DIABETES MANAGEMENT VIA GAUSSIAN PROCESS BANDITS

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF ENGINEERING AND SCIENCE

OF BILKENT UNIVERSITY

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR

THE DEGREE OF

MASTER OF SCIENCE

IN

ELECTRICAL AND ELECTRONICS ENGINEERING

By

Ahmet Alparslan Çelik

October 2021

DIABETES MANAGEMENT VIA GAUSSIAN PROCESS BANDITS
By Ahmet Alparslan Çelik
October 2021

We certify that we have read this thesis and that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Cem Tekin(Advisor)

Aykut Koç

Elif Vural

Approved for the Graduate School of Engineering and Science:

Ezhan Karaşan
Director of the Graduate School

ABSTRACT

DIABETES MANAGEMENT VIA GAUSSIAN PROCESS BANDITS

Ahmet Alparslan Çelik
M.S. in Electrical and Electronics Engineering
Advisor: Cem Tekin
October 2021

Management of chronic diseases such as diabetes mellitus requires adaptation of treatment regimes based on patient characteristics and response. There is no single treatment that fits all patients in all contexts; moreover, the set of admissible treatments usually varies over the course of the disease. In this thesis, we address the problem of optimizing treatment regimes under time-varying constraints by using volatile contextual Gaussian process bandits. In particular, we propose a variant of GP-UCB with volatile arms, which takes into account the patient's context together with the set of admissible treatments when recommending new treatments. Our Bayesian approach is able to provide treatment recommendations to the patients along with confidence scores which can be used for risk assessment. We use our algorithm to recommend bolus insulin doses for type 1 diabetes mellitus patients. We test our algorithm on *in-silico* subjects that come with open source implementation of the FDA-approved UVa/Padova type 1 diabetes mellitus simulator. We also compare its performance against a clinician. Moreover, we present a pilot study with a few clinicians and patients, where we design interfaces that they can interact with the model. Meanwhile, we address issues regarding privacy, safety, and ethics. Simulation studies show that our algorithm compares favorably with traditional blood glucose regulation methods.

Keywords: multi-armed bandit, volatile bandit, Gaussian processes, personalized medicine, type 1 diabetes mellitus, clinical decision support systems.

ÖZET

GAUSS SÜRECİ HAYDUTLARI İLE ŞEKER HASTALIĞI YÖNETİMİ

Ahmet Alparslan Çelik
Elektrik ve Elektronik Mühendisliği, Yüksek Lisans
Tez Danışmanı: Cem Tekin
Ekim 2021

Şeker hastalığı gibi kronik hastalıkların yönetimi, hasta öznitelikleri ve tedaviye verdiği yanıta göre tedavi rejimlerinin uyarlanmasını gerektirir. Tüm bağlamlarda tüm hastalara uyan tek bir tedavi yoktur; dahası, kabul edilebilir tedaviler kümesi genellikle hastalığın seyrine göre değişir. Bu tezde, değişken bağlamsal Gauss süreci haydutlarını kullanarak zamanla değişen kısıtlar altında tedavi rejimlerini eniyileme problemini ele alıyoruz. Özellikle, yeni tedaviler önerirken, hastanın bağlamını ve kabul edilebilir tedaviler kümesini dikkate alan, değişken kollu bir GP-UCB uyarlaması öneriyoruz. Bayesci yaklaşımımız, hastalara risk değerlendirmesi için kullanılabilecek güven puanları ile birlikte tedavi önerileri sunabilmektedir. Algoritmamız tip 1 şeker hastaları için bolus insülin önermede kullanılmıştır. Algoritmamız FDA onaylı UVa/Padova tip 1 şeker hastalığı simülatörünün açık kaynak uyarlamasıyla gelen bilgisayar ortamındaki deneklerde test edilmiştir. Performansı bir klinisyenle kıyaslanmıştır. Ayrıca, birkaç klinisyen ve hasta ile, onların modelle etkileşime girebileceği arayüzler tasarladığımız bir pilot çalışma sunulmuştur. Bu sırada, gizlilik, güvenlik ve etik ile ilgili konular ele alınmıştır. Simülasyon çalışmaları algoritmamızın geleneksel kan şekeri düzenleme yöntemleriyle en iyi şekilde karşılaştırıldığını göstermektedir.

Anahtar sözcükler: çok-kollu haydut, değişken haydut, Gauss süreci, kişiselleştirilmiş tıp, tip 1 diyabet, klinik karar destek sistemleri.

Acknowledgement

This work is a result of continuous collaboration with my advisor, Assoc. Prof. Cem Tekin in a three-years-long joint interdisciplinary project carried out by Bilkent University, Ministry of Health of Turkey, and ESEN Inc., and funded by TÜBITAK. His trust, guidance, technical and editorial advice has been essential to my professional growth and completion of this dissertation. I am grateful to him for helping me to expand my understanding of the subject and enhance my research skills.

I want to thank Bilkent for financial support, the opportunity to meet with such resourceful people, and access invaluable resources throughout my study. My years in Bilkent have contributed significantly to my professional and personal development. I am thankful to the members of our research group (CYBORG), İlker, Alireza, Andi, and others for engaging conversations, and research feedback. I also want to express my gratitude to Kayra for his precious friendship for all these years and Seçil for being a great partner during the covid lockdown and also making me a better person.

Finally, I am indebted to my family for continuous support in everything I do and accomplish. It is a privilege to have them. I am and will always be grateful for their love.

This work was supported by TÜBITAK under grant 215E342. I want to thank Yıldız Pekcioglu and Prof. Sema Ucak-Basat, MD, for their kind assistance in the clinical data collection. Sema Ucak-Basat is a Professor in the Department of Internal Diseases, Health Sciences University, Umraniye Training and Research Hospital, Istanbul, Turkey. Yıldız Pekcioglu works in the same hospital as a registered dietitian.

Contents

List of Figures

List of Tables

List of Abbreviations

Abbreviation	Description
BG	blood glucose
CBR	case-based reasoning
CF	insulin correction factor
CGM	continuous glucose monitoring
CMAB	contextual multi-armed bandit
FDA	United States Food and Drug Administration
GP	Gaussian process
HBGI	high blood glycemic index
ICR	insulin to carbohydrate ratio
IDF	International Diabetes Federation
LBGI	low blood glycemic index
MAB	multi-armed bandit
MDP	Markov decision process
RL	reinforcement learning
T1DM	type 1 diabetes mellitus
TİTCK	The Turkish Medicines and Medical Devices Agency
TÜBITAK	Scientific and Technological Research Council of Turkey
TURDEP	The Turkish Diabetes Epidemiology Study

List of Publications

This thesis includes content from following publication(s):

1. A. A. Celik and C. Tekin, "Optimizing dynamic treatment regimes via volatile contextual gaussian process bandits," in *Reinforcement Learning for Real Life (RL4RealLife) Workshop in the 38th International Conference on Machine Learning*, 2021

Chapter 1

Introduction

The problem of sequential decision-making under uncertainty has been the subject of decades of study in statistics, operational research, electrical engineering, computer science, and economics. Thanks to the support of authorities, significant investments, interdisciplinary effort, access to good quality data, and a significant increase in computation power in the last few decades, people have started applying these theoretical studies to challenging real-life problems such as gaming, healthcare, self-driving cars, finance, news recommendation, and advertising. This dissertation is an interdisciplinary work where we apply multi-armed bandits, i.e., a mature sequential decision-making framework, on optimizing dynamic treatment regimes problem in personalized medicine. We primarily address the problem of glycemic control in type 1 diabetes mellitus by proposing VCGP-UCB algorithm. Moreover, we perform tests on *in-silico* subjects that come with open source implementation of a FDA-approved simulator. We also compare its performance against a clinician. Furthermore, we present a pilot study with a few clinicians and patients, where we design interfaces that doctors and users can interact with the model. Meanwhile, we address a few issues regarding privacy, safety, and ethics.

1.1 Personalized medicine

Treatment of chronic diseases requires long term commitment and adaptation to ever evolving conditions. As there is no one-size-fits-all approach, treatments must be adapted based on changing patient characteristics. Within this context, there has been a surge of interest in using machine learning techniques for identifying optimal personalized treatment regimes.

Since management of chronic diseases requires repeatedly making decisions, more data about the patient's response is accumulated over time. Moreover, data collected from the patient depends on the course of the treatment. Therefore, supervised learning methods—which require offline training data—are unable to produce accurate recommendations in the long run. As the patient characteristics evolve over time, treatment must be adjusted to maximize the benefit while minimizing the risks. This requires an intricate balance between exploration and exploitation. The best treatment under the current context must be identified with sequential experimentation while ensuring safety and efficiency at the same time.

In this work, we model optimization of dynamic treatment regimes as a volatile contextual Gaussian process (GP) bandit. The predictive power and non-parametric flexibility of GPs allow us to accurately model the relationship between treatment and response under different patient contexts. Moreover, sequential experimentation via upper confidence bounds constructed using the posterior mean and covariance functions of the GP allows us perform safe experimentation over a set of admissible treatments calculated based on the current context. Our framework allows flexibility in forming the set of admissible treatments, and enables safe experimentation among a set of treatments identified by clinical guidelines or baseline interpretable formula-based systems. In particular, we focus on using our framework for personalized bolus insulin dose recommendation for type 1 diabetes mellitus (T1DM) patients.

1.2 Glycemic control

According to International Diabetes Federation (IDF) Diabetes Atlas, the global prevalence of diabetes mellitus has consistently climbed up to double 20 years ago. In 2017, diabetes allocated a population size of 451 million, 5 million deaths, and USD 850 billion of healthcare expenditure. Diabetes explains 9.9% of the global morality among people aged 20-99 years [4]. Almost half of the patients are reported unaware or undiagnosed. Similarly, The Turkish Diabetes Epidemiology Study (TURDEP) says that the number of patients in Turkey tripled in 15 years from 2.5 million in 1998 to 7 million in 2013 at a staggering rate [5, 6]. IDF predicts this number will be 12 million around 2035. It is also reported that Turkey has the highest prevalence (14.8%) among Europe [7].

Diabetes is a chronic and incurable disease that has to be managed for a lifetime. Considering the complexity of the problem, we need a diverse set of alternatives from affordable to high-end to address these challenges. Clinical decision-support systems play a crucial role in this process. [8] touches out the surge of the ongoing integration of decision-support software for insulin bolus dosing to diabetes-related products. Authorities also publish consensus reports on the standards, regulations, technical and ethical challenges for Diabetes Digital App Technology [9]. All these developments motivate our work. Rest of this section, we explain diabetes mellitus and give a literature review for diabetes control systems.

1.2.1 Diabetes mellitus

There are two major types of diabetes mellitus, namely, type 1 and type 2. Type 1 diabetes is a chronic autoimmune disease characterized by insulin deficiency due to pancreatic β cell loss. As a result, the body can no longer produce the insulin it needs. Patients with this form of diabetes need to regulate their blood glucose (BG) by regularly administering bolus insulin, generally close to their meal intakes. Lack of insulin regulation in diabetic patients can have serious adverse effects due to hypoglycemia and hyperglycemia, i.e., low and very high BG levels, respectively, which might result in immediate hospitalization or even death, or long term damage to various organs at risk unless effectively treated. Some of the long-term complications are cardiovascular disease, retinopathy, and nephropathy.

Type 2 diabetes, on the other hand, is characterized by insufficient insulin production due to abnormal pancreatic β cell function or insulin resistance. The body produces insulin, but either it is inadequate or unable to respond to its effects. As a consequence, excess glucose levels accumulate in the plasma. According to IDF, almost half of them are unaware of their illness since symptoms may take years to develop [4]. Usually, they do not need regular insulin therapy to survive since they can produce enough insulin to avoid ketoacidosis. In most cases, a healthy diet and physical activity can get them manage their illness. If not, regular exogenous insulin administration can address their lack of blood glucose regulation.

Insulin regulation in diabetic patients is generally a complicated process, since the optimal insulin dose depends on a variety of exogenous and endogenous contexts such as time of the day, pre-meal BG level (aka fasting BG level), carbohydrate content of the meal, basal insulin levels, active insulin remaining from previous doses (aka insulin on board), hormone cycle, past and future physical activity and its intensity, acute diseases, comorbidity, alcohol intake, menstruation, pregnancy, Hemoglobin A1c level (i.e. HbA1c), cholesterol level, blood pressure at rest, family health history of diabetes, weight, age, gender, etc., some of which could have missing values.

When we compare both types, we see that type 1 diabetes accounts for 5-10% of the diabetes population whereas type 2 diabetes 90-95% [10]. Type 1 diabetes is also more common amongst young children; in contrast, type 2 diabetes among adults. In this dissertation, we focus on type 1 diabetes due to the availability of UVa/Padova T1DM simulation environment and *in-silico* patients.

1.2.2 Literature review

This section reviews automatic glycemic control methods in diabetes mellitus. In the literature, there are mainly three categories of models for BG regulation: standard formula-based bolus insulin calculators [11], closed-loop control systems [12] and models based on reinforcement learning [13].

Standard formula-based bolus insulin calculators are widely used to calculate pre-meal bolus insulin dose thanks to their simplicity and interpretability [14]. An example is

$$\text{Bolus insulin} = \left(\frac{\text{CHO}}{\text{ICR}} + \frac{\text{G}_M - \text{G}_T}{\text{CF}}\right)^+ \tag{1.1}$$

where $(a)^+ = \max\{a, 0\}$, CHO (g) the estimated amount of carbohydrate intake, ICR (g/U) is the insulin-to-carbohydrate-ratio, G_M (mg/dl) is the measured meal time BG, G_T (mg/dl) is the target BG and CF (mg/dl/U) is the insulin correction factor [11]. While standard bolus insulin calculators serve as a transparent and interpretable baseline, they do not achieve optimal glycemic control due to the inability to capture the complexity of glucose metabolism and ignoring other contextual variables which might have an effect on post-meal BG levels [11]. This leads to miscalculations with regard to the appropriate insulin dosage.

Case-Based Reasoning (CBR) is another common framework to build calculators [15], also called adaptive or advanced bolus calculators. CBRs provide new recommendations based on past observations by estimating ICR and/or CF according to user context and computing the intended bolus dose via the standard bolus calculator. They can be thought of as contextual and adaptive versions of the standard calculator. To the best of our knowledge, some of the leading CBR works are [16, 17, 18]. Authors of [17, 18] also proposed a software system architecture, named *advanced insulin bolus calculator for diabetes* [19], to deploy their models as a bolus advisory system. They share a usability survey and a six-week-long clinical pilot study [20]. However, CBRs based models have some fundamental drawbacks. (i) They require a CGM device to monitor BG level since their update mechanism uses series of BG measurements within a given time frame after meal intake to compute the area under the BG curve and/or minimum BG level. Whereas our work only requires two measurements per meal, namely fasting BG and postprandial BG. (ii) They suffer from the cold-start problem, which is still not fully solved yet. They require a couple of weeks long, good-quality personalized data to start with. On the other hand, our work only needs a single sample to initialize and start its recommendations. (iii) They fail to extrapolate treatment effects beyond the range of past observations effectively. Their recommendations are a simple function of closest neighbors from past observations. In the case of a new query dissimilar to past observation, their recommendations may not be reliable. Our work can compute confidence scores for its recommendations, visualize its treatment policies, and provide insights into various what-if scenarios regarding the treatment. Details are further discussed in Section 3.3. (iv) CBR methods assume linearity and independence for context variables. They compute the distance between observations as the weighted average of absolute or Euclidean differences of their Cartesian coordinates. The weights signify the degree of relevance and are set by a domain expert. In reality, context variables can be highly non-linear and co-dependent. For instance, variable time of the day has a periodic nature due to hormonal cycles. Alternatively, variables fasting BG and physical activity intensity can be correlated. CBRs are insufficient to capture such rich relationships. Our work models similarity as covariance/kernel functions, i.e., a rich set of parametric functions whose parameters can be learned from data via maximum likelihood estimation. It is also possible to learn any custom covariance function themselves that explains the relationship in data [21]. (v) Their sequential update mechanism has no theoretical analysis to ensure convergence to optimal glycemic control over time. However, our work provides a theoretical guarantee for convergence to near-optimal glycemic control, as stated in Section 3.4.

