Contents lists available at ScienceDirect

Chemometrics and Intelligent Laboratory Systems
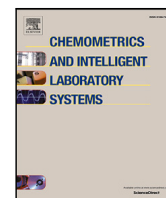
Combining wavelet transform with convolutional neural networks for hypoglycemia events prediction from CGM data

Jorge Alvarado [a,*], J. Manuel Velasco [b], Francisco Chavez [a], Francisco Fernández-de-Vega [a], J. Ignacio Hidalgo [b]

[a] Universidad de Extremadura, Santa Teresa de Jornet 38, Mérida, Spain
[b] Universidad Complutense de Madrid, Profesor José García Santesmases 9, Madrid, Spain

ABSTRACT

Estimating future blood glucose levels is an essential and challenging task for people with diabetes. It must be carried out based on variables such as current glucose, carbohydrate intake, physical activity, and insulin dosing. Accurate estimation is essential to maintain glucose values in a healthy range and avoid dangerous events of low glucose levels (hypoglycemia) and extremely high glucose values (hyperglycemia). Those situations maintained in time can cause not only permanent long-term damage but also short-term complications and even the death of the person. This paper proposes a new method to predict and detect hypoglycemic events over a 24-h time horizon. The technique combines applying the wavelet transform to glucose time series and deep learning convolutional neural networks. We have experimented with real data collected from 20 different people with type 1 diabetes. Our technique can also be applied to predict hyperglycemia. We incorporate a data augmentation technique consisting of a rolling windows system that improves the accuracy of the prediction. The uncertainty of the data is considered by the addition of controlled noise. The results show that the predictions obtained are accurate (higher than 88% of accuracy, sensitivity, specificity, and precision), confirming the effectiveness of the proposed method.

1. Introduction

Diabetes Mellitus (DM) is an increasingly relevant disease in the world, especially in developed countries. According to the World Health Organization, it will be one of the leading causes of death by 2030. DM is characterized by elevated blood glucose levels due to the fact that the pancreas does not produce insulin or the cells present resistance to the action of insulin. There are two main types of DM, type 1 (T1DM) and type 2 (T2DM). The pancreas of a person suffering from T1DM does not produce enough insulin, making the cells unable to regulate and process the glucose in the blood. This causes the blood glucose (BG) concentration to rise to potentially unhealthy levels. To solve the lack of insulin, people with T1DM apply injections of pharmaceutical insulin, *insulin boluses*, to replace what is not produced by the pancreas. The administration of these insulin doses is critical for the person since a miscalculation in the amount or type of insulin can cause the glucose to reach or down to dangerous levels. T2DM is the most extended and is usually controlled with a change in diet and

exercise habits combined with non-injected medicament. Sometimes is necessary also to administer some amount of insulin, as in T1DM.

Injecting the appropriate amount of insulin is challenging because different factors can affect this decision. It is crucial to correctly estimate future glucose values after the administration of insulin to avoid low glucose level events (hypoglycemia) and high values (hyperglycemia). Those are potentially dangerous to health if their effects are sustained over time and can even cause the death of the person. Other important tasks include estimating the grams of carbohydrates ingested, or evaluating exercise, stress, etc.

One of the most important factors in determining future glucose levels and preventing hypoglycemia and hyperglycemia events is knowing the glucose value at the time of prediction. Nowadays, people with diabetes widely use continuous glucose monitor (CGM) systems, consisting of a sensor that records and monitors interstitial glucose values. When the blood glucose level falls below a threshold, usually 70 mg/dL (5.5 mmol/L), it is said that the person is suffering hypoglycemia. On the opposite side, hyperglycemia occurs when the glucose

level is above 180 mg/dL (10 mmol/L). The main objective of glucose management protocols is to remain between those values as much time as possible, i.e. to maximize the time in the [70, 180] mg/dL range, or just *time in range*.

In particular, hypoglycemia is one of the main complications of diabetes [1]. These episodes of low glucose levels are dangerous to health as they can cause damage to multiple organs. If the effects of hypoglycemia are sustained over time, they can cause long-term health complications such as an increased risk of kidney disease, problems in the nervous system, etc. If the glucose level is lower than 54 mg/dL (3.0 mmol/L), a severe hypoglycemia episode occurs, which, if sustained over time, can be potentially very dangerous because it may produce cognitive impairment and even death of the person. Hypoglycemia episodes are usually classified depending on the moment of the day they happen and are usually established in three classes: daily, postprandial, and nocturnal hypoglycemia. Nocturnal hypoglycemia events are probably the most dangerous situation since they often go unnoticed due to the patient being asleep. As being asymptomatic episodes, its effects may be maintained over time, seriously affecting health and, eventually, causing the syndrome of *the dead in bed.*

Several works in the past addressed the problem of predicting hypoglycemia, although most of them need to know much historical information on several physiological variables. In this paper, we proposed an innovative method for predicting hypoglycemia events using only information on glucose values recorded by commercial CGM systems. Our method combines wavelet transform of glucose time series with deep learning to predict hypoglycemia events in the following 24 h. This novel *change of domain*, from *time* to *frequency*, generates valuable information through images that will feed a deep learning system of convolutional neural networks (CNN).

The main contributions of this paper are the following:

- To the best of our knowledge, this is the first work that tries to predict hypoglycemic events 24 h before they may occur.
- Wavelets Transforms (WT) has been used for the first time for hypoglycemia prediction.
- We propose the combination of WT and Deep Learning, transforming the signal from the time to the time–frequency domain. Our technique is not limited to blood glucose data; it can be applied to other time series with similar characteristics, particularly those featuring slight periodic behavior.
- Our technique can be applied also to predict hyperglycemia.
- We present a data augmentation (DA) technique consisting of a rolling windows system that improves the accuracy of the prediction.
- The uncertainty of the data is considered by the addition of controlled noise to it.

The rest of the paper is organized as follows. Section 2 revises the related work on both, the problem of hypoglycemia prediction, and the use of wavelets over time series. In Section 3 we explain in detail our approach and the techniques used in this work. Section 4 details the experimental setup and the results obtained by our technique. Finally, conclusions and future work are presented in Section 5.

2. Related work

In recent years, several attempts have been made to solve the problem of glucose prediction in general, and hypoglycemia and hyperglycemia prediction, in particular. There are different ways of modeling glucose dynamics, including data-driven models based on time-series data. Some of the techniques used in these models make use of machine learning (ML) algorithms such as Random Forest (RF) [2], K-Nearest Neighbors (KNN) [3], Support Vector Machine (SVM) or Autoregressive Integrated Moving Average (ARIMA) [4]. On the other hand, evolutionary computational methods such as Genetic Programming,

Grammatical Evolution [5], and Genetic Algorithms have also been applied.

Thanks to advances in high-performance computing, the use of neural networks (NN) offers promising results in glucose prediction [6]. The main types of Neural Networks are Multilayer Perceptron (MLP), Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Convolutional Neural Networks (CNN).

2.1. Hypoglycemia related work

Oviedo et al. [7] use SVM to predict postprandial hypoglycemia events with a time horizon of 240 min after the meal. Real data obtained using a CGM from ten patients with type 1 diabetes have been used. Two hypoglycemia thresholds were used to train and test the models: $Level1 : Glucose \leq 70$ mg/dL (3.9 mmol/L) and $Level2 : Glucose \leq 54$ mg/dL (3.0 mmol/L). The results showed a sensitivity of 79% and 71% for the first and second levels, respectively. One of the main limitations of the proposal is that it focuses only on postprandial hypoglycemia and not on events that can occur during the 24 h of the day.

In [8], Vu et al. proposed the use of a Random Forest (RF) to predict nocturnal hypoglycemia events, with an extended time horizon of up to 360 min. Only BG values obtained by a CGM have been used as input data. The dataset is large, with about 9800 real patients and an average of 100 days recorded per patient. The predictions have been divided into several approaches depending on the time of night: overall night (midnight to 6 a.m.), early night (midnight to 3 a.m.), and late night (3 a.m. to 6 a.m.). The results show that the model has good accuracy in the early evening with an Area Under the ROC Curve (AUC) of 0.90, and the prediction performance after 3 a.m. with an AUC of 0.75 can be improved. Compared with other works, this proposal is limited to detecting nocturnal hypoglycemia.

Dave et al. [9] propose a predictive model of hypoglycemia events based on machine learning techniques using Logistic Regression (LR) and Random Forest (RF). The time horizon to be predicted is a short term of 30 and 60 min. Features extracted from raw data from a CGM have been used as input. The threshold for detecting hypoglycemia events is Level 1, i.e. 70 mg/dL (3.9 mmol/L). The dataset comprises the real data of 112 patients with type 1 diabetes and 90-day average records for each patient. The results show that the proposed prediction model has a sensitivity greater than 91% at the 60-minute horizon and high performance for nocturnal hypoglycemia with a sensitivity of 95%. On the other hand, the predictive model's performance for horizons longer than 60 min is unknown.

In [10], Seo et al. used four machine learning models (RF, KNN, SVM, LR) to predict postprandial hypoglycemia events with a time horizon of 30 min. The dataset consists of 104 real patients, and for each patient, the BG level was recorded for an average of 3 days using a CGM. The different prediction models were compared, and the RF algorithm obtained the best accuracy in detecting hypoglycemia events with an average prediction performance with an AUC of 0.97, a sensitivity of 89.6%, and a specificity of 91.3%. This approach has several limitations; firstly, it focuses only on hypoglycemia events after meals and the short-term detection time horizon.

An LSTM neural network is used to predict hypoglycemia events in [11]. The time horizon for detecting these events is limited to only 30 min. The data have been obtained using a contact lens with an electrode that measures the level of BG contained in tear fluid. The dataset is relatively small as the number of patients used is not mentioned; only 2000 glucose measurements have been used. The results obtained by the predictive model show an accuracy of 80%.

Also, Porumb et al. [12] use neural networks, precisely two different approaches: a CNN and a combination of a CNN + RNN. This proposal aims to detect nocturnal hypoglycemia events with a probability of occurrence without a defined time horizon. The dataset is limited because there are only four real patients. The cardiogram (ECG) signal

at 5-minute intervals, and BG levels have been recorded for 14 consecutive days for each patient. The CNN model structures the network in 15 convolutional layers plus one fully connected layer. The CNN + RNN model comprises a CNN network with five convolutional layers and an RNN network consisting of a single LSTM layer. The results obtained from the models show an average predictive performance for all patients with an accuracy of 82.4%, a sensitivity of 87.5%, and a specificity of 81.7%.

A systematic literature review on data-based algorithms and models using real data for blood glucose and hypoglycemia prediction can be found in [13]. As Felizardo et al. have shown, all the approaches rely on the use of recorded data directly or with typical data preprocessing steps. In this paper, we propose a completely novel view that transforms the data from the time domain to the frequency domain, allowing the artificial intelligence system to be fed with images suitable for training deep learning systems.

