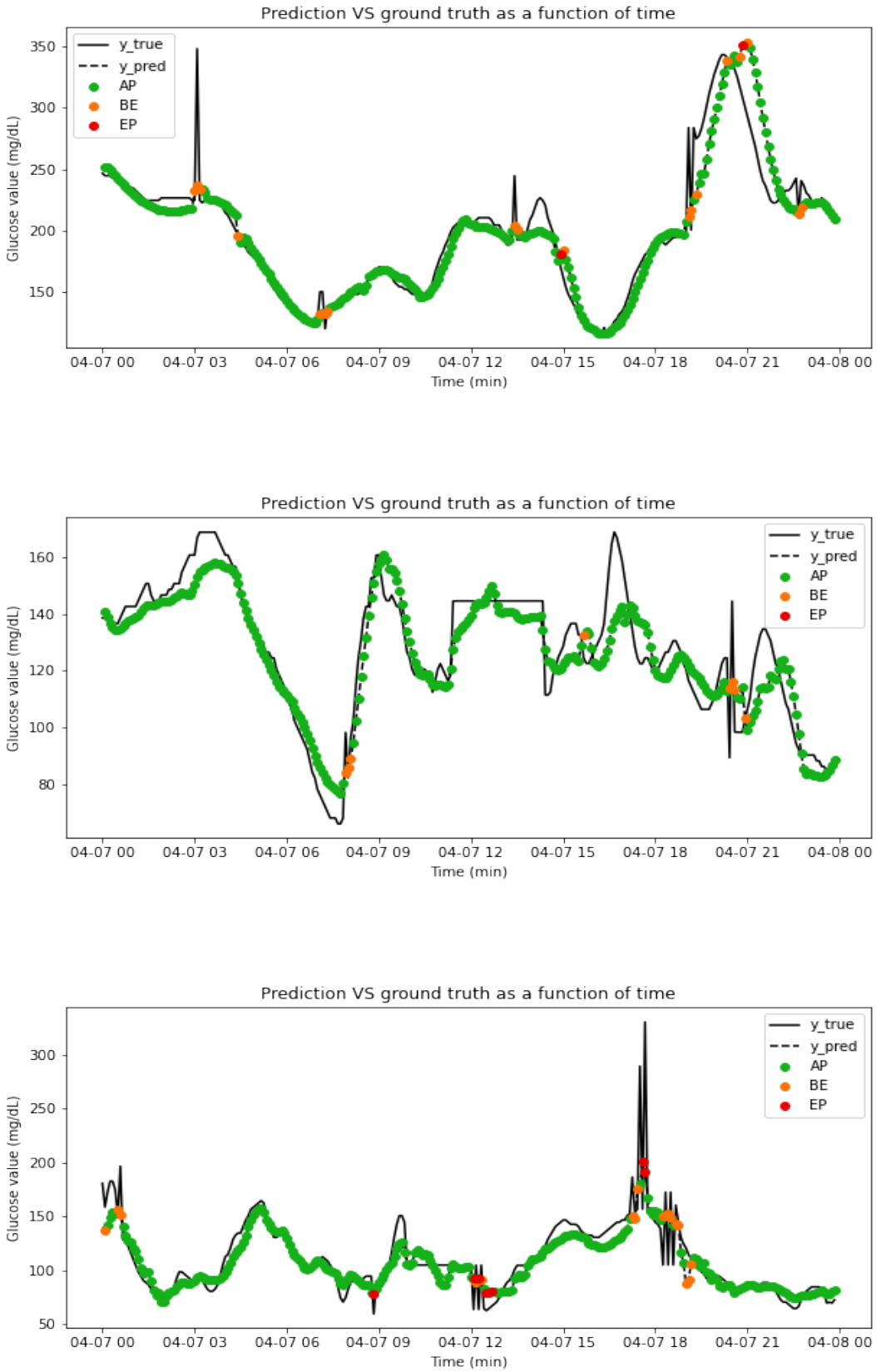

Fig. 21: GRU model results for patients 4 to 6. Prediction Horizon: 30 minutes

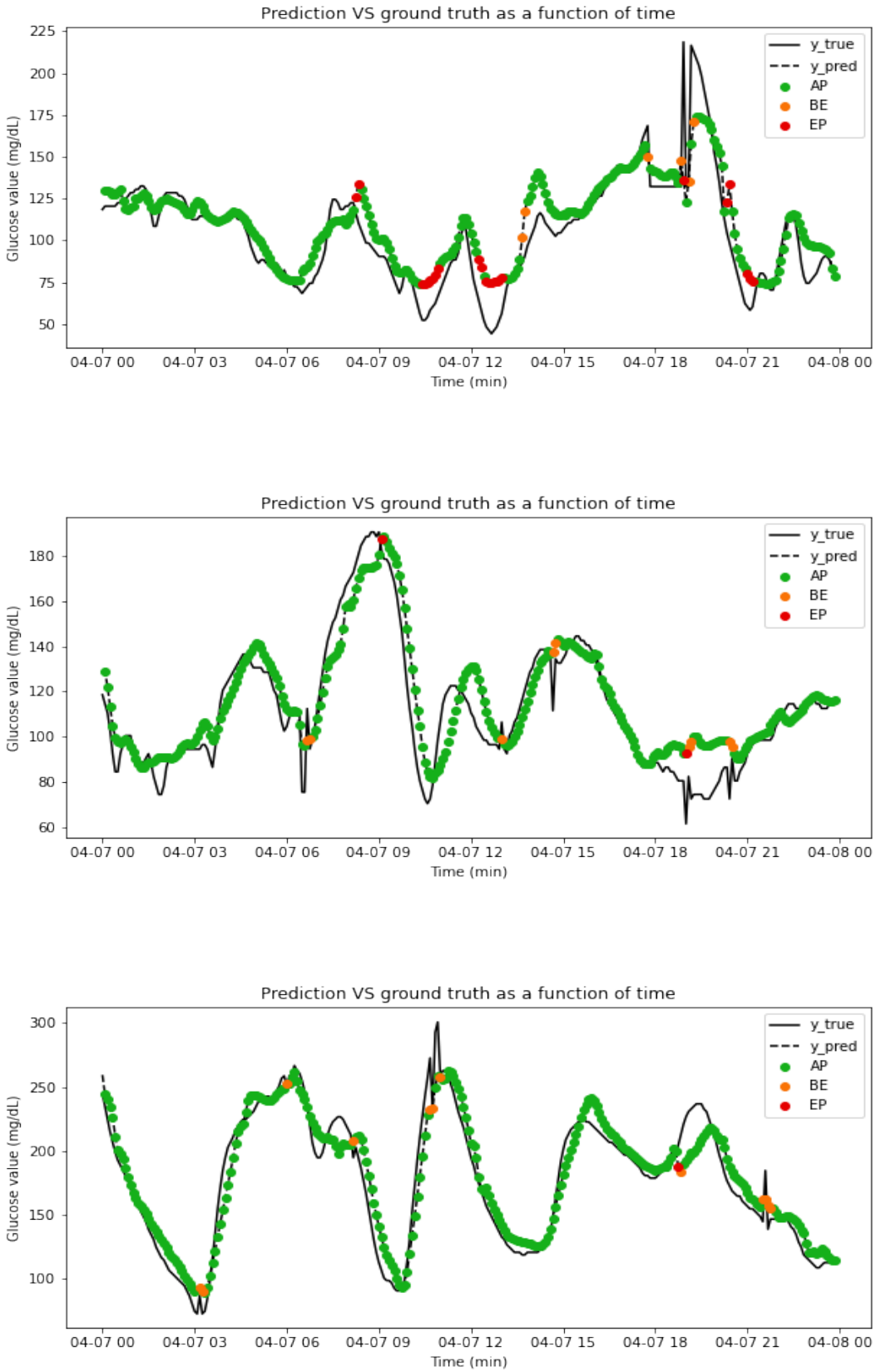

Fig. 22: GRU model results for patients 7 to 9. Prediction Horizon: 30 minutes

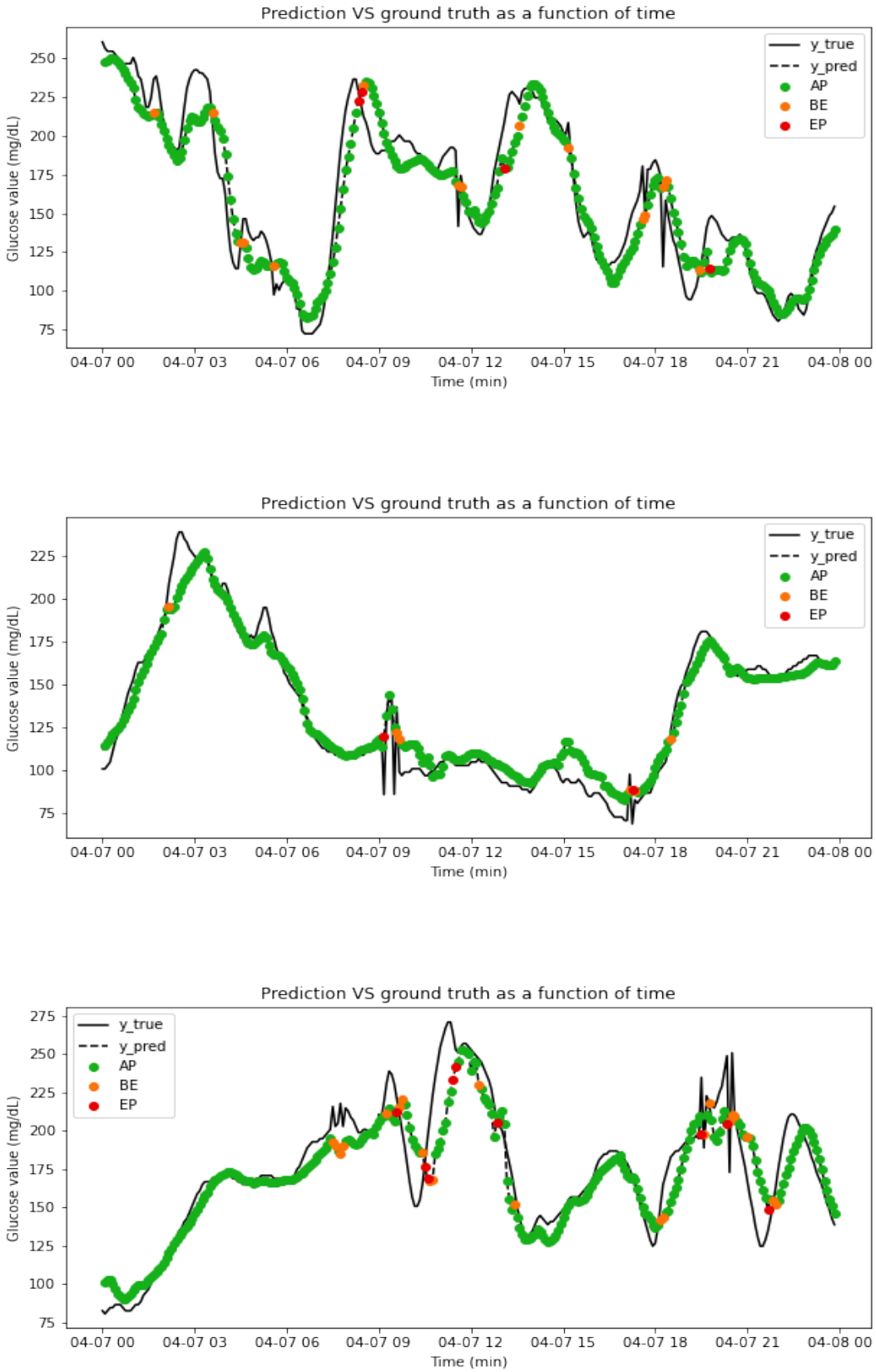

Fig. 23: GRU model results for patients 10 to 12. Prediction Horizon: 30 minutes

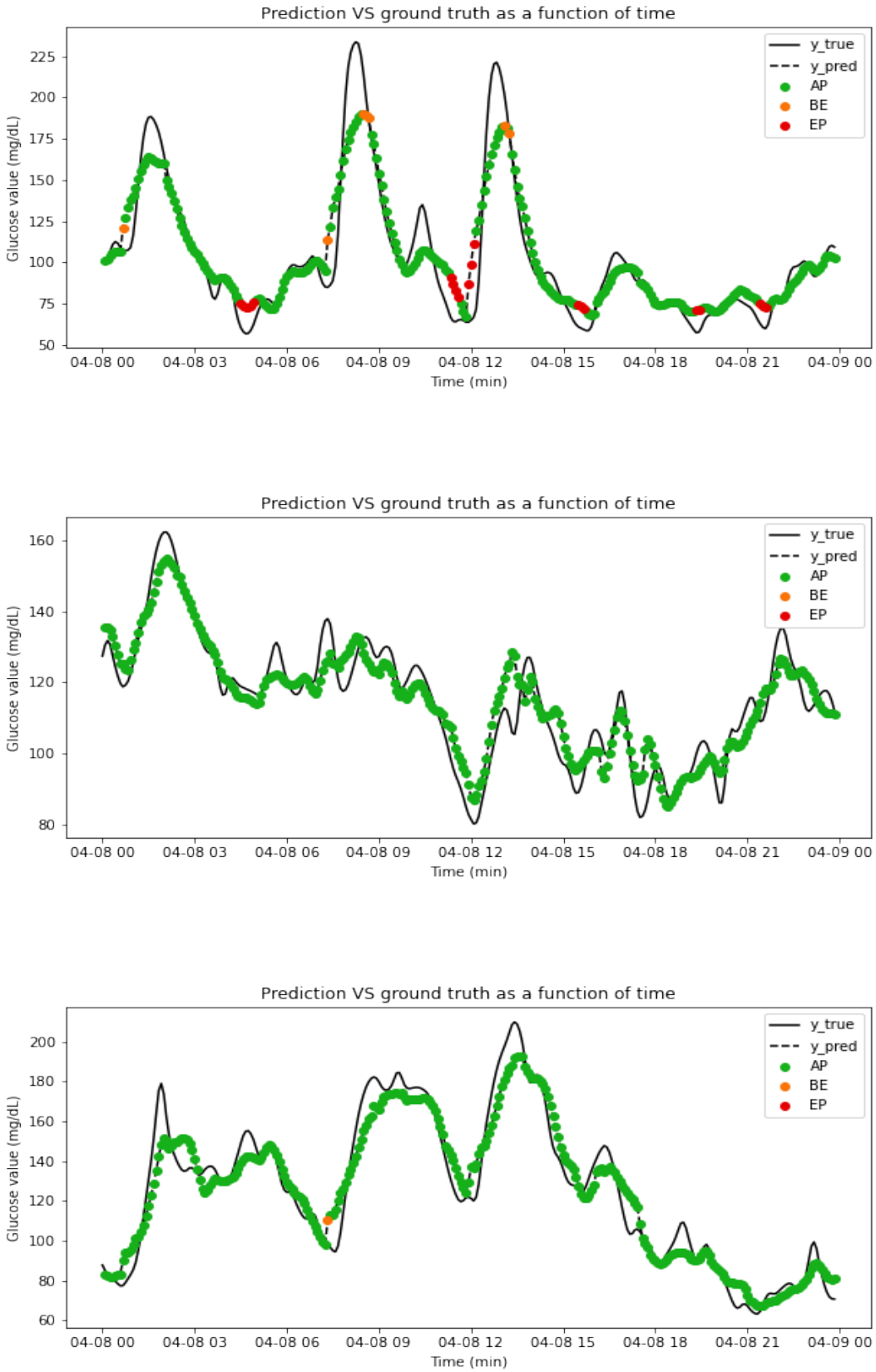

Fig. 24: GRU model results for one patient from each in silico dataset age group. Prediction Horizon: 30 minutes