Closed-loop controllers. A different line of research exists under the name *artificial pancreas*. It refers to the closed-loop control of BG in diabetes, which is usually achieved by subcutaneous CGM and insulin infusion. Within this context, control-theoretic approaches such as proportional-integral-derivative (PID) [22] and model predictive control (MPC) [23] have been widely investigated. In recent years, reinforcement learning (RL) based control models are also developed [13]. [24] consider V-learning for estimating dynamic treatment regimes using a parametric class of policies, where exploration can be achieved using ϵ-greedy style randomization. They use V-learning to estimate an optimal treatment policy for T1DM patients. Alternatively, [25] studies the same problem using Deep Q-network with GRU and 1-D CNN and it achieves favorable results against PID methods. It is the only comparable glycemic control method against our work, reported in deep learning for diabetes survey [26]. Most of these methods are designed for use with insulin-pump therapy and CGM—tools which may not be readily available for most diabetic patients. In contrast, our algorithm provides recommendations tailored for traditional BG regulation, which can be performed by using finger-stick BG meters and multiple daily bolus insulin injections, and does not require access to expensive medical equipment. Compared to our work, most of other RL-based solutions are not interpretable due to their black-box nature, and they do not come with rigorous performance analysis.

Markov decision processes. Another related strand of literature makes use of Markov decision processes (MDPs) for clinical decision-making [27]. MDPs are especially suitable for problems in which past and present outcomes in the course of treatment are highly correlated. Within the context of diabetes management, using MDPs is more suitable for insulin-pump therapy with continuous glucose monitoring (CGM), where insulin is administered continuously over time. On the other hand, in this work, we focus on diabetes treatment with multiple daily injections (e.g., 3-4 times a day). Since bolus insulin is rapid acting, its effect mostly wears out before the next meal. Note that it is up to patients to inject the recommended amount to themselves or not, unlike CGMs, where patients have to take small but frequent bolus doses. Moreover, our context includes fasting BG, which efficiently captures residual effect of the previous insulin dose. Thus, conditioned on the context, outcome of each dose can be regarded as i.i.d., and hence, contextual bandits are a better fit for our problem. Indeed, bandit algorithms can be applied in other dynamic treatment assignment and drug dosage problems, where the effect of treatment significantly decays before the next decision-epoch.

1.3 Multi-armed bandits

The multi-armed bandit (MAB) is a sequential decision-making paradigm under uncertainty, initially proposed by Robbins [28]. The learning agent sequentially picks an arm from an action set and receives random rewards. Its objective is to maximize its total reward while the reward distribution of the arms is not known. This means the agent, over rounds, has to balance between exploration and exploitation to identify arms with high rewards without loosing too much time on arms with low rewards [29]. The metric used to evaluate the performance of the agent is regret. It is defined as the total sum of the differences between rewards earned by an oracle always choosing the optimal action and the rewards of the learning agent.

1.3.1 Contextual bandits

An important extension to classic MAB is the contextual MAB (CMAB) model [30, 31]. In this setup, the learning agent observes a side-information, aka context, then makes an arm selection. Like MAB, its objective is to maximize its total reward by using the information in the context. Its regret is computed as the total sum of the differences between rewards earned by an oracle, always choosing the optimal action given the context and the rewards of the learning agent. Prior works on contextual bandits such as [32] and [33] mostly consider the problem within the context of online recommender systems.

1.3.2 Volatile (sleeping) bandits

Volatile bandit setup introduces dynamic arm availability in the MAB problem [34]. In each round, the learning agent can only choose among the arms available/awake for that round, which is a time-varying subset of the action set. The agent tries to maximize its total reward by making its best to choose as many available high-reward arms as possible. Generally, MAB algorithms assume the availability of all arms at all times. However, this assumption does not hold for many applications. For example, items can be out of stock in e-commerce problems, or specific treatments may not be available in personalized dose recommendation problems.

1.3.3 Gaussian process bandits

GP bandits are first introduced with bounds on the regret that depend on information gain in [35], followed with a contextual version [36]. GP bandit models have recently gained attention due to their trustable nature as the posterior covariance can fully capture the uncertainty in recommendations. Likewise, their performance can be theoretically proven via a rigorous regret analysis. These aspects are critical in healthcare applications, in which the prevention of unwanted consequences during treatment is of the highest priority. Our work builds on the formalism of contextual bandits and GP bandits. It allows us to recommend doses adapted to both patient and meal event characteristics.

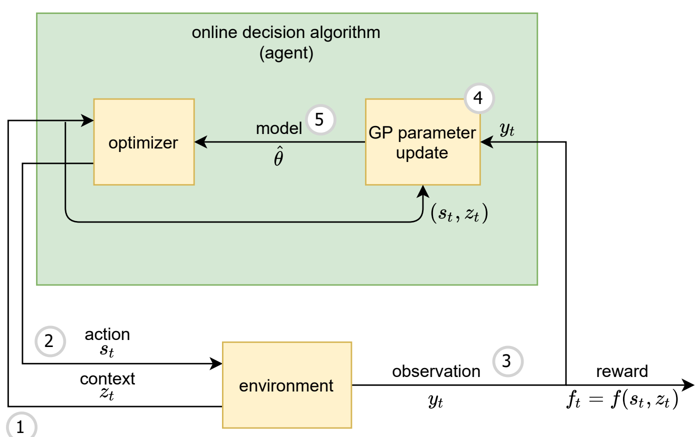


Figure 1.1: Illustration of contextual Gaussian process bandit algorithm.

Figure 1.1 describes the contextual Gaussian process bandits. The learner observes a side-information, aka context, then makes an arm selection. Consequently, the learner gets observation from the environment. Then, the algorithm uses this as feedback; inside, it updates its beliefs by learning the selected covariance function parameters and reports new model parameters to the online optimizer. As the iterations go on, regret diminishes to zero.

1.4 Our contribution

Our contributions are two fold. First, we adapt contextual Gaussian Process Upper Confidence Bound (CGP-UCB) algorithm given in [36] to the volatile arm setup, and call this adaptation VCGP-UCB. To better highlight the contribution, we share the acquisition functions and regrets for both CGP-UCB and VCGP-UCB by using the notation introduced in Section 3.2. S denotes the set of all treatments (finite), $S_t \subseteq S$ denote the set of admissible treatments in round t. CGP-UCB has regret of $\tilde{O}(\sqrt{T\gamma_T})$ and uses the acquisition function in Eq. 1.2. In contrast, VCGP-UCB has regret of $\tilde{O}(\sqrt{T\gamma_T^{vol}})$ and uses the acquisition function in Eq. 1.3. Our setup is general in the sense that it allows S_t to be a function of context in round t. Since only a subset of treatments are available in each round, this setup corresponds to a bandit problem with volatile arms.

$$s_t = \operatorname*{argmax}_{s \in S} \mu_{t-1}(s, z_t) + \beta_t^{1/2}\sigma_{t-1}(s, z_t), \tag{1.2}$$

$$s_t = \operatorname*{argmax}_{s \in S_t} \mu_{t-1}(s, z_t) + \beta_t^{1/2}\sigma_{t-1}(s, z_t), \tag{1.3}$$

For any admissible sequence of treatments, we show that the regret of VCGP-UCB is $\tilde{O}(\sqrt{T\gamma_T^{vol}})$, where γ_T^{vol} represents the volatility-adapted maximum information gain. This term is always less than or equal to the unrestricted maximum information gain γ_T in [36]. Therefore, for typical covariance functions such as squared-exponential or Matern, regret growth is sublinear in time. In terms of diabetes treatment, this implies that the average number of decision-epochs in which a suboptimal treatment is suggested converges to zero. To the best of our knowledge, this work is the first to apply Bayesian bandit strategies for optimal BG control.

The second half of our contribution is focused on automatic BG control in diabetes mellitus patients. We apply our algorithm in the optimal glycemic control problem for type 1 diabetes subjects. We test it on type 1 *in-silico* subjects and compare it against a clinician. We also did a pilot study with a few patients and doctors where we designed interfaces for doctors and patients to interact

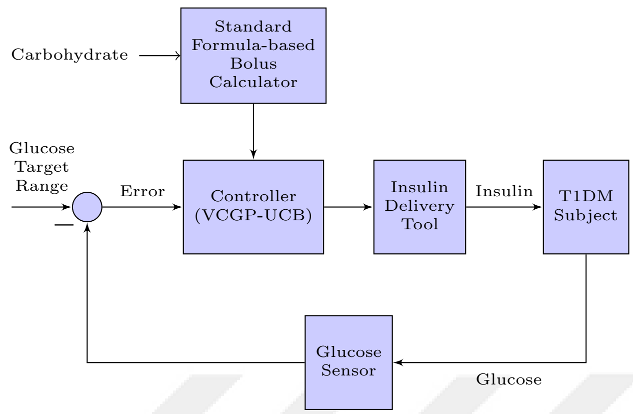


Figure 1.2: Our system model. We use VCGP-UCB to perform safe experimentation around the recommendation produced by a standard formula-based bolus calculator.

with the model and deployed the model into a platform. Our algorithm can work with a finger-stick BG meter and a smartphone/computer and does not require access to expensive medical equipment. For a given meal, it only requires two BG measurements, namely pre-meal and post-meal BG measurements. Our model can be deployed either on the client or server side. Alternatively, it can be integrated into any decision-support system and scaled via the Internet.

Table 1.1: Comparison with the related bandit models.

Properties	This work	GP-UCB [35]	CGP-UCB [36]	SAFE-OPT [37]	STAGE-OPT [38]	Safe-LUCB [39]	SGP-UCB [40]
Contextual	Yes	No	Yes	No	No	No	No
GP prior	Yes	Yes	Yes	Yes	Yes	No	Yes
Arm volatility	Exogenous†	No	No	Endogenous‡	Endogenous‡	Endogenous‡	Endogenous‡
Arm set	Finite	Can be infinite	Can be infinite	Finite	Can be infinite	Can be infinite	Finite
Context set	Can be infinite	-	Can be infinite	-	-	-	-
Information gain term in the regret bound	Volatility-adapted maximum	Maximum	Maximum	-	Maximum	-	Maximum

†Volatility induced by an exogenous process (e.g., context arrivals). ‡Volatility induced by safety constraints.

We provide a detailed comparison of our work with related works in bandits and BG control for diabetes in Tables 1.1 and 1.2.

Table 1.2: Comparison with the diabetes treatment models.

Properties	This work	Standard bolus calculator [11]	Adaptive bolus calculator [18]	Bio-inspired artificial pancreas [41]	Actor-Critic [42]
Model	GP bandits	Formula-based	CBR & R2R†	Bio-modeling via DEs‡	Actor-Critic RL
Adaptiveness (to patient response)	High	Limited	High	Limited	High
Theoretical performance (regret) guarantees	Yes	No	No	No	No
Safe exploration	Yes	No	No	No	No

†CBR: Case-Based Reasoning, R2R: Run-To-Run control. ‡DE: Differential Equation.

1.5 Organization

The rest of the thesis is organized as follows. In Chapter 2, we explain our evaluation environment UVa/Padova T1DM Simulator and our modifications along the way. Chapter 3 formalizes the VCGP-UCB problem by introducing notations and assumptions, providing the learning algorithm, and analyzing its regret. In Chapter 4, we provide experimental results and their limitations. Chapter 5 focuses on the pilot study, provides user interfaces and explains how users interact with the model without ethical misconduct, safety, or privacy issues. Chapter 6 concludes the theses.

Chapter 2

UVa/Padova T1DM Simulator

We use UVa/Padova type 1 diabetes mellitus simulator as our evaluation environment. The simulator is a product of more than decade-long research done at the University of Padova, Italy, and the University of Virginia, USA. It is a glucose-insulin model created as an *in-silico* environment for closed-loop hormone controller design, testing, and validation. It was approved by U.S. FDA in 2013 as a reliable framework to substitute certain pre-clinical trials. We test our algorithm on *in-silico* subjects that come with the open source implementation of the simulator.

For the time being, there are three main versions: S2008 [43], S2013 [44] and S2017 [45]. S2008 and S2013 model single meal scenario while S2017 models single day of three meals by incorporating intraday variability of insulin sensitivity. Due to the lack of availability of the parameters of *in-silico* subjects, we carried on our evaluations with S2008. We use 30 virtual patients (10 for each age group, adult, adolescent, and child) that come with the simulator [1]. We have added more details regarding the model, its variables, and parameters in Appendix A.

We explain how the simulator works and some modifications we have done along the way.

2.1 Preliminaries

The simulator takes a set of inputs and produces a time series of BG levels. The inputs are (1) *in-silico* subject id (e.g., *adult#001*), (2) fasting BG level (mg/dl), (3) carbohydrate intake amount (g) and time (min) relative to the beginning of the simulation, (4) insulin intake amount (U) and time (min) relative to the beginning of the simulation and (5) duration for simulation (hours), i.e., how long of a subject's BG levels should be returned. The output is subcutaneous blood glucose levels. In Fig. 2.1, we provide a sample output for the subject *adult#002* with fasting BG of 150. It has 60g carbohydrate as its meal intake at min 60. We immediately inject 10 U of bolus insulin. The output does not contain any measurement noise, when necessary, we add various Gaussian noise later on in the experiments.

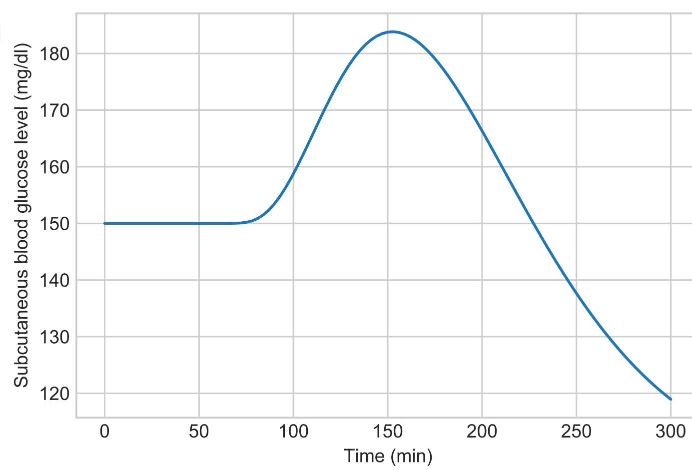


Figure 2.1: Sample output for the T1DM simulator.

As the simulator is only reliable for a single meal scenario, we do separate runs for each carbohydrate intake event. For a single run, we return a 5-hours-long sequence of BG levels of a virtual patient with 3 min discretization. During the run, we process the meal intake always at the same time, one hour immediately after the beginning of the simulation. Insulin can be injected before, during, or after the meal intake.

2.2 Modifications

We list several issues and our proposed solutions regarding *simglucose* the UVa/Padova 2008 T1DM simulator.

Custom Insulin Controller. The simulator simulates an insulin pump for a given carbohydrate intake amount, which automatically computes and injects basal and bolus insulin doses. Basal dose in U/min is computed as $u2ss * BW/6000$ where $u2ss$ and BW denote steady-state insulin rate per kg (pmol/(L*kg)) and body weight in kg, respectively and 1 U/min = 6000 pmol/L. Values of $u2ss$ for all *in-silico* patients come with the simulator. Similarly, bolus dose is computed via the standard formula-based bolus calculator. The simulator supports Cozmo and Insulet brand pumps. In our case, we want the model to use our insulin recommendations. Thus, we implemented a custom insulin controller, which uses our recommended bolus insulin amount. We run our simulations with this custom bolus controller. Note that we use the same basal dose formula as before.

Fasting BG. The simulator assigns a different and constant initial BG level for each of the 30 virtual subjects. BG levels for all the subjects start with these initial levels and change depending on patient parameters, carbohydrate, and insulin intake amounts and time. We modify the simulator to take a custom initial BG level as an input. In this way, blood glucose levels also depend on the initial BG level. Note that the simulator only supports single-meal scenarios, so we run a separate simulation for each meal event. That is, initial BG levels act as fasting BG levels since they are the pre-meal BG measurement. Therefore, we can use fasting BG as a context variable for our recommendations.