2.2. Wavelets related work

The classification of time series has been approached from different points of view in the scientific literature. Initially, the idea was to find a distance function to establish a sense of closeness between different time series that would allow them to be grouped. The basic idea is to use the Euclidean distance and then group the series using an algorithm such as K-Nearest Neighbors. However, Euclidean distance cannot catch the structure and behavioral patterns of complex time series. Researchers have tried different distance functions to address this problem, perhaps the best known and most successful being the Dynamic Time Warping technique [14,15]. A survey that compiles the time series classification techniques knowledge before the arrival of deep learning is the paper of Wang et al. [16].

In the work by Ye et al. [17], we find the idea of transforming complex shapes found in nature (such as plant leaves) into time series and then applying time series classification techniques to them. However, this approach can also work in the other direction, i.e. transforming the time series to other domains to use different distance functions and classification techniques. An example of this idea is transforming the time series into a sequence of characters for subsequent use of language processing techniques [18]. However, the most widely adopted option is using a mathematical transformation as a preliminary step to the use of classification techniques, as we can find in [19]. Time series analysis using mathematical transformations is more than half a century old [20]. A summary of the transforms used to classify time series that do not include the wavelet transform can be found in the survey by Ji et al. [19].

Of all the mathematical transforms successfully employed in recent years, the wavelet transform [21] deserves special mention. Researchers in the financial world were probably the first to envision the vast possibilities of applying the wavelet transform to time series. As an example, we have the work of Masset (2015) [22], and Bolman et al. [23]. This research has been extended to other time series, as in Grant and Islam [24], or in Li et al. 2016 [25]. We can also find examples that mix wavelets and modern techniques such as convolutional networks [26].

Recently, we can find examples of researchers concerned with the classification of multivariate time series, Pasos et al. [27], and Middlehurst et al. [28]. The second one studies the inclusion of an ensemble of classifiers, an option that could also be used for univariate time series.

In the field of diabetes, we find the example of Ashok et al. [29]. This approach consists of a system to measure glucose in a non-invasive way using a laser beam and a photodetector. The Haar wavelet transform is used to decompose the signals the sensor receives. A predictive model based on a neural network with backpropagation is responsible for glucose prediction.

So, our work can be considered an extension or continuation of these works to predict hypoglycemia scenarios within a one-day horizon. Within the research literature focused on predicting hypoglycemia

events, we can find several examples using machine learning techniques, mostly with prediction horizons ranging from 30 min to 4 h, although none of them for 24 h.

3. Methodology

The method proposed in this paper, *Wavelet Transformation of Glucose* prediction system (WTG), is graphically explained in Fig. 1. The process starts from time series consisting on BG recordings at 15-minute intervals, 96 recordings in total per day:

1. A DA process is applied to the data (see Section 3.3).
2. The data for each patient is separated in 24-hour time series of BG values.
3. A threshold for the hypoglycemia T_{hypo} level is fixed by the user. The default value is 70 mg/dL.
4. The 24-hour BG time series are classified in two categories, (see Fig. 2), according to the following criteria:

    - **hypoglycemia**: at least one value in the following 24 h is lower than the established T_{hypo}.
    - **Non-hypoglycemia**: values on the following 24 h equal to or greater than T_{hypo}.

5. Once the classification has been performed, a spectrogram is generated from each glucose 24-hour time series by applying the wavelet transform. The spectrogram is an image resulting from the application of the corresponding wavelet function to each glucose value. The wavelet functions used are explained in Section 3.1. We selected 24 h in order to obtain images with enough information after applying the wavelet transform.
6. A deep learning Convolutional Neural Network (CNN) is trained with the spectrogram images generated in the previous step. Two different CNN architectures have been used to compare the performance. They will be detailed in Section 3.2.

Fig. 3 shows an example of a prediction of the proposed method. First, the glucose records of the last 24 h are collected as a vector of 96 values. The wavelet transform is applied to the data vector, generating the spectrogram image. The image is the input to the model, and the output is a numerical value resulting from applying the softmax function. For the example shown in Fig. 3, the output is $p = 0.28$, therefore, the prediction is labeled as *hypoglycemia*.

3.1. Wavelet transforms

The evolution of blood glucose levels is affected by numerous biorhythms whose cycles span different scales (weekly, daily, intake-related, etc..) but are not stationary. In other words, we do not expect the presence of these cycles to be constant but restricted to specific time segments. We need to use a mathematical transformation that can characterize the glucose time series with the information on the present frequencies, their magnitudes, and their variation over time. The most suitable tool for this task is the wavelet transform.

While the short time Fourier transform (STFT) [30] allows us to decompose a signal into multiple signals within frequency ranges of equal width, the wavelet transform goes further and provides a decomposition based on signals of equal bandwidth on a logarithmic scale, i.e. we have a resolution whose grain is frequency dependent. With the wavelet transform, we have high temporal and low-frequency resolutions for high-frequency events (we expect few such events in the glucose dynamics). It gives good frequency resolution but poor temporal resolution for low-frequency events (more common in the glucose time series). The latter makes the wavelet the most suitable transform for characterizing the evolution of glucose levels in the time–frequency plane.
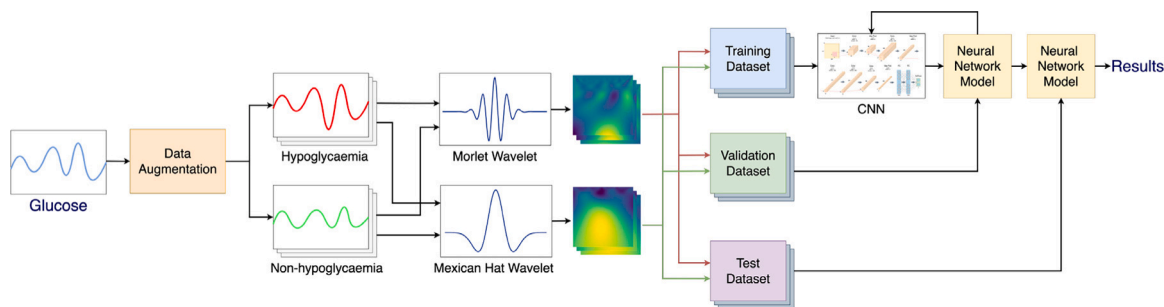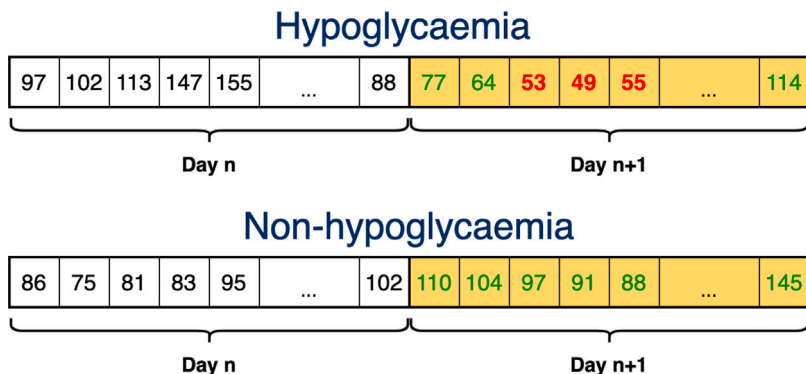
Fig. 1. Proposed method.

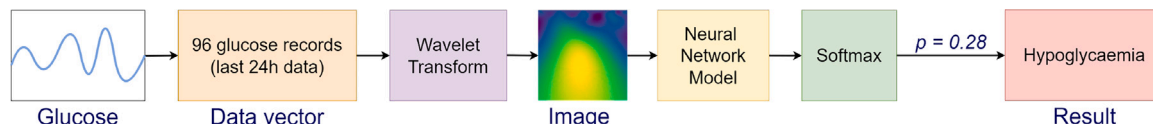**Fig. 2.** Classification of daily BG data into two categories: hypoglycemia and non-hypoglycemia.

Fig. 3. Prediction example of the proposed method.

The wavelet is a function with a small set of non-stationary oscillations. If these oscillations were stationary, we could say they have a particular frequency. If we change the size, i.e. the scale of the wavelet, we also change the corresponding hypothetical stationary frequency of the oscillations. Thus, when we perform a convolution of the wavelet with a given scale through our time series, we look for segments of the time series that show a frequency behavior equal to the hypothetical frequency of the wavelet. Changing the size (i.e. the scale) of the wavelet and repeating the convolution process (i.e. shifting the wavelet through our time series and calculating the convolution of the two functions) will give us the coefficients of the wavelet transform for different frequencies. The set of these coefficients can be represented in the form of a "scaleogram" or the form of a power spectrum. We will use this visual representation of the wavelet transform to characterize the time series of blood glucose levels.

So with the different values of "scale" and "shift", we will have a set of generating functions of the wavelet transform with the form:

$$\Psi_{scale,shift}(t) = \frac{1}{\sqrt{scale}} \Psi\left(\frac{t - shift}{scale}\right)$$

In addition, the shape of the oscillations produces different families of wavelets: Mexican Hat, Morlet, Shannon or Spline, among others. Following the results published in [31], the wavelet functions used in this paper are:

- Mexican Hat wavelet (also called Ricker wavelet). This function is mathematically defined as:

$$\Psi(t) = \frac{2}{\sqrt{3}\sqrt[4]{\pi}} e^{-\frac{r^2}{2}} (1 - r^2)$$

- Morlet Wavelet. This function is mathematically defined as:

$$\Psi(t) = e^{-\frac{r^2}{2}} cos(5t)$$

The Mexican Hat and Morlet wavelets have been chosen because they are the most used of the Continuous Wavelet Transforms (CWT) family. The advantages of CWTs are that they allow time–frequency analysis, which is important in the field for processing biomedical signals, such as glucose time series. These two families of wavelets complement each other, and their joint use improves both temporal and frequency analysis [31].

3.2. Convolutional Neural Networks

Convolutional Neural Networks (CNN) are a type of neural network widely used for image recognition, object detection, classification, anomaly detection, and time series-based prediction, among others.

