Universidad Nacional de Colombia
Sede Medellín
Facultad de Ciencias

Posgrados en Matemáticas

Tesis de Maestría

Modelamiento y Asimilación de Datos de la Respuesta Glicémica en Humanos

Por: Daniel Fonnegra García

Director: Dr. Jorge Mario Ramírez Osorio

Marzo 2020

Acknowledgements

This project wouldn't have taken place without the priceless guidance of Jorge Mario Ramírez, who, from the undergraduate courses was a motivation to deeply explore the essence of math in any field of knowledge. I had never imagined myself proposing models for medical applications.

I also thank my parents Jaime Fonnegra and Jenny García, my brother Alejandro Fonnegra and my aunt Marta Fonnegra for being the support I needed in difficult times and for pushing me to become what I am today.

Last but not the least, I have to thank my friends, especially Jean Alvarado, Melissa Alguilar, Daniel Osorio and Santiago Rojas who shared with me the best years I have had so far and with whom I had the pleasure of sharing a home for an incredible year.

Contents

Resumen

Diseñamos un modelo fenomenológico parsimonioso para la homeostasis de la glucosa en humanos sanos. El modelo consiste en un sistema diferencial no lineal de dos depósitos que depende de un conjunto de parámetros con significado fisiológico, cuyos valores se encuentran ajustando el modelo a un conjunto de datos proporcionado. Los datos disponibles consisten en mediciones subcutáneas casi continuas de la glucosa junto con una lista de valores nutricionales de las comidas ingeridas por diferentes usuarios. El conjunto de parámetros del modelo se divide entonces en los que dependen de la comida y los que deben ser constantes en todas las comidas para cada usuario por separado. Con esta división, proponemos un algoritmo para predecir los parámetros de la comida teniendo como entrada el valor nutricional de la comida. Los resultados validan nuestro modelo porque los valores de los parámetros se encuentran dentro de los rangos normales de los humanos según la literatura disponible, mientras que al mismo tiempo, se ajustan los datos con errores muy bajos. Se propone un regresor de árbol aleatorio para predecir los valores de los parámetros dependientes de la comida que mejor se ajustan al modelo a partir de los valores nutricionales de la comida registrados por los usuarios. Encontramos que, desafortunadamente, los datos del valor nutricional de las comidas carecen de integridad y no pudimos encontrar un modelo que se ajustara a la relación entre el valor nutricional y los parámetros de la comida.

Palabras clave: glucosa, insulina, modelamiento matemático, homeóstasis, absorción

Modelamiento y Asimilación de Datos de la Respuesta Glicémica en Humanos

Abstract

We design a phenomenological parsimonious model for glucose homeostasis in healthy humans. The model consists of a two-reservoir nonlinear differential system depending on a set of parameters with physiological meaning, which values are found by fitting the model to a provided data set. The available data consists of almost continuous sub-cutaneous measurements of glucose together with a list of nutritional values of the meals ingested by different users. The set of model parameters is then split into those that are meal-dependent, and those that should be constant across meals for each user separately. With this split, we propose an algorithm to predict the meal parameters by having as input the nutritional value of the meal. The results validate our model because the parameter values fall within human normal ranges according to the available literature, while at the same time, fitting the data with very low errors. A random tree regressor is proposed to predict the values of the meal-dependent parameters that best fit the model from the meal's nutritional values logged by the users. We find that, unfortunately, the meals nutritional value data lack integrity and we could not find a model that fitted the relation between nutritional value and meal parameters.

Keywords: glucose, insulin, mathematical modeling, homeostasis, absorption

Modeling and Data Assimilation of the Glycemic Response in Humans

1 Introduction

Glucose is a simple sugar which the body absorbs, mainly from carbohydrates, when eating meals. It is used by our brain and body cells to metabolize ATP, the richest energy intermediary in the human body. Glucose homeostasis is one of the self regulation processes of the body in charge of keeping blood glucose within a certain range. This body orchestration contains lots of actors which perform their own roles; the small intestine is in charge of extracting glucose from meal digestion and transport it to the portal vein. In this manner, glucose enters the blood stream where it is distributed through our body. When glucose levels raise, a group of glucose-sensitive cells in the pancreas called "β-cells" start secreting insulin. Insulin is an hormone which activates the glucose uptake in the cells so that it becomes ATP through a process called glycolysis. As a consequence, blood glucose levels start falling and insulin secretion stops and the remaining amount ends up being metabolized by the liver, kidney and muscles. On the other hand, if the glucose levels fall below the expected ranges after long fasting periods, another group of glucose-sensitive cells in the pancreas called "α-cells" secrete an hormone called glucagon. Glucagon, stimulates the glucose production in the liver through a process called glycogenolysis. [11]

Nutrients are substances which organisms use to perform the tasks required for living. Nowadays, human being diet is a very complex composition of many different kind of nutrients prepared and combined in many different ways making feeding one of the main marks of a culture. This diversity in diet increased the complexity and generated many gray zones when trying to classify the ideal composition of a meal for a healthy life. The *nutritional value* is a set of labels which different jurisdictions around the world defined to quantify the composition of a meal based on some substances that the body requires to live. The most important groups in the nutritional value are the so called *macronutrients*: carbohydrates, proteins and fats. According to the National Academy of Medicine of the U.S, the recommended dietary allowance for adults between 18 and 30 years is composed of 300g of carbohydrates, 150g of proteins and 75g of fats[1]. From all macronutrients, carbohydrates are the main source of glucose in the body because they are simple chains of sugars which are easily broken down mechanically (chewing) and chemically (enzymes in the saliva and the intestin) into three monosaccharides: glucose, fructose and galactose [19]. Proteins, on the other hand, are macromolecules of aminoacids which constitute the 20% of the human body and have a main role in almost all vital functions and activities such as walking, digesting and proper immune system functioning [19]. Finally, fats, constitute the secondary energy source of the body which functions as a reserve because glucose cannot be stored in the body for a long period. In contrast, fats are dense sources of energy which can be stored in the fatty tissues almost without limits [19]. Although not part of the macronutrients, in this project we will study also the impact of the fibers in the rate of appearance of glucose since it has a main role in the digestion process by promoting the movement of the material through the intestine. It is important to mention that fibers are not digested and so they are not a source of glucose nor energy in the body[2].

The study of glucose homeostasis has had an increasing importance in this century due to the high incidence of diabetes cases in the world. According to WHO, diabetes is the major cause of

[1]Institute of Medicine. Dietary Reference Intakes: The essential guide to nutrient requirements. Washington (DC): The National Academies Press; 2006.

[2]Kim Y, et al. Dietary fibre intake and mortality from cardiovascular disease and all cancers: A meta-analysis of prospective cohort studies. Archives of Cardiovascular Disease. 2016;109:39.

blindness, kidney failure, heart attacks, stroke and lower limb amputation, and it is one of the ten leading causes of death in the world which numbers have almost doubled in the last three decades [10].

The objective of this project is to design a mathematical model with a set of parameters with physiological sense that describe the dynamics of glucose and which can be fit using a small amount of data. A model with two state variables is proposed: $G(t)$ and $I(t)$. $G(t)$ denotes the blood glucose levels in $\mathrm{mg\,dL^{-1}}$ as this is the easiest and most common way to measure glucose. On the other hand $I(t)$ would ideally denote the blood insulin levels, but as this is a hard to measure variable, $I(t)$ will denote a control for glucose levels, and must be strongly related to insulin, maybe tissue insulin, blood insulin or a combination of both. Only glucose data measured through a non-invasive sensor is available.

We have found a wide range of complexity among the many studies attempting to mathematically model glucose dynamics: from minimal linear models with four parameters to highly non-linear models with 30+ parameters. One of the most interesting models, presented in [8], compares seven models using three critera: identifiability, meaning of parameters and goodness of fit. The results, showed that the model with the best performance was the one where the interaction between $G(t)$ and $I(t)$ has the simple form $-G(t)I(t)$. Another very interesting model was the one proposed by Dalla Man [1]. Contrary to the previous one, this model was not about simplicity but phenomenology. They detail the whole glucose homeostatic process, design a model with around 35 parameters with physiological meaning and estimate their values for the mean population.

These approaches satisfy simplicity and phenomenological meaning individually but, they lack of usability. The first model didn't consider the rate of absorption but a controlled intravenous glucose input, and the second one contains a large number of parameters that make it impossible to fit the model to the individual dynamics of users. Additionally, both models find the values for the parameters under very controlled conditions. For this reason, we propose a model and fit it with data collected in the daily life of test users. It is a non-linear model consisting of two containers: Glucose and "Insulin", each with one source and one sink. For glucose, we consider meal ingestion $A(t)$ as the only source, and tissue uptake stimulated by insulin $U(G, I)$ as the only sink. For insulin, we consider β-cells secretion $S(G)$ as the only source and degradation $E(I)$ as the only sink.

$$\frac{dG}{dt} = A(t) - U(G, I)$$
$$\frac{dI}{dt} = S(G) - E(I)$$

There are many other sources and sinks of glucose like endogenous production and brain consumption [1]. Nevertheless as the glucose levels are held in a narrow range even during long fasting periods, we can approximately say that these endogenous mechanisms tend to cancel out or at least be negligible compared to the postpandrial dynamics.

In order to validate the proposed model and answer interesting questions, some test users provided a dataset of glucose measurements and meal registers. The glucose measurements consists of a time series of blood glucose in $\mathrm{mg\,dL^{-1}}$ with a sample period of $\sim$ 15min obtained by a non-invasive glucose sensor. The meal registries contain the logging time of the meal, the amount of

calories in Cal, carbohydrates, proteins, fats and fibers in grams, and a description of the meal.

Over this data, a first exploratory data analysis was performed. It was found that the glucose time series has low amplitude noise with wide and short periods without data, while the meal data has lots of non consistent values when comparing the nutritional value with the description and relations between carbs and glucose peaks, which are more or less supposed to be directly related. To deal with these data issues, the glucose time series was filtered and interpolated, the nutritional value was fit using an analysis scheme method and the time series were divided by windows of individual meals.

At this point, we shall discuss what we call "reality" in this context, so that the proposed model can be compared to other models or to different solutions of the proposed one. From an ontological point of view, reality is the actual state of a system which in our case is the amount of glucose and insulin circulating in the blood plasma. A model is an approximation to the description of that reality, and in our case is a parametric set of solutions to a proposed differential equation from which we will choose the solutions which best approximate the closest evidence we have from reality namely the glucose measurements in the plasma. Here, the word parametric refers to a set of free parameters which describe the human scale in which the glucose-insulin dynamics interact. As there is a causality relation between plasma insulin and glucose levels, the missing knowledge of the first one can be estimated by supposing that the "right" levels of insulin get to the "right" levels of glucose. This could be very accurate if the mechanisms through which the insulin and glucose interact weren't so uncertain, meaning that they depend on unpredictable or hard to quantify variables like the human humor. A part of the very uncertain reality of the system is that the only evidence we have of its parameters are wide ranges within which most humans lie. Then we can enclose the set of solutions for those parameters that fall within those ranges. Even with that filter, the set of solutions which have a very good approximation is also big and so the parametric set.

One could simply say that from the set of all approximations we should choose the one which best fits the reality measure under some definition of error. An approach like this is ignoring the fact that a measure of reality is not reality itself but a mere approximation which is generally more accurate than models and is also subject of uncertainty. Then, it is more appropriate to choose models with results that yield within the uncertainty range of the measurements. This inability of choosing an unique solution is then considered as the uncertainty of the model parameters.

With the data cleaned and a proposed model, we can solve more interesting questions related to the project objective:

- Is there a set of parameters within the human range which fits the model to the glucose data?
- Can the model parameters be divided in metabolism related and meal related parameters so that the first are constant for all meal?
- Is there a relation between the meal related parameters and the nutritional value of the meal?

With this in mind, we divided the set of model parameters in metabolism related parameters and meal related parameters, the first ones are supposed to be strongly correlated for all windows and

the second ones are independent as glucose ingestion is a human choice. Then, to search the parameter values so that the model best fits the glucose data, we defined an error function that given a set of parameters, solves the differential equation and then computes the mean squared error between the solution and the data. This error function is the required input of the minimization algorithm L-BFGS-B used to perform the parameter fitting for each window. In order to find the constant values of the human parameters, we performed a first fitting for all windows and obtained a list of values for each parameter. Then, we chose the mode as the fixed estimation of the human parameters. Next, leaving the human parameters constant, we performed a second fitting from which we obtain the absorption parameters estimation for each meal. The results validated the proposed glucose model because first, the metabolic parameters fell in the normal human ranges for healthy people and second, because the model could be fit even when the metabolic parameters were taken constant.

Finally, to find a relation between the meal related parameters and the nutritional value logged by the user, a bottom-up approach was followed. First, a visual analysis showed no relation between any pair of variables: they all looked almost randomly distributed without any trend. Second, as mentioned before in the nutritional value paragraph, it is expected a strong trend between the ingested carbs amount and total glucose absorption as this is the main source of glucose in meal, namely:

$$\text{carbs} \propto \int_0^\infty A(t)dt$$

Unfortunately, not even this check was successful. Then a random forest regressor was implemented and resulted in a correlation coefficient lower than 0.3 in all cases, not enough to be considered statistically significant. The last attempt for finding relation was implementing a deep neural network combined with feature engineering, the resulting mean squared error was very close to the variance of the data, so the model best prediction is the average of the outputs.

This leads us to conclude that the provided meal data wasn't enough precise for modeling the dynamics of glucose ingestion. It is worthy to mention that it is very unlikely that this problem is related to the proposed model, as the absorption parameters always fell in human normal ranges and were consistent with the glucose data.

2 Mathematical Model

2.1 Previous Models

Glucose modeling has become a topic of major interest in the last 40 years due to the increase in the data availability and the technological advance in the measurement methods. As diabetes is one of the ten main causes of death, understanding the dynamics of glucose regulation has become a topic of major importance.

In one of the main referents of glucose modeling [8], the authors review a set of seven parsimonious models and through a set of criterions, they could choose the best performing model. To do so, they made a set of experiments in animals described as follows: First, after a full night fasting period, intravenous glucose was injected to the animals, then blood samples were taken every

minute to measure glucose $G(t)$ and insulin $I(t)$ concentrations. There were two types of models proposed: three leaving implicit the glucose regulation by insulin and four adding the insulin concentrations as an explicit variable. In this manner, the authors evaluated many different ways of modeling the glucose dynamics from previous authors results and phenomenological descriptions of the glucose homeostasis. The models were evaluated by using three criterions: *Identifyability*, which measures the uncertainty of the parameters after fitting the models to the real data, *meaning of parameters* which considers the physiological sense of the parameters values and *goodness of fit* which measures the level of fitting of the model to the data. These criterions showed that the best performing was one of the models with explicit insulin defined as:

$$\begin{aligned} \frac{dG}{dt} &= (p_1 - X)G + p_4 \\ \frac{dX}{dt} &= -p_2 X + p_3 I(t) \end{aligned} \tag{1}$$

Where $X(t) = (k_4 + k_6)I'(t)$, $I'(t)$ is a remote compartment from which insulin $I(t)$ acts and p_i, k_i are the parameters of the model. There are two important facts from this result that will be used later in this project:

- From the compared models, the best way of modeling the glucose insulin interaction, is with the simple interaction term $I'(t)G(t)$, in the sense that it seems physiologically found and has a good fitting to the data.
- The insulin excretion rate can be modeled with an exponential decay term $-k_3 X$.

In addition, after the model selection, the authors in [8] focus on finding an estimation for the *insulin sensitivity* $I_s = 7.0 \times 10^{-4} \text{min}\,/\mu\text{U/mL}$ which, physiologically speaking, is related to the scale at which insulin boosts the glucose concentrations reduction. With this we will have the first accurate estimation of the parameters in the model to be proposed in subsequent sections.

In contrast in [1], another well known author proposed a very strictly designed phenomenological model which, based on advanced studies of the glucose homeostasis process, described deeper features of the glucose-insulin interaction. The model starts by talking about the main known mechanisms of glucose secretion and excretion and builds the base set of equations defined as:

$$\begin{aligned} \frac{dG_p(t)}{dt} &= EGP(t) + Ra(t) - U_{\text{ii}}(t) - E(t) - k_1 G_p(t) + k_2 G_t(t) \\ \frac{dG_t(t)}{dt} &= -U_{\text{id}}(t) + k_1 G_p(t) - k_2 G_t(t) \end{aligned} \tag{2}$$

There, $G_p(t)$, $G_t(t)$ represent the glucose concentrations in plasma and tissues respectively, $EGP(t)$ the endogenous glucose production stimulated by the low levels of blood glucose, $Ra(t)$ rate of glucose appearance which is the glucose coming from the meal ingestion, U_{ii} the insulin independent glucose utilization which happens in organs like the brain and the liver by diffusion, $E(t)$ the glucose excretion in the kidneys, $U_{\text{id}}(t)$ the insulin dependent glucose utilization where the cells are stimulated by insulin to absorb glucose and k_i which are the exchange parameters between the glucose in the tissues and the blood. Including the expressions for each term in equation (2) the model has more than 30 equations, so it is very suitable to precisely describe the body mechanisms but unsuitable for fitting to the dynamics of a particular user as it would require huge

amounts of data. Nevertheless, beyond the phenomenological understanding, this model contains two additional results which will be important later in this project:

- As the author presents estimations for each term in equation (2), we can compare and determine if some terms are negligible compared to others in order to simplify the model. An example of this is the endogenous glucose production which is shown to provide glucose ten times less than the rate of exogenous glucose appearance after a meal.
- The author also computes estimation of the model parameters, giving us another source for our model parameters validation.

