Machine Learning Methods for Prediction of Physical System Behavior

by

Subhasish Das

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved April 2022 by the
Graduate Supervisory Committee:

Sandeep K.S. Gupta, Chair
Ayan Banerjee
Premananda Indic
Antonia Papandreou-Suppappola

ARIZONA STATE UNIVERSITY

May 2022

ABSTRACT

The advancement and marked increase in the use of computing devices in health care for large scale and personal medical use has transformed the field of medicine and health care into a data rich domain. This surge in the availability of data has allowed domain experts to investigate, study and discover inherent patterns in diseases from new perspectives and in turn, further the field of medicine. Storage and analysis of this data in real time aids in enhancing the response time and efficiency of doctors and health care specialists. However, due to the time critical nature of most life-threatening diseases, there is a growing need to make informed decisions prior to the occurrence of any fatal outcome. Alongside time sensitivity, analyzing data specific to diseases and their effects on an individual basis leads to more efficient prognosis and rapid deployment of cures. The primary challenge in addressing both of these issues arises from the time varying and time sensitive nature of the data being studied and in the ability to successfully predict anomalous events using only observed data.

This dissertation introduces adaptive machine learning algorithms that aid in the prediction of anomalous situations arising due to abnormalities present in patients diagnosed with certain types of diseases. Emphasis is given to the adaptation and development of algorithms based on an individual basis to further the accuracy of all predictions made. The main objectives are to learn the underlying representation of the data using empirical methods and enhance it using domain knowledge. The learned model is then utilized as a guide for statistical machine learning methods to predict the occurrence of anomalous events in the near future. Further enhancement of the learned model is achieved by means of tuning the objective function of the algorithm to incorporate domain knowledge. Along with anomaly forecasting using multi-modal data, this dissertation also investigates the use of univariate time series data towards the prediction of onset of diseases using Bayesian nonparametrics.

DEDICATION

I would like dedicate this work to my parents and my brother.

ACKNOWLEDGEMENTS

The culmination of this dissertation document was a result of the belief and effort put in by various individuals. I would like to express my sincere and heartfelt gratitude to each and every one of them.

I want to thank God for granting to me the determination, patience and wisdom needed to complete my dissertation work and for making the dream of a doctoral degree into reality.

I would like to thank Dr. Sandeep K.S. Gupta for his unwavering and helpful guidance, deep insights, and advice in academic and non-academic matters. It was a privilege to have been advised by him over the course of my doctoral journey.

I would also like to express my gratitude towards Dr. Ayan Banerjee for always being there when I needed him. His vast understanding of mathematical concepts and his calm composure were a blessing in times of conference and journal submissions. I would also like to thank Dr. Premananda Indic for his guidance and feedback on my work and for taking the time to serve on my committee. I would also like to express my heartfelt gratitude to Dr. Antonia Papandreou-Suppappola for being a pillar of support throughout my time at Arizona State University. Her continued guidance in matters both academic and personal saw me through numerous difficult situations. I would also like to take the time to extend my gratitude to all my colleagues at the iMPACT lab.

Finally, I would like to thank my parents for believing in me, their constant love and support over the years!

TABLE OF CONTENTS

CHAPTER Page

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Chapter 1

INTRODUCTION

1.1 Motivation

Rapid growth in the availability and accessibility of computational medical devices has caused a large amount of medical data to become available to domain experts like physicians, caregivers and non-domain experts with computational and mathematical backgrounds. Medical institutions of all scales record and collect patient data across various modalities on a daily basis leading to the transformation of the field into a data rich domain. As a result of data availability, new avenues of medical research have become prominent - computational research using health care data, bioinformatics, biostatistics and others. However, the increase in available data coupled with the complexity of the biological processes being investigated and the time sensitive nature of most disease, necessitates the design and development of techniques to model the underlying representations from the data and make informed decisions using the learned models to prevent fatal outcomes. Short-term forecasting of anomalous behavior in physical systems allows domain experts to mitigate and prevent threatening circumstances. In contrast, long-term variations and patterns uncovered from observed data allow experts to model physical system behavior and iteratively improve these models to fit patients at an individual level.

Medical data availability has caused the use of data driven machine learning techniques, previously applied to domains with large scale datasets, to further boost the process of disease discovery and analysis to improve patient care. Diseases or anomalies are a result of imbalance in the physical system of the human body. For example,

osteoclasts and osteoblasts are responsible for the resorption and formation of bone respectively. The slightest imbalance of this tightly coupled process leads to osteoporosis. These anomalies in the physical system are usually developed over a period of time. Therefore, clinical datasets have an inherent temporal factor embedded in them that contributes a significant amount to understanding and uncovering the nature of the disease being studied.

Empirical data driven learning techniques can be applied to learn models solely using observed data. On the contrary, domain specific non-linear physical system numerical models are formulated based on the physical nature of the system concerned and in turn are often difficult to simulate and contain simplifying assumptions. Therefore it is important to consider both aspects, observed data and the underlying domain specific knowledge, while developing predictive solutions in such domains so that they are accurate and explainable.

This work introduces patient specific adaptive techniques to utilize both temporal and morphological attributes of the physical system to predict possible future anomalous behavior of the system. The emphasis is specifically laid on developing techniques that incorporate domain specific knowledge into real time observations to enhance the accuracy of the predictions made. Changes in physiological parameters observed during temporal and morphological anomalies play a key role in understanding and in turn forecasting their occurrences at a later time.

1.2 Background

1.2.1 Prediction of Blood Glucose Levels in Type I Diabetes Mellitus Patients

According to a recent study (Sun *et al.* (2022)), global diabetes prevalence in adults between the ages of 20 to 79 years has been estimated to be around 10.5% of the

global population. Diabetes Mellitus or diabetes, is a chronic disease that causes life threatening episodes (Remme (2022); Dybjer *et al.* (2022); Escobedo-de la Peña *et al.* (2021); Lee *et al.* (2021)) that can disable affected individuals (Chamberlain *et al.* (2022); Chen *et al.* (2021); Gregg and Menke (2021)), requires expensive treatment regimes and reduces overall life expectancy. Affected individuals are categorized as either Type 1 Diabetes Mellitus (T1DM) where the body is unable to produce insulin, or as Type 2 Diabetes Mellitus (T2DM) where the body is unable to efficiently utilize the insulin produced.

Cyber-physical control systems are used in many critical infrastructures including smart grids, medical systems and autonomous vehicles. An important feature of such control systems is that the controller algorithm interacts with the physical system through sensing and actuation. A popular method used for developing a controller strategy for these systems is model predictive control. In such a strategy, a model of the physical system is used to obtain an estimation of the state variables that can accurately describe the physical system at a given time in the future. The prediction is then used in the decision algorithm of the controller. Often the prediction model may not accurately represent the practical deployment of the system. In such cases, the prediction model will be inaccurate and it can lead to wrong decisions made by the controller. In addition to inaccuracies, the physical system prediction models also require personalization. This involves evaluation of controller parameters and correct configuration to match the operation of the specific physical system instance. Such personalizations can only be obtained by observing the control system and utilizing the input output data. Hence, data driven approaches such as machine learning techniques can potentially be utilized to iteratively improve a physical model.

A case in point is that of closed loop blood glucose control systems or artificial pancreas. It uses the Bergman minimal model to estimate the future blood glucose

level of a person based on the current measurements of blood glucose, blood insulin, and interstitial insulin levels. The Bergman minimal model is only good for fasting conditions and does not take into account physical activity of a person or meal intake. Hence, the only marketed closed loop blood glucose control system can only operate during the night.

Being able to predict blood glucose levels in diabetes patients specially T1DM patients is critical in preventing hyperglycemic and hypoglycemic events from occurring. Bergman *et al.* (1981) presents a mathematical model to simulate the blood glucose levels given a specific set of biological parameters which is used in models for simulating blood glucose levels for blood glucose monitoring devices.

The artificial pancreas (AP) project (Cobelli *et al.* (2011)), a cyber-physical system, was developed as a counter measure to help T1DM patients regulate and stabilize blood glucose levels in free living scenarios. AP is a closed-loop blood glucose control system comprised of a glucose sensor, an insulin infusion device and a control algorithm. The control algorithm utilizes the readings from the glucose sensor and based on pre-computed thresholds regulates the amount of insulin to be injected as a response to the rise in blood glucose levels. The aforementioned control algorithms used in AP are based on mathematical models that describe blood glucose kinetics. One of the most commonly used blood glucose models is the Bergman minimal model (BMM) (Bergman *et al.* (1981)) which utilizes an individual's current blood glucose level, insulin level and interstitial insulin levels to estimate the blood glucose level in the near future. Several control algorithms for AP have been developed using the BMM (Banerjee *et al.* (2013); Banerjee and Gupta (2013, 2014)) and have shown that using such models in the control algorithm require a certain degree of personalization - extracting and tuning the model parameters on a per-person basis. This level of personalization can be attained by observing the control system and utilizing

the input-output data streams. As a result, data-driven approaches such as machine learning can be efficiently utilized to build and improve these predictive models.

Since the AP project has been widely researched, a large number of statistical and machine learning methods have been developed and iterated through in search for an optimal algorithm for the task of predicting blood glucose levels in the near future. Sparacino *et al.* (2007) use real time Continuous Glucose Monitor (CGM) device data, which is a time series, and apply two prediction methods namely using a first-order polynomial model for describing the time series and a model based on an auto-regression (AR) model. Of the two models presented, the AR model performs better than the first-order polynomial model. The authors in (Parker *et al.* (1999)) use model predictive controllers(MPCs) to predict the occurrence of hyper- and hypoglycemic event before they can occur. The non-linear system for glucose dynamics is represented as a linear input-output system by a novel algorithm developed by them. The authors also further improve the performance of prediction using a Kalman filter combined with their MPC implementation. Naive machine learning methods like Support Vector Regression (SVR) (Georga *et al.* (2012); Bunescu *et al.* (2013); Georga *et al.* (2011)) K-means clustering and k-nearest neighbors (Karegowda *et al.* (2012)) have also been used for blood glucose prediction with varying levels of success.

Neural Networks are one of the most popular and widely used forms of representation learning in the fields of machine learning and data mining. A single layer neural network is able to learn an almost accurate approximation of a function or the representation of a model. A fairly recent concept - deep learning using neural networks has made significant progress in the fields of data mining, artificial intelligence and natural language processing. The success of deep learning is based on the availability of large scale data and the advent of computation power in the form of multi-core

general purpose graphics processing units. These large-scale datasets usually have an underlying representation that is followed by all the samples in it. Deep learning networks strive to learn the best representation possible from these datasets and then apply the learned representations on test samples with high levels of classification or prediction accuracy. However in the event of these datasets being modeled using complex models that are highly non-linear in nature, or the entire dataset having a large amount of noise, deep learning networks may not be able to learn the underlying representation correctly. This in turn would lead to high prediction or classification errors by the same network on test datasets. In order to overcome these limitations of deep neural networks, a newer concept called Guided Deep Learning has begun to be applied in certain fields. The concept of guided deep learning is to be able to provide the deep neural network with a notion of the underlying model and help it learn the representation in a more efficient manner.

More recently, artificial neural networks (ANN) have been used for predicting blood glucose in diabetes patients due to their universal function approximation capabilities (Pérez-Gandía *et al.* (2010); Zecchin *et al.* (2012); Pappada *et al.* (2011); Tresp *et al.* (1999)). Since then neural networks have been widely used for prediction of blood glucose levels. For example - (Tresp *et al.* (1999)) use a recurrent neural network to predict the blood glucose level over a future horizon. Their work demonstrates that the application of neural networks for prediction has advantages over predictions using a compartment based model. The use of a wavelet based neural network along with principal component analysis to extract features from a time series of blood glucose level obtained from (Kok (2004)) is in (Zainuddin *et al.* (2009)). In (Kok (2004)), the authors use an artificial neural network based approach with acceptable levels of accuracy. We find the use of an artificial neural network for on-line glucose prediction using data from CGM devices but it is not able to predict sudden

changes in blood glucose levels that may occur Pérez-Gandía *et al.* (2010). Pappada *et al.* (2011) predict blood glucose levels from CGM devices using a neural network model with significant model accuracy.

When tested over longer periods of time, these ANN based solutions were found to have high prediction error. To overcome the restrictions of ANN based solutions and to introduce an even deeper level of automation, deep learning based solutions are explored. Deep neural networks (DNN) (Schmidhuber (2015)) are composed of a large number of hidden layers and are able to learn underlying representations more accurately from data with minimum level of human intervention. Deep neural networks have been successfully used to learn complex representations from datasets including images (Krizhevsky *et al.* (2012)) and also in natural language processing (Collobert and Weston (2008)). In certain applications, using deep learning without providing any guidance to the neural network can lead to problems including lower rates of predictions and classification. For example in the case of self-driving cars (Bojarski *et al.* (2016)), neural networks are used to learn how to navigate traffic conditions. However, in order to be able to successfully mimic a human driver, the system must be able to evaluate proper driving techniques. By providing the neural network with an idea of how the data is modeled, overfitting of data can be prevented in a large number of cases. Also guided deep neural networks will be more efficient in filtering out unwanted noise from real data as noisy samples would not be consistent with the guide given to the deep network.

Several deep learning based solutions have also been developed for prediction of blood glucose in diabetic patients (Mhaskar *et al.* (2017); Faruqui *et al.* (2019); Zhu *et al.* (2018); Gu *et al.* (2017); Munoz-Organero (2020)). Alongside DNN based solutions, due to the time varying nature of blood glucose data, recurrent neural networks have also been used for near-term prediction (El Idriss *et al.* (2019); Rabby

et al. (2021); Sun *et al.* (2018); Wang *et al.* (2020); Martinsson *et al.* (2018)). However DNN based solutions are still prone to overfitting while training. To overcome this limitation and to introduce aspects of the physiological model that is used for modeling the dataset, guided deep learning was introduced and has been used in various fields (Read *et al.* (2019); Yin *et al.* (2019); Zeng *et al.* (2018); Hu *et al.* (2020)). The authors in (Alashkar *et al.* (2017)) demonstrate that using an example-rules based guided deep neural network provides results superior to using only rule-based classification or only deep neural networks. This is because by providing a set of example-rules to the deep neural network results in the network learning the parameters that affect the dataset more accurately. The parameters which do not have a significant effect are not given much priority.

1.2.2 Prediction of Near-term Bradycardia in Pre-term Infants

Infants who are born prior to completion of 37 weeks of gestation are termed as preterm infants. They are usually kept under strict care and monitoring since their vital organs are still in the process of initial development. Due to their premature birth, these infants are very susceptible to health problems like hypoxemia (low oxygenation of blood), apnea, cerebral problems, gastrointestinal problems, immune system problems and various others that have long and short term effects (Blackburn (1995)).

A critical problem in preterm infants is bradycardia, which is a slower than normal heart rate that indicates low blood oxygen levels (Upton *et al.* (1992)). Infant and preterm infant heart rate, which is the number of times the heart beats per minute, is usually over 100 beats per minute (bpm) (Perlman and Volpe (1985)). During episodes of bradycardia, their heart rates are lower which leads to the reduction of blood velocity thus impacting the amount of oxygenated blood that can be circulated

to the developing organs (Pichler *et al.* (2003)). Bradycardia has both short and long term effects (Poets *et al.* (2015)) with the most severe being loss of life. Early detection of bradycardia is thus crucial to avoid negative long term effects. It is also to be noted that since these are preterm infants that are being monitored, their sudden movements often cause motion artifacts which may be wrongly detected as the onset of bradycardia. Along with early detection false alarms must be addressed to ensure that real episodes of bradycardia are addressed in time. A false alarm is caused when normal heartbeats are incorrectly detected as bradycardia events.

Various methods have been considered in the literature to predict the onset of bradycardia events. Most of these methods depend on metrics related to the R wave of the QRS complex, such as the peak-to-peak R-R interval (RRI) extracted from ECG signals (Pan and Tompkins (1985)). In (Gee *et al.* (2017)), point process theory was used to model instantaneous measures of RRIs between heart beats for use in predicting bradycardia prior to onset with a 15% false alarm rate. Bayesian online change point detection was used in (Gee *et al.* (2018)) to estimate sequential transitions between RRIs that lead to bradycardia; the RRIs were modeled using a lognormal probability density function (PDF). Multivariate regression predictive modeling was used to identify clinically significant bradycardia parameters with a 5% level of confidence (Truong (2018)). Predictive modeling was also used in employing decision trees (Mahmud *et al.* (2019)) with time-frequency based ECG features to classify active bradycardia events with 86.7% accuracy. Note that the aforementioned methods require features or prior knowledge of the occurrence of bradycardia. In the recent years, nonparametric modeling has drawn a great deal of attention in many areas of research (Mittal and Paragios (2004); Moraffah (2019); Moraffah and Papandreou-Suppappola (2019)).

1.3 Contributions

In this section we highlight the research problem and its accompanying challenges followed by the contributions made by this work to address the concerned problem.

1.3.1 Model Guided Neural Networks for Prediction of Blood Glucose Levels in T1DM Patients

Formally, we define the research problem as - given a non-linear physical system with time varying inputs and outputs, we want to develop a data driven model that can reliably predict the output state of the system for unseen system inputs while incorporating domain specific knowledge into the model, assuming that the system inputs, outputs and parameters are observable.