Thanks to the advances in Deep Learning in recent years, multiple CNNs architectures have been developed. In this paper, we have experimented with different CNNs. First, we used the AlexNet [32] architecture, developed in 2012 by Alex Krizhevsky et al. The models have been trained from scratch without Transfer Learning because the Python module Keras Applications[1] used to implement the networks has no pre-trained AlexNet models. As can be seen in Fig. 4, the architecture of AlexNet embodies five convolutional layers, three pooling layers and
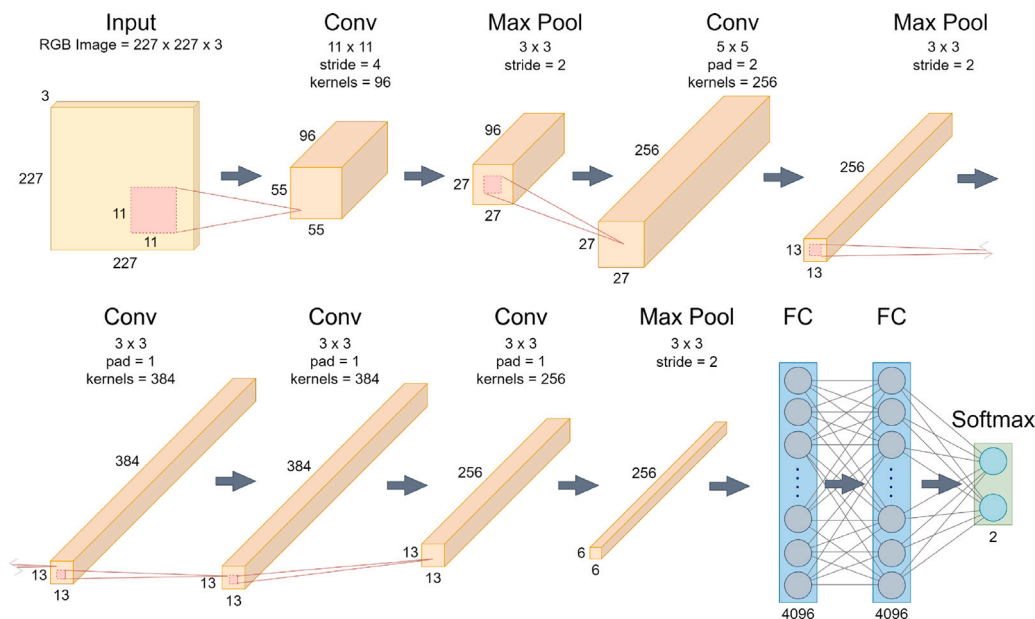
---

[1] https://keras.io/api/applications/

Fig. 4. AlexNet architecture.

two fully connected layers. The input of this network is RGB images of size 227×227 pixels, to which the first convolution of size 11×11 is applied, followed by a 3×3 max pooling. Next, the second convolution of size 5×5 and a 3×3 max pooling is performed. The resulting feature map is subjected to three successive 3×3 convolutions and a 3×3 max pooling operation. Finally, two fully connected layers of size 4096 neurons and the classification layer. This last layer uses the softmax function and has been adapted to our problem by setting two outputs, one for each category to be predicted. The softmax function is a mathematical function that takes a vector of real values as input and produces a real number as output, with values ranging between 0 and 1. This function is widely used in classification tasks and is frequently employed as the final layer in neural network models to transform raw scores into probabilities associated with classes. The softmax function is defined mathematically as:

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{N} e^{z_j}}$$

We also take use of the DenseNet [33] architecture, developed in 2017 by Gao Huang et al. Transfer Learning has been employed using a network pre-trained with the ImageNet [34] dataset. We have fine-tuned the fully connected layers and the last layer, leaving the rest of the layers frozen to improve the accuracy of the models. Fig. 5 shows the DenseNet configuration used in this work, known as DenseNet-121. The network has as input RGB images of size 224×224 pixels and then a 7×7 convolution is applied. DenseNet is organized into dense blocks, and transition layers are placed between these blocks. A dense block combines several layers formed by the composite function of batch normalization (ReLU) and a 3×3 or 1×1 convolution operation. These layers are interconnected so that the produced feature maps are concatenated at the end of the block, adding new information that can be reused. The transition layers are responsible for subsampling the feature maps between the dense blocks by batch normalization following a 1×1 convolution and a 2×2 max pooling. At the end of the last dense block, an average pooling of 7×7 is applied, and the output is connected to a dense layer of size 512. Finally, the classification layer has been adapted to our problem by using a layer of size 2 with a softmax function.

3.3. Data augmentation by rolling windows

In modern deep learning applications, the performance of neural networks is profoundly influenced by the size and diversity of the dataset on which they are trained. However, in specific domains such as endocrinology, acquiring extensive actual patient data for training purposes can be exceedingly challenging, mainly due to privacy concerns and the limited availability of comprehensive databases. In the context of this situation, data augmentation (DA) [35] has been used, offering an ingenious solution to enhance the utility of limited datasets, as is the case in blood glucose datasets [36]. DA is a fundamental machine learning and computer vision technique widely adopted to expand the size and diversity of training datasets artificially. It involves creating new, modified data points from the existing dataset by applying various transformations, such as rotation, scaling, cropping, or, in our case, a rolling window technique. These transformed data points diversify the dataset, reducing the risk of overfitting and improving the model's ability to generalize to unseen data. In the context of our research, DA is essential due to the scarcity of hypoglycemia events in our dataset. To overcome this limitation, we have innovatively devised a data augmentation technique that uses a rolling window approach. By employing this method, we effectively generate additional data points, each representing a different portion of the original time series data, thus simulating various hypoglycemia events. This augmentation amplifies the dataset and introduces diversity in the patterns of hypoglycemia events, making our model more robust and capable of handling a more comprehensive range of scenarios. Our approach, using a rolling window for DA in the context of hypoglycemia prediction, is an innovative adaptation tailored to our specific needs that, to the best of our knowledge, has not been implemented before. The problem we face is:

If we use an entire one day for each patient prediction, we would have only one spectrogram for each day of data. Moreover, the detection of hypoglycemia events should be performed once a day. In addition, not all the days present hypoglycemia events, so very few images would be produced, and consequently, the models would be unable to correctly detect them.

We propose in this paper to apply a technique consisting of a rolling window of values. A general description of the method can be observed
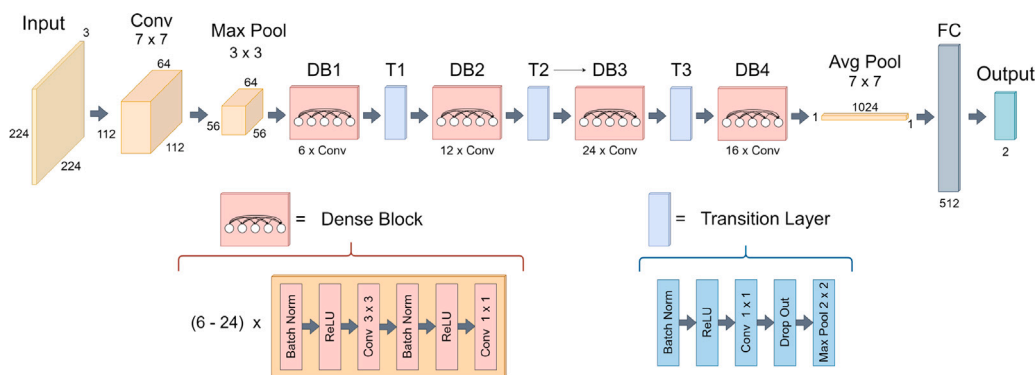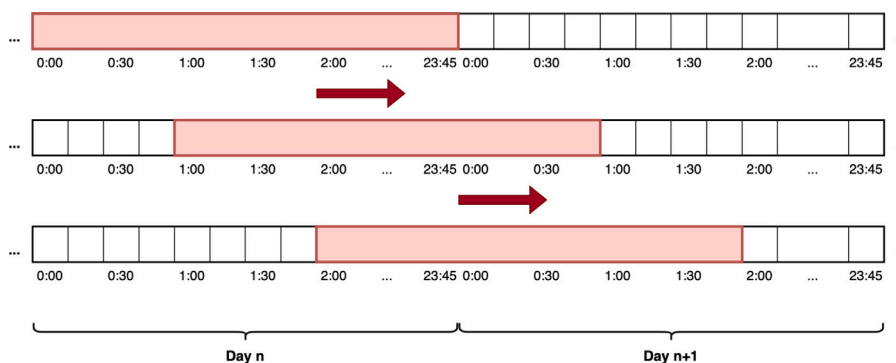
Fig. 5. DenseNet-121 architecture.

Fig. 6. DA using a rolling window of 1-hour size.

in Fig. 6. A rolling window of size 96 records (one day of data) with a 1-hour offset has been applied to the time series of glucose data. In this way we can generate up to 24 images per patient per day. As described, a wavelet function is applied to the glucose data to generate the corresponding spectrogram, obtaining up to 24 samples per day.

When comparing the models, we can see that the general models with rolling window obtains in test an accuracy of 80.15% (+/−0.09), sensitivity of 79.55% (+/−0.82), specificity of 80.83% (+/−1.23), precision of 81.21% (+/−1.70), AUC of 0.88 (+/−0.01) versus 50.76% (+/−4.05), 50.55% (+/−3.58), 51.15% (+/−4.87), 58.29% (+/−8.17), 0.53 (+/−0.04) obtained without rolling window. In the case of the individual models, we cannot perform this comparison because the models are unreliable, and some cannot be trained since so few images are generated without a rolling window. As can be seen, the rolling window allows us to obtain enough images to obtain robust models.

4. Experimental results

4.1. Experimental set-up

The experiments were carried out using a computer with 2 x Intel Xeon Silver 4310 at 2.1 GHz, 4 x NVIDIA A100 GPUs and 512 GB of RAM. The proposed method has been developed using Python with the TensorFlow and Keras libraries, together with the package called PyWavelets, which includes a selection of functions for the application of the wavelet transforms needed in the proposed work.

Table 1 shows the values of the hyperparameters applied to train the CNN models. To improve the results, a time-based learning rate scheduler has been used, which allows to gradually reduce the value of the learning rate over the training time. The learning rate for each epoch is calculated using the following expression:

$$lr = \frac{lr_0}{(1 + kt)}$$

Table 1
Hyperparameters are used to train the models with the different CNN architectures. SGD: Stochastic Gradient Descent. Adam: Adaptive Moment Estimation.

Param/Network	AlexNet	DenseNet-121
Epochs	300	300
Validation Interval (epochs)	30	30
Initial Learning Rate	0.001	0.001
Learning Rate Scheduler	Time-based Decay	Time-based Decay
Batch size	64	64
Optimizer	SGD	Adam
Loss Function	Categorical Crossentropy	Categorical Crossentropy

where lr_0 is the initial learning rate, k is the decay hyperparameter, and t is the epoch number. In this work, k is the result of dividing the initial learning rate by the total number of epochs.

In order to carry out the experiments, validate the learning of the algorithm used and test the results, it is necessary to divide the dataset as follows: 75% of the images have been used for training, 15% for validation, and 10% for testing. We ensure that there was not leakage of data from the training to testing sets and that images from the same patient were not present in both training and validation/testing sets.