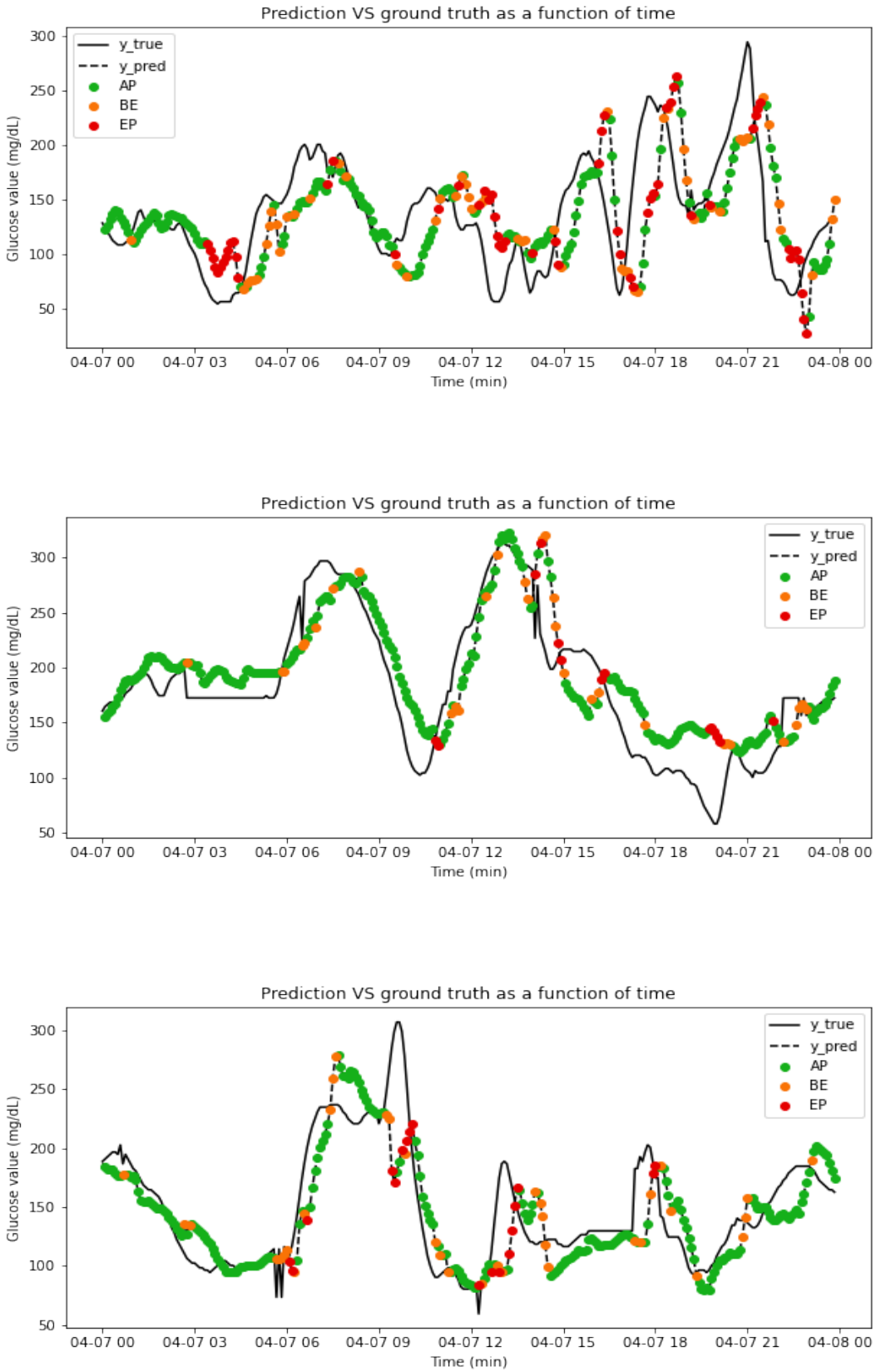

Fig. 25: GRU model results for patients 1 to 3. Prediction Horizon: 60 minutes

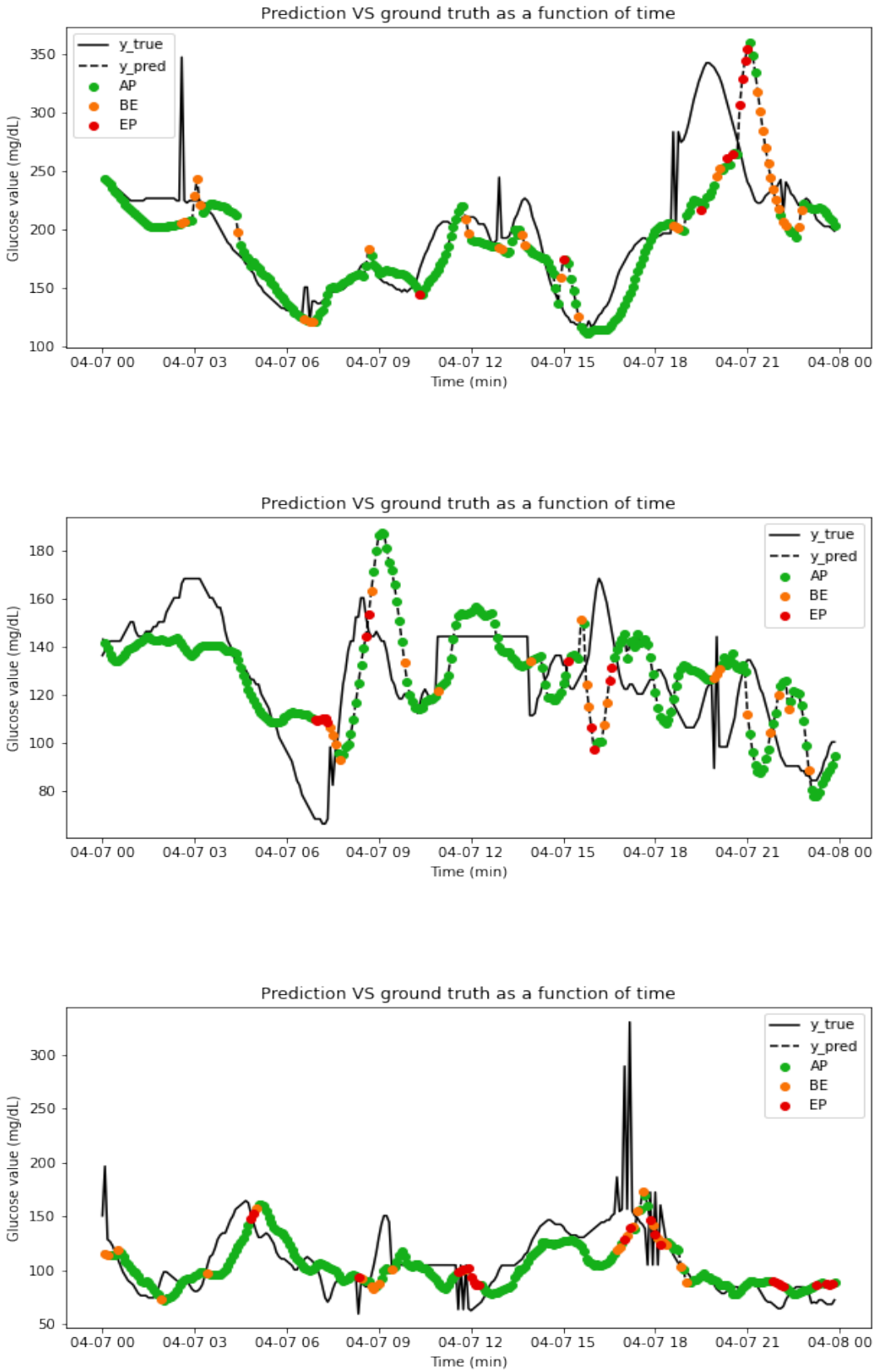

Fig. 26: GRU model results for patients 4 to 6. Prediction Horizon: 60 minutes

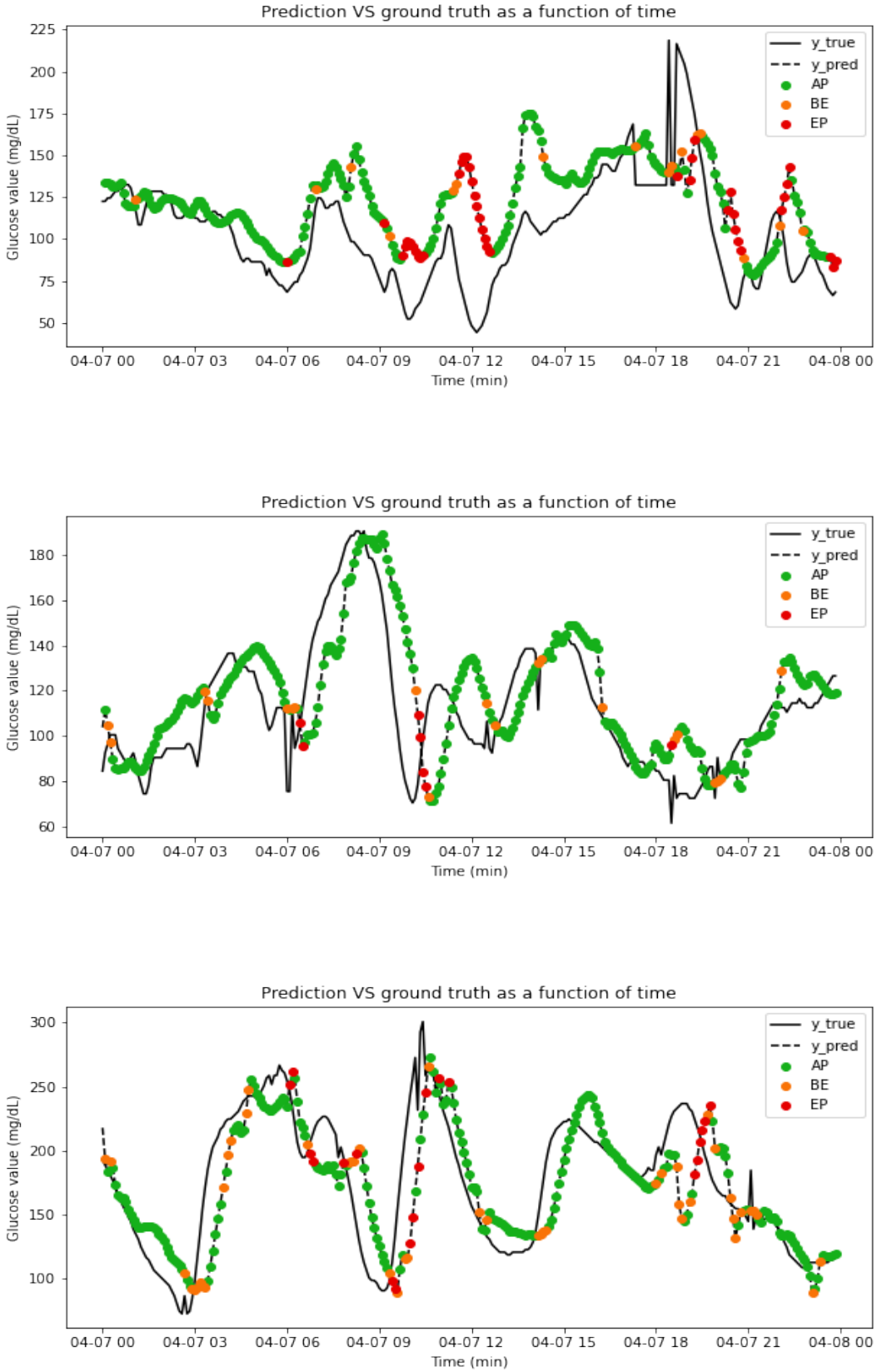

Fig. 27: GRU model results for patients 7 to 9. Prediction Horizon: 60 minutes

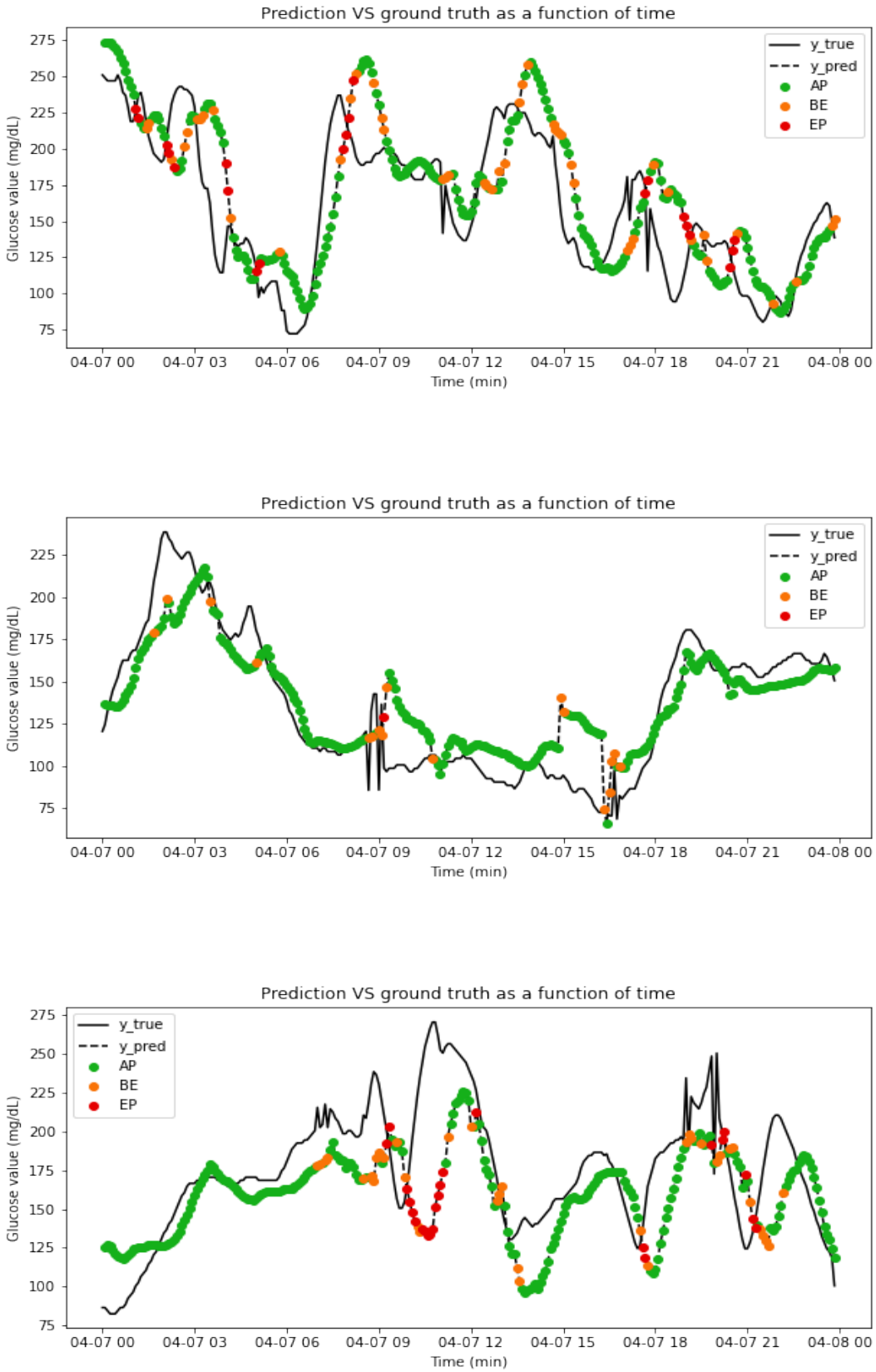

Fig. 28: GRU model results for patients 10 to 12. Prediction Horizon: 60 minutes

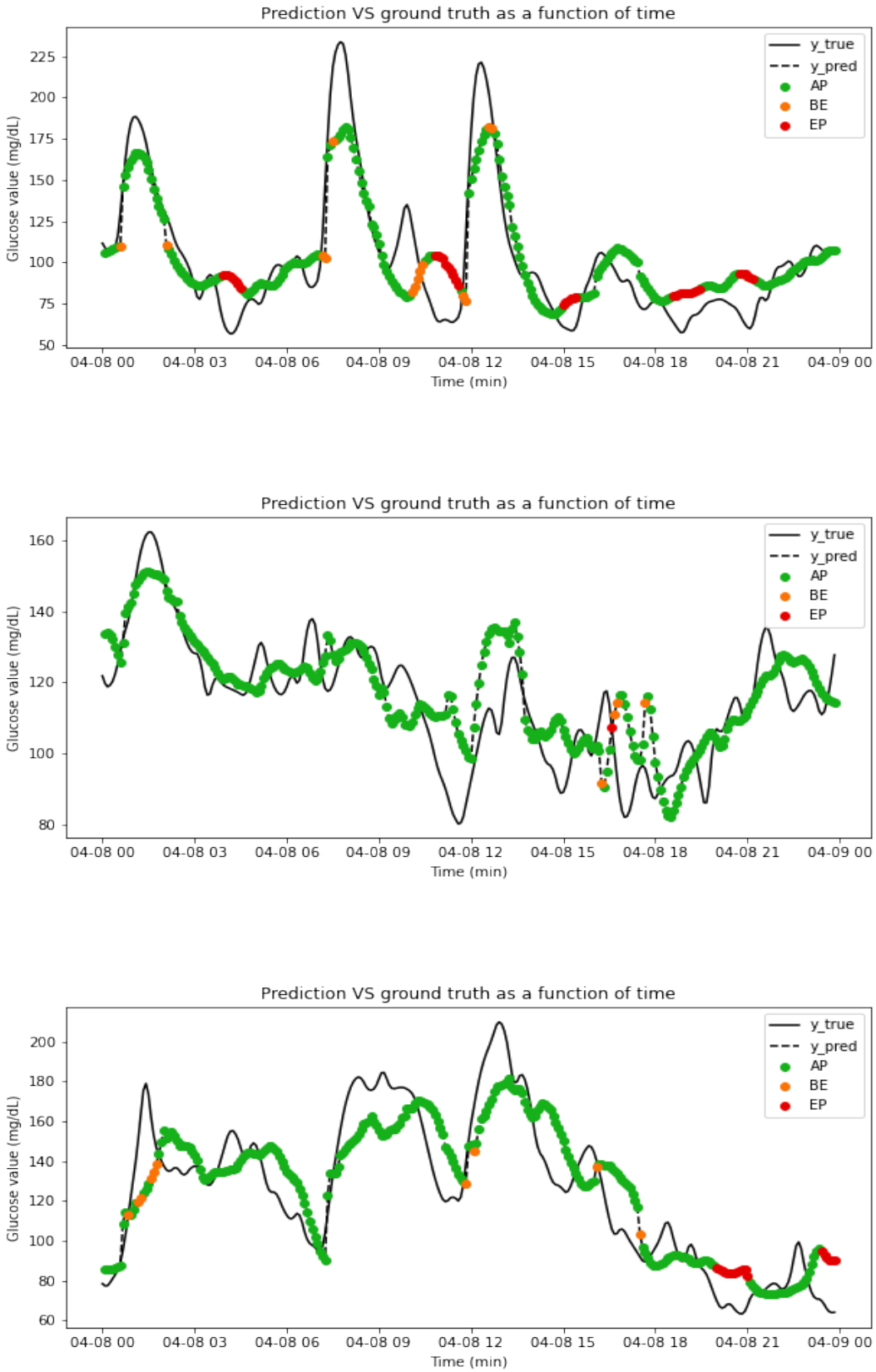

Fig. 29: GRU model results for one patient from each in silico dataset age group. Prediction Horizon: 60 minutes