2.2 Proposed Model

Models like the ones presented in [1, 2, 8] are very useful when studying the phenomenology of the glycemic response of the average population. Unfortunately, the first two depend on a large number of parameters that make it hard to fit the dynamics of a sample person. The second was fitted under controlled conditions and its absorption term is an intravenous source of glucose, so both lack of usability in a daily context.

Here we propose a simple 2-containers model (Figure 1.). The first container represents the plasma-glucose state variable ($G(t)$ in $\mathrm{mg\,dL^{-1}}$) and the second one the controller variable ($I(t)$ in $\mathrm{pmol\,L^{-1}}$) (We do not call this plasma-insulin because as insulin data is hardly found, there is no way to check if the dynamics of this controller correspond to the ones of plasma insulin, tissue insulin or a combination). The glucose container has one source $A(t)$ and one sink $U(G(t), I(t))$ representing the glucose absorption and utilization respectively. The insulin container has also one source $S(G(t), I(t))$ and one sink $E(I(t))$ representing the insulin secretion and excretion respectively.

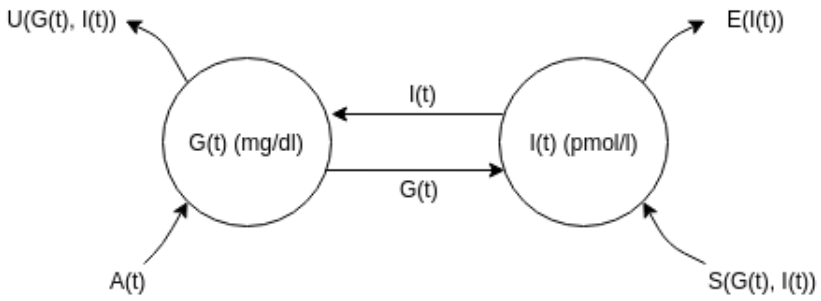

Figure 1: Containers model

2.2.1 Glucose Absorption

In this simplified model, only one source for glucose is considered: glucose ingestion denoted by Ra in (2) and by A in what follows. The phenomenology of glucose ingestion is very complex because it depends on lots of human metabolism variables and the variety of components in a meal [2, 5]. On the other hand, the functional shape of the absorption is, in general, simple (See Figure 2).

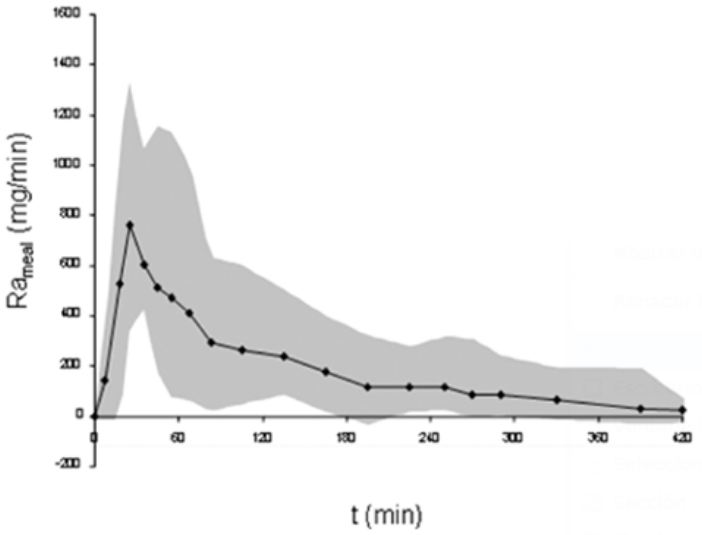

Figure 2: Typical response for postpandrial rate of glucose appearance $A(t)$ as modeled by [2]

So, instead of trying to find a phenomenological model for the absorption term, a set of parametric functions with a shape like in Figure 2 will be proposed.
Let $\mathscr{A}$ be the family of parametric functions $A(t; p_1, p_2, p_3)$ such that:

$$A(t) = \frac{p_1}{t^2(e^{\frac{p_2}{t}} + p_3)} \ , \ t > 0 \ , \ A(0) = 0, \tag{3}$$

with:

$$p_1 \geq 0, \ p_2 > 0, \ p_3 \geq -1 \tag{4}$$

Although this function has the desired shape as shown in Figure 3, the parameters do not represent meaningful features of the absorption function (like the size of the peak, the time of the peak or the rate of growth and decay), so the next step is to change the base of the parameter space so that the new basis do represent meaningful features in the absorption function.

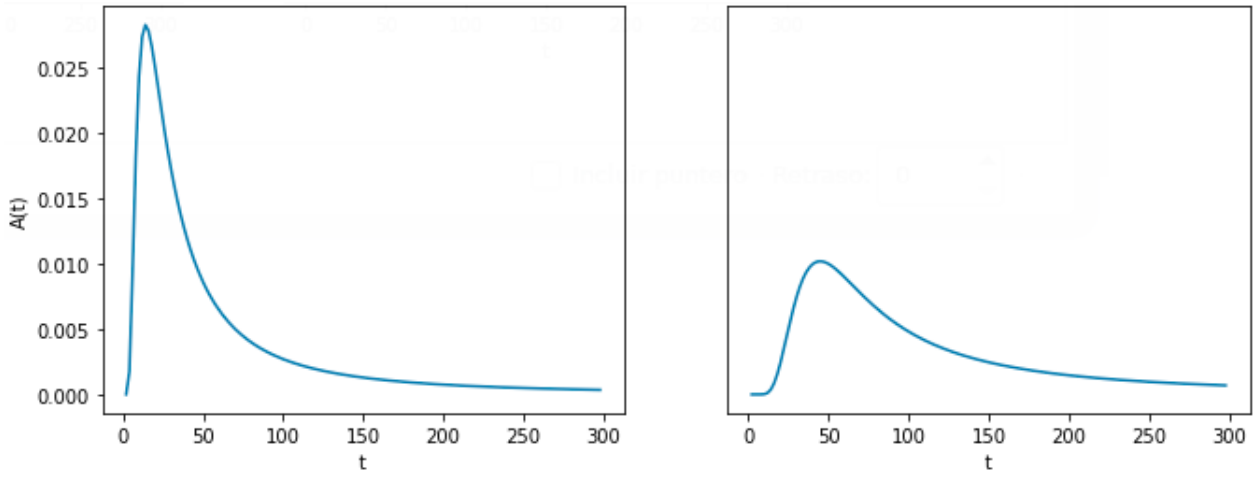

Figure 3: (a) Rate of glucose appearance for $p_1 = 50\text{mg}\,\text{dL}^{-1}\,\text{min}^{-1}$, $p_2 = 30\text{min}$, $p_3 = 0.5$
(b) Rate of glucose appearance for $p_1 = 400\text{mg}\,\text{dL}^{-1}\,\text{min}^{-1}$, $p_2 = 120\text{min}$, $p_3 = 5.0$

To find the time and size of the peak we compute the derivate of the absorption:

$$\frac{dA}{dt} = \frac{-p_1(2t(e^{\frac{p_2}{t}} + p_3) - p_2 e^{\frac{p_2}{t}})}{t^4(e^{\frac{p_2}{t}} + p_3)^2} \tag{5}$$

Lets define $s = \frac{p_2}{t}$, then with $\frac{dA}{dt} = 0$

$$s = \frac{2(e^s + p_3)}{e^s} \implies (s-2)e^{s-2} = 2e^{-2}p_3 \tag{6}$$

The last equality, can be easily solved using the lambert function $W(z)$ as $W(ze^z) = z$, so:

$$s(p_3) = 2 + W(2e^{-2}p_3) \tag{7}$$

Now, calling $t_{\max}$ the time of the maximum, we get to the first parameter change equation

$$p_2 = s(p_3)t_{\max} \tag{8}$$

Then, evaluating $t = t_{\max}$ in (1)

$$\begin{aligned} A(t_{\max}) = A_{\max} &= \frac{p_1}{t_{\max}^2(e^s + p_3)} = \frac{2p_1}{t_{\max}^2 s e^s} \\ \implies p_1 &= \frac{A_{\max} t_{\max}^2 s e^s}{2} \end{aligned} \tag{9}$$

Finally, we set $p_3 = e^a - 2$. This ensures that when $0 \leq a < \infty \implies -1 \leq p_3 < \infty$ and sets an exponential scale p_3, this is useful because we checked computationally that by varying p_3 linearly, the function $A(t)$ remains almost constant.
With this parameter base changed, the absorption function looks like (See Figure 4):

$$A(t) = \frac{A_{\max} t_{\max}^2 s(a) e^{s(a)}}{2t^2(e^{\frac{s(a)t_{\max}}{t}} + e^a - 2)}, \quad s(a) = 2 + W(2e^{-2}(e^a - 2)) \tag{10}$$

Figure 4 shows that the size of the peak has the same value of $A_{\max}$ and the time at which the peak occurs is $t_{\max}$. The parameter a is a little bit more abstract but one may check that as long as it increases, the peak gets sharper.

2.2.2 Glucose Utilization

Glucose utilization refers to the mechanisms through which glucose is converted to ATP within the cells and later to energy used to perform all of the cell's vital functions. There are two main mechanisms of glucose utilization: insulin-independent and insulin-dependent glucose utilization. The first occurs in the brain and erythrocytes and is absorbed by the cells through diffusion, and the latter occurs in tissues throughout the body. As seen in [1] the amount of insulin independent glucose utilization is of the same order as the glucose endogenous production (recall $EGP(t)$ from Equation (2)), this makes sense because when oral glucose ingestion is not provided, both systems must stabilize so that the plasma glucose remains around a basal level. For this reason, we will not consider endogenous production nor insulin independent utilization in the model.

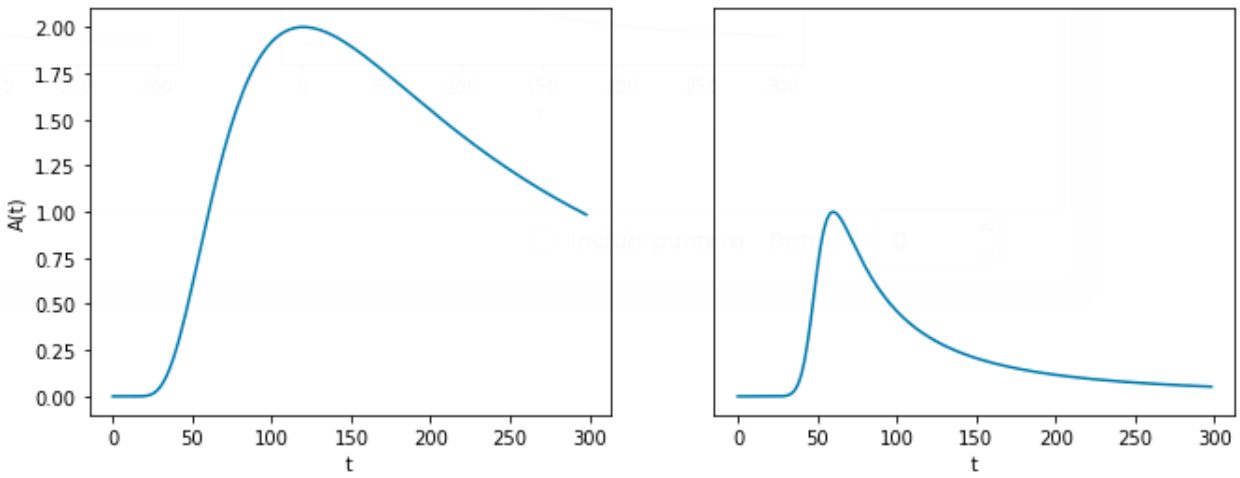

Figure 4: (a) Rate of glucose appearance $A(t)$ in (10) for $A_{\max} = 2\text{mg}\,\text{dL}^{-1}\,\text{min}^{-1}$, $t_{\max} = 120\text{min}$, $a = 1$ (b) Rate of glucose appearance for $A_{\max} = 1\text{mg}\,\text{dL}^{-1}\,\text{min}^{-1}$, $t_{\max} = 60\text{min}$, $a = 10$

Based on the well known minimal model [8], we proposed the simplest form for insulin-dependent glucose utilization which we also refer as the "interaction term" given by:

$$U(G(t), I(t)) = I_s G(t) I(t) \tag{11}$$

Where the parameter I_s is the insulin sensitivity. In [8] the enhancement of glucose disappearance due to glucose concentrations is defined as:

$$\text{EG}(t) = -\frac{\partial}{\partial G}\left(\frac{dG}{dt}\right) \tag{12}$$

which they use to define insulin sensitivity as

$$S = -\frac{\partial \text{EG}_{\text{ss}}}{\partial I_{\text{ss}}} \tag{13}$$

where the subindex "ss" refeers to the stationary state. One may check from (11) that $S = I_s$, so we can later compare their values to validate the model.

2.2.3 Insulin Secretion

When the glucose raises and surpasses a trigger level, the pancreas starts secreting insulin to the blood. Here we propose a translated Exponential Linear Unit (elu) function for this interaction because it has the following properties:

1. elu(x) is linear for $x \geq 0$, so we can fit translation parameters so that the glucose trigger is set to a reasonable value. Besides, a linear insulin secretion is a well performing estimation for this interaction based on [1].

2. elu(x) is exponential for $x < 0$, this is desirable because ensures that elu(x) is differentiable but rapidly decreasing.

So the translated elu funtion proposed is:

$$S(G(t)) = \begin{cases} I_{\text{slope}}(G(t) - G_{\text{act}}) & G(t) > G_{\text{linear}} \\ e^{I_{\text{slope}}(G(t)-G_{\text{act}})-1} & G(t) \leq G_{\text{linear}} \end{cases} \tag{14}$$

Where $G_{\text{linear}} = \frac{1}{I_{\text{slope}}} + G_{\text{act}}$. As a result $S \in C^1(\mathsf{R})$ which will prove useful in the optimization method to be used below. Note that the dimensionless secretion function due to glucose S defined in (14) is scaled by a factor of $1\text{pmol}\,\text{L}^{-1}\,\text{min}^{-1}$ which we decided to omit for simplicity but which is required for units consistency.

2.2.4 Insulin Excretion

Insulin has two reduction mechanisms, tissue degradation and liver extraction. Most authors [7, 8], even the ones with the more complex models [1], propose exponential decaying models for both terms. So as our controller variable is probably a combination between plasma and tissue insulin, it makes sense to use a single exponential decaying model for the insulin excretion:

$$E(I(t)) = -I_{\text{decay}} I(t) \tag{15}$$

2.2.5 Final Model

Before writing the final model, we must state one last consideration. It is very likely that the time between two meals is not long enough so that the absorption of the last meal is negligible at the time when the next meal ingestion starts. So by means of increasing the model precision we will add the $A_{-1}(t)$ term which corresponds to the absorption remaining of a prior meal. So, our model can be written as:

$$\begin{aligned} \frac{dG}{dt} &= A(t) + A_{-1}(t) - U(G(t), I(t)) \\ \frac{dI}{dt} &= S(G(t)) - E(I(t)) \\ I(0) &= I_o \\ G(0) &= G_0 \end{aligned} \tag{16}$$

Where A is defined by (10), U by (11), S by (14) and E by (15).

2.3 Model Parameters

With the model well defined, its necessary to clarify the physiological meaning of the parameters, its units and normal ranges. It was easy to find most common values and ranges for our parameters in the available literature. Table 1 summarizes the results.

The first approximation of common, minimum and maximum value was obtained by taking the average from multiple references. Then during the process of parametrizing the model to fit the data we found it necessary to slightly expand the ranges and adjust the common values so that the optimization could get to a better local minima. We believe there is no problem by changing these ranges and values since our model is not exactly the same as the ones in literature. The initial ranges serve the purpose of an useful baseline.

Param	**Description**	**Common Value**	**Min Value**	**Max Value**	**Units**	**Refs**
$A_{\max}$	Maximum glucose absorption rate after the current meal	0.472	0.0	10.0	$\frac{\text{mg}}{\text{dL}\,\text{min}}$	[2, 3]
$t_{\max}$	Time of the maximum glucose absorption rate	60	10	180	min	[2, 3]
a	Dimensionless parameter which defines the sharpness of the absorption function	0.159	0	10	-	-
I_s	Insulin sensitivity	1.58×10^{-4}	10^{-5}	10^{-3}	$\frac{\text{L}}{\text{pmol}\,\text{min}}$	[8, 7, 14, 15]
I_{slope}	Rate of insulin secretion due to plasma glucose, scaled by a factor of $1\text{pmol}\,\text{L}^{-1}\,\text{min}^{-1}$	0.158	10^{-2}	1.0	$\frac{\text{dL}}{\text{mg}}$	[7, 14]
G_{act}	Beta-cells trigger where the insulin secretion due to plasma glucose concentration starts	86.4	40.0	100.0	$\frac{\text{mg}}{\text{dL}}$	[1]
I_{decay}	Exponential decay constant of insulin in tissues and liver	0.0208	0.001	0.1	min^{-1}	[7, 14]
I_o	Initial insulin	30.0	0	140	$\frac{\text{pmol}}{\text{L}}$	[1, 3]

Table 1: Model parameters description, ranges and units found in the relevant literature.

3 Data

3.1 Description

The data set consists of two files for each of the 18 test users: the first file contains the time series with glucose measurements, and the second one the nutritional value of each meal registered by the user.

3.1.1 Glucose Time Series

The first file, contains five columns but only the following three are used in the model: The user id, the local activity timestamp and the glucose amount.

- The user id is a hashed id used to index the user but without exposing its real identity.
- The local activity timestamp is an ISO8601 formatted datetime containing the day and hour of the user in the timezone where the measure was performed.
- The glucose amount is the sensor measure of plasma glucose in $\mathrm{mg\,dL^{-1}}$, depending on the sensor it contains more or less one measure every 5 to 15 minutes with a 15% error of the measure range.