Two different types of models have been trained: general models with the data of all patients and personal models with the data of each patient. In both types of models, images have been generated for each patient. Then, images from the majority class were randomly eliminated so that both classes were balanced with the same number of images. In the case of general models, they are composed by putting the images of all personal models without cross-validation.

4.2. Data

The results shown below are obtained from using a dataset consisting of 20 adults with type 1 diabetes. These data are real data, obtained

Table 2
Clinical characterization of each patient. ID, Gender, HbA1c, Age, Years of disease, Weight, Height, Treatment (MDI: Multiples doses of Insulin; ISCI. Infusion Subcutaneal continuous of Insulin).

Measure	HUPA0001P	HUPA0002P	HUPA0003P	HUPA0004P	HUPA0005P	HUPA0006P	HUPA0007P	HUPA0008P	HUPA0011P	HUPA0014P
Gender	Female	Male	Female	Male	Female	Male	Female	Female	Female	Female
HbA1c [%]	8.2	7.1	7.3	7.8	6.9	7.8	6.6	8.0	6.0	8.5
Age [years]	56.3	48.6	43.4	41.2	41.9	22.1	37.6	18.5	41.9	50.0
DX time [years]	15.5	36.5	12.5	8.5	39.5	13.5	10.1	8.7	15.2	12.9
Weight [kg]	59.0	82.4	62.0	88.0	58.5	71.0	102.6	83.5	51.0	61.0
Height [cm]	161	186	182	180	161	170	183	161	164	155
ISCI/MDI	ISCI	ISCI	ISCI	ISCI	ISCI	ISCI	ISCI	ISCI	ISCI	MDI

Measure	HUPA0015P	HUPA0016P	HUPA0017P	HUPA0020P	HUPA0022P	HUPA0023P	HUPA0024P	HUPA0025P	HUPA0026P	HUPA0027P
Gender	Female	Female	Female	Male	Male	Male	Male	Male	Female	Male
HbA1c [%]	6.4	6.5	8.2	9.7	6.7	7.7	8.3	7.0	7.2	7.0
Age [years]	43.1	29.9	26.3	45.7	59.6	22.9	47.9	38.1	61.8	26.4
DX time [years]	11.2	20.1	24.2	13.5	14.6	0.8	35.9	20.3	21.5	23.7
Weight [kg]	58.6	64.9	61.8	71.6	77.6	55.5	80.5	104.7	80.0	76.0
Height [cm]	162	157	167	168	179	173	174	188	165	185
ISCI/MDI	MDI	ISCI	MDI	MDI	ISCI	MDI	MDI	ISCI	MDI	MDI

Table 3
Glucose records, number of glycemic events and number of images generated for each patient.

Patient ID	Glucose records	Hypo events	Training images: All	Training images: Hypo	Test images: All	Test images: Hypo
HUPA0001P	1312	9	190	95	26	13
HUPA0002P	1312	34	32	16	6	3
HUPA0003P	1312	22	56	28	10	5
HUPA0004P	1027	12	92	46	14	7
HUPA0005P	1306	16	202	101	28	14
HUPA0006P	725	12	30	15	4	2
HUPA0007P	1304	18	84	42	12	6
HUPA0008P	1304	15	82	41	12	6
HUPA0011P	1248	10	170	85	24	12
HUPA0014P	1304	10	150	75	22	11
HUPA0015P	1248	14	134	67	20	10
HUPA0016P	1309	32	42	21	6	3
HUPA0017P	1293	11	202	101	28	14
HUPA0020P	1268	13	150	75	20	10
HUPA0022P	1248	21	56	28	10	5
HUPA0023P	1248	12	172	86	24	12
HUPA0024P	1248	15	106	53	16	8
HUPA0025P	1247	29	74	37	12	6
HUPA0026P	16 409	173	2568	1284	346	173
HUPA0027P	12 574	177	1220	610	164	82
Mean	**2562.30**	**32.75**	**290.60**	**145.30**	**40.20**	**20.10**
Std	**148.13**	**7.29**	**60.64**	**30.32**	**8.02**	**4.00**

by the medical team of the Hospital Principe de Asturias in Alcalá de Henares, Spain. The ethical committee of the Hospital approved the study, and patients signed an informed consent.

The data was obtained using Abbott FreeStyle Libre CGM sensors, that records glucose values in 15-minute intervals. Table 2 shows the clinical characteristics of the participants taking part in the study. The data presented in the table are: sex (50% female), age (40.16 +/− 11.86), weight (69.31 +/− 13.62) kg, height (169.31 +/− 9.69 cm), mean HBA1c (glycosylated haemoglobin) 7.45 (+/−0.93), years of illness 17.9 (+/−9.7), treatment: 60 >= 60% continuous subcutaneous insulin infusion (CSII) and 40 >= 40% multiple dose insulin (MDI). Moreover, Table 3 shows the distribution of hypoglycemia events per patient. Although, in general, the DA process manages to generate a high number of images, in specific datasets such as Patient HUPA0002P and HUPA0003P, the number of images generated is lower because hypoglycemia events are concentrated in time.
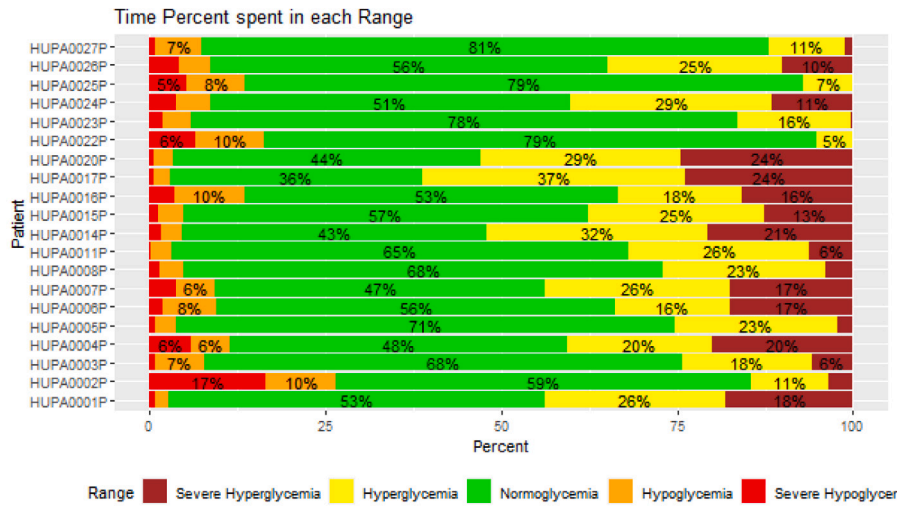
Fig. 7 shows the percentage of time in range for each patient. This value translates as the time the patient maintains the glucose value between 70 and 180 mg/dL. This value is displayed along with the percentages of time patients are in other situations, which are as follows: severe hypoglycemia (<54 mg/dL), hypoglycemia (54–70 mg/dL), hyperglycemia (180–250 mg/dL) and severe hyperglycemia (>250 mg/dL). It can be observed that most of the patients are most

of the time in the 70 and 180 mg/dL range. This may be due to the fact that these participants were not using the CGM regularly at the time of the study. Recent studies show that the use of CGM often increases the percentage of time in range of people with diabetes. In addition, Fig. 7(b) shows the patients' interquartile range and median glucose value. It can be seen that there is a large variability in the data, as the median glucose differs between patients. Along with these observations, we can highlight an important intra-patient glycaemic variability, as the standard deviation of glucose is high, which can be considered as another source of difficulty in prediction.

4.3. Results

Table 4 shows the accuracy of the models trained with the different CNN architectures and wavelet types, establishing a hypoglycemia threshold of <70 mg/dL. The combination of DenseNet-121 and the Mexican Hat wavelet produced the best results, with an accuracy of 80%, also in a general way, using Morlet wavelet.

On the other hand, the results of each patient's personal models are presented in Tables 5 and 6. Unlike the previous model, these models have trained only with the data of one patient and not the whole set. To establish the necessary comparisons, the general and the individual patient models have been trained using Mexican Hat and

(a)
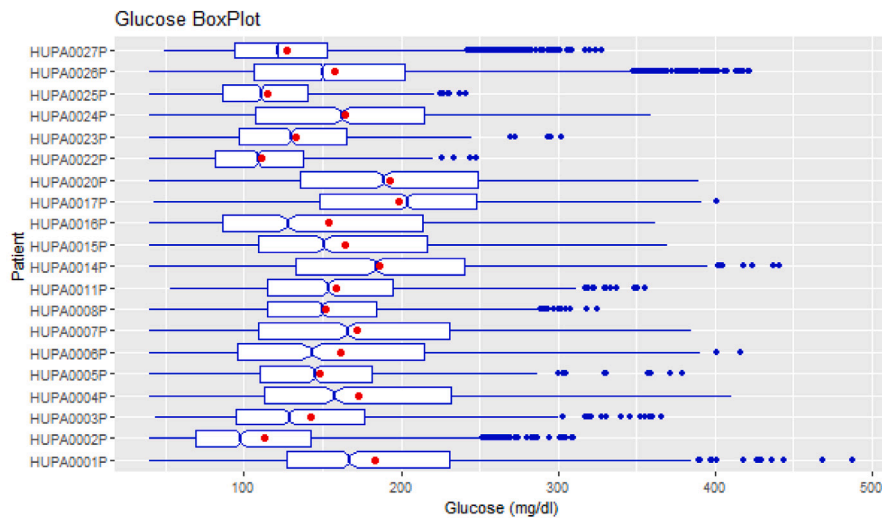
(b)

Fig. 7. Figure (a) shows the percentage of time the patient has a very low glucose level (<54 mg/dL), low ([54, 70] mg/dL), in range ([70, 180] mg/dL), high ((180, 250] mg/dL), and very high (>250 mg/dL). Figure (b) shows the interquartile ranges of glucose. The mean value is shown with a red dot. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 4
Comparison of general models trained with the different base networks and wavelets. Hypoglycemia threshold: <70 mg/dL.