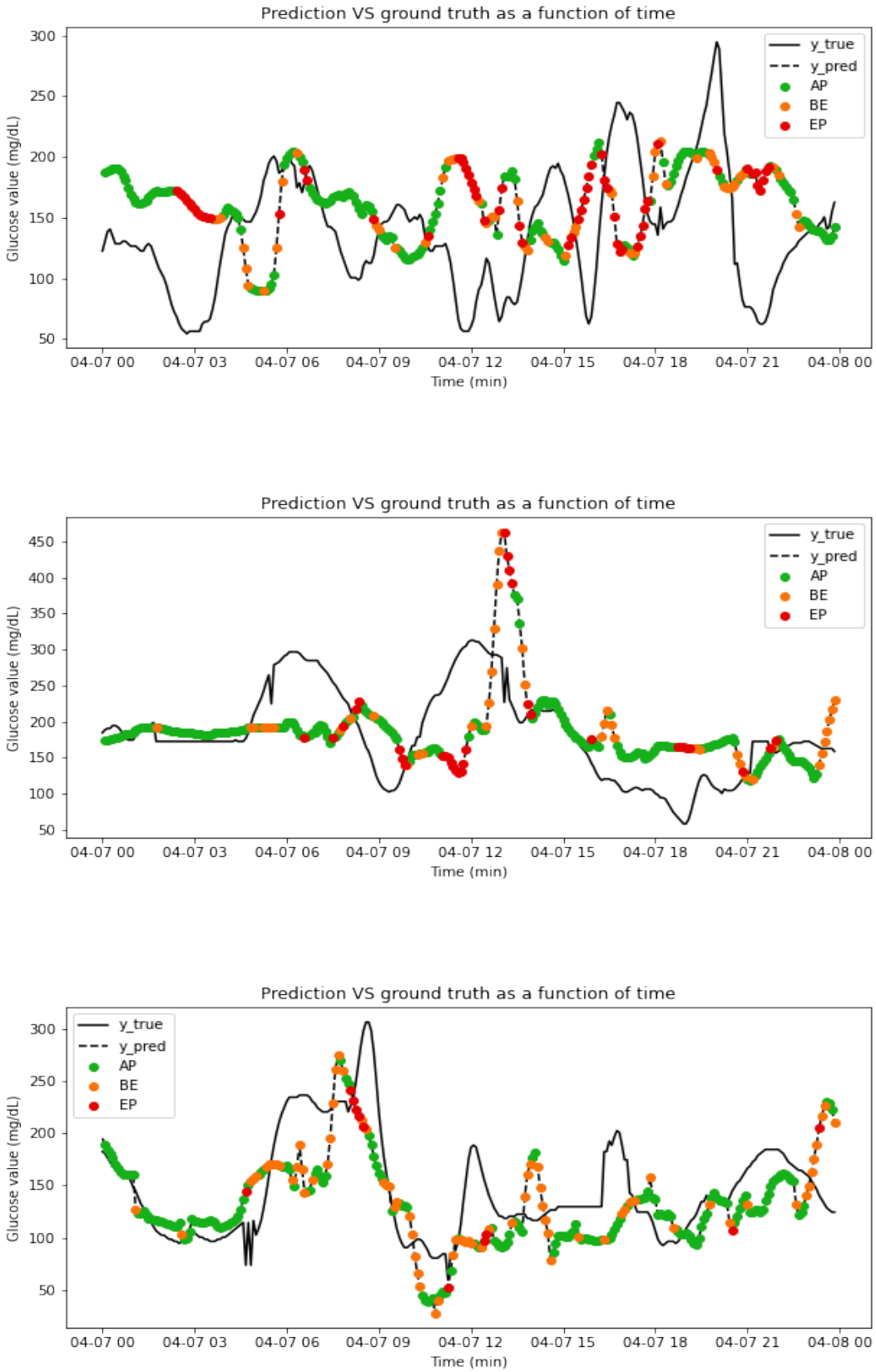

Fig. 30: GRU model results for patients 1 to 3. Prediction Horizon: 120 minutes

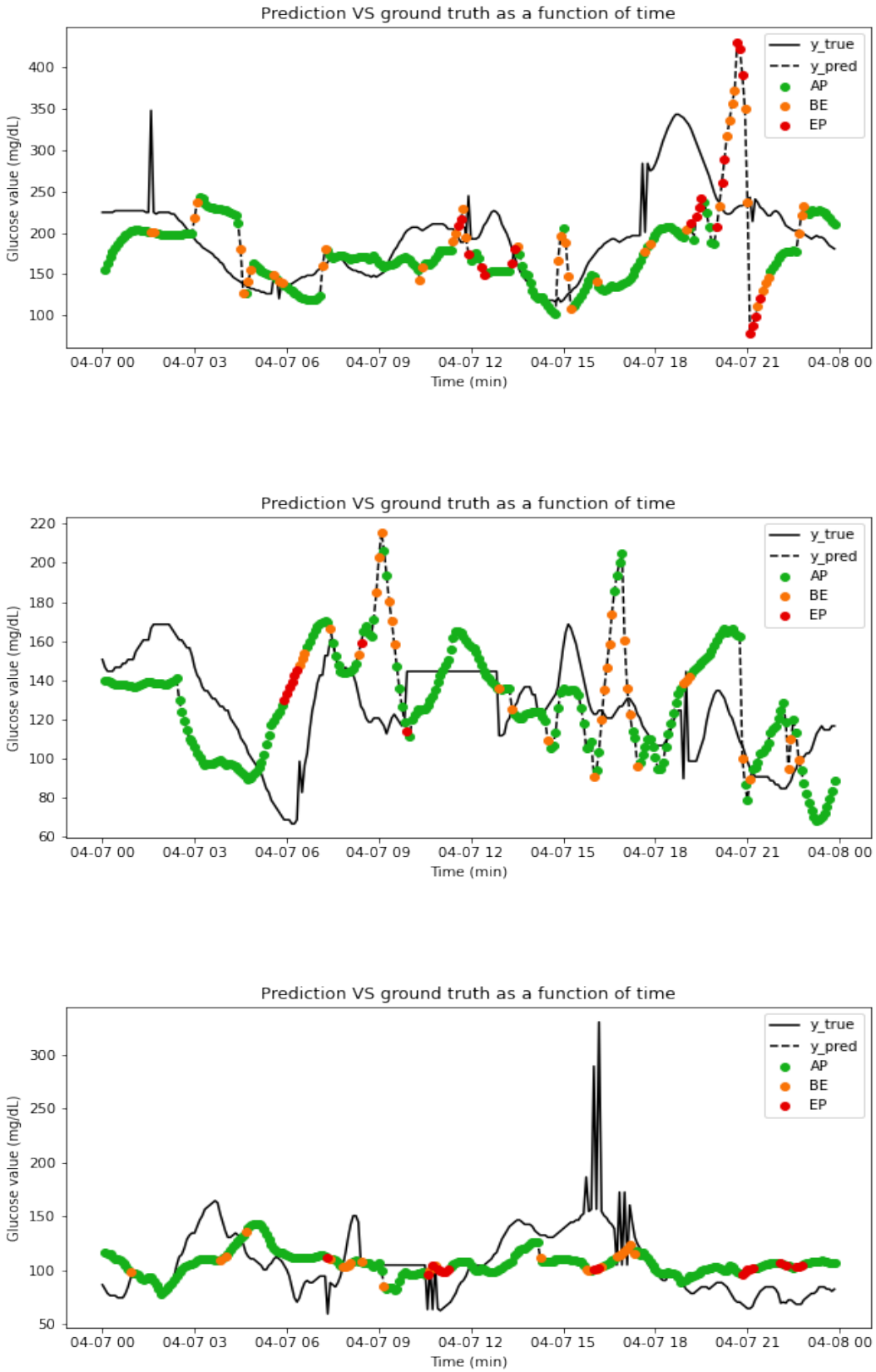

Fig. 31: GRU model results for patients 4 to 6. Prediction Horizon: 120 minutes

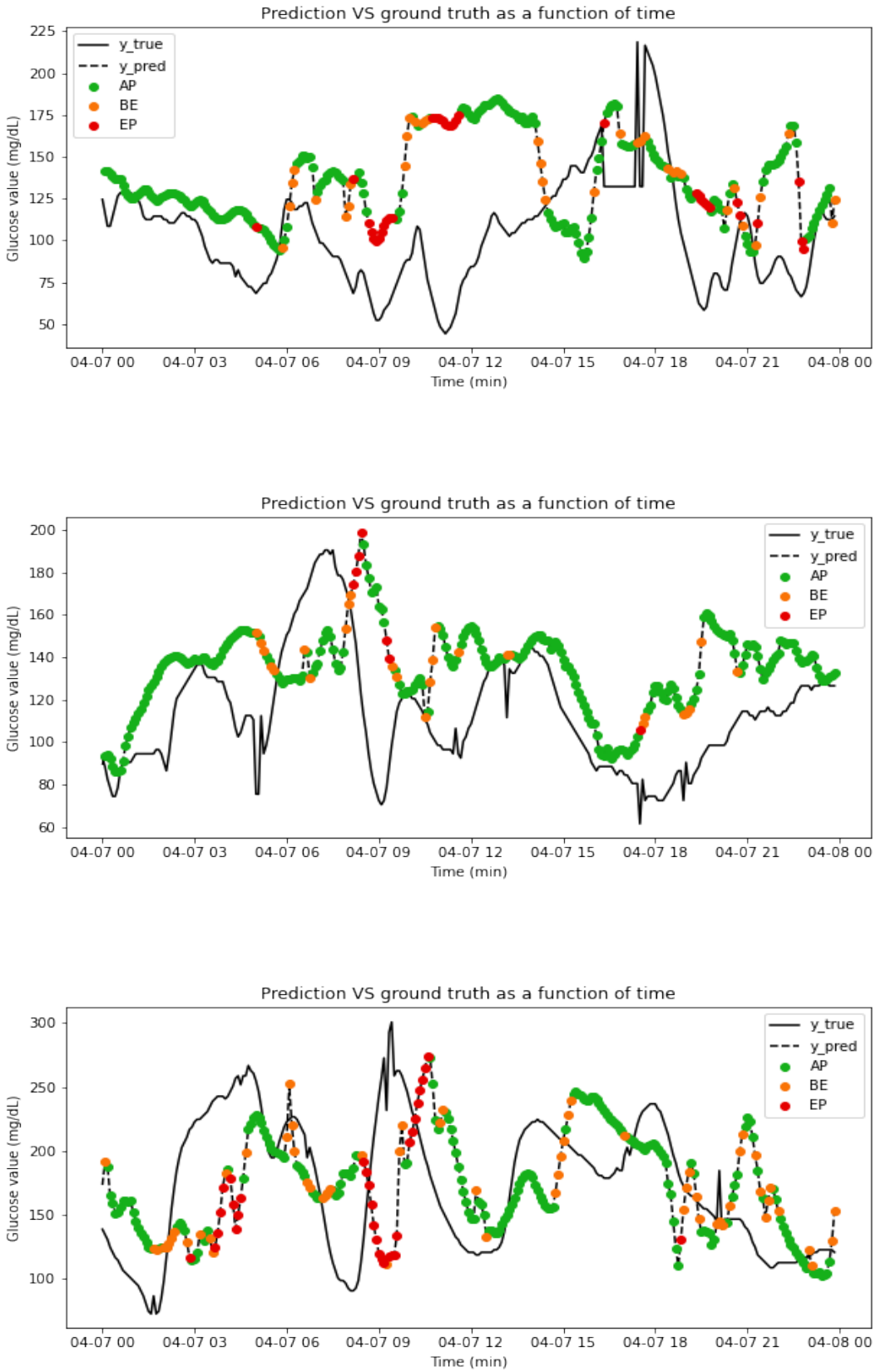

Fig. 32: GRU model results for patients 7 to 9. Prediction Horizon: 120 minutes

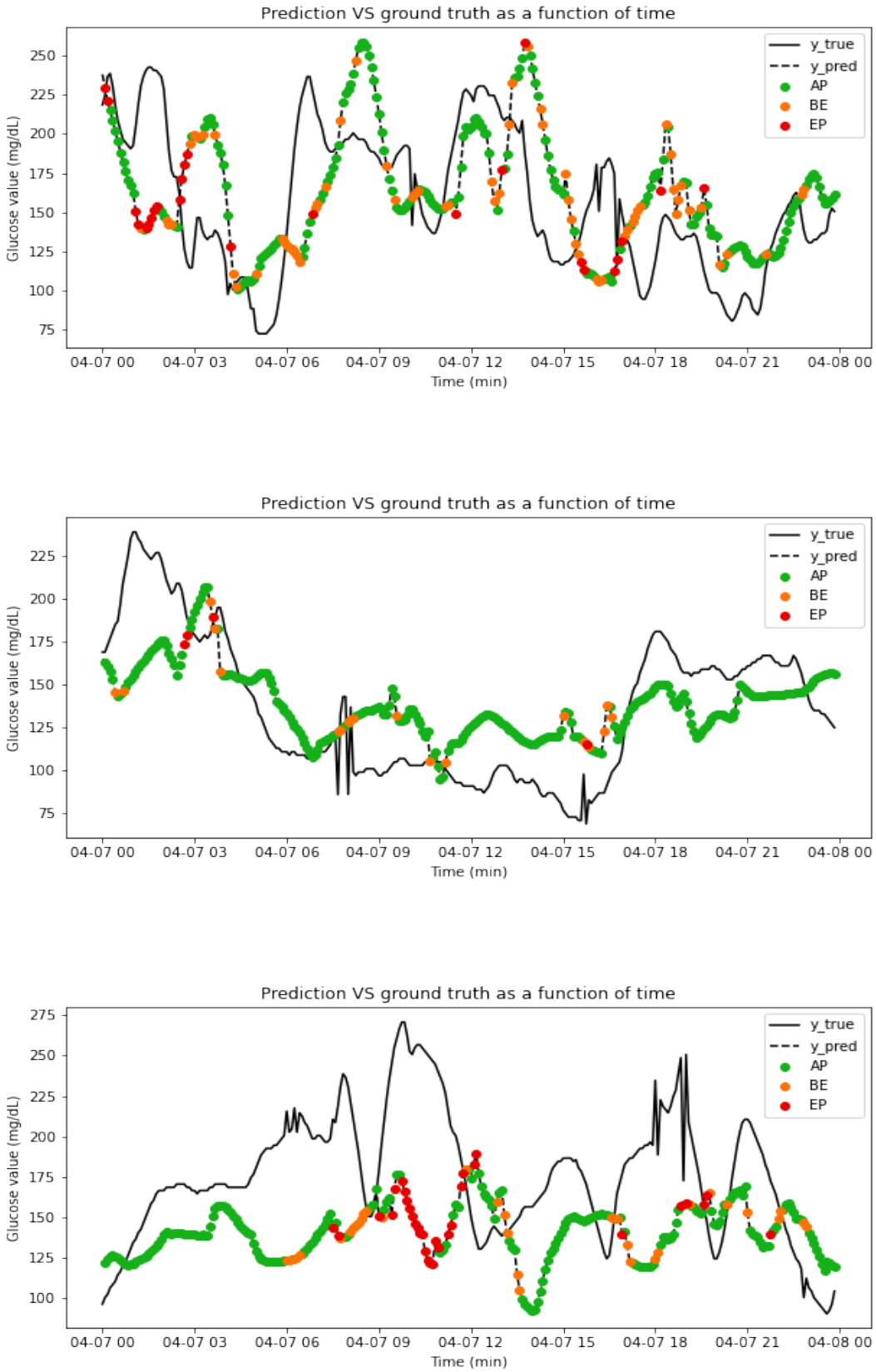

Fig. 33: GRU model results for patients 10 to 12. Prediction Horizon: 120 minutes

