Integration of clinical criteria into the training of deep models: Application to glucose prediction for diabetic people

Maxime de Bois, Mounim El Yacoubi, Mehdi Ammi

Contents lists available at ScienceDirect

Smart Health

journal homepage: www.elsevier.com/locate/smhl

Integration of Clinical Criteria into the Training of Deep Models: Application to Glucose Prediction for Diabetic People

Maxime De Bois[a], Mounîm A. El Yacoubi[b], Mehdi Ammi[c]

[a]*CNRS-LIMSI, université Paris-Saclay, Orsay, France*
[b]*Samovar, CNRS, Télécom SudParis, Institut Polytechnique de Paris, Palaiseau, France*
[c]*université Paris 8, Saint-Denis, France*

ARTICLE INFO

Communicated by -

2000 MSC:
68T20
92C50

Keywords:
deep learning
clinical acceptability
neural network
glucose prediction
diabetes

ABSTRACT

The standard way to train neural-network-based solutions in healthcare does not consider clinical criteria, leading to models that are not necessarily clinically acceptable. In this study, we look at this problem from the perspective of the forecasting of future glucose values of people with diabetes. We propose a new training methodology that achieves the best possible tradeoff between accuracy and medical requirements set by health authorities. Starting from a solution maximizing the prediction accuracy, we progressively relax the accuracy constraints to focus more on the medical ones. This is achieved by considering a new loss function specifically designed for glucose prediction. We evaluate the proposed approach on both people with type-1 and type-2 diabetes. We show that it improves the clinical acceptability of the predictions. Moreover, for given clinical criteria, we are able to find the optimal solution that maximizes the accuracy while at the same time meeting clinical the criteria.

1. Introduction

With 4.2 million of imputed deaths in 2019, diabetes is undoubtedly one of the major diseases of our modern world (Federation (2019)). There are three main categories of diabetes: type-1 diabetes mellitus, type-2 diabetes mellitus and gestational diabetes. Compared to healthy persons, people with diabetes experience trouble in the regulation of their blood glucose level within an acceptable range (homeostasis around 90 mg/dL). The pancreas is responsible for most of the regulation in healthy individuals, releasing two different hormones: the insulin and the glucagon (see Figure 1). However, for people with diabetes, this negative feedback loop is damaged. In type-1 diabetes, the pancreas does not secrete insulin anymore. On the other hand, in type-2 diabetes, the body cells get increasingly resistant to the action of insulin causing the pancreas to not be able to produce enough insulin. People with diabetes can still achieve the regulation of blood glucose through the use of medication and the careful monitoring of several aspects of their life such as the food they eat or their physical activity. However, this task is very difficult and can lead to severe consequences if not done correctly. Failing to regulate the blood glucose level puts the person with diabetes at risk of getting in states of hypoglycemia and hyperglycemia. In hypoglycemia (blood glucose level below 70 mg/dL), the person faces short-term consequences such as clumsiness, trouble talking, loss of consciousness or even death depending on the severity of the hypoglycemia. On the other hand, with hyperglycemia (blood glucose level above 180 mg/dL), the consequences are more long-term with an increased risk of cardiovascular diseases, amputation because of poor blood flow, or blindness.

In the recent years, a lot of researchers have been interested in the creation of glucose predictive models (Oviedo et al. (2017)). Using past glucose values, *carbohydrate* (CHO) intakes and insulin infusions information, the models can forecast the future glucose values 30 to 60 minutes ahead of time (Oviedo et al. (2017)). For people with diabetes, being able to know the future values of their glycemia could be highly beneficial as hypo/hyperglycemia events could be anticipated. Historically, glucose predictive models were based on autoregressive processes (Sparacino

e-mail: maxime.debois@limsi.fr (Maxime De Bois)

http://dx.doi.org/10.1016/j.smhl.xxxx.xx.xxx
Received 17 January 2021; Received in final form -; Accepted -; Available online -
2352-6483/

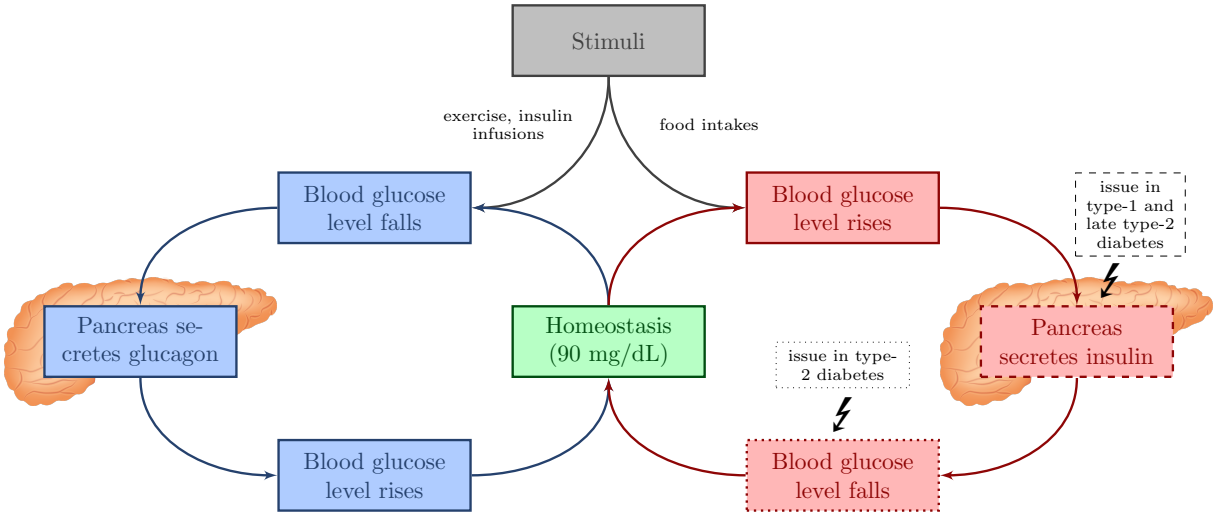


Fig. 1: Blood glucose level negative feedback loop.

et al. (2007); Saiti et al. (2020)). However, thanks to the advance in machine learning, but also to the increased availability of data, we are currently witnessing a shift in favor of more complex models, and in particular models based on artificial neural networks. The use of standard feedforward neural networks has been explored with, for instance, the works of Pappada et al. (2011), Georga et al. (2013) and Ali et al. (2018). Recurrent neural networks, and in particular those based on *long short-term memory* (LSTM) units, are probably the most popular deep models for glucose prediction. Aliberti et al. (2019) showed that they are more accurate than standard autoregressive models. Mirshekarian et al. (2017) demonstrated their superiority over *support vector regression* (SVR) models that use expert physiological features. Moreover, they have also been shown to benefit from the addition of various input features such as the heart rate or the skin conductance (Mirshekarian et al. (2019); Martinsson et al. (2019)). Lastly, other neural-network-based solutions have been recently tried out. Among them, we can highlight the promising use of convolutional neural networks (De Bois et al. (2020b); Zhu et al. (2018)).

Models based on neural networks are trained by backpropagating the gradient of the average error to the weights of the network. In glucose prediction, as in almost all regression problems, the average error is computed as the mean squared error (MSE). As a consequence, the models are trained to maximize the accuracy of the predictions. However, in the benchmark study we recently conducted (De Bois et al. (2020a)), we showed that a good statistical accuracy does not ensure that the predictions are clinically acceptable. Indeed, some errors, despite their relatively low magnitude, can be very dangerous for the person with diabetes (e.g., errors in the hypoglycemia region). To address this issue, Del Favero *et al.* proposed the *glucose mean-squared error* (gMSE) loss function that amplifies the weighting of the errors based on the observed glycemic region (Del Favero et al. (2012)). They showed that using the gMSE instead of the standard MSE decreases the number of dangerous predictions at the cost of reducing the average statistical accuracy of the model. While their methodology is promising, their study has several limitations that we aim at addressing. First, as the approach has been evaluated on virtual people with diabetes using autoregressive models, it is unclear how it translates to more complex models and to real people. Also, their approach focuses on only one aspect of the clinical acceptability of the predictions, which is the point clinical accuracy. Another aspect of the clinical acceptability of the predictions is the clinical accuracy of predicted variations (i.e., the difference between two successive predictions compared to the observed variations), which is taken into account in the widely used *continuous glucose-error grid analysis* (CG-EGA) metric (Kovatchev et al. (2004)). Indeed, inaccurate predicted glucose variations can be very dangerous as they can confuse the person with diabetes in the understanding of the future evolution of his/her glycemia.

Our contributions are:

1. We propose the *coherent mean squared glycemic error* (gcMSE) loss function. Compared to the standard MSE loss function, it includes constraints directly related to the clinical acceptability of the predictions. In particular, it penalizes the model during its training not only on prediction errors, but also on predicted variations errors (De Bois et al. (2019)). Moreover, it makes possible to increase the importance of specific regions in the error space (e.g., the hypoglycemia region).
2. Optimizing the parameters of the gcMSE loss function is a multi-objective optimization problem. Indeed, by incentivizing the model to focus more making clinically acceptable predictions, we reduce the statistical accuracy constraints. However, for the model to be useful for the people with diabetes, the predictions needs to be accurate. To address this challenge, we propose the PICA (*progressive improvement of the clinical acceptability*) algorithm that iteratively relaxes the accuracy constraints so that the focus of the learning is progressively more in favor on the satisfaction of the clinical constraints. This enables the creation of a model that maximizes the accuracy while at the same time that satisfying the given clinical constraints.
3. We evaluate the proposed solutions on two diabetes datasets, the IDIAB dataset and the OhioT1DM dataset, characterized by their heterogeneity. Whereas the IDIAB dataset, collected by ourselves, is made of 6 individuals with type-2 diabetes, the OhioT1DM dataset has been released by Marling *et al.* and comprises data from 6 individuals with type-1 diabetes (Marling & Bunescu (2018)).
4. We have open-sourced the code written in Python that has been used in this study in a GitHub repository (De Bois (2020)).

The paper is organized as follows. First, after introducing the CG-EGA metric in more details, we present the whole framework of integrating clinical acceptability criteria within the training of deep models. Then, we describe the machine learning pipeline, with the preprocessing of the data, the models we used, and the evaluation process. Finally, before concluding, we present and discuss the experimental results.