Mexican Hat Wavelet							
Base Network	Train	Validation	Test				
	Accuracy	Accuracy	Accuracy	Sensitivity	Specifity	Precision	AUC
AlexNet	99.86 (+/−0.01)	70.79 (+/−1.13)	70.91 (+/−3.06)	71.50 (+/−2.31)	71.06 (+/−3.82)	69.99 (+/−9.99)	0.77 (+/−0.02)
DenseNet-121	98.61 (+/−0.09)	79.81(+/−0.49)	80.15 (+/−0.82)	79.55 (+/−1.22)	80.83 (+/−1.23)	81.21 (+/−1.70)	0.88 (+/−0.01)
Morlet Wavelet							
Base Network	Train	Validation	Test				
	Accuracy	Accuracy	Accuracy	Sensitivity	Specifity	Precision	AUC
AlexNet	99.95 (+/−0.00)	64.59 (+/−0.71)	64.99 (+/−1.10)	64.88 (+/−1.46)	65.18 (+/−1.11)	65.54 (+/−2.26)	0.71 (+/−0.01)
DenseNet-121	98.79 (+/−0.07)	73.91 (+/−0.68)	72.98 (+/−0.84)	72.98 (+/−1.27)	73.08 (+/−1.46)	73.08 (+/−2.63)	0.81 (+/−0.01)

Table 5

Comparison of personal models trained with DenseNet-121 and using Mexican Hat Wavelet. Results are shown as the average of all runs. Hypoglycemia threshold: <70 mg/dL. Red text indicates values below 50%.

Mexican Hat Wavelet							
Patient ID	Train	Validation	Test				
	Accuracy	Accuracy	Accuracy	Sensitivity	Specifity	Precision	AUC
HUPA0001P	100.00 (+/−0.00)	94.12 (+/−1.47)	98.72 (+/−1.81)	100.00 (+/−0.00)	97.62 (+/−3.37)	97.44 (+/−3.63)	1.00 (+/−0.00)
HUPA0002P	100.00 (+/−0.00)	87.50 (+/−7.05)	76.67 (+/−18.60)	75.00 (+/−45.83)	73.25 (+/−18.03)	53.33 (+/−37.20)	0.99 (+/−0.04)
HUPA0003P	100.00 (+/−0.00)	82.33 (+/−4.23)	43.33 (+/−9.07)	43.78 (+/−12.15)	43.30 (+/−7.57)	36.00 (+/−8.00)	0.41 (+/−0.09)
HUPA0004P	100.00 (+/−0.00)	80.74 (+/−3.43)	71.19 (+/−3.44)	71.43 (+/−4.98)	71.15 (+/−2.05)	71.43 (+/−4.16)	0.81 (+/−0.04)
HUPA0005P	100.00 (+/−0.00)	96.25 (+/−1.25)	97.74 (+/−3.26)	97.35 (+/−4.20)	98.44 (+/−3.69)	98.33 (+/−3.99)	1.00 (+/−0.01)
HUPA0006P	100.00 (+/−0.00)	70.56 (+/−8.26)	100.00 (+/−0.00)	100.00 (+/−0.00)	100.00 (+/−0.00)	100.00 (+/−0.00)	1.00 (+/−0.00)
HUPA0007P	100.00 (+/−0.00)	94.38 (+/−3.73)	86.39 (+/−5.48)	98.06 (+/−5.96)	79.78 (+/−6.06)	74.44 (+/−9.36)	0.94 (+/−0.05)
HUPA0008P	100.00 (+/−0.00)	93.33 (+/−4.82)	86.39 (+/−4.56)	100.00 (+/−0.00)	79.01 (+/−5.67)	72.78 (+/−9.11)	0.89 (+/−0.05)
HUPA0011P	100.00 (+/−0.00)	98.73 (+/−1.64)	85.42 (+/−2.34)	99.67 (+/−1.80)	77.64 (+/−2.85)	71.11 (+/−4.68)	0.96 (+/−0.02)
HUPA0014P	100.00 (+/−0.00)	88.89 (+/−2.17)	91.36 (+/−4.75)	95.95 (+/−4.66)	88.49 (+/−4.66)	86.67(+/−9.31)	0.99 (+/−0.01)
HUPA0015P	100.00 (+/−0.00)	93.97 (+/−2.37)	85.67 (+/−4.03)	94.20 (+/−7.61)	80.29 (+/−3.21)	76.67 (+/−4.71)	0.94 (+/−0.04)
HUPA0016P	100.00 (+/−0.00)	83.33 (+/−8.12)	67.22 (+/−8.03)	93.89 (+/−15.80)	61.22 (+/−6.32)	38.89 (+/−12.42)	0.73 (+/−0.09)
HUPA0017P	100.00 (+/−0.00)	94.83 (+/−2.03)	94.76 (+/−4.49)	91.39 (+/−6.86)	99.48 (+/−1.94)	99.52 (+/−1.78)	1.00 (+/−0.00)
HUPA0020P	100.00 (+/−0.00)	91.89 (+/−2.05)	91.17 (+/−2.79)	98.59 (+/−3.59)	85.95 (+/−3.67)	83.67 (+/−4.82)	0.98 (+/−0.01)
HUPA0022P	100.00 (+/−0.00)	77.00 (+/−9.00)	74.67 (+/−7.18)	75.02 (+/−7.56)	75.06 (+/−7.16)	75.33 (+/−8.46)	0.78 (+/−0.05)
HUPA0023P	100.00 (+/−0.00)	96.18 (+/−1.72)	93.19 (+/−2.28)	88.16 (+/−3.53)	100.00 (+/−0.00)	100.00 (+/−0.00)	0.99 (+/−0.01)
HUPA0024P	100.00 (+/−0.00)	91.33 (+/−2.21)	92.08 (+/−3.20)	87.76 (+/−2.67)	98.01 (+/−5.30)	97.92 (+/−5.67)	0.95 (+/−0.03)
HUPA0025P	100.00 (+/−0.00)	84.29 (+/−2.86)	94.72 (+/−5.04)	93.17 (+/−7.32)	97.46 (+/−5.70)	97.22 (+/−6.21)	0.99 (+/−0.02)
HUPA0026P	99.17 (+/−0.08)	85.53 (+/−0.67)	84.50 (+/−1.09)	86.87 (+/−1.33)	82.45 (+/−1.39)	81.31 (+/−1.79)	0.92 (+/−0.01)
HUPA0027P	97.59 (+/−0.25)	84.30 (+/−1.27)	85.93 (+/−2.16)	89.06 (+/−3.41)	83.62 (+/−3.27)	82.20 (+/−4.70)	0.94 (+/−0.01)
Mean	99.84	88.47	85.06	88.97	83.61	79.71	0.92
Std	0.55	7.17	13.01	13.41	14.46	18.85	0.13

Table 6

Comparison of personal models trained with DenseNet-121 and using Morlet Wavelet. Results are shown as the average of all runs. Hypoglycemia threshold: <70 mg/dL. Red text indicates values below 50%.

Morlet Wavelet							
Patient ID	Train	Validation	Test				
	Accuracy	Accuracy	Accuracy	Sensitivity	Specifity	Precision	AUC
HUPA0001P	100.00 (+/−0.00)	91.14 (+/−1.73)	91.92 (+/−3.76)	92.22 (+/−5.99)	92.16 (+/−3.77)	92.05 (+/−4.21)	0.98 (+/−0.01)
HUPA0002P	100.00 (+/−0.00)	81.11 (+/−5.67)	95.56 (+/−7.37)	93.33 (+/−11.06)	100.00 (+/−0.00)	100.00 (+/−0.00)	1.00 (+/−0.00)
HUPA0003P	100.00 (+/−0.00)	96.00 (+/−4.90)	54.00 (+/−5.54)	54.17 (+/−6.07)	53.89 (+/−5.24)	60.00 (+/−0.00)	0.72 (+/−0.04)
HUPA0004P	100.00 (+/−0.00)	70.74 (+/−3.78)	83.10 (+/−6.52)	77.10 (+/−6.22)	94.10 (+/−9.66)	94.76 (+/−8.64)	0.91 (+/−0.06)
HUPA0005P	100.00 (+/−0.00)	86.58 (+/−1.51)	78.69 (+/−4.07)	87.09 (+/−7.91)	73.74 (+/−3.41)	68.10 (+/−5.13)	0.85 (+/−0.03)
HUPA0006P	100.00 (+/−0.00)	71.11 (+/−7.37)	87.50 (+/−12.50)	100.00 (+/−0.00)	83.33 (+/−16.67)	75.00 (+/−25.00)	1.00 (+/−0.00)
HUPA0007P	100.00 (+/−0.00)	93.33 (+/−1.56)	83.06 (+/−4.56)	84.98 (+/−7.36)	81.96 (+/−4.01)	81.11 (+/−5.67)	0.90 (+/−0.04)
HUPA0008P	100.00 (+/−0.00)	85.42 (+/−4.06)	96.67 (+/−5.09)	96.03 (+/−6.60)	97.94 (+/−5.28)	97.78 (+/−5.67)	0.99 (+/−0.02)
HUPA0011P	100.00 (+/−0.00)	80.00 (+/−2.67)	90.28 (+/−3.93)	89.12 (+/−5.42)	92.31 (+/−5.81)	92.22 (+/−6.06)	0.98 (+/−0.01)
HUPA0014P	100.00 (+/−0.00)	82.89 (+/−2.23)	72.58 (+/−4.31)	76.61 (+/−6.84)	70.34 (+/−5.25)	66.06 (+/−8.44)	0.82 (+/−0.05)
HUPA0015P	100.00 (+/−0.00)	78.46 (+/−3.53)	82.67 (+/−5.73)	79.65 (+/−7.28)	88.25 (+/−8.21)	89.00 (+/−8.31)	0.92 (+/−0.04)
HUPA0016P	100.00 (+/−0.00)	98.75 (+/−3.75)	83.33 (+/−6.09)	97.78 (+/−8.31)	76.11 (+/−6.71)	68.89 (+/−8.31)	0.91 (+/−0.10)
HUPA0017P	100.00 (+/−0.00)	90.25 (+/−1.75)	89.52 (+/−3.19)	87.22 (+/−5.15)	92.70 (+/−3.18)	93.10 (+/−3.44)	0.96 (+/−0.02)
HUPA0020P	100.00 (+/−0.00)	90.11 (+/−2.51)	92.00 (+/−2.77)	100.00 (+/−0.00)	86.40 (+/−4.10)	84.00 (+/−5.54)	0.98 (+/−0.02)
HUPA0022P	100.00 (+/−0.00)	86.33 (+/−4.82)	74.67 (+/−7.18)	99.17 (+/−4.49)	67.14 (+/−6.57)	50.00 (+/−14.38)	0.96 (+/−0.04)
HUPA0023P	100.00 (+/−0.00)	80.29 (+/−2.96)	84.86 (+/−2.94)	85.51 (+/−5.44)	85.17 (+/−4.60)	84.72 (+/−5.73)	0.94 (+/−0.01)
HUPA0024P	100.00 (+/−0.00)	82.00 (+/−4.00)	59.17 (+/−6.40)	62.61 (+/−9.76)	57.65 (+/−5.57)	47.08 (+/−11.03)	0.64 (+/−0.05)
HUPA0025P	100.00 (+/−0.00)	78.57 (+/−0.00)	88.06 (+/−5.13)	84.62 (+/−1.64)	93.10 (+/−9.49)	92.78 (+/−10.26)	0.87 (+/−0.03)
HUPA0026P	99.29 (+/−0.10)	82.19 (+/−0.93)	76.83 (+/−1.45)	76.71 (+/−1.50)	77.06 (+/−2.17)	77.11 (+/−3.01)	0.85 (+/−0.01)
HUPA0027P	98.04 (+/−0.24)	78.39 (+/−0.84)	79.51 (+/−2.25)	79.66 (+/−3.07)	79.74 (+/−3.35)	79.55 (+/−4.85)	0.87 (+/−0.02)
Mean	99.87	84.18	82.20	85.18	82.15	79.67	0.90
Std	0.45	7.31	10.71	11.82	12.48	15.10	0.09

Morlet wavelets. The different characteristics of each patient and the lack of sufficient data in some of them explain the variations between the results obtained by the models. Moreover, like the general models, the Mexican Hat Wavelet obtains better results, but the difference is insignificant. It is worth highlighting the case of the patient with ID *HUPA0003P*, the low accuracy obtained by the model is because the patient has had poor glycaemic control, which has caused most of the images generated to be within the hypoglycemia range, as can be seen in Fig. 8.