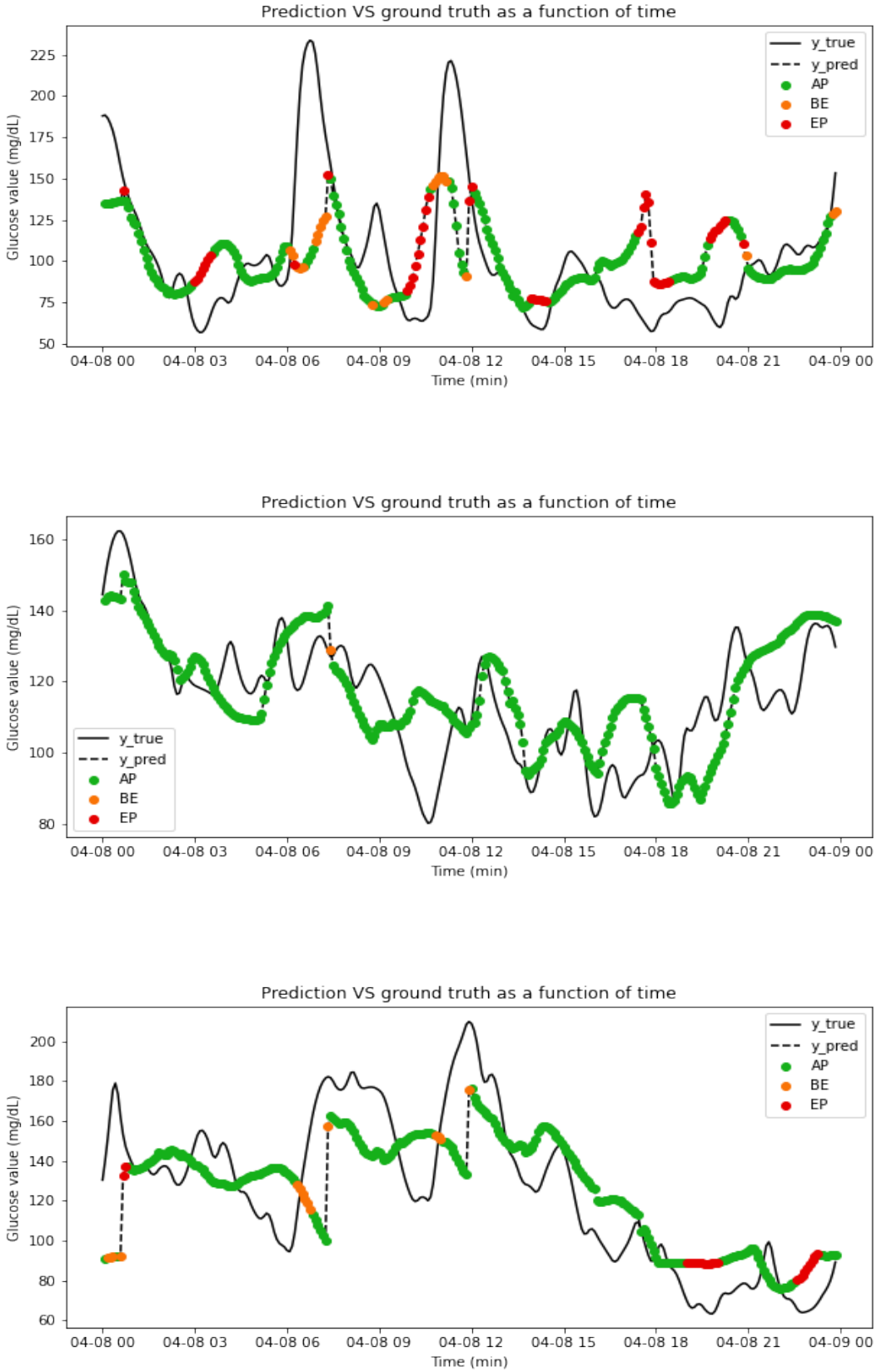

Fig. 34: GRU model results for one patient from each in silico dataset age group. Prediction Horizon: 120 minutes