In the set of model equations given in [2], plasma glucose concentration G has a initial condition denoted as G_b. G_b represents the initial BG level in these equations. We set this variable to a custom value before the simulations start and update all other initial conditions that depend on the value of G_b. We follow the sequence of modifications given below. All the variables and parameters are named in Appendix A.

1. The glucose mass in plasma, G_p, has initial condition denoted as G_{pb}. It is updated as $G_{pb} = G_b * V_G$ by (A1) in [2].

2. The endogenous glucose production, EGP, has initial condition denoted as EGP_b. It is updated as $EGP_b = k_{p1} - k_{p2} * G_{pb} - k_{p3} * I_d$ where $I_d = I_b$ by (A3) and (A4) in [2].

3. The glucose mass in slowly equilibrating tissues, G_t, has initial condition denoted as G_{tb}. It is updated as $G_{tb} = (F_{snc} - EGP_b + k_1 * G_{pb})/k_2$ by (A20) in [2].

4. The Michaelis Menten parameter of glucose utilization at zero insulin action, V_{m0}, is a constant for a virtual subject. It is updated as $V_{m0} = (EGP_b - F_{snc}) * (K_{m0} + G_{tb})/G_{tb}$ by (A21) in [2].

Subcutaneous Glucose Sensor. There are various types of glucose sensors: intravenous, continuous glucose monitoring, and subcutaneous glucose sensor. The open source implementation of UVa/Padova 2008 Simulator [1] only supports continuous glucose sensors of Dexcom, GuardianRT, and Navigator. They add some unique noise, which depends on sensor parameters, on top of the subcutaneous blood glucose signal. However, since we want to simulate a subcutaneous glucose sensor (i.e., finger-stick BG meter) in our setup, we directly use noise-free subcutaneous glucose levels and, when necessary, add various Gaussian noise later on in the experiments. In reality, the measurement noise is highly device-specific, and it may have high variance. Nonetheless, commercially available BG meters have small enough noise not to cause a problem. We further address this issue in Section 4.4.

Subcutaneous glucose level, $G_s(t)$, is defined as [44]

$$\dot{G}_s(t) = -\frac{1}{T_s} G_s(t) + \frac{1}{T_s} G(t) \tag{2.1}$$

where G and $k_s := 1/T_s$ denote plasma glucose concentration and rate constant of glucose absorption, respectively. The values of k_s come with the simulator.

Chapter 3

VCGP-UCB Algorithm

3.1 Problem Formulation

Let S denote the set of all treatments (finite), Z denote the set of all patient contexts (can be infinite) and T denote the number of iterations. For $t \in [T] := \{1, \ldots, T\}$, let z_t be the patient context and $S_t \subseteq S$ be the set of admissible treatments in round t. Our setup is general in the sense that it allows S_t to be a function of z_t. Since only a subset of treatments are available in each round, this setup corresponds to a bandit problem with volatile arms. We consider the problem of sequentially optimizing effectiveness of the given treatments, which is characterized by an unknown reward function $f : S \times Z \to \mathbb{R}$. Within the context of bolus insulin recommendation, reward measures closeness of postprandial BG to target BG (see Section 4 for details).

At the beginning of reach round t, the learner receives a patient context $z_t \in Z$, chooses a treatment $s_t \in S_t$, and then, observes a noisy reward $y_t = f(s_t, z_t) + \epsilon_t$, where ϵ_t denotes the zero mean Gaussian noise with σ^2 variance, independent across the rounds. The objective is to maximize the cumulative reward $\sum_{t=1}^{T} f(s_t, z_t)$ without knowing f beforehand. The optimal treatment in round t is denoted by $s_t^* = \text{argmax}_{s \in S_t} f(s, z_t)$. Suboptimality of the treatment in round t is given as $r_t = f(s_t^*, z_t) - f(s_t, z_t)$, which is also called the instantaneous regret. The cumulative regret is the sum of instantaneous regrets, i.e., $R_T = \sum_{t=1}^{T} r_t$. It is well known that maximizing the cumulative reward is equivalent to minimizing the cumulative regret.

Let $X = S \times Z$ denote the set of treatment-context pairs. We assume that f is sampled from a known Gaussian process $GP(\mu, k)$ which is fully characterized by its mean function $\mu : X \to \mathbb{R}$, $\mu(x) = \mathbb{E}[f(x)]$ and covariance (kernel) function $k : X \times X \to \mathbb{R}, k(x, x') = \mathbb{E}[(f(x) - \mu(x))(f(x') - \mu(x')))]$. We assume that $\mu \equiv 0$ and that the variance is bounded, i.e., $k(x, x) \leqslant 1$ for all $x \in X$. Given an observation history $\mathcal{A}_t = \{x_1, \ldots, x_t\}$, where $x_t = (s_t, z_t)$, the posterior mean and variance of the GP at point x can be calculated as follows [46]:

$$\begin{aligned}
\mu_t(x) &= \boldsymbol{k}_t(x)^T (\boldsymbol{K}_t + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{y}_t, \\
k_t(x, x') &= k(x, x') - \boldsymbol{k}_t(x)^T (\boldsymbol{K}_t + \sigma^2 \boldsymbol{I})^{-1} \boldsymbol{k}_t(x'), \\
\sigma_t^2(x) &= k_t(x, x),
\end{aligned}$$

where $\boldsymbol{k}_t(x) = [k(x_1, x), \ldots, k(x_t, x)]^T$, $\boldsymbol{K}_t$ is the kernel matrix $[k(x, x')]_{x, x' \in \mathcal{A}_t}$ and $\boldsymbol{y}_t = [y_1, \ldots, y_t]^T$.

3.2 The Learning Algorithm

As our learning algorithm, we use contextual Gaussian Process (CGP) algorithm in [36] by adapting it to work under the volatile setting. Its pseudocode is given in Algorithm 1. Basically, at round t, VCGP-UCB selects treatment

$$s_t = \operatorname*{argmax}_{s \in S_t} \mu_{t-1}(s, z_t) + \beta_t^{1/2} \sigma_{t-1}(s, z_t), \tag{3.1}$$

where β_t is a non-decreasing function of t (which will be specified later). The selected treatment is the one with the highest upper confidence bound (UCB) index among all admissible treatments in round t. Within the context of bolus insulin dosage for diabetes treatment S_t is set by defining a safe experimentation region around the treatment suggested by a formula-based bolus calculator.

Algorithm 1 VCGP-UCB algorithm

1: **Input:** Input space X; GP prior $\mu_0 = 0$, σ_0, k
2: **for** $t = 1$ to T **do**
3: Observe context z_t, and the set of admissible treatments S_t
4: Choose $s_t = \text{argmax}_{s \in S_t} \mu_{t-1}(s, z_t) + \sqrt{\beta_t}\sigma_{t-1}(s, z_t)$
5: Observe $y_t = f(s_t, z_t) + \epsilon_t$
6: Update GP posterior to obtain μ_t and σ_t
7: **end for**

3.3 Postprandial BG predictions based on GPs

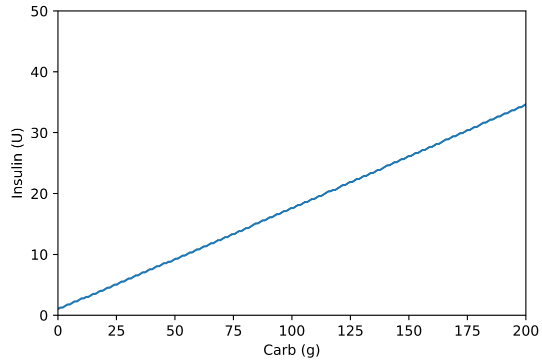
Treatment recommendations can be accompanied by postprandial BG predictions, by using another GP as a surrogate model for postprandial BG surface $g : S \times Z \to \mathbb{R}$. In this case, each noisy postprandial BG measurement is represented by $g_t = g(s_t, z_t) + \tilde{\eta}_t$, where $\tilde{\eta}_t$ represents the zero mean Gaussian measurement noise with known variance.

The GP posterior for g opens up a plethora of opportunities to get valuable insights with regard to treatment by extrapolating treatment effects beyond the range of data. They can be used as health analytics in clinical decision support systems to better understand patient's physiological response for various what-if scenarios related with treatment. We accomplish this through computing certain probabilities and visualizing relations between any context variables Z, treatment S and the GP response variable. Z contains a set of variables that explains the effects of treatment on the response variable; thus, they are selected by a domain expert. In our work, we use single response variable as opposed to GPs with multiple response variables.

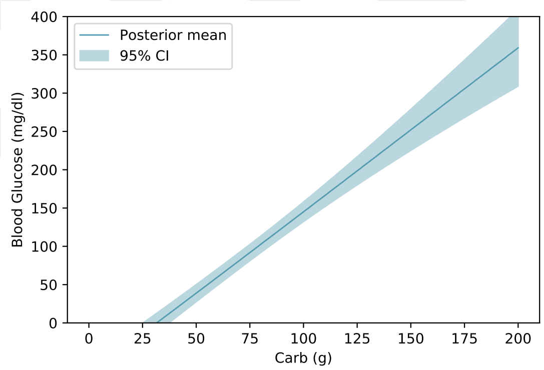
Considering our choice of the subject matter, we demonstrated these insights in diabetes by using *in-silico* T1DM subject *adult#003* from UVa/Padova T1DM 2008 Simulator. The context variables are amount of carbohydrate intake, fasting BG level, time between meal and postprandial BG measurement, and time between meal and insulin intake. The treatment and response variables are bolus insulin dose and postprandial BG measurement, respectively. We use the data of VCGP-UCB SimCalc and same kernel parameters given in Section 4.1. We set the noise variance to 25 and the GP response variable is postprandial BG as a replacement for *loss*.

We can compute probabilities of euglycemia (i.e. target BG range is $[70, 180]$ mg/dl), hyperglycemia (i.e. target BG range is > 180 mg/dl) and hypoglycemia (i.e. target BG range is < 70 mg/dl) events. As a matter of fact, we can compute probabilities for any given range of postprandial BG target as it is explained in Remark 1. One relevant use case may be to provide the probability of euglycemia with the treatment recommendation as a confidence value. To illustrate, assume that we have a context of 120 g carb, 150 min measurement time, and 130 mg/dl fasting BG; and we consider injecting 24 U of bolus insulin. The probabilities of euglycemia, hyperglycemia, and hypoglycemia are 0.61, 0.00, and 0.39, respectively. Likewise, for postprandial BG range of $[160, 210]$, the probabilities of within, above and below range are 0.00, 0.00, and 1.00, respectively. Moreover, we can also draw three kinds of relations: (a) the relation between any one of the context variables and treatment, (b) the relation between a context variable and response variable, and (c) the relation between treatment and response variable. All three relations are illustrated in Fig 3.1, respectively.

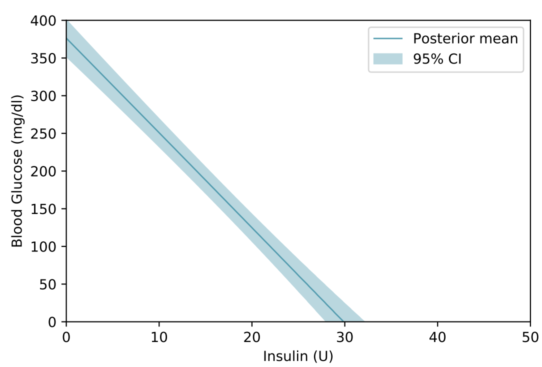
Remark 1. *Given a treatment-context pair (s_t, z_t) and BG target range (g_{low}, g_{high}), probability of within, below and above the range events, $P(g_{low} < g(s_t, z_t) < g_{high})$, $P(g(s_t, z_t) < g_{low})$ and $P(g(s_t, z_t) > g_{high})$ respectively, can be computed from posterior distribution with mean $\tilde{\mu}_{t-1}(s_t, z_t)$ and variance $\tilde{\sigma}^2_{t-1}(s_t, z_t)$ at round t where $g(s_t, z_t) \sim \mathcal{N}(\tilde{\mu}_{t-1}(s_t, z_t), \tilde{\sigma}^2_{t-1}(s_t, z_t))$. Suppose that $X \sim \mathcal{N}(0, 1)$ and F_X denotes cumulative distribution function of random variable X. Thus, $P(g(s_t, z_t) < g_{low}) = F_X(\frac{g_{low} - \tilde{\mu}_{t-1}(s_t,z_t)}{\tilde{\sigma}_{t-1}(s_t,z_t)})$, $P(g(s_t, z_t) > g_{high}) = 1 - F_X(\frac{g_{high} - \tilde{\mu}_{t-1}(s_t,z_t)}{\tilde{\sigma}_{t-1}(s_t,z_t)})$ and $P(g_{low} < g(s_t, z_t) < g_{high}) = F_X(\frac{g_{high} - \tilde{\mu}_{t-1}(s_t,z_t)}{\tilde{\sigma}_{t-1}(s_t,z_t)}) - F_X(\frac{g_{low} - \tilde{\mu}_{t-1}(s_t,z_t)}{\tilde{\sigma}_{t-1}(s_t,z_t)})$.*



(a) Carb vs. insulin relation given that postprandial BG is 112.5 mg/dl.



(b) Carb vs. postprandial BG relation given that insulin amount is 15 U.



(c) Insulin vs. postprandial BG relation given that carb amount 120 g.

Figure 3.1: Relational insights obtained from the GP posterior for g. Time between meal and postprandial BG measurement, time between meal and insulin intake, and fasting BG are kept as 150 min, 0 min, and 130 mg/dl respectively.

3.4 Regret Analysis

In this section, we bound the regret of VCGP-UCB. A regret bound that grows sublinearly over time, i.e., $o(T^\gamma)$ for $\gamma < 1$ implies that the treatment recommendations made by VCGP-UCB are as good as the optimal sequence of treatments for the given contexts in the long-run.

3.4.1 Main Regret Bound

Consider a fixed sequence of patient contexts $\boldsymbol{z}_T = [z_1, \ldots, z_T]$. Let $X_t = (z_t, S_t)$ and Let $\boldsymbol{X}_T = X_1 \times \cdots \times X_T$ represent the Cartesian product of all admissible treatment-context pairs up to round T. For a given sequence of treatment-context pairs A, let $\boldsymbol{y}_A$ denote the $|A|$-dimensional vector whose ith element corresponds to the reward observed for the ith treatment-context pair in A. The quantity governing our regret bounds is the volatility-adapted maximum information gain after T rounds (given $\boldsymbol{z}_T$), which is defined as

$$\gamma_T^{vol} = \max_{A \in \boldsymbol{X}_T} I(\boldsymbol{y_A}; \boldsymbol{f_A}) \ ,$$

where $\boldsymbol{f_A} = [f(x)]_{x \in A}$ and $I(\boldsymbol{y_A}; \boldsymbol{f_A})$ is the mutual information between f and reward observations at points in A.

First, we state our main theorem. Its proof is built on the proofs by [35] and [36].

Theorem 1. *Fix $\delta \in (0,1)$. When volatile CGP-UCB is run with $\beta_t = 2log(|S|t^2\pi^2/6\delta)$, for an arbitrary fixed sequence of contexts $\boldsymbol{z}_T$, we have*

$$Pr\{R_T \leqslant \sqrt{C_1 T \beta_T \gamma_T^{vol}} \quad \forall T \geqslant 1\} \geqslant 1 - \delta \ ,$$

where $C_1 = 8/\log(1 + \sigma^{-2})$.