2. Integrating Clinical Criteria into the Training of Deep Models

In this section we propose a method to integrate clinical criteria based on the CG-EGA into the training of deep models. First, we introduce the CG-EGA metric, how it is computed and used to assess the clinical acceptability of the predictions. Then, we present the gcMSE loss function that integrates the clinical constraints. Finally, we propose a methodology to use this new loss function in practice.

2.1. Presentation of the CG-EGA

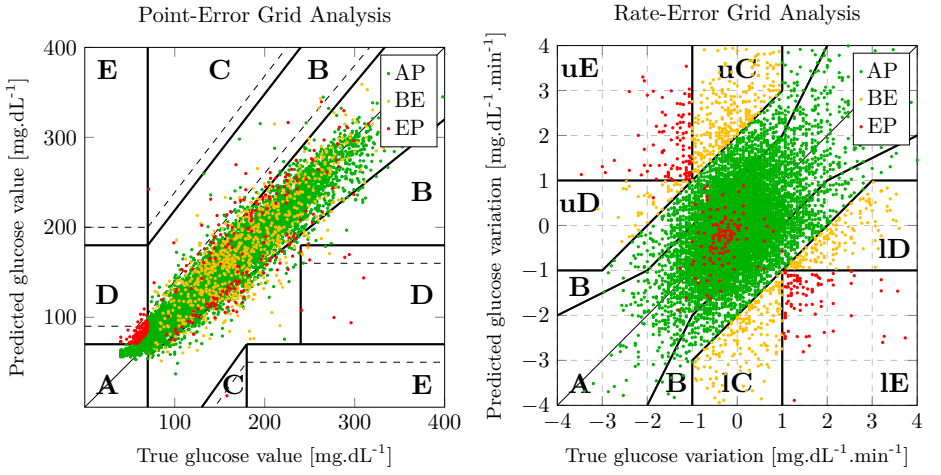


Fig. 2: Example of the CG-EGA classification with the P-EGA (left) and R-EGA (right).

Originally proposed by Kovatchev *et al.* for the evaluation of the clinical acceptability of blood glucose sensors (Kovatchev et al. (2004)), the *continuous glucose-error grid analysis* (CG-EGA) is a widely used metric to assess the clinical acceptability of glucose predictive models (Zarkogianni et al. (2015); Li et al. (2018); Georga et al. (2016); Yu et al. (2018)). It is made of the combination of two different evaluation grids: the *point-error grid analysis* (P-EGA) and the *rate-error grid analysis* (R-EGA). While the P-EGA measures the clinical accuracy of the predictions, the R-EGA measures the clinical accuracy of the predicted variations. The predicted variations are computed as the rate of change between two consecutive predictions. Both grids attribute a score from A (best) to E (worst) to a given prediction, evaluating the dangerousness of the prediction. Figure 2 gives a graphical representation of the P-EGA and the R-EGA. The scores in both grids are then combined into a final label assessing the clinical acceptability of the prediction. A prediction can either be an *accurate prediction* (AP), a *benign error* (BE), or an *erroneous prediction* (EP).

Table 1 details the reasoning behind the CG-EGA scores. First, the CG-EGA has a different behavior depending on the glycemic region (hypoglycemia, euglycemia, or hyperglycemia) the person with diabetes is in. Essentially, the glycemic region impacts the way bad R-EGA scores (C to E) are accounted. Bad R-EGA regions are split into upper and lower regions (e.g., uE and lE) to have more flexibility in the assessment of the final CG-EGA score. For instance, in the hypoglycemia region, a lE score in the R-EGA, representing a fast predicted decrease in glycemia while a fast increase is observed, can lead to a benign error (BE) if the last prediction is accurate (A in the P-EGA). In the hypoglycemia region, the CG-EGA states that it is not dangerous for the patient to predict a decrease in glycemia as it will not lead to life-threatening actions from the user. On the other hand, the absence of detection of negative variations in the uD and uE zones is extremely dangerous: the hypoglycemia is becoming much worse, which could result in consequences such as coma or even death. Overall, for a prediction to be labelled as an accurate prediction (AP), it needs good scores (A or B) in both the P-EGA and R-EGA.

In summary, compared to standard accuracy metrics such as the *root mean squared error* (RMSE), the CG-EGA also evaluates the accuracy of the predicted variations. And, most importantly, the evaluation depends on the observed glycemic region. These aspects should be taken into account if we want to add clinical constraints based on the CG-EGA into the training of the models.

2.2. Coherent Mean Squared Error

In deep learning, the models are trained by backpropagating the gradient of the loss function to the weights of the articial neural network. By modifying the objective function, it is possible to modify the predictive behavior of the model. We can find numerous loss functions in the literature, the most used being the cross-entropy for classification problems and the *mean squared error* (MSE) for regression problems. Since the task of glucose prediction is a regression task, deep models in the field use the MSE in their training. Equation 1 describes the MSE as the squared

Table 1: Classification of glucose predictions performed by the CG-EGA. Depending on the scores obtained on the P-EGA and R-EGA, a prediction is classified as an accurate prediction (AP), a benign error (BE) or erroneous prediction (EP).

		P-EGA										
		Hypoglycemia			*Euglycemia*			*Hyperglycemia*				
		A	D	E	A	B	C	A	B	C	D	E
R-EGA	A	AP	EP	EP	AP	AP	EP	AP	AP	EP	EP	EP
	B	AP	EP	EP	AP	AP	EP	AP	AP	EP	EP	EP
	uC	BE	EP	EP	BE	BE	EP	BE	BE	EP	EP	EP
	lC	BE	EP	EP	BE	BE	EP	BE	BE	EP	EP	EP
	uD	EP	EP	EP	BE	BE	EP	BE	BE	EP	EP	EP
	lD	BE	EP	EP	BE	BE	EP	EP	EP	EP	EP	EP
	uE	EP	EP	EP	EP	EP	EP	EP	EP	EP	EP	EP
	lE	BE	EP	EP	EP	EP	EP	EP	EP	EP	EP	EP

AP: Accurate Prediction; BE: Benign Error; EP: Erroneous Prediction

difference between the observed $\boldsymbol{g}$ and predicted $\hat{\boldsymbol{g}}$ glucose values, averaged over N samples. In this study, we propose modifications to the MSE loss function to improve the clinical acceptability of the predictions.

$$MSE(\boldsymbol{g}, \hat{\boldsymbol{g}}) = \frac{1}{N}\sum_{n=1}^{N}(g_n - \hat{g}_n)^2 \tag{1}$$

First, as shown by the analysis of the CG-EGA, it is essential to penalize predicted variation errors in addition to prediction errors. To do this, we can use the coherent mean squared error (cMSE) loss function, previously proposed in a work from our team (De Bois et al. (2019)). The cMSE is the MSE of the predictions weighted by the MSE of the predicted variations. Equation 2 describes the cMSE loss function with $\Delta\boldsymbol{g}$ and $\Delta\hat{\boldsymbol{g}}$ representing, respectively, the observed and predicted glucose variations. We call the weighting coefficient c the coherence factor. It represents the relative importance we give to the accuracy of the predicted variations compared to the accuracy of the predictions.

$$\begin{aligned} cMSE(\boldsymbol{g}, \hat{\boldsymbol{g}}) &= MSE(\boldsymbol{g}, \hat{\boldsymbol{g}}) + c \cdot MSE(\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}}) \\ &= \frac{1}{N}\sum_{n=1}^{N}(g_n - \hat{g}_n)^2 + c \cdot (\Delta g_n - \Delta \hat{g}_n)^2 \end{aligned} \tag{2}$$

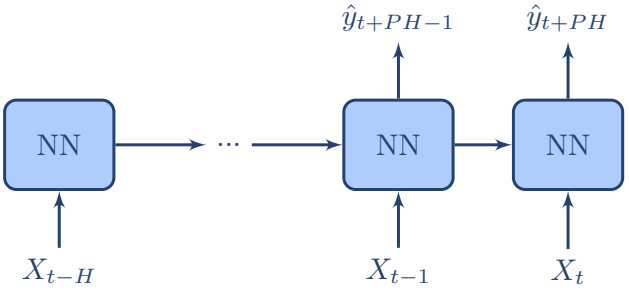


Fig. 3: General architecture of a two-output recurrent neural network that has been unrolled H times, where H is the length of the history of input data to the model. X_t are the input data to the model at time t (e.g., glucose, insulin, and carbohydrates at time t), and $\hat{y}_{t+PH}$ is the model prediction (e.g., blood glucose prediction) at $t + PH$, where PH is the prediction horizon.

To use the cMSE in the training process, we can use a recurrent neural network (e.g., LSTM) with two outputs (see Figure 3). The two outputs represent the prediction at the given prediction horizon PH and the prediction at $PH - \Delta T$, ΔT being the time interval between two predictions. For instance, with a prediction interval of 5 minutes and a prediction horizon of 30 minutes, the network outputs the predictions at the horizons 30 and 25 minutes. These two outputs enable the computation of the predicted variations, as depicted by Equation 3. The architecture of recurrent neural networks is particularly suited to this task as it naturally computes the prediction of the previous time-step (see Figure 3).

$$\Delta\hat{g}_{t+PH} = \frac{\hat{g}_{t+PH} - \hat{g}_{t+PH-\Delta T}}{\Delta T} \tag{3}$$

2.3. Coherent Mean Squared Glycemic Error

The analysis of the CG-EGA showed us that the magnitude of the glucose prediction and predicted variation errors are not fully correlated with clinical errors. Moreover, even though clinical errors are generally of high magnitude, they are quite rare in practice, thus representing only a small portion of the gradient in the updating of the network's weights during its training. Therefore, minimizing the MSE (or, equivalently, the cMSE) does not directly reduce the number of clinical errors. Indeed, most of the weights' updates are focused towards the improvement of the accuracy of predictions that already have a good clinical acceptability. In the field of multi-class classification, it is very common to weight samples from under-represented classes by artificially increasing their presence within the training set. In their work on object recognition within images, Lin *et al.* proposed to dynamically weight the learning samples according to their difficulty (a sample being considered easy when the probability of the corresponding class is very high, showing a high degree of confidence of the model in the prediction) (Lin et al. (2017)). By reducing the weights of **easy samples**, the training of the model focuses on the samples for which it has the most difficulty. Finally, Del Favero *et al.* proposed, in the context of glucose prediction, to modify the MSE to better account for the dangerous regions of the P-EGA (Del Favero et al. (2012)). In particular, they proposed that samples with observed hypoglycemia or hyperglycemia are given a higher weighting. Although their work was evaluated on autoregressive models and virtual patients, their results showed that this new loss function reduces the number of predictions in zone D and E of the P-EGA.