The research challenges that are addressed while developing a solution are defined as follows - for a given non-linear physical system with inputs $X_i = \{x_1^i, x_2^i, ..., x_N^i\}$ and their corresponding system outputs y_i, where $i \in 1, 2, 3, ..., N$,

- develop a solution that predicts the output state ahead in time with high levels of accuracy
- is adaptable to individual instances of the system
- is explainable with respect to the domain
- functions with reduced volumes of training data

This dissertation develops and demonstrates a neural network framework where an initial physical model can be used as a guide to a deep learning network to develop a more accurate and personalized data driven machine learning prediction model. This proposed framework is then enhanced with the addition of a model based objective

function for the neural network in conjunction with the use of recurrent neural networks to improve the understanding of temporal attributes. We apply this proposed model guided learning model towards the prediction of blood glucose data.

The following contributions were made in this area:

1. Model Guided Deep Learning: This work develops and implements a type of guided deep learning - one that uses a model to help the deep neural network learn the underlying representation. For data intensive applications, using deep neural networks it is possible to increase the accuracy of predictions or classifications. However, deep neural networks can also over-fit data and lead to lower rates of accuracy in some cases. In order to prevent these drawbacks while being able to achieve a high level of prediction accuracy we propose to introduce a *guide* to assist the neural network learn the underlying representation of the model and achieve convergence faster. The selected guide model is a complex non-linear physiological model (Bergmann Minimal model modified to account for physical activity and endogenous glucose production) that is representative of the data. As an application of the proposed model guided deep learning method, we predict the blood glucose levels in T1DM patients using data collected from CGM devices in a free living scenario.

2. Loss Function Optimization: Loss functions measure the relative drift of predictions (or classifications) made by a neural network when compared to the ground truth. Efforts are made to either minimize or maximize this drift based on the problem formulation. While applying deep learning in the proposed method, we try to minimize a loss function to ensure accurate predictions. A new custom loss function is implemented to further enhance the proposed model guided learning and this work studies the effects of applying the model through

the loss function as opposed to directly to the network.

3. Recurrent Neural Network based Model Guided Learning: A novel recurrent neural network based model guided deep learning utilizing the proposed custom loss function. Recurrent neural networks are widely applied in the prediction of time series data due to their innate ability to remember past observations and iteratively improved the learned model over time. By exploiting this inherent ability of recurrent neural networks in conjunction with the proposed loss function, this work implements and studies the effects of such a system compared to off-the-shelf implementations for the task of blood glucose prediction.

1.3.2 Prediction of Near-term Bradycardia in Pre-term Infants

Formally, we define the research problem as - given a non-linear physical system with a time varying output that can be observed and is prone to the occurrence of anomalous events, we want to develop a model that can reliably predict the onset of these aforementioned anomalous events prior to their occurrence in the near future.

The research challenges that are addressed while developing a solution are defined as follows - for a given non-linear physical system with N sequential output observations $(x_1, x_2, ..., x_N)$,

- develop an anomaly forecasting solution that has high prediction accuracy
- reduces the false positive rate
- utilizes only the observed data i.e. has no prior knowledge of system parameters
- functions with reduced volumes of training data

This dissertation develops and demonstrates a nonparametric solution that uses a small number of features from the observed output of the physical system to accurately

predict the onset of anomalies in the data prior to their occurrence in the near future. We apply this proposed nonparametric solution towards the prediction of bradycardia in preterm infants.

The contribution made while addressing this research problem is listed below.

1. Bradycardia Prediction using Non-parametric Kernel Density Estimation: We propose and implement a novel method to predict the onset of near-term bradycardia in preterm infants without prior knowledge. The proposed nonparametric method robustly and accurately estimates a nonspecific probability density function of the continuous non-bradycardia ECG segments and constructs an RRI confidence set that depends on a desirable level of detection accuracy. We then introduce a test to achieve the false alarm rate of 5% by inverting the test. In particular, this method can achieve probability of false alarm, P_{FA}, as low as 5%, which is significantly more accurate compared to the existing methods. This method does not require any prior features to be extracted from the ECG signal. The method also demonstrates through simulation we can robustly predict the onset of bradycardia for a given false alarm rate.

Chapter 2

MODEL GUIDED DEEP LEARNING APPROACH TOWARDS PREDICTION OF BLOOD GLUCOSE LEVELS

2.1 Problem Definition

Cyber-physical systems often use predictive models of the physical system for the decision making process. Although these predictive models may be theoretically sound, when applied in practice they may suffer from various inaccuracies which in turn degrades the performance of the system as a whole. Dealing with these inaccuracies in the implementation of predictive models is a complex task since the physical system is usually dependent on a large number of parameters. Instead of dealing with several different instances of the physical system and trying to resolve issues for each of them, we propose a way to improve the prediction capability of these suboptimal predictive models in the presence of artifacts in practice.

2.2 Novel Model Guided Deep Learning Approach

The predictive models used in cyber-physical systems are highly non-linear by nature. Since neural networks are capable of approximating non-linear computable functions, we use it to extract prediction models using a data driven approach. We opt to use a deep neural network since the added number of hidden layers allows it learn the underlying non-linear representation more efficiently and with a higher level of precision. Instead of allowing the deep neural network to freely learn from the data and extract models, we provide it with a *guide* in the form of the predictive model. This guide model must be integrated with the deep neural network.In this paper we

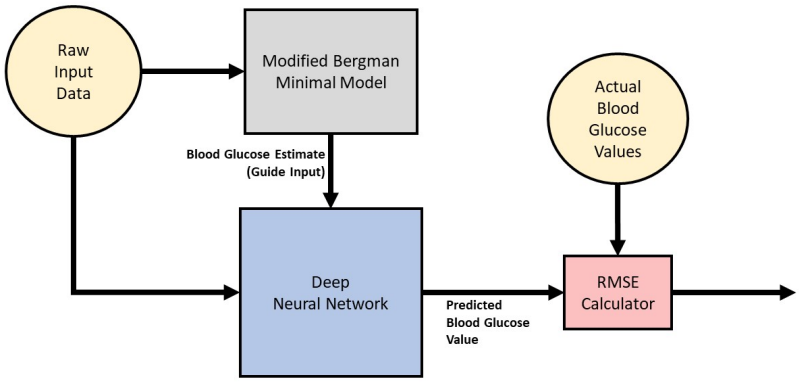


Figure 2.1: Architecture of Guided Deep Learning Using a Predictive Physiological Model as the Guide

integrate the predictive model into the deep neural network as an additional input feature for the network. For each sample of input features, the predictive model is used to generate an estimated value for the parameter that needs to be predicted. The estimate generated by the predictive model for each sample in the input data is a result of processing that sample using a mathematical model and hence captures the non-linearity of the predictive model. This estimated value is then added as an additional input feature for the deep neural network. The estimate of the parameter from the predictive model aids the deep neural network in learning only those parameters that are significant for correctly approximating the underlying model.

We show the usage of this proposed model guided deep learning on the closed loop glucose control system for T1DM patients. The predictive model used as the guide for our deep neural network is a physiological model called the Bergman Minimal Model. We improve upon the existing Bergman Minimal Model and use this modified version. Figure 2.1 is represents the overall architecture for our model guided deep learning approach applied to the prediction of blood glucose levels in T1DM patients.

2.2.1 *Modified Bergmann Minimal Model*

The model currently mostly used to research on the metabolism of glucose and insulin regulation is the Bergman minimal model. This model was proposed by the authors in Bergman *et al.* (1981) for the interpretation of glucose and insulin dynamics. The Bergman Minimal Model is a two compartment model - one model for glucose disappearance and the other model for insulin kinetics. The figure below shows the minimal model for glucose disappearance and the minimal model for insulin kinetics as per Van Riel (2004).

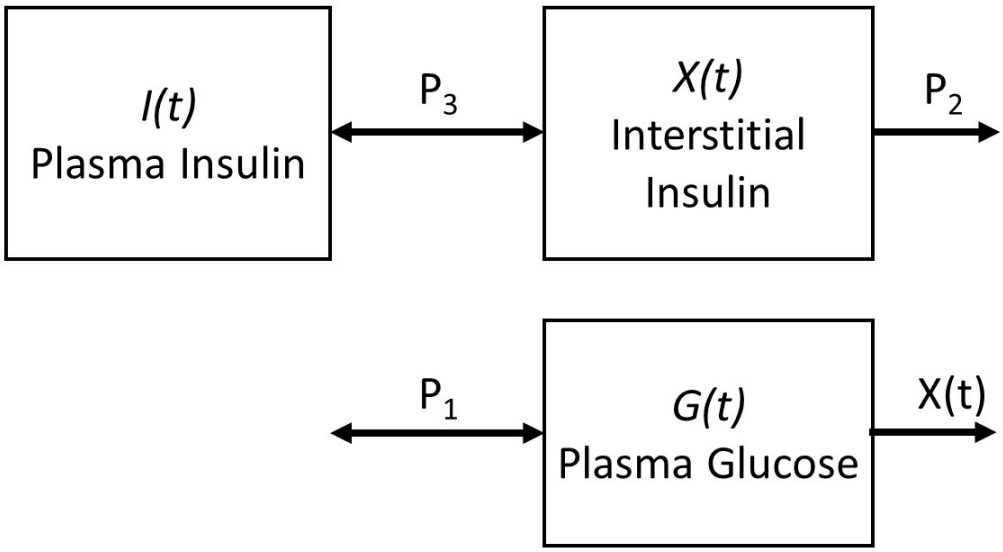


Figure 2.2: Two Compartment Bergman Minimal Model. Source - Van Riel (2004)

The equations describing the minimal model are given below -

$$\frac{dG(t)}{dt} = k_1(G_b - G(t)) - X(t)G(t), \qquad G(t_0) = G_0, \tag{2.1}$$

$$\frac{dX(t)}{dt} = k_2(I(t) - I_b) - k_3X(t), \qquad X(t_0) = 0, \tag{2.2}$$

t is the independent model variable time in minutes, t_0 is the time of the glucose injection, $G(t)$ is the plasma glucose concentration[mg/dL], $I(t)$ is the plasma insulin level[μU/mL] and $X(t)$ is the interstitial insulin activity. In order to improve blood glucose prediction in the original minimal model, physical activity factor was added along with the inclusion of Endogenous Glucose Production (EGP). As per Hovorka *et al.* (2004), the insulin sensitivity is linearly and very tightly associated to the EGP sensitivity. So in order to quantify the effects of interstitial insulin on the endogenous production of glucose, the EGP is accumulated with $X(t)$. By adding these two extra factors along with the meal intake, a more accurate way of simulating the dynamics of insulin was ensured.

This modified Bergman Minimal Model is what we use as a *guide* model and input to our neural networks. This modified implementation of the Bergman Minimal Model is implemented as a preprocessing step for our dataset and then fed to the neural networks as the *guide* input.

2.3 Evaluation Metric

The metric chosen for evaluating the different prediction algorithms was Root Mean Squared Error (RMSE). We compute the RMSE for each of the tested algorithms using the formula given below -

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(BG_{actual} - BG_{predicted})^2} \tag{2.3}$$

To evaluate the Bergman Minimal Model and the modified Bergman Minimal Model estimators we use the same formula for RMSE as mentioned above. The only change made was to use the estimated blood glucose values in these two cases. We determine the performance of a predictor (or estimator) in this paper by observing

the RMSE values for the predictions made. Since the dataset had been normalized prior to training and testing, an RMSE value closer to zero indicates that a more accurate prediction of blood glucose levels.

2.4 Dataset, Results and Discussion

2.4.1 Dataset and Preprocessing

Since the blood glucose level predictions depend on biological parameters that vary from person to person, the data was collected, trained and tested on an individual basis. Our dataset comprises of T1DM patient data collected from seven patients over a period of seven days. The CGM device collects blood glucose data every five minutes. A 16G accelerometer attached to the waist of each patient recorded their physical activity levels throughout the day and was also set to collect data every five minutes. Insulin sensitivity, basal and bolus rates and the EGP rate was determined for each patient individually and then recorded. The estimated blood glucose level from the modified Bergman Minimal Model is also fed as an input to the deep neural network. This dataset is standardized to zero mean and unity standard deviation. This was done to ensure a uniform scale among all the collected features and also to ensure that the deep neural network reached convergence faster.

2.4.2 Experimental Setup

The proposed solution was implemented on a personal computer system comprising of the following parts -

- Intel Core i7 4790 4 core CPU (base clock 3.60 GHz)
- 32GB RAM
- NVIDIA GTX 745 GPU with 4GB VRAM

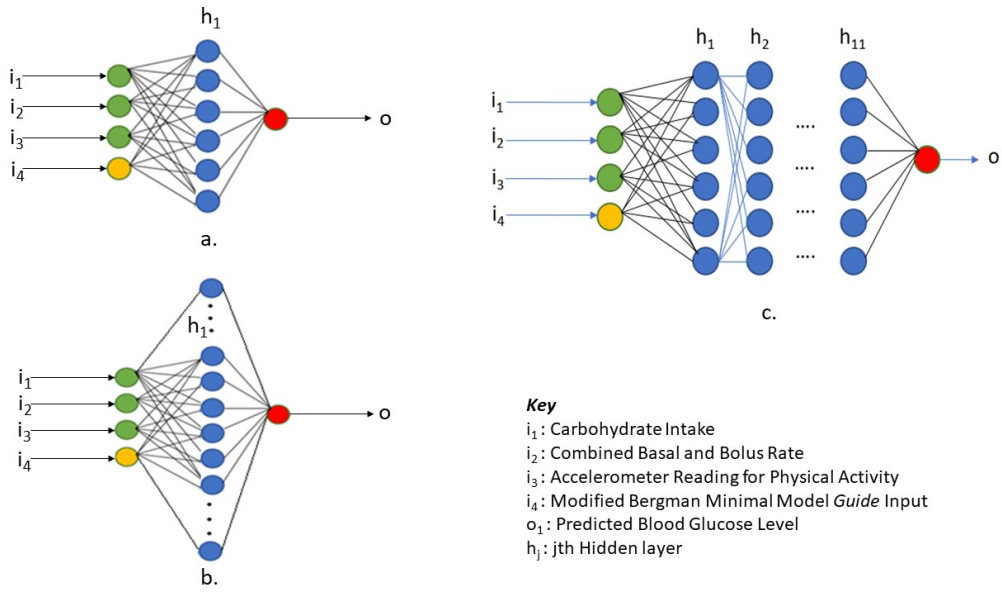


Figure 2.3: (a) A Single Layer Artificial Neural Network. (b) A Wide Neural Network Composed of a Single Layer of Five Hundred Neurons. (c) A Deep Neural Network Composed of Eleven Hidden Layers with Each Layer Composed of Fifty to Eighty Neurons.

- Python 3.6 for neural network
- MATLAB 2019a for preprocessing

We use three variants of neural networks ranging from a single layer neural network (Figure 2.3a.) to a wide network with a single layer (Figure 2.3b.) to a deep neural network (Figure 2.3c.). Each type of neural network used is described below.

The neural network model (NN-Model I) is a single layer artificial neural network. This neural network is composed of an input layer with input neurons corresponding to the number of input features, one hidden layer with five neurons and a single output neuron. The activation functions used for the hidden and output layer are the tan hyperbolic and linear function respectively. The performance of this neural network is reported later.

The wide neural network model (NN-Model II) is a neural network with an input layer, a single hidden layer and an output layer. The input layer is composed of the

same number of neurons as the number of input features. The hidden layer is composed of five hundred neurons, while the output layer only has a single neuron. This type of neural network is called a wide network because of the large number of neurons present in the hidden layer. The activation functions for the hidden and output layers are similar to the artificial neural network mentioned earlier- tan hyperbolic for the hidden layer and linear activation for the output layer.

The third variant of neural network tested was the deep neural network model (NN-Model III). The deep neural network, shown in Figure 2.3c, is composed of thirteen layers - one input layer with four external neurons, one for each of the input features, eleven hidden layers composed of neurons ranging between fifty to eighty neurons in each layer and one output layer comprised of one external neuron, providing the prediction of the blood glucose level. By method of trial and error, we optimize the number of hidden layers required to make our deep neural network. Using lesser than eleven hidden layers for our given dataset resulted in underfitting of the data while using in excess of eleven layers would result in over-fitting the data. The tan hyperbolic function is used as the activation function for each neuron in the hidden layers. The output layer neuron uses a linear activation function.

For all three models of neural networks we use mean squared error as the loss function and the Adam optimizer Kingma and Ba (2014). The initial weight distribution for the neurons is done according to Glorot and Bengio (2010) for the tan hyperbolic function. A batch size of 10 is used for all the neural network models and the learning rate was set to 0.001 and the epsilon value was set to 10^{-8}.

The modified Bergman minimal model and its effect on the estimation of blood glucose levels are tested using four different variants of the proposed guide model. In order to create each of the four different guide models one or more input features was added or removed from the modified Bergman Minimal Model implementation and

Table 2.1: Guide Model and the Corresponding Model Variant It Is Based On

Guide Model No.	Guide Input Defined by
GM-I	Bergman Minimal Model
GM-II	Bergman Minimal Model with the effect of Endogenous Glucose Production
GM-III	Bergman Minimal Model with the effect of physical activity
GM-IV	Bergman Minimal Model with the effect of Endogenous Glucose Production & physical activity

the same feature(s) was added or removed from the input to the neural network it was fed into. The four variants of the guide model are listed in Table 2.1. We will henceforth refer to each guide model as seen in Table 2.1.