The above theorem provides an information-type regret bound for volatile CGP-UCB. The maximum information gain γ_T^{vol} depends on both the context sequence and the set of admissible treatments. This contrasts with the information gain terms defined in [35] and [36], which for our setting can be written as

$$\gamma_T = \max_{A \in \tilde{\boldsymbol{X}}_T} I(\boldsymbol{y_A}; \boldsymbol{f_A}) ,$$

where $\tilde{\boldsymbol{X}}_T = \times_{t=1}^{T} X$ is the Cartesian product of T copies of $X = S \times Z$. It is obvious that $\gamma_T^{vol} \leqslant \gamma_T$, and in practice γ_T^{vol} might be much smaller than γ_T. The reason for this is that we are constrained in each round to pick a treatment from the set of admissible treatments S_t instead of S, and we incur regret only when the selected treatment differs from the best available treatment of that round. Therefore, sublinear regret bounds in [35] and [36] obtained for squared exponential and Matern kernels also hold in our setting.

As a result, Theorem 1 shows that when such kernels are used, time-averaged regret given as R_T/T converges to zero with high probability. In the context of BG regulation, this implies that our algorithm's insulin recommendations converge to the optimal insulin recommendations over time.

3.4.2 Proof of Theorem 1

Lemma 1. *(Lemma 5.1 in [35]) Fix the sequence of contexts $\boldsymbol{z}_T$. Let $\delta \in (0,1)$ and $\beta_t = 2log(|S|\pi_t\delta)$, where $\sum_{t\geqslant 1} \pi_t^{-1} = 1$, $\pi_t > 0$. Then, the following event holds with probability at least $1-\delta$.*

$$|f(s, z_t) - \mu_{t-1}(s, z_t)| \leqslant \beta_t^{1/2}\sigma_{t-1}(s, z_t) \quad \forall s \in S, \forall t \geqslant 1 .$$

Lemma 2. *Fix $t \geqslant 1$. If $|f(s, z_t) - \mu_{t-1}(s, z_t)| \leqslant \beta_t^{1/2}\sigma_{t-1}(s, z_t)$ for all $s \in S_t$, then the instantaneous regret r_t is bounded by $2\beta_t^{1/2}\sigma_{t-1}(x_t)$.*

Proof. Let $s_t^* \in \text{argmax}_{s\in S_t} f(s, z_t)$ be an optimal action. By definition of s_t : $\mu_{t-1}(s_t, z_t) + \beta_t^{1/2}\sigma_{t-1}(s_t, z_t) \geqslant \mu_{t-1}(s_t^*, z_t) + \beta_t^{1/2}\sigma_{t-1}(s_t^*, z_t) \geqslant f(s_t^*, z_t)$, where the last inequality is due to Lemma 1. Therefore, $r_t = f(s_t^*, z_t) - f(s_t, z_t) \leqslant \beta_t^{1/2}\sigma_{t-1}(s_t, z_t) + \mu_{t-1}(s_t, z_t) - f(s_t, z_t) \leqslant 2\beta_t^{1/2}\sigma_{t-1}(s_t, z_t)$. □

Lemma 3 (Lemma 5.3 in [35])**.** *Given $\boldsymbol{z}_T$, the information gain for the points selected can be expressed in terms of the predictive variances. If $\boldsymbol{f_T} = (f(x_t)) \in \mathbb{R}^T$:*

$$I(\boldsymbol{y}_T; \boldsymbol{f}_T) = \frac{1}{2}\sum_{t=1}^{T} \log(1 + \sigma^{-2}\sigma_{t-1}^2(x_t)).$$

Lemma 4. *Fix the sequence of contexts $\boldsymbol{z}_T$. Pick $\delta \in (0,1)$ and let β_t be defined as in Lemma 1.1. Then, the following holds with probability at least $1-\delta$:*

$$R_T \leqslant \sqrt{T\beta_T C_1 I(\boldsymbol{y}_T; \boldsymbol{f}_T)} \leqslant \sqrt{TC_1\beta_T\gamma_T^{vol}} \quad \forall T \geqslant 1 \ ,$$

where $C_1 = 8/\log(1+\sigma^{-2})$.

Proof. The proof is similar to the proof of Lemma 5.4 in [35]. By Lemmas 1.1 and 1.2, we have that

$$Pr\{r_t^2 \leqslant 4\beta_t\sigma_{t-1}^2(x_t) \quad \forall t \geqslant 1\} \geqslant 1-\delta \ .$$

Also note that

(i) β_t is non-decreasing in t, and thus, $\beta_T \geqslant \beta_t$ for $\forall T \geqslant t$.

(ii) Note that $\sigma^{-2}\sigma_{t-1}^2(x_t) \leqslant \sigma^{-2}k(x_t,x_t) \leqslant \sigma^{-2}$, where $k(x_t,x_t) \leqslant 1$ due to assumption of bounded variance. Let $C_2 = \sigma^{-2}/\log(1+\sigma^{-2})$. Note that $C_2 \geqslant 1$ and $\frac{s^2}{\log(1+s^2)} \leqslant C_2$ for $s^2 \in [0,\sigma^{-2}]$.

(iii) By cannons i, ii and Lemma 3 we bound r_t.

$$\begin{aligned}
\sum_{t=1}^{T} r_t^2 &\leqslant \sum_{t=1}^{T} 4\beta_T\sigma^2 C_2 \log(1+\sigma^{-2}\sigma_{t-1}^2(x_t)) \\
&= 8\beta_T\sigma^2 C_2 \frac{1}{2}\sum_{t\geqslant 1} \log(1+\sigma^{-2}\sigma_{t-1}^2(x_t)) \\
&\leqslant \beta_T C_1 \gamma_T^{vol} \ .
\end{aligned}$$

Finally, by cannons i, ii and iii, we bound R_T. Result follows from Cauchy-Schwarz inequality.

$$\begin{aligned} R_T^2 = (\sum_{t=1}^{T} r_t)^2 &\leqslant T \sum_{t=1}^{T} r_t^2 \\ &\leqslant T\beta_T C_1 \gamma_T^{vol} \ . \end{aligned}$$

Thus, $R_T \leqslant \sqrt{T\beta_T C_1 \gamma_T^{vol}}$. □

Chapter 4

Illustrative Results

We perform *in silico* evaluation with UVa/Padova T1DM 2008 Simulator [43], [1]. We use all 10 *in-silico* adult patients included in the simulator in our experiments. We use GPy [47] to implement VCGP-UCB. We compare our model against CGP-UCB [36] with non-volatile arms and formula-based bolus calculators [14] with different amount of miscalibration. We demonstrate that our algorithm can successfully personalize well-accepted formula-based calculator results with safe exploration. We highlight robustness by introducing various miscalibrations to the standard formula-based calculator.

As our evaluation metrics, we use glycemic outcome measures that are widely accepted by the diabetes management community to evaluate glycemic control [48] and glycemic risk measures [49]. Glycemic outcome metrics are mean BG, percentage time in BG target range [70,180] mg/dl (i.e. euglycemia) (%inT), percentage time below target (i.e. hypoglycemia) (%<T) and percentage time above target (i.e. hyperglycemia) (%>T). Glycemic risk indices are low blood glycemic index (i.e. risk of hypoglycemia) (LBGI), high blood glycemic index (i.e. risk of hyperglycemia) (HBGI) and risk index (RI=LBGI+HBGI). LBGI is used to group subjects regarding to their long-term risk for hypoglycemia. The risk categories are minimal, low, moderate and high risk, with LBGI of below 1.1, 1.1-2.5, 2.5-5.0, and above 5.0, respectively [50].

The BG measurement scale (20-600 mg/dl) is asymmetric. Hypoglycemia range (below 70 mg/dl) is much narrower than the hyperglycemia range (above 180 mg/dl). BG values are mapped to risk space where the minimum value of 0 is achieved at BG value of 112.5 mg/dl while its maximum value of 100 is achieved at 20 mg/dl and 600 mg/dl. We get risk value of 7.7 at the BG values of 70 and 180 mg/dl. Given a set of postprandial BG measurements, HBGI and LBGI are defined as the average hyperglycemia and hypoglycemia risk scores, respectively, where RI denotes overall risk score and is equal to the sum of HBGI and LBGI. The lower the risk values of LBGI and HBGI gets, the less the risk of hypoglycemia and hyperglycemia becomes since the postprandial BG gets closer to 112.5 mg/dl [50].

4.1 Experimental Setup

The context variables for our algorithms are the amount of carbohydrate intake, fasting BG level, time between meal and insulin intake, and time between meal and postprandial BG measurement. This is because (i) for a given *in-silico* patient in the simulator, the resulting postprandial BG values with respect to time are a function of amount of carbohydrate intake, fasting BG level, and amount of insulin intake, (ii) we learn individualized models per patient to avoid unnecessary and misleading domain specific assumptions, so we only include variables related with intra-patient variability and discard all other variables characterizing inter-patient treatment response and finally (iii) due to limitations of standard GP regression, we want to keep the number of input dimensions as small as possible to have faster convergence to optimal glycaemic control policy and avoid the curse of dimensionality, as it is further discussed in Section 4.4.

We generate 30 different meal events with different fasting BG values. Carbohydrate intake and fasting BG values are uniformly sampled from ranges 20-80 g and 70-180 mg/dl, respectively. Time between meal and insulin intake is set to 0. Time between meal and postprandial BG measurement is set to 150 minutes.

Bolus insulin dose ranges from 0 to 50 units with 0.1 increments. Min-max normalization is used for all contexts and arms by mapping to unit interval. We use a composite ARD (automatic relevance determination) kernel function defined over joint context-arm set for GP-based algorithms. The kernel consists of additive combination of Matern 5/2 and linear covariance functions. Matern 5/2 length scales are set as 0.5 for carbohydrate intake, fasting BG level and bolus insulin, and 5 for others. Variances are set as 1 for all kernels. Noise variance is set to 1. β_t is set to 4.

The algorithms have the BG target of 112.5 mg/dl, which is regarded as the clinical center of the BG scale, i.e. the BG value associated with zero risk index [50]. In accordance with this, in order to evaluate performance of bolus recommendations given contexts, we define the following loss function based on resulting postprandial BG, similar to [51]

$$loss(\tilde{g}) = \begin{cases} (\tilde{g}-6)^2/5 & \tilde{g} < 6 \\ (\tilde{g}-6)^2/10 & \tilde{g} \geqslant 6 \end{cases} \tag{4.1}$$

where $\tilde{g} = g/18.75$ is such that $\tilde{g} = 6$ corresponds to $g = 112.5$ mg/dl postprandial BG. Note that the BG measurement scale is asymmetric since the hypoglycemia range (below 70 mg/dl) is numerically much narrower than the hyperglycemia range (above 180 mg/dl). Moreover, hypoglycemia is considered to be more risky than hyperglycemia. This justifies choosing a higher loss for hypoglycemia. In each round t, learning algorithms observe the context z_t, choose a bolus dose s_t, observe the postprandial BG measurement g_t, and receive reward $y_t = -loss(\tilde{g}_t)$.

During the experiments, arm volatility is introduced through safety constraints for the insulin dose. These constraints are defined by taking formula-based bolus calculators (see Eq. 1.1) as baseline models. Our first calculator uses ICR and CF values of each patient that comes with the simulator, which are not very well-tuned. We name this calculator as SimCalc. We also provide results for a manually tuned version of SimCalc named as Fine-tuned SimCalc. For this, we scale down ICR and CF by a fixed amount in order to prevent hypoglycemia or hyperglycemia as much as possible. It represents the realistically unattainable optimal formula-based calculator whose existence is solely justified for being a baseline. In addition, we test against two types of miscalibrated formula-based calculators that cause either a certain amount of hypoglycemia or hyperglycemia by scaling down and up ICR and CF values by the same fixed constant, respectively. We name these calculators as HypoCalc and HyperCalc. For each of the calculators SimCalc, HypoCalc and HyperCalc, we perform personalization through safe-exploration with margins of $\pm$ 60% of corresponding formula-based calculator recommendation by using VCGP-UCB. We report the corresponding results under the names VCGP-UCB SimCalc, VCGP-UCB HypoCalc and VCGP-UCB HyperCalc.

We train separate VCGP-UCB models for each patient. GPs are initialized with 2 samples per patient with carbohydrate intake, fasting BG levels 30 g, 70 g and 100 mg/dl, 150 mg/dl respectively. Bolus doses for these events are given using SimCalc.

4.2 In-Silico Evaluation

For each method of treatment, we report postprandial BG distribution of all patients in Fig. 4.1, and the average glycemic metrics of all patients in Table 4.1. The results indicate that VCGP-UCB sufficiently compensates for miscalibrations in calculators as it shifts the mean BG on average 24% towards the target value (i.e., 112.5 mg/dl) even when the baseline calculators function inefficiently. We also observe an overall 30% increase in the frequency of euglycemic events, consequently a notable drop in hypoglycemic and hyperglycemic events. VCGP-UCB fine-tunes glycemic control by exploring around the doses recommended by calculators. Also, it is safer than CGP-UCB and the baselines as it has lower LBGI and HBGI values. This results from restricting dose exploration around baseline calculators.

Regret of VCGP-UCB with respect to the best available treatment over time given in Section 3.1 is shown in Fig. 4.2. This regret is computed by averaging over all patients. VCGP-UCB achieves around 80% smaller regret compared to CGP-UCB since its exploration is restricted to set of admissible doses around the formula-based calculator.

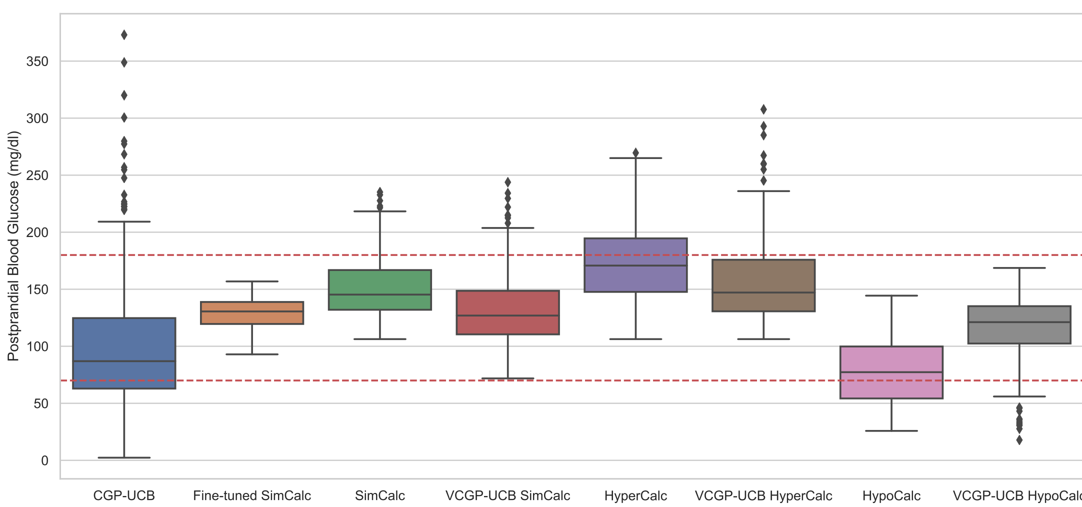


Figure 4.1: Box plot of postprandial BG distributions of different methods.

In addition, we define the regret with respect to BG target as

$$R_T^g = \sum_{t=1}^{T} |g_{target} - \mathbb{E}[g_t | s_t, z_t]|,$$

where $g_{target} = 112.5$ mg/dl. Fig. 4.3 shows how R_T^g averaged over all patients evolves over time. R_T^g increases faster for VCGP-UCB with SimCalc and HyperCalc because the safe exploration margins are not wide enough to include optimal insulin dose for some patients and contexts. Note that these calculators are more skewed towards the hyperglycemia region.

Table 4.1: Glycemic results averaged over glycemic outcome of all ten in-silico adult patients. For postprandial BG, mean and ± one standard deviation are reported.