Figure 5 shows a preview of the provided data for one test user between May 31th and June 1st.

3.1.2 Meal Data

The second file, contains 13 columns from which we will use eight. user id, local activity timestamp, meal name list, amount of calories, carbohydrates, proteins, fats and fiber.

- The meal name list is a comma separated list containing the name of each item in the current meal.
- The amount of calories are Calories.
- The amount of carbohydrates, proteins, fats and fibers are in grams.

Table 2 shows a preview of the meal data. The user id and the local activity timestamp were omitted in the table because they have the same format as in the glucose data.

3.1.3 Data acquisition

Due to confidentiality terms of the company which provided us the data, we cannot mention many details of the data acquisition. Broadly, the glucose data was taken with a subcutaneous sensor worn by the user on his/her arm for a continouos period of time. The sensor estimates plasma glucose concentrations with a sampling rate of 15min with and uncertainty level of around 15%. Nutritional value data was manually logged by the user as follows. Through a cellphone app, the users searched a large database of commonplace meals for the meal they were about to take. The database includes menu items at most restaurants with average serving sizes. If the meal was prepared by the user, then he or she had to log the ingredients, courses or meal components, and their approximate quantities. This introduces big sources of error in the project like the following:

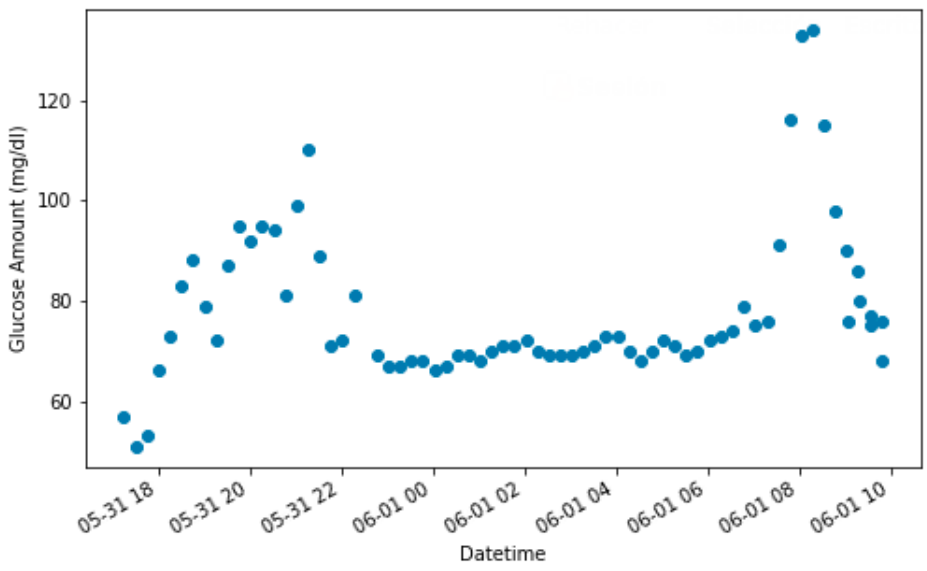

Figure 5: Glucose Time Series Data

Meal List	**Calories** (kcal)	**Carbs** (g)	**Proteins** (g)	**Fats** (g)	**Fiber** (g)
Hummus, Onions, Vegetarian Hot Dogs, Olive Oil, Lentils, Beer, White Rice, Vegetable Salad	1261	127	44	66	24
Watermelon	23	5.8	0.46	0.11	0.30
Cereals, QUAKER, Instant Oatmeal Organic, Regular, Unsweetened Fortified Rice Milk, Coffee, Strawberries	298	57	8	5	6
Brewed Green Tea	2	0	0.54	0	0
Whole Wheat Toast, Arugula, Guacamole	265	29	6	16	10

Table 2: Meal Data sample containing the nutritional value for some ingested foods logged by one of the test users

- If the user didn't have clear at which moment he/she should log the data, there might be someones who log it when starting to eat, when they finished it or when they start to cook it. Even if they had a standard, logging it is something likely to be forgotten.
- Not all packages have a nutritional value table, so when this happens the user will probably ignore those contents logging an underestimation of the food content.
- It is very unlikely that the users measure the weight of each ingredient of a food, instead he/she probably make a visual estimation.
- When the users did not cook the food, the estimation might be even worst because it could be based on what they think the food contains and the method of cooking.

We will show in section five that the nutritional value data logged by the user as described previously seems to be very noisy with meals reported in the wrong timestamp and nutritional values which showed no relation with the glucose signal.

3.2 Data Cleaning

Data acquisition is always prone to lots of sources of error, even more when the acquisition is performed under the discretion of users like the meal data. In the next lines we will list some issues that we found in the glucose and meal data and show what we did for cleaning.

- The glucose series with acquisition period of $\sim$ 15min was not enough as we are trying to fit a model to windows of 2-3 hours (8-12 data points).
  Solution: The data was augmented to a resolution of 5min by performing a cubic spline interpolation. This makes sense as homeostasis is a very smooth process.
- Some spans of the time series were very noisy which is probably related to temporary malfunctioning of the sensor because the sensor measurements were back to normal after some time.
  Solution: After the data augmentation, a low pass filter was implemented using a centered rolling mean of 30min.
- The timestamps in the glucose series were not uniform and there were spans wider than 15min without data. The non-uniformity of the data is not a big deal except for some posterior preprocessing algorithms, but the missing values are.
  Solution: The interpolation of the first item solved both problems but we limited it to 60 minutes as this is a reasonable period where abrupt changes are not expected to occur.
- The timestamp of the meal data was not correctly logged by the user as this was completely open to his or her prerogative. Sometimes the meal appears after the glucose peak, sometimes many hours before it and sometimes it was not even reported, we know this because there were some high glucose peaks with no logged meal nearby.
  Solution: Defining a valley as a the instant t_i of the discrete time series G such that $G(t_{i-1}) \geq G(t_i) < G(t_{i+1})$, and a peak the instant t_i such that $G(t_{i-1}) < G(t_i) \geq G(t_{i+1})$, each meal timestamp was moved to the valley closer to them in the interpolated glucose series. If there were multiple meals within the same valley, they were aggregated in a single meal. This wouldn't solve all of the cases and could even generate noise, but it was a solution

different to manual fixing with the best performance from many different methods that we tried.

- There were meals whose glucose peak was not consistent with the reported meal carbohydrates amount. i.e. very high peaks associated to a coffee cup and very low peaks associated a hamburger with coke.
  Solution: The details of how to fix this are explained later in section 6.

Figure 6 contains a comparison between the data before cleaning and after cleaning.

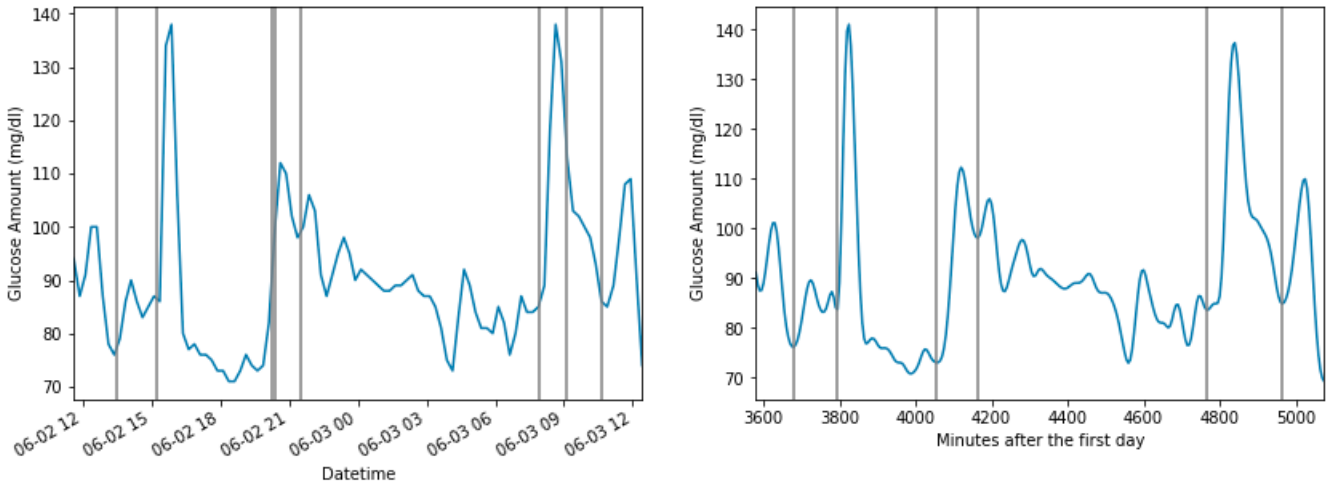

Figure 6: (a) Glucose time series before cleaning (b) Glucose time series after cleaning. The vertical lines correspond to the timestamps of the meal. Notice for example that the third and fourth lines from left to right in (a) were moved to its closest valley and aggregated in a single meal in (b).

3.3 Data Segmentation

Trying to fit a model to the full time series of glucose makes no sense as our model suppositions hold only for postpandrial processes. So instead of that, the time series was divided by windows starting with meal ingestion and finishing in the first valley after the meal glucose peak ensuring that each window contains a single peak (See Figure 7). Typically, the windows cover a span from 1 to 3 hours. Table 3 shows the number of glucose windows, for some of the test users, obtained after the data segmentation.

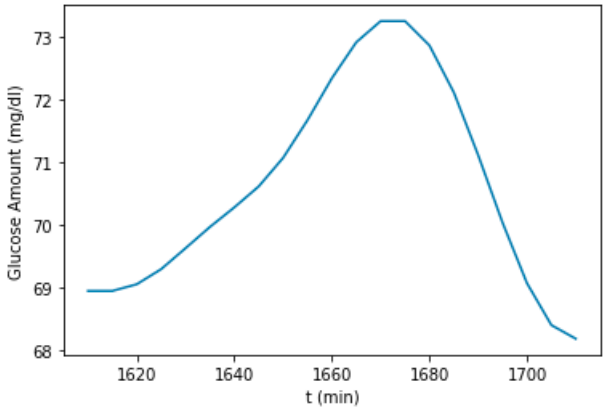

Figure 7: Example of glucose window obtained after data segmentation. The time axis represents the number of minutes that have passed since the first log of user h5mxumnh after the ingestion of one cup of coffee.

User id	**Number of windows**
1xvbq4gd	52
6ev6bjpg	50
7bucr6u7	104
526lmnt7	95
h5mxumnh	247
hj0jqplt	45
iz5z5ajq	70
nah5br7n	38
pcrxnt65	79
Topaz	62
x2lubmmk	35
Total	877

Table 3: Number of windows for each user obtained after data segmentation. The best and most reliable data corresponds to user h5mxumnh

4 Parameter Fitting

With the model chosen and the data cleaned we can begin to discuss how the parameters of the model can be found and how they are related to a person and the meal he or she ingests. This parameter finding will be the first check of the performance of the model in the sense that if the parameter values fall within human normal ranges, we can say that the model is causal to the endogenous human parameters and the exogenous meal parameters. As mentioned before, the dynamics of the digestion processes are very complex, so in the next chapter we will propose a black box model which will be trained to relate meal information (calories, carbs, proteins, fats, fiber) with the absorption parameters in the model ($A_{\max}$, $t_{\max}$, a). Fortunately, the meal information is not relevant in the current model so we will focus on finding the model endogenous parameters.

We will build our parameter finding algorithm standing upon the following two suppositions:

1. Our parameters can be divided into two classes "Human parameters" and "Meal parameters" representing attributes that depend strongly on the each person's metabolism, and the content of each ingested meal respectively.

2. The human parameters are approximately constant for each person. It is been shown [20] that the metabolic responses to glucose change with the circadian cycles, so the human-parameters should present slight variations throughout the day. However, the model showed a high performance without this consideration, and as we are looking for simplicity, we will not include this variation in this project.

Looking into the model (16) and the Table 1, it makes sense that the parameters mostly related with the meal parameters are $A_{\max}$, $t_{\max}$, a because they are the parameters related to the postpandrial glucose absorption $A(t)$. On the other hand, I_s, I_{slope}, G_{act}, I_{decay} are mostly related to each person's metabolism because they describe the response of the body to changes in glucose concentrations.

4.1 Optimization Algorithm

In order to find a set of parameters $\{p_i\}_{i=1}^n$ which minimize the distance between a parametric function $f_{p_1 \dots p_n} \in C(\mathsf{R})$ and a "true" function $g \in C(\mathsf{R})$ we need to define three things:

1. The parametric function.

2. The information we have about the true function.

3. A metric for $C(\mathsf{R})$ or error function.

4. A box of boundaries in the parameter space $R \subseteq \mathsf{R}^9$.

5. A point $p_o \in \mathsf{R}^9$ where the optimization will start.

4.1.1 Parametric Function

For simplicity reasons we will define $\{p_i\}_{i=1}^9 = \{G_o, I_o, A_{\max}, t_{\max}, a, I_s, I_{\text{slope}}, G_{\text{act}}, I_{\text{decay}}\}$ with the order preserved. Our parametric function will be the glucose function obtained from solution of the model (16)), i.e,

$$f_{p_1 \dots p_9}(t) = G(t). \tag{17}$$

Note that we have included G_o and I_o as model parameters, although from those, only I_o must be fit as G_o is given by the data. To solve the model, we used the python implementation in the scipy library [21] of the Explicit Runge-Kutta method of order three [9].

4.1.2 True function information

Let us denote our true glucose function $G^t(t)$. The only information we have about it is a set of measurements $\{G_{t_i}\}_{i=1}^m$ where m is the number of measurements.

4.1.3 Error function

Let $\mathbf{t} = \{t_i\}_{i=1}^m$ be the measurement times and define the "sampling" functional as:

$$\begin{aligned} F_{\mathbf{t}} : C(\mathsf{R}) &\to \mathsf{R}^m \\ f &\mapsto (f(t_1),\, f(t_2), \ldots f(t_m)) \end{aligned} \tag{18}$$

$M_{\mathbf{t}}(f, g) : C(\mathsf{R}) \times C(\mathsf{R}) \to \mathsf{R}$ is our error measurement given by:

$$M_{\mathbf{t}}(f, g) = \text{mse}(F_{\mathbf{t}}(f), F_{\mathbf{t}}(g))) \tag{19}$$

where mse is the mean squared error function defined as:

$$\text{mse}(\mathbf{y}, \mathbf{y}') = \frac{1}{m} \sum_{i=1}^{m} (\mathbf{y}_i - \mathbf{y}'_i)^2 \tag{20}$$

with $\mathbf{y}$, $\mathbf{y}' \in \mathsf{R}^m$ for any m.

4.1.4 Box of boundaries and starting point

The following box of boundaries and starting points were compiled from the ranges shown in Table 1 combined with an iterative process of evaluating different ranges using two criteria: convergence and error measure.

Let $R \subseteq \mathsf{R}_+^9$ be a compact set of allowed values for $(G_o,\, I_o,\, \ldots I_{\text{decay}})$ such that $G_o = G^t(0)$ and the rest lie in the ranges specified in Table 1. Let $\mathbf{p}_o \in R$ be the starting point for optimization such that the parameter values are the common values presented in Table 1.

With these five requirements well defined, we chose a python implementation in the scipy library of the L-BFGS-B algorithm for bound-constrained optimization [12, 13] to minimize $M_{\mathbf{t}}(G(t), G^t(t))$. So,

$$(p_1, p_2, \ldots p_9) = \underset{\mathbf{p} \in R}{\text{argmin}}\, M_{\mathbf{t}}(G(t), G^t(t)) \tag{21}$$

From which we will obtain a list of parameters for each window.
The method L-BFGS-B is a limited memory optimization algorithm with constraints of the form $l < x < r$, generally used to solve big non-linear problems. The method computes an approximation of the Hessian matrix that can be expensive when the number of optimization variables is large. Although, that's not our case, we will see later that the computing time is a meaningful deal for our purposes and so approximating the Hessian matrix is still a plus.

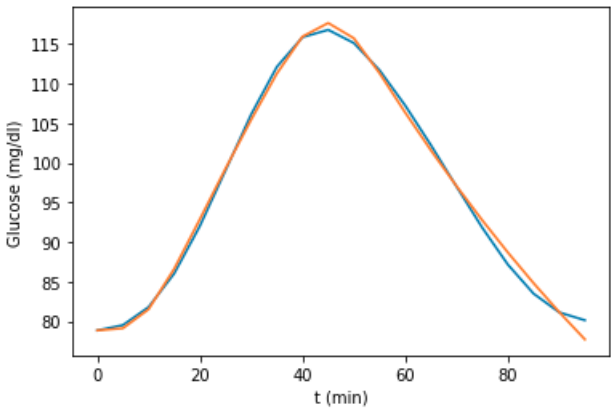

Figure 8: Human parameter fitting results for user h5mxumnh with a sample standard deviation of 1.4mg dL^{-1} and parameters $A_{\max} = 1.57\mathrm{mg}\,\mathrm{dL}^{-1}$, $t_{\max} = 32\,\mathrm{min}$, $a = 0$, $I_{\text{slope}} = 0.256\mathrm{dL}\,\mathrm{mg}^{-1}$, $I_{\text{decay}} = 0.0056\mathrm{min}^{-1}$, $I_{\text{scale}} = 0.000107\mathrm{L}\,\mathrm{pmol}^{-1}\,\mathrm{min}^{-1}$, $G_{\text{act}} = 104\mathrm{mg}\,\mathrm{dL}^{-1}$, $I_o = 0\mathrm{pmol}\,\mathrm{L}^{-1}$

4.2 Human Parameters Fitting

In the beginning of this chapter, we mentioned two suppositions which we will use to fit our model, the second one stated that there is a subset of parameters called the "human parameters" which are approximately constant in time for each person. In this first step we describe an algorithm to find them. By using the optimization algorithm we found the set of parameters which best approximates (21) for every window, this fitting was "successful" for around 85% of the windows in the sense that fit to the measured glucose window with a standard deviation lower than 2mg dL^{-1}. Figure 8 shows an example of a successful fit.