We also use guided Support Vector Regression (SVR) using the guide models from Table 2.1 to predict the blood glucose values. The predictions made using this guided SVR approach is compared and contrasted in the results and discussion sections.

Training and testing of each of the aforementioned algorithms was done on a per patient basis. For each patient, the training dataset was composed of six days worth of data and then testing was done on the remaining day of data split into twelve hour sets. Each patient dataset was subject to each of the guide models mentioned in Table 2.1 above and then the data was input to each of the NN-Models with and without the model guide for training and testing.

Table 2.2: Bergman Minimal Model and Modified Bergman Minimal Model Prediction RMSE Values

	Modified Bergman Minimal Model	Bergman Minimal Model
P1	0.422	0.461
P2	0.387	0.450
P3	0.527	0.476
P4	0.467	0.513
P5	0.447	0.573
P6	0.442	0.488
P7	0.425	0.477

2.4.3 Results

In this section we present the results of all our experiments and demonstrate that our proposed model guided deep learning neural network outperforms the existing methods of predicting blood glucose levels in T1DM patients.

The Bergman Minimal Model and our modified Bergman Minimal Model provide estimates of the blood glucose levels with high RMSE values (Table 2.2). By accounting for meal intake, physical activity and EGP our modified Bergman Minimal Model performs slightly better in estimating the blood glucose levels as compared to the original minimal model.

We evaluate the three neural network models NN-Models I, II and III when they are not provided with any guide input. The RMSE values obtained for each of the three NN-Models for the seven patients are shown in Table 2.3. It is to be noted that the deep neural network, NN-Model III, has the lowest RMSE values for most of the

Table 2.3: RMSE for NN-models I, II and III Without Any Guide Model Input

	NN-Model I	NN-Model II	NN-Model III
P1	0.54	0.45	0.38
P2	0.36	0.44	0.45
P3	0.45	0.68	0.31
P4	0.33	0.42	0.43
P5	0.47	0.32	0.60
P6	0.66	0.54	0.26
P7	0.32	0.65	0.55

test cases. This can be attributed to the increased non-linearity of NN-Model III due to the increased number of hidden layers and thus it is able to learn the non-linearity of the underlying model better. Although, the learned representation does not result in accurate predictions as seen from Figure 2.4

On applying the four variants of our guide model to our NN-Models I, II and III the prediction RMSE value is always lower as compared to using the NN-Models without a guide. Performance of the single layer neural network NN-Model I increases significantly (lowered RMSE values) and we can see that it is able to learn the underlying representation more accurately than without the guide input. From Table 2.4, the RMSE values show that this NN-Model I outperforms the Bergman Minimal Model estimator and the guided SVR for most cases. It however has varied performance across with respect to the four guide models used. Using NN-Model II with the four guide models results in RMSE values (refer to Table 2.5) that are not always lower than those obtained for NN-Model I. Since NN-Model II is also composed of

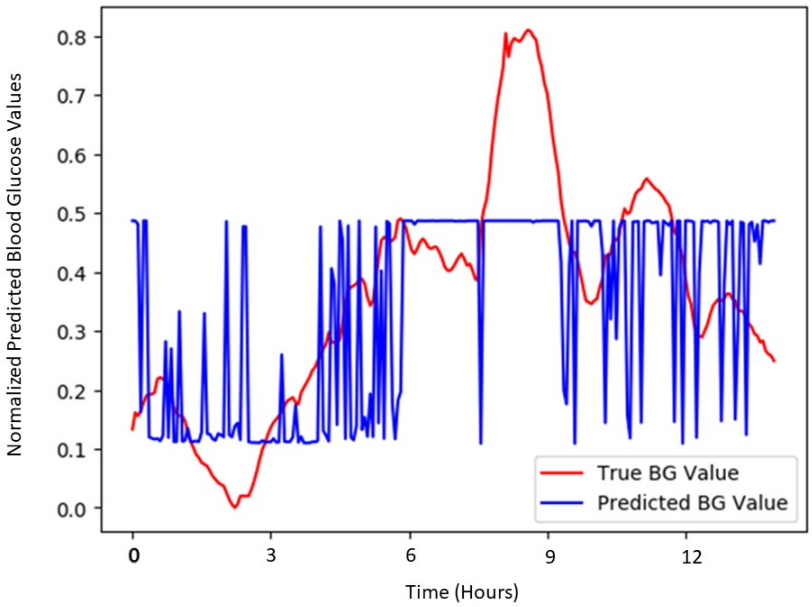


Figure 2.4: Actual and Predicted Values of Blood Glucose for Patient 5 Using NN-model III and No Guide Model Input

only a single hidden layer but with a much larger number of neurons in it, the network may be learning the insignificant parameters more than NN-Model I.

Table 2.4: RMSE for NN-Model I Using the Various Guide Models

	GM-I	GM-II	GM-III	GM-IV
P1	0.075	0.053	0.049	**0.045**
P2	0.096	0.207	**0.057**	0.067
P3	**0.017**	0.2635	0.08	0.056
P4	0.085	0.139	**0.047**	0.052
P5	0.04	0.207	**0.031**	**0.031**
P6	**0.01**	0.07	0.021	0.0114
P7	0.02	**0.018**	**0.018**	0.0218

Table 2.6 shows the RMSE values for the predictions made by the model guided SVR using the guide models from Table 2.1. The predictions made by the guided SVR also observed to be more accurate than the estimates obtained using only the physiological models. GM-IV model used as a guide to the SVR produces the lowest prediction error in most cases.

Table 2.6 also shows the RMSE values for each variant of the guide model when passed to NN-Model III for prediction. For all the cases NN-Model III when combined with GM-IV has the lowest RMSE values and can predict the blood glucose levels better than the other guide models for this type of neural network model. The decrease in the RMSE values for prediction using NN-Model III can be accounted for by the increased number of hidden layers present. The extra hidden layers allow NN-Model III to better understand the non-linearity of the physiological model used here and this when combined with the guide, NN-Model III learns the required representation better than without the guide input.

Table 2.5: RMSE for NN-Model II Using the Various Guide Models

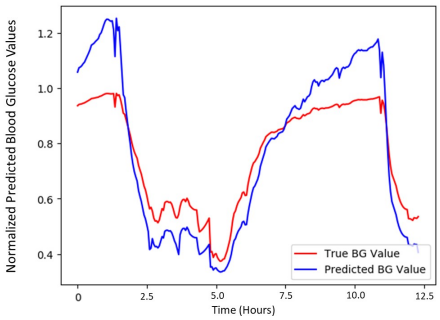
	GM-I	GM-II	GM-III	GM-IV
P1	0.1006	0.076	0.0552	**0.0511**
P2	0.1219	0.2139	0.0883	**0.0304**
P3	0.086	0.269	**0.025**	0.0539
P4	0.1313	0.1692	0.0638	**0.0610**
P5	0.0511	0.2056	0.0451	**0.0356**
P6	0.081	0.07	**0.024**	0.0273
P7	0.051	0.05	0.022	**0.021**

Table 2.6: RMSE for Model Guided SVR and Model Guided Deep Neural Network NN-Model III

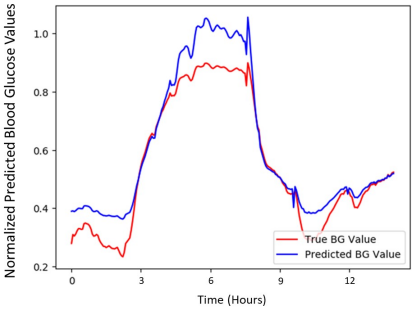
Models	Guided SVR				NN-Model III			
	GM-I	GM-II	GM-III	GM-IV	GM-I	GM-II	GM-III	GM-IV
P1	0.428	0.241	0.164	0.169	0.0176	0.017	0.01	**0.00293**
P2	0.221	0.054	0.043	0.044	0.0113	0.198	0.029	**0.0028**
P3	0.335	0.157	0.128	0.154	0.0205	0.238	0.011	**0.00909**
P4	0.417	0.275	0.256	0.249	0.0132	0.0857	0.012	**0.00879**
P5	0.369	0.174	0.162	0.167	0.0174	0.2055	0.0063	**0.0079**
P6	0.376	0.219	0.204	0.194	0.0165	0.05	0.0148	**0.0053**
P7	0.451	0.231	0.16	0.091	0.0174	0.25	0.07	**0.0019**

2.4.4 Discussion

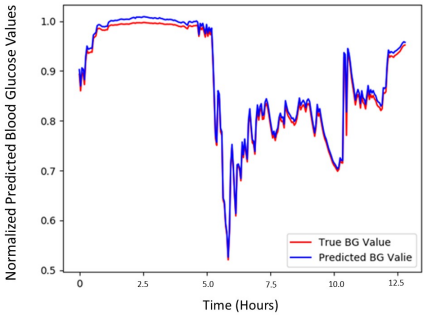
From the results shown in section 2.4.3 , we demonstrate that our proposed approach of model guided deep learning is able to significantly lower the prediction errors as compared to using only the predictive model or a free learning deep neural network. Applying our proposed approach to a closed loop glucose control system for prediction of blood glucose levels, we saw a significant improvement in the predictions made for each patient. The guide model used for our deep neural network is the modified Bergman Minimal Mode. Compared to the Bergman Minimal Model and modified Bergman Minimal Model being used individually to estimate blood glucose level for our dataset, our proposed approach has an improvement in the RMSE values by a factor of almost 100. Our proposed approach improves on the RMSE values on an average by a factor of almost 50 compared to a deep neural network without any



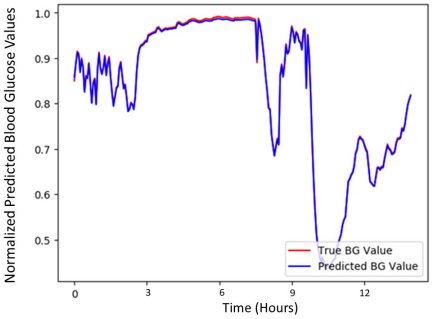
(a) Prediction Using NN-Model III and GM-I



(b) Prediction Using NN-Model III and GM-II



(c) Prediction Using NN-Model III and GM-III



(d) Prediction Using NN-Model III and GM-IV

Figure 2.5: Prediction Using the Four Different Variants of the Modified Bergman Minimal Model and NN-Model III

guide and by a factor of 32 compared to using a guided SVR approach.

For NN-Model III, the RMSE values obtained on the test sets for each patient is always much lesser when using a model guide as compared to without using the guide. Inspite of having the same architecture of the deep neural network, the model guided deep neural network makes more accurate predictions. Figure 2.5 shows the predictions made by the deep neural network NN-Model III for all for model guides (GM I-IV) for Patient 1. We observe that the most accurate predictions are made by using GM-IV i.e. our modified Bergman Minimal Model as the guide input. The quality of the predictive model used as the guide impacts the predictions made by the deep neural network. In this regard when the original physiological model (GM-I) is improved by adding the effects of EGP (GM-II) and the effects of physical

activity (GM-III) and then combining both (GM-IV) the predictions made by the model guided deep neural network become increasingly closer to the ground truth values. Therefore we can conclude that by introducing a model based guide to the free learning deep neural network we are able to significantly improve the predictions made by the deep neural network.

Chapter 3

A FRAMEWORK FOR MODEL GUIDED LONG SHORT-TERM MEMORY NEURAL NETWORKS FOR PREDICTION OF BLOOD GLUCOSE LEVELS

3.1 Problem Definition

Forecasting blood glucose levels with a high level of accuracy is an essential requirement for the artificial pancreas project to be successful. In this regard, we have seen multiple methods based on statistical and machine learning techniques being developed throughout literature. However, most of these methods are reliant only on empirical evidence i.e. observations from the glucose monitoring device while some of the more recent work also include other parameters like meal input and physical activity. For all these methods that consider the observed glucose monitor data, they do not account for the difference in the observed interstitial blood glucose from the monitoring devices and the actual blood glucose levels. It is essential to account for the difference in the interstitial and venous blood glucose levels in order to develop a model that can accurately estimate the blood glucose levels for a given period of time. In this work, we propose to build a framework for predicting blood glucose levels in T1DM patients using a model guided long short term memory (LSTM) network. Using a mathematical model that encapsulates the kinetics of diffusion of blood glucose from the bloodstream to the tissue level we incorporate the missing domain knowledge required for accurate estimation of blood glucose in a closed loop system.

3.2 Novel Framework for Model Guided Long Short-Term Memory Neural Networks with Custom Loss Function

In this section we formulate the blood glucose prediction problem as a time-series prediction problem using supervised learning, briefly discuss the modified BMM used as our model guide, discuss the LSTM architecture and how it is used for the regression task and finally present our proposed framework.

3.2.1 Time Series Problem Formulation

Blood glucose readings from CGM sensors are crucial for patients to monitor and administer insulin doses on a daily basis. CGM sensors record blood glucose levels at frequent intervals (common interval used is a recording every 5 minutes). CGM data can be modeled as time-series data of the form $(x_1, x_2, x_3, ..., x_i, x_{i+1}, ..., x_n)$ where each x_i is the CGM observation at the i-th instant. Since sequential time-series data can be modeled

Most LSTM based prediction solutions only consider this single CGM time-series for the blood glucose prediction task. While some may add additional features like meal information, physical activity (exercise, daily step count, etc.), and bolus insulin, in this work we consider the following as input features -

- CGM sensor reading: BG^{actual}
- carbohydrate intake information: C^{intake}
- physical activity by means of accelerometer readings: $Acc^{physical}$
- bolus insulin injected: I^{bolus}
- lastly the estimate of blood glucose levels using the modified BMM (model guide): $BG^{modelguide}$

Since an observation is made for each of the aforementioned features at every i-th timestep, we formulate each sample in our dataset as a multi-variate time-series sample of the form $(BG_i^{actual}, Acc_i^{physical}, I_i^{bolus}, C_i^{intake}, BG_i^{modelguide})$. However, this sequential time-series data must be converted to an input-output pair form such that any j-th sample X_j has a ground truth label or value y_j associated with it for the task of learning. We convert our multi-variate time-series dataset into a multi-dimensional input-output sequence. Using this newly formed dataset from our multi-variate time-series data, we apply our proposed MG-LSTM approach for blood glucose prediction.

3.2.2 Modified Bergman Minimal Model

In order to model the effects of insulin sensitivity and pancreatic responsiveness in diabetic patients, the Bergman minimal model was developed. This is a two-compartment model that is used to quantify both pancreatic responsiveness and insulin sensitivity for a given subject. Pancreatic responsiveness can be described as the ability of the pancreatic β-cells to respond to and dispose of glucose in the blood. Insulin sensitivity refers to how sensitive the body's cells are in response to insulin. A higher insulin sensitivity results in a more efficient utilization of blood glucose.

According to the two compartment minimal model, the glucose kinetics model is described using (3.1) and (3.2) and the insulin sensitivity is calculated as $S_I = -P_3/P_2$ and its unit is $min^{-1}/\mu U$ per ml. The definition of X and the remaining parameters are described in Bergman *et al.* (1979).

$$\frac{dG(t)}{dt} = (P_1 - X)G(t) - P_1 G_b, \qquad G(t_0) = G_0, \tag{3.1}$$

$$\frac{dX(t)}{dt} = P_2 X(t) + P_3 I(t), \qquad X(t_0) = 0, \tag{3.2}$$

where:

- t is the independent model variable time in minutes,
- t_0 is the time of injection of external glucose,
- *G(t)* is the concentration of plasma glucose,
- *I(t)* is the plasma insulin level, and
- *X(t)* is the interstitial insulin activity.

The model of insulin kinetics that is utilized for the calculation of the pancreatic responsivity parameters is given by (3.3) as

$$\frac{dI(t)}{dt} = \gamma(G(t) - h)t - nI(t) \tag{3.3}$$

However, the regulation of blood glucose is not limited to the effects of insulin and the amount of carbohydrates consumed by individuals. Other physiological factors like endogenous glucose production (EGP) Young (2005) and the amount of physical activity Yardley *et al.* (2012) undergone by an individual also play an important role. In this work, we utilize the modified Bergman minimal model found in Agrawal (2017); Das *et al.* (2017). This modified BMM accounts for the relation of insulin sensitivity and endogenous glucose production Hovorka *et al.* (2004),the effects of physical activity Colberg *et al.* (2015) and the meal intake sub-model Gillis *et al.* (2007). The original BMM (3.1), (3.2) and (3.3) are modified as shown in (3.4), (3.5) and (3.6) respectively.

$$\frac{dG(t)}{dt} = -(P_1 + PA)(G(t) - G_b) - X(t)G(t) + \frac{m(t) + EGP_0(1 - X(t)}{V_g} \tag{3.4}$$

$$\frac{dX(t)}{dt} = P_2X(t) + P_3I(t) \tag{3.5}$$

$$\frac{dI(t)}{dt} = \begin{cases} \gamma(G(t) - h)t - n(I(t) + I_b) + \frac{U(t)}{V_I}, if G(t) \geq h \\ -n(I(t) + I_b) + \frac{U(t)}{V_I}, otherwise \end{cases} \tag{3.6}$$

where:

- $G(t)(mg/dl)$ is the relative differential plasma glucose,
- $G_b(mg/dl)$ is the basal glucose,
- X(unitless) represents the remote effects of insulin on glucose distribution and endogenous glucose production,
- $I(t)(\mu I/dl)$ is the blood insulin concentration,
- $I_b(\mu I/dl$ is the basal insulin concentration,
- $P_1(min^{-1})$ is the glucose "mass action" rate constant,
- $P_2(min^{-1})$ is the rate constant expressing the spontaneous decrease of tissue glucose uptake ability,
- $P_3(min^{-1})$ is the insulin-dependent increase in tissue glucose uptake ability per unit of insulin concentration excess over baseline insulin,
- $n(min^{-1})$ is the first order decay rate constant for insulin in plasma,
- $\gamma(min^{-1})$is the rate of pancreatic release of insulin after the bolus, per minute and per mg/dl of glucose concentration above the target glycemia,
- $h(mg/dl)$ is the pancreatic "target glycemia",
- $V_I(dl/kg)$ is the glucose volume distribution,
- $V_G(dl/kg)$ is the insulin volume distribution,

- $m(t)(mg)$ is the effect of carbohydrate intake on plasma glucose,
- $EGP_0(mg/dl)$ represents endogenous glucose production extrapolated to the zero insulin concentration, and
- $U(t)(\mu I/dl)$ is the modification for T1DM in which insulin appears only from an exogenous source.