On average, the prediction models fitted to each patient have an accuracy of better than 80%.

4.4. Discussion

Table 7 shows a comparison of our general model with and without noise with other machine learning techniques and neural networks. The models have been trained with data from all patients, and the hypoglycemia threshold <70 mg/dL has been used. In addition, the DA technique described in Section 3.3 has been used without generating images in the rest of the techniques, i.e., glucose time series have been used as input to the models. The results in Tables 4, 5, 6 and 7 are the average of 30 runs performed for each model. Our model with 1% noise applied is the best performing, obtaining 89% accuracy, 88%

Table 7

Comparison of our model with other machine learning techniques and neural networks. Results are shown as the average of all runs. Red text indicates values below 50%.

| Model	Accuracy	Sensitivity	Specificity	Precision
WTG with 1% noise	**89.42 (+/−0.55)**	88.75 (+/−0.94)	**90.14 (+/−0.74)**	**90.31 (+/−0.84)**
WTG without noise	**80.15 (+/−0.82)**	79.55 (+/−1.22)	80.83 (+/−1.23)	81.21 (+/−1.70)
Extra-Trees Classifier	72.25 (+/−1.20)	**98.52 (+/−0.44)**	19.39 (+/−1.94)	71.09 (+/−1.26)
Bagging Classifier	71.91 (+/−1.26)	97.27 (+/−0.65)	20.91 (+/−1.88)	71.22 (+/−1.26)
Random Forest Classifier	71.84 (+/−1.32)	97.06 (+/−0.60)	21.13 (+/−2.15)	71.23 (+/−1.37)
Gradient Boosting	69.33 (+/−1.30)	94.64 (+/−0.64)	18.43 (+/−2.05)	70.01 (+/−1.31)
Gaussian Process Classifier	68.52 (+/−0.91)	91.81 (+/−1.02)	21.68 (+/−1.69)	70.23 (+/−1.08)
Decision Tree Classifier	68.44 (+/−1.39)	92.97 (+/−3.03)	19.14 (+/−4.73)	69.84 (+/−1.47)
AdaBoost Classifier	62.26 (+/−1.24)	70.41 (+/−1.29)	45.88 (+/−2.28)	72.36 (+/−1.52)
Gaussian Naive Bayes	60.25 (+/−1.55)	61.06 (+/−1.79)	58.60 (+/−2.16)	74.81 (+/−1.11)
Multilayer Perceptron Classifier	56.40 (+/−10.16)	60.69 (+/−30.90)	48.40 (+/−32.62)	72.50 (+/−5.26)
Stochastic Gradient Descent Classifier	54.69 (+/−13.53)	63.35 (+/−41.36)	37.19 (+/−42.64)	66.02 (+/−19.92)
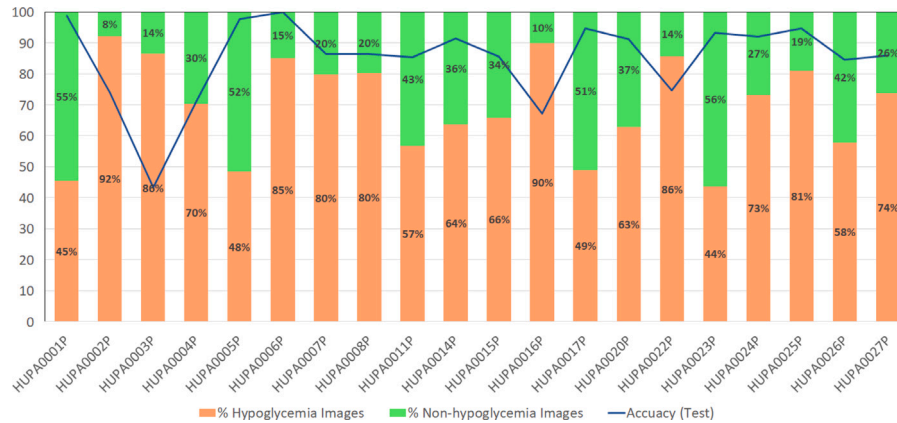
Fig. 8. Percentage of images generated with hypoglycemia (orange bars) and non-hypoglycemia (green bars) and the test accuracy (blue line) obtained by the models for each patient. Hypoglycemia threshold: <70 mg/dL, Wavelet: Mexican Hat. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

sensitivity, 90% specificity, and 90% precision. The results show that the other techniques obtain a higher sensitivity than our approach but achieve a very low specificity, even below 20%.

Fig. 8 shows the percentage of images generated with hypoglycemia and non-hypoglycemia and the test accuracy obtained by the models for each patient. These models have been trained with a hypoglycemia threshold <70 mg/dL and Mexican Hat Wavelet. It can be seen that the accuracy of the model is not related to the percentage of hypoglycemia images due to the different characteristics well as the amount of data available for each patient.

In Fig. 9, we present the ROC (Receiver Operating Characteristic) curves [37], AUC (Area Under the Curve) [38], and KS (Kolmogorov–Smirnov) statistic [39,40] which are commonly used measures in evaluating the performance of binary classification models.

ROC curves summarize the trade-off between a model's True Positive Rate (TPR) and the False Positive Rate (FPR) for various classification thresholds. The TPR is the proportion of actual positives correctly classified as positives, while the FPR is the proportion of actual negatives incorrectly classified as positives. A ROC curve plots the TPR against the FPR at different threshold settings, and the curve illustrates how well the model can distinguish between the positive and negative classes. The dashed line represents a random classifier, so the further the ROC line departs from the dashed line, the better the classifier. Another way of putting it is that the closer the ROC curve is to the upper left corner, the better.

The AUC is the Area Under the ROC Curve and is a commonly used metric for evaluating the overall performance of a binary classification model. The AUC ranges between 0 and 1, with 0.5 indicating random guessing and 1 indicating a perfect classifier. The KS statistic is another commonly used measure of model performance in binary classification. It is the maximum difference between the positive and negative classes'

cumulative distribution functions (CDFs). The KS statistic ranges between 0 and 1, with 0 indicating no separation between the classes and 1 indicating perfect separation.

In Fig. 9, we can see that classifiers using DenseNet have AUC values above 0.8, clearly better than those of classifiers using AlexNet. This fact can also be seen visually in the ROC lines (left column) and the separation between the cumulative distribution functions (right column). Similarly, there is a clear improvement when using the Mexican hat shape for the Wavelet function over the Morl for both DenseNet and AlexNet.

4.5. Uncertainty analysis

As seen before, the models have been trained using a fixed hypoglycemia threshold. In real life, the threshold may change due to each patient's characteristics. Therefore, an uncertainty analysis has been performed to check whether the proposed method is affected in a range of hypoglycemia thresholds, in particular, between 65 and 75 mg/dL. Fig. 10 shows the plots of the average and standard deviation of the test accuracy of personal models at different hypoglycemia thresholds. Figs. 10(a) and 10(b) show the results after 30 runs for the Mexican Hat wavelet and the Morlet wavelet, respectively. As can be seen, there are no significant changes in accuracy at the different hypoglycemia thresholds. There is little variability in the model outputs when the hypoglycemia threshold is modified.

4.6. Sensibility analysis

We conducted a sensitivity analysis to determine how much noise our method can handle without losing reliability. During this analysis, we deliberately added noise to the original glucose time series in our
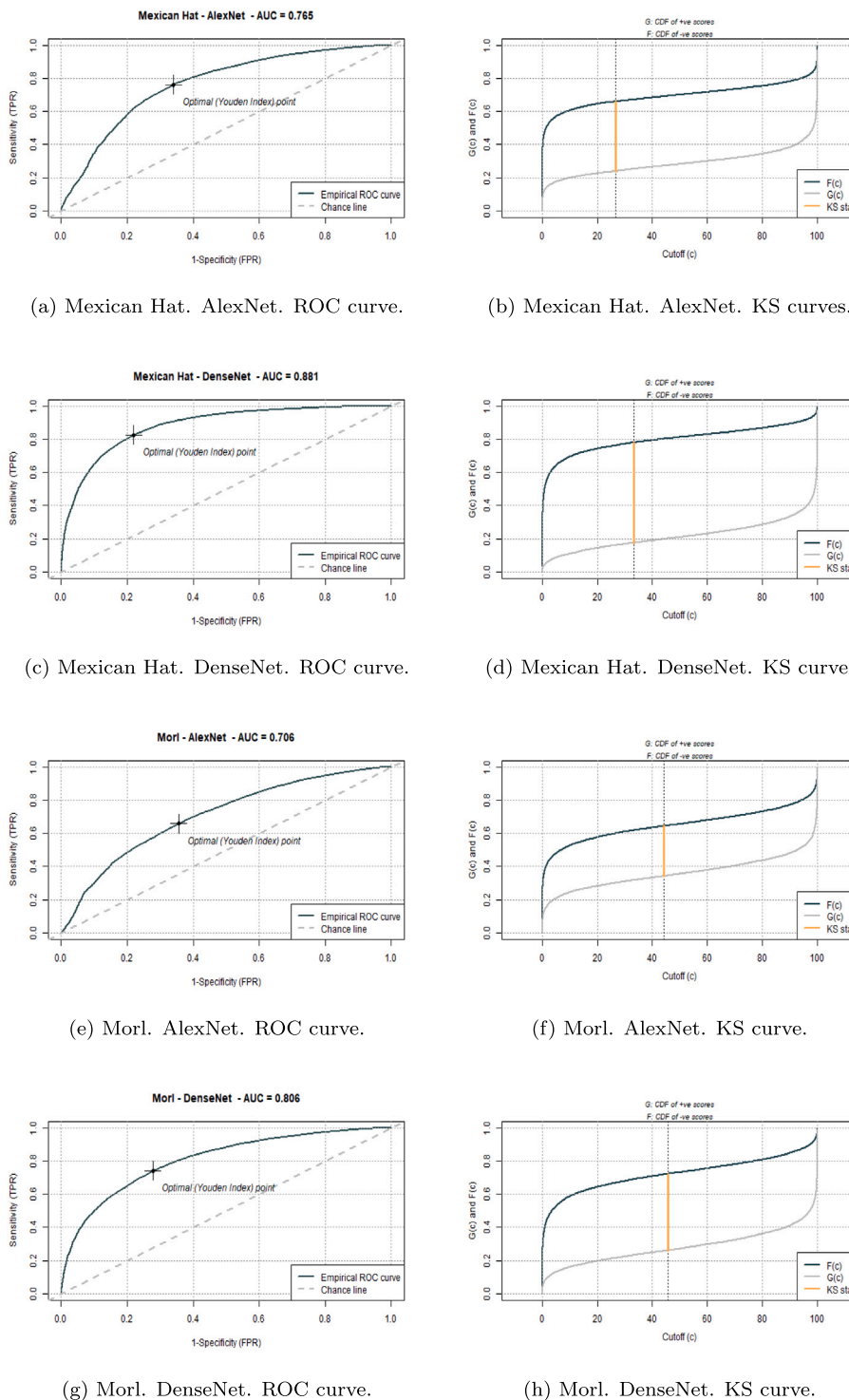
(a) Mexican Hat. AlexNet. ROC curve.