Method	BG (mg/dl)	%inT	%<T	%>T	LBGI	HBGI	RI
CGP-UCB	99.63±57.72	0.55	0.33	0.12	34.39	8.54	42.93
Fine-tuned SimCalc	128.85±10.55	1.00	0.00	0.00	0.08	1.02	1.10
SimCalc	152.05±18.00	0.83	0.00	0.17	0.01	3.90	3.90
VCGP-UCB SimCalc	131.00±23.95	0.92	0.00	0.08	0.73	3.28	4.01
HyperCalc	174.60±30.33	0.60	0.00	0.40	0.01	7.62	7.62
VCGP-UCB HyperCalc	156.46±23.86	0.78	0.00	0.22	0.01	4.75	4.76
HypoCalc	78.13±18.06	0.58	0.42	0.00	13.00	0.15	13.15
VCGP-UCB HypoCalc	115.92±24.24	0.92	0.08	0.00	11.66	1.01	12.68
VCGP-UCB SimCalc (sparse GP with 10 inducing points)	132.92±26.09	0.91	0.00	0.09	0.70	3.67	4.37
VCGP-UCB SimCalc (sparse GP with 20 inducing points)	131.36±24.27	0.92	0.00	0.08	0.73	3.34	4.08
VCGP-UCB SimCalc (sparse GP with 30 inducing points)	131.00±23.95	0.92	0.00	0.08	0.73	3.28	4.01
VCGP-UCB SimCalc (± 40% exploration margin)	138.08±18.71	0.89	0.00	0.11	0.21	2.97	3.19
VCGP-UCB SimCalc (± 80% exploration margin)	123.53±27.21	0.89	0.06	0.06	1.56	3.71	5.26
VCGP-UCB SimCalc ($\tilde{\epsilon}_5$*)	130.82±24.57	0.91	0.01	0.09	0.79	3.22	4.01
VCGP-UCB SimCalc ($\tilde{\epsilon}_{10}$*)	131.05±25.37	0.91	0.02	0.08	1.01	3.23	4.24
VCGP-UCB SimCalc ($\tilde{\epsilon}_{20}$*)	131.84±32.27	0.84	0.04	0.12	2.01	3.62	5.63

*$\tilde{\epsilon}_s$ denotes zero mean Gaussian noise with s^2 variance added on top of postprandial BG value returned by the simulator.

In Table 4.1, we also investigate the affect of BG measurement noise on performance of VCGP-UCB when used with SimCalc. Results show that BG noise up to 20 mg/dl standard deviation is well tolerated by VCGP-UCB. In addition, we also study how the change of safety margin affects the performance of VCGP-UCB when used with SimCalc. On average, expanding the exploration margin enhances proximity to the target BG and vice versa. Although it depends on the baseline around which exploration is performed, a larger margin may cause a slight increase in risk indices and frequency of hypoglycemic and hyperglycemic events.

Finally, we would like to comment on computational complexity of VCGP-UCB. Calculation of GP posterior in each round from n past observations has $\mathcal{O}(n^3)$

time complexity and $\mathcal{O}(n^2)$ memory complexity due to the matrix inversion and determinant operations of $n \times n$ kernel matrix $\boldsymbol{K}_t$. This results in a rapid increase in computational complexity when more data becomes available over the course of the treatment. One solution is to use sparse GP approximation. It reduces the time complexity to $\mathcal{O}(mn^2)$, where m is the number of inducing points. Since it is advisable to pick inducing points smaller than the dimension of the kernel matrix $\boldsymbol{K}_t$ (i.e., smaller than the number of data points) and we have a maximum of 30 data points in our experimental setup, we report results for inducing points of 10, 20, and 30 for sparse GP regression in Table 4.1. We observe no significant performance difference as a result of using sparse GP when calculating UCB indices.

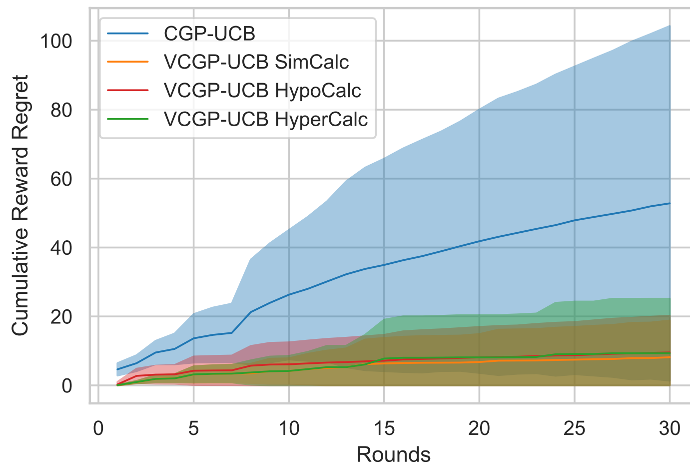


Figure 4.2: Cumulative reward regrets of VCGP-UCB algorithms. Error bars represent ± one standard deviation.

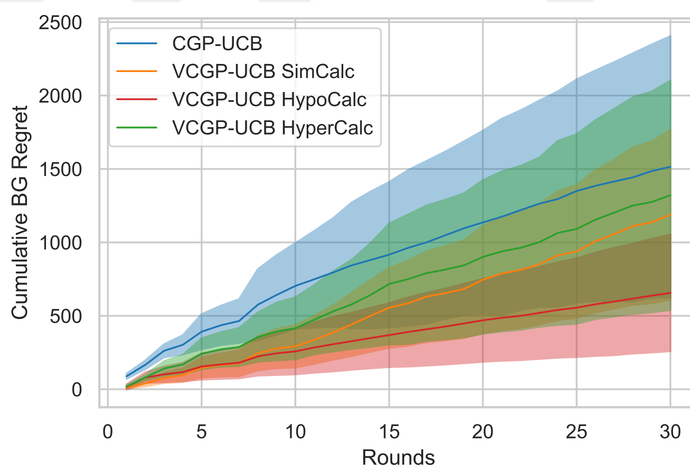


Figure 4.3: Cumulative BG regrets of VCGP-UCB algorithms. Error bars represent ± one standard deviation.

4.3 Comparison Against Subjective Assessment of an Individual Clinician

We also compare our algorithm against a clinician's subjective viewpoint on insulin recommendation. We show that VCGP-UCB better processes a patient's historical responses to treatment and perform much more personalized recommendations than a clinician can possibly do. We expect this phenomenon to especially be true when the size of the longitudinal data increases to a certain degree that the clinicians have a hard time analyzing it and obtaining any valuable insights.

This experiment uses five *in-silico* adult patients with identifiers 2, 5, 7, 8, and 10. We separately generate 20 training and 20 test contexts for each of the five patients. Training contexts represent patient's historical responses to the treatment. We ask a clinician to go through the training data and make insulin recommendations for each test context. Later, we train VCGP-UCB on training data and get recommendations for test data with and without any posterior update to the model. A posterior update means that as we iterate over the test set, we sequentially add the current sample on top of the training set immediately after we get the treatment recommendation and compute the corresponding postprandial BG level. In contrast, absence of posterior update means that we use the same training set in the recommendation for all test samples. We add independent and identically distributed zero-mean Gaussian noise with the variance of 25 on top of the postprandial BG response since training data has the same noise on the postprandial BG values. We use the same kernel parameters, noise variance, and BG target given above. We set the context variable time between meal and insulin intake to zero. Since effects of short-acting insulin on blood glucose peak around 2-3 hours range after insulin intake, we report postprandial BG responses for 120, 150, and 180 minutes. We both share the data and explain its collection process in Appendix B to enable different lines of comparisons for future methods in diabetes research.

Table 4.2: Glycemic results averaged over glycemic outcome of all five in-silico adult patients. For postprandial BG, mean and ± one standard deviation are reported.

	Method	BG (mg/dl)	%inT	%<T	%>T	LBGI	HBGI	RI
t_{meas} = 120 min	Clinician	197.44±40.73	0.37	0.00	0.63	0.00	11.99	11.99
	SimCalc	182.67±33.19	0.60	0.00	0.40	0.00	8.98	8.98
	CGP-UCB (without posterior update)	80.74±34.96	0.48	0.47	0.05	29.31	3.42	32.73
	CGP-UCB (with posterior update)	107.74±64.50	0.63	0.26	0.11	12.86	9.91	22.77
	VCGP-UCB SimCalc (without posterior update)	172.02±32.05	0.63	0.00	0.37	0.07	7.66	7.72
	VCGP-UCB SimCalc (with posterior update)	157.85±27.33	0.80	0.00	0.20	0.03	5.16	5.18
t_{meas} = 150 min	Clinician	177.43±32.19	0.60	0.00	0.40	0.00	8.16	8.16
	SimCalc	157.31±22.16	0.82	0.00	0.18	0.00	4.53	4.53
	CGP-UCB (without posterior update)	50.21±26.58	0.24	0.74	0.02	86.38	2.24	88.62
	CGP-UCB (with posterior update)	99.69±60.19	0.53	0.34	0.13	30.31	11.98	42.29
	VCGP-UCB SimCalc (without posterior update)	147.77±34.02	0.76	0.00	0.24	0.54	5.81	6.34
	VCGP-UCB SimCalc (with posterior update)	129.40±23.16	0.94	0.00	0.06	0.64	2.23	2.87
t_{meas} = 180 min	Clinician	151.98±23.51	0.82	0.00	0.18	0.20	4.34	4.54
	SimCalc	132.56±11.84	1.00	0.00	0.00	0.07	1.35	1.42
	CGP-UCB (without posterior update)	28.85±18.00	0.05	0.95	0.00	141.93	0.53	142.47
	CGP-UCB (with posterior update)	94.24±77.87	0.38	0.46	0.16	38.31	22.31	60.62
	VCGP-UCB SimCalc (without posterior update)	127.37±35.97	0.72	0.12	0.16	2.87	5.26	8.14
	VCGP-UCB SimCalc (with posterior update)	117.75±32.57	0.86	0.05	0.09	2.90	3.73	6.63

*t_{meas} denotes time between meal and postprandial BG measurement.

We report our results in Table 4.2. VCGP-UCB overall considerably outperforms the clinician and CGP-UCB in proximity to the target BG and the average frequency of euglycemic events, especially with the posterior update. We observe that CGP-UCB performs the worst of all, which is primarily due to lack of safety constraints and the bias in the training data. Since the training data is created with SimCalc treatment recommendations, it contains sampling bias. When we train CGP-UCB on the data, the model gets considerable uncertainty for the regions outside the data. Consequently, its desire for exploration outweighs exploitation due to the acquisition function Eq. 3.1. Since there is no safety constraint, it always recommends maximum possible insulin dose of 50 U, which results in a large number of hypoglycemia events. On the other hand, VCGP-UCB always recommends on border of the exploration margin as the training data suggest that it should recommended higher doses, but no dose higher than the highest safe dose. Thanks to safety constraints, VCGP-UCB's recommendations do not deviate substantially from the set of safe doses, and even result in less hyperglycemia cases compared to the recommendations of the clinician.

4.4 Limitations and Future Research Directions

While experimental results demonstrate the effectiveness of the proposed approach in BG regulation, the current model and algorithm have certain limitations. Next, we discuss these and potential remedies.

Safety. Safety of VCGP-UCB depends on the *reliability* of baselines around which exploration is performed. Exploration helps improving performance when clinically accepted baselines are inaccurate or miscalibrated up to a certain degree. However, utmost care should be taken when choosing the baselines, as faulty baselines can result in unsafe recommendations.

Bias. Our algorithm makes treatment decisions solely based on the chosen asymmetrical loss function such that given a postprandial BG target, below target cases are punished twice as much as above target cases. This may introduce underestimation bias for hyperglycemia cases in treatment selection. Future research would be to completely eliminate the need for a loss function and make decisions directly from a GP surface where the response variable is postprandial BG instead of a custom loss function.

Curse of dimensionality. The vanilla GP regression is not able to cope with high input space dimensions. As the number of dimensions increases, the number of decision epochs (i.e., sample size) required to learn response surface grows exponentially. If the input dimensions are somewhat uncorrelated and not very large in size, this can be addressed with variable reduction methods such as sensitivity analysis or automatic relevance determination. Otherwise, we can initially construct a small set of uncorrelated futures with the help of linear or non-linear dimensionality reduction methods, where we may later apply variable reduction methods [52].

Model mismatch. In real-world, BG measurements are recorded by BG sensors. Device specific measurement noise can disturb the accuracy of the GP-based model. In particular, noise on the BG measurement will propagate through Eq. 4.1, resulting in non-Gaussian noise on rewards. This contamination will create model mismatch and can reduce the effectiveness of VCGP-UCB especially when BG measurement noise has high variance. Nevertheless, for most of the commercially available BG meters (fingerstick testing) measurement errors are small enough to meet International Organization for Standardization (ISO) criteria [53].

There are two possible solutions to mitigate model mismatch. One can use Warped GPs [54, 55] to find a nonlinear transformation of the reward data, which can be accurately modeled using GPs. Alternatively, one can assume that BG measurement error is Gaussian and use a GP to model BG surface. This GP can be used to construct confidence intervals of BG values given treatment and context. Optimistic reward of each admissible treatment can be found by minimizing $loss(\tilde{g})$ over the confidence interval of BG values. Then, the treatment that minimizes the optimistic loss can be recommended.

Other sources of errors that require further investigation include errors made by the patient in reporting the correct value of carbohydrate intake (context) and insulin dosage (arm).

Chapter 5

Pilot Study

This work is a part of a three-years-long joint interdisciplinary project carried out by Bilkent University, Ministry of Health of Turkey, and ESEN Inc., and funded by TÜBITAK. The goal was to design, deliver, and pilot test a platform to recommend bolus insulin doses for T1DM patients. To this end, we develop various learning algorithms, one of which is VCGP-UCB. The platform is a CRUD (i.e., Create, Read, Update, and Delete) application with a client-server architecture. It supports user management, data management and data analysis. It facilitates entering, storing, viewing, changing health-related electronic information using computer-based forms and reports insights and analytics.

This chapter touches on some technical details and addresses important privacy, ethical, and safety issues. In the end, we share conceptual user interface designs describing the core functionality of the platform. At the time of writing this thesis, the pilot study is still ongoing, so we could not share the questionnaire and its results where we measure the platform's effectiveness in the eyes of clinicians and patients. Except for the user interface design, the learning model design, and implementation, ESEN Inc developed and deployed the platform as a web and mobile application. Prof. Sema Ucak-Basat, MD, from Department of Internal Diseases, Health Sciences University, Umraniye Training, and Research Hospital, Istanbul, Turkey, was in charge of executing the pilot study.

5.1 Preliminaries

There are two types of users in the platform: patients and doctors. Each doctor can have many patients registered to them, and each patient can have only a single doctor assigned to them. Patients can enter and delete their data regarding to their physiological signals and meal- or insulin-intake. Later, these data points are presented to their doctors for approval. Samples cleared by their doctors are fed to the models. Patients can view their treatment indicators and statistics. They can get treatment recommendations and communicate (i.e., private text-based messaging) with their doctors through the platform to ask their questions. Doctors supervise and give treatment recommendations to their patients. They can check longitudinal data their patients entered and approve individual data points for use in the learning model. They can see treatment insights of their patients. They can also create a new model for patients.

We use a single model per patient; in other words, we do not share models across different patients to avoid unnecessary and misleading domain-specific assumptions. It also allows us to use fewer context variables related to intra-patient variability and discard all other variables characterizing inter-patient treatment response. Context variables explain treatment effects on the response variable; thus, they are best selected by domain experts. We pick the following variables after consulting with a few clinicians. Context variables are amount of carbohydrate intake (g), fasting BG level (mg/dl), time between meal and insulin intake (min), time between meal and postprandial BG measurement (min), and day time (min passed since the midnight scaled to range of (0,5)). The model has mainly two methods: *recommend* and *update*. Doctors get treatment recommendations via *recommend* method. It can be called as many times as needed. *update* method is used to enter new data into the model and update it. At the end of each day, data gathered for each patient is fed into the models, followed by an update of kernel parameters from the data via maximum likelihood estimation.

5.2 Privacy Policy

To honor our patient's data privacy rights, we follow a strict policy. (i) Doctors can only see data of patients that are assigned to them. (ii) Patients can delete their data any time they want. (iii) The platform is deployed to the Ministry of Health of Turkey servers, so all personal health data is stored in databases protected by the government infrastructure. (iv) ESEN Inc has followed software security standards in the making of the platform.