3.2.3 Long Short-Term Memory Neural Network

General ANNs are capable function approximators but the flow of information is unidirectional i.e. from the input layer to the output layer while travelling through the hidden layers. This implies that the current computed output of the network is not considered as an input while calculating the output of the next sample. However, in RNNs there is a directional cycle in the process of transmission of information. This cyclic flow of information allows the RNN to *remember* the previous output while computing the output of the current sample, thereby bringing in the concept of *memory*. However, RNNs do suffer from gradient disappearance and gradient explosion Rather *et al.* (2015). To overcome this problem, Long Short Term Memory (LSTM) networks were developed Hochreiter and Schmidhuber (1997). LSTM networks are a special kind of RNN that are capable of learning long-term dependencies. They were designed to avoid the disappearing and exploding gradient problems that are faced by RNNs. Like RNNs, LSTMs also have a chain of repeating modules of neural network. The repeating module (shown in Figure 3.1) in an LSTM comprises of four neural network layers which interact in a special way.

The key to LSTMs is the cell state C which runs through the entire cell with minor modifications made to it through linear interactions. This cell state is responsible for *remembering* (addition of past state information) and *forgetting* (removal of past state

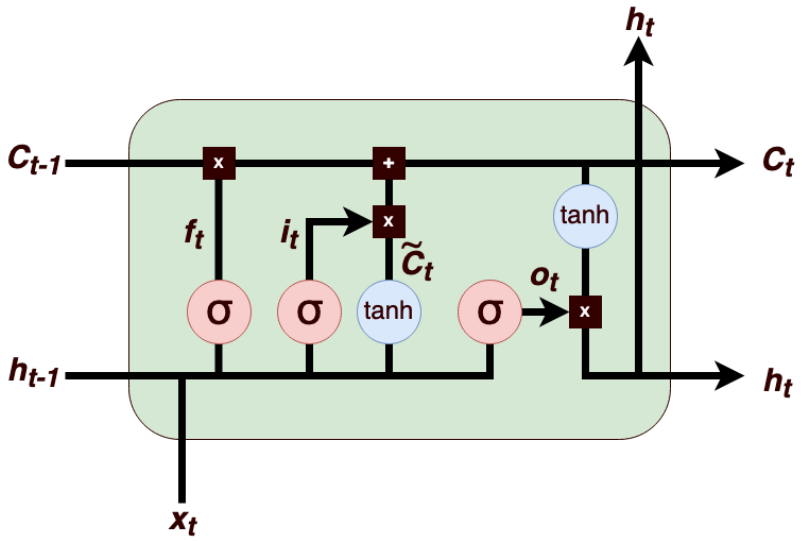


Figure 3.1: Long Short-Term Memory (Lstm) Unit Displaying the Four Neural Network Layers Used for It's Memory Capability

information) handled by the presence of various gates within the cell. Each gate is comprised of a sigmoid neural network layer and a pointwise multiplication operation. There are three such gates in every LSTM cell - input gate, output gate and forget gates.

If the input sequence is given by $(x_1, x_2, ..., x_T)$ and the hidden layer representations as $(h_1, h_2, ..., h_T)$, then at time t:

$$f_t = \sigma(W_f \cdot [h_{(t-1)}, x_t] + b_f) \tag{3.7}$$

$$i_t = \sigma(W_i \cdot [h_{(t-1)}, x_t] + b_i) \tag{3.8}$$

$$\tilde{C}_t = tanh(W_C \cdot [h_{(t-1)}, x_t] + b_C) \tag{3.9}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{3.10}$$

$$o_t = \sigma(W_o \cdot [h_{(t-1)}, x_t] + b_o) \tag{3.11}$$

$$h_t = o_t * tanh(C_t) \tag{3.12}$$

where, i_t, o_t, f_t are the input gate, output gate and forget gate layer respectively, C_t is the cell state at the given instant t, W_f, W_i, W_C and W_o are the weights of the forget, input, cell state and output layer respectively, b_f, b_i, b_C and b_o are the bias values for each of the layers respectively, $\sigma(\cdot)$ is the sigmoid function and $tanh(\cdot)$ is the tan hyperbolic function.

In every LSTM, the first step is to decide how much information must be discarded or remembered from the previous cell state. This is done by means of the forget gate (f_t) layer (3.7). By comparing the previous hidden layer information h_{t-1} and the current sample x_t, the forget gate layer outputs a number between 0 and 1 (by means of the sigmoid function) for each value present in the cell state C_{t-1}. Next, the LSTM decides on what new information it will store in the cell state - the input gate layer (3.8) decides which values need to be updated while a $tanh$ layer (3.9) creates a candidate vector $\tilde{C}_t$ to be added to the current cell state C_t using (3.10). Lastly, the output of the LSTM cell is generated using a sigmoid layer (3.11), which decides what parts of the cell state must be sent out, multiplied by running the cell state C_t through a $tanh$ function as shown in (3.12). This output h_t is transmitted to the next LSTM cell and the process is repeated in the next LSTM cell. By utilizing these four special neural network layers in each neuron, LSTM networks are capable of remembering long sequences of past data.

In order to utilize the LSTM network for the task of prediction, we add a fully connected layer at the end of our LSTM network to replicate the task of linear regression (3.13).

$$y_t = W_y * h_t + b_y \tag{3.13}$$

where y_t is the prediction made at time t, W_y is the weight matrix associated with the output of the LSTM network h_t and b_y is the bias/threshold of the linear regression layer.

Table 3.1: Description of Configurations of the Proposed Model Guided LSTM Neural Network

Configuration	**Description of Configured Neural Network Framework**	**Model Guide Used**
NG-LSTM	Stacked LSTM with no model guide as input feature and no model guided loss	None
MG-LSTM I	Stacked LSTM with model guide as input feature only	Modified BMM
MG-LSTM II	Stacked LSTM with model guide as input feature and model guided loss function	Modified BMM
MG-LSTM III	Stacked LSTM with hypothetical model guide as input feature and model guided loss function	Hypothetical Model Guide

3.2.4 Custom Loss Function

For neural networks, the objective or loss function plays an important role in determining how well the neural network performs. Each neural network comprises of an input layer, several hidden layers and an output layer. Data samples (X_i) are fed as inputs to the network through the input layer followed by a series of multiplications (with layer wise weights denoted by W_{layer}) and layer wise activation through the hidden layers to get an output ($\hat{y}$) through the output layer. The neural network output is then compared to the ground truth value (y) corresponding to each sample and the difference between the two is computed as the error term (3.15). This computed error term provides us with a metric to evaluate the performance of a model. The error term is then optimized by means of backpropagation, which modifies the weights of each layer based on the error term.

$$\hat{y_i} = f(X_i, W_{layer}, b_{layer}) \tag{3.14}$$

$$error = y - \hat{y} \tag{3.15}$$

The choice of the error metric or loss function is application dependent. One of the most commonly used loss functions for the task of regression is Mean Squared Error (MSE) (also called the L_2 norm)(3.16)

$$L_{emp} = MSE = \frac{1}{N}\sqrt{\sum_{i=1}^{N}(y_i - \hat{y}_i)} \tag{3.16}$$

where for each $i - th$ sample, y_i is the ground truth or actual value, $\hat{y}_i$ is the estimated output from the neural network and N is the total number of samples present in the data. MSE is sensitive towards outliers and therefore penalizes outliers to a greater degree. Blood glucose levels change in an abrupt manner depending on certain factors like meal intake, amount of physical activity and EGP. These sudden changes may result in the creation of outlier samples (sudden spikes or drops) in data which in turn will be penalized heavily by a loss function such as MSE. In order to correct this issue and with the purpose of embedding model knowledge into the neural network we propose to use a loss function based on correcting the model using our proposed model guide estimate in section 3.2. Our proposed loss function L_{mg} (3.17) is based on using MSE between the model guide estimate y_i^{mg} and the output of the neural network $\hat{y}_i$

$$L_{mg} = \frac{1}{N}\sqrt{\sum_{i=1}^{N}(y_i^{mg} - \hat{y}_i)} \tag{3.17}$$

By combining the loss calculated from empirical evidence (3.16) and the loss calculated from the physiological model guide (3.17) in a weighted manner we get our custom loss function L as

$$L = \alpha_1 * L_{emp} + (1 - \alpha_1) * L_{mg} \tag{3.18}$$

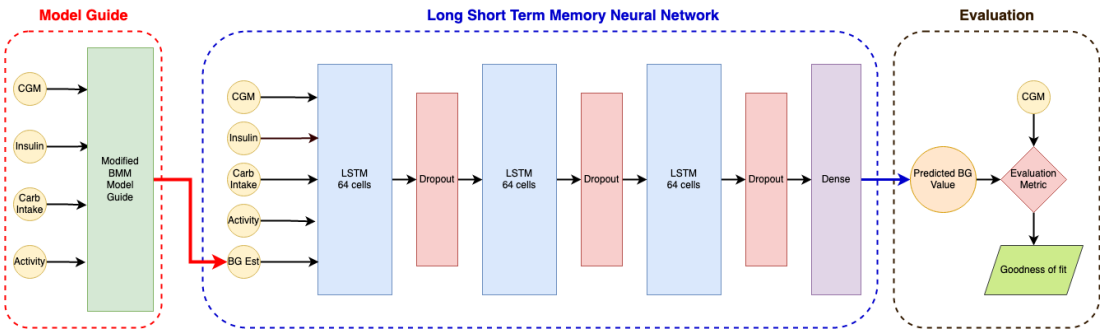


Figure 3.2: Stacked LSTM Model Framework

where α_1 and $(1 - \alpha_1)$ are weighting factors that control the priority given to empirical loss and physiological model based loss respectively. The values of both are determined based on multiple experimental runs combined with the efficiency of the model guide.

3.2.5 Model Guided LSTM

In this work, we propose to implement a *model* guided LSTM based neural network (Figure 3.2) to predict changes in blood glucose levels in Type 1 Diabetes Mellitus patients. We propose a LSTM neural network that is capable of learning from the evidence while incorporating aspects of the model that describes the physiological process of interest. In this work the model guide can be applied to the neural network in two aspects - as an input feature and as part of the neural networks loss function. For the task of glucose prediction we use the modified Bergman minimal model as discussed in Section 3.2.2 as our guide model. The blood glucose estimate generated by the modified BMM is used as an input feature. We also use this model guide estimate as part of the loss function as discussed in Section 3.2.4. We show that by using domain knowledge in conjunction with the representation learned by a neural network, we are able to improve the prediction accuracy of the overall network and fine tune the network to each individual patient.

3.3 Evaluation Metrics

Evaluation metrics chosen for this work are the root mean squared error (RMSE) and adjusted R^2 value. RMSE is widely used in literature as a good measure of accuracy to compare prediction errors of either different models or model configurations. Adjusted R^2 on the other hand determines the usefulness of adding additional input features by assessing their impact on the output variable. In order for a model to be considered a good predictor for blood glucose, the RMSE of the predicted output when compared to the ground truth must be relatively small. At the same time, addition of the model guide feature must reflect an increase in the adjusted R^2 value. This would aid in the process of understanding the impact of adding the model guide to the learning ability of a neural network. We discuss each of the chosen metrics below.

3.3.1 Root Mean Square Error

While fitting a regression model, we want to evaluate how well our model "fits" the given data. Root mean square error (RMSE) is a commonly used metric that tells us how far apart the predicted values are from the ground truth, on an average. The lower the value of RMSE, the better the model "fits" the corresponding dataset.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)} \tag{3.19}$$

We calculate RMSE for each i-th observation as shown in (3.19) using the ground truth value $y_i = BG_i^{actual}$ and the output of our proposed model guided LSTM network $\hat{y}_i = BG_i^{predicted}$.

3.3.2 *Adjusted R^2*

While RMSE values tell us how well a model "fits" the corresponding dataset by measuring the distance between the predicted value made by our model and the ground truth, we use a second evaluation metric - adjusted R^2 which shows the degree to which the input/predictor variables are able to explain the variation in the output/predicted variable for a given regression model.

The R^2 metric (3.20) informs us about how well the independent/predictor variables of our model are able to explain the variation in the dependent/predicted variable. This metric takes values between 0 and 1 - the higher the R^2 value the better the model fits the data.

$$R^2 = 1 - \frac{\sum(y_i - \hat{y_i})}{\sum(y_i - \bar{y})} \tag{3.20}$$

where for every i-th observation,

- y_i is the ground truth value
- $\hat{y_i}$ is the predicted value
- $\bar{y}$ is the mean of the observed values

The R^2 value has one shortcoming - increasing the number of independent variables will always cause the overall R^2 value to increase. Therefore the addition of independent variables to the regression model will result in a higher R^2 value implying that if the variables added are redundant to the model, we would still get a high value of R^2, which is not ideal or explainable. To correct this issue, we use the adjusted R^2 (3.21) that takes into account the number of independent variables used to predict in a model. This way it allows us to determine whether the addition of new variables is actually useful while fitting the model.

$$Adjusted R^2 = 1 - \frac{(1 - R^2)(N - 1)}{(n - k - 1)} \tag{3.21}$$

where,

- R^2 is calculated using (3.20)
- N is the total number of observations used
- k is the number of independent variables used

Adjusted R^2 values range between -1 to 1 indicating the goodness of fit between the independent variables and the predicted variable. Higher positive values of adjusted R^2 imply that the independent variables used in the model are able to explain the variation in the predicted variables to a greater degree.

3.4 Dataset, Results and Discussion

In this section we discuss the the BG forecasting accuracy of our proposed MG-LSTM model framework for our dataset using various train to test splits of the data. We also discuss the impact of the model guide and the discrepancies observed in the dataset.

3.4.1 Dataset and Preprocessing

The variation of blood glucose levels is specific to each individual based on their pancreatic β-cell responsiveness and insulin sensitivity along with other factors. Our dataset comprises of data collected from eight T1DM patients in a controlled environment over a period of seven days. Each patient was outfitted with a CGM device, an insulin delivery pump along with a 16G accelerometer device to record physical activity. We recorded the CGM reading and accelerometer reading every five minutes.

Meal intake was recorded in terms of the amount of carbohydrates consumed while insulin sensitivity, basal and bolus insulin, EGP rates were computed on a per patient basis. The collected data was cleaned and preprocessed prior to being used for the purposes of training and testing. The cleaning process was done to ensure that our data is free from noisy readings, artifacts and erroneous readings. For the purposes of training the neural network, the dataset was preprocessed and standardized to zero mean and unity variance to ensure a uniform scale across all the collected features and for faster convergence.

3.4.2 Experimental Setup

Since blood glucose sequences exhibit characteristics of strong non-linearity, randomness and are non-stationary by nature, the application of general prediction methods is difficult. To this effect we use our proposed model guided LSTM based network to accurately predict blood glucose levels in T1DM patients within a prediction horizon (pH) of thirty minutes. The overall process from data acquisition to prediction for each patient is described as follows:

1. Acquire patient data using CGM, accelerometer, meal information and insulin injected. Clean the data and standardize to zero mean and unity variance prior to splitting for training, validation and testing.

2. Use the acquired patient data to learn modified BMM parameters and estimate the model guide values using the modified BMM.

3. Using a 60:20:20 train to validation to test split, we train our MG-LSTM network and use the validation split to fine-tune network hyper-parameters. Model performance is then tested using the test split and the RMSE and adjusted R^2 values are computed.

4. Using a 40:20:40 train to validation to test split, we train our MG-LSTM network and use the validation split to fine-tune network hyper-parameters. Model performance is then tested using the test split and the RMSE and adjusted R^2 values are computed.

5. We compare the performance of using a 60:10:30 training to validation to test split and using a 40:20:40 split of the dataset using the MG-LSTM model.