(b) Mexican Hat. AlexNet. KS curves.

(c) Mexican Hat. DenseNet. ROC curve.

(d) Mexican Hat. DenseNet. KS curve.

(e) Morl. AlexNet. ROC curve.

(f) Morl. AlexNet. KS curve.

(g) Morl. DenseNet. ROC curve.

(h) Morl. DenseNet. KS curve.

Fig. 9. ROC curves.

training data, ranging from 1% and 100%. This served the purpose of gauging our method's resilience and its performance in the presence of varying levels of noise. The reason for introducing this noise was to assess our method's capacity to work effectively in the face of the uncertainty and variability inherent in real-world glucose data. By systematically increasing the noise levels, we sought to understand how this affected the accuracy of our predictions. It is important to emphasize that our method is tailored for real-world use, and the intentional introduction of controlled noise helped us evaluate its ability to handle noisy input data while maintaining the reliability of its predictions.

Concerning the Continuous Glucose Monitoring System (CGMS) data, it is worth noting that this data source already introduces some level of noise and uncertainty due to factors like sensor accuracy and physiological variations, as documented in existing literature [41–43]. However, our sensitivity analysis focused on introducing an additional, controlled noise layer to assess how well our method performs under varying noise levels, extending beyond what is naturally present in CGMS data. This enabled us to gain a deeper understanding of how our technique responds to different noise levels and its ability to make accurate predictions in real-world scenarios.
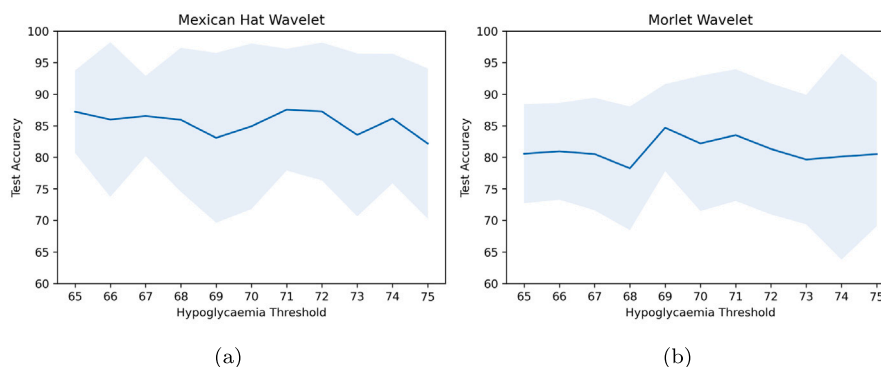
Fig. 10. Uncertainty analysis. Plots of the average (solid line) and standard deviation (blue shadow) of the accuracy in test of personal models at different hypoglycemia thresholds between 65 and 75 mg/dL. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
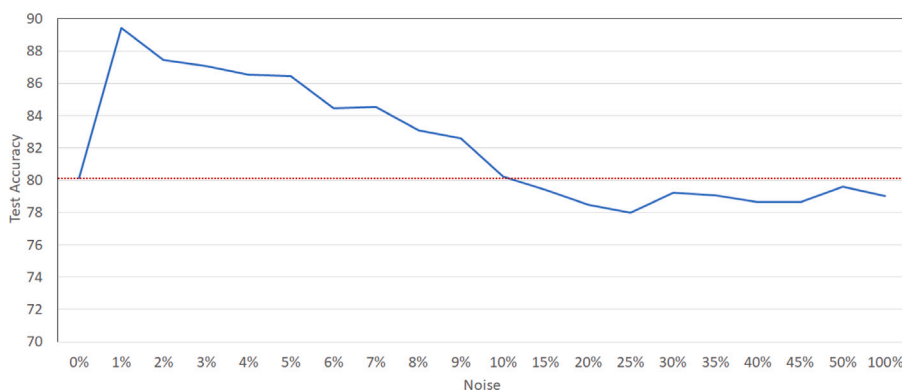

Fig. 11. Sensibility analysis: The red dot line marks the accuracy of the model trained in the absence of noise. The blue line indicates the accuracy when adding different percentages of noise to the dataset. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Once the noisy time series are obtained, they are incorporated into the training set. The images are generated using the wavelet transform in the same way as in Section 3.

Fig. 11 shows the sensitivity analysis. It can be observed that the accuracy without noise is approximately 80%. When noise is applied, the accuracy rises due to the increase in the number of hypoglycemia images, i.e., the training dataset grows. From a noise level of 10%, the accuracy degrades due to more images of the non-hypoglycemia class being generated because there is less probability of generating an image that is within the hypoglycemia range (40–70 mg/dL) than the non-hypoglycemia range (>70 mg/dL).

5. Conclusions and future work

In this work, we have proposed a new method, *WTG Prediction system*, for the detection of hypoglycemia events in people with diabetes by combining the application of the wavelet transform and Deep Learning techniques, specifically, the use of convolutional neural networks. The main advantage is that it is that we only use BG levels, and the technique is easily implementable on a mobile phone, which allows the person to improve she/his quality of life. The method is also applicable to the detection of any BG level, which makes it specially interesting for predicting dangerous situations such as hyperglycemia events.

Our results show that WTG performs successfully in predicting hypoglycemic events 24 h in advance. We have approached the problem from two points of view, a general model where we have used all the patient data available to us and, a customized approach for each patient. The results show that customized models outperformed general models. We can affirm that the customization of intelligent models based on the data of each patient is positive for the prediction process.

In order to study the sensitivity of the proposed system, controlled noise has been incorporated to data for training the general models.

Surprisingly, the application of this noise produces an improvement in the models, as seen in Table 7. By applying noise to the time series, we also increase the number of spectrograms and, thus, the information used for training the models. It should be noted that when more than 10% noise is applied, the accuracy decreases because, with these noise levels, information on hypoglycemia events is lost, which means that the model does not have the information it needs to fit. This decrease occurs because a hypoglycemic image is less likely to be generated as there is a smaller threshold of values (40–70 mg/dL).

The data used in this work come from 20 real patients with different characteristics obtained from a study in a Spanish hospital. At this point, it is worth mentioning that at the time of data collection, the patients were unfamiliar with the use of CGMs to monitor their glucose levels, which may have influenced the time and range of data obtained from each patient.

Comparison of our model with other machine learning techniques shows that *WTG* obtains better accuracy results for the data of this study. Wavelet Deep learning models will be integrated into a mobile application that combines glucose monitoring, recommendations and alarm signals. Our 24-hour prediction will produce hypoglycemia alerts and recommend changes in their diet or insulin routine every 15 min.

The results obtained in this study allow us to address the problem of hyperglycemia detection in the future. Since this is a classification problem, we can employ the same technique described in this work using data from the same patients. In addition, we have the idea of a personalized adjustment of the hypoglycemia and hyperglycemia thresholds using intelligent algorithms to tailor the model designed for each patient more precisely. We are also working on a double DA technique applying a 1% noise to the time series combined with the moving window technique explained in Section 3.3 on data augmentation for personal models after the good results obtained in Tables 5, 6 and

Fig. 11. Finally, we will study our technique with different prediction horizons and the application of Early Stopping to avoid overfitting during model training.

It should be noted that our technique can be applied to other time series with similar behavior. For instance, in environmental monitoring, time series data from environmental sensors, such as weather conditions or pollution levels, could benefit from our approach for forecasting and identifying extreme events, such as severe weather patterns or pollution peaks. Additionally, our methodology is adaptable to energy forecasting, facilitating predictions of energy demand and detecting unusual consumption patterns. Moreover, the financial sector, including stock market data, currency exchange rates and cryptocurrency prices, demands time series analysis for prediction and anomaly detection. While financial time series typically exhibit less periodicity than blood glucose data, our approach could still be valuable in predicting stock price fluctuations and detecting financial anomalies.

CRediT authorship contribution statement

Jorge Alvarado: Methodology, Programming, Experiments, Draft Writing & editing. **J. Manuel Velasco:** Data Analysis, Methodology, Writing – review & editing. **Francisco Chavez:** Methodology, Writing – review & editing. **Francisco Fernández-de-Vega:** Writing – review & editing. **J. Ignacio Hidalgo:** Supervision, Writing – review & editing, Project administration.

Declaration of competing interest

The authors declare no conflict of interest.

Data availability

Data will be made available on request.

Acknowledgments

References

[1] J. Morales, D. Schneider, Hypoglycemia, Am. J. Phys. Med. 127 (10) (2014) S17–S24.

[2] M.R. Vahedi, K.B. MacBride, W. Wunsik, Y. Kim, C. Fong, A.J. Padilla, M. Pourhomayoun, A. Zhong, S. Kulkarni, S. Arunachalam, et al., Predicting glucose levels in patients with type1 diabetes based on physiological and activity data, in: Proceedings of the 8th ACM MobiHoc 2018 Workshop on Pervasive Wireless Healthcare Workshop, 2018, pp. 1–5.

[3] N.S. Tyler, C.M. Mosquera-Lopez, L.M. Wilson, R.H. Dodier, D.L. Branigan, V.B. Gabo, F.H. Guillot, W.W. Hilts, J. El Youssef, J.R. Castle, et al., An artificial intelligence decision support system for the management of type 1 diabetes, Nat. Metab. 2 (7) (2020) 612–619.