Taking inspiration from their work, we propose to dynamically penalize prediction errors as well as predicted variation errors. This new loss function, named *coherent mean squared glycemic error* (gcMSE), penalizes predictions differently depending on the P-EGA and R-EGA regions (see Equation 4). In Equation 4b, P_X and p_x, $X \in \{A, B, uC, lC, uD, lD, uE, lE\}$ and $x \in \{a, b, uc, lc, ud, ld, ue, le\}$, represent the P-EGA regions and their respective weights. Contrary to the original P-EGA, we have segmented the C, D and E regions in two, as it is already the case for the R-EGA. This gives us more flexibility in assigning the weights. Equivalently, in Equation 4c, R_X and r_x, $X \in \{A, B, uC, lC, uD, lD, uE, lE\}$ and $x \in \{a, b, uc, lc, ud, ld, ue, le\}$ represent the regions of the R-EGA and their respective weights.

$$gcMSE(\boldsymbol{g}, \hat{\boldsymbol{g}}) = P(\boldsymbol{g}, \hat{\boldsymbol{g}}) \cdot MSE(\boldsymbol{g}, \hat{\boldsymbol{g}}) + c \cdot R(\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}}) \cdot MSE(\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}}) \tag{4a}$$

with,

$$P(\boldsymbol{g}, \hat{\boldsymbol{g}}) = \begin{cases} p_a, & \text{if } \{\boldsymbol{g}, \hat{\boldsymbol{g}}\} \in P_A \\ p_b, & \text{if } \{\boldsymbol{g}, \hat{\boldsymbol{g}}\} \in P_B \\ p_{uc}, & \text{if } \{\boldsymbol{g}, \hat{\boldsymbol{g}}\} \in P_{uC} \\ p_{lc}, & \text{if } \{\boldsymbol{g}, \hat{\boldsymbol{g}}\} \in P_{lC} \\ p_{ud}, & \text{if } \{\boldsymbol{g}, \hat{\boldsymbol{g}}\} \in P_{uD} \\ p_{ld}, & \text{if } \{\boldsymbol{g}, \hat{\boldsymbol{g}}\} \in P_{lD} \\ p_{ue}, & \text{if } \{\boldsymbol{g}, \hat{\boldsymbol{g}}\} \in P_{uE} \\ p_{le}, & \text{if } \{\boldsymbol{g}, \hat{\boldsymbol{g}}\} \in P_{lE} \end{cases} \tag{4b}$$

and,

$$R(\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}}) = \begin{cases} r_a, & \text{if } \{\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}}\} \in R_A \\ r_b, & \text{if } \{\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}}\} \in R_B \\ r_{uc}, & \text{if } \{\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}}\} \in R_{uC} \\ r_{lc}, & \text{if } \{\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}}\} \in R_{lC} \\ r_{ud}, & \text{if } \{\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}}\} \in R_{uD} \\ r_{ld}, & \text{if } \{\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}}\} \in R_{lD} \\ r_{ue}, & \text{if } \{\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}}\} \in R_{uE} \\ r_{le}, & \text{if } \{\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}}\} \in R_{lE} \end{cases} \tag{4c}$$

Using the gcMSE instead of the standard MSE introduces 14 new hyperparameters to be optimized: the coherence factor c, and the weights associated with the P-EGA and R-EGA regions. This task being particularly laborious, we propose simplifications reducing the number of hyperparameters:

- First, it is not interesting to improve the accuracy of the predicted variations in zones A and B. Indeed, all predictions belonging to these zones are clinically sufficiently accurate. Thus, we can set $r_a = r_b = 0$.
- From the perspective of the possible maximization of the AP rate, BE and EP predictions can be seen as equally important. This allows us to set most of the C, D and E zones to the same value. Moreover, the coherence factor c alone allows us to weight the compromise we want between the accuracy of the predictions and the accuracy of predicted variations. Thus, we can set all these weights to 1.
- Only the hypoglycemic P-EGA regions D and E (P_{uD} and P_{uE}) require a special treatment in order to increase the importance of samples in the hypoglycemic region. We denote the weight associated to these areas by p_{hypo}.

Equation 5 summarizes the design simplifications, allowing the gcMSE cost function to have only 3 hyperparameters: p_{ab}, p_{hypo}, and c. The choice of these hyperparameters depends on both the learning objective and the experimental conditions. The coherence factor c must be chosen depending on the importance of the loss function $MSE(\boldsymbol{\Delta g}, \boldsymbol{\Delta \hat{g}})$ compared to the $MSE(\boldsymbol{g}, \hat{\boldsymbol{g}})$. The choice of the coefficient p_{hypo} must be made

according to the size of the datasets. When few hypoglycemic samples are available, it is possible to give a value of $p_{hypo} > 1$. As for p_{ab}, it represents the accuracy constraint we give during the training of the model. The lower its value, the more its training focuses on improving its clinical acceptability at the expense of its accuracy.

$$P(\boldsymbol{g}, \hat{\boldsymbol{g}}) = \begin{cases} p_{ab}, & \text{if } \{\boldsymbol{g}, \hat{\boldsymbol{g}}\} \in \{P_A, P_B\} \\ p_{hypo}, & \text{if } \{\boldsymbol{g}, \hat{\boldsymbol{g}}\} \in \{P_{uD}, P_{uE}\} \\ 1, & \text{else} \end{cases} \tag{5a}$$

and,

$$R(\Delta\boldsymbol{g}, \Delta\hat{\boldsymbol{g}}) = \begin{cases} 0, & \text{if } \{\Delta\boldsymbol{g}, \Delta\hat{\boldsymbol{g}}\} \in \{R_A, R_B\} \\ 1, & \text{else} \end{cases} \tag{5b}$$

2.4. Progressive Improvement of the Clinical Acceptability

In order to be able to use the gcMSE loss function, we need to formulate the general learning objective, and in particular the relative importance of improving the clinical acceptability. Indeed, as shown in the work of Del Favero et al. (2012), an improvement in the clinical acceptability is often matched by a deterioration in the statistical accuracy . Our previous work also showed that when too little constraints are set on the accuracy of the model, the predictions end up being of no use for the user De Bois et al. (2019).

The presence of two objectives competing against each other makes this problem a multi-objective optimization (MOO) problem. In the MOO field, there is often no optimal solution (a solution that is the best one for all the objectives), but a set of solutions that are said to be *Pareto-optimal* (Marler & Arora (2004)). We can define a solution that is Pareto-optimal as a solution for which there exists no other solution that is simultaneously better for all the objectives. The solving of a MOO problem is generally a two-step process, where the Pareto-optimal solutions are first identified and one of them is then selected given selection criteria.

These two steps are challenging in our application context. First, while selection criteria could be formulated as clinical acceptability requirements, no official standards have been set by the health authorities for glucose predictive models yet. Second, finding a single solution is computationally expensive as it involves the full training of a neural network. It makes the identification of the set of Pareto-optimal solutions through a standard grid search approach not practical. While other approaches based on genetic programming, often, used in the MOO field (e.g., NSGA-II, Deb et al. (2000)) converge faster, they present the same issue.

To address these challenges, we propose the *progressive improvement of clinical acceptability* (PICA) algorithm that leverages our understanding of the search space. First, we define a hypothetical selection criterion as a minimum threshold in AP or/and a maximum threshold in EP following the CG-EGA (e.g., minimum 95% of predictions being labeled as AP by the CG-EGA). Our optimization problem can then be reformulated as the maximization of the accuracy of the predictions while meeting the set clinical criteria. To reduce the number of solutions that need to be computed, we start from a Pareto-optimal solution maximizing the accuracy of the model without considering the clinical acceptability of the predictions. Other solutions are then computed by progressively relaxing the accuracy constraints, gradually shifting the emphasis on the clinical acceptability. By doing so, we aim at navigating the Pareto front, only computing solutions that are worth considering for our problem. Once the clinical criterion is met, we stop the search of other solutions and select the last one as the solution that maximizes the accuracy while satisfying the clinical constraints.

Algorithm 1: Progressive Improvement of the Clinical Acceptability (PICA)

Data: clinical criteria C, model M, update coefficient α, smoothing coefficient β
Result: Model maximizing the accuracy while respecting the clinical criteria C or -1

$i \leftarrow 0$
$M_0 \leftarrow$ train(MSE)
$\boldsymbol{g}_0, \hat{\boldsymbol{g}}_0 \leftarrow$ predict(M_0)
$\hat{\boldsymbol{g}}_0^* \leftarrow$ smooth($\hat{\boldsymbol{g}}_0, \beta$)
while $C(\boldsymbol{g}_i, \hat{\boldsymbol{g}}_i^*) = False\ and\ MASE(\boldsymbol{g}_i, \hat{\boldsymbol{g}}_i^*) < 1$ **do**
  $i \leftarrow i + 1$
  $\text{gcMSE}_i \leftarrow$ gcMSE with $p_{ab} \leftarrow \alpha^{i-1}$
  $M_i \leftarrow$ finetune(M_0, gcMSE_i)
  $\boldsymbol{g}_i, \hat{\boldsymbol{g}}_i \leftarrow$ predict(M_i)
  $\hat{\boldsymbol{g}}_i^* \leftarrow$ smooth($\hat{\boldsymbol{g}}_i, \beta$)
if $MASE(\boldsymbol{g}_i, \hat{\boldsymbol{g}}_i^*) < 1$ **then**
  return M_i
else
  return -1

Algorithm 1 gives the technical details of the steps made by PICA algorithm. The updating law of the weights p_{ab}, representing the constraints in the statistical accuracy, is to be chosen according to the experimental conditions. In this study, we use the law defined by the Equation 6 (with $\alpha \in [0, 1]$ being the speed of the relaxation of the accuracy constraints). As for the MASE metric (*mean absolute scaled error*, proposed by