Note that the model (16) shows a term $A_{-1}(t)$ which depends on the absorption parameters of the previous meal. So, the most natural approach would be to fit each window in order and use the parameters found in the current window for the next one. This approach contains a subtle disadvantage: by doing this, the optimization process cannot be performed in parallel as every window needs to wait for the optimization process of the last window to be finished. The optimization process takes around 4 minutes per window in a personal computer with an Intel® Core i5-7600K processor and 8Gb RAM. Since there are test users which contains 247 windows as shown in Table 3, without parallel computing it would take around 16 hours which would take each iteration when trying to tune the starting point and the human ranges of the parameters. To solve this, we proposed the following parallelizable algorithm:

Algorithm 4.1. Given a set of glucose windows for an user:

1. Perform a first optimization in parallel supposing that the previous window contains no meal.
2. Next, perform the optimization using for A_{-1} the values $A_{\max}$, $t_{\max}$ and a of the previous window obtained in the last optimization.
3. At the end of each optimization we compute the new error using for A_{-1} with the values $A_{\max}$, $t_{\max}$ and a of the previous window obtained in the current optimization.
4. If this error converges (standard deviation $< 5\mathrm{mg}\,\mathrm{dL}^{-1}$), then it means that the absorption parameters are not changing significantly and the algorithm stops.

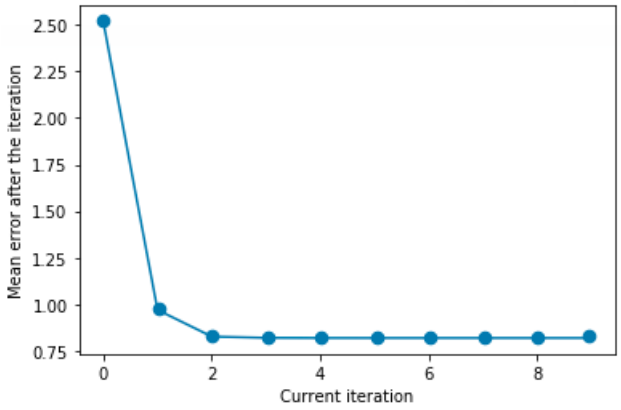

Figure 9: Optimization error for user h5mxumnh obtained after each iteration of the parallel Algorithm 4.1 showing that the error rapidly converges.

User Id	I_s $(\frac{\text{L}}{\text{pmol min}})$	I_{slope} $(\frac{\text{dL}}{\text{mg}})$	I_{decay} (min^{-1})	G_{act} $(\frac{\text{mg}}{\text{dL}})$
h5mxumnh	0.000208	0.342	0.0218	86.5
1xvbq4gd	0.000164	0.175	0.0143	88.9
6ev6bjpg	0.000061	0.262	0.0407	78.6
7bucr6u7	0.000132	0.160	0.0214	85.8
hj0jqplt	0.000125	0.173	0.0210	73.5
nah5br7n	0.000078	0.271	0.0123	80.9

Table 4: Summary of human parameters for some test users obtained after following the Algorithm 4.1 and choosing the mode in histograms like the ones shown in 10

Figure 9 shows that almost at the second iteration the algorithm already converges. So although its hard to prove analytically the convergence of this algorithm, for all test users it converged fast and besides, it is 10x faster that the non-parallelizable algorithm in a server with 28 cores and 16Gb RAM.

The result is a set of parameters $\{\mathbf{p}^i\}_{i=1}^N$ where N is the number of windows for the current test user. Then we can build a histogram for every parameter I_s, I_{slope}, G_{act} and I_{decay} and check its distribution, then, if the distribution is unimodal, we choose the mode of the params histograms as the fixed value. Figure 10 shows that all histograms are unimodal. Table 4 summarizes the resulting human parameters for each user.

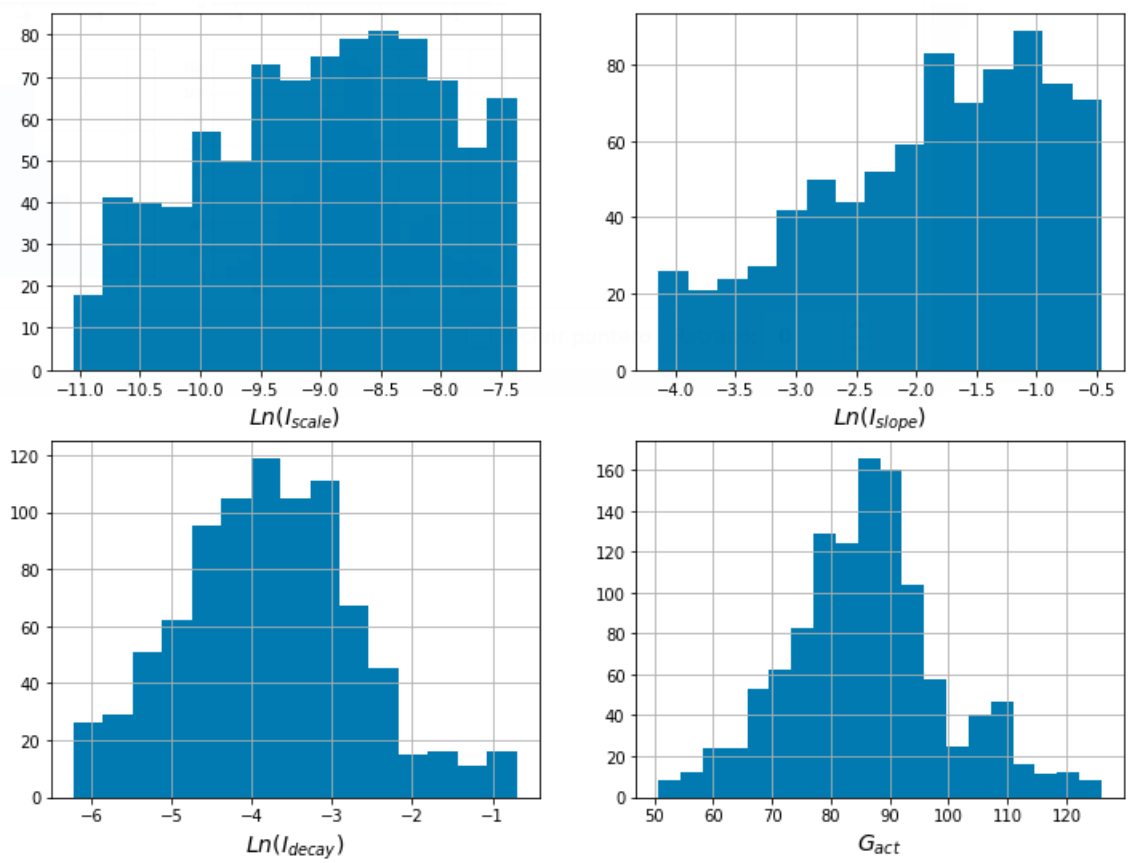

Figure 10: Logarithmic Histograms of I_{scale}, I_{slope}, I_{decay} and Histogram of G_{act} representing the distribution of the "human parameters" for user h5mxumnh, obtained by following the Algorithm 4.1

4.3 Meal parameters fitting

Now, with the human parameters found for every user, we can use again the optimization algorithm (following the parallel procedure again) but only for I_o, $A_{\max}$, $t_{\max}$ and a on each window. Like in the human parameters fitting, the results were very promising, Figure 11 shows the logarithmic distribution of the final errors, with 93% of the errors (mean squared error) falling behind $100\text{mg}^2\,\text{dL}^{-2}$. Figure 12 shows the level of fitting of an average window.

Therefore, the results clearly show that the model is able to describe the dynamics of the glucose homeostasis process and estimate a set of parameters which describe each person metabolism.

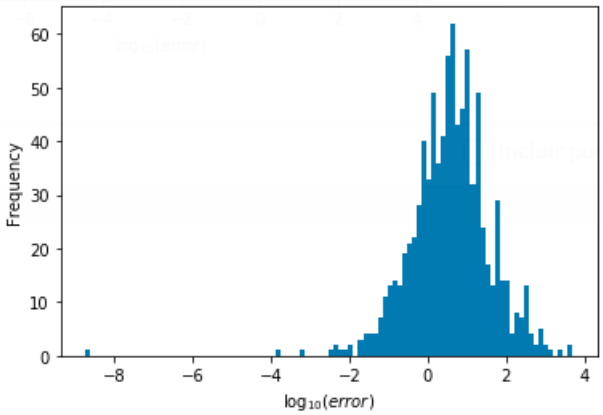

Figure 11: Optimization error for each windows of all users obtained in the meals parameter fitting represented by a logarithmic histogram, useful to visualize overall performance of the algorithm.

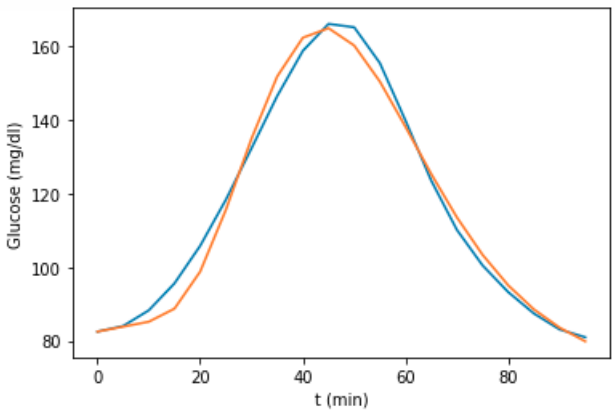

Figure 12: Meal parameter for user h5mxumnh fitting results with a sample standard deviation of $4.2\text{mg}\,\text{dL}^{-1}$ and parameters $A_{\max} = 1.77\text{mg}\,\text{dL}^{-1}$, $t_{\max} = 24\,\text{min}$, $a = 1.04$, $I_{\text{slope}} = 0.159\text{dL}\,\text{mg}^{-1}$, $I_{\text{decay}} = 0.0208\text{min}^{-1}$, $I_{\text{scale}} = 0.000158\text{L}\,\text{pmol}^{-1}\,\text{min}^{-1}$, $G_{\text{act}} = 86.5\text{mg}\,\text{dL}^{-1}$, $I_o = 0\text{pmol}\,\text{L}^{-1}$

5 Relation between nutritional value and absorption

Very few information is available about the relation between oral meal ingestion and plasma glucose absorption. This is probably due to the complexity of the digestion and chemical processes involved, and the number of random factors and hard-to-measure variables like the eating pace, the order of ingested nutrients, the position of the body and the nutritional value of the meal itself. For this reason, most of the published papers related to glucose modeling [8, 7] refer to clinical studies where intravenous glucose is injected or patients are provided with very standardized and easy to digest meals.

As described before, for this project, the test users provided an approximate nutritional value of the meal and the ingestion time. We will propose a black box statistical and deep learning model which will try to find the relation between the nutritional value (Calories, Carbs, Proteins, Fats, Fibers) with the absorption parameters ($A_{\max}$, $t_{\max}$, a)

5.1 Exploratory data analysis on the meal data

Let's first explore the data in order to find information and relations prior to the model construction. The scatter plot in Figure 13 contains the histograms with data from all users of each variable (in the diagonal) and the scatter plot for each couple of nutritional value variables.

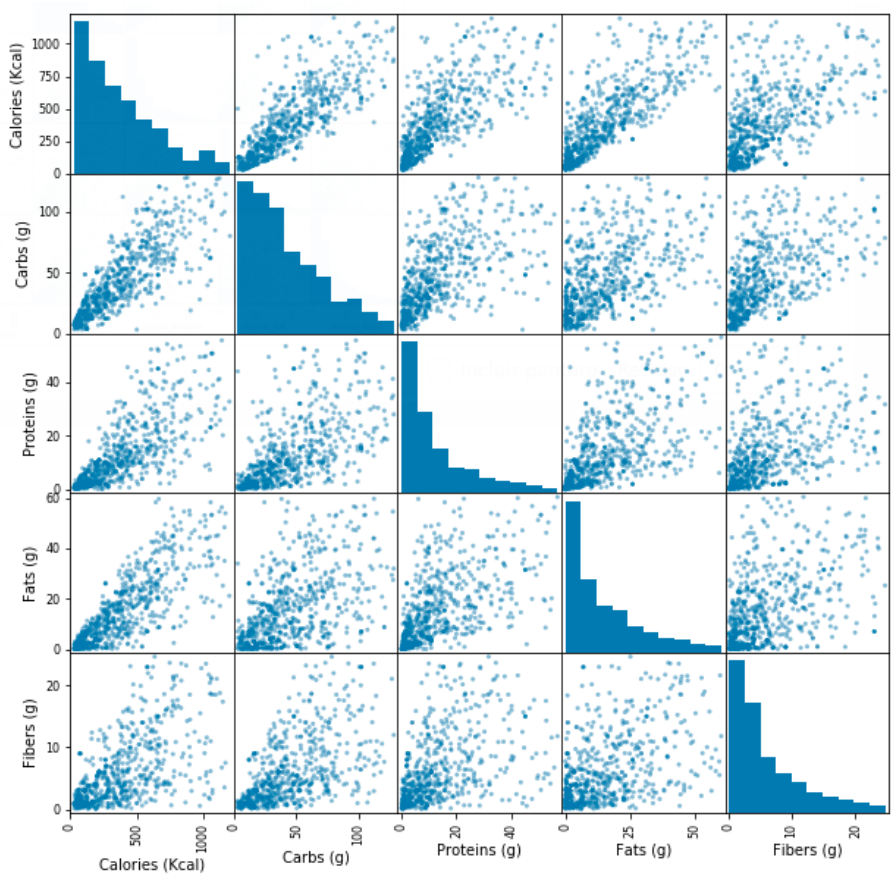

Figure 13: Scatter plot of the nutritional value for each pair of variables with the histogram of each one in the diagonal. Here, it was used the data for all users.

Some points to notice in this chart:

- The data for all users is composed mostly of light meals of less than 500kcal each, which is the recommended value for a breakfast or snacks, although the latter should not surpass 200kcal [23].
- There is a clear increasing trend between every couple of variables. This is probably due to the fact that bigger meals are most likely to contain more of every nutritional component and vice versa.
- The most disperse plots are the ones related to fibers. This is probably due to the fact that digestive enzymes cannot break down the fibers, so they do not have a direct contribution to calories or carbohydrates, even though fibers are carbohydrates.

Motivated by the increasing trend of the scatter plots, lets examine the following well known relation [22]: One gram of carbohydrates, protein and fats contain 4, 4 and 9 calories respectively. Figure 14 tests this relation using the data for all users, obtaining a slope of 0.996 and an $R^2 = 0.959$. It aligns with the fact that the user acquired the nutritional content using an app that contains nutritional value tables that generally use this method to compute the number of calories in a meal.

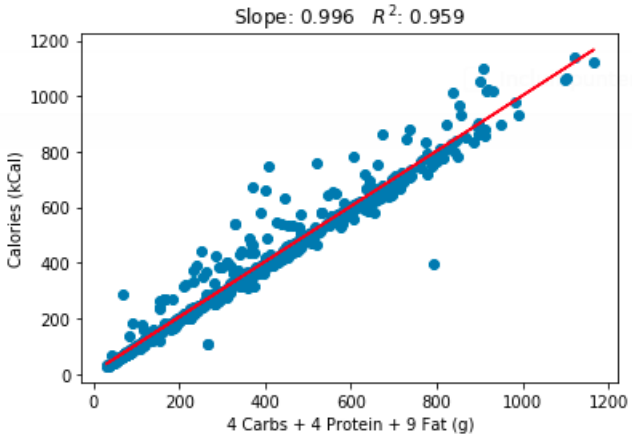

Figure 14: Relation between Calories of logged meals and the amount of Carbs, Proteins and Fats. Here, it was used the data for all users.

From the nutritional elements, the main source of glucose are the carbohydrates. Proteins and fats have other functions like producing hormones, repairing tissues and absorption processes, then intuitively, there should be an increasing trend between carbs content in a meal and the size of the glucose peak associated to that meal. In fact, the data provided contains no relation between the size of the glucose peak, see Figure 15 and although a direct relation is not expected, as there might be slow glucose processing meals due to fats leading to low peaks, it makes no sense to have glucose peaks close to 0 after a meal ingestion of 100-200 grams of carbohydrates. This fact leaves us with some possibilities:

1. The nutritional value is very noisy and was not correctly logged.

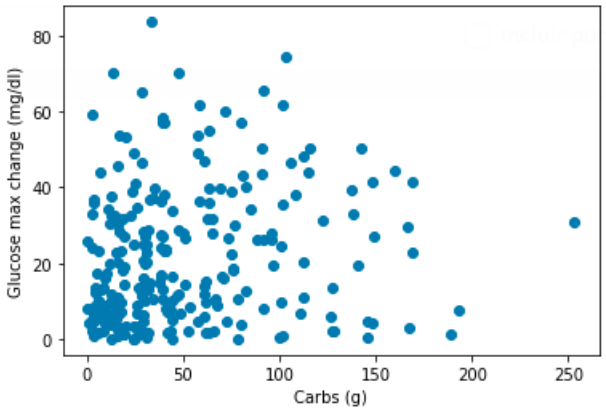

Figure 15: Glucose max change given by $\max_t G(t) - G(0)$ vs logged carbs in the meal for user h5mxumnh

2. The meal was registered hours after the real ingestion which would lead to a wrong meal to window association.

3. There are meals which produce an extremely flat glucose graph

4. All of the above.

Another check that can be performed is that, by mass conservation, the total glucose absorption must be around 74% of the number of ingested carbs as some part of it is retained by the intestine and liver [4]. So, as our model had a very good performance with absorption parameters inside the normal ranges, the following should hold:

$$\text{carbs} \sim \frac{V_G W}{0.74} \int_0^\infty A(t)dt \tag{22}$$

V_G is the volume of distribution defined as the theoretical volume of blood through which glucose is distributed, divided by the person's weight. In average, for adults $V_G = 1.8\text{dL}\,\text{kg}^{-1}$ [1]. W is the weight of the person in kilograms. Figure 16 shows that this relation is not held by the data. So, like the model performance was very good with parameters within human ranges and due to the result in Figure 15 we can again suspect about the meal data integrity.