The proposed model guided stacked LSTM framework was implemented using the Google Colab Research Environment and a desktop computer. Specifications of the runtime environment used in the Google Colab Research environment are listed below:

- Intel Xeon CPU (base clock 2.20GHz)
- 16 GB RAM
- NVIDIA Tesla K80 GPU with 24GB VRAM
- Python 3.7.13

Specifications of the desktop computer used for preprocessing the dataset and generating the modified Bergman minimal model estimate are listed below:

- Intel Core i7 4790 4 core CPU (base clock 3.60GHz)
- 32GB RAM
- NVIDIA GTX 745 GPU with 4GB VRAM
- MATLAB R2019a

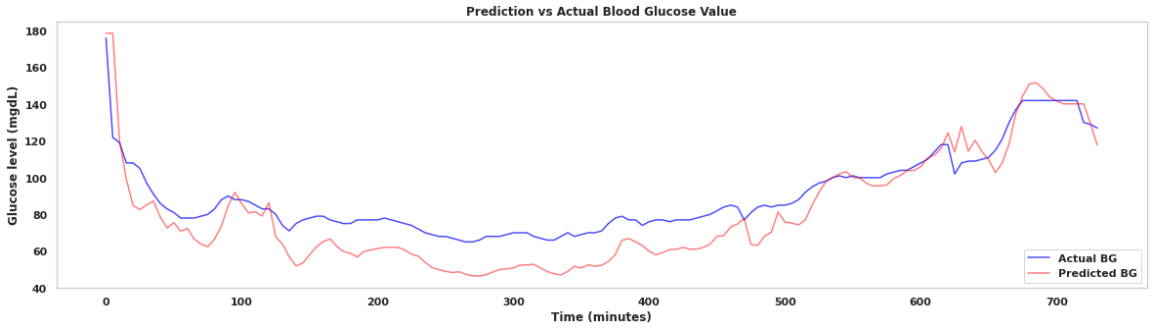
3.4.3 MG-LSTM Configurations

In this work, we configure and use two variations of our proposed MG-LSTM framework utilizing the modified BMM model (as discussed in Section 3.2.2) as our guide model. We also configure a third variation of the proposed MG-LSTM framework using a hypothetical guide model. The details for each of the configurations is listed in Table. 3.1.

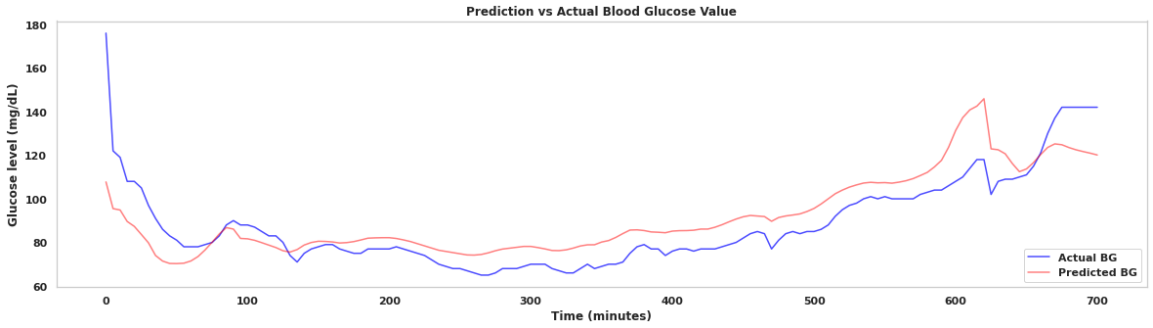
Each variation of our proposed MG-LSTM framework and NG-LSTM solution was set to run for 500 epochs with an early stopping criteria of patience level 20. The batch size was set to 16, learning rate to 0.0001 using the root mean squared propagation optimizer. We note that for each of the configurations used convergence of the proposed framework was achieved in less than 200 epochs consistently.

3.4.4 Results for 60:20:20 Data Split

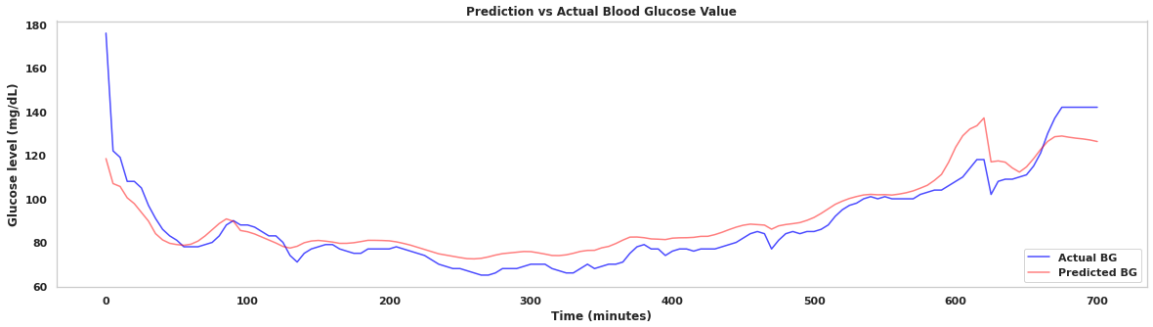
Traditionally a 60:40 train to test split ratio is used for the task of learning models. Each patient's data is split using a 60:20:20 percent split ratio for training, validation and testing respectively. We set the prediction horizon (pH) to 30 minutes for all of our experiments. This is done to closely mimic the prediction behavior of real-time CGM devices used by patients. All four configurations of our proposed model (Table 3.1) are trained and tested using this data split. The experimental results using this data split and our proposed models are summarized in Table. 3.2. The average time to train each configuration using this data split is around 65 ± 5 seconds while the average inference time is around 0.3 ± 0.15 seconds. Results from this table demonstrate that our proposed model achieves significantly better forecasting performance across all cases as opposed to using no model guide for forecasting BG levels.



(a) NG-LSTM Performance



(b) MG-LSTM I Performance



(c) MG-LSTM II Performance

Figure 3.3: Performance of Model Guided LSTM in Predicting Blood Glucose Levels in Patient P5 for the next 700 Minutes (Approximately 11.5 Hours); (a) Using Configurations NG-LSTM (b) Using Configuration MgGLSTM I and (c) Using Configuration MG-LSTM II

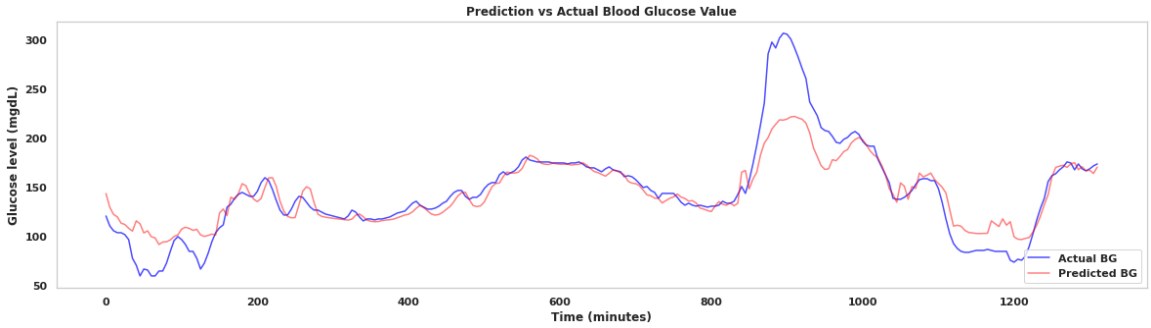
Using this data set split, without a model guide, the stacked LSTM (NG-LSTM) has RMSE values in the range of 18.96 to 39.09 mg/dL while the average RMSE is 28.745 mg/dL across all cases. On application of the model guided MG-LSTM in

Table 3.2: Comparison of RMSE of the Various Configurations of the Proposed Model Guided LSTM Neural Network Framework Using 60:40 Training to Testing Data Split

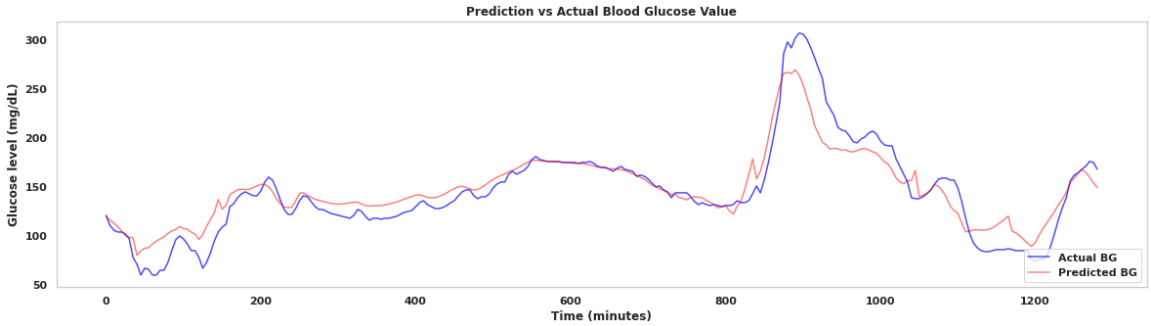
	NG-LSTM		**MG-LSTM I**		**MG-LSTM II**		**MG-LSTM III**	
Patient	**RMSE**	**Adjusted R^2**	**RMSE**	**Adjusted R^2**	**RMSE**	**Adjusted R^2**	**RMSE**	**Adjusted R^2**
P1	33.909	0.839	23.796	0.822	31.875	0.760	23.800	0.800
P2	32.880	0.796	24.612	0.754	31.172	0.765	21.890	0.826
P3	29.200	0.368	17.380	0.610	14.452	0.640	17.160	0.710
P4	25.507	0.739	16.584	0.508	21.445	0.383	15.510	0.551
P5	18.960	0.197	13.476	0.620	9.860	0.633	6.720	0.760
P6	28.705	0.629	23.235	0.684	20.831	0.711	20.950	0.721
P7	33.736	0.889	23.793	0.790	27.105	0.776	23.476	0.790
P8	26.038	0.766	24.564	0.538	26.862	0.485	21.230	0.548

configuration MG-LSTM I, we observe that RMSE for each patient shows significant decrease in RMSE values (ranges between 13.48 to 23.79 mg/dL) and the average RMSE across all patients is reduced to 20.828 mg/dL. Using MG-LSTM II, model guide as an input and in the loss function, we observe that the RMSE ranges between 9.86 to 31.75 mg/dL and the average RMSE increases slightly to 22.95 mg/dL.

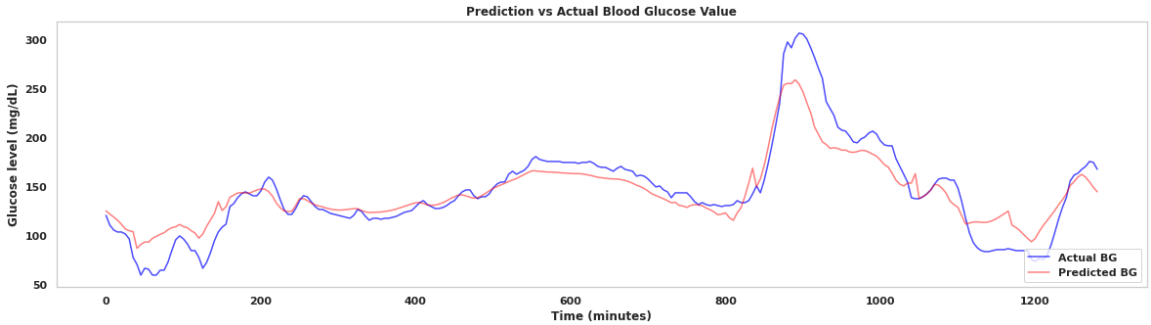
The proposed model's forecasting performance using pH=30 minutes is illustrated in Figure 3.3 for patient P5 using configurations NG-LSTM (Figure 3.3a), MG-LSTM I (Figure 3.3b) and MG-LSTM II (Figure 3.3c). While the introduction of the model guide as an input feature decreases the error between the predicted BG values and the ground truth, introduction of the model guide into the loss function further enhances this effect.



(a) NG-LSTM Performance



(b) MG-LSTM I Performance



(c) MG-LSTM II Performance

Figure 3.4: Performance of Model Guided LSTM in Predicting Blood Glucose Levels in Patient P8 for the next 1200 Minutes (Approximately 20 Hours); (a) Using Configurations NG-LSTM (b) Using Configuration MgGLSTM I and (c) Using Configuration MG-LSTM II

3.4.5 Results for 40:20:40 Data Split

Aside from using the standard 60:40 split for training to testing data, we investigate the effects of using smaller amounts of training data using our proposed model.

Table 3.3: Comparison of RMSE of the Various Configurations of the Proposed Model Guided LSTM Neural Network Framework Using 40:60 Training to Testing Data Split

	NG-LSTM		**MG-LSTM I**		**MG-LSTM II**		**MG-LSTM III**	
Patient	**RMSE**	**Adjusted R^2**	**RMSE**	**Adjusted R^2**	**RMSE**	**Adjusted R^2**	**RMSE**	**Adjusted R^2**
P1	26.190	0.854	26.400	0.750	31.310	0.710	26.450	0.747
P2	25.480	0.810	23.120	0.790	21.120	0.760	25.120	0.710
P3	24.060	0.700	34.150	0.320	28.790	0.440	34.150	0.310
P4	24.820	0.860	19.343	0.720	24.240	0.630	19.340	0.720
P5	46.690	0.682	58.280	0.433	39.590	0.690	58.280	0.433
P6	36.930	0.492	27.586	0.510	26.940	0.490	27.586	0.510
P7	46.683	0.796	28.890	0.770	38.043	0.720	28.990	0.770
P8	28.573	0.697	22.916	0.580	23.073	0.590	19.790	0.580

The training data split is reduced to 40% of the total available data per patient. The validation split is kept at 20% while the testing set is increased to 40%. The results of applying all configurations of our proposed model on this data split are summarized in Table 3.3. Results from this table indicate that reducing the amount of training data does not affect the overall performance of our proposed model. The average time to train each configuration using this data split is around 52 ± 5 seconds while the average inference time is around 0.3 ± 0.15 seconds.

We observe that using a smaller training set, the RMSE ranges between 24.06 - 46.69 mg/dL using NG-LSTM (no model guide) and the average RMSE is 32.07 mg/dL across all cases. In comparison, using the modified BMM as a model guide (MG-LSTM II) results in RMSE values between 21.12 - 39.59 mg/dL and the average RMSE reduces to 29.014 mg/dL across all patients. From the results observed in this section, we note that our proposed MG-LSTM framework is capable of forecasting

Table 3.4: Comparison of RMSE Values Using Configuration NG-LSTM with 60:40 Train to Test Split and Configuration MG-LSTM II with 40:60 Training to Test Data Split

	60% Training Data		**40% Training Data**	
	NG-LSTM		**MG-LSTM II**	
Patient	**RMSE**	**Adjusted R^2**	**RMSE**	**Adjusted R^2**
P1	33.909	0.839	31.310	0.710
P2	33.909	0.839	21.120	0.760
P3	29.200	0.368	28.790	0.440
P4	25.507	0.739	24.240	0.630
P5	18.960	0.197	39.590	0.690
P6	28.705	0.629	26.940	0.490
P7	33.736	0.889	38.043	0.720
P8	26.038	0.766	23.073	0.590

BG levels with limited training data. The forecasting performance of our proposed model using a 40:20:40 data split with pH=30 minutes for Patient P8 is illustrated in Figure 3.4.

Table. 3.4 shows the performance of NG-LSTM using 60% training data compared to MG-LSTM II using 40% training data. Using these results and apart from a couple of outliers (patients P5 and P7), we demonstrate that our proposed model achieves lower prediction error while using a reduced amount of data for training. This makes such a framework suitable for use in conditions where the model does not have a large amount of data to learn from.

Table 3.5: Comparison with Existing Methods for Prediction of Blood Glucose

Publication	Methods Used	Features	Evaluation Metric	Prediction Accuracy
Kok (2004)	ANN	CGM	RMSE	2.7055 (Normalized)
Zainuddin *et al.* (2009)	WNN	CGM, Meal, Insulin, Physical Activity	RMSE	0.0504 (Normalized)
Pérez-Gandía *et al.* (2010)	ANN	CGM	RMSE	19.51 ± 5 mg/dL
Pappada *et al.* (2011)	ANN	CGM, Meal, Insulin, Physical Activity	RMSE	43.9 ± 6.5 mg/dL
Sparacino *et al.* (2007)	ARIMA	CGM	MSPE	18.78 mg/dL
Yang *et al.* (2018)	ARIMA	CGM	RAD	141.3mg/dL
Georga *et al.* (2012)	SVR	CGM, Meal, Insulin, Physical Activity	RMSE	16.23 ± 3.87 mg/dL
Bunescu *et al.* (2013)	SVR	CGM, Meal, Insulin, Physical Activity	RMSE	19.5 mg/dL
Mhaskar *et al.* (2017)	Deep ANN	CGM	Accuracy	77.97%
Gu *et al.* (2017)	LSTM	CGM	RMSE	14.04 mg/dL
Zhu *et al.* (2018)	Deep CNN	CGM	RMSE	21.73 ± 2.52 mg/dL
Faruqui *et al.* (2019)	LSTM	CGM, Weight, Food, Physical Activity	Accuracy	84.12%
Munoz-Organero (2020)	LSTM	CGM, Insulin, Food	RMSE	6.42 mg/dL
Li *et al.* (2020)	ESN	CGM	RMSE	23.57mg/dL
Martinsson *et al.* (2018)	RNN-LSTM	CGM	RMSE	20.1 mg/dL
Sun *et al.* (2018)	RNN-LSTM	CGM	RMSE	21.7 mg/dL
Proposed Work	**MG-LSTM**	**CGM, Weight, Food, Physical Activity, Model Guide**	**RMSE**	**13.48 mg/dL**

3.4.6 Discussion on Model Guide and Adjusted R^2 Value

Our proposed MG-LSTM framework improves the forecasting accuracy as compared to using only a stacked LSTM framework. However, we observe some discrepancies in the adjusted R^2 value across all experiments performed. As discussed in section 3.3.2, the adjusted R^2 value informs us about the variation of the predicted variable based on the addition of predictor variables. Addition of the model guide as an input feature causes an increase of the adjusted R^2 value for most patients using our proposed model. However, for some patients we note a decrease in this value. This decrease in the adjusted R^2 value can be attributed to the model being used to guide the LSTM network. For this work the modified BMM Agrawal (2017); Das *et al.* (2017) was used as the model guide for MG-LSTM I and MG-LSTM II. Since the BMM uses interstitial blood glucose levels to estimate the intravenous blood glucose levels, we hypothesize that using a more accurate physiological model would

further improve the performance of our proposed model.