Hyndman & Koehler (2006), see Equation 7), it is used as a stopping criterion when the chosen clinical criteria are not achievable. The algorithm stops when the MASE exceeds 1, meaning that a naïve prediction (a prediction that is equal to the last known observation) is more accurate than the predictions made by the model in average. Finally, we smooth the predictions by using an exponential smoothing technique. It is used to attenuate the important fluctuations of the predictions in the first steps of the algorithm. By being small, it allows a significant gain in the clinical acceptability, in return for a minimal loss of accuracy. For more details on the exponential smoothing of the predictions, please refer to the post-processing steps in Section 3.3.

$$p_{ab} = \alpha^{i-1} \tag{6}$$

$$MASE(\boldsymbol{g}, \hat{\boldsymbol{g}}, PH) = \frac{\frac{1}{N} \cdot \sum_{n=1}^{N} |g_n - \hat{g}_n|}{\frac{1}{N-PH} \cdot \sum_{n=PH}^{N} |g_n - g_{n-PH}|} \tag{7}$$

Algorithm 2: Standard Grid Search

Data: clinical criteria C, model M, grid step size α, smoothing coefficient β, maximal number of iteration N
Result: Model maximizing the accuracy while respecting the clinical criteria C or -1

Function `train_and_test(`*loss*`):`
  $m \leftarrow$ train($loss$)
  $\boldsymbol{g}, \hat{\boldsymbol{g}} \leftarrow$ predict(m)
  $\hat{\boldsymbol{g}}^* \leftarrow$ smooth($\hat{\boldsymbol{g}}, \beta$)
  return m, $\boldsymbol{g}$, $\hat{\boldsymbol{g}}^*$
$grid \leftarrow [1, \alpha^{-1}, ..., \alpha^{N-1}]$
M_0, $\boldsymbol{g}_0$, $\hat{\boldsymbol{g}}_0^* \leftarrow$ train_and_test(MSE)
for $i \leftarrow 1$ *to* N **do**
  $\text{gcMSE}_i \leftarrow$ gcMSE with $p_{ab} \leftarrow grid[i]$
  M_i, $\boldsymbol{g}_i$, $\hat{\boldsymbol{g}}_i^* \leftarrow$ train_and_test(gcMSE_i)
$candidates \leftarrow [M_0, ..., M_N]$ if $C(\boldsymbol{g}_i, \hat{\boldsymbol{g}}_i^*) = True$
if *candidates is not empty* **then**
  return $\underset{MASE}{\text{argmin}}\, candidates$
else
  return -1

To better understand what the benefits of using the PICA algorithm are, we can compare it to a standard grid search of the hyperparameter p_{ab} which is described by Algorithm 2. To optimize by grid search, we first need to define a search space. Here we characterize the search space by the step size α in a logarithmic scale and by the number of elements inside the grid. With the same value of α, loss functions evaluated by the PICA algorithm are guaranteed to be also evaluated by the grid search. Instead of stopping the search when the best p_{ab} coefficient is found, a standard grid search waits to compute all the different solutions before selecting the best one. Among these solutions, the solutions that satisfy the clinical constraints but have a worse accuracy than the best solution are not computed by the PICA algorithm, making it faster. As a consequence, the best solutions selected by both algorithms are identical. Moreover, each iteration (except the first one) is in itself faster using the PICA algorithm as we are finetuning the first model maximizing the accuracy, instead of fully training a new one from scratch. Finetuning a model requires much less epochs than a full training, and thus allows the algorithm to run even faster.

3. Experimental Methodology

In this section, we present the whole methodology that has been followed for the evaluation of the proposed loss functions and the PICA algorithm. First, we present the experimental datasets and their preprocessing. Then, we provide details about the post-processing of the predictions and the evaluation of the models. Finally, we describe the different models with their implementation.

We have made the code implementation of the whole study available in a GitHub repository (De Bois (2020)).

3.1. Experimental Data

In this study, we used two datasets made of several people with diabetes: the IDIAB dataset and the OhioT1DM dataset. While the IDIAB has been collected by us between 2018 and 2019 after the approval by the French ethical committee (ID RCB 2018-A00312-53), the OhioT1DM dataset has recently been released by Marling & Bunescu (2018).

3.1.1. IDIAB Dataset (I)

The IDIAB dataset is made of 6 individuals with type-2 diabetes (5F/1M, age 56.5 ± 9.14 years old, body mass index of 33.52 ± 4.17 kg/m^2). The patients had been monitored for 31.17 ± 1.86 days in free-living conditions. We collected glucose values (in mg/dL) by using FreeStyle Libre continuous glucose monitoring devices (Abbott Diabetes Care). As for carbohydrate (CHO) intakes (g) and insulin infusion values (unit), they have been manually recorded with the mySugr coaching application for diabetes.

3.1.2. OhioT1DM Dataset (O)

The OhioT1DM dataset is made of data coming from 6 people with type-1 diabetes (4F/2M, age between 40 and 60 years old, body mass index not disclosed) that had been monitored for 8 weeks in free-living conditions. For more information concerning the experimental system, please refer to Marling & Bunescu (2018). We restrict ourselves to the glucose values, the insulin infusions, and the CHO intakes to remain consistent with IDIAB data.

3.2. Preprocessing

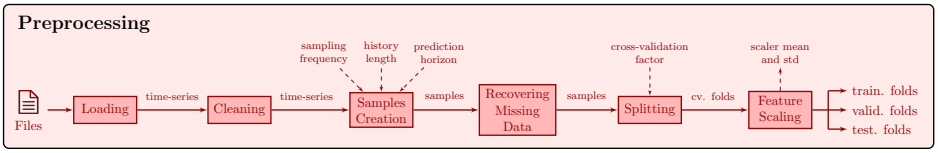


Fig. 4: Preprocessing of the data.

The preprocessing stage aims at preparing the data for their use in the training and the evaluation of the models. It is made of several steps depicted by Figure 4 and described in the following paragraphs.

3.2.1. Cleaning

The glucose time-series from the IDIAB dataset is comprised of several erroneous values. These values are characterized by peaks lasting only one sample (see Figure 5). We decided to remove these samples from the data as keeping them would be hurtful for the training as well as for the evaluation of the models. Instead of removing them by hand, we used an automated methodology proposed in our previous work (De Bois et al. (2019)). A sample is flagged as erroneous if the surrounding rates of change are incoherent with the typical distribution of rates of change, and if they are of opposite signs.

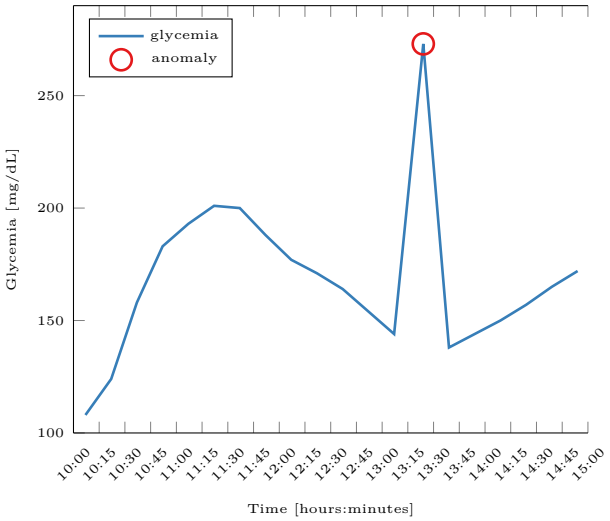


Fig. 5: Glycemia of one patient from the IDIAB dataset, for which the value recorded at 13h24 is an anomaly as it is incoherent with the overall signal.

3.2.2. Samples Creation

The two datasets have been resampled to a sample every 5 minutes which is the sampling frequency of the OhioT1DM glucose signal. While we took the mean of the glucose signals, the CHO and insulin values have been accumulated.

The input samples have been obtained by using a sliding window of length H of 3 hours (36 samples) on the three signals. The prediction objective is, for each sample, the glucose value 30 minutes (6 samples) in the future (prediction horizon, PH, of 30 minutes).

3.2.3. Recovering Missing Data

Both datasets contain numerous missing values coming either from sensor or human errors. Moreover, contrary to the OhioT1DM dataset, the upsampling of the IDIAB glucose signal (from 15 minutes to 5 minutes) has also introduced a lot of missing values. We can artificially recover some of them by following this strategy for every sample:

1. linearly interpolate the glucose history when the missing value is surrounded by two known glucose values;
2. extrapolate linearly in the opposite case, usually when the missing glucose value is the most recent data;
3. discard samples when the ground truth y_{t+PH} is not known to prevent training and testing on artificial data.

3.2.4. Splitting

The datasets are split into training, validation, and testing sets. While the testing set is used for the final evaluation of the models, the validation is used as a prior evaluation for the optimization of the models' hyperparameters.

The testing set is made of the last 10 days for the OhioT1DM dataset and of the last 5 days for the IDIAB dataset, the latter being around two times smaller. The remaining days have been split into training and validation sets following an 80%/20% distribution with 5 permutations.

3.2.5. Feature Scaling

Finally, the samples have been standardized (zero mean and unit variance) with respect to their training set.

3.3. Post-processing and Evaluation

The evaluation of the predictive models is done following the steps described by Figure 6. In this study, we focus on models that are personalized to the patient and that predict future glucose values with a 30-minute prediction horizon. Before evaluating the predictions, we follow two mandatory post-processing steps. We rescale and reshape the predictions to their original scale and shape (see the preprocessing step). Finally, an optional step is the smoothing of the predictions of the models, as it is done in the PICA algorithm. In the experimental results section, we will report the performance of the models with both smoothed and raw predictions.

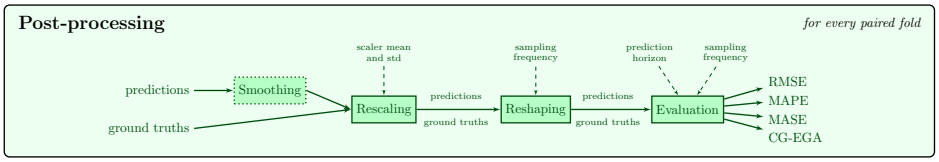


Fig. 6: Post-processing and evaluation of the predictions.

3.3.1. Exponential Smoothing

The PICA algorithm involves the smoothing of the predictions at each iteration. The goal of the smoothing is to reduce excessive fluctuations in the predicted glucose signal. These oscillations are not representative of actual glucose variations and are therefore dangerous for the patient.