5.3 Ethical Policy

To avoid any ethical misconduct, we take the following steps. Doctors and patients are made aware of the potential risks by signing a clarification text regarding platform use. Patients can only receive recommendations from their doctors. In other words, they cannot directly interact with the learning model to avoid any misuse or misunderstanding.

5.4 Safeguards

We are in favor of the following safeguards to mitigate risks further. All these safeguards can be enabled/disabled when needed.

Access to Learning Model. Only doctors have access to the learning model insights and recommendations. Patients get their treatment details from their doctors through the platform. This means doctors make the final judgment call when it comes to treatment details. In the case of a malfunction in the model, patients can still get a reliable treatment.

Data Entry. The model has a predefined set of context variables and their corresponding domains. One use of this is to sanity check if the values entered by patients are within the predefined range for all variables. We do not process samples outsides their domains. Also, we only accept fasting BG measurements up to 90 min prior to meal intake.

Doctor Supervision. Data quality in the statistical model has a high impact on the quality of the recommendations, especially for non-parametric models like Gaussian Processes. Thus, although patients manage their data in the platform, doctors have complete control over the data fed to the learning model. The doctor assigned to a particular patient has to approve data points one at a time before they are allowed to be processed in the model. They can also later remove approved data points. In a way, we prevent errors in the data entry.

Outlier Detection. The Gaussian process can predict postprandial BG mean and variance as explained in Section 3.1. New data comes to update our model in the form of (context, treatment, postprandial BG). Before updating the model, we first compute postprandial BG mean and variance for given context and treatment. Then, we check whether the sample is within a 99% confidence interval. Thus, we naturally eliminate outliers.

Calibration Data. The Gaussian process is a non-parametric learning model. The data heavily influence its posterior mean and variance predictions. When the data size is small and the data has low variance, its posterior variance predictions can be smaller than they should be. This causes curves in 3.1 be irregular. We address this problem by adding a few samples acting like calibration in the extreme points (i.e., zero carbohydrate & max insulin, max carbohydrate & zero insulin).

Hard Constraints. Doctors can enter multiple inequality constraints with variables carbohydrate and/or insulin amounts. In this way, doctors can eliminate some dangerous predictions such as low carbohydrate high insulin, or vice versa. The model only returns a treatment if it satisfies the constraints.

Logging. We log all the actions within the model for debugging and forensic purposes.

5.5 Screen Blueprints

We give seven core user interfaces to interact with our model. We briefly explain their purpose and how users come to interact with them. Interfaces are primarily conceptual drawings whose aim is to explain how the platform is supposed to function. ESEN Inc finalized user interfaces based on the following designs we provided.

Visualize Context Variables

This is a doctors-only page. The screen in Fig. 5.1 shows the list of context variables, their units, and value ranges in the statistical learning model. After having a consensus with a domain expert, they are manually set in the source code. Neither doctors nor patients have permission to update them.

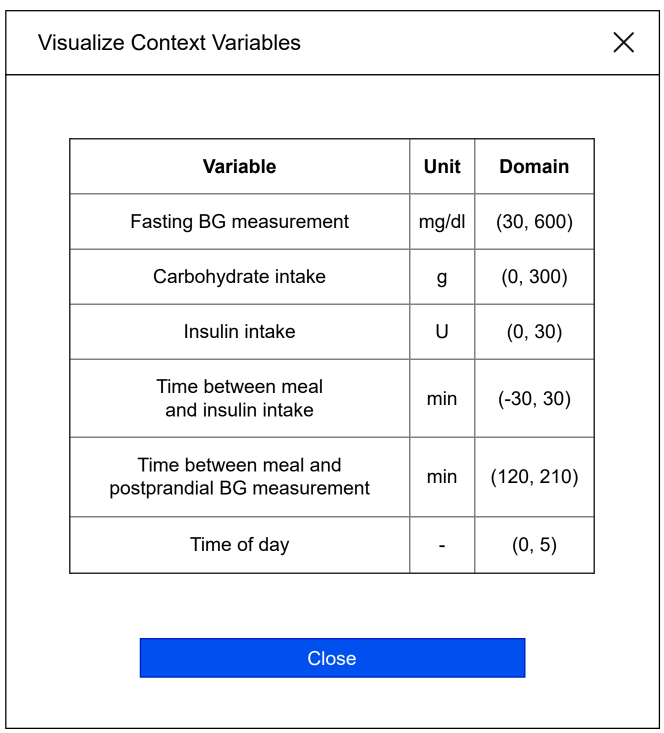


Figure 5.1: Visualization context variables, their units and value ranges.

Create a Patient-Specific Model

This is a doctors-only page. The screen in Fig. 5.2 only takes inputs required from the doctor of the patient to initialize the model. Context variables for the model are provided in Section 5.1. Doctors can update ICR and CF values; delete or create models anytime they want.

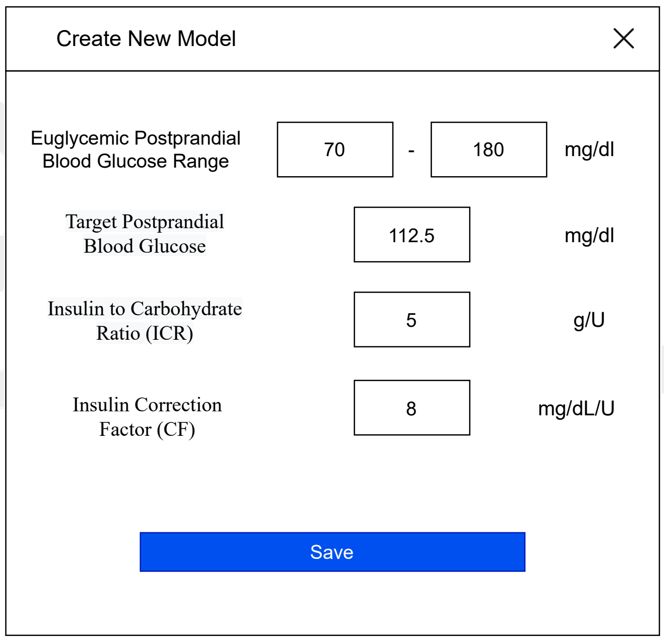


Figure 5.2: Creation of a patient-specific model.

Data Entry

This is a patients-only page. The screen in Fig. 5.2 demonstrates a sample data entry. We check whether context values are within their correct ranges and fasting BG measurements time is at a maximum of 90 min before meal intake. The user experience of the patient data entry screen is open to further improvements. Some of the glucose sensors can be integrated into the platform via smartphone to stream relevant data automatically. Alternatively, patients can also enter values separately for each variable at any given time.

Figure 5.3: Data entry to the platform.

We use the nutrition guide published by the Ministry of Health of Turkey to estimate carbohydrate intake amounts from patients' meals [56]. Likewise, we utilize the list of glucose meters and diabetes medications sold in the Turkish healthcare market. Patients register the glucose meter and insulin used in data entry. Doctors select the medicines of their choice to give a medical prescription. This allows us to keep a healthy track of what devices and medications are used to treat patients.

We obtained the lists from TİTCK-Regulatory and Supervisory Authority [57]. TİTCK is a Turkish government authority that is responsible for licensing medicines and medical devices sold in the healthcare market in Turkey. The list of glucose meters is collected from the database called *The Turkish Medicines and Medical Devices National Data Bank* [58]. The list of diabetes medications is acquired from a periodic publication titled *E-Prescription Medicine List* [59].

Enter Activity

This is a patients-only page. The screen in Fig. 5.4 demonstrates activity entry to the platform. We disqualify all data on the date of the activity. *Exercise* intensity has three values: light (e.g., 30 min walking), moderate (e.g., swimming, tennis), and high (e.g., football, running). *Acute diseases* have values of flu, cold, illness, diarrheal illness, others. *Pain or ache* have the values of head, waist, chest, back, abdomen, others.

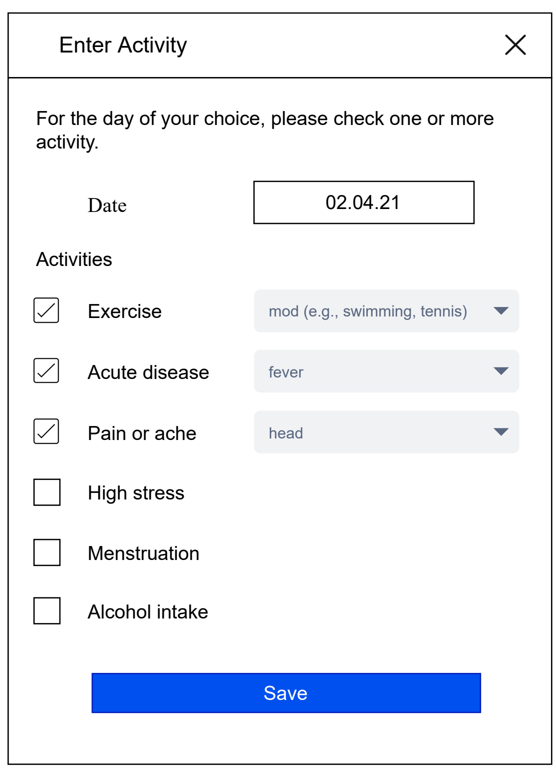


Figure 5.4: Entering an activity to the platform.

Data Supervision

This is a doctors-only page. Although patients can enter any data they want, doctors have full control over what goes into the model. This includes supervising all data points fed into the model and managing all data points already in the model. It acts both as a safeguard and a sanity check mechanism for the data. Doctors review the data and reject all out-of-ordinary samples if they deem necessary. They can also choose not to take action. In such a case, the data can be accessed on the same page for later reviews. Approved data will be fed to the learning model of that particular patient. Rejected data can still be accessed from the patient data pool but never used in the learning model. Doctors can also list all data points used in the model to modify or remove data points.

The screen in Fig. 5.5 demonstrates data verification process. The doctor can approve the first sample, but they prefer not to take any action to keep the sample for a later review. The last sample is rejected due to high-intensity exercise and the outlier warning. Outlier detection is explained in Section 5.4. Doctors can also list all data points used in the model to modify or remove data points.

Approve Data

Meal Intake Date / Time	Carb (g)	Fasting Blood Glucose (mg/dl)	Insulin (U)	Postprandial BG Measurement Time (min)	Postprandial Blood Glucose (mg/dl)	Outlier Warning	Activity	Approve	Reject
30.04.21 / 13:15	50	130	7	150	135			☐	☐
29.04.21 / 18:30	64	120	5	120	140		light exercise (walking)	☑	☐
28.04.21 / 07:30	15	10	10	100	160	!	(1) high intensity exercise (football), (2) acute disease (fever)	☐	☑

Figure 5.5: Data supervision by a doctor.

Recommend Bolus Insulin

This is a doctors-only page unless doctors deliberately permit their patients to access this page. If they do, the warning at the top in Fig. 5.6, i.e., "The patient cannot access this page!" will turn into "The patient can access this page!". The screen demonstrates the process of getting a recommendation from the platform. It also points out risks involving this recommendation. In case of a suspicious recommendation, doctors can report this by clicking on the "Report Anomaly" button. This will create a technical support ticket with the recommendation details and a snapshot of the model.

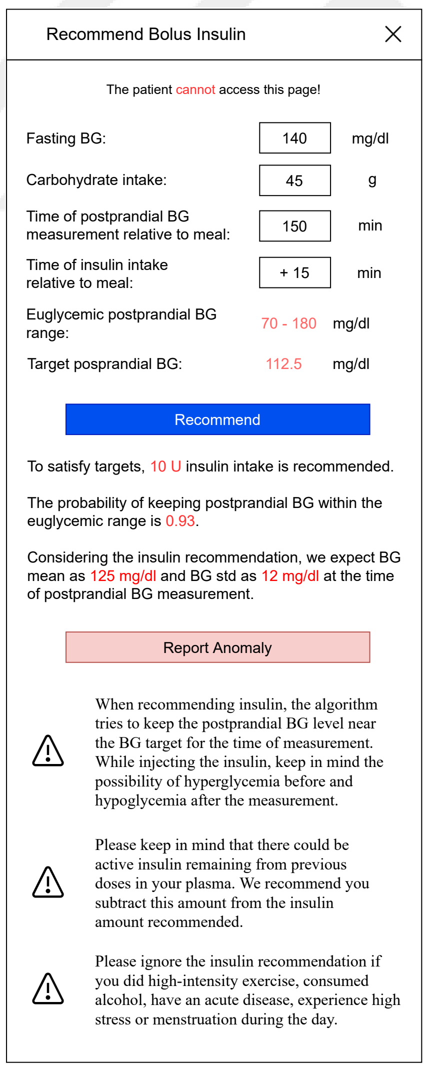


Figure 5.6: Bolus insulin recommendation.

Visualize Treatment Analytic

This page is available to both patients and doctors. Fig. 5.7 reports treatment statistics and performance indicators. Risk indices are further discussed in Section 4.

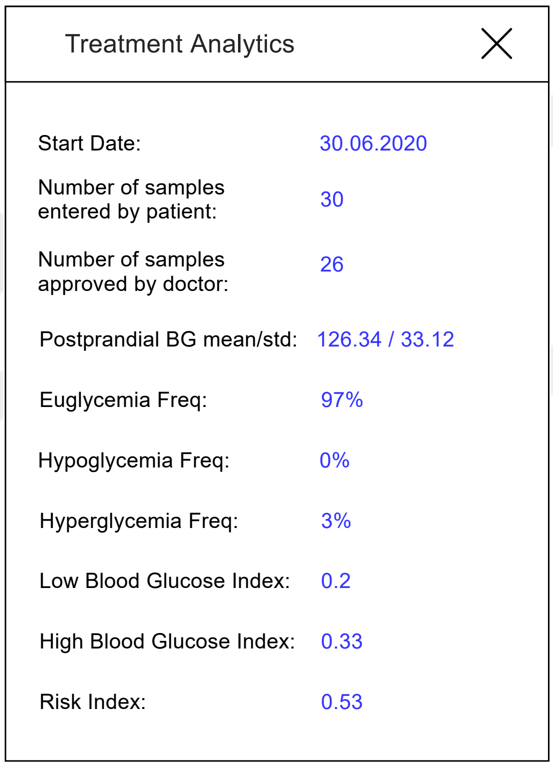


Figure 5.7: Visualization treatment analytic.

Visualize Treatment Insights

This is a doctors-only functionality. Doctors have access to all insights, including figures and probabilities explained in Section 3.3. In case of a suspicious recommendation, doctors can report this by clicking on the "Report Anomaly" button. This will create a technical support ticket with the recommendation details and a snapshot of the model.

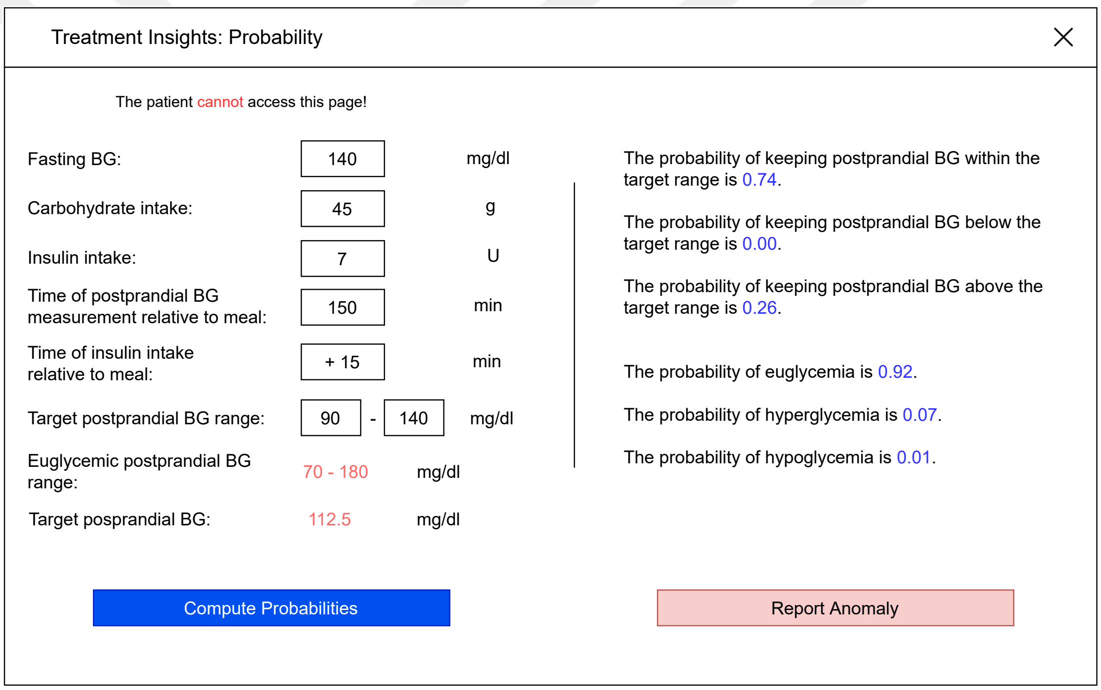


Figure 5.8: Visualization of the treatment insights. It shows the computation of probabilities for any given range of postprandial BG levels.