We investigate the effects of using a hypothetical physiological model which can estimate BG levels with low estimation error i.e. the model estimate is almost comparable to the ground truth. The BG estimate from our hypothesized physiological model is obtained using (3.22)

$$BG^{HypEst} = BG^{actual} + Noise \tag{3.22}$$

where $Noise$ is generated from $\mathcal{N}(0, 3.5)$. Using BG^{HypEst} as the model guide input for our proposed model, we observe in Table. 3.2 and Table 3.3, a significant drop in RMSE across all patients using both data splits. We also note an increase in the adjusted R^2 value at the same time. Switching out the modified Bergman minimal model for a more accurate hypothetical model causes an increase in the adjusted R^2 value for most patients for both data splits. This observation shows that the physiological model chosen as the guide for the framework is an integral part to achieving a solution that has high predictive accuracy.

3.4.7 Discussion on RMSE

In this work, the evaluation metric root mean squared error or RMSE was used to evaluate the predictive accuracy of the proposed framework. RMSE is used as an evaluation metric for most regression based prediction solutions since it depicts the deviation of the prediction from the ground truth values. We do note that our proposed framework when applied to challenging problems such as the prediction of blood glucose in diabetes mellitus patients require a few additional considerations. Blood glucose levels can be demarcated as normal (between 80-120 mg/dL), hypoglycemic (below 80 mg/dL) and hyperglycemic (above 180 mg/dL). The presence of three separate levels found in the measurement of a single variable can lead to some

discrepancies while designing a predictive solution for it. To account for each level of blood glucose during the task of prediction, we suggest minor modifications to the evaluation metric as a future work.

Instead of evaluating the RMSE metric using the ground truth and the predicted values, we suggest the inclusion of conditional RMSE evaluation based on the predicted level of blood glucose. This will allow the inclusion of hyperglycemic and hypoglycemic regions to be evaluated alongside the normal blood glucose levels. As a result, the framework would not only learn the normal levels but also the regions of hyperglycemia and hypoglycemia causing lower values of RMSE and enhancing the accuracy of prediction even further. Alongside an increase in prediction accuracy, this change in the evaluation metric will increase the explainability of the model guided framework as applied to the task of blood glucose prediction. Predicting blood glucose levels and being able to provide feedback on the predictions made transforms a simple predictive solution into an explainable solution.

3.4.8 Discussion on Time Invariant Physical Model Guide

The physical numerical model that is chosen as the model guide for our proposed framework plays has a significant impact on the overall predictive accuracy as shown in Section 3.4.6. The application of our proposed framework to the prediction of blood glucose levels in diabetes mellitus patients highlights the requirement for consideration of the time variant nature of the physical model.

Blood glucose levels change over time - meal consumption, physical activity, external insulin dose are some of the factors that play a major role in its variance. However, the physical model used for the proposed solution in its current state of implementation does not account for instantaneous changes as they occur. While training the proposed model guided framework, it is feasible to compute the model

guide estimate prior to being given as input to the stacked LSTM network. However during the testing or inferencing period being able to compute the model guide estimate instantaneously would be beneficial since it would reflect the current state of the physical system closely. In our current state of implementation, the model guide is pre-computed even during the testing phase. This is primarily to save on the time required to generate each point estimate instantaneously. The modified Bergman minimal model is a set of coupled ordinary differential equations with a large number of patient specific time-varying parameters. Computing these time-varying parameters on an instantaneous basis is not tractable and as a result computing the model guide estimate is not feasible in real time. However, relaxation of constraint to recompute patient specific time varying parameters could lead to a simplification in the computation of the modified Bergman minimal model estimate over time. It is to be noted that the computation of the model guide estimate during model inferencing is prone to the inclusion of time lag in the predictions made. Since the LSTM network is dependent on the model guide as an input feature, any delays in the computation of the model guide will directly affect the time taken for the next prediction to be made.

To avoid this shortcoming, this work pre-computes the model guide estimate for the inference phase. This assumption is made to simulate the environment in which this proposed framework is expected to work - model predictive controller systems. These systems are based on low power hardware and use simplifying assumptions to linearize models so that they can be evaluated in real time. However, using a neural network based solution like ours, we intend to achieve higher levels of predictive accuracy while preserving the non-linear properties of the physical model. The implementation of a real time evaluation of the model guide and its impact on the overall performance of the proposed framework is suggested future work.

Chapter 4

BRADYCARDIA PREDICTION IN PRETERM INFANTS USING NONPARAMETRIC KERNEL DENSITY ESTIMATION

4.1 Problem Definition

Prediction of the onset of bradycardia in preterm infants is crucial to the process of administering timely care to prevent any fatal outcomes from occurring. Although various techniques exist to detect and alert health care experts prior to the onset of bradycardia events they are unable to be accurate and avoid false alarms at the same time. In this paper we propose a novel method to predict the onset of near-term bradycardia in preterm infants without prior knowledge while increasing the prediction accuracy and lowering the number of false alarms overall.

4.2 Novel Bradycardia Prediction Approach Using Nonparametric Kernel Density Estimation

ECG data obtained from a preterm infant is first preprocessed and segmented (described in Section 4.4.1) such that each segment contains both normal and bradycardia beats. Using continuous normal beats, we extract R-peak information and estimate our nonspecific probability density function. After setting a desired level of false alarm to be tolerated by the system, a threshold region is found using the estimated density and a generated threshold plane. We use R-peaks from further along in time and test against this threshold region to determine the onset of near-term bradycardia. Section 4.4.1 describes a statistical guarantee to achieve the desired false alarm rate while Section 4.3 presents a practical way to construct the prediction

set.

4.2.1 Nonparametric Prediction Test

We construct a hypothesis test to predict bradycardia in infants' heartbeat. To this end, we employ a nonparametric method for estimating a kernel-based probability density function to model RRI information to use in predicting the onset of bradycardia. We consider ECG segments that are first processed to remove the baseline wander. We then detect the R peaks using Pan-Tompkins algorithm (Pan and Tompkins (1985)). Assuming that the number of R peaks in an ECG segment is N. We define the R-tuple $x_n = (t_n, R_n)$, $n = 1, \ldots, N$ and $x_n \in \mathcal{X}$, to be the time of the peak occurrence and the peak amplitude R_n, respectively. We assume that the R-tuple set $X_N = \{x_1,\ x_2,\ \ldots,\ x_N\}$ is identically and independently drawn from an unknown density $p(x)$. We exploit the kernel density estimator to estimate $p(x)$ using the data points X_n as

$$\hat{p}_H(x) = \frac{1}{N} \sum_{n=1}^{N} K_H(x - x_n) \tag{4.1}$$

for the positive definite bandwidth matrix H and $K_H(x) = |H|^{-1/2}\, K\big(H^{-1/2}x\big)$ where K is smoothing kernel function in $\mathbb{R}$. The estimated density depends directly on the smoothing kernel K_H and the bandwidth H. For the simplicity, we assume that K_H is Gaussian and $H = h^2\, I_2$, where I_2 is the (2×2) identity matrix. We learn the best value of h using leave-one-out cross validation to ensure accurate prediction and avoid overfitting Celisse (2014).

Given a desirable probability of false alarm, P_{FA}, we design a hypothesis testing that produces $(1 - P_{\text{FA}})$ confidence. We define the null hypothesis $\mathcal{H}_0$ as the hypothesis that the density of the next R-tuple x_{N+1} is the same as that of the previous R-tuples in the set X_N; that is, $\mathcal{H}_0 \colon\ x_{N+1} = x$, for all possible values of $x \in \mathcal{X}$. We aim to

construct a confidence set $\mathcal{A}_{\mathcal{X}}$, that consists of all values X_N, such that the probability of the next R-tuple, x_{N+1}, belonging to this set satisfies the following condition

$$\Pr(x_{N+1} \in \mathcal{A}_{\mathcal{X}}) \geq (1 - P_{\text{FA}}).$$

Note that inverting the test produces a prediction set. It is shown that the prediction set $\mathcal{A}_{\mathcal{X}}$ is finite and distribution-free Lei *et al.* (2013). To this end, we utilize the kernel density estimator $\hat{p}_H^a(x)$ based on the augmented data set $X_N \cup \{x\}$ for a fixed value of $x \in \mathcal{X}$. The rank of $\hat{p}_H^a(x_1), \ldots, \hat{p}_H^a(x_{N+1})$ is uniformly distributed under the null hypothesis. Thus, For each value of x, the p-value η_x is given by

$$\eta_x = \frac{1}{N+1} \sum_{n=1}^{N} \mathbb{I}\Big(\hat{p}_H^a(x_n) \leq \hat{p}_H^a(x)\Big) \tag{4.2}$$

where $\mathbb{I}$ is the indicator function. We thus define the $(1 - P_{\text{FA}})$ confidence set as $\mathcal{A}_{\mathcal{X}} = \{x : \eta_x \geq P_{\text{FA}}\}$. As mentioned earlier the prediction set $\mathcal{A}_{\mathcal{X}}$ is distribution-free and is only determined using the finite set X_N Shafer and Vovk (2008). Implementing this method is impractical, thus we introduce a bigger set that contains $\mathcal{A}_{\mathcal{X}}$ and can easily be constructed.

4.2.2 Unsupervised Prediction

Given the confidence set $\mathcal{A}_{\mathcal{X}}$, we can successfully predict the next bradycardia event with a 95% accuracy by setting $P_{\text{FA}} = 0.05$. However, as it may not always be possible to compute the set $\mathcal{A}_{\mathcal{X}}$, we construct instead a feasible but larger confidence set $\mathcal{B}_{\mathcal{X}}$ that is easier to compute and preserves the same accuracy. Define $y_n = \hat{p}_H(x_n)$ for $n = 1, \ldots, N$. Assume that y_i is sorted in an ascending order, that is, $y_1 \leq \cdots \leq y_N$. We construct the prediction set $\mathcal{B}_{\mathcal{X}}$ as

$$\mathcal{B}_{\mathcal{X}} = \{x : \hat{p}_H(x) \geq \mathcal{C}_k\},$$

for the threshold plane is computed as $\mathcal{C}_k = y_k - (K_H(0)/N|H|^{1/2})$ where $k = \lfloor (N + 1)P_{\text{FA}} \rfloor$.

Proposition 1 (Lei *et al.* (2013) Theorem 3.4) *The set $\mathcal{B}_\mathcal{X} \supset \mathcal{A}_\mathcal{X}$ satisfies*

$$Pr(x_{N+1} \in \mathcal{A}_\mathcal{X}) \geq (1 - P_{FA}) \implies Pr(x_{N+1} \in \mathcal{B}_\mathcal{X}) \geq (1 - P_{FA}). \tag{4.3}$$

In particular, the prediction set $\mathcal{B}_\mathcal{X}$ follows from the projection of estimated density which is above the threshold $\mathcal{C}_k$.

4.3 Evaluation Metric

To evaluate our proposed method, we define the evaluation metric *estimated predictive error* (EPE) as

$$\text{EPE} = \frac{\text{Number of False Alarms}}{\text{Number of R-tuples tested}} \tag{4.4}$$

Given an ECG segment containing N peaks, a desired level of P_{FA}, and the prediction set $\mathcal{B}_\mathcal{X}$ over the normal heartbeats in that segment, we can compute the estimated predictive error by testing whether R-tuples from a future time belong to $\mathcal{B}_\mathcal{X}$. Our proposed method implies that the R-tuple $x_m = (t_m, R_m)$ is predicted, with (1 - P_{FA})% confidence, to be the onset of bradycardia if $x_m \notin \mathcal{B}_\mathcal{X}$, $m > N$. If the R-tuple x_m is a bradycardia R-tuple and $x_m \in \mathcal{B}_\mathcal{X}$, $m > N$, then it is a false alarm and counts towards the estimated predictive error.

The definition of the EPE indicates that, for a given probability of false alarm P_{FA}, lower value of the estimated predictive error results in the fewer false alarms being raised. Thus, EPE is a measure of performance which is used for different values of P_{FA} to demonstrate how well our proposed method performs.

For a fixed value of P_{FA}, we compute the total estimated predictive error for each preterm infant as the average of the estimated predictive error (EPE) over all ECG

segments for that preterm infant. Similarly, lower value of total estimated predictive error indicates better performance of the method.

4.4 Dataset, Results and Discussion

In this section, we take a look at the dataset used for this problem and associated means of preprocessing it. Results obtained from the implementation of the proposed method on the chosen dataset is presented and discussed.

4.4.1 Dataset and Preprocessing Data

To demonstrate the performance of our proposed method, we make use of the dataset provided by the MIT Preterm Infant Cardio-respiratory Signals (PICS) database (Gee *et al.* (2017); Goldberger *et al.* (2000)). ECG data from ten preterm infants with post-conceptional ages ranging between $29\frac{3}{7}$ and $34\frac{2}{7}$ weeks are collected for approximately 20 to 70 hours per infant at a sampling frequency of 500Hz as shown in Table 4.1. ECG data taken from each infant in the database is subject to the removal of baseline wander by using a high-pass filter with cut-off frequency between 0.5-0.6 Hz. We then remove the motion and disconnection artifacts from the signal by visual inspection. Following the aforementioned steps, we obtain the cleaned-up ECG signal for each infant in the database. To locate the R-peaks in the cleaned-up data, we employ the Pan-Tompkins algorithm (Pan and Tompkins (1985)). Finally, each signal is segmented by using the annotations provided by the database regarding the location of occurrence of bradycardia events. Each signal contains both regular data points (data collected five minutes prior to the first bradycardia event) and bradycardia beats (data collected two minutes after the first bradycardia event).

Table 4.1 displays the number of bradycardia events and the number of ECG segments generated for each preterm infant in the dataset. For each infant, R-tuples

Table 4.1: Duration of ECG, Number of Bradycardia Events, Number of ECG Segments for Ten Preterm Infants

Preterm Infant	1	2	3	4	5	6	7	8	9	10
Duration (hours)	45.6	43.8	43.7	46.8	48.8	48.6	20.3	24.6	70.3	45.1
Bradycardia Events	77	72	80	66	72	56	34	28	97	40
ECG Segments	77	72	80	66	72	56	34	28	97	40

are generated for each segment and utilized in our proposed model.

4.4.2 Experimental Setup

Using the preprocessed data from Section 4.4.1, we perform the following procedure for each infant over each segment. For a desired false alarm rate P_{FA}, we choose he bandwidth parameter h through leave-one out cross-validation. Assuming a Gaussian kernel, we estimate the density to be $\hat{p}_H(x)$ for the normal heartbeats in each segment. We then compute the threshold plane $\mathcal{C}_k$ and generate the prediction set $\mathcal{B}_{\mathcal{X}}$ for each segment as the area under the intersection of the density $\hat{p}_H(x)$ and the threshold plane $\mathcal{C}_k$. We compute the estimated predictive error (EPE) by testing whether R-tuples from the near future (during and after the bradycardia event) belong to the prediction set $\mathcal{B}_{\mathcal{X}}$. If a future R-tuple from the bradycardia region belongs to $\mathcal{B}_{\mathcal{X}}$ then it is considered to be an error. The above process is repeated for each preterm infant for the false alarm rates, P_{FA}, of 5%, 4%, 3%, 2% and 1%.