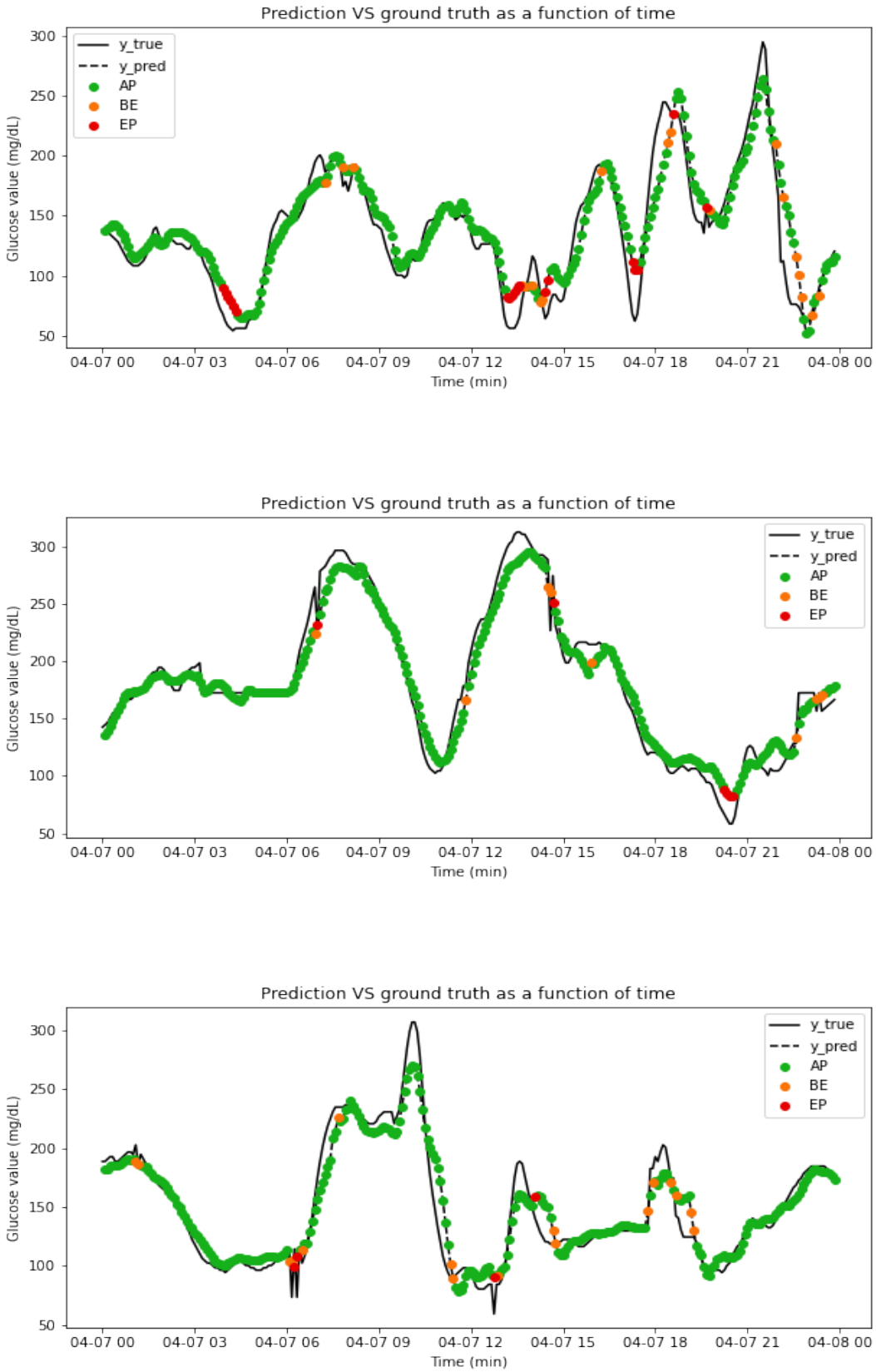

Fig. 35: MLP model results for patients 1 to 3. Prediction Horizon: 30 minutes

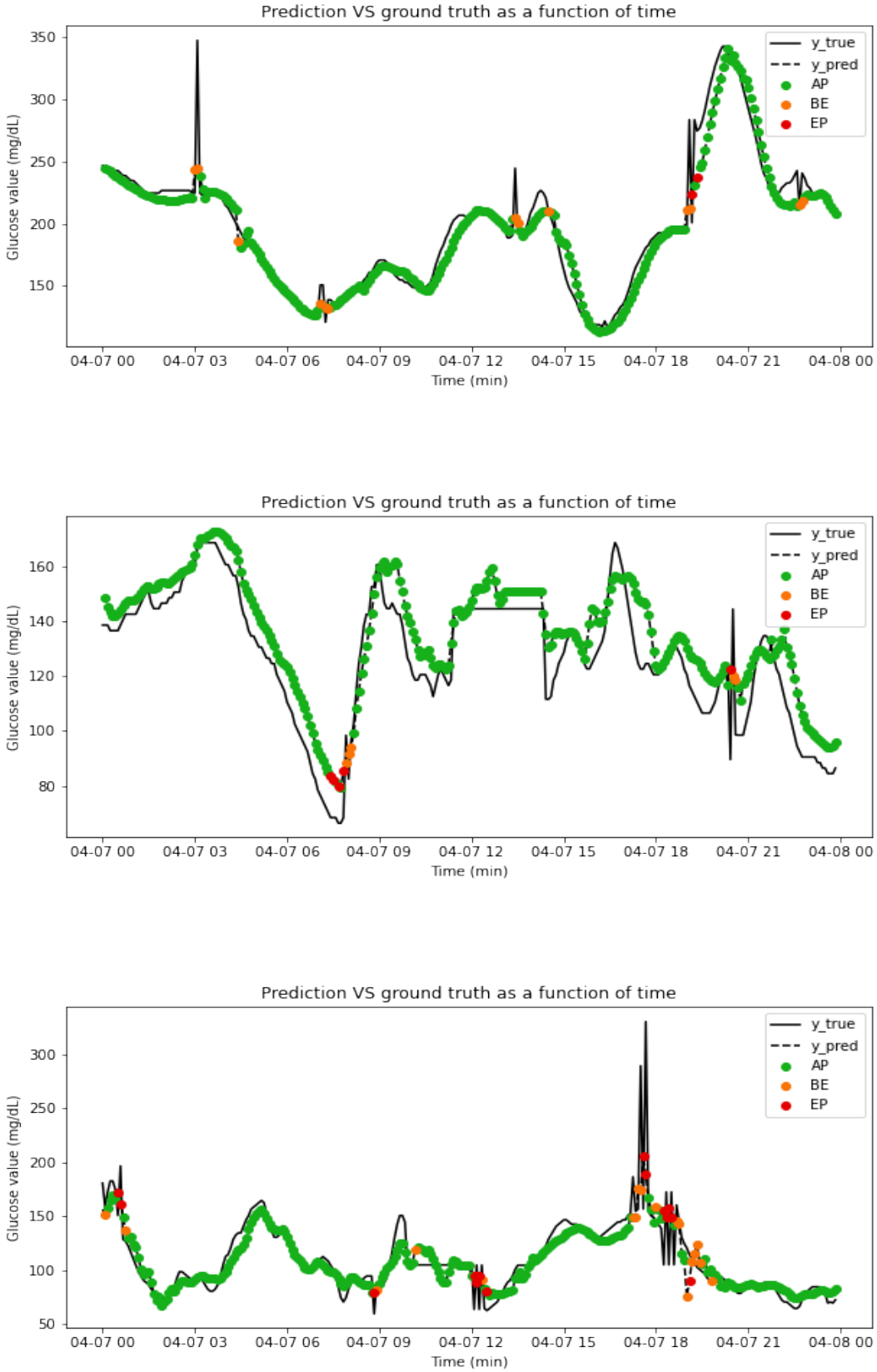

Fig. 36: MLP model results for patients 4 to 6. Prediction Horizon: 30 minutes

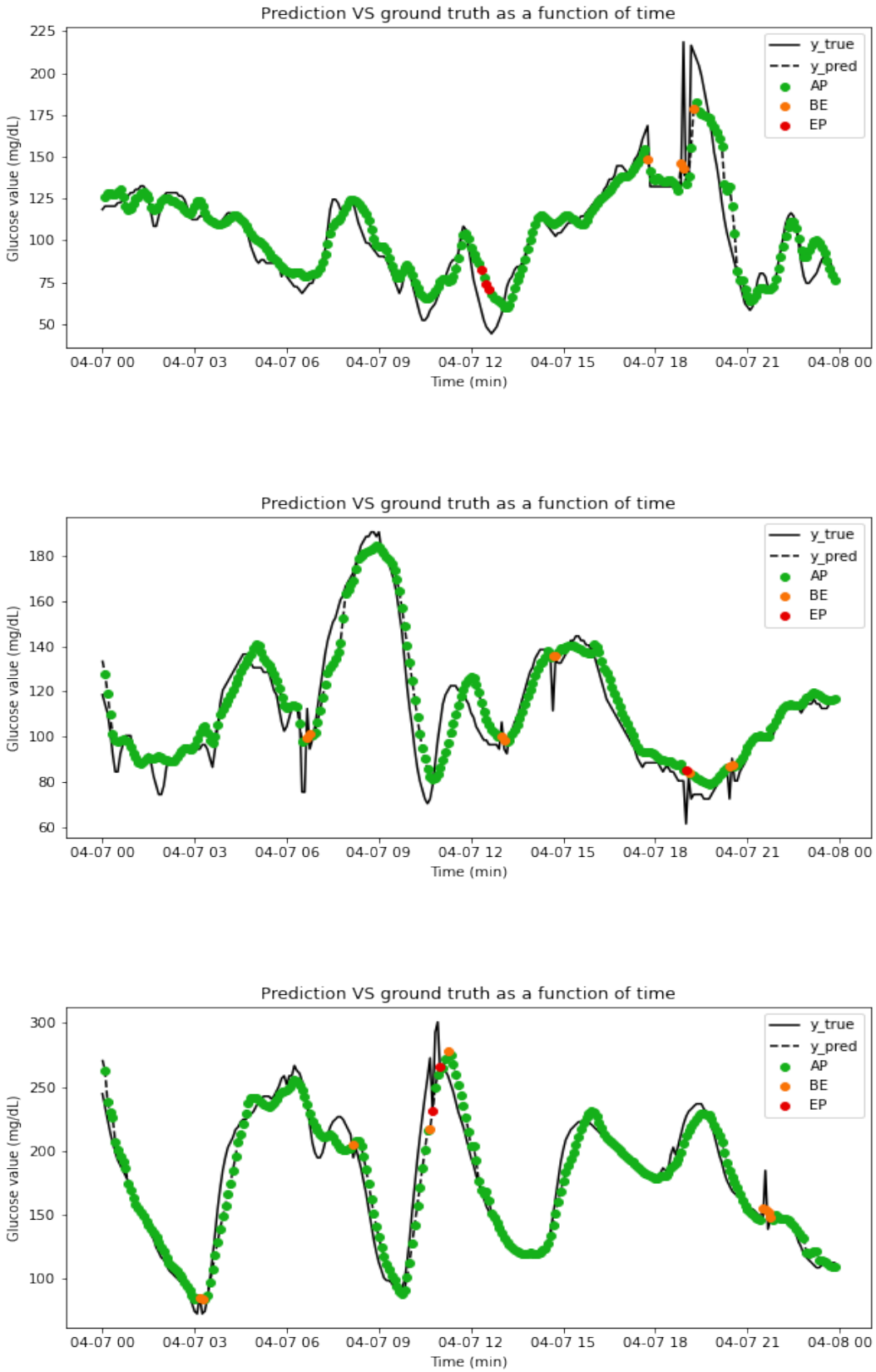

Fig. 37: MLP model results for patients 7 to 9. Prediction Horizon: 30 minutes

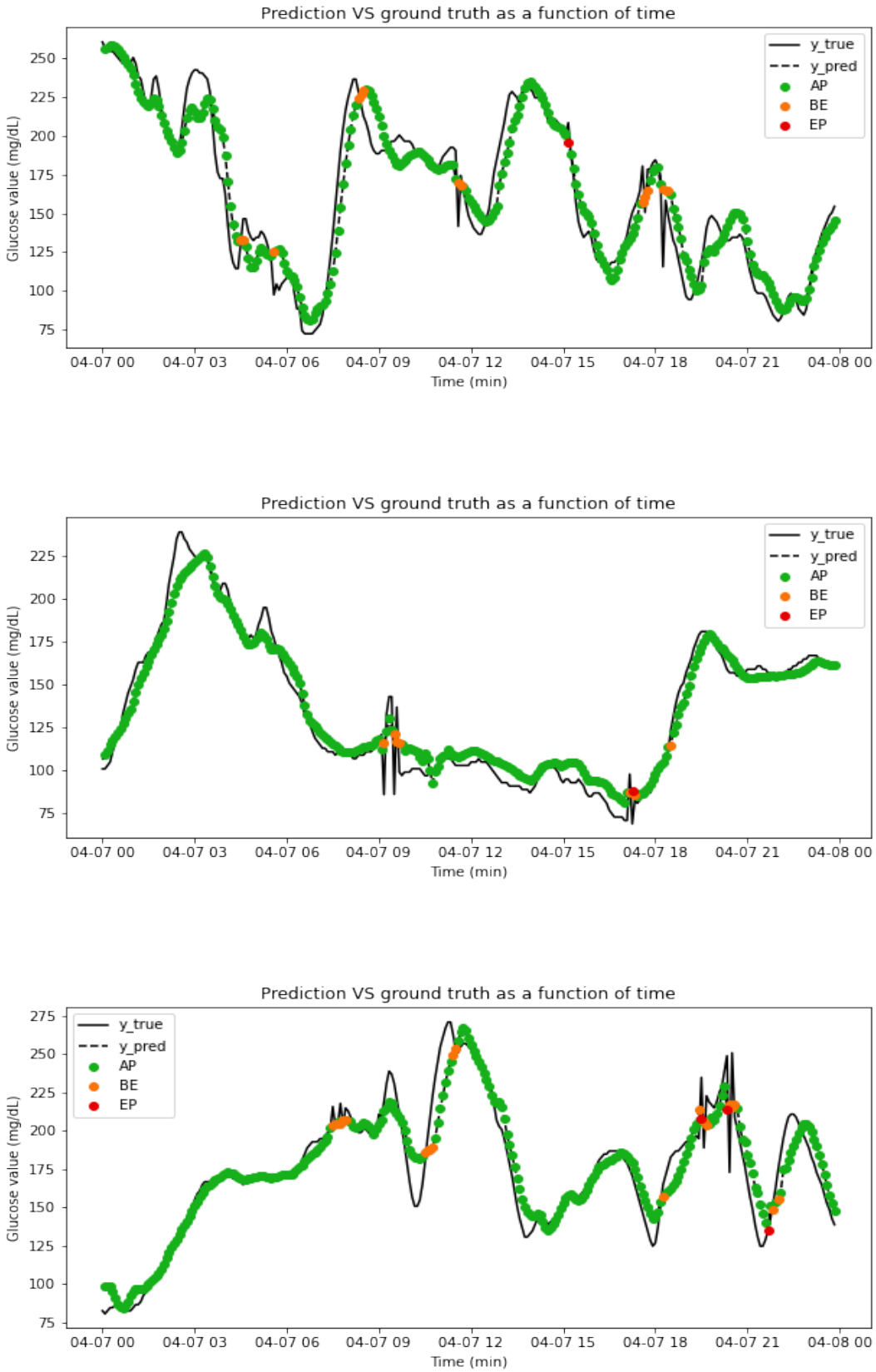

Fig. 38: MLP model results for patients 10 to 12. Prediction Horizon: 30 minutes

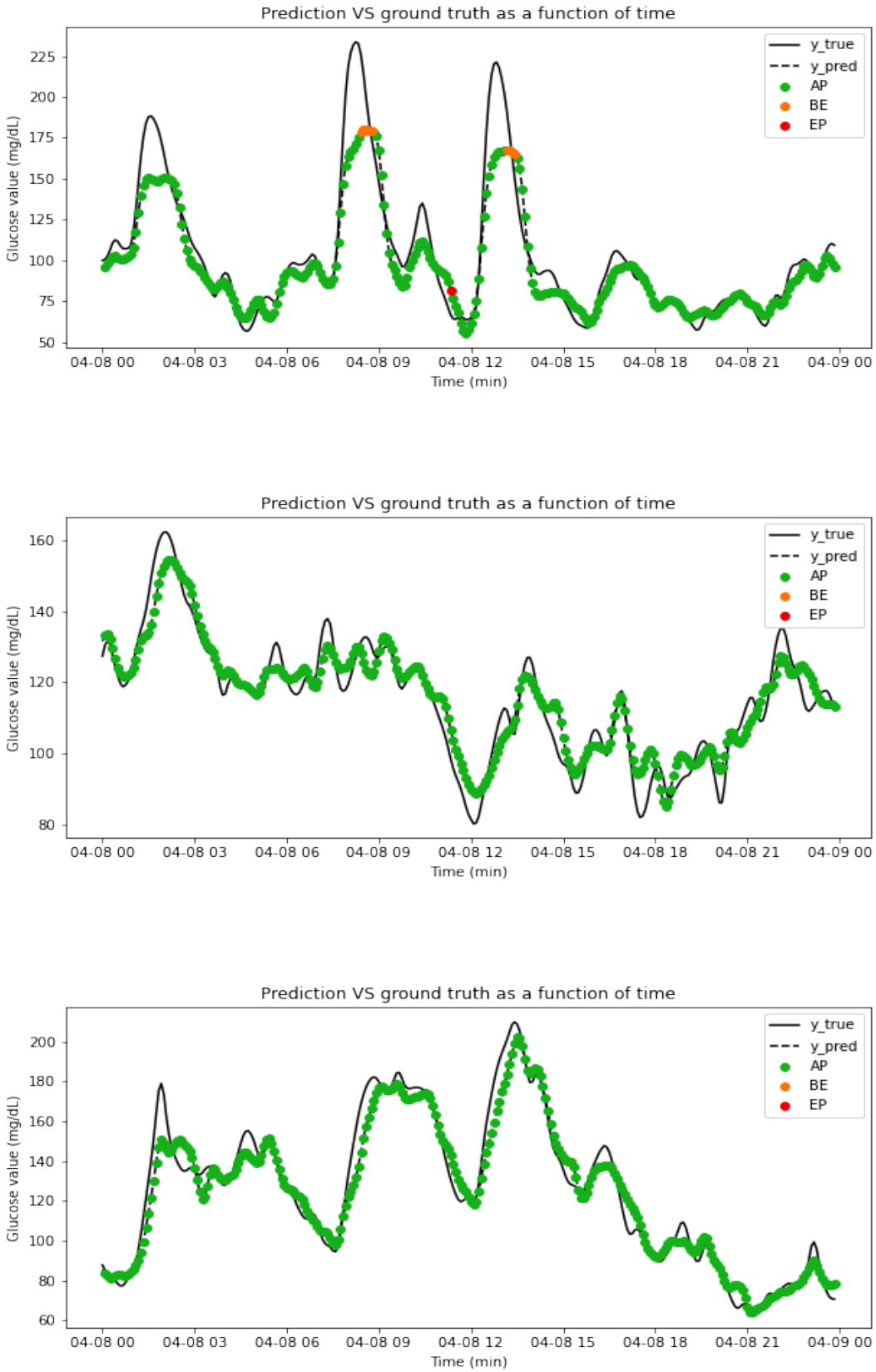

Fig. 39: MLP model results for one patient from each in silico dataset age group. Prediction Horizon: 30 minutes