5.2 Carbs correction

To deal with the fact that the meal data is very likely to have been wrongly measured, we propose a model for carbs that combined with the measurements will give a better estimation of the ingested carbs. It is desirable that the model holds the following conditions:

1. It is general enough so that the results are as unbiased as possible to our model.

2. All of its parameters and variables are known or can be computed with the provided data.

3. The model uncertainty can be computed because it is needed by the data assimilation method that we will describe later.

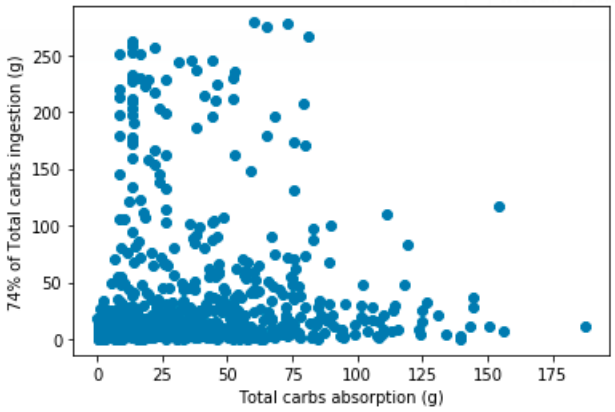

Figure 16: Total carbs absorption vs the expected carbs absorption defined in (22) for all users.

With a model like this, we can use a data assimilation method called "Analysis Scheme" to make a combination between the proposed model. A very general model for glucose can be written as a simple mass conservation equation of the form:

$$\frac{dG}{dt} = A(t) - U(G, I) \tag{23}$$

Where A is the exogenous glucose absorption and U is the insulin dependent glucose utilization. Now by integrating both sides of the equation from 0 to the last timestamp of the current window t^*:

$$\Delta G = \int_0^{t^*} A(t)dt - \int_0^{t^*} U(G, I)dt \tag{24}$$

We integrate only until t^* to hold the second condition of the model, after t^* we do not have information. Then by multiplying and dividing the first term of (24) by $\int_0^\infty A(t)dt$, setting $a^* = \int_0^{t^*} A(t)dt / \int_0^\infty A(t)dt$ the fraction of glucose absorbed up to time t^* and using equation 22 we get

$$\Delta G = \frac{0.74a^*}{V_G W}\text{carbs} - \int_0^{t^*} U(G, I)dt \tag{25}$$

Then we will set $U(G, I) = I_s G(t)I(t)$ not because it is the term we used in our model but because it has been validated [8] and used by many a authors of glucose minimal models. For $G(t)$ we can use the measured glucose to avoid model bias but as we do not have insulin data, we do not have any option than using the insulin model:

$$\begin{aligned} \frac{dI}{dt} &= S(G(t)) - I_{\text{decay}} I(t) \\ I(0) &= I_o \end{aligned} \tag{26}$$

Finally, by solving for carbs in equation (25) we get:

$$\text{carbs} = \frac{V_G W}{0.74a^*}\left(\Delta G + \int_0^{t^*} U(G, I)dt\right) \tag{27}$$

We have to mention that there is clearly some undesired bias to our model in equation (27). First, the a^* factor is biased to the absorption term, but as it is just the percentage of the integral until time t^*, it does not have a big bias to the functional shape of $A(t)$. Second, the $I(t)$ is probably strongly biased to our model but the one which we cannot do anything about as no insulin measurements were provided. Third, the parameters I_{decay}, I_{slope}, I_s were optimized to fit our model but as they were found to be around normal people values the bias to the model is also nothing to worry about.

We now propose a scheme to combine the results of the equation (27) and the nutritional value logged by the user, to obtain a better estimation of the actual carbohydrates ingested.

5.2.1 Analysis scheme

Data Assimilation is a field of mathematics which studies methods to combine models with data. There can be lots of objectives when combining models with data like parameter finding, error analysis, analysis scheme, amongst others. Analysis scheme is a method for combining some measurement of a system state with the result given by a proposed model so that the estimation of the true state of the system is improved.

Given a study system, let y^t be the true state, y^f the model estimate, y^m a measurement of y^t, ϵ^f and ϵ^m the errors of the model and measurement respectively, namely:

$$y^f = y^t + \epsilon^f \tag{28}$$
$$y^m = y^t + \epsilon^m \tag{29}$$

Lets suppose that the measurements and models are unbiased, so $\mathsf{E}[\epsilon^f] = 0$ and $\mathsf{E}[\epsilon^m] = 0$. In general, there is no reason why the errors of the model and measurement to be correlated, so lets suppose that its covariance is $\text{Cov}(\epsilon^m, \epsilon^f) = 0$.

What we want is to find an unbiased linear estimator y^a:

$$y^a = y^t + \epsilon^a = \alpha_1 y^f + \alpha_2 y^m \tag{30}$$

such that:

$$\text{var}(y^a) = \inf_{\substack{\alpha_1, \alpha_2 \in \mathsf{R} \\ \mathsf{E}[y^a] = y^t}} \text{var}(\alpha_1 y^f + \alpha_2 y^m) \tag{31}$$

There, var is the variance of a random variable. After solving (31) the best estimation y^a and it is uncertainty σ_a^2 are given by [6]:

$$y^a = \frac{\sigma_m^2}{\sigma_f^2 + \sigma_m^2} y^f + \frac{\sigma_f^2}{\sigma_f^2 + \sigma_m^2} y^m \tag{32}$$
$$\sigma_a^2 = \frac{\sigma_f^2 \sigma_m^2}{\sigma_f^2 + \sigma_m^2} \tag{33}$$

This is a really intuitive result as the weights for each estimator are proportional to the variance of its converse.

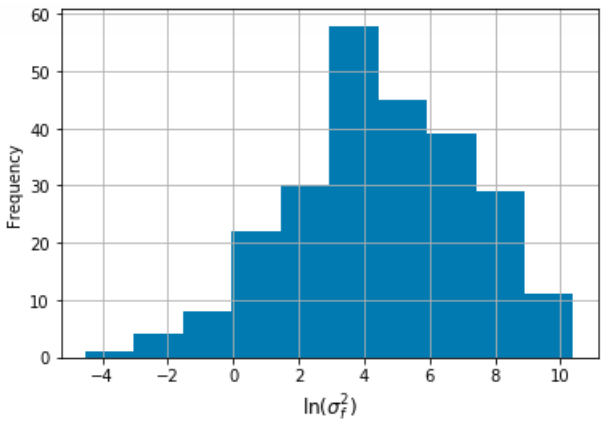

Figure 17: Logarithmic plot of the sample variance σ_f^2 for which we took 500 random samples of the human parameters and used the correction model proposed in (27) for user h5mxumnh.

In order to use the previous results to improve the carbs ingestion estimation, let c^a be the minimum variance estimation, c^f = carbs given by equation (27), c^m the logged carbohydrates of the meal and ϵ^a, ϵ^f and ϵ^m their corresponding uncertainties. Then from (32), (33):

$$c^a = \frac{\sigma_m^2}{\sigma_f^2 + \sigma_m^2} c^f + \frac{\sigma_f^2}{\sigma_f^2 + \sigma_m^2} c^m \tag{34}$$

$$\sigma_a^2 = \frac{\sigma_f^2 \sigma_m^2}{\sigma_f^2 + \sigma_m^2} \tag{35}$$

With the data provided by the test users it is not feasible to directly estimate the uncertainty of the meal data as its errors sources are varied. So we propose an apriori pessimistic estimation of the uncertainty as $\sigma_m = 20g$ which corresponds to more or less one and a half slices of bread.

To estimate the model error σ_f we fitted first a parametric distribution to the parameters in the insulin model derived from the histograms shown in Figure 10. For one of the test users the distributions were:

- $I_s \sim \text{LogNormal}(-8.75,\ 0.947)$
- $I_{\text{slope}} \sim \text{LogNormal}(-1.84,\ 0.952)$
- $I_{\text{decay}} \sim \text{LogNormal}(-3.87,\ 1.11)$
- $G_{\text{act}} \sim \text{Normal}(86.5,\ 13.3)$

Using the fitted distributions, we performed 500 simulations per glucose window and computed the variance per window of the estimated carbohydrates. The results are shown in Figure 17. We could, of course, assign uncertainty to all of the components in equation (27) like ΔG, V_G, W and a^* but the human parameters may have uncertainties of one full magnitude order, so we decided to ignore them.

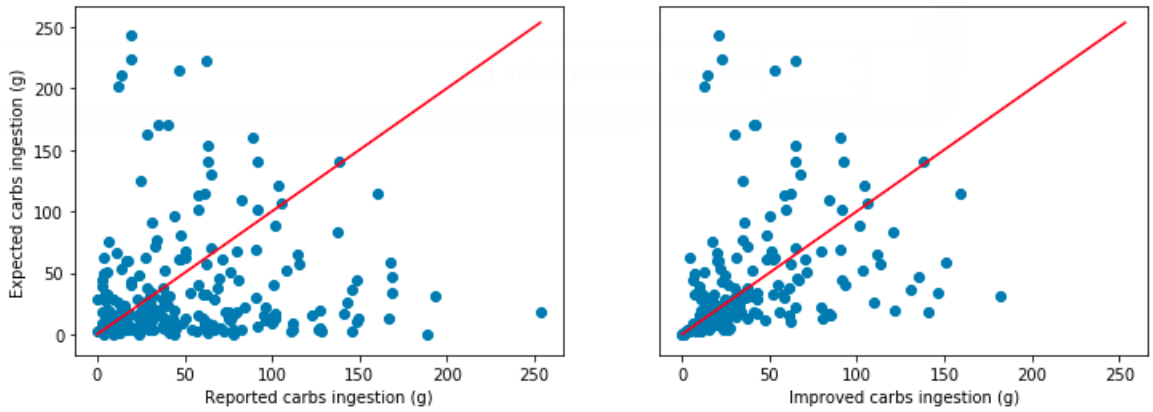

Figure 18: (a) Carbs ingestion before improvement, represented by the carbs logged by user h5mxumnh (b) Carbs ingestion after improvement when using equation (34).

There is a wide range of variances depending on the window, then for the ones with a large value of σ_f, the carbs correction (27) will not be significant as $\frac{\sigma_f^2}{\sigma_f^2+\sigma_m^2} \gg \frac{\sigma_m^2}{\sigma_f^2+\sigma_m^2}$. Figure 18 shows the carbohydrates ingestion before and after the data assimilation improvement, we see that, as expected, some of the dots get closer to the $y = x$ line (drawn in red) but the ones with large variance remain in the same position.

6 Prediction of absorption parameters

The exploratory data analyses in the previous chapters, leave us with deep questions regarding the impact of the data integrity on the prediction model. One might be optimistic and say that there is a deep pattern in the data that cannot be seen with shallow charts, but that maybe complex black box models can fit to those dynamics and make decent predictions. In this chapter we propose a method for predicting the absorption parameters of each glucose window by using the meal nutritional value.

First we need to introduce the conceptual framework of learning sets and regressors.

Definition 6.1. Let $\{\mathcal{X}_i\}_{i=1}^n$, $\mathcal{Y}$ be a collection of sets, usually called *inputs* and *output spaces* respectively, where $\mathcal{X}_i$, $\mathcal{Y}$ are usually a subset of R or a finite set of labels. Values $\mathbf{x} = (x_1, x_2, \dots x_n) \in \mathcal{X}_1 \times \mathcal{X}_2 \times \dots \mathcal{X}_n$ and $y \in \mathcal{Y}$ are usually called *input vectors* and *output values* respectively. A *learning set* $\mathcal{L}$ is a finite set of pairs $(\mathbf{x}_i, y_i)$ where x_i is an input vector, y_i an output value and $\mathcal{X}_i$ and $\mathcal{Y}$ are generally a subset of R or a finite set of labels. Let $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \times \dots \mathcal{X}_n$, $\mathcal{Y} \subseteq \mathsf{R}$ be the input and output spaces respectively. A *regressor* is a function $\phi_{\mathcal{L}} : \mathcal{X} \to \mathcal{Y}$ built with a learning set $\mathcal{L}$.

To build the sets of input spaces for our problem we thought about the features that could be measured prior to the meal ingestion and that impact in some way the absorption parameters. The following are the proposed input spaces:

- $\mathcal{X}_1$, $\mathcal{X}_2$, $\mathcal{X}_3$, $\mathcal{X}_4$, $\mathcal{X}_5 = \mathsf{R}^+$ will represent the space of values for nutritional value variables: calories, carbs, proteins, fats and fibers. Although we know that there is a relation between

carbohydrates, proteins, fats and calories shown in Figure 14, when training models it is better to include features relations to simplify the optimization process.

- $\mathcal{X}_6 = \{0, 1, 2 \dots 23\}$ will represent the possible hour of the day. It might be an important categorical variable due to the circadian cycles.
- $\mathcal{X}_7 = \{0, 1, 2 \dots 6\}$ will represent the day of the week. We considered this measure because of the feeding routine of people. For example, there is a higher chance that people eats less healthy meal on weekends than on weekdays.
- $\mathcal{X}_8 = \{\text{breakfast, lunch, dinner}\}$ represents the moment of the day where the meal was ingested. It is defined as follows: From 4am to 12pm, the meal is a breakfast, from 12pm to 6pm it is a lunch and the rest are dinners. This could bring meaningful information because there is maybe a different reaction of the body to meal ingestion when breaking fast or when sleeping.

We are trying to predict the parameters of the absortpion function in (10) $A_{\max}$, $t_{\max}$ and a the output spaces $\mathcal{Y}_1$, $\mathcal{Y}_2$, $\mathcal{Y}_3$ are all equal to R^+ representing each of the absorption parameters. The following workflow describes the process we used for training and validating an automated learning model which predicts the absorption parameters:

1. Let $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \dots \mathcal{X}_8$ be the input space. With the data provided, for every window of the user i we can obtain an input vector $\mathbf{x} \in \mathcal{X}$ and built a set $\mathbf{X}_i \subset \mathcal{X}$.
2. For every window of the user i take the values $A_{\max}$, $t_{\max}$, a obtained by fitting in section 4.3 and build three sets $\mathbf{y}_{i1}$, $\mathbf{y}_{i2}$, $\mathbf{y}_{i3}$.
3. Build the full learning set for each output space $\mathcal{L}_j^{\text{full}} = \bigcup_i \mathbf{X}_i \times \mathbf{y}_{ij}$, $j = 1, 2, 3$
4. Partition each of the sets $\mathcal{L}_j^{\text{full}}$ into two sets $\mathcal{L}_j$, $\mathcal{T}_j$ called learning and testing sets respectively, $j = 1, 2, 3$. This splitting is usually performed by randomly picking elements from $\mathcal{L}_i^{\text{full}}$ until $|\mathcal{L}_i| = 0.7|\mathcal{L}_i^{\text{full}}|$.
5. Choose and train three regressors $\phi_{\mathcal{L}_1}$, $\phi_{\mathcal{L}_2}$, $\phi_{\mathcal{L}_3}$ that try to approximate y with $\phi_{\mathcal{L}_j}(\mathbf{x})$ for all $(\mathbf{x}, y) \in \mathcal{L}_j$, $j = 1, 2, 3$
6. Let $\mathbf{y}_i^{\text{test}} = \{y \,|\, (\mathbf{x}, y) \in \mathcal{T}_i\}$ be the set of testing outputs and $\text{mse}(\phi_{\mathcal{L}_i})$ the model error defined as:
$$\text{mse}(\phi_{\mathcal{L}_i}) = \frac{1}{N} \sum_{(\mathbf{x},y)\in\mathcal{T}_i} (\phi_{\mathcal{L}_i}(\mathbf{x}) - y)^2 \tag{36}$$
Evaluate the performance of each $\phi_{\mathcal{L}_i}$ using the determination coefficient R^2 defined as:
$$R_i^2 = 1 - \frac{\text{mse}(\phi_{\mathcal{L}_i})}{\text{var}(\mathbf{y}_i^{\text{test}})} \tag{37}$$
where var is the sample variance function. The coefficient of determination R^2 can be interpreted as the percentage of the variance of the outputs y that can be explained by the model. Normally it falls in the range $[0, 1]$ where 1 means that the model perfectly predicts the outputs and 0 means that the model is as good predicting as the mean of the data,

although it is not constrained to the range $[0, 1]$ as models can make an arbitrarily wrong prediction. As an additional note, the square of the Pearson's correlation coefficient r^2 equals the coefficient of determination when the model $\Phi_{\mathcal{L}}$ is a linear regression.

6.1 Random Forest Regressor

Nowadays, there is an infinite sea of automated learning models with outstanding performances for very complex problems like climate prediction, artificial vision and natural language processing. Tree-based models, together with deep neural networks, are two of the currently most used models because they:

- can learn patterns in very non-linear regions,
- can handle categorical and numerical data,
- usually can handle missing data,
- are robust to data outliers.

For the absorption parameters prediction we decided to use the Random Forest Regressor which is an improvement of the classical Tree Regressors in the sense that yields smaller variances in the outputs. The following sections present a minimal theoretical background of tree based models so that we can precisely describe our specific application of the Random Forest Regressor.