Specifications of the desktop computer used for this implementation are listed below:

- Intel Core i7 4790 4 core CPU (base clock 3.60 GHz)
- 32GB RAM

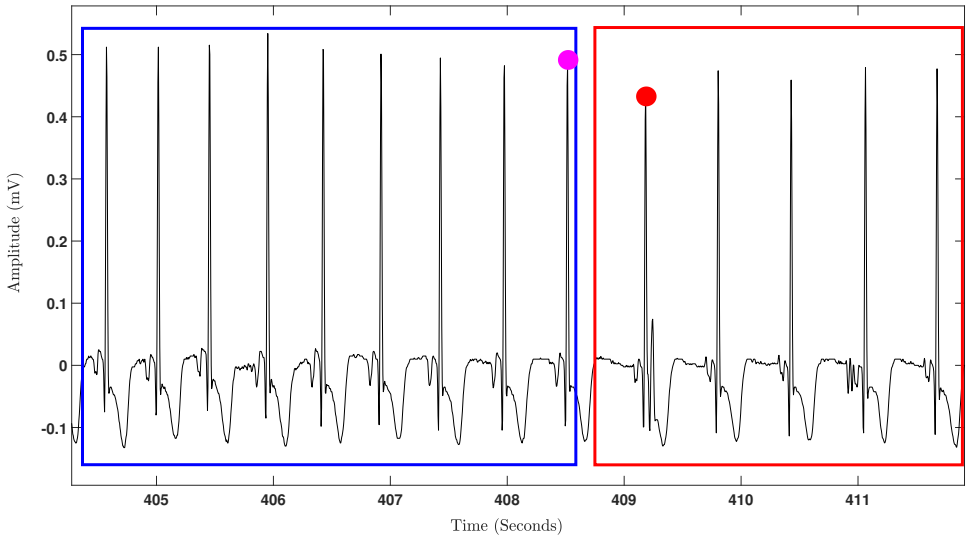


Figure 4.1: Preterm Infant ECG Segments: Normal Region (Blue Box) and Bradycardia Region (Red Box)

- NVIDIA GTX 745 GPU with 4GB VRAM
- MATLAB 2019a

The average runtime for computing the kernel density using the aforementioned system for each ECG segment is 120 ± 30 seconds. We note that this computation time can be lowered using a server based computation system.

Table 4.2: Average Estimated Predictive Error for Ten Preterm Infants for P_{FA} Between 0.05 To 0.01

Preterm Infant	1	2	3	4	5	6	7	8	9	10
$P_{\text{FA}} = 0.05$	0.092	0.099	0.092	0.04	0.04	0.19	0.08	0.194	0.091	0.087
$P_{\text{FA}} = 0.04$	0.095	0.284	0.387	0.29	0.31	0.45	0.31	0.295	0.223	0.26
$P_{\text{FA}} = 0.03$	0.35	0.457	0.681	0.321	0.38	0.425	0.498	0.342	0.195	0.46
$P_{\text{FA}} = 0.02$	0.82	0.68	0.943	0.91	0.87	0.9	0.89	0.694	0.83	0.87
$P_{\text{FA}} = 0.01$	0.89	0.901	0.991	0.99	0.99	0.94	0.99	0.82	0.998	0.907

4.4.3 *Results*

Figure 4.1 displays a portion of an ECG segment for preterm infant 1 containing both normal and bradycardia heartbeats. Using leave-one-out cross validation, we estimate the bandwidth parameter h for the Gaussian kernel to be 33.2313. Figure 4.2 provides a pictorial demonstration of our proposed method for fixed value of $P_{\text{FA}} = 0.05$ and the kernel density estimator with the bandwidth $h = 33.2313$. In particular, we estimate the threshold plane $\mathcal{C}_k$ to be 0.0028 and used it to compute the prediction set $\mathcal{B}_{\mathcal{X}}$ for this specific segment. The contour region obtained from the intersection of the estimated density $\hat{p}_H(x)$ and plane $\mathcal{C}_k$ provides the prediction set $\mathcal{B}_{\mathcal{X}}$. The prediction set $\mathcal{B}_{\mathcal{X}}$ is shown in Figure 4.3. All R-tuples that lie inside this contour region are considered to be normal heartbeats. For the ECG segment shown in Figure 4.1, we tested two R-tuples that indicate the last normal beat (R-tuple $(408.6, 0.48)$ denoted by a pink marker) and the first bradycardia beat (R-tuple $(409.3, 0.44)$ denoted by a red marker). We observe that the R-tuple for the last

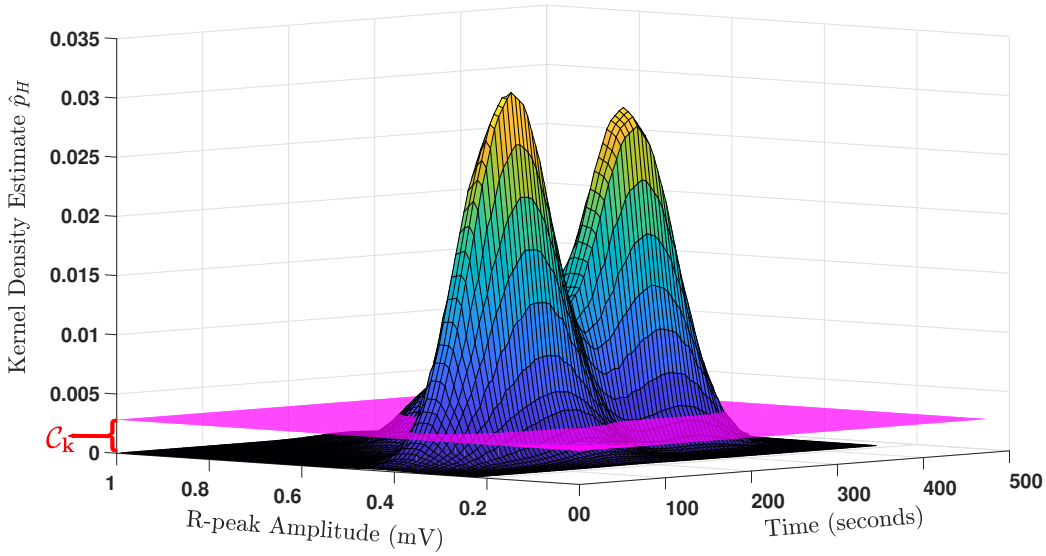


Figure 4.2: Kernel Density Estimator $\hat{p}_h(x)$ and Threshold Region $\mathcal{C}_k$ for Preterm Infant 1 ECG Segment

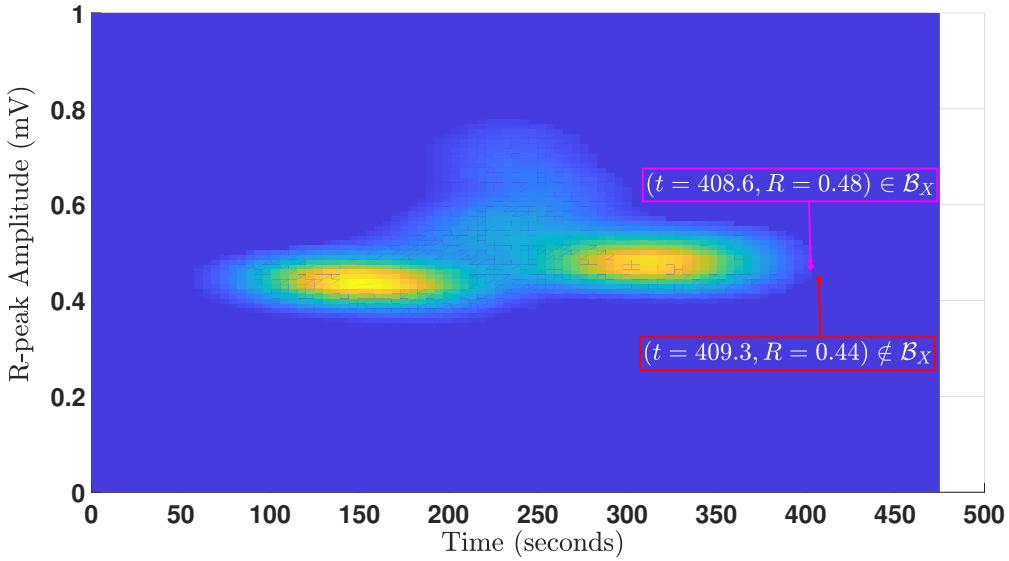


Figure 4.3: Region $\mathcal{B}_{\mathcal{X}}$ Obtained from the Intersection of the Density and the Threshold Region

normal beat lies just within the prediction set (denoted by the contour region) while the R-tuple for the bradycardia beat lies outside the contour region. Using this threshold region, we can predict the onset of near-term bradycardia by generating R-tuples from future ECG data and locating the first R-tuple that lies outside this region. We use ten random R-tuples from a future time for each segment to compute the estimated predictive error for the corresponding segment. As mentioned in Section 4.3, the estimated predictive error value indicates how well our method performs at a desired P_{FA}. Low values of this metric imply that our method is able to predict the onset of bradycardia with the specified level of accuracy and rate of false alarm. This process is repeated for all ECG segments for preterm infant 1 for P_{FA} values ranging between 0.05 to 0.01 (5% to 1%). We compute the total predictive error for each P_{FA} value as shown in Figure 4.4 for preterm infant 1. Figure 4.4 demonstrates that our proposed method works best at $P_{\text{FA}} = 0.05$ and is able to predict the onset of bradycardia with 95% accuracy as expected. The total average estimated predictive error is lowest for this value of P_{FA} indicating very few bradycardia R-tuples are

mispredicted. We also observe that as we lower the probability of false alarm the total estimated predictive error increases.

Similarly, we compute the total average estimated predictive error for each of the ten preterm infants for P_{FA} values of 5%, 4%, 3%, 2% and 1% shown in Table 4.2. A similar trend can be observed across all ten preterm infants in the database - as the probability of false alarm P_{FA} is reduced, we see a significant increase in the average estimated predictive error (Table 4.2). Hence we demonstrate that our proposed method is able to predict the onset of bradycardia with higher accuracy and low false alarm rate as compared to other existing methods.

4.4.4 Discussion

In the previous section, we demonstrated that our proposed method works and is able to predict the onset of near-term bradycardia with an accuracy of $(1 - P_{\text{FA}})$ for a desired probability of false alarm P_{FA}. We also showed that our proposed method achieves the highest prediction accuracy compared to prior work while maintaining

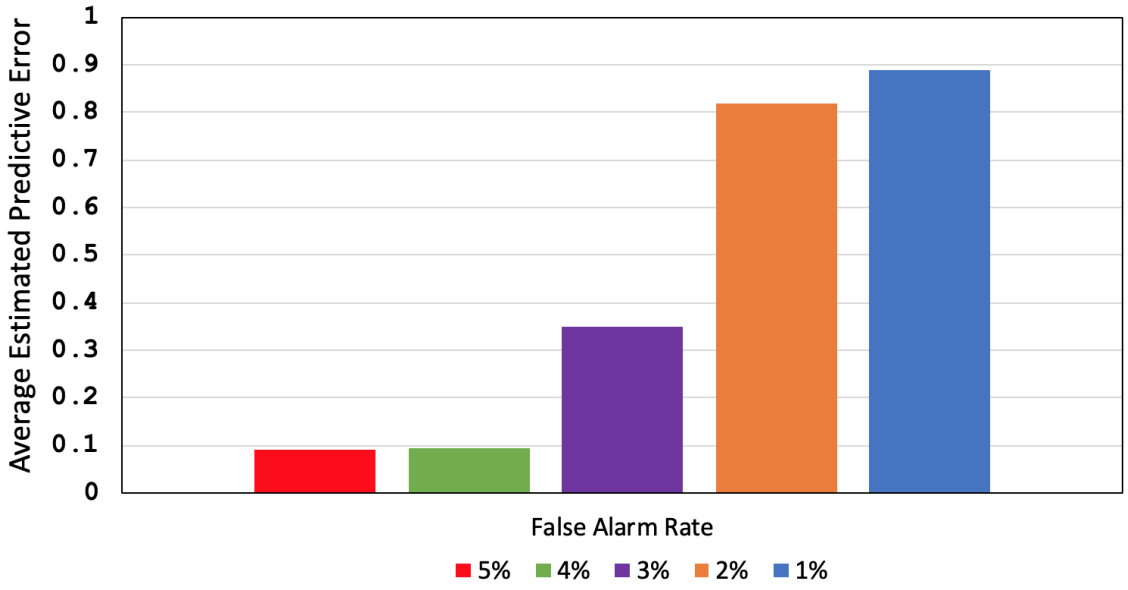


Figure 4.4: Average Estimated Predictive Error for Varying P_{FA} in Infant 1

the lowest rate of false alarm.

As shown in Figure 4.4 and Table 4.2, we observe that as the P_{FA} value is lowered from 5% (0.05) to 1% (0.01), the total average estimated predictive error increases. The total average estimated predictive error is lowest at $P_{\text{FA}} = 0.05$ with a slight increase across all infants at $P_{\text{FA}} = 0.04$. The error increases drastically as the P_{FA} value is lowered to 0.03, 0.02 and 0.01. However, this trend lies within our expectation. By lowering the P_{FA} value, the contour region under the intersection of the threshold plane $\mathcal{C}_k$ and density $\hat{p}_H(x)$ increases that is, the prediction set $\mathcal{B}_{\mathcal{X}}$ grows. The enlargement of the prediction set allows for R-tuples that do not belong to the actual set to erroneously be included in the region. This fact results in the fact that bradycardia R-tuples are mistaken as normal beats - false alarms. In practice, it is not feasible to remove false alarms in the entirety. Instead, we strive to minimize the rate of false alarms while trying to maximize the prediction accuracy as much as possible. Our proposed method demonstrates this and shows that it is not feasible to guarantee accurate predictions with an expect level of confidence beyond a certain value of P_{FA}.

Chapter 5

CONCLUSION AND FUTURE WORK

5.1 Conclusion

This work focused on developing adaptive statistical machine learning algorithms for physical systems for the purpose of forecasting system behavior and anomalous trends in the system.

For Type 1 Diabetes Mellitus patients, with the use of personal glucose monitoring devices and insulin infusion pumps, affected individuals can self-monitor their blood glucose levels and inject the required amount of insulin as and when needed. However, since these monitoring devices use model predictive controllers for calculating the required insulin dose based on thresholding and naive prediction algorithms, there is still a risk of extreme glycemic events occurring. This work also notes that a common data driven model is vastly outperformed by a per patient model in the case of estimating and predicting the blood glucose level in Type 1 Diabetes Mellitus patients. Furthermore, since the data acquired from each patient is unique to their physical system, it is necessary to be able to determine the underlying representations that are unique to each patient and thereby enhance the level of care that can be provided. This dissertation proposes and develops a novel model guided neural network framework that utilizes multi-variate time series data from various modalities to forecast trends in the physical system. The proposed work combines the empirical evidence acquired from various sensors and human input with the physiological model to enhance the accuracy of predictions made. This novel solution also optimizes the objective function of the proposed model guided neural network by incorporating do-

main knowledge within it. Also, by utilizing the inherent nature of recurrent neural networks to exploit the temporal trends, the proposed novel solution increases the ability to forecast trends in each individual by a significant margin while being able to operate with lower amount of data. The proposed approaches are particularly suitable for use in free living scenarios since it reduces the risk of extreme glycemic events by making accurate forecasts for insulin dose calculation. Utilizing this approach in a model predictive controller will allow for accurate real-time predictions which can be improved over time by conditioning the model using the physical system's unique parameters.

The proposed model guided neural network framework utilizes both empirical evidence and domain knowledge to improve the accuracy of predictions and adds an element of explainability to the overall solution. Since neural networks were chosen in this work as the empirical learning algorithm, it is necessary to be able to explain and understand the predictions made by the model with respect to the understanding of the physical system it is being applied to. By incorporating domain knowledge within such a learning algorithm, we demonstrate that this not only improves the performance of the learning solution but also adds a level of explainability that was not present before. An application of this proposed model guided framework, as shown in this work, is towards the prediction of blood glucose levels in Type I Diabetes Mellitus patients which demonstrates the significant improvement gained in predicting blood glucose levels. This work can also be adapted and used in various other fields which use physical based models to develop predictive solutions. The only requirement that must be met is the availability of a physical numerical model that can be used as the model guide. It is to be noted that quality of the physical model also plays an important role i.e. a physical model that can generate better approximates of the estimates of the required solution will increase the overall learning capability of the

proposed solution. Therefore the choice of the learning algorithm and the physical model guide are both important in utilizing the proposed model guided framework for predictive solutions.

On the other hand, utilizing univariate time series data from preterm infants in the form of electrocardiogram observations, this work proposed and developed a per patient non-parametric model for predicting the on-set of near-term bradycardia. Using non-parametric kernel density estimate and thresholding based on features extracted from the observations, the proposed method significantly lowers the rate of false alarm while being accurate in the predictions made. By allowing the health care experts to dictate the tolerable rate of false alarm, this work introduces a level of customization that can adapt to their various needs. This is done without causing a reduction in the overall prediction accuracy of the system.

5.2 Future Work

Several extension of this work is possible. Some of them are listed below:

5.2.1 Forecasting Blood Glucose Levels in Type 1 Diabetes Mellitus Patients

1. Improved physiological modeling of blood glucose kinetics: The Bergman minimal model provides an approximation of interstitial blood glucose levels and not the intravenous blood glucose levels. In order to further improve the impact of the guide model on the neural network system, a physiological model that estimates the intravenous blood glucose levels would vastly improve the forecasting ability of our work.

2. Estimation of patient specific parameters: An improvement to the overall model prediction accuracy would be observed in the case of extracting and estimating all of the required parameters on a per-patient basis. The novel approaches presented in

this dissertation utilize patient specific attributes as available although some of the parameters were global approximations found in literature and are widely accepted.

3. Model mixing: While this work investigates the percent contribution of the empirical model learned by the system and the physiological guide model, a study involving a more accurate physiological model would aid in understanding and increasing the overall performance.

4. Modeling the physiological model as a learning system: To make the proposed work dynamic and adaptable to patients in real time, an avenue that can be explored is the implementation of a stacked neural network that is a close approximation of the physiological model being used.