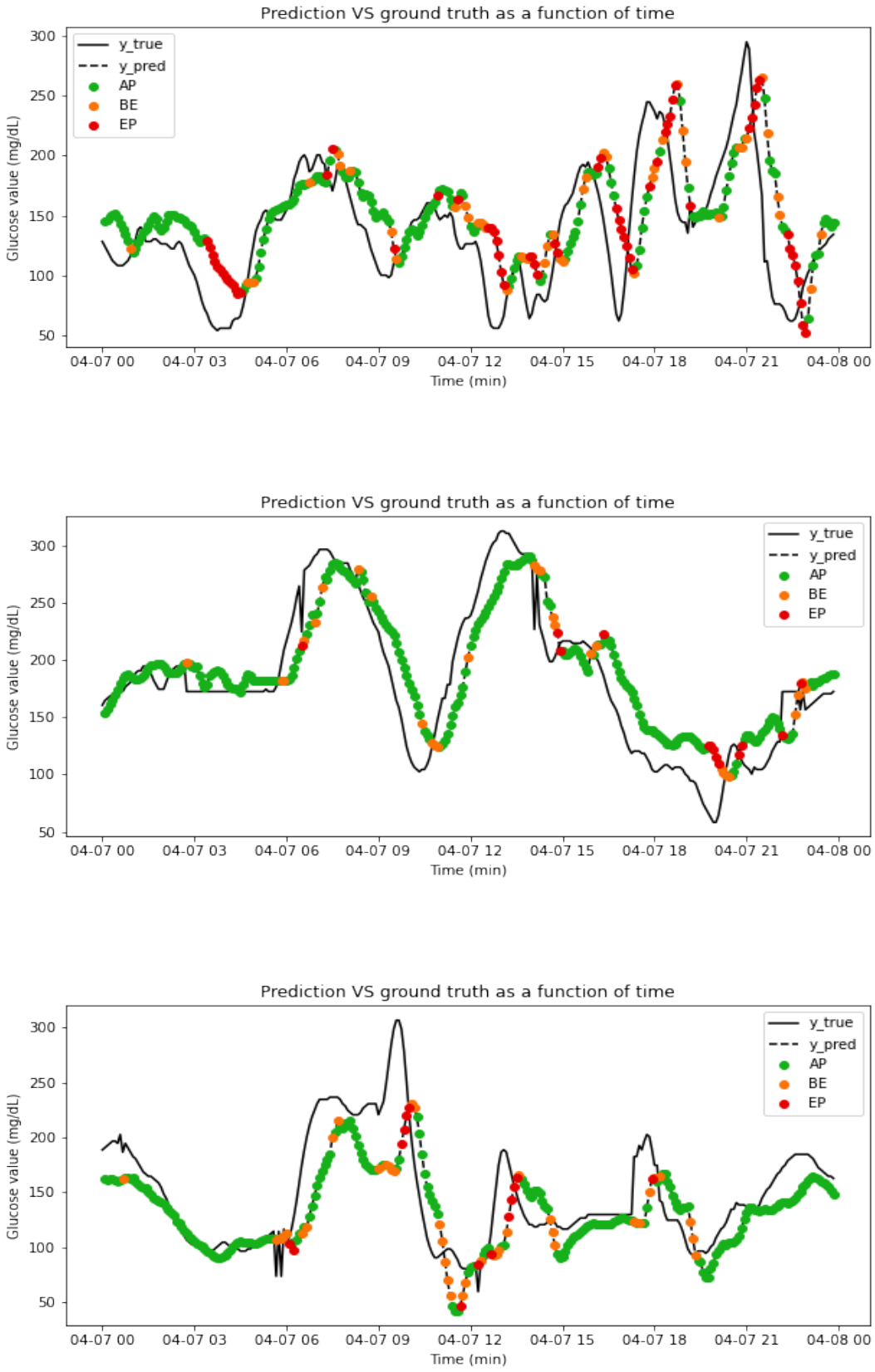

Fig. 40: MLP model results for patients 1 to 3. Prediction Horizon: 60 minutes

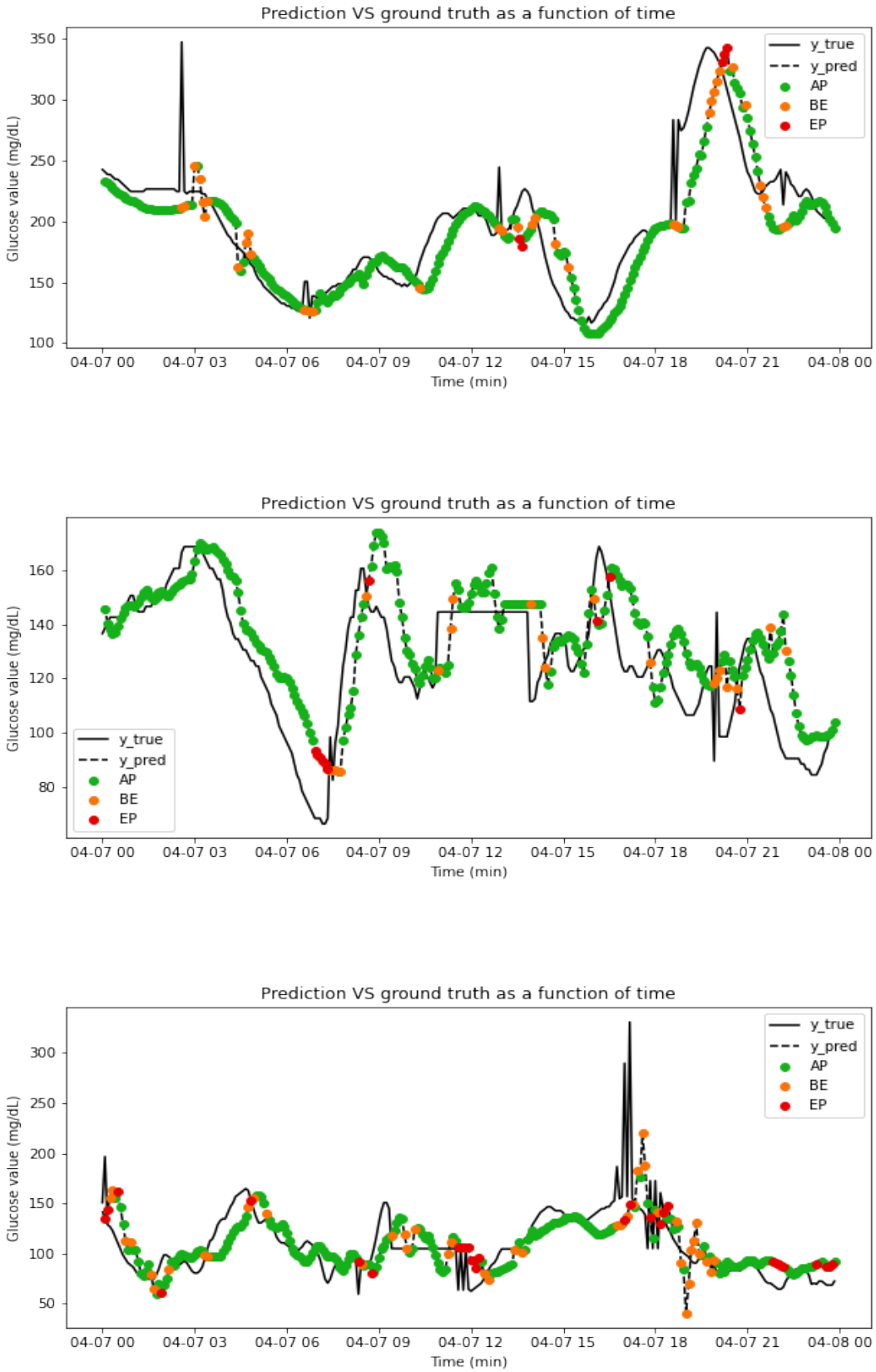

Fig. 41: MLP model results for patients 4 to 6. Prediction Horizon: 60 minutes

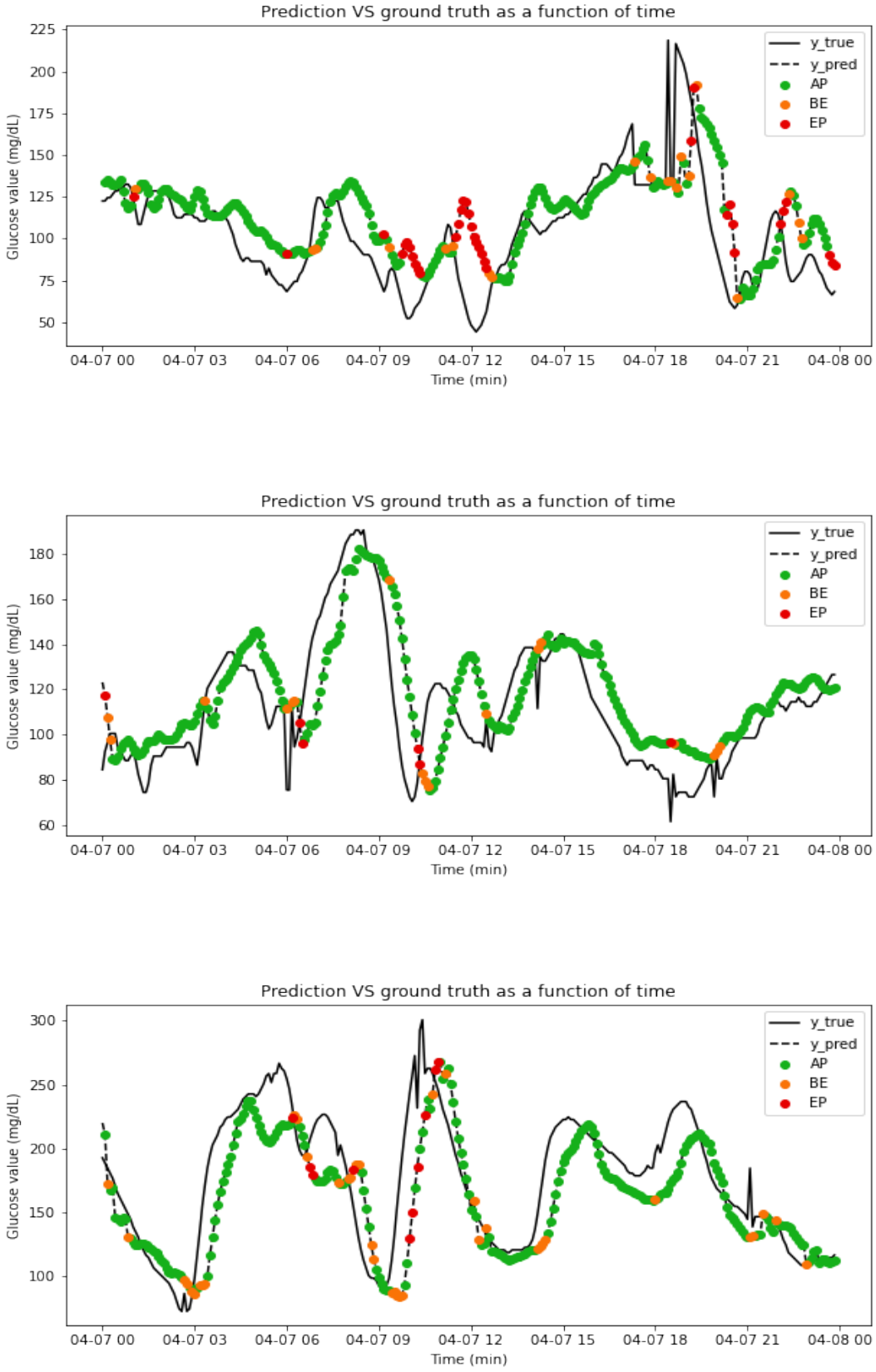

Fig. 42: MLP model results for patients 7 to 9. Prediction Horizon: 60 minutes

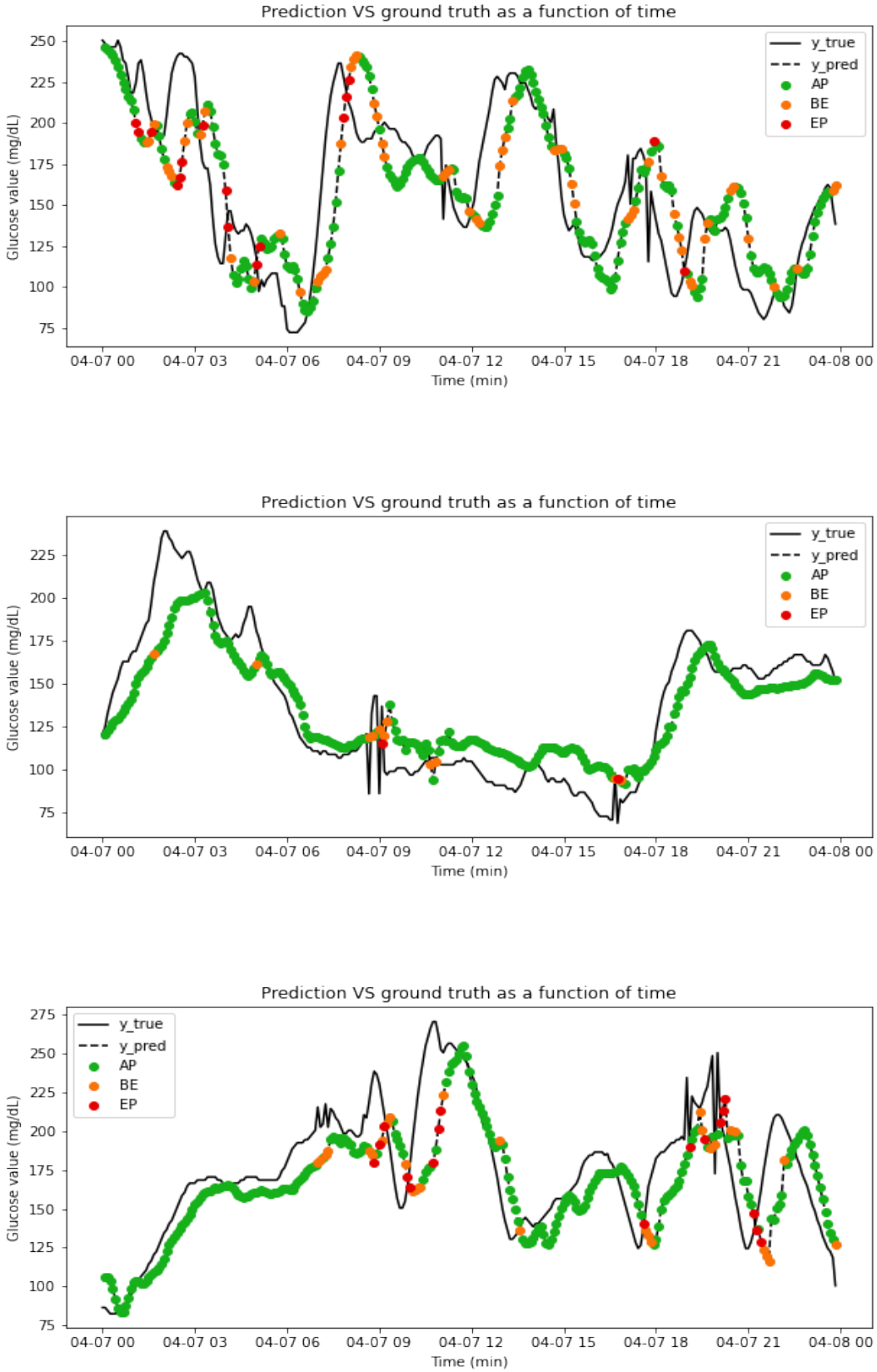

Fig. 43: MLP model results for patients 10 to 12. Prediction Horizon: 60 minutes

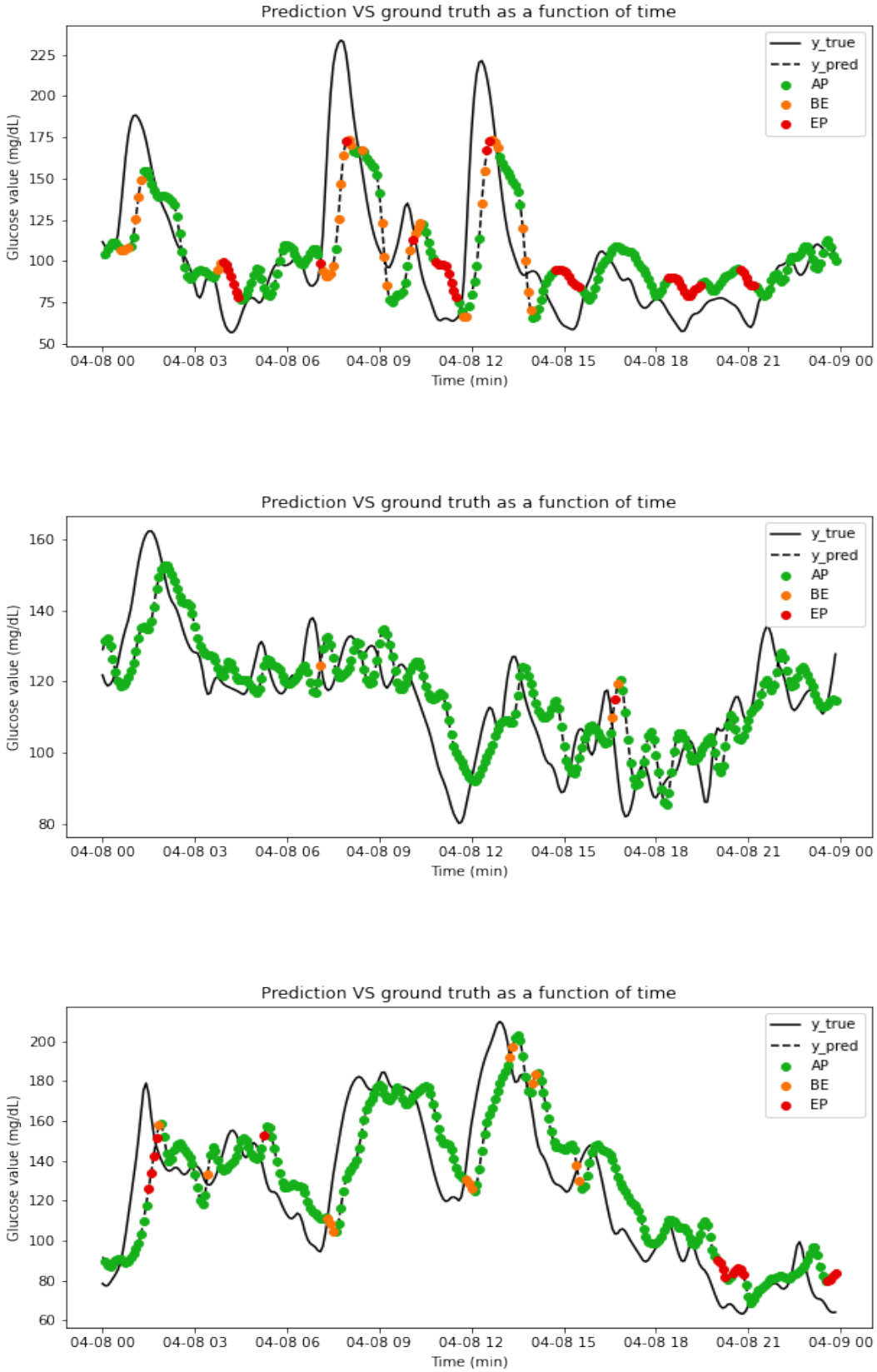

Fig. 44: MLP model results for one patient from each in silico dataset age group. Prediction Horizon: 60 minutes