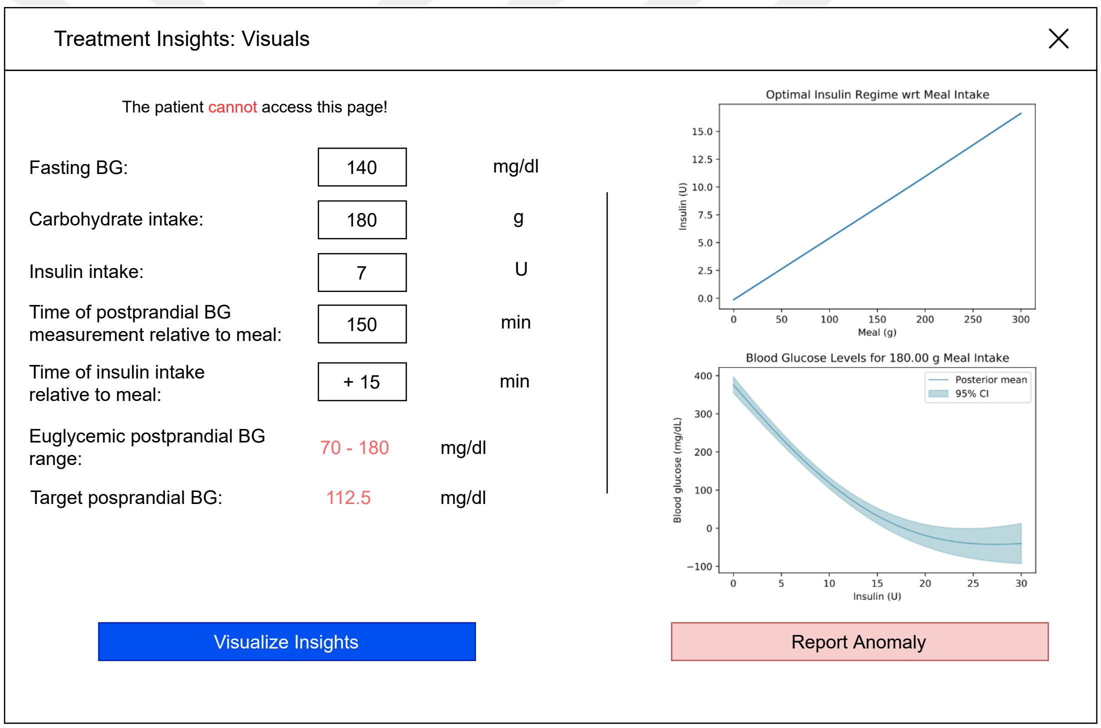


Figure 5.9: Visualization of relational insights. It shows the optimal bolus insulin regimen and insulin-blood glucose relationship in case of a meal intake.

Chapter 6

Conclusion

We adapted Contextual Gaussian Process Upper Confidence Bound algorithm to the volatile bandit setup, and proposed VCGP-UCB. We showed how VCGP-UCB can be used to optimize treatment regimes under time-varying constraints. VCGP-UCB achieves $\tilde{O}(\sqrt{T\gamma_T^{vol}})$ regret, and enables safe exploration around a formula-based treatment strategy. This demonstrates the applicability of bandit algorithms in fine-tuning treatment decisions around interpretable baseline treatment strategies employed in clinical practice. We used our algorithm as a closed-loop system for BG regulation in type 1 diabetes mellitus patients. Simulation results show that our algorithm has the potential to improve BG regulation compared to formula-based methods.

Bibliography

[1] J. Xie, "Simglucose v0.2.1." https://github.com/jxx123/simglucose, 2018.

[2] C. Dalla Man, M. D. Breton, and C. Cobelli, "Physical activity into the meal glucose—insulin model of type 1 diabetes: In silico studies," 2009.

[3] A. A. Celik and C. Tekin, "Optimizing dynamic treatment regimes via volatile contextual gaussian process bandits," in *Reinforcement Learning for Real Life (RL4RealLife) Workshop in the 38th International Conference on Machine Learning*, 2021.

[4] N. Cho, J. Shaw, S. Karuranga, Y. d. Huang, J. da Rocha Fernandes, A. Ohlrogge, and B. Malanda, "Idf diabetes atlas: Global estimates of diabetes prevalence for 2017 and projections for 2045," *Diabetes Research and Clinical Practice*, vol. 138, pp. 271–281, 2018.

[5] I. Satman, B. Omer, Y. Tutuncu, S. Kalaca, S. Gedik, N. Dinccag, K. Karsidag, S. Genc, A. Telci, B. Canbaz, *et al.*, "Twelve-year trends in the prevalence and risk factors of diabetes and prediabetes in turkish adults," *European journal of epidemiology*, vol. 28, no. 2, pp. 169–180, 2013.

[6] I. Satman, T. Yilmaz, A. Sengül, S. Salman, F. Salman, S. Uygur, I. Bastar, Y. Tütüncü, M. Sargin, N. Dinççag, *et al.*, "Population-based study of diabetes and risk characteristics in turkey: results of the turkish diabetes epidemiology study (turdep)," *Diabetes care*, vol. 25, no. 9, pp. 1551–1556, 2002.

[7] I. D. Federation, "Idf diabetes atlas," *Brussels: International Diabetes Federation*, vol. 128, pp. 40–50, 2013.

[8] D. C. Klonof, "The current status of bolus calculator decision-support software," *Journal of Diabetes Science and Technology*, vol. 6, no. 5, pp. 990–994, 2012.

[9] G. A. Fleming, J. R. Petrie, R. M. Bergenstal, R. W. Holl, A. L. Peters, and L. Heinemann, "Diabetes digital app technology: benefits, challenges, and recommendations. a consensus report by the european association for the study of diabetes (easd) and the american diabetes association (ada) diabetes technology working group," *Diabetes Care*, vol. 43, no. 1, pp. 250–260, 2020.

[10] D. Mellitus, "Diagnosis and classification of diabetes mellitus," *Diabetes Care*, vol. 28, no. S37, pp. S5–S10, 2005.

[11] S. Schmidt and K. Nørgaard, "Bolus calculators," *Journal of Diabetes Science and Technology*, vol. 8, no. 5, pp. 1035–1041, 2014.

[12] C. Cobelli, E. Renard, and B. Kovatchev, "Artificial pancreas: past, present, future," *Diabetes*, vol. 60, no. 11, pp. 2672–2682, 2011.

[13] M. Tejedor, A. Z. Woldaregay, and F. Godtliebsen, "Reinforcement learning application in diabetes blood glucose control: A systematic review," *Artificial Intelligence in Medicine*, p. 101836, 2020.

[14] J. Walsh, R. Roberts, and T. Bailey, "Guidelines for optimal bolus calculator settings in adults," *Journal of Diabetes Science and Technology*, vol. 5, no. 1, pp. 129–135, 2011.

[15] A. Aamodt and E. Plaza, "Case-based reasoning: Foundational issues, methodological variations, and system approaches," *AI communications*, vol. 7, no. 1, pp. 39–59, 1994.

[16] F. Torrent-Fontbona and B. López, "Personalized adaptive cbr bolus recommender system for type 1 diabetes," *IEEE Journal of Biomedical and Health Informatics*, vol. 23, no. 1, pp. 387–394, 2018.

[17] P. Herrero, P. Pesl, M. Reddy, N. Oliver, P. Georgiou, and C. Toumazou, "Advanced insulin bolus advisor based on run-to-run control and case-based

reasoning," *IEEE Journal of Biomedical and Health Informatics*, vol. 19, no. 3, pp. 1087–1096, 2014.

[18] P. Herrero, J. Bondia, O. Adewuyi, P. Pesl, M. El-Sharkawy, M. Reddy, C. Toumazou, N. Oliver, and P. Georgiou, "Enhancing automatic closed-loop glucose control in type 1 diabetes with an adaptive meal bolus calculator–in silico evaluation under intra-day variability," *Computer Methods and Programs in Biomedicine*, vol. 146, pp. 125–131, 2017.

[19] P. Pesl, P. Herrero, M. Reddy, M. Xenou, N. Oliver, D. Johnston, C. Toumazou, and P. Georgiou, "An advanced bolus calculator for type 1 diabetes: system architecture and usability results," *IEEE Journal of Biomedical and Health Informatics*, vol. 20, no. 1, pp. 11–17, 2015.

[20] P. Pesl, P. Herrero, M. Reddy, N. Oliver, D. G. Johnston, C. Toumazou, and P. Georgiou, "Case-based reasoning for insulin bolus advice: evaluation of case parameters in a six-week pilot study," *Journal of Diabetes Science and Technology*, vol. 11, no. 1, pp. 37–42, 2017.

[21] D. Duvenaud, *Automatic model construction with Gaussian processes*. PhD thesis, University of Cambridge, 2014.

[22] J. L. Sherr, E. Cengiz, C. C. Palerm, B. Clark, N. Kurtz, A. Roy, L. Carria, M. Cantwell, W. V. Tamborlane, and S. A. Weinzimer, "Reduced hypoglycemia and increased time in target using closed-loop insulin delivery during nights with or without antecedent afternoon exercise in type 1 diabetes," *Diabetes Care*, vol. 36, no. 10, pp. 2909–2914, 2013.

[23] B. P. Kovatchev, E. Renard, C. Cobelli, H. C. Zisser, P. Keith-Hynes, S. M. Anderson, S. A. Brown, D. R. Chernavvsky, M. D. Breton, A. Farret, *et al.*, "Feasibility of outpatient fully integrated closed-loop control: first studies of wearable artificial pancreas," *Diabetes Care*, vol. 36, no. 7, pp. 1851–1858, 2013.

[24] D. J. Luckett, E. B. Laber, A. R. Kahkoska, D. M. Maahs, E. Mayer-Davis, and M. R. Kosorok, "Estimating dynamic treatment regimes in mobile health

using V-learning," *Journal of the American Statistical Association*, vol. 115, no. 530, pp. 692–706, 2020.

[25] I. Fox and J. Wiens, "Reinforcement learning for blood glucose control: Challenges and opportunities," 2019.

[26] T. Zhu, K. Li, P. Herrero, and P. Georgiou, "Deep learning for diabetes: a systematic review," *IEEE Journal of Biomedical and Health Informatics*, 2020.

[27] C. C. Bennett and K. Hauser, "Artificial intelligence framework for simulating clinical decision-making: A Markov decision process approach," *Artificial Intelligence in Medicine*, vol. 57, no. 1, pp. 9–19, 2013.

[28] H. Robbins, "Some aspects of the sequential design of experiments," *Bulletin of the American Mathematical Society*, vol. 58, no. 5, pp. 527–535, 1952.

[29] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine Learning*, vol. 47, no. 2, pp. 235–256, 2002.

[30] T. Lu, D. Pál, and M. Pál, "Contextual multi-armed bandits," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 485–492, JMLR Workshop and Conference Proceedings, 2010.

[31] A. Slivkins, "Contextual bandits with similarity information," in *Proceedings of the 24th annual Conference On Learning Theory*, pp. 679–702, JMLR Workshop and Conference Proceedings, 2011.

[32] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proceedings of the 19th international conference on World Wide Web*, pp. 661–670, 2010.

[33] J. Langford and T. Zhang, "The epoch-greedy algorithm for contextual multi-armed bandits," *Advances in Neural Information Processing Systems*, vol. 20, no. 1, pp. 96–1, 2007.

[34] R. Kleinberg, A. Niculescu-Mizil, and Y. Sharma, "Regret bounds for sleeping experts and bandits," *Machine learning*, vol. 80, no. 2, pp. 245–272, 2010.

[35] N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger, "Information-theoretic regret bounds for Gaussian process optimization in the bandit setting," *IEEE Transactions on Information Theory*, vol. 58, no. 5, pp. 3250–3265, 2012.

[36] A. Krause and C. S. Ong, "Contextual Gaussian process bandit optimization," in *Advances in Neural Information Processing Systems*, pp. 2447–2455, 2011.

[37] Y. Sui, A. Gotovos, J. Burdick, and A. Krause, "Safe exploration for optimization with Gaussian processes," in *International Conference on Machine Learning*, pp. 997–1005, 2015.

[38] Y. Sui, J. Burdick, Y. Yue, *et al.*, "Stagewise safe bayesian optimization with Gaussian processes," in *International Conference on Machine Learning*, pp. 4781–4789, 2018.

[39] S. Amani, M. Alizadeh, and C. Thrampoulidis, "Linear stochastic bandits under safety constraints," in *Advances in Neural Information Processing Systems*, pp. 9256–9266, 2019.

[40] S. Amani, M. Alizadeh, and C. Thrampoulidis, "Regret bound for safe Gaussian process bandit optimization," in *Learning for Dynamics and Control*, pp. 158–159, PMLR, 2020.

[41] M. Reddy, P. Herrero, M. El Sharkawy, P. Pesl, N. Jugnee, H. Thomson, D. Pavitt, C. Toumazou, D. Johnston, P. Georgiou, *et al.*, "Feasibility study of a bio-inspired artificial pancreas in adults with type 1 diabetes," *Diabetes Technology & Therapeutics*, vol. 16, no. 9, pp. 550–557, 2014.

[42] E. Daskalaki, P. Diem, and S. G. Mougiakakou, "An actor–critic based controller for glucose regulation in type 1 diabetes," *Computer Methods and Programs in Biomedicine*, vol. 109, no. 2, pp. 116–125, 2013.

[43] B. P. Kovatchev, M. Breton, C. Dalla Man, and C. Cobelli, "In silico preclinical trials: a proof of concept in closed-loop control of type 1 diabetes," 2009.

[44] C. D. Man, F. Micheletto, D. Lv, M. Breton, B. Kovatchev, and C. Cobelli, "The uva/padova type 1 diabetes simulator: new features," *Journal of Diabetes Science and Technology*, vol. 8, no. 1, pp. 26–34, 2014.

[45] R. Visentin, E. Campos-Náñez, M. Schiavon, D. Lv, M. Vettoretti, M. Breton, B. P. Kovatchev, C. Dalla Man, and C. Cobelli, "The uva/padova type 1 diabetes simulator goes from single meal to single day," *Journal of Diabetes Science and Technology*, vol. 12, no. 2, pp. 273–281, 2018.

[46] C. E. Rasmussen, "Gaussian processes in machine learning," in *Summer School on Machine Learning*, pp. 63–71, Springer, 2003.

[47] GPy, "GPy: A Gaussian process framework in python." `http://github.com/SheffieldML/GPy`, 2012.

[48] D. M. Maahs, B. A. Buckingham, J. R. Castle, A. Cinar, E. R. Damiano, E. Dassau, J. H. DeVries, F. J. Doyle, S. C. Griffen, A. Haidar, *et al.*, "Outcome measures for artificial pancreas clinical trials: a consensus report," *Diabetes Care*, vol. 39, no. 7, pp. 1175–1179, 2016.