5. Type II Diabetes Mellitus: This work utilizes data and physiological factors specific to Type I Diabetes Mellitus patients. However, the model guided LSTM framework can be applied for Type II Diabetes Mellitus patients as well with minor modifications. Primarily, a physical model that captures the blood glucose kinetics, especially insulin diffusion rate and insulin concentration in the blood is required to account for the low insulin absorption rates characteristic of this variation.

6. Application of proposed model guided solution to other fields: This work applied the proposed model guided solution for the prediction of blood glucose in Type I diabetes patients and demonstrated an improvement in the predictions made. Since a generalized framework is proposed and then implemented for this particular domain of blood glucose, this proposed solution could be applied to other research areas like remote sensing, physical sciences, finances and so on, where accurate and explainable predictive solutions are required. The availability of a physical numerical model and the choice of the empirical learning algorithm would be critical while adapting the proposed model in this work for other domains.

5.2.2 Predicting the onset of Bradycardia in Preterm Infants

1. Decreasing computation time: The work proposed in this dissertation utilizes kernel density estimation to predict upcoming bradycardia events. Generation of this density function is expensive in time and can hamper the performance in a real time implementation. Exploration of methods to generate the density function in a shorter amount of time using multithread workloads.

2. Inclusion of observations from other modalities: Electrocardiogram data is shown to be sufficient for the purpose of bradycardia prediction in preterm infants. However, the inclusion of data from other modalities like breathing patterns, blood markers and other biological processes will improve the proposed work further.

3. Application to other research problems: Predictive models generated using only observed data are required in multiple fields that span various research domains. Adaptation and application of the proposed solution to similar research problems in other research domains is left as future work.

In addition, the developed model guided neural network and kernel density estimation approach can be applied to various areas like remote sensing, financial, meteorological time-series data for the purpose of forecasting trends in the data.

REFERENCES

Agrawal, A., *Evaluation of a Guided Machine Learning Approach for Pharmacokinetic Modeling*, Ph.D. thesis, Arizona State University (2017).

Alashkar, T., S. Jiang, S. Wang and Y. Fu, "Examples-rules guided deep neural network for makeup recommendation.", in "AAAI", pp. 941–947 (2017).

Banerjee, A. and S. K. Gupta, "Spatio-temporal hybrid automata for safe cyber-physical systems: A medical case study", in "2013 ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS)", pp. 71–80 (IEEE, 2013).

Banerjee, A. and S. K. Gupta, "Analysis of smart mobile applications for healthcare under dynamic context changes", IEEE Transactions on Mobile Computing **14**, 5, 904–919 (2014).

Banerjee, A., Y. Zhang, P. Jones and S. Gupta, "Using formal methods to improve home-use medical device safety", Biomedical instrumentation & technology **47**, s1, 43–48 (2013).

Bergman, R. N., Y. Z. Ider, C. R. Bowden and C. Cobelli, "Quantitative estimation of insulin sensitivity.", American Journal of Physiology-Endocrinology And Metabolism **236**, 6, E667 (1979).

Bergman, R. N., L. S. Phillips and C. Cobelli, "Physiologic evaluation of factors controlling glucose tolerance in man: measurement of insulin sensitivity and beta-cell glucose sensitivity from the response to intravenous glucose.", Journal of clinical investigation **68**, 6, 1456 (1981).

Blackburn, S., "Problems of preterm infants after discharge", Journal of Obstetric, Gynecologic, & Neonatal Nursing **24**, 1, 43–49 (1995).

Bojarski, M., D. Del Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang *et al.*, "End to end learning for self-driving cars", arXiv preprint arXiv:1604.07316 (2016).

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

Celisse, A., "Optimal cross-validation in density estimation with the l^2-loss", The Annals of Statistics **42**, 1879–1910 (2014).

Chamberlain, R. C., K. Fleetwood, S. H. Wild, H. M. Colhoun, R. S. Lindsay, J. R. Petrie, R. J. McCrimmon, F. Gibb, S. Philip, N. Sattar *et al.*, "Foot ulcer and risk of lower limb amputation or death in people with diabetes: A national population-based retrospective cohort study", Diabetes Care **45**, 1, 83–91 (2022).

Chen, S., S. Zhao, C. Dalman, H. Karlsson and R. Gardner, "Association of maternal diabetes with neurodevelopmental disorders: autism spectrum disorders, attention-deficit/hyperactivity disorder and intellectual disability", International journal of epidemiology **50**, 2, 459–474 (2021).

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

Colberg, S. R., R. Laan, E. Dassau and D. Kerr, "Physical activity and type 1 diabetes: time for a rewire?", Journal of diabetes science and technology **9**, 3, 609–618 (2015).

Collobert, R. and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning", in "Proceedings of the 25th international conference on Machine learning", pp. 160–167 (ACM, 2008).

Das, S., A. Agrawal, A. Banerjee and S. K. Gupta, "Model guided deep learning approach towards prediction of physical system behavior", in "2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA)", pp. 1079–1082 (IEEE, 2017).

Dybjer, E., A. K. Dahl Aslan, G. Engström, E. D. Nilsson, K. Nägga, P. M. Nilsson and L. B. Hassing, "Type 1 diabetes, cognitive ability and incidence of cardiovascular disease and death over 60 years of follow-up time in men", Diabetic Medicine p. e14806 (2022).

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

Escobedo-de la Peña, J., R. A. Rascón-Pacheco, I. de Jesús Ascencio-Montiel, E. González-Figueroa, J. E. Fernández-Gárate, O. S. Medina-Gómez, P. Borja-Bustamante, J. A. Santillán-Oropeza and V. H. Borja-Aburto, "Hypertension, diabetes and obesity, major risk factors for death in patients with covid-19 in mexico", Archives of medical research **52**, 4, 443–449 (2021).

Faruqui, S. H. A., Y. Du, R. Meka, A. Alaeddini, C. Li, S. Shirinkam and J. Wang, "Development of a deep learning model for dynamic forecasting of blood glucose level for type 2 diabetes mellitus: secondary analysis of a randomized controlled trial", JMIR mHealth and uHealth **7**, 11, e14452 (2019).

Gee, A. H., R. Barbieri, D. Paydarfar and P. Indic, "Predicting bradycardia in preterm infants using point process analysis of heart rate", IEEE Transactions on Biomedical Engineering **64**, 2300–2308 (2017).

Gee, A. H., J. Chang, J. Ghosh and D. Paydarfar, "Bayesian online changepoint detection of physiological transitions", in "IEEE Engineering in Medicine and Biology Society", pp. 45–48 (2018).

Georga, E. I., V. C. Protopappas, D. Ardigo, M. Marina, I. Zavaroni, D. Polyzos and D. I. Fotiadis, "Multivariate prediction of subcutaneous glucose concentration in type 1 diabetes patients based on support vector regression", IEEE journal of biomedical and health informatics **17**, 1, 71–81 (2012).

Georga, E. I., V. C. Protopappas and D. I. Fotiadis, "Glucose prediction in type 1 and type 2 diabetic patients using data driven techniques", Knowledge-oriented applications in data mining pp. 277–296 (2011).

Gillis, R., C. C. Palerm, H. Zisser, L. Jovanovic, D. E. Seborg and F. J. Doyle III, "Glucose estimation and prediction through meal responses using ambulatory subject data for advisory mode model predictive control", (2007).

Glorot, X. and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks", in "Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics", pp. 249–256 (2010).

Goldberger, A. L., L. A. N. Amaral, L. Glass, J. M. Hausdorff *et al.*, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals", Circulation **101**, e215–e220, URL `http://physionet.mit.edu/physiobank/database/picsdb` (2000).

Gregg, E. W. and A. Menke, "Diabetes and disability", (2021).

Gu, W., Z. Zhou, Y. Zhou, M. He, H. Zou and L. Zhang, "Predicting blood glucose dynamics with multi-time-series deep learning", in "Proceedings of the 15th ACM conference on embedded network sensor systems", pp. 1–2 (2017).

Hochreiter, S. and J. Schmidhuber, "Long short-term memory", Neural computation **9**, 8, 1735–1780 (1997).

Hovorka, R., V. Canonico, L. J. Chassin, U. Haueter, M. Massi-Benedetti, M. O. Federici, T. R. Pieber, H. C. Schaller, L. Schaupp, T. Vering *et al.*, "Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes", Physiological measurement **25**, 4, 905 (2004).

Hu, C., Y. Zuo, C. Chen, S. P. Ong and J. Luo, "Genetic algorithm-guided deep learning of grain boundary diagrams: addressing the challenge of five degrees of freedom", Materials Today **38**, 49–57 (2020).

Karegowda, A. G., M. Jayaram and A. Manjunath, "Cascading k-means clustering and k-nearest neighbor classifier for categorization of diabetic patients", International Journal of Engineering and Advanced Technology **1**, 3, 147–151 (2012).

Kingma, D. and J. Ba, "Adam: A method for stochastic optimization", arXiv preprint arXiv:1412.6980 (2014).

Kok, P., "Predicting blood glucose levels of diabetics using artificial neural networks", Research Assignment for Master of Science, Delft University of Technology (2004).

Krizhevsky, A., I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks", in "Advances in neural information processing systems", pp. 1097–1105 (2012).

Lee, S., J. Zhou, C. L. Guo, W. T. Wong, T. Liu, I. C. K. Wong, K. Jeevaratnam, Q. Zhang and G. Tse, "Predictive scores for identifying patients with type 2 diabetes mellitus at risk of acute myocardial infarction and sudden cardiac death", Endocrinology, diabetes & metabolism **4**, 3, e00240 (2021).

Lei, J., J. Robins and L. Wasserman, "Distribution-free prediction sets", Journal of the American Statistical Association **108**, 278–287 (2013).

Li, N., J. Tuo, Y. Wang and M. Wang, "Prediction of blood glucose concentration for type 1 diabetes based on echo state networks embedded with incremental learning", Neurocomputing **378**, 248–259 (2020).

Mahmud, S. M., H. Wang and Y. Kim, "Accelerated prediction of bradycardia in preterm infants using time-frequency analysis", in "Int. Conference on Computing, Networking and Communications", pp. 468–472 (2019).

Martinsson, J., A. Schliep, B. Eliasson, C. Meijner, S. Persson and O. Mogren, "Automatic blood glucose prediction with confidence using recurrent neural networks", in "KHD@ IJCAI", (2018).

Mhaskar, H. N., S. V. Pereverzyev and M. D. Van der Walt, "A deep learning approach to diabetic blood glucose prediction", Frontiers in Applied Mathematics and Statistics **3**, 14 (2017).

Mittal, A. and N. Paragios, "Motion-based background subtraction using adaptive kernel density estimation", in "Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.", vol. 2, pp. II–II (Ieee, 2004).

Moraffah, B., "Inference for multiple object tracking: A Bayesian nonparametric approach", arXiv preprint arXiv:1909.06984 cs.LG (2019).

Moraffah, B. and A. Papandreou-Suppappola, "Random infinite tree and dependent Poisson diffusion process for nonparametric Bayesian modeling in multiple object tracking", in "International Conference on Acoustics, Speech, and Signal Processing", pp. 5217–5221 (2019).

Munoz-Organero, M., "Deep physiological model for blood glucose prediction in t1dm patients", Sensors **20**, 14, 3896 (2020).

Pan, J. and W. J. Tompkins, "A real-time QRS detection algorithm", IEEE Transactions on Biomedical Engineering **32**, 230–236 (1985).

Pappada, S. M., B. D. Cameron, P. M. Rosman, R. E. Bourey, T. J. Papadimos, W. Olorunto and M. J. Borst, "Neural network-based real-time prediction of glucose in patients with insulin-dependent diabetes", Diabetes technology & therapeutics **13**, 2, 135–141 (2011).

Parker, R. S., F. J. Doyle and N. A. Peppas, "A model-based algorithm for blood glucose control in type i diabetic patients", IEEE Transactions on biomedical engineering **46**, 2, 148–157 (1999).

Pérez-Gandía, C., A. Facchinetti, G. Sparacino, C. Cobelli, E. Gómez, M. Rigla, A. de Leiva and M. Hernando, "Artificial neural network algorithm for online glucose prediction from continuous glucose monitoring", Diabetes technology & therapeutics **12**, 1, 81–88 (2010).

Perlman, J. M. and J. J. Volpe, "Episodes of apnea and bradycardia in the preterm newborn: impact on cerebral circulation", Pediatrics **76**, 3, 333–338 (1985).

Pichler, G., B. Urlesberger and W. Müller, "Impact of bradycardia on cerebral oxygenation and cerebral blood volume during apnoea in preterm infants", Physiological measurement **24**, 3, 671 (2003).

Poets, C. F., R. S. Roberts, B. Schmidt, R. K. Whyte, E. V. Asztalos, D. Bader, A. Bairam, D. Moddemann, A. Peliowski, Y. Rabi *et al.*, "Association between intermittent hypoxemia or bradycardia and late death or disability in extremely preterm infants", Jama **314**, 6, 595–603 (2015).

Rabby, M. F., Y. Tu, M. I. Hossen, I. Lee, A. S. Maida and X. Hei, "Stacked lstm based deep recurrent neural network with kalman smoothing for blood glucose prediction", BMC Medical Informatics and Decision Making **21**, 1, 1–15 (2021).

Rather, A. M., A. Agarwal and V. Sastry, "Recurrent neural network and a hybrid model for prediction of stock returns", Expert Systems with Applications **42**, 6, 3234–3241 (2015).

Read, J. S., X. Jia, J. Willard, A. P. Appling, J. A. Zwart, S. K. Oliver, A. Karpatne, G. J. Hansen, P. C. Hanson, W. Watkins *et al.*, "Process-guided deep learning predictions of lake water temperature", Water Resources Research **55**, 11, 9173–9190 (2019).

Remme, C. A., "Sudden cardiac death in diabetes and obesity: mechanisms and therapeutic strategies", Canadian Journal of Cardiology (2022).

Schmidhuber, J., "Deep learning in neural networks: An overview", Neural networks **61**, 85–117 (2015).

Shafer, G. and V. Vovk, "A tutorial on conformal prediction", Journal of Machine Learning Research **9**, 371–421 (2008).

Sparacino, G., F. Zanderigo, S. Corazza, A. Maran, A. Facchinetti and C. Cobelli, "Glucose concentration can be predicted ahead in time from continuous glucose monitoring sensor time-series", IEEE Transactions on biomedical engineering **54**, 5, 931–937 (2007).

Sun, H., P. Saeedi, S. Karuranga, M. Pinkepank, K. Ogurtsova, B. B. Duncan, C. Stein, A. Basit, J. C. Chan, J. C. Mbanya *et al.*, "Idf diabetes atlas: Global, regional and country-level diabetes prevalence estimates for 2021 and projections for 2045", Diabetes research and clinical practice **183**, 109119 (2022).

Sun, Q., M. V. Jankovic, L. Bally and S. G. Mougiakakou, "Predicting blood glucose with an lstm and bi-lstm based deep neural network", in "2018 14th symposium on neural networks and applications (NEUREL)", pp. 1–5 (IEEE, 2018).

Tresp, V., T. Briegel and J. Moody, "Neural-network models for the blood glucose metabolism of a diabetic", IEEE Transactions on Neural networks **10**, 5, 1204–1213 (1999).

Truong, H., "Predicting adverse outcomes in preterm infants using early bedside monitor data", Thesis for Bachelor of Science in Physics, College of William and Mary, Williamsburg, VA (2018).

Upton, C., A. Milner and G. Stokes, "Episodic bradycardia in preterm infants.", Archives of disease in childhood **67**, 7 Spec No, 831–834 (1992).

Van Riel, N., "Minimal models for glucose and insulin kinetics", (2004).

Wang, W., M. Tong and M. Yu, "Blood glucose prediction with vmd and lstm optimized by improved particle swarm optimization", IEEE Access **8**, 217908–217916 (2020).

Yang, J., L. Li, Y. Shi and X. Xie, "An arima model with adaptive orders for predicting blood glucose concentrations and hypoglycemia", IEEE journal of biomedical and health informatics **23**, 3, 1251–1260 (2018).

Yardley, J. E., G. P. Kenny, B. A. Perkins, M. C. Riddell, J. Malcolm, P. Boulay, F. Khandwala and R. J. Sigal, "Effects of performing resistance exercise before versus after aerobic exercise on glycemia in type 1 diabetes", Diabetes care **35**, 4, 669–675 (2012).

Yin, C., R. Zhao, B. Qian, X. Lv and P. Zhang, "Domain knowledge guided deep learning with electronic health records", in "2019 IEEE International Conference on Data Mining (ICDM)", pp. 738–747 (IEEE, 2019).

Young, A., "Inhibition of glucagon secretion", Advances in pharmacology **52**, 151–171 (2005).

Zainuddin, Z., O. Pauline and C. Ardil, "A neural network approach in predicting the blood glucose level for diabetic patients", Int J Comput Intell **5**, 1, 72–79 (2009).

Zecchin, C., A. Facchinetti, G. Sparacino, G. De Nicolao and C. Cobelli, "Neural network incorporating meal information improves accuracy of short-time prediction of glucose concentration", IEEE transactions on biomedical engineering **59**, 6, 1550–1560 (2012).

Zeng, G., J. Zhou, X. Jia, W. Xie and L. Shen, "Hand-crafted feature guided deep learning for facial expression recognition", in "2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)", pp. 423–430 (IEEE, 2018).

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