6.1.1 Regression Trees

In our approach of tree-based models we use rooted binary trees with the following conventions:

1. 0 will be the root node.
2. If e is a node's name $e0$, $e1$ will be the left and right children of e respectively.
3. Let $L(T)$ be the set of nodes without children in T, often called terminal nodes or leaves.
4. Let e, $s \in T$. $e \rightsquigarrow s$ is the set of nodes in the trajectory from e to s, both included.

Figure 20 shows a representation of a regression tree $T = \{0, 00, 01, 010, 011\}$ where $L(T) = \{00, 010, 011\}$ and $0 \rightsquigarrow 010 = \{0, 01, 010\}$.

Definition 6.2. A regression tree $T = \{0, 00, 01, \dots\}$ is a rooted directed binary graph where:

1. The root node 0 has associated the set $\mathcal{X}$.
2. Every pair of children $e0$, $e1$ of a node e have associated a set X_{e0}, $X_{e1} \subseteq \mathcal{X}$ such that $X_{e0} \uplus X_{e1} = \mathcal{X}$, where $\uplus$ is the disjoint set union.
3. Every $l \in L(T)$ has an associated a value $s_l \in \mathcal{Y}$.

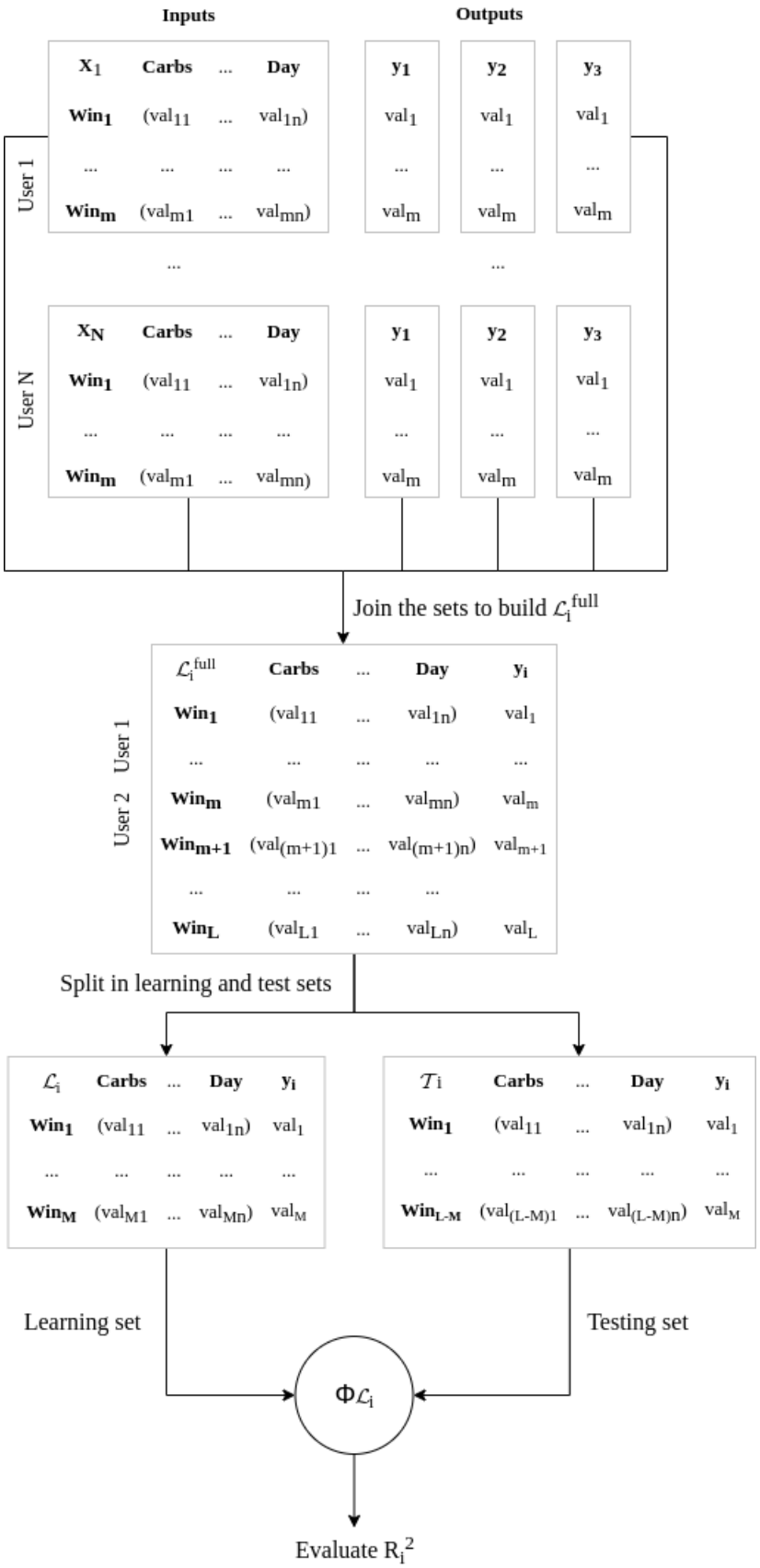

Figure 19: Workflow for training and evaluating a regressor with a given set of inputs and outputs for every user, where $M = |\mathcal{L}_i|$, $L = |\mathcal{L}_i^{\text{full}}|$

Proposition 6.1. Let T be a regression tree and $L(T)$ the set of all terminal nodes of T. If $l \in L(T)$ and

$$S_l = \bigcap_{k \in 0 \rightsquigarrow l} X_k \tag{38}$$

then:

$$\mathcal{X} = \bigcup_{e \in L(T)}^{\bullet} S_e \tag{39}$$

This means that the terminal nodes of a tree induce a partition of the set $\mathcal{X}$.

Proof. Lets show the proposition by induction on the number of nodes $2n+1$ (as all binary trees have an odd number of nodes).

1. For the base case $n = 0$ the proof is trivial because $T = \{0\}$ contains only the root node and $X_0 = \mathcal{X}$.

2. Lets suppose that for all trees with $2n+1$ nodes, the proposition holds. Let T be a regression tree with $2(n+1)+1$ nodes. As the tree is finite, there must exist a node e with two terminal children $e0, e1 \in L(T)$. Let T^* be the regression tree with the nodes $e0$, $e1$ removed, then by using the inductive hypothesis $\mathcal{X} = \cup_{t \in L(T^*)} S_t$ we get:

$$\begin{aligned}
\bigcup_{t \in L(T)} S_t &= \left(\bigcup_{\substack{t \in L(T) \\ t \notin \{e0, e1\}}} S_t \right) \cup S_{e0} \cup S_{e1} \\
&= \left(\bigcup_{\substack{t \in L(T) \\ t \notin \{e0, e1\}}} S_t \right) \cup (S_e \cap X_{e0}) \cup (S_e \cap X_{e1}) \\
&= \left(\bigcup_{\substack{t \in L(T) \\ t \notin \{e0, e1\}}} S_t \right) \cup (S_e \cap (X_{e0} \cup X_{e1})) \\
&= \left(\bigcup_{\substack{t \in L(T) \\ t \notin \{e0, e1\}}} S_t \right) \cup S_e = \bigcup_{t \in L(T^*)}^{\bullet} S_t = \mathcal{X}
\end{aligned} \tag{40}$$

3. It is easy to see that $S_t \cap S_e = \textsf{?}$ for all $t, e \in L(T)$ because:

   (a) $S_t \cap S_e = \textsf{?}$ for all $t, e \in L(T) - \{e0,\, e1\}$ by the inductive hypothesis.

   (b) $S_{e0} \cap S_{e1} = (S_e \cap X_{e0}) \cap (S_e \cap X_{e1}) = \textsf{?}$

   (c) Let $t \in L(T) - \{e0,\, e1\}$, then $S_{e0} \cap S_t = X_{e0} \cap S_e \cap S_t = \textsf{?}$. The last equality holds by the inductive hypothesis and of course it holds also for S_{e1}

□

It follows from (39) that we can build a regressor $\phi : \mathcal{X} \to \mathcal{Y}$ induced by T given by

$$\phi(\mathbf{x}) = \sum_{l \in L(T)} s_l \mathbb{1}_{S_l}(\mathbf{x}) \tag{41}$$

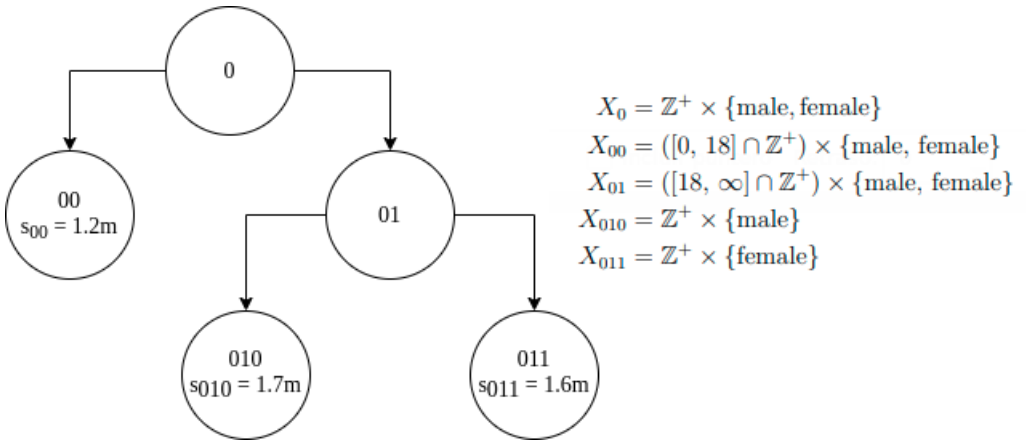

Figure 20: Example of regression which predicts a person height with x_1 being the age of the person and x_2 the biological sex

Where $\mathbb{1}_{S_l}(\mathbf{x})$ is the indicator function on the set S_l defined in (38). In other words, a regression tree induces a partition of a set $\mathcal{X}$ and assigns a value in R to each subset, the latter is called the *predicted value*.

Usually, the regressor implementation does not define $\phi(x)$ as (41) but instead stores the sets X_e assigned to each node e and uses the following algorithm to make predictions.

Algorithm 6.1. In order to predict a value y of a regressor tree for an input vector $\mathbf{x} \in \mathcal{X}$:

1. Start in the root node.
2. In the current node e get the partition $\{X_{e0},\ X_{e1}\}$ of the children.
3. If $x \in X_{e0}$ the current node becomes $e0$, otherwise $e1$.
4. Repeat step 2 until the node is a terminal one.
5. The predicted value $\phi(\mathbf{x})$ equals the label s_l of the terminal node.

Figure 20 shows a representation of a regression tree which predicts the height of a person, there, $\mathcal{X} = \mathsf{Z}^+ \times \{\text{male, female}\}$, $\mathcal{Y} = \mathsf{R}$ and $\mathbf{x} = (x_1, x_2)$ are the age and sex of the person respectively.

Now that we have defined regression trees and knowing that they induce a regressor $\phi(x)$, the next step is: Given a learning set $\mathcal{L}$, build a regressor $\phi_{\mathcal{L}}(x)$ which best fits the data $(\mathbf{x}, y) \in \mathcal{L}$. There are many proposed algorithms to build a regression tree from data. We will follow the greedy framework introduced in [18]. It has been proven that although greedy algorithms hardly converge to the optimal solution $\phi^*_{\mathcal{L}}$, they are computationally efficient and their performance is not very different to the lookahead approach [18]. The idea behind the algorithm is to start from a tree with a root node and add children nodes by successively partitioning the learning set $\mathcal{L}$ so that, at the terminal nodes, the remaining part of $\mathcal{L}$ contains outputs y with very similar values. We will start with some required definitions so that we can later describe the optimization algorithm.

Definition 6.3. Let e be a node and s its parent. The remaining part of $\mathcal{L}$ in node e is defined as:

$$\mathcal{L}_e = \hat{\mathcal{L}}(X_e, \mathcal{L}_s),\ \mathcal{L}_0 = \mathcal{L} \tag{42}$$

where $\hat{\mathcal{L}}$ is the function which restricts $H \subseteq \mathcal{L}$ to $A \subseteq \mathcal{X}$ and is defined as:

$$\hat{\mathcal{L}}(A, H) := \{(\mathbf{x}, y) \in H \,|\, \mathbf{x} \in A\}\ \ H \subseteq \mathcal{L},\ A \subseteq \mathcal{X} \tag{43}$$

Definition 6.4. Let $\mathcal{X}$, $\mathcal{Y}$ be input and output spaces and $\mathcal{L}$ a learning set on those spaces. The "entropy" of node e restricted to $A \in \mathcal{X}$ is given by:

$$i(e, A) = \frac{1}{|\hat{\mathcal{L}}(A, \mathcal{L}_e)|} \sum_{(\mathbf{x},y)\in\hat{\mathcal{L}}(A,\mathcal{L}_e)} (y - \bar{y})^2 \tag{44}$$

Where $\bar{y}$ is the sample mean of the outputs in $\hat{\mathcal{L}}(A, \mathcal{L}_e)$. For every node $e \in T$, $i(e) = i(e, X_e)$ is called the entropy in node e. Also, the change in entropy at node e when partitioning $\mathcal{X}$ into $\{A, A^c\}$ is defined as:

$$\Delta i(e, A) = i(e) - \frac{|\hat{\mathcal{L}}(A, \mathcal{L}_e)|}{|\mathcal{L}_e|} i(e, A) - \frac{|\hat{\mathcal{L}}(A^c, \mathcal{L}_e)|}{|\mathcal{L}_e|} i(e, A^c) \tag{45}$$

The name entropy comes from the fact that it is measuring the level of dispersion (or disorder) of a set in a node. We will see in the next proposition that as long as we keep partitioning the learning set $\mathcal{L}$, the entropy is reduced ($\Delta i(e, A) \geq 0$). This adds sense to the name "entropy" because the entropy in physics is an extensive property.

Proposition 6.2. $\Delta i(t, A) \geq 0$ for all $A \in \mathcal{P}(\mathcal{X})$

Proof. First of all, it is clear from the definition that

$$\hat{\mathcal{L}}(A, \mathcal{L}_e) \cup \hat{\mathcal{L}}(A^c, \mathcal{L}_e) = \mathcal{L}_e \Rightarrow\ \ |\hat{\mathcal{L}}(A, \mathcal{L}_e)| + |\hat{\mathcal{L}}(A^c, \mathcal{L}_e)| = |\mathcal{L}_e| \tag{46}$$

Let $\bar{y}$, $\bar{y}_A$, $\bar{y}_{A^c}$ be the sample mean of the y over $\mathcal{L}_e$, $\hat{\mathcal{L}}(A, \mathcal{L}_e)$ and $\hat{\mathcal{L}}(A^c, \mathcal{L}_e)$ respectively, then

$$\begin{aligned} \sum_{(\mathbf{x},y)\in\mathcal{L}_e} (y - \bar{y})^2 &= \sum_{(\mathbf{x},y)\in\hat{\mathcal{L}}(A,\mathcal{L}_e)} (y - \bar{y})^2 + \sum_{(\mathbf{x},y)\in\hat{\mathcal{L}}(A^c,\mathcal{L}_e)} (y - \bar{y})^2 \\ &\geq \sum_{(\mathbf{x},y)\in\hat{\mathcal{L}}(A,\mathcal{L}_e)} (y - \bar{y_A})^2 + \sum_{(\mathbf{x},y)\in\hat{\mathcal{L}}(A^c,\mathcal{L}_e)} (y - \bar{y_{A^c}})^2 \end{aligned} \tag{47}$$

The last inequality holds because $\bar{y}_A$, $\bar{y}_{A^c}$ are the estimators which minimize the sample variance. Finally by using (46) and (44),

$$\begin{aligned} &|\mathcal{L}_e| i(e) \geq |\hat{\mathcal{L}}(A, \mathcal{L}_e)| i(t, A) + |\hat{\mathcal{L}}(A^c, \mathcal{L}_e)| i(t, A^c) \\ &\Delta i(e, A) \geq 0 \end{aligned} \tag{48}$$

□

We will introduce now the concept of *stopping condition* for the algorithm. It will define the set of nodes that remain as terminal nodes.

Definition 6.5. Let T be a regression tree and $L(T)$ the set of all terminal nodes in T. A stopping condition C is a function $C : L(T) \to \{0, 1\}$. When $C(t) = 1$ we will say that the node remains as a terminal node. Otherwise, if $C(t) = 0$, the tree can grow from node t.

With these previous definitions, we are ready to describe one of the algorithms to build a regression tree.

Algorithm 6.2. Let $\mathbf{C} = \{C_i\}_i$ a set of stopping conditions and $\mathcal{L}$ a learning set. A regression tree built with $\mathcal{L}$ and restricted to the stopping conditions C_i can be built as follows:

1. Start with a tree which contains only the root node. The current node is the root node.

2. If for current node e, there exists i such that $C_i(e) = 1$ stop here for this node and the value s_e of this node will be:
$$s_e = \frac{1}{\mathcal{L}_e} \sum_{(\mathbf{x},y)\in\mathcal{L}_e} y. \tag{49}$$

3. In the current node e choose one set X_{e0} such that maximizes the change in entropy, namely,
$$X_{e0} \in \arg\max_{A\in\mathcal{P}(\mathcal{X})} \Delta i(e, A) \tag{50}$$
where $\mathcal{P}(\mathcal{X})$ is the set of partitions of $\mathcal{X}$.

4. If $\Delta i(e, X_{e0}) = 0$, stop here for this node compute s_e as step 2, otherwise add two new nodes $e0$ and $e1$ to the tree and associate the sets X_{e0} and X_{e0}^c to them.