[49] B. P. Kovatchev, M. Straume, D. J. Cox, and L. S. Farhy, "Risk analysis of blood glucose data:az quantitative approach to optimizing the control of insulin dependent diabetes," *Journal of Theoretical Medicine*, vol. 3, no. 1, pp. 1–10, 2000.

[50] B. P. Kovatchev, D. J. Cox, A. Kumar, L. Gonder-Frederick, and W. L. Clarke, "Algorithmic evaluation of metabolic control and risk of severe hypoglycemia in type 1 and type 2 diabetes using self-monitoring blood glucose data," *Diabetes Technology & Therapeutics*, vol. 5, no. 5, pp. 817–828, 2003.

[51] R. Keramati, C. Dann, A. Tamkin, and E. Brunskill, "Being optimistic to be conservative: Quickly learning a CVaR policy," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 4436–4443, 2020.

[52] R. Tripathy, I. Bilionis, and M. Gonzalez, "Gaussian processes with built-in dimensionality reduction: Applications to high-dimensional uncertainty propagation," *Journal of Computational Physics*, vol. 321, pp. 191–223, 2016.

[53] R. M. Bergenstal, "Evaluating the accuracy of modern glucose meters," *Insulin*, vol. 3, no. 1, pp. 5–14, 2008.

[54] E. Snelson, C. E. Rasmussen, and Z. Ghahramani, "Warped Gaussian processes," *Advances in Neural Information Processing Systems*, vol. 16, pp. 337–344, 2004.

[55] M. Lázaro-Gredilla, "Bayesian warped Gaussian processes," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1619–1627, 2012.

[56] N. Yardım, N. Çelikay, and E. Çelikcan, "Beslenme rehberi," *TC Sağlık Bakanlığı Halk Sağlığı Genel Müdürlüğü, Ankara*, 2019.

[57] "Türkiye cumhuriyeti sağlık bakanlığı türkiye ilaç ve tıbbi cihaz kurumu." `https://www.titck.gov.tr`. (Accessed on 01/04/2021).

[58] TİTCK, "Türkiye ilaç ve tıbbi cihaz ulusal bilgi bankası." `https://titubb.titck.gov.tr/`. (Accessed on 01/04/2021).

[59] TİTCK, "Skrs 3 e-reçete ilaç listesi." `https://www.titck.gov.tr/dinamikmodul/43`. (Accessed on 01/04/2021).

Appendix A

T1DM In-Silico Subjects

We want to share details regarding the UVa/Padova 2008 Simulator and particularly its virtual subjects. The model has 21 variables describing the glucose-insulin system of a type 1 diabetes subject where they contain a total of 26 free parameters [43]. Hence, each *in-silico* subject is represented by a parameter vector of size 26. We provide names of the variables and free parameters in Table A.2 and A.3, respectively. Similarly, we also list the parameters for the standard formula-based bolus insulin calculator for all in-silico patients in Table A.1. The open source implementation comes with 30 virtual patients with full set of parameters (10 for each age group, adult, adolescent, and child) [1], which we use in our simulations. The simulator author claims to collect them from their copy of the UVa/Padova 2008 Simulator academic version in MATLAB.

Table A.1: The standard formula-based bolus insulin calculator parameters of 30 *in-silico* subjects, which come with the open source implementation [1].

Patient	Body Weight (kg)	Age	ICR (g/U)	CF (mg/dl/U)	Total Daily Insulin (U)
adult#001	102.32	61	10	8.77	50.42
adult#002	111.10	65	8	9.21	57.87
adult#003	81.63	27	9	17.93	56.43
adult#004	63.00	66	16	42.65	33.81
adult#005	94.07	52	5	8.23	68.32
adult#006	66.10	26	10	18.21	61.39
adult#007	91.23	35	22	26.15	42.01
adult#008	102.79	48	13	12.25	42.78
adult#009	74.60	68	5	7.64	67.21
adult#010	73.86	68	5	10.69	64.45
adolescent#001	68.71	18	12	15.04	36.73
adolescent#002	51.05	19	5	13.18	62.03
adolescent#003	44.79	15	23	33.53	24.24
adolescent#004	49.56	17	14	21.82	35.25
adolescent#005	47.07	16	12	20.91	34.00
adolescent#006	45.41	14	7	17.70	49.58
adolescent#007	37.90	16	8	12.49	43.64
adolescent#008	41.22	14	4	11.94	63.39
adolescent#009	43.89	19	21	20.01	24.08
adolescent#010	47.38	17	14	31.87	33.17
child#001	34.56	9	25	42.72	17.47
child#002	28.53	9	23	36.92	18.18
child#003	41.23	8	22	31.05	16.02
child#004	35.52	12	25	40.72	19.82
child#005	37.79	10	7	33.63	40.93
child#006	41.00	8	19	39.98	20.22
child#007	45.54	9	8	25.00	36.21
child#008	23.73	10	15	30.88	21.49
child#009	35.53	7	25	35.32	17.39
child#010	35.21	12	18	29.22	20.65

Table A.2: Names of 21 variables describing the glucose-insulin system of a type 1 diabetes subject in UVa/Padova T1DM 2008 model [2].

Notation	Description
G_p (mg/kg)	glucose mass in plasma and rapidly equilibrating tissues
G_t (mg/kg)	glucose mass in, and in slowly equilibrating tissues
G (mg/dl)	plasma glucose concentration
EGP $(mg/kg/min)$	endogenous glucose production
Ra $(mg/kg/min)$	glucose rate of appearance in plasma
E $(mg/kg/min)$	renal excretion
U_{ii} $(mg/kg/min)$	insulin-independent glucose utilization
U_{id} $(mg/kg/min)$	insulin-dependent glucose utilization
I_p $(pmol/kg)$	insulin mass in plasma
I_l $(pmol/kg)$	insulin mass in liver
I $(pmol/liter)$	plasma insulin concentration
R_i $(pmol/kg/min)$	rate of appearance of insulin in plasma
I_d $(pmol/liter)$	delayed insulin
Q_{sto} (mg)	amount of glucose in the stomach (solid, Q_{sto1}, and liquid phase, Q_{sto2})
Q_{gut} (mg)	glucose mass in the intestine
F_{cns} $(mg/kg/min)$	glucose uptake by the brain and erythrocytes
X $(pmol/liter)$	insulin in the interstitial fluid
I_{sc1} $(pmol/kg)$	amount of nonmonomeric insulin in the subcutaneous space
I_{sc2} $(pmol/kg)$	amount of monomeric insulin in the subcutaneous space
$IIR(t)$ $(pmol/kg/min)$	exogenous insulin infusion rate
HE $(dimensionless)$	hepatic insulin extraction

Table A.3: Names of 26 free parameters describing a virtual subject in UVa/Padova T1DM 2008 model [2].

Notation	Description
V_G (dl/kg)	distribution volume of glucose
k_1 (min^{-1})	rate parameter of glucose kinetics
k_2 (min^{-1})	rate parameter of glucose kinetics
m_1 (min^{-1})	rate parameter of insulin kinetics
m_2 (min^{-1})	rate parameter of insulin kinetics
m_3 (min^{-1})	rate parameter of insulin kinetics
m_4 (min^{-1})	rate parameter of insulin kinetics
k_{p1} $(mg/kg/min)$	extrapolated EGP at zero glucose and insulin
k_{p2} (min^{-1})	liver glucose effectiveness
k_{p3} $(mg/kg/min\ per\ pmol/liter)$	parameter governing amplitude of insulin action on the liver
k_i (min^{-1})	rate parameter accounting for delay between insulin signal and insulin action
k_{gri} (min^{-1})	rate of grinding
k_{empt} (min^{-1})	rate constant of gastric emptying, which is a nonlinear function of Q_{sto}
k_{abs} (min^{-1})	rate constant of intestinal absorption
f	fraction of intestinal absorption that actually appears in plasma
BW (kg)	body weight
D (mg)	amount of ingested glucose
V_{m0} $(mg/kg/min)$	Michaelis-Menten parameter of glucose utilization at zero insulin action
K_{m0} (mg/kg)	Michaelis-Menten parameter of glucose utilization at zero insulin action
V_{mx} $(mg/kg/min\ per\ pmol/liter)$	disposal of insulin sensitivity
p_{2U} (min^{-1})	rate constant of insulin action on peripheral glucose utilization
k_{e1} (min^{-1})	glomerular filtration rate
k_{e2} (mg/kg)	renal threshold of glucose
k_d (min^{-1})	rate constant of insulin dissociation
k_{a1} (min^{-1})	rate constant of nonmonomeric insulin absorption
k_{a2} (min^{-1})	rate constant of monomeric insulin absorption

Appendix B

Clinical Data Collection

This section explains how our clinical data comes into existence in two steps: creating small data and collecting insulin recommendations from a clinician we collaborate with in a local hospital.

In preparation of the data, we use five of the ten *in-silico* adult patients that come with UVa/Padova T1DM 2008 Simulator, with identifiers 2, 5, 7, 8, and 10 [43], [1]. For each patient, there are 20 training and 20 test contexts. Training data acts as a patient's historical responses to the treatment, and test data as clinician's subjective assessments. Clinician primarily gets insights regarding patient's treatment response by examining training data; then, proceed with their recommendation on test data.

The context variables are carbohydrate intake named as *Carb*, fasting blood glucose named as *Fasting BG*, and time between a meal and postprandial BG measurement named as *Measurement Time*. Since we learn individualized models in our work, we discard variables characterizing inter-patient treatment responses on purpose to avoid false or misleading assumptions. We only include variables deemed valuable to intra-patient variability for diabetes. We get insulin recommendations for each training sample with SimCalc with the BG target of (70+180)/2=125 mg/dl. Then, we compute the postprandial BG response of the

Table B.1: Clinical data regarding to five *in-silico* adult patients of UVa/Padova T1DM 2008 Simulator.

Patient	Training Data					Test Data		
	Carb (g)	Fasting BG (mg/dl)	Measurement Time (min)	SimCalc Insulin (U)	Postprandial BG (mg/dl)	Carb (g)	Fasting BG (mg/dl)	Clinician Insulin (U)
adult#002	200	140	170	29.5	199.0	60	135	6.0
	90	105	200	10.6	143.0	40	145	4.5
	125	130	210	15.8	139.0	80	110	7.2
	65	90	170	4.9	149.0	20	155	2.7
	65	140	120	10.3	179.0	45	115	4.1
	65	145	120	8.5	191.0	55	95	5.0
	30	135	130	5.1	150.0	68	135	6.5
	75	140	190	10.7	141.0	60	130	5.6
	140	120	180	18.1	165.0	60	95	5.4
	45	90	210	1.9	116.0	50	90	4.5
	20	150	140	5.8	135.0	130	140	12.2
	150	125	160	15.4	213.0	63	140	6.1
	150	160	170	21.6	185.0	130	130	11.9
	60	100	140	4.9	156.0	65	160	6.9
	65	100	160	6.2	151.0	30	90	2.7
	30	95	140	0.6	130.0	52	155	5.6
	40	115	160	3.9	153.0	75	95	6.8
	30	130	190	3.8	133.0	90	150	8.8
	25	160	190	6.3	126.0	75	95	6.8
	60	95	210	3.9	138.0	65	105	5.9
adult#005	35	140	190	10.0	132.0	15	110	2.9
	100	135	170	16.8	206.0	60	120	11.8
	30	105	130	3.6	158.0	88	120	17.3
	155	140	200	33.5	168.0	75	170	19.9
	70	155	190	16.1	151.0	30	140	15.8
	75	165	200	18.7	135.0	75	130	15.2
	100	140	140	22.4	219.0	45	100	8.8
	110	170	210	31.3	106.0	65	170	15.4
	120	95	190	23.2	131.0	40	110	7.8
	63	145	190	14.7	151.0	15	100	3.0
	71	155	190	15.3	165.0	65	120	12.7
	45	165	210	11.8	143.0	50	165	12.2
	45	95	200	4.3	129.0	15	150	4.5
	75	150	160	15.8	199.0	50	160	11.9
	20	160	140	8.7	162.0	40	90	7.8
	65	110	180	11.9	144.0	50	140	10.9
	75	95	160	8.9	178.0	45	115	8.8
	20	130	150	4.2	152.0	125	160	26.7
	50	100	120	8.5	171.0	50	120	9.8
	65	150	190	15.8	150.0	35	150	8.5
adult#007	55	95	160	1.4	160.0	23	155	1.4
	140	130	210	8.0	77.0	50	165	2.8
	15	95	180	0.0	130.0	78	140	3.6
	45	90	150	0.8	144.0	55	155	2.8
	65	160	150	3.4	164.0	60	170	3.2
	30	140	190	2.1	112.0	40	125	1.8
	60	140	190	2.5	145.0	12	155	0.9
	135	150	210	6.5	100.0	80	145	3.8
	25	95	190	0.0	135.0	75	120	3.2
	76	105	200	3.3	111.0	70	165	3.6
	55	145	210	3.0	114.0	140	140	6.3
	30	115	200	1.2	127.0	60	90	2.6
	14	120	190	0.4	127.0	46	100	2.0
	10	135	180	0.7	128.0	50	120	2.2
	60	170	160	3.8	144.0	40	90	1.7
	65	130	120	3.1	190.0	57	150	2.9
	60	155	190	4.8	90.0	20	130	1.0
	120	95	120	4.3	230.0	75	135	3.5
	45	155	190	2.7	123.0	70	150	3.4
	40	150	180	3.3	107.0	50	150	2.5
adult#008	60	145	210	6.1	143.0	70	125	2.7
	135	120	160	11.7	192.0	90	110	3.4
	30	100	120	0.2	140.0	60	140	2.4
	66	105	130	3.1	174.0	30	150	1.4
	25	105	210	0.3	122.0	100	170	4.3
	63	150	120	6.1	199.0	60	165	2.8
	23	170	130	4.9	151.0	50	150	2.2
	50	115	200	2.3	139.0	65	150	2.8
	55	155	210	6.3	133.0	45	165	2.2
	25	100	120	0.0	142.0	20	130	0.9
	80	145	200	8.2	140.0	20	145	1.1
	18	140	180	2.7	135.0	45	95	1.7
	60	90	160	1.8	144.0	74	135	3.0
	100	130	150	10.0	180.0	77	130	3.0
	72	150	120	7.1	198.0	20	95	0.7
	120	155	160	14.4	168.0	58	145	2.5
	15	105	180	0.0	118.0	115	90	4.4
	100	110	200	6.4	148.0	51	140	2.1
	150	105	130	9.5	246.0	135	160	5.5
	50	95	200	1.2	129.0	25	150	1.3
adult#010	60	130	140	15.3	188.0	14	160	5.4
	65	150	120	15.3	252.0	80	90	17.8
	115	115	120	17.2	329.0	45	160	12.3
	45	90	180	6.8	119.0	60	125	13.6
	85	125	140	16.7	234.0	70	130	16.1
	100	145	160	20.1	201.0	55	90	12.2
	65	165	210	17.9	114.0	37	95	8.2
	50	155	150	13.3	184.0	45	95	10.0
	65	90	160	10.2	153.0	90	170	22.9
	75	140	140	19.9	196.0	100	165	24.8
	74	165	180	22.6	128.0	26	135	6.7
	70	110	170	15.6	144.0	140	155	33.1
	80	135	210	16.9	112.0	175	165	41.5
	150	125	160	22.9	270.0	30	100	6.7
	44	115	140	7.1	182.0	10	165	4.8
	60	150	120	15.4	234.0	60	140	14.5
	60	155	150	15.2	190.0	45	140	11.2
	150	165	170	37.6	188.0	65	145	15.9
	60	120	150	13.9	170.0	85	105	18.9
	60	165	180	14.0	164.0	30	160	9.0

corresponding patient by using the simulator and add independent and identically distributed zero-mean Gaussian noise with the variance of 25. We assume the time between a meal and insulin intake to be zero. *Carb*, *Fasting BG* and *Measurement Time* values are picked from ranges 10-200 g, 90-170 mg/dl, 120-210 min, respectively. We ask a clinician to go through the training data and make insulin recommendations for each test context. These recommendations are listed under the name *Clinician Insulin*. The data is given in Table B.1.