We chose the exponential smoothing technique rather than the moving average technique because it gives more weight to recent predictions. Exponential smoothing can be defined as recursive, with each value of the smoothed signal being equal to a weighting between the value of the original signal and the previous value of the smoothed signal (see Equation 8, where $\hat{g}_t^*$ represents the smoothed value of the glucose prediction $\hat{g}_t$ and β the smoothing coefficient) (Brown (2004)).

$$\hat{g}_t^* = \begin{cases} \hat{g}_0, & \text{if } t = 0 \\ \beta \cdot \hat{g}_t + (1-\beta) \cdot \hat{g}_{t-1}^*, & \text{else} \end{cases} \tag{8}$$

The higher β is, the stronger is the weight given to the original signal, and the less smooth the outputted signal is. The choice of the β smoothing coefficient in $[0, 1]$ must be made carefully. Indeed, a too aggressive smoothing will result in a temporal shift of the signal. In the context of glucose prediction, this will greatly reduce the accuracy of the model, and therefore its usefulness for the patient.

To our knowledge, although common in signal processing (e.g., power consumption prediction - Taylor & McSharry (2007)), no post-processing smoothing has been done in the literature of glucose prediction. We can nevertheless note the occasional use of low-pass filters (which act similarly to the exponential smoothing technique) on the input signal (Sparacino et al. (2007); Pérez-Gandía et al. (2010)).

3.4. Metrics

To evaluate the models, we use four different metrics: the *root mean-squared error* (RMSE), the *mean absolute percentage error* (MAPE), the *mean absolute scaled error* (MASE) and the CG-EGA. For each metric, the performance is averaged over the 5 test subsets of each patient linked to a 5-fold cross-validation on the training/validation permutations. They are then also averaged on all the patients from the same dataset. The RMSE, MAPE and MASE metrics give a complementary measure of the accuracy of the prediction. While the RMSE is closely related to the prediction scale, the MAPE is scale independent and is expressed in percentage. As for the MASE, it measures the average usefulness of the predictions compared to naïve predictions (predictions equal to the last known observations). The MASE is computed following Equation 7, presented in the previous section. On the other hand, the CG-EGA measures the clinical acceptability of the prediction by analyzing the clinical accuracy as well as the coherence between successive predictions. It classifies a prediction either as an *accurate prediction* (AP), a *benign error* (BE), or an *erroneous prediction* (EP). A high AP rate and a low EP rate are necessary for a model to be clinically acceptable. The rates can be either averaged over all the test samples, or over the samples within a specific glycemic region (i.e., hypoglycemia, euglycemia and hyperglycemia).

3.5. Glucose Predictive Models

The aim of the study is to improve the clinical acceptability of deep models. To this end, we have first proposed a new loss function cMSE which penalizes the model during its training not only on prediction errors but also on predicted variation errors. We have then proposed the gcMSE, which is the cMSE customized to the task of glucose prediction. In particular, it introduces weighting coefficients based on the CG-EGA to enhance the clinical acceptability of the model. Finally, we proposed the PICA algorithm that progressively improves the clinical acceptability of the models through the use of the gcMSE function. The models that we present here aim at evaluating these different proposals.

As reference models, we use the *support vector regression* model (SVR) and *long short-term memory* recurrent neural network (LSTM) from the GLYFE benchmark study (De Bois (2019)). Since the preprocessing steps are identical in this study and the present one, the results are fully comparable. The SVR and LSTM models represent, respectively, the best model and the best deep model in this benchmark.

- The **SVR** model uses the *radial basis function* (RBF) kernel. All its hyperparameters have been individually optimized for every patient. The kernel coefficient, the penalty, and the wideness of the no-penalty tube have been grid searched in the ranges $[10^{-4}, 10^{-2}]$, $[10^{0}, 10^{3}]$, and $[10^{-3}, 10^{0}]$ respectively.

- The **LSTM** model has 2 hidden layers made of 256 long short-term memory units. It is trained with the Adam optimizer (mini-batches of 50 samples) and the MSE loss function. The learning rate has been grid-searched within $[10^{-4}, 10^{-3}]$. Finally, the early stopping methodology (after 50 epochs of non-improvement on the validation set) and a L2 penalty (10^{-4}) have been used for regularization purposes.

First, to analyze the potential improvement of the clinical acceptability through the cMSE and gcMSE cost functions, we evaluate the pcLSTM and gpcLSTM models respectively. These two models are based on a two-output LSTM architecture, which, apart from the presence of the two outputs, is identical to the LSTM model of the GLYFE benchmark study. They are respectively trained to minimize the cMSE and gcMSE loss functions with a coherence factor c set to 8 for the IDIAB dataset and 2 for the OhioT1DM dataset. The difference in the coherence factor between the two sets is explained by a MSE of the predicted variations being approximately 4 times higher for the OhioT1DM dataset. As for the coefficients p_{ab} and p_{hypo} of the gcMSE, we have set them to 1 and 10 respectively. These coefficients are identical to those from the first iteration of the PICA algorithm. In addition, we propose to evaluate an additional variant of the gcMSE whose coefficient p_{ab} is set to 0. This model, denoted gpcLSTM$_{\text{CA}}$, is a model that aims at maximizing the clinical acceptability, without taking into account the accuracy of the model beyond clinical acceptability needs.

The PICA algorithm uses the exponential smoothing technique to stabilize successive predictions. In order to fully evaluate the impact of the loss functions and the PICA algorithm, we use the exponential smoothing technique on all the models presented in this study. The smoothed variant of each model is represented by a superscript asterisk (e.g., LSTM*, pcLSTM*, gpcLSTM$^*_{\text{CA}}$). All these models use a smoothing coefficient of 0.85, as it degrades only slightly the accuracy of the predicted signal.

The PICA algorithm makes a compromise between the gpcLSTM* and gpcLSTM$^*_{\text{CA}}$ models. The emphasis on clinical acceptability of this compromise is progressive over the iterations of the algorithm. However, the accuracy constraint, through the coefficient p_{ab} is never equal to 0 (model gpcLSTM$^*_{\text{CA}}$), because such a model has an accuracy far too low to be useful for people with diabetes. This is why the PICA algorithm stops when the MASE exceeds the value of 1 on the validation set. We represent by the model gpcLSTM$^*_{\text{PICA}}$ the results obtained when the PICA algorithm stops. These results represent the upper bounds of clinical acceptability while maintaining a useful accuracy. In the PICA algorithm, we use the update law of the coefficient p_{ab} presented by Equation 6. It involves the coefficient α, the rate at which the constraint in accuracy is relaxed, which has been set to 0.9 in this study. A higher coefficient gives better control over the final trade-off, in return for a slower execution time (more iterations before convergence). The PICA algorithm uses the exponential smoothing technique on the model's predictions to increase the stability of the predicted signal. The smoothing coefficient β, as for all the smoothed variants of the other models, has been fixed at 0.85.

4. Results

In this section we present the experimental results of this study. These results are represented in the form of two tables: Table 2 and 3. While Table 2 describes the general results of the different models in terms of RMSE, MAPE, MASE and general CG-EGA, Table 3 gives a more detailed description, by region, of the CG-EGA.

Within our two reference models, SVR and LSTM, the SVR model is the model with the best clinical acceptability (general or regional CG-EGA) for comparable accuracy. In particular, the SVR model has one of the best clinical acceptability in the hypoglycemia region (69.39% and 49.71% AP for the IDIAB and OhioT1DM datasets respectively). The exponential smoothing improves the clinical acceptability of the SVR model

Table 2: Mean (with standard deviation) of statistical accuracy (RMSE, MAPE, and MASE) and general clinical acceptability (CG-EGA) for a prediction horizon of 30 minutes and for the IDIAB and OhioT1DM datasets.

Model	RMSE	MAPE	MASE	CG-EGA (general)		
				AP	BE	EP
IDIAB Dataset						
SVR	20.32 (6.02)	**8.66 (0.44)**	0.85 (0.15)	92.69 (2.81)	5.34 (2.06)	1.97 (1.23)
LSTM	**19.85 (6.00)**	9.04 (1.11)	**0.85 (0.10)**	92.20 (2.99)	5.05 (1.71)	2.76 (1.82)
SVR*	20.67 (6.20)	8.86 (0.44)	0.88 (0.15)	93.62 (2.57)	4.47 (1.69)	1.92 (1.35)
LSTM*	20.27 (6.30)	9.25 (1.21)	0.87 (0.09)	93.16 (3.13)	4.16 (1.75)	2.68 (2.00)
pcLSTM	21.89 (5.68)	10.28 (1.34)	0.96 (0.11)	94.04 (3.26)	3.20 (1.66)	2.76 (2.07)
pcLSTM*	22.63 (6.04)	10.64 (1.40)	1.00 (0.11)	94.24 (3.35)	**2.94 (1.73)**	2.82 (2.07)
gpcLSTM	21.21 (5.64)	9.35 (0.92)	0.91 (0.13)	94.03 (2.66)	3.91 (1.48)	2.06 (1.54)
gpcLSTM*	21.86 (5.94)	9.66 (0.95)	0.94 (0.13)	94.53 (2.84)	3.38 (1.55)	2.08 (1.57)
gpcLSTM$_{CA}$	40.68 (11.20)	18.14 (5.55)	1.91 (0.55)	95.34 (2.76)	3.29 (2.56)	**1.37 (0.91)**
gpcLSTM*$_{CA}$	41.15 (11.18)	18.36 (5.47)	1.93 (0.54)	**95.35 (2.87)**	3.20 (2.61)	1.45 (0.92)
gpcLSTM*$_{PICA}$	24.03 (7.15)	10.43 (1.18)	1.03 (0.09)	95.00 (2.74)	3.38 (1.99)	1.61 (1.22)
OhioT1DM Dataset						
SVR	**20.15 (2.33)**	**9.12 (2.11)**	**0.85 (0.02)**	83.35 (3.91)	12.38 (2.83)	4.28 (1.83)
LSTM	20.46 (2.08)	9.24 (2.10)	0.86 (0.02)	80.03 (4.17)	14.83 (2.88)	5.14 (2.11)
SVR*	20.17 (2.30)	9.18 (2.12)	**0.85 (0.02)**	85.00 (4.05)	10.97 (2.72)	4.03 (1.90)
LSTM*	20.43 (2.03)	9.26 (2.10)	0.86 (0.02)	82.14 (3.94)	13.06 (2.51)	4.81 (2.04)
pcLSTM	21.53 (2.23)	10.07 (2.32)	0.93 (0.03)	87.45 (3.76)	8.46 (2.05)	4.09 (2.14)
pcLSTM*	21.71 (2.22)	10.19 (2.35)	0.94 (0.03)	87.89 (3.61)	8.15 (1.94)	3.96 (2.12)
gpcLSTM	21.66 (2.69)	9.65 (2.14)	0.92 (0.03)	86.97 (3.63)	9.50 (2.52)	3.53 (1.48)
gpcLSTM*	21.82 (2.69)	9.76 (2.16)	0.93 (0.03)	87.59 (3.45)	9.01 (2.31)	3.41 (1.49)
gpcLSTM$_{CA}$	47.70 (6.31)	22.43 (2.76)	2.37 (0.53)	90.46 (2.85)	7.16 (1.66)	**2.37 (1.28)**
gpcLSTM*$_{CA}$	47.82 (6.27)	22.47 (2.76)	2.37 (0.53)	**90.51 (2.88)**	**7.12 (1.64)**	2.37 (1.30)
gpcLSTM*$_{PICA}$	23.50 (2.49)	10.46 (2.09)	1.01 (0.03)	88.72 (3.59)	8.20 (2.23)	3.08 (1.64)

Table 3: Mean (with standard deviation) of per-region clinical acceptability (CG-EGA) for a prediction horizon of 30 minutes and for the IDIAB and OhioT1DM datasets.