5. Repeat from 2 to all new nodes.

It is important to mention that this algorithm will stop at some point because $\mathcal{L}$ is a finite set and in the worst scenario the splitting will end with $|\mathcal{L}_l| = 1$ for all terminal nodes.

Definition 6.6. A *Greedy regression tree* is a regression tree built with the Algorithm 6.2

At this point we have defined a way to build a regression tree with a learning set $\mathcal{L}$ and restricted to some conditions C_i. We are thus ready to point out the most important theorem of regression trees which states that by successively growing the tree, the error prediction of the induced regressor over the learning set $\mathcal{L}$ is reduced. Its proof is beyond the scope of this work but can be found in [18].

Theorem 6.1. [16] *Let T be a regression tree and ϕ it is induced regressor. Let l be a terminal node such that $|\mathcal{L}_l| > 1$ and T^* a new regression tree produced by a non-trivial splitting of the node l with its induced regressor ϕ^*. Then,*
$$mse(\{(\phi^*(\boldsymbol{x}), y) \,|\, (\boldsymbol{x}, y) \in \mathcal{L}\}) \leq mse(\{(\phi(\boldsymbol{x}), y) \,|\, (\boldsymbol{x}, y) \in \mathcal{L}\}) \tag{51}$$

The definition of *Greedy Random Forests* is still a little general for a software implementation because the way of computing $\arg\max_{A\in\mathcal{P}(\mathcal{X})} \Delta i(e, A)$ and the stopping conditions C_i is not specified. In this framework, taking into account that $\mathcal{P}(\mathcal{X})$ is generally very large, we will restrict

the domain for finding the arg max from $Q(\mathcal{X}) \subseteq \mathcal{P}(\mathcal{X})$ containing only the subsets formed by semi-infinite intervals for ordinal spaces or $\mathcal{P}(\mathcal{X}_i)$ for categorical spaces. In other words:

$$\begin{aligned} Q(X) &= \bigcup_i Q(\mathcal{X}_i) \\ Q(\mathcal{X}_i) &= \begin{cases} \{\{\mathbf{x} \,|\, x_i \leq \nu\} \,|\, \nu \in \mathcal{X}_i\} & \text{if } \mathcal{X}_i \text{ is ordinal} \\ \{\{\mathbf{x} \,|\, x_i \in B\} \,|\, B \in \mathcal{P}(\mathcal{X}_i)\} & \text{if } \mathcal{X}_i \text{ is categorical} \end{cases} \end{aligned} \tag{52}$$

We now specify the stopping conditions. The usual stopping conditions, which we will use in this algorithm are $C_i = \mathbb{1}_{A_i}$ where A_i are the sets of nodes such that [16]:

1. The remaining part of $\mathcal{L}$ is full of nodes with the same input values:

$$A_1 = \{l \in L \,|\, \forall (\mathbf{x}, y), (\mathbf{x}', y') \in \mathcal{L}_l,\, \mathbf{x} = \mathbf{x}'\} \tag{53}$$

2. A minimum number of learning values is remaining or less:

$$A_2 = \{l \in L \,|\, |\mathcal{L}_l| \leq N_{\min}\} \tag{54}$$

3. The depth of a node is greater or equal than $D_{\max}$:

$$A_3 = \{l \in L \,|\, d_l \geq D_{\max}\} \tag{55}$$

   where d_l is the number of ancestors of l. We will denote $D_{\max} = \infty$ whenever there is no limit in the depth of the tree.

4. The maximum entropy change is lower than a trigger value β:

$$A_4 = \{l \in L \,|\, \max_{A \subseteq Q(\mathcal{X})} \Delta i(l, A) \leq \beta\} \tag{56}$$

There, $N_{\min}$, $D_{\max}$, β are called the "hyperparameters" of the regression tree algorithm and must be set a priori by the modeler.

Regression trees have shown to be able to handle heterogeneous data (categorical or ordinal), robust to outliers or errors in labels and can find very complex relations between the input and output spaces. Unfortunately, they are very prone to overfit the relations (considerably bigger error in the testing set compared to the learning set) and have also a high variance (high sensitivity to changes in the input variables) [16].

There is a well-known method to reduce variance of models with high variance and low bias called bagging [18]. It basically consists of building multiple models of the same type with random samples $\mathcal{L}^m$ of the learning set to finally average their predictions. When the base models are *Regression Trees* the resulting bagging model is called a *Random Forest Regressor*.

6.1.2 Random Forest Regression

Definition 6.7. Let $\mathcal{L}$ be a learning set. Let $\{\mathcal{L}^m\}_{m=1}^M$ be a succession of of uniform samples with replacement of size $N = |\mathcal{L}|$ from $\mathcal{L}$. Here, "with replacement" means that $\mathcal{L}^m$ can contain any $(\mathbf{x}, y) \in \mathcal{L}$ multiple times. Let T^m be the greedy regression tree built from the learning set $\mathcal{L}^m$ and its induced regressor ϕ_m. A *random forest* $\Phi_{\mathcal{L}}$ is a regressor represented by a set of greedy regression trees $\{T^m\}_{m=1}^M$ such that if $\mathbf{x} \in \mathcal{X}$

$$\Phi_{\mathcal{L}}(\mathbf{x}) = \frac{1}{M} \sum_{m=1}^{M} \phi_m(\mathbf{x}) \tag{57}$$

An interesting result from Random Forests is that they induce a estimator for the importance of the input variables in the output, it is called the *feature importance* estimator. Intuitively, it is the weighted sum of the entropy changes for all nodes that split under an input variable j averaged through the set of trees. Then the most important variables are the ones which generate more partitions in the trees and reduce considerably the entropy after those partitions. Of course the feature importance could also be well defined for regression trees, but it is not usually used as regression trees tend to have a big variance and so the estimation $I(j)$ is very unstable when the learning set changes. This motivates the following definitions.

Definition 6.8. Let $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \dots \mathcal{X}_n$, $\mathcal{Y}$ be input and output spaces respectively and $\{T^m\}_{m=1}^M$, Φ a random forest regressor. $I(j)$ is called the importance of the feature j and is defined as:

$$I(j) = \frac{1}{M} \sum_{m=1}^{M} \sum_{t \in T^m} \frac{|\mathcal{L}_t|}{|\mathcal{L}|} \mathbb{1}_{[j_t = j]} \Delta i(t, X_{t0}) \tag{58}$$

Where $\mathbb{1}_{[j_t=j]}$ denotes the indicator function which equals 1 only when the splitting of the node t into $t0$ and $t1$, in the Algorithm 6.2 to build the tree, was made over the space $\mathcal{X}_j$. Recalling from (52) that our partition domain $Q(\mathcal{X})$ is restricted to partitions on one input space at the time.

With the Random Forest Regressor well defined, we can proceed to describe the specific implementation to predict the absorption parameters $A_{\max}$, $t_{\max}$, a.

6.1.3 Random Forest Regressor Implementation

Let $\{\mathcal{L}_i\}_{i=1}^3$, $\{\mathcal{T}_i\}_{i=1}^3$ be the learning and testing sets built with the data provided by our tests user and following the algorithm specified at the beginning of the section. Let $\mathcal{T}_i^{\text{val}}$, $\mathcal{T}_i^{\text{test}}$ random samples of size $\frac{|\mathcal{T}_i|}{2}$ of each testing set $\mathcal{T}_i$. To build our regressors $\Phi_{\mathcal{L}_i}$ we used the implementation of the Random Forest Regressor in the python library, *sklearn*, and performed a grid search using the validation set to choose the best hyperparameters, as specified in the following algorithm.

Algorithm 6.3. The grid search algorithm consists of:

1. Select a set of values for each hyperparameter N_{est}, $D_{\max}$ which are the number of trees in the forest and the max depth of each tree, for which we chose the sets $\bar{N}_{\text{est}} = \{10, 50, 100\}$ and $\bar{D}_{\max} = \{\infty, 3, 6, 10\}$. Where $D_{\max} = \infty$ means that there is no depth limitation.

2. For every $\mathbf{h} \in \bar{N}_{\text{est}} \times \bar{D}_{\max}$:

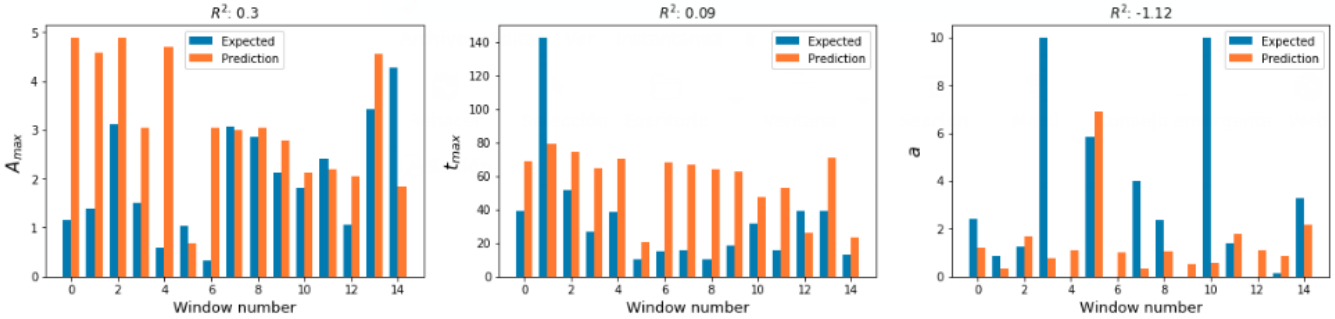

Figure 21: Random forest regressor predictions of the absorption parameters with their corresponding R^2 defined in (37) for a sample of windows of user h5mxumnh. We used a bar chart to compare the expected value and the prediction side by side.

   (a) Build a random forest $\Phi_{\mathcal{L}_i,\mathbf{h}}$ by using the Algorithm 6.2 to build the regression trees.

   (b) Evaluate the $R^2_{i,\mathbf{h}}$ coefficient defined in 37 on the set $\mathcal{T}_i^{\text{val}}$.

3. Choose the random forest $\Phi_{\mathcal{L}_i}$ with the lower $R^2_{i,\mathbf{h}}$ evaluated on $\mathcal{T}_i^{\text{val}}$.

By following the Algorithm 6.3 we found the set of regressors $\{\Phi_{\mathcal{L}_i}\}_{i=1}^3$ that best predict the absorption parameters $A_{\max}$, $t_{\max}$, a. For all of them the best values for the hyperparameters were $N_{\text{est}} = 10$ and $D_{\max} = 6$.

The performance of an automated learning model must never be evaluated over $\mathcal{L}_i$ nor $\mathcal{T}_i^{\text{val}}$ because the model was built to best fit them both, so it is biased to those sets. The right way to check the performance is to evaluate the model in the set $\mathcal{T}_i^{\text{test}}$ which was never part of the model construction, so that we can verify if the model is able to predict patterns that it still does not know. Then, after evaluating every regressor $\Phi_{\mathcal{L}_i}$ in the test set $\mathcal{T}_i^{\text{test}}$ the results (See Figure 21) show that except for the $A_{\max}$ model, the models were not able to find a relation between the input and output matrix. The R^2 coefficient of $A_{\max}$ is still low. Although the determination coefficients R_i^2 are very low, the fact that the $A_{\max}$ had the best results was expected. Recall from the preliminaries that carbohydrates are the main source of glucose because they are basically chains of sugars which are easily broken down by proteins to glucose. So, at meal ingestion there should be a close relation between carbs and $A_{\max}$ which would make this prediction easy if the data integrity was good enough. There aren't any indications that there is a close relation between $t_{\max}$, a and the nutritional value, they could be related more to the actual state of the body, the activities previous and after the ingestion, or to deeper meal details like the way it was cooked or the chemical composition of the meal.

Furthermore, the fact that the carbs and the absorption parameter $A_{\max}$ are closely related implies that the contribution of the carbs to the prediction of the absorption is the largest. By using the feature importance definition (58) normalized to 1, the results showed a contribution of the carbs of around 90% to the prediction of the absorption parameter $A_{\max}$. See Figure 22. The rest of the importance values do not deserve analysis as their values are very small and could be due to the noise of the data so they are not comparable. We will not plot the importances for $\Phi_{\mathcal{L}_2}$ and $\Phi_{\mathcal{L}_3}$ as their performances weren't good enough.

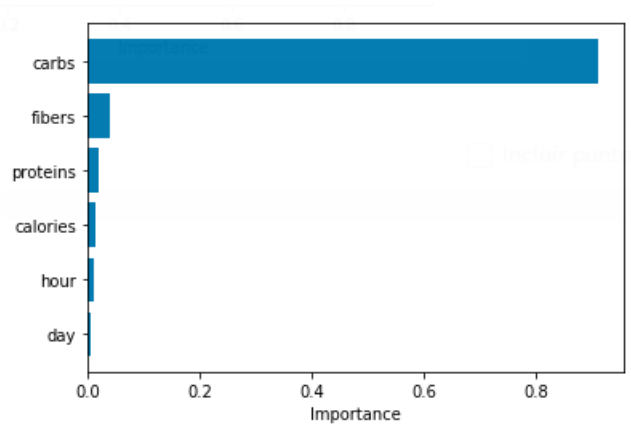

Figure 22: $A_{\max}$ model feature importances $I(j)$ defined by equation (58) for the regressor $\Phi_{\mathcal{L}_1}$

7 Prediction of Glucose

In this chapter, we will predict the glucose concentration in plasma for the next two or three hours after the meal ingestion with its corresponding uncertainty band. To do so, we need first to suppose an a priori probability distribution for the human parameters I_s, I_{decay}, I_{slope}, G_{act} and the nutritional value calories, carbs, proteins, fats and fibers. This will be our method to insert uncertainty in our deterministic models. In the first place, there is no given data from which computing the uncertainty of the nutritional value, so, as discussed at the end of the Chapter 5, we will suppose that the nutritional value is a random variable with normal distribution with μ equal to the reported value and σ of the order of the nutritional value of one and a half slices of bread. In the second place, the probability distributions of the human parameters as described also at the end of Chapter 5. Table 5 summarizes the distributions proposed for one of the test users.

These distributions were the last ingredient required to implement the prediction algorithm. Let's recall that the data is partitioned by a set of windows so, the following algorithm describes how to predict the glucose curve per window.

Algorithm 7.1. Let's suppose that we want to predict the glucose plasma concentration after some meal ingestion for which we know the nutritional value (calories, carbs, protein, fat, fiber).

1. At the end of section 6, the absorption parameters were computed for all windows. Then we use the $A_{\max}$, $t_{\max}$, a for the closest previous meal ingestion to compute the $A_{-1}(t)$ term in the model.

2. For most windows $I_o = 0$. This is not surprising because we are choosing the glucose windows to start in a local minimum. If $I_o > 0$, $A(0) = 0$ and $A_{-1}(0) \sim 0$ then $dG/dt|_{t=0} < 0$. For this reason, we chose to leave $I_o = 0$ for the prediction algorithm. It is important to recall that $I(t)$ is not exactly the plasma insulin concentration, but some variable related to it. It could be even more related the tissue insulin which is likely to be 0 as the insulin rapidly degrades in tissues.

3. Generate a list of modeled glucose time series by performing the following steps N times:

Parameter	**Distribution**	μ	σ
calories	$\mathcal{N}(\mu, \sigma^2)$	Value given by the user	100kcal
carbs			20g
protein			4g
fat			3g
fiber			2g
$I_{\mathbf{s}}$	$\log\mathcal{N}(\mu, \sigma^2)$	-8.75	0.947
I_{slope}	$\log\mathcal{N}(\mu, \sigma^2)$	-1.84	0.952
I_{decay}	$\log\mathcal{N}(\mu, \sigma^2)$	-3.87	0.952
G_{act}	$\mathcal{N}(\mu, \sigma^2)$	$86.5\frac{\text{mg}}{\text{dL}}$	$13.3\frac{\text{mg}}{\text{dL}}$

Table 5: Human parameters and nutritional value distribution for user h5mxumnh

(a) Generate a random sample of the human parameters I_s, I_{decay}, I_{slope}, G_{act} and the nutritional value (calories, carbs, protein, fat, fiber) with the distributions given in Table 5.

(b) With the random sample (calories, carbs, protein, fat, fiber) and the information provided by the test user, build an input vector $\mathbf{x} \in \mathcal{X}$ as described at the beginning of the chapter 7. Set $A_{\max} = \Phi_{\mathcal{L}_1}(\mathbf{x})$, $t_{\max} = \Phi_{\mathcal{L}_2}(\mathbf{x})$, $a = \Phi_{\mathcal{L}_3}(\mathbf{x})$ by using the regressors built with the Algorithm 6.3.

(c) Solve the glucose model using the random sample, $I_o = 0$ and $G_o = G^t(0)$, solve the glucose model to obtain $G(t; A_{\max}, t_{\max}, \dots I_{\text{decay}})$.

4. Let $G^i(t)$ be the i-th random sample evaluated at time t. The predicted glucose $\hat{G}(t)$ and its uncertainty $\sigma_G(t)$ are calculated time-wise as:

$$\hat{G}(t) = \frac{1}{N}\sum_{i=1}^{N} G^i(t) \tag{59}$$

$$\sigma_G(t) = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}(G^i(t) - \hat{G}(t))^2} \tag{60}$$

By following this algorithm with $N = 500$, we got a list of predictions for windows in the testing set $\mathcal{T}^i$. Of course, we expect the predictions not to be good enough because, as we have seen in previous chapters, the data provided was very noisy to make decent predictions. There were three different types of results in the predictions (See Figure 23):

- The completely deviated predictions which didn't fit the size of the glucose peak nor its time. Unfortunately this was the result for around 80% of the windows and it is clearly due to the big deviation of the regressors.