[4] I. Rodríguez-Rodríguez, I. Chatzigiannakis, J.-V. Rodríguez, M. Maranghi, M. Gentili, M.-Á. Zamora-Izquierdo, Utility of big data in predicting short-term blood glucose levels in type 1 diabetes mellitus through machine learning techniques, Sensors 19 (20) (2019) 4482.

[5] J.I. Hidalgo, J.M. Colmenar, G. Kronberger, S.M. Winkler, O. Garnica, J. Lanchares, Data based prediction of blood glucose concentrations using evolutionary methods, J. Med. Syst. 41 (9) (2017) 1–20.

[6] F. Tena, O. Garnica, J. Lanchares, J.I. Hidalgo, Ensemble models of cutting-edge deep neural networks for blood glucose prediction in patients with diabetes, Sensors 21 (21) (2021) 7090.

[7] S. Oviedo, I. Contreras, C. Quirós, M. Giménez, I. Conget, J. Vehi, Risk-based postprandial hypoglycemia forecasting using supervised learning, Int. J. Med. Inform. 126 (2019) 1–8.

[8] L. Vu, S. Kefayati, T. Idé, V. Pavuluri, G. Jackson, L. Latts, Y. Zhong, P. Agrawal, Y.-c. Chang, Predicting nocturnal hypoglycemia from continuous glucose monitoring data with extended prediction horizon, in: AMIA Annual Symposium Proceedings, Vol. 2019, American Medical Informatics Association, 2019, p. 874.

[9] D. Dave, D.J. DeSalvo, B. Haridas, S. McKay, A. Shenoy, C.J. Koh, M. Lawley, M. Erraguntla, Feature-based machine learning model for real-time hypoglycemia prediction, J. Diabetes Sci. Technol. 15 (4) (2021) 842–855.

[10] W. Seo, Y.-B. Lee, S. Lee, S.-M. Jin, S.-M. Park, A machine-learning approach to predict postprandial hypoglycemia, BMC Med. Inform. Decis. Mak. 19 (1) (2019) 1–13.

[11] T.M. Quan, T. Doike, D.C. Bui, S. Arata, A. Kobayashi, M.Z. Islam, K. Niitsu, et al., AI-based edge-intelligent hypoglycemia prediction system using alternate learning and inference method for blood glucose level data with low-periodicity, in: 2019 IEEE International Conference on Artificial Intelligence Circuits and Systems, AICAS, IEEE, 2019, pp. 201–206.

[12] M. Porumb, S. Stranges, A. Pescapè, L. Pecchia, Precision medicine and artificial intelligence: a pilot study on deep learning for hypoglycemic events detection based on ECG, Sci. Rep. 10 (1) (2020) 1–16.

[13] V. Felizardo, N.M. Garcia, N. Pombo, I. Megdiche, Data-based algorithms and models using diabetics real data for blood glucose and hypoglycaemia prediction–a systematic literature review, Artif. Intell. Med. 118 (2021) 102120.

[14] C.A. Ratanamahatana, E. Keogh, Making time-series classification more accurate using learned constraints, in: Proceedings of the 2004 SIAM International Conference on Data Mining, SDM, pp. 11–22, http://dx.doi.org/10.1137/1.9781611972740.2, arXiv:https://epubs.siam.org/doi/pdf/10.1137/1.9781611972740.2, URL https://epubs.siam.org/doi/abs/10.1137/1.9781611972740.2.

[15] C.-J. Hsu, K.-S. Huang, C.-B. Yang, Y.-P. Guo, Flexible dynamic time warping for time series classification, Procedia Comput. Sci. 51 (2015) 2838–2842, http://dx.doi.org/10.1016/j.procs.2015.05.444, International Conference On Computational Science, ICCS 2015. URL https://www.sciencedirect.com/science/article/pii/S1877050915012521.

[16] X. Wang, A.A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, E.J. Keogh, Experimental comparison of representation methods and distance measures for time series data, Data Min. Knowl. Discov. 26 (2012) 275–309.

[17] L. Ye, E.J. Keogh, Time series shapelets: a new primitive for data mining, in: KDD, 2009.

[18] J. Lin, E. Keogh, S. Lonardi, B. Chiu, A symbolic representation of time series, with implications for streaming algorithms, in: Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, DMKD '03, ACM, New York, NY, USA, 2003, pp. 2–11, http://dx.doi.org/10.1145/882082.882086, URL http://doi.acm.org/10.1145/882082.882086.

[19] C. Ji, X. Zou, Y. Hu, S. Liu, A 2D transform based distance function for time series classification, in: CollaborateCom, 2018.

[20] G.E.P. Box, D.R. Cox, An analysis of transformations, J. R. Stat. Soc. Ser. B Stat. Methodol. 26 (2) (1964) 211–252, URL http://www.jstor.org/stable/2984418.

[21] S.P. Nanavati, P.K. Panigrahi, Wavelet transform, Resonance 9 (3) (2004) 50–64, http://dx.doi.org/10.1007/BF02834988.

[22] P. Masset, Analysis of financial time series using wavelet methods, in: C.-F. Lee, J.C. Lee (Eds.), Handbook of Financial Econometrics and Statistics, Springer New York, New York, NY, 2015, pp. 539–573, http://dx.doi.org/10.1007/978-1-4614-7750-1_19.

[23] R. Bolman, T. Boucher, Data mining using morlet wavelets for financial time series, in: Proceedings of the 8th International Conference on Data Science, Technology and Applications - DATA, INSTICC, SciTePress, 2019, pp. 74–83, http://dx.doi.org/10.5220/0007922200740083.

[24] P. Grant, M.Z. Islam, Signal classification using smooth coefficients of multiple wavelets, 2021, arXiv abs/2109.09988.

[25] D. Li, T.F. Bissyandé, J. Klein, Y.L. Traon, Time series classification with discrete wavelet transformed data: Insights from an empirical study, in: SEKE, 2016.

[26] P. Liu, H. Zhang, W. Lian, W. Zuo, Multi-level wavelet convolutional neural networks, CoRR abs/1907.03128, 2019, arXiv:1907.03128.

[27] A.P. Ruiz, M. Flynn, J. Large, M. Middlehurst, A.J. Bagnall, The great multivariate time series classification bake off: a review and experimental evaluation of recent algorithmic advances, Data Min. Knowl. Discov. 35 (2) (2021) 401–449, http://dx.doi.org/10.1007/s10618-020-00727-3.

[28] M. Middlehurst, J. Large, M. Flynn, J. Lines, A. Bostrom, A.J. Bagnall, HIVE-COTE 2.0: a new meta ensemble for time series classification, CoRR abs/2104.07551, 2021, arXiv:2104.07551.

[29] V. Ashok, N. Kumar, Determination of blood glucose concentration by using wavelet transform and neural networks, Iran. J. Med. Sci. 38 (1) (2013) 51.

[30] E. Sejdić, I. Djurović, J. Jiang, Time–frequency feature representation using energy concentration: An overview of recent advances, Digit. Signal Process. 19 (1) (2009) 153–183, http://dx.doi.org/10.1016/j.dsp.2007.12.004, URL https://www.sciencedirect.com/science/article/pii/S105120040800002X.

[31] X. Mi, H. Ren, Z. Ouyang, W. Wei, K. Ma, The use of the Mexican Hat and the Morlet wavelets for detection of ecological patterns, Plant Ecol. 179 (2004) 1–19.

[32] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. 25 (2012) 1097–1105.

[33] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, Densely connected convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 4700–4708.

[34] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, L. Fei-Fei, ImageNet: A large-scale hierarchical image database, in: CVPR09, 2009.

[35] M.A. Tanner, W.H. Wong, From EM to data augmentation: The emergence of MCMC Bayesian computation in the 1980s, Statist. Sci. 25 (4) (2010) 506–516, URL http://www.jstor.org/stable/23061098.

[36] J.M. Velasco, O. Garnica, S. Contador, J.M. Colmenar, E. Maqueda, M. Botella, J. Lanchares, J.I. Hidalgo, Enhancing grammatical evolution through data augmentation: Application to blood glucose forecasting, in: G. Squillero, K. Sim (Eds.), Applications of Evolutionary Computation, Springer International Publishing, Cham, 2017, pp. 142–157.

[37] M.S. Pepe, The Statistical Evaluation of Medical Tests for Classification and Prediction, in: Oxford Statistical Sciences Series, 2003.

[38] K.H. Zou, A.J. O'Malley, L. Mauri, Receiver-operating characteristic analysis for evaluating diagnostic tests and predictive models, Circulation 115 (5) (2007) 654–657.

[39] D.S. Dimitrova, V.K. Kaishev, S. Tan, Computing the Kolmogorov-Smirnov distribution when the underlying CDF is purely discrete, mixed, or continuous, J. Stat. Softw. 95 (10) (2020) 1–42, http://dx.doi.org/10.18637/jss.v095.i10, URL https://www.jstatsoft.org/index.php/jss/article/view/v095i10.

[40] M.R.A. Khan, T. Brandenburger, ROCit: Performance assessment of binary classifier with visualization, 2020, R package version 2.1.1. URL https://CRAN.R-project.org/package=ROCit.

[41] X. Xie, J.C. Doloff, V. Yesilyurt, A. Sadraei, J.J. McGarrigle, M. Omami, O. Veiseh, S. Farah, D. Isa, S. Ghani, I. Joshi, A. Vegas, J. Li, W. Wang, A. Bader, H.H. Tam, J. Tao, H.-j. Chen, B. Yang, K.A. Williamson, J. Oberholzer, R. Langer, D.G. Anderson, Reduction of measurement noise in a continuous glucose monitor by coating the sensor with a zwitterionic polymer, Nat. Biomed. Eng. 2 (12) (2018) 894–906, http://dx.doi.org/10.1038/s41551-018-0273-3.

[42] O. Garnica, J. Lanchares, J. Velasco, J. Hidalgo, M. Botella, Noise spectral analysis and error estimation of continuous glucose monitors under real-life conditions of diabetes patients, Biomed. Signal Process. Control 61 (2020) 101934, http://dx.doi.org/10.1016/j.bspc.2020.101934, URL https://www.sciencedirect.com/science/article/pii/S1746809420300902.

[43] H. Zhao, C. Zhao, An automatic denoising method with estimation of noise level and detection of noise variability in continuous glucose monitoring, IFAC-PapersOnLine 49 (7) (2016) 785–790, http://dx.doi.org/10.1016/j.ifacol.2016.07.284, 11th IFAC Symposium on Dynamics and Control of Process SystemsIncluding Biosystems DYCOPS-CAB 2016. URL https://www.sciencedirect.com/science/article/pii/S2405896316304918.