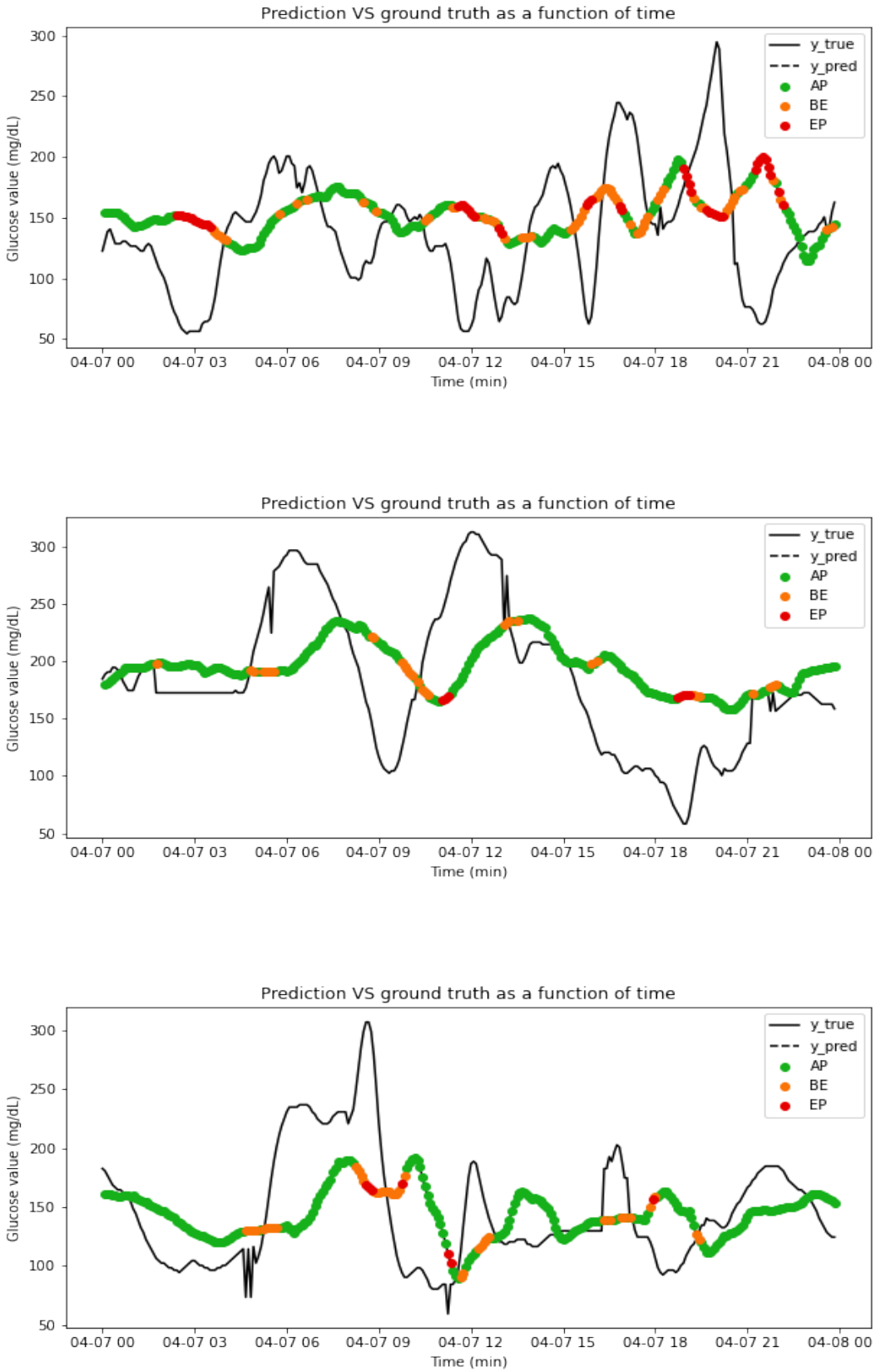

Fig. 45: MLP model results for patients 1 to 3. Prediction Horizon: 120 minutes

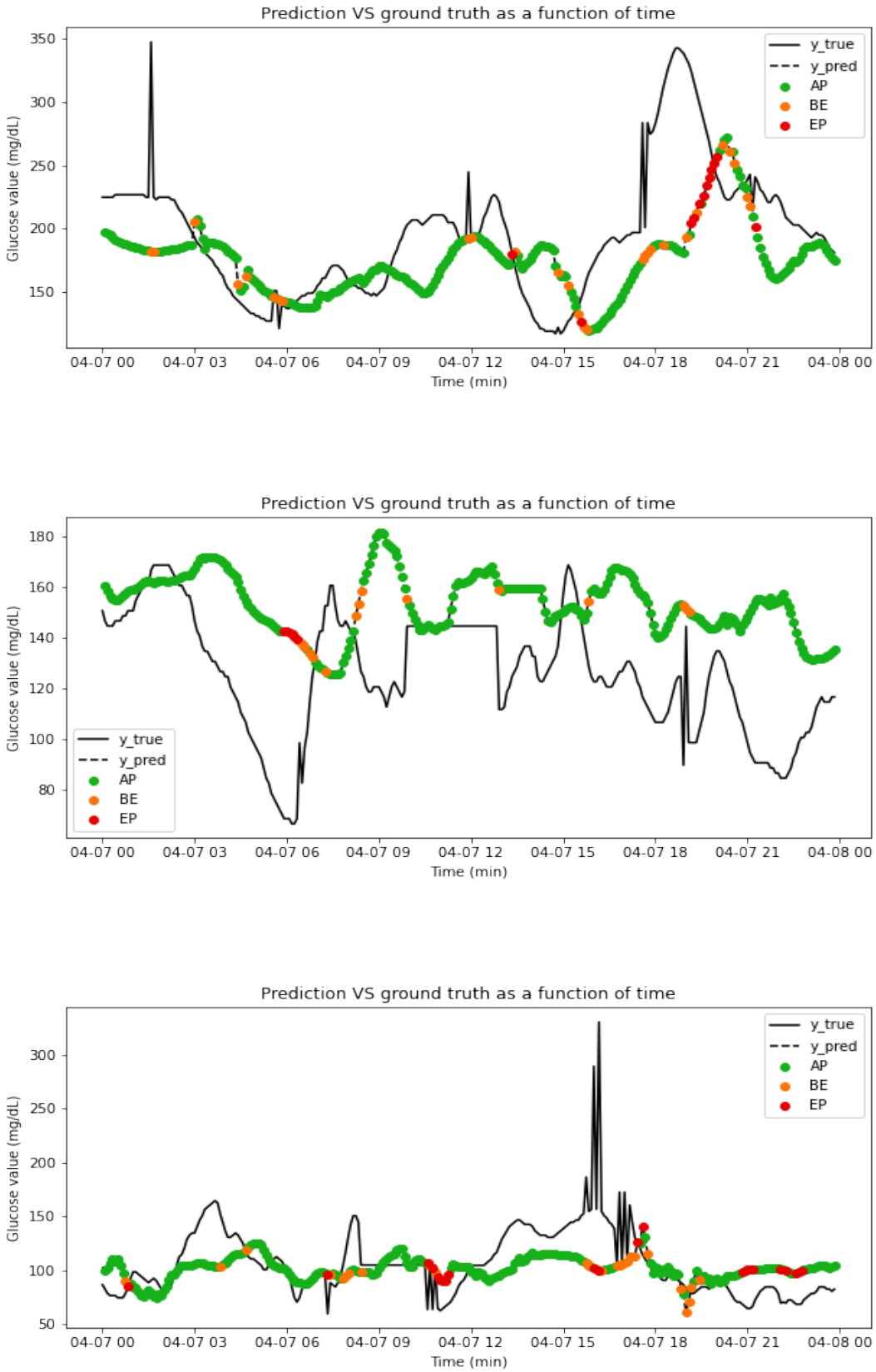

Fig. 46: MLP model results for patients 4 to 6. Prediction Horizon: 120 minutes

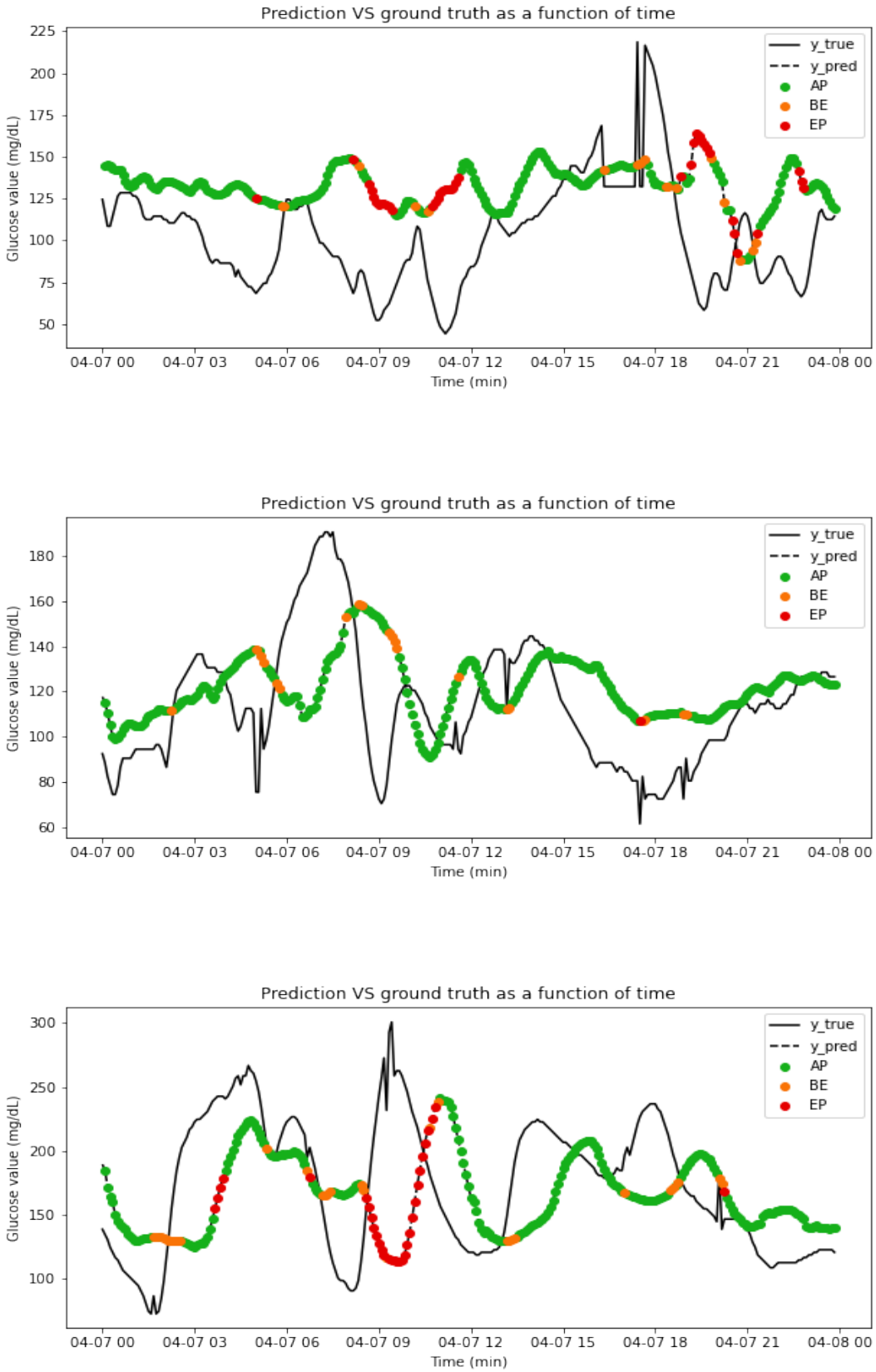

Fig. 47: MLP model results for patients 7 to 9. Prediction Horizon: 120 minutes

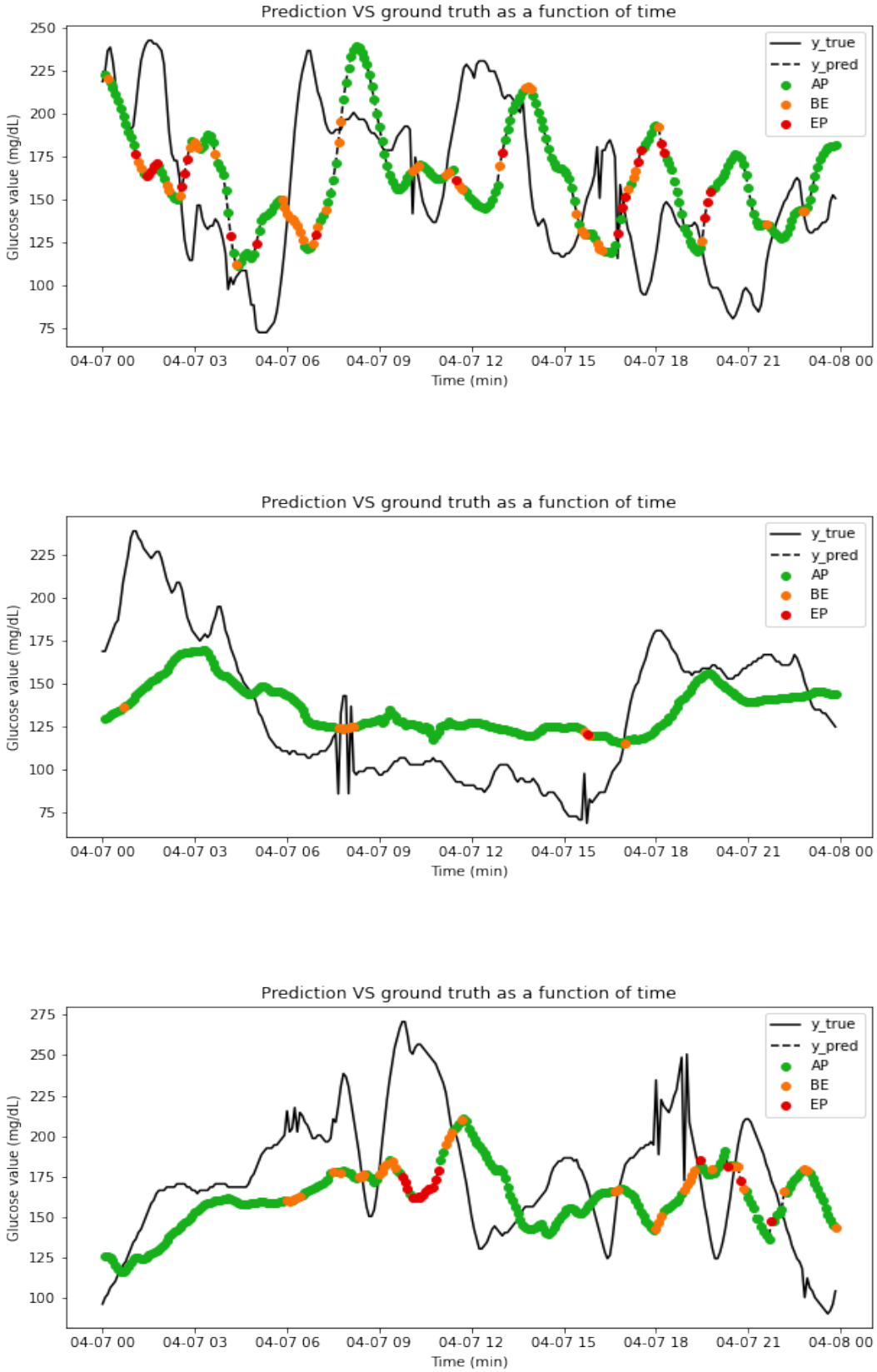

Fig. 48: MLP model results for patients 10 to 12. Prediction Horizon: 120 minutes

	GRU-Prediction Horizon 30 min					GRU-Prediction Horizon 60 min					GRU-Prediction Horizon 120 min				
	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC
Patient 1	18.892391	88.526037	8.384819	3.089144	0.966249	40.274475	68.766637	18.012422	13.220941	0.801749	63.269291	68.520179	18.295964	13.183857	0.326419
Patient 2	17.195335	93.027361	4.942630	2.030009	0.960387	41.832581	79.148181	14.551908	6.299911	0.734239	67.259438	67.982063	19.192825	12.825112	0.247361
Patient 3	14.509402	93.121693	4.850088	2.028219	0.954646	28.348448	79.875887	11.879433	8.244681	0.777506	42.766861	73.297491	18.996416	7.706093	0.421633
Patient 4	13.148166	95.326279	4.497354	0.176367	0.953192	27.350071	87.411348	10.195035	2.393617	0.776917	46.062668	79.301075	13.799283	6.899642	0.328951
Patient 5	11.467636	95.145631	2.912621	1.941748	0.946741	26.150654	86.690328	7.187223	6.122449	0.694724	41.730217	79.282511	12.825112	7.892377	0.242267
Patient 6	10.893461	94.008811	4.052863	1.938326	0.932108	20.328718	86.979628	5.845881	7.174491	0.686082	28.404961	83.616831	6.893465	9.489705	0.210077
Patient 7	14.577750	87.753304	2.907489	9.339207	0.952803	32.329914	77.236492	7.705934	15.057573	0.745157	47.586308	71.530886	13.607878	14.861235	0.399586
Patient 8	11.526854	95.066079	2.907489	2.026432	0.906421	19.757084	87.068202	8.325952	4.605846	0.709708	33.482487	84.780662	10.564011	4.655327	0.448742
Patient 9	17.020027	91.791703	6.884378	1.323919	0.958346	39.983471	75.865129	16.770186	7.364685	0.767269	73.456718	61.076233	20.627803	18.295964	0.145973
Patient 10	18.831707	87.466902	7.237423	5.295675	0.951286	37.582390	74.356699	14.374445	11.268855	0.783579	59.786617	65.560538	18.654709	15.784753	0.413447
Patient 11	14.845116	96.299559	2.907489	0.792952	0.933518	25.310377	90.345438	6.288751	3.365810	0.785687	39.378639	85.765443	7.341092	6.893465	0.380702
Patient 12	16.050974	93.915344	4.497354	1.587302	0.928557	32.759018	82.358156	11.879433	5.762411	0.661826	47.353935	81.182796	12.096774	6.720430	0.095203
	MLP-Prediction Horizon 30 min					MLP-Prediction Horizon 60 min					MLP-Prediction Horizon 120 min				
	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC
Patient 1	16.234322	87.819947	9.090909	3.089144	0.972024	35.487259	70.452529	17.568767	11.978705	0.855071	56.304840	72.556054	18.116592	9.327354	0.587814
Patient 2	13.408535	94.616064	3.353928	2.030009	0.974978	31.025957	84.383319	11.268855	4.347826	0.844323	55.841991	83.856502	11.479821	4.663677	0.326277
Patient 3	12.094470	94.532628	3.262787	2.204586	0.970692	28.823565	83.067376	10.017730	6.914894	0.794256	42.229961	84.946237	9.050179	6.003584	0.367564
Patient 4	10.950402	95.061728	4.585538	0.352734	0.968707	23.068766	87.943262	9.574468	2.482270	0.847942	38.660561	87.365591	9.050179	3.584229	0.459866
Patient 5	11.904263	94.616064	2.559576	2.824360	0.964623	21.742987	86.424135	7.897072	5.678793	0.809143	42.641312	87.354260	6.905830	5.739910	0.324825
Patient 6	10.355428	93.392070	4.317181	2.290749	0.935905	18.388359	82.196634	9.123118	8.680248	0.752769	26.820324	84.064458	5.640107	10.295434	0.256603
Patient 7	13.202324	96.211454	2.378855	1.409692	0.959589	27.162329	78.742250	7.794508	13.463242	0.795083	42.070217	80.035810	7.341092	12.623098	0.441129
Patient 8	9.221599	94.449339	3.788546	1.762115	0.941932	18.206858	88.928255	6.111603	4.960142	0.768817	27.336077	90.062668	5.908684	4.028648	0.366246
Patient 9	15.776894	91.350397	6.972639	1.676964	0.965193	41.318443	79.591837	15.084295	5.323869	0.820786	65.078415	70.941704	12.914798	16.143498	0.283283
Patient 10	14.348913	91.350397	6.619594	2.030009	0.971547	31.346500	70.718722	17.923691	11.357587	0.843483	49.973095	72.825112	16.143498	11.031390	0.571561
Patient 11	9.415866	96.828194	1.938326	1.233480	0.980375	20.682875	92.382640	4.605846	3.011515	0.891947	34.879848	91.584602	4.207699	4.207699	0.602390
Patient 12	12.184621	94.885362	4.409171	0.705467	0.963570	25.630928	86.081560	9.397163	4.521277	0.801930	40.323952	85.215054	10.842294	3.942652	0.331921

Fig. 49: Collective results of GRU and MLP models for real patients' data

	GRU-Prediction Horizon 30 min					GRU-Prediction Horizon 60 min					GRU-Prediction Horizon 120 min				
	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC
count	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
mean	12.354505	97.077510	1.702668	1.219822	0.956158	15.893426	93.188220	2.676056	4.135723	0.905075	32.508832	83.810143	5.877763	10.312094	0.647176
std	5.556977	2.401939	1.643895	1.358267	0.016304	4.980453	3.959299	1.733105	2.841401	0.067040	17.352149	7.424824	3.069827	5.060228	0.101553
min	6.404021	92.757306	0.000000	0.127065	0.936669	10.381623	85.275288	0.384123	1.664533	0.756176	12.732293	70.481144	0.780234	4.811443	0.483234
25%	7.835460	96.442186	0.254130	0.508259	0.944280	12.191740	93.085787	1.472471	2.016645	0.901996	23.668325	80.559168	4.746424	6.501951	0.571841
50%	12.546040	97.204574	1.651842	0.889454	0.950926	15.818130	94.302177	2.304738	3.201024	0.919285	29.113681	84.590377	6.566970	9.882965	0.667118
75%	14.371001	99.078780	2.287166	1.143583	0.970670	17.479722	96.030730	3.969270	4.929577	0.943091	38.074995	88.296489	8.387516	11.248375	0.720321
max	24.874054	99.872935	5.209657	4.828463	0.979997	27.889151	96.798976	5.889885	10.627401	0.978136	71.088638	94.408322	8.842653	21.066320	0.795110

	MLP-Prediction Horizon 30 min					MLP-Prediction Horizon 60 min					MLP-Prediction Horizon 120 min				
	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC
count	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
mean	11.773639	97.585769	1.130877	1.283355	0.965465	26.878444	81.792574	8.860435	9.346991	0.774933	42.054414	78.192458	6.475943	15.331599	0.303259
std	5.371407	3.059539	1.303335	1.988834	0.012864	14.870158	12.765171	6.676927	6.850425	0.062751	26.378869	13.299635	4.805309	9.189720	0.156924
min	5.787713	89.707751	0.000000	0.000000	0.950847	11.348889	54.417414	0.256082	2.688860	0.683023	16.373219	49.804941	0.130039	5.721717	0.072017
25%	7.389561	97.045743	0.000000	0.158831	0.955557	16.767357	77.624840	2.752881	4.993598	0.724350	27.908891	71.066320	2.568270	8.550065	0.160495
50%	12.219069	98.094028	0.762389	0.571792	0.960634	26.087239	81.690141	10.115237	6.658131	0.771076	36.298281	81.079324	6.501951	14.759428	0.327373
75%	14.204211	99.714104	2.033037	1.238882	0.976518	30.888011	92.253521	12.772087	11.267606	0.823640	48.781395	87.061118	10.500650	18.172952	0.402498
max	23.393539	100.000000	3.684879	6.607370	0.983941	62.600555	96.030730	21.510883	24.071703	0.868695	107.414001	94.148244	13.524057	36.671001	0.524390

Fig. 50: Collective results of GRU and MLP models for in silico children data

	GRU-Prediction Horizon 30 min					GRU-Prediction Horizon 60 min					GRU-Prediction Horizon 120 min				
	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC
count	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
mean	8.864441	99.263056	0.431988	0.304956	0.959442	12.596170	98.438113	1.203405	0.358482	0.900809	25.423682	93.459190	5.006350	1.534460	0.661984
std	1.976604	0.974734	0.719295	0.746217	0.027063	1.539911	1.241090	0.851850	0.741219	0.084442	7.688961	4.377565	3.100760	1.722765	0.115016
min	6.202753	97.331639	0.000000	0.000000	0.907725	10.674062	96.030730	0.256082	0.000000	0.746590	13.621196	84.265280	1.040312	0.000000	0.412331
25%	7.657103	99.396442	0.031766	0.000000	0.939969	11.682683	98.401084	0.896287	0.000000	0.842531	19.691350	91.254876	2.405722	0.260078	0.631516
50%	8.309815	99.618806	0.253968	0.063532	0.967901	12.044689	98.783611	1.024328	0.128041	0.936938	25.280671	94.668401	5.201560	0.780234	0.669583
75%	10.322196	99.841250	0.381194	0.127065	0.980262	12.962278	99.103713	1.375090	0.256000	0.959196	29.550904	96.163849	6.339402	2.568270	0.745508
max	12.477183	100.000000	2.414231	2.414231	0.984680	15.671826	99.743918	3.329065	2.432778	0.974652	37.314964	98.831169	10.533160	5.201560	0.804025