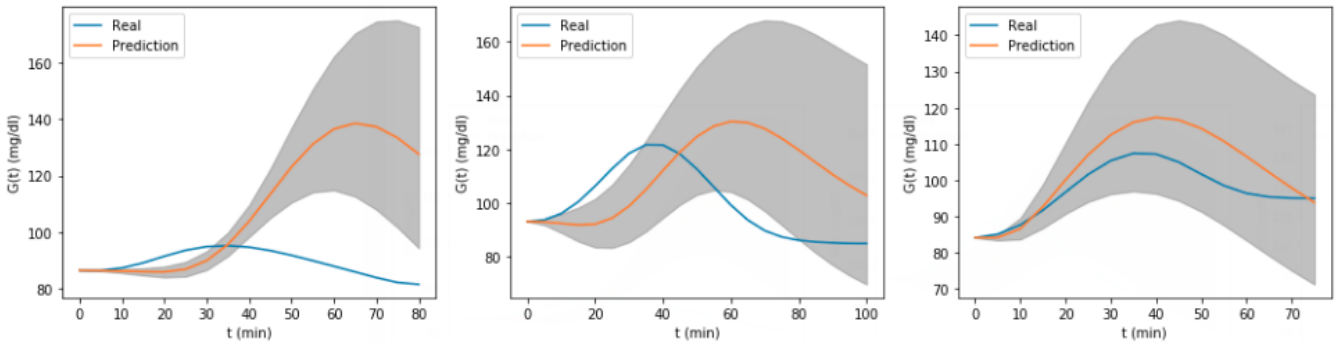

Figure 23: Glucose prediction for some test windows of user h5mxumnh, the gray areas represents the uncertainty of the prediction computed as $G^i(t) \pm \sigma_G(t)$. (a) Completely deviated prediction. (b) Prediction with shifted peak. (c) Good prediction

- The prediction with shifted peak where the size of the peak was close to the real one but its time was shifted. This is the case when the prediction of $A_{\max}$ was good but not for $t_{\max}$ which is likely to happen because $R_1^2 = 0.3$, $R_2^2 = 0.09$. This was the result for around 18% of the windows.
- The good predictions corresponding to the 2% of the windows. This result does not represent a significant enough sample to say that it is due to the goodness of the model.

The large uncertainty bounds are a common pattern in models related to glucose dynamics (See [1, 2]), because the metabolism is so complex that the parameters estimation is full of assumptions that not necessarily hold in reality. We also computed a percentual error measure for the prediction in a window as:

$$\text{error} = \frac{||\hat{G}(t) - G(t)||_2}{\Delta G_{\max}}, \quad \Delta G_{\max} = \max_t G(t) - G(0) \tag{61}$$

Where $\Delta G_{\max}$ is the size of the peak for that window. In Figure 24 most values fall after 1.5 which represent an error of 31%.

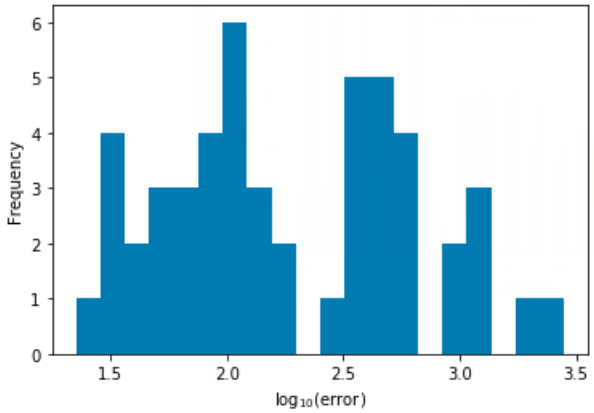

Figure 24: Logarithmic plot of the prediction errors defined in (61) of the test set $\mathcal{T}_i$ of user h5mxumnh

8 Conclusions and Discussion

In this Chapter, we combined all of the results of the previous chapters and made predictions of the glucose plasma concentration just by knowing the nutritional value of the meal. Lets first make a quick review of what we needed and did to reach this objective.

First, by using the knowledge of the glucose homeostasis process and based on previously proposed models, we built a 2-containers model to represent the interaction between the plasma glucose and some variable which must be strongly related with insulin. We have then a model

$$\begin{aligned} \frac{dG}{dt} &= A(t) + A_{-1}(t) - U(G(t), I(t)) \\ \frac{dI}{dt} &= S(G(t)) - E(I(t)) \\ I(0) &= I_o \\ G(0) &= G_0 \end{aligned} \tag{62}$$

which solution in this chapter will be denoted by $G(t; A_{\max}, t_{\max}, I_{\text{decay}})$, $I(t; A_{\max}, t_{\max}, \dots I_{\text{decay}})$ to explicitly show the dependency with the parameters. For most of this parameters we were able to found some common values and ranges in previous related researches (Table 1).

The following step was to analize, segment and clean the data provided by a list of test users. It was composed of a time series containing plasma glucose measurements and a table with the nutritional value and description of the meals ingested. The plasma glucose time series had an acceptable integrity with some sampling and missing data problems, which were easily solved with simple processing methods. The nutritional value table was very imprecise in the time at which the meal ingestion was reported and in the nutritional value of the meals. Fortunately, as the glucose time series was good enough we were able to fit the model to find the values of the parameters. For this we partitioned the data into windows containing only data some hours after each meal ingestion.

In third place, we separated the parameters of the model into two types: human and meal parameters. The first class is related to the parameters impacted mostly by the person's metabolism which are supposed to remain almost constant, like I_s or G_{act} and the latter to the parameters impacted mostly by the meal ingestion like $A_{\max}$ or $t_{\max}$. Thus, we performed a first fitting of the model with the data to find the distribution of the human parameters and choose, per each user, the best estimation of the human parameters (Figure 10 and Table 4). After that, we performed the same optimization but by keeping the human parameters constant to find the meal parameters of each window.

Finally, we proposed an algorithm to make predictions of the glucose concentration after a meal ingestion by using as input only the nutritional value of the meal. For this, we used three Random Forest Regressors ($\Phi_{\mathcal{L}_1}, \Phi_{\mathcal{L}_2}, \Phi_{\mathcal{L}_3}$) to predict the absorption parameters $A_{\max}$, $t_{\max}$, a from the nutritional value. Then with the human parameters obtained by optimization and the glucose model proposed, we got an estimate of the future glucose concentration. Unfortunately, due to the lack of integrity in the data, we couldn't fit the relation between the nutritional value and the absorption parameters.

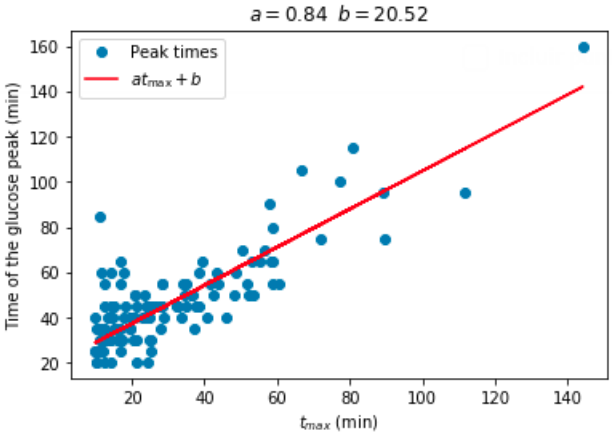

Figure 25: Relation between t^* and $t_{\max}$ for user h5mxumnh. We used this user because this is the only one who has a significant amount of windows with good fittings to make a linear regression.

Before talking about the conclusions of this project, we analyze some additional interesting results obtained when looking deeper into the data and the model. First, we know intuitively that the absorption function and the glucose have something in common: they both reach a maximum value at some time t from which they start decreasing to their base value, so, it is interesting to ask if there exist a relation between the times at which both peaks occur. We called $t_{\max}$ the time of the peak in the absorption function A and t^* the time of the glucose peak:

$$t^* = \arg\max_t G(t) - G(0) \tag{63}$$

Figure 25 shows a linear trend between $t_{\max}$ and t^*. What is more interesting is the slope and intercept of the linear regression. First we could say that the slope value 0.84 is very close to 1 and so, the intercept of 20.5min is telling us that the absorption peak is usually shifted 20.5 minutes from glucose peak. This matches the observations made by different authors (See [2, 3]).

Another important detail to analyze is whether the large bias and uncertainty bands in the predictions is mostly due to the uncertainty of the human parameters obtained by optimization in Chapter 4 or the inability of the regressors $\Phi_{\mathcal{L}_i}$ to find a relation between the nutritional value and the absorption parameters. In order to anwser this question we performed the simulations shown in Figure 26:

a) First, suppose that we somehow found regressors $\Phi_{\mathcal{L}_1}$, $\Phi_{\mathcal{L}_2}$, $\Phi_{\mathcal{L}_3}$ for the absorption parameters $A_{\max}$, $t_{\max}$, a such that $R_i^2 = 0.75$, which is a normal prediction performance when the data has an acceptable integrity. If that is the case, by using equation (37), the mean squared error of each absorption parameter prediction must be:

$$\text{mse}(\mathbf{y}_i^{\text{eval}}, \mathbf{y}_i^{\text{test}}) = 0.25\text{var}(\mathbf{y}_i^{\text{test}}). \tag{64}$$

Then in the Algorithm 7.1 instead of using $\Phi_{\mathcal{L}_i}$ to get the values of the absorption parameters, we would use:

$$p_i = \text{value}_i + \epsilon_i, \tag{65}$$

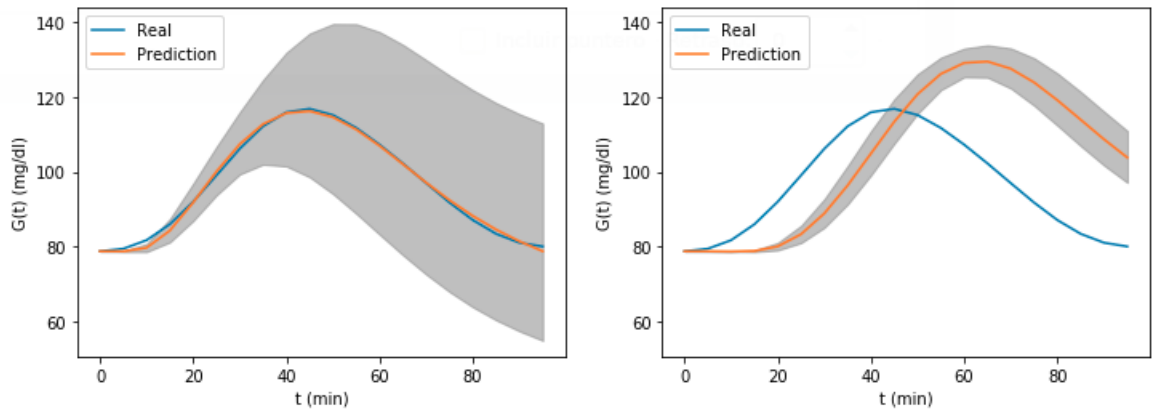

Figure 26: (a) Simulation with an ideal unbiased model for absorption parameters prediction for a test window of user h5mxumnh.
(b) Simulation without uncertainty in the human parameters for the same window.

where p_i denotes the i-th absorption parameter, value_i is its real value and ϵ_i is a random variable $\epsilon_i \sim \text{Normal}(0,\ 0.25\text{var}(\mathbf{y}_i^{\text{test}}))$

b) In contrast, lets compute the absorption parameters with the regressors $\Phi_{\mathcal{L}_1}$, $\Phi_{\mathcal{L}_2}$, $\Phi_{\mathcal{L}_3}$ obtained with the Algorithm 6.3 but in the prediction Algorithm 7.1 we suppose that human parameters have no uncertainty.

With these simulations, we isolate the errors injected by the absorption parameters predictions and the uncertainty in the parameters of the model (See Figure 26), panel (a) shows a large uncertainty bands while panel (b) shows a large bias. Comparing the uncertainty bands in Figure 26 one can conclude that the human parameters are the main source of variance in the model. This is probably due to the fact that some of the human parameters might vary within one full order of magnitude. In contrast, the low variance of the Random Forest Regressors, ensures that although some of the nutritional value variables have big variance, the absorption parameters have a low variance which results in low variance in the prediction due to the absorption parameters as can be seen in Figure 26 (b). Nevertheless, the absorption parameter regressors induce a large bias in the prediction, compared to the human parameters which provide almost zero bias. One could say that the prediction in panel (b) is an estimation of glucose concentration if the person had eaten what they reported they ate.

We conclude this project by bringing the most important facts of the results:

- The parsimonious model proposed is able to fit the glucose dynamics of a person just by identifying four metabolic parameters I_{s}, I_{decay}, I_{slope}, G_{act} and fitting three parameters related to the absorption, A_{max}, t_{max}, a, which vary over the meals.
- The values of the metabolic and absorption parameters fall in the normal human ranges, so we can say that the model is a good representation of the phenomenology of the real system.
- An advantage of the phenomenological models over the black-box models is that we can use fundamental laws like mass conservation or continuity to get additional information from the

system. This was seen when we were able to propose a method to analytically calculate the amount of carbs that the person ate and compare them with the logged ones.

- The fact that the amount of carbs logged wasn't related to the size of the glucose peak shows a lack of integrity in the data, see Figure 15. It should have at least an increasing trend.
- The results show that there is a fixed shift between the glucose and absorption peaks. For one of the test users, it was of around 20 minutes.

Bibliography

[1] Dalla Man, C., Rizza, R. and Cobelli, C. (2007). Meal Simulation Model of the Glucose-Insulin System. IEEE Transactions on Biomedical Engineering, 54(10), pp.1740-1749.

[2] Dalla Man, C., Camilleri, M. and Cobelli, C. (2006). A System Model of Oral Glucose Absorption: Validation on Gold Standard Data. IEEE Transactions on Biomedical Engineering, 53(12), pp.2472-2478.

[3] Woerle, H., Meyer, C., Dostou, J., Gosmanov, N., Islam, N., Popa, E., Wittlin, S., Welle, S. and Gerich, J. (2003). Pathways for glucose disposal after meal ingestion in humans. American Journal of Physiology-Endocrinology and Metabolism, 284(4), pp.E716-E725.

[4] Pennant, M., Bluck, L., Marcovecchio, M., Salgin, B., Hovorka, R. and Dunger, D. (2008). Insulin Administration and Rate of Glucose Appearance in People With Type 1 Diabetes. Diabetes Care, 31(11), pp.2183-2187.

[5] Wachters-Hagedoorn, R., Priebe, M., Heimweg, J., Heiner, A., Englyst, K., Holst, J., Stellaard, F. and Vonk, R. (2006). The Rate of Intestinal Glucose Absorption Is Correlated with Plasma Glucose-Dependent Insulinotropic Polypeptide Concentrations in Healthy Men. The Journal of Nutrition, 136(6), pp.1511-1516.

[6] Evensen, G. (2007). Data assimilation. Springer.

[7] Ackerman, E., Gatewood, L., Rosevear, J., & Molnar, G. (1965). Model studies of blood-glucose regulation. The Bulletin Of Mathematical Biophysics, 27(S1), 21-37. doi: 10.1007/bf02477259

[8] Bergman, R., Ider, Y., Bowden, C., & Cobelli, C. (1979). Quantitative estimation of insulin sensitivity. American Journal Of Physiology-Endocrinology And Metabolism, 236(6), E667. doi: 10.1152/ajpendo.1979.236.6.e667

[9] P. Bogacki, L.F. Shampine, A 3(2) Pair of Runge-Kutta Formulas, Appl. Math. Lett. Vol. 2, No. 4. pp. 321-325, 1989.

[10] Diabetes. (2020). Retrieved 10 February 2020, from https://www.who.int/en/news-room/fact-sheets/detail/diabetes

[11] Röder, P., Wu, B., Liu, Y., & Han, W. (2016). Pancreatic regulation of glucose homeostasis. Experimental Molecular Medicine, 48(3), e219-e219. doi: 10.1038/emm.2016.6

[12] SciPy v1.5.0 Reference Guide. (2020). Retrieved 21 June 2020, from https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.minimize.html

[13] Zhu, C and R H Byrd and J Nocedal. 1997. L-BFGS-B: Algorithm 778: L-BFGS-B, FORTRAN routines for large scale bound constrained optimization. ACM Transactions on Mathematical Software 23 (4): 550-560.

[14] Shiang, K., Kandeel, F. (2010). A computational model of the human glucose-insulin regulatory system. Journal Of Biomedical Research, 24(5), 347-364. doi: 10.1016/s1674-8301(10)60048-6

[15] Caumo, A., Bergman, R., Cobelli, C. (2000). Insulin Sensitivity from Meal Tolerance Tests in Normal Subjects: A Minimal Model Index. The Journal Of Clinical Endocrinology Metabolism, 85(11), 4396-4402. doi: 10.1210/jcem.85.11.6982

[16] Louppe, G. (2014). Understanding Random Forests. University of Liège.

[17] Scikit-learn: Machine Learning in Python, Pedregosa et al., JMLR 12, pp. 2825-2830, 2011.

[18] Breiman, Arcing Classifiers, Annals of Statistics 1998.

[19] Calabrese, A., Gibby, C., Meinke, B., Revilla, M., Titchenal, A. Human nutrition.

[20] Kalsbeek, A., la Fleur, S., & Fliers, E. (2014). Circadian control of glucose metabolism. Molecular Metabolism, 3(4), 372-383. doi: 10.1016/j.molmet.2014.03.002

[21] SciPy v1.5.0 Reference Guide. (2020). Retrieved 28 April 2020, from https://docs.scipy.org/doc/scipy/reference/generated/scipy.integrate.solve_ivp.html

[22] Charrondiere, U., Chevassus-Agnes, S., Marroni, S., Burlingame, B. (2004). Impact of different macronutrient definitions and energy conversion factors on energy supply estimations. Journal Of Food Composition And