Model	CG-EGA (per region)								
	Hypoglycemia			*Euglycemia*			*Hyperglycemia*		
	AP	BE	EP	AP	BE	EP	AP	BE	EP
IDIAB Dataset									
SVR	69.39 (33.51)	0.35 (0.70)	30.27 (33.54)	95.17 (2.01)	4.33 (1.83)	0.50 (0.47)	89.51 (6.09)	7.43 (3.86)	3.06 (2.53)
LSTM	40.94 (30.73)	**0.00 (0.00)**	59.06 (30.73)	95.78 (1.48)	3.83 (1.55)	0.39 (0.38)	89.55 (5.60)	7.35 (3.21)	3.10 (2.45)
SVR*	66.37 (31.47)	0.17 (0.35)	33.45 (31.51)	96.13 (1.81)	3.49 (1.66)	0.39 (0.36)	90.61 (5.67)	6.60 (3.23)	2.79 (2.79)
LSTM*	37.99 (31.22)	**0.00 (0.00)**	62.01 (31.22)	96.71 (1.35)	2.95 (1.46)	0.33 (0.38)	91.02 (6.04)	6.18 (3.67)	2.80 (2.58)
pcLSTM	34.59 (29.27)	**0.00 (0.00)**	65.41 (29.27)	97.58 (0.90)	2.13 (0.82)	0.29 (0.20)	92.60 (5.81)	4.94 (3.18)	2.46 (2.80)
pcLSTM*	32.20 (27.83)	**0.00 (0.00)**	67.80 (27.83)	**97.96 (0.98)**	**1.81 (0.91)**	**0.23 (0.11)**	92.81 (6.25)	**4.68 (3.48)**	2.51 (2.85)
gpcLSTM	64.79 (24.95)	**0.00 (0.00)**	35.21 (24.95)	96.60 (1.11)	3.03 (0.99)	0.37 (0.26)	92.06 (5.12)	5.42 (2.83)	2.51 (2.46)
gpcLSTM*	61.87 (25.17)	**0.00 (0.00)**	38.13 (25.17)	97.23 (1.17)	2.46 (1.02)	0.31 (0.22)	92.65 (5.60)	4.85 (3.09)	2.50 (2.68)
gpcLSTM$_{CA}$	**87.95 (9.58)**	1.71 (3.43)	**10.34 (8.15)**	97.37 (1.36)	2.12 (1.03)	0.51 (0.40)	92.17 (4.46)	5.11 (4.52)	2.72 (2.39)
gpcLSTM*$_{CA}$	87.77 (9.53)	1.71 (3.43)	10.51 (8.13)	97.50 (1.32)	1.97 (0.97)	0.52 (0.44)	92.10 (4.69)	5.03 (4.70)	2.87 (2.33)
gpcLSTM*$_{PICA}$	68.49 (27.85)	0.57 (1.14)	30.94 (28.22)	97.35 (1.18)	2.32 (1.08)	0.33 (0.15)	**93.16 (4.84)**	5.08 (3.53)	**1.76 (1.49)**
OhioT1DM Dataset									
SVR	49.71 (18.75)	5.62 (4.02)	44.67 (18.70)	86.35 (4.24)	10.71 (3.26)	2.94 (1.23)	80.85 (3.24)	14.77 (3.01)	4.37 (1.84)
LSTM	38.37 (23.17)	3.97 (3.72)	57.67 (24.23)	83.78 (5.33)	12.70 (4.06)	3.52 (1.47)	76.86 (3.70)	17.87 (2.73)	5.27 (2.21)
SVR*	46.95 (21.11)	5.97 (4.05)	47.09 (21.65)	87.83 (4.22)	9.46 (3.21)	2.71 (1.22)	82.81 (3.43)	13.12 (2.98)	4.07 (2.00)
LSTM*	37.34 (23.50)	4.11 (4.15)	58.56 (24.17)	85.71 (4.83)	11.10 (3.58)	3.19 (1.37)	79.27 (3.55)	15.85 (2.40)	4.88 (2.24)
pcLSTM	25.28 (19.11)	3.64 (3.73)	71.08 (19.35)	90.79 (3.43)	6.93 (2.53)	2.28 (1.01)	85.78 (3.64)	10.83 (2.55)	3.40 (2.03)
pcLSTM*	23.82 (18.23)	3.72 (3.48)	72.45 (18.55)	91.20 (3.17)	6.67 (2.35)	2.13 (0.96)	86.33 (3.54)	10.44 (2.50)	3.23 (1.96)
gpcLSTM	53.66 (22.59)	4.34 (3.83)	42.00 (22.86)	89.39 (3.91)	7.99 (2.90)	2.63 (1.12)	84.61 (3.84)	11.79 (3.20)	3.61 (2.01)
gpcLSTM*	52.37 (22.06)	4.32 (3.15)	43.30 (22.42)	90.02 (3.69)	7.47 (2.77)	2.52 (1.04)	85.27 (3.69)	11.31 (2.95)	3.42 (2.02)
gpcLSTM$_{CA}$	**91.17 (8.50)**	1.26 (2.08)	**7.57 (8.01)**	91.61 (2.03)	6.62 (1.39)	1.77 (0.74)	**87.97 (5.00)**	**8.67 (2.64)**	**3.36 (2.63)**
gpcLSTM*$_{CA}$	91.02 (8.49)	**1.21 (1.97)**	7.77 (8.00)	**91.71 (2.02)**	**6.55 (1.34)**	**1.75 (0.77)**	87.95 (5.05)	8.69 (2.69)	**3.36 (2.62)**
gpcLSTM*$_{PICA}$	61.30 (20.12)	2.92 (2.38)	35.79 (20.23)	90.84 (3.57)	7.04 (2.57)	2.11 (1.07)	86.48 (3.95)	10.07 (2.66)	3.45 (2.31)

(SVR* model) by -12.79%[1] of AP rate for an increase of +0.90% in RMSE (decrease in accuracy). The LSTM* model is subject to similar changes

[1]Here we represent the decrease, in %, of what is metrically improvable. For the AP, which has a maximum of 100%, the ratio of change is calculated as $(100 - AP_1)/(100 - AP_2)$.

with -11.44% AP and +0.98% RMSE. Table 3 shows that these improvements in clinical acceptability occur in the euglycemia or hyperglycemia regions, and not in the hypoglycemia region (small decrease in AP).

The pcLSTM model and its smoothed variant pcLSTM*, using the cMSE loss function as well as the two-output architecture of the LSTM network, are showed to improve the clinical acceptability while deteriorating the accuracy. In particular, the pcLSTM* model compared to the LSTM* model has -24.18% AP, and +8.95% RMSE. The improvement in clinical acceptability is greater for the OhioT1DM dataset (-32.19% AP) than for the IDIAB dataset (-16.16% AP). For a comparable decrease in accuracy, the OhioT1DM dataset benefits more from the cMSE loss function than the IDIAB set. Moreover, the pcLSTM* model has among the best clinical acceptability scores in the euglycemia and hyperglycemia regions. However, in comparison with the LSTM or LSTM* models, the clinical acceptability in the hypoglycemia region is deteriorated, especially for the OhioT1DM dataset.

The gpcLSTM and gpcLSTM* models, using the gcMSE loss function, cMSE customized to blood glucose prediction, show a degradation of the RMSE and an improvement of the AP rate similar to the pcLSTM and pcLSTM* models. However, the gpcLSTM and gpcLSTM* models have a lower EP rate (-19.53% and -20.07% respectively), suggesting an improved clinical acceptability. Table 3 shows that this improvement is mainly in the hypoglycemia region with much lower EP rates.

The models gpcLSTM$_{\text{CA}}$ and gpcLSTM$^*_{\text{CA}}$ use a gcMSE function with the coefficient p_{ab} of 0. Thus, these models focus only on improving the clinical acceptability. By not seeking to improve the accuracy of predictions beyond the required clinical accuracy (P-EGA Zone B), these models have a very poor RMSE, MAPE and MASE. Nevertheless, they have the best clinical acceptability, with the highest AP and the lowest EP rates. The improvement is particularly important in the hypoglycemia region, as can be seen in Table 3.

The gpcLSTM$^*_{\text{PICA}}$ model represents the last iteration of the PICA algorithm with a MASE on the validation set of less than 1. This model is intended to maximize the clinical acceptability, while having a reasonable accuracy (MASE less than 1). Compared to the gpcLSTM$^*_{\text{CA}}$ model, it has a slightly lower clinical acceptability (but better than all other models, thanks in particular to its low EP rate).

5. Discussion

The results show us that the exponential smoothing technique reduces the benign error (BE) rate in favor of a better AP rate, by reducing the amplitude of the variations between successive predictions. This improvement is valid for most of the models and has for counterpart a rather small decrease in the statistical accuracy of the model. Thus, exponential smoothing, used softly (coefficient β of 0.85) is an efficient method to improve the stability of the prediction signal, making it safer for the people with diabetes. However, it remains useless in the hypoglycemia range where the majority of clinical prediction errors are due to poor accuracy.

The benefits from using the cMSE loss function on glucose predictions are similar: successive glucose predictions are more consistent with each other, resulting in a large reduction in the BE rate. The effects are greater for the OhioT1DM dataset, which sees its EP rate decrease at the same time. It can be explained by a higher noise in the predicted glucose signal of the OhioT1DM dataset, noise that comes from the initial glucose signal. With its lower sampling frequency, the IDIAB glucose signal manages to be less noisy in comparison. The cMSE allows successive predictions to be made with a rate of change that better reflects the actual rate of change and thus improves its clinical acceptability. However, like exponential smoothing, improvements in clinical acceptability are not generalized to all glycemic regions. In particular, the hypoglycemic region appears to suffer from the use of cMSE with an increase in its EP rate, especially for the OhioT1DM dataset.

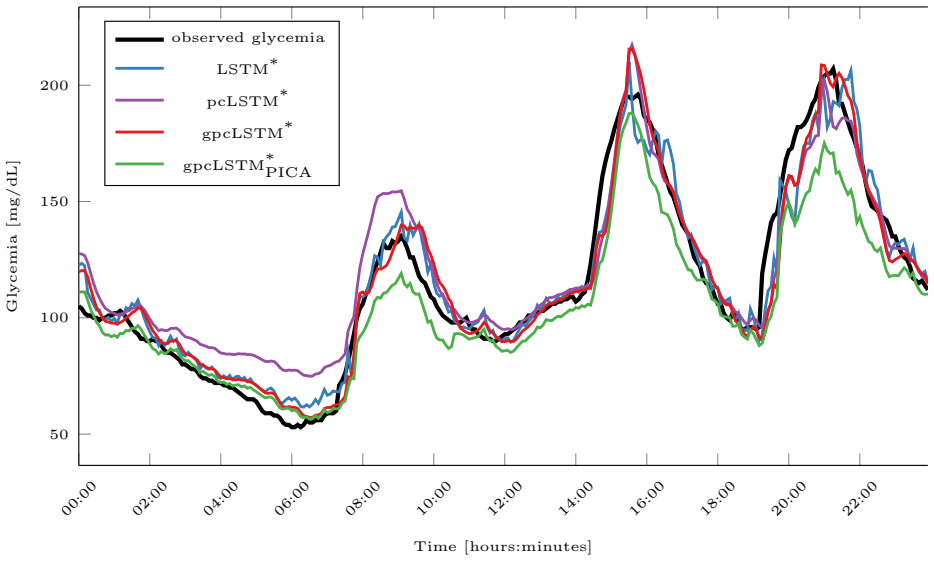


Fig. 7: Predictions of the LSTM*, pcLSTM*, gpcLSTM* and gpcLSTM$^*_{\text{PICA}}$ models for the patient 575 from the OhioT1DM dataset for a given day.

The gcMSE action is more focused on the decrease of the EP rate, as shown by the models gpcLSTM, gpcLSTM$_{\text{CA}}$, gpcLSTM$^*_{\text{PICA}}$. In contrast with the exponential smoothing technique and the cMSE loss function, the gcMSE improves all glycemic regions, and in particular the

hypoglycemic region. Moreover, these improvements allow the LSTM neural network to surpass, in clinical acceptability, the SVR model which is the best model of the GLYFE benchmark study. Through Figure 7, we can appreciate the differences in the predictions of the different models. First, we can see the large variations and noise in the predicted glucose signal of the LSTM model. These oscillations are reduced for the other models, becoming closer to the observed glucose signal. However, when using the cMSE loss function (pcLSTM* signal in purple), we witness a large loss of accuracy in the hypoglycemia region (between 4:00 and 8:00 am). While the signal gpcLSTM$^*_{\text{PICA}}$ is very close to the signal observed in the hypoglycemia region, this is achieved at the cost of an overall drop in accuracy. Finally, gpcLSTM*, is a compromise between the two.

Although we can conclude on the strength of using the gcMSE loss function in the training of deep models predicting future glucose levels of people with diabetes, the different results show us that there are many possible tradeoffs between accuracy and clinical acceptability. The PICA algorithm proposed in this study aims at selecting efficiently the best compromise between accuracy and clinical acceptability based on selection criteria. Figure 8 gives a graphical representation of the changes in MASE, general AP rate and general EP rate of the models throughout the PICA algorithm for all the patients. As previously discussed, there is no clinical criterion for glucose predictive models yet, so the only criterion for stopping the algorithm here was the MASE exceeding 1. The figure first shows us that the number of iterations before stopping the algorithm is variable from one dataset to another, and also from one patient to another (25.0 ± 3.96 for the IDIAB dataset, and 11.66 ± 5.06 for the OhioT1DM dataset). This is explained, first of all, by the variable initial accuracy of the different patients, some patients being easier to predict than others (see iteration 0 on Figures 8a and 8b). As we have observed through the analysis of Table 2, the main improvements in clinical acceptability are made at the first iteration (iteration 1) of the algorithm when introducing the gcMSE loss function and exponential smoothing. Nevertheless, throughout the algorithm, the clinical acceptability gradually improves at the expense of the accuracy. We can see that the rate of deterioration and improvement is different from one patient to another, showing the very high inter-person variability of the diabetic population.

From Figure 8, we can also derive the computing time gained by using the PICA algorithm instead of standard grid search in the identification of the optimal solution. Here the calculations are made given that a full training of a model and its finetuning last for 250 and 50 epochs, respectively. In average, 1492 and 833 epochs were needed for the PICA algorithm for the IDIAB and OhioT1DM datasets respectively. In comparison, a grid search of 30 and 20 iterations would have taken a total of 7750 and 5250, yielding a 5 to 6-fold decrease of the computing time made by the PICA algorithm.

Even though there is currently no clinical criterion for glucose prediction models, we can analyze the use of two hypothetical criteria through Table 4: a minimum AP rate, and a maximum EP rate. As expected, the harder the clinical criteria (higher threshold and/or combination of criteria), the lower the number of patients passing the clinical test. Only one patient in the IDIAB dataset managed to have simultaneously more than 97% AP and less than 1% EP. In addition, we can note a greater success of IDIAB patients on these clinical tests, compared to OhioT1DM patients. As previously mentioned, these differences in clinical performance are due to the difference in experimental systems. While the final evaluation of the OhioT1DM dataset is done every 5 minutes, it is done every 15 minutes for the IDIAB dataset. In addition, the glucose signal of IDIAB patients is overall less noisy, and therefore more stable and easier to predict. Thus, for a future practical use, the clinical criteria must be rigorously standardized.

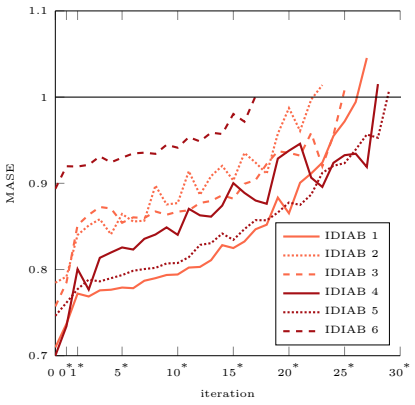
Finally, we note that the MASE on the testing set (the one reported in Tables 2 and 3) is slightly higher than 1 (1.03 and 1.01 for the IDIAB and OhioT1DM datasets). Using such a stopping criterion, we could have assumed that the final MASE on the testing set would be less than 1, as it is the case on the validation set. This happens because the test subset is not fully representative of the validation subset. This is due to the general small quantities of data in the datasets, negatively impacting the representativeness of these subsets. We also note that the standard deviation for the IDIAB dataset is higher, showing that the final value of the MASE is highly variable depending on the subject. Thus, the accuracy of the PICA algorithm would be improved by using more data (which would also improve the performance of the models in general).

Table 4: Number of patients within a given dataset that can satisfy different clinical criteria (minimal AP rate or maximal EP rate) through the PICA algorithm.

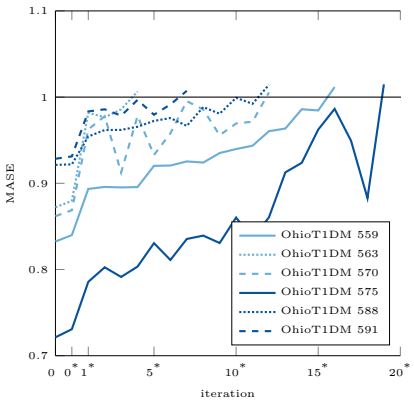
Clinical Criterion		**Dataset**	
AP (≥)	EP (≤)	IDIAB	Ohio
80	-	6	6
90	-	6	3
95	-	4	0
97	-	3	0
-	7	6	6
-	5	6	4
-	3	6	3
-	1	4	0
80	7	6	6
90	5	6	3
95	3	4	0
97	1	2	0

6. Conclusion

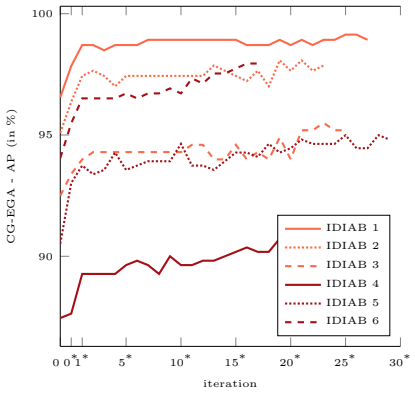
In this study, we have proposed a framework for the integration of clinical criteria into the training of deep models. Clinical criteria are often different from standard statistical metrics used as loss functions. As a consequence, the best model, given a loss function used during its training,



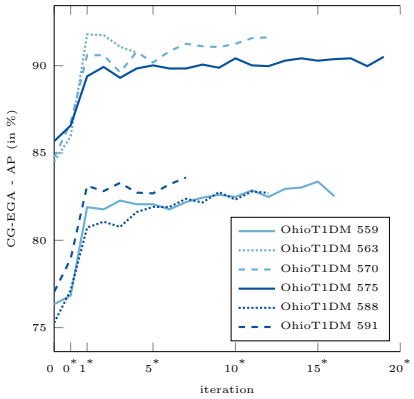
(a) MASE evolution of the IDIAB patients



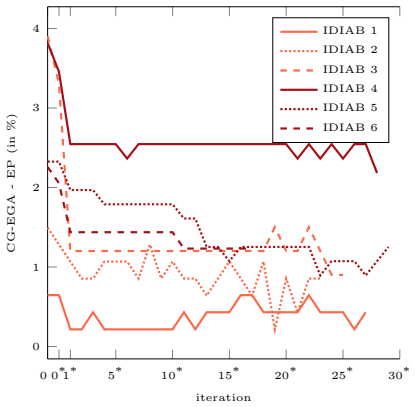
(b) MASE evolution of the OhioT1DM patients



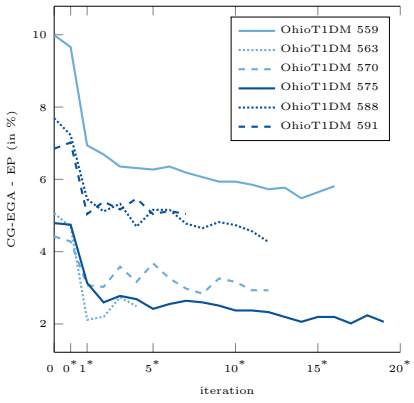
(c) AP evolution of the IDIAB patients



(d) AP evolution of the OhioT1DM patients



(e) EP evolution of the IDIAB patients



(f) EP evolution of the OhioT1DM patients

Fig. 8: Evolution of the MASE and CG-EGA (AP and EP) metrics throughout the PICA algorithm for the IDIAB and OhioT1DM datasets. Iterations 0 and 0^* respectively represent the results of the model trained with the MSE loss function before and after smoothing the predictions.

is not necessarily the model with the best clinical acceptability. We address this issue from the perspective of the challenging task of predicting future glucose values of people with diabetes.

In glucose prediction, the CG-EGA metric measures the clinical acceptability of the predictions. In particular, it assesses the safety of the predictions by looking at the prediction accuracy and the predicted rate of change accuracy. Moreover, the metric behaves differently for the different glycemic regions, some errors being more dangerous than others without being high amplitude errors. Starting from the cMSE loss function we proposed in a previous work (De Bois et al. (2019)) that penalizes the model during its training not only on prediction errors but also on predicted variation errors, we proposed to personalize the loss function to the glucose prediction task. Based on the CG-EGA, this personalization, called gcMSE, weights the errors differently depending on the scores obtained in the P-EGA and R-EGA. Finally, we proposed the PICA algorithm to obtain the solution that maximizes the accuracy of the model while at the same time satisfying given clinical criteria.

We evaluate the different proposed loss functions and the PICA algorithm with two different diabetes datasets, the IDIAB and the OhioT1DM dataset. First, we showed that the cMSE loss function increases the coherence of successive predictions, improving the clinical acceptability of the models. However, this improvement comes at the cost of a decrease in the accuracy of the model. Then, we showed that the gcMSE further improves the clinical acceptability by reducing the rate of life-threatening errors. Finally, we demonstrate the usefulness of the PICA algorithm that help in the selection of the desired tradeoff between general accuracy and clinical acceptability.

LSTM recurrent neural networks are not the only models that can use the proposed approaches. In future works, it would be interesting to apply them to other promising models. For instance, they could be used with models that, by nature, predict the whole signal trajectory up to the prediction horizon (e.g., kernel adaptive filters Yu et al. (2018)).

The analysis of different clinical criteria showed that not all the patients were able to meet them easily. This is related to the difficulty of the glucose prediction task of the patient, varying from patient to patient, but also to the nature of dataset, and in particular to the devices used for the data collection. These factors would need to be taken into account when creating future regulations for the use of such models by people with diabetes.

References

Ali, J. B., Hamdi, T., Fnaiech, N., Di Costanzo, V., Fnaiech, F., & Ginoux, J.-M. (2018). Continuous blood glucose level prediction of type 1 diabetes based on artificial neural network. *Biocybernetics and Biomedical Engineering*, *38*, 828–840.

Aliberti, A., Pupillo, I., Terna, S., Macii, E., Di Cataldo, S., Patti, E., & Acquaviva, A. (2019). A multi-patient data-driven approach to blood glucose prediction. *IEEE Access*, *7*, 69311–69325.

Brown, R. G. (2004). *Smoothing, forecasting and prediction of discrete time series*. Courier Corporation.

De Bois, M. (2019). Glyfe. URL: `https://github.com/dotXem/GLYFE` doi: `10.5281/zenodo.3234605`.

De Bois, M. (2020). Integration of clinical criteria into the training of deep models: Application to glucose prediction for diabetic people. URL: `https://github.com/dotXem/DeepClinicalGlucosePrediction` doi: `10.5281/zenodo.3904234`.

De Bois, M., Ammi, M., & El Yacoubi, M. A. (2019). Model fusion to enhance the clinical acceptability of long-term glucose predictions. In *2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE)* (pp. 258–264). doi:`10.1109/BIBE.2019.00053`.

De Bois, M., Ammi, M., & Yacoubi, M. A. E. (2020a). Glyfe: Review and benchmark of personalized glucose predictive models in type-1 diabetes. *arXiv preprint arXiv:2006.15946*, .

De Bois, M., El Yacoubi, M. A., & Ammi, M. (2019). Prediction-coherent lstm-based recurrent neural network for safer glucose predictions in diabetic people. In *International Conference on Neural Information Processing* (pp. 510–521). Springer.

De Bois, M., Yacoubi, M. A. E., & Ammi, M. (2020b). Adversarial multi-source transfer learning in healthcare: Application to glucose prediction for diabetic people. *arXiv preprint arXiv:2006.15940*, .

Deb, K., Agrawal, S., Pratap, A., & Meyarivan, T. (2000). A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: Nsga-ii. In *International conference on parallel problem solving from nature* (pp. 849–858). Springer.

Del Favero, S., Facchinetti, A., & Cobelli, C. (2012). A glucose-specific metric to assess predictors and identify models. *IEEE transactions on biomedical engineering*, *59*, 1281–1290.

Federation, I. D. (2019). Atlas du diabete de la fid neuvième édition 2019. URL: `https://www.federationdesdiabetiques.org/`.

Georga, E. I., Principe, J. C., Polyzos, D., & Fotiadis, D. I. (2016). Non-linear dynamic modeling of glucose in type 1 diabetes with kernel adaptive filters. In *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (pp. 5897–5900). IEEE.

Georga, E. I., Protopappas, V. C., Ardigò, D., Polyzos, D., & Fotiadis, D. I. (2013). A glucose model based on support vector regression for the prediction of hypoglycemic events under free-living conditions. *Diabetes technology & therapeutics*, *15*, 634–643.

Hyndman, R. J., & Koehler, A. B. (2006). Another look at measures of forecast accuracy. *International journal of forecasting*, *22*, 679–688.

Kovatchev, B. P., Gonder-Frederick, L. A., Cox, D. J., & Clarke, W. L. (2004). Evaluating the accuracy of continuous glucose-monitoring sensors: continuous glucose–error grid analysis illustrated by therasense freestyle navigator data. *Diabetes Care*, *27*, 1922–1928.

Li, N., Tuo, J., & Wang, Y. (2018). Chaotic time series analysis approach for prediction blood glucose concentration based on echo state networks. In *2018 Chinese Control And Decision Conference (CCDC)* (pp. 2017–2022). IEEE.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., & Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision* (pp. 2980–2988).

Marler, R. T., & Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization*, *26*, 369–395.

Marling, C., & Bunescu, R. C. (2018). The ohiot1dm dataset for blood glucose level prediction. In *KHD@ IJCAI* (pp. 60–63).

Martinsson, J., Schliep, A., Eliasson, B., & Mogren, O. (2019). Blood glucose prediction with variance estimation using recurrent neural networks. *Journal of Healthcare Informatics Research*, (pp. 1–18).

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

Mirshekarian, S., Shen, H., Bunescu, R., & Marling, C. (2019). Lstms and neural attention models for blood glucose prediction: Comparative experiments on real and synthetic data. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)* (pp. 706–712). IEEE.

Oviedo, S., Vehí, J., Calm, R., & Armengol, J. (2017). A review of personalized blood glucose prediction strategies for t1dm patients. *International journal for numerical methods in biomedical engineering*, *33*, e2833.

Pappada, S. M., Cameron, B. D., Rosman, P. M., Bourey, R. E., Papadimos, T. J., Olorunto, W., & Borst, M. J. (2011). Neural network-based real-time prediction of glucose in patients with insulin-dependent diabetes. *Diabetes technology & therapeutics*, *13*, 135–141.

Pérez-Gandía, C., Facchinetti, A., Sparacino, G., Cobelli, C., Gómez, E., Rigla, M., de Leiva, A., & Hernando, M. (2010). Artificial neural network algorithm for online glucose prediction from continuous glucose monitoring. *Diabetes technology & therapeutics*, *12*, 81–88.

Saiti, K., Macaš, M., Lhotská, L., Štechová, K., & Pithová, P. (2020). Ensemble methods in combination with compartment models for blood glucose level prediction in type 1 diabetes mellitus. *Computer Methods and Programs in Biomedicine*, *196*, 105628.

Sparacino, G., Zanderigo, F., Corazza, S., Maran, A., Facchinetti, A., & Cobelli, C. (2007). Glucose concentration can be predicted ahead in time from continuous glucose monitoring sensor time-series. *IEEE Transactions on biomedical engineering*, *54*, 931–937.

Taylor, J. W., & McSharry, P. E. (2007). Short-term load forecasting methods: An evaluation based on european data. *IEEE Transactions on Power Systems*, *22*, 2213–2219.

Yu, X., Rashid, M., Feng, J., Hobbs, N., Hajizadeh, I., Samadi, S., Sevil, M., Lazaro, C., Maloney, Z., Littlejohn, E. et al. (2018). Online glucose prediction using computationally efficient sparse kernel filtering algorithms in type-1 diabetes. *IEEE Transactions on Control Systems Technology*, *28*, 3–15.

Zarkogianni, K., Mitsis, K., Litsa, E., Arredondo, M.-T., Fico, G., Fioravanti, A., & Nikita, K. S. (2015). Comparative assessment of glucose prediction models for patients with type 1 diabetes mellitus applying sensors for glucose and physical activity monitoring. *Medical & biological engineering & computing*, *53*, 1333–1343.

Zhu, T., Li, K., Herrero, P., Chen, J., & Georgiou, P. (2018). A deep learning algorithm for personalized blood glucose prediction. In *KHD@ IJCAI* (pp. 64–78).