	MLP-Prediction Horizon 30 min					MLP-Prediction Horizon 60 min					MLP-Prediction Horizon 120 min				
	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC
count	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
mean	8.210732	99.822109	0.165184	0.012706	0.970712	19.399254	94.865737	4.122788	1.011475	0.817632	32.883890	92.782886	4.278233	2.938882	0.368222
std	2.368525	0.393696	0.359643	0.040181	0.020385	5.836262	3.323512	2.645524	1.025161	0.094467	12.819655	6.597038	2.934497	4.302525	0.139425
min	4.962057	98.856417	0.000000	0.000000	0.919274	10.740902	89.756722	1.023018	0.000000	0.570758	15.937193	78.413524	0.130039	0.000000	0.165079
25%	7.264142	100.000000	0.000000	0.000000	0.970098	16.621570	92.125480	2.048656	0.256082	0.810434	22.721705	89.076723	2.243173	0.097529	0.242929
50%	7.722662	100.000000	0.000000	0.000000	0.976428	19.122488	95.774648	3.777209	0.511918	0.835905	32.308611	95.383615	4.161248	1.495449	0.385654
75%	9.218261	100.000000	0.000000	0.000000	0.981883	23.751687	97.311140	6.754161	1.856594	0.873841	41.500383	97.269181	7.152146	3.673602	0.459971
max	12.193980	100.000000	1.016518	0.127065	0.989208	28.906946	98.593350	7.554417	2.688860	0.899787	55.882278	99.610390	7.802341	13.914174	0.589924

Fig. 51: Collective results of GRU and MLP models for in silico adolescents data

	GRU-Prediction Horizon 30 min					GRU-Prediction Horizon 60 min					GRU-Prediction Horizon 120 min				
	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC
count	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
mean	7.550709	99.771283	0.114358	0.114358	0.954159	12.689116	98.207426	1.024328	0.768246	0.855997	19.710589	96.189857	2.574772	1.235371	0.614956
std	1.357671	0.321451	0.139835	0.211351	0.012779	2.105162	1.697583	0.825398	1.031417	0.042472	4.471942	2.457435	1.689729	1.224870	0.110104
min	5.892049	98.983482	0.000000	0.000000	0.926670	10.148026	94.238156	0.128041	0.000000	0.797578	15.212330	92.457737	1.040312	0.000000	0.423411
25%	6.591548	99.745870	0.000000	0.000000	0.948504	11.368925	97.887324	0.352113	0.032010	0.822414	16.319117	93.790637	1.235371	0.292588	0.544094
50%	7.544500	99.872935	0.063532	0.000000	0.954886	11.842073	98.655570	0.832266	0.384123	0.861692	17.905090	96.879064	1.560468	0.845254	0.616895
75%	8.060529	100.000000	0.222363	0.190597	0.964701	14.167733	99.327785	1.632522	1.184379	0.874797	22.803583	98.276983	4.063719	1.983095	0.686524
max	9.728820	100.000000	0.381194	0.635324	0.967541	16.219912	99.871959	2.432778	3.329065	0.928185	28.284880	98.699610	5.591678	3.250975	0.793893

	MLP-Prediction Horizon 30 min					MLP-Prediction Horizon 60 min					MLP-Prediction Horizon 120 min				
	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC	RMSE	Zone A	Zone B	Zone E	CC
count	10.000000	10.000000	10.0	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000	10.000000
mean	6.997410	99.961881	0.0	0.038119	0.962777	15.348079	96.030730	2.676056	1.293214	0.783911	23.670108	97.087126	1.495449	1.417425	0.302694
std	1.025625	0.085762	0.0	0.085762	0.013039	2.988397	2.617804	2.101719	0.985259	0.065195	5.063611	3.145045	1.496861	2.373741	0.148345
min	5.708236	99.745870	0.0	0.000000	0.933061	11.643366	91.933419	0.384123	0.384123	0.632273	17.898304	89.726918	0.000000	0.000000	0.092456
25%	6.173727	100.000000	0.0	0.000000	0.959654	12.944038	93.501921	0.960307	0.480154	0.769179	19.139359	96.228869	0.227568	0.000000	0.169024
50%	6.969598	100.000000	0.0	0.000000	0.963538	14.514661	97.375160	1.920615	1.024328	0.785978	22.538851	98.309493	0.910273	0.195059	0.330843
75%	7.445566	100.000000	0.0	0.000000	0.973024	17.675703	97.695262	4.001280	1.696543	0.828675	28.882710	99.219766	2.795839	1.918075	0.434675
max	9.084062	100.000000	0.0	0.254130	0.976120	20.386848	98.847631	6.786172	3.329065	0.864536	30.122854	99.609883	4.161248	7.412224	0.472473

Fig. 52: Collective results of GRU and MLP models for in silico adults data

Chapter references

[Alf+20] Ganjar Alfian et al. "Blood glucose prediction model for type 1 diabetes based on artificial neural network with time-domain features". In: *Biocybernetics and Biomedical Engineering* 40.4 (Oct. 2020), pp. 1586–1599. DOI: 10.1016/j.bbe.2020.10.004.

[CAK08] William Clarke, Stacey Anderson, and Boris Kovatchev. "Evaluating Clinical Accuracy of Continuous Glucose Monitoring Systems:Continuous Glucose – Error Grid Analysis (CG-EGA)". In: *Current Diabetes Reviews* 4.3 (2008), pp. 193–199. DOI: 10.2174/157339908785294389.

Chapter 8

Conclusion and future work

In the present work a hybrid model was developed, based on 3 CMs combined with two different ANNs: a recurrent one and the current state of the art MLP for glucose prediction. Important points to be considered are that both models benefit from being integrated in a hybrid model (increase of performance), and that MLP performs best in the presence of only one feature. The above fact leaves more room for improvement for recurrent models, which are the go-to method for time series forecasting problems. Improvements can be made in a variety of sectors, some of which will be mentioned in this chapter, concluding the present thesis.

Data windowing

In the present work, a look-back windows of 6 observations (30 minutes) was used, which is the most common in suck problems. However, different data windows can be used, and perhaps even the state of the art adaptive data windowing method [Ben+15], in which pre-trained ANNs are used to optimize data the data windowing procedure.

Optimizers

Adam was found to be the best optimizer among the standard ones that were tested. However, the idea of using RNNs instead of a fixed algorithm for optimization purposes [And+16] can also be implemented in the present context, and possibly lead to improvement of results.

Transfer learning

In the context of the present work, personalized models were developed for each patient in the population. Using pre-trained models on different datasets

and then training and evaluating them on different data is the process known as transfer learning. It has the potential to lead to better generalization of the model by learning different relations for each patient, and then approximating the glucose prediction equation with better accuracy, and in a way that adapts to a wider range of the population.

More accurate CMs

Since hybrid models were developed, more recent CMs could be used to increase reliability of the data fed to the training process of the models. Another idea to be tested would be to construct CMs which are calculated via ANN outputs and then feed those outputs to train a predictive model. Different multilevel modelling approaches can also be utilized, with addition of more CMs. For example, to the author's knowledge, no CM for glucagon has been utilized in this context in the literature.

Training/validation/testing on different datasets

With the purpose of generalizing the model, using different patients' data for each step in training, validation and testing phase of the model could be implemented. A drawback of this method is that an ample amount of data would be needed, along with a possibly high model complexity, significantly increasing the number of parameters, and the computational power needed as a result.

With diabetes on the rise globally, demand for better all-in-one control systems is also on the rise. In the current thesis a hybrid multilevel modelling approach for predicting glucose values was presented, with its possible applications and drawbacks. More work towards this direction can lead to future devices that could enable T1DM patients to live a normal life, free of much of the complications of this disease.

Chapter references

[And+16] Marcin Andrychowicz et al. "Learning to Learn by Gradient Descent by Gradient Descent". In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS'16. Barcelona, Spain: Curran Associates Inc., 2016, pp. 3988–3996. ISBN: 9781510838819.

[Ben+15] Yahyia Benyahmed et al. "Adaptive sliding window algorithm for weather data segmentation". In: *Journal of theoretical and applied information technology* 80 (2015), pp. 322–333.

Bibliography

[Alf+20] Ganjar Alfian et al. "Blood glucose prediction model for type 1 diabetes based on artificial neural network with time-domain features". In: *Biocybernetics and Biomedical Engineering* 40.4 (Oct. 2020), pp. 1586–1599. DOI: 10.1016/j.bbe.2020.10.004.

[Ali+18] Jaouher Ben Ali et al. "Continuous blood glucose level prediction of Type 1 Diabetes based on Artificial Neural Network". In: *Biocybernetics and Biomedical Engineering* 38.4 (2018), pp. 828–840. DOI: 10.1016/j.bbe.2018.06.005.

[And+16] Marcin Andrychowicz et al. "Learning to Learn by Gradient Descent by Gradient Descent". In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS'16. Barcelona, Spain: Curran Associates Inc., 2016, pp. 3988–3996. ISBN: 9781510838819.

[Ben+15] Yahyia Benyahmed et al. "Adaptive sliding window algorithm for weather data segmentation". In: *Journal of theoretical and applied information technology* 80 (2015), pp. 322–333.

[CAK08] William Clarke, Stacey Anderson, and Boris Kovatchev. "Evaluating Clinical Accuracy of Continuous Glucose Monitoring Systems:Continuous Glucose – Error Grid Analysis (CG-EGA)". In: *Current Diabetes Reviews* 4.3 (2008), pp. 193–199. DOI: 10.2174/157339908785294389.

[Cgm] *Healthline: Everything You Need to Know About Insulin*. https://www.healthline.com/diabetesmine/what-is-continuous-glucose-monitor-and-choosing-one.

[Cho+14] Kyunghyun Cho et al. "On the Properties of Neural Machine Translation: Encoder–Decoder Approaches". In: *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*. Doha, Qatar: Association for Computational Linguistics, Oct. 2014, pp. 103–111. DOI: 10.3115/v1/W14-4012. URL: https://www.aclweb.org/anthology/W14-4012.

[Das+12] Elena Daskalaki et al. "Real-Time Adaptive Models for the Personalized Prediction of Glycemic Profile in Type 1 Diabetes Patients". In: *Diabetes Technology & Therapeutics* 14.2 (2012), pp. 168–174. DOI: 10.1089/dia.2011.0093.

[Fed19] International Diabetes Federation. *IDF Diabetes Atlas-Ninth Edition 2019*. 2019.

[GBC16] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. The MIT Press, 2016. ISBN: 0262035618.

[Geo+13] E. I. Georga et al. "Multivariate Prediction of Subcutaneous Glucose Concentration in Type 1 Diabetes Patients Based on Support Vector Regression". In: *IEEE Journal of Biomedical and Health Informatics* 17.1 (2013), pp. 71–81. DOI: 10.1109/titb.2012.2219876.

[Geo+15] Eleni I. Georga et al. "Evaluation of short-term predictors of glucose concentration in type 1 diabetes combining feature ranking with regression models". In: *Medical & Biological Engineering & Computing* 53.12 (2015), pp. 1305–1318. DOI: 10.1007/s11517-015-1263-1.

[GGJ13] Winston Garcia-Gabin and Elling W. Jacobsen. "Multilevel Model of Type 1 Diabetes Mellitus Patients for Model-Based Glucose Controllers". In: *Journal of Diabetes Science and Technology* 7.1 (Jan. 2013), pp. 193–201. DOI: 10.1177/193229681300700125.

[Gir+09] Céline M. Girardin et al. "Continuous glucose monitoring: A review of biochemical perspectives and clinical use in type 1 diabetes". In: *Clinical Biochemistry* 42.3 (2009), pp. 136–142. DOI: 10.1016/j.clinbiochem.2008.09.112.

[Ham+18] Takoua Hamdi et al. "Accurate prediction of continuous blood glucose based on support vector regression and differential evolution algorithm". In: *Biocybernetics and Biomedical Engineering* 38.2 (2018), pp. 362–372. DOI: 10.1016/j.bbe.2018.02.005.

[Hov+04] Roman Hovorka et al. "Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes". In: *Physiological Measurement* 25.4 (2004), pp. 905–920. DOI: 10.1088/0967-3334/25/4/010.

[HS97] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-term Memory". In: *Neural computation* 9 (Dec. 1997), pp. 1735–80. DOI: 10.1162/neco.1997.9.8.1735.

[Kar+17] Anuj Karpatne et al. "Physics-guided Neural Networks (PGNN): An Application in Lake Temperature Modeling". In: (Oct. 2017).

[KB14] Diederik Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: *International Conference on Learning Representations* (Dec. 2014).

[Lst] *Illustrated Guide to LSTM's and GRU's: A step by step explanation*. https://towardsdatascience.com/illustrated-guide-to-lstms-and-gru-s-a-step-by-step-explanation-44e9eb85bf21.

[Man+14] Chiara Dalla Man et al. "The UVA/PADOVA Type 1 Diabetes Simulator". In: *Journal of Diabetes Science and Technology* 8.1 (2014), pp. 26–34. DOI: 10.1177/1932296813514502.

[Mar+19] John Martinsson et al. "Blood Glucose Prediction with Variance Estimation Using Recurrent Neural Networks". In: *Journal of Healthcare Informatics Research* 4.1 (2019), pp. 1–18. DOI: 10.1007/s41666-019-00059-y.

[Mit97] Tom Mitchell. *Machine Learning*. New York: McGraw-Hill, 1997. ISBN: 9780071154673.

[MRC07] Chiara Dalla Man, Robert A. Rizza, and Claudio Cobelli. "Meal Simulation Model of the Glucose-Insulin System". In: *IEEE Transactions on Biomedical Engineering* 54.10 (Oct. 2007), pp. 1740–1749. DOI: 10.1109/tbme.2007.893506.

[NC00] Gianluca Nucci and Claudio Cobelli. "Models of subcutaneous insulin kinetics. A critical review". In: *Computer Methods and Programs in Biomedicine* 62.3 (July 2000), pp. 249–257. DOI: 10.1016/s0169-2607(00)00071-7.

[Pap+11] Scott M. Pappada et al. "Neural Network-Based Real-Time Prediction of Glucose in Patients with Insulin-Dependent Diabetes". In: *Diabetes Technology & Therapeutics* 13.2 (2011), pp. 135–141. DOI: 10.1089/dia.2010.0104.

[PG+10] C. Pérez-Gandía et al. "Artificial Neural Network Algorithm for Online Glucose Prediction from Continuous Glucose Monitoring". In: *Diabetes Technology & Therapeutics* 12.1 (2010), pp. 81–88. DOI: 10.1089/dia.2009.0076.

[PGD00] R. S. Parker, E. P. Gatzke, and F. J. Doye. "Advanced model predictive control (MPC) for type I diabetic patient blood glucose control". In: *Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No.00CH36334)*. Vol. 5. 2000, 3483–3487 vol.5. DOI: 10.1109/ACC.2000.879216.

[Sai+20] Kyriaki Saiti et al. "Ensemble methods in combination with compartment models for blood glucose level prediction in type 1 diabetes mellitus". In: *Computer Methods and Programs in Biomedicine* 196 (2020), p. 105628. DOI: 10.1016/j.cmpb.2020.105628.

[SG64] Abraham. Savitzky and M. J. E. Golay. "Smoothing and Differentiation of Data by Simplified Least Squares Procedures." In: *Analytical Chemistry* 36.8 (July 1964), pp. 1627–1639. DOI: 10.1021/ac60214a047.

[SPU71] Louis M. Sherwood, Edith E. Parris, and Roger H. Unger. "Glucagon Physiology and Pathophysiology". In: *New England Journal of Medicine* 285.8 (1971), pp. 443–449. DOI: 10.1056/nejm197108192850806.

[TLT02] J.A. Tamada, M. Lesho, and M.J. Tierney. "Keeping watch on glucose". In: *IEEE Spectrum* 39.4 (2002), pp. 52–57. DOI: 10.1109/6.993789.

[Tur+13] Kamuran Turksoy et al. "Hypoglycemia Early Alarm Systems Based on Multivariable Models". In: *Industrial & Engineering Chemistry Research* 52.35 (May 2013), pp. 12329–12336. DOI: 10.1021/ie3034015.

[WWM13] Youqing Wang, Xiangwei Wu, and Xue Mo. "A Novel Adaptive-Weighted-Average Framework for Blood Glucose Prediction". In: *Diabetes Technology & Therapeutics* 15.10 (2013), pp. 792–801. DOI: 10.1089/dia.2013.0104.

[Zar+11] K. Zarkogianni et al. "An Insulin Infusion Advisory System Based on Autotuning Nonlinear Model-Predictive Control". In: *IEEE Transactions on Biomedical Engineering* 58.9 (2011), pp. 2467–2477. DOI: 10.1109/TBME.2011.2157823.

[Zar+15] K. Zarkogianni et al. "Comparative assessment of glucose prediction models for patients with type 1 diabetes mellitus applying sensors for glucose and physical activity monitoring". In: *Medical & Biological Engineering & Computing* 53.12 (2015), pp. 1333–1343. DOI: 10.1007/s11517-015-1320-9.

[Zec+14] C. Zecchin et al. "Jump neural network for online short-time prediction of blood glucose from continuous monitoring sensors and meal information". In: *Computer Methods and Programs in Biomedicine* 113.1 (2014), pp. 144–152. DOI: 10.1016/j.cmpb.2013.09.016.

[ZZ99] Ling Zhang and Bo Zhang. "A geometrical representation of McCulloch-Pitts neural model and its applications". In: *IEEE Transactions on Neural Networks* 10.4 (1999), pp. 925–929. DOI: 10.1109/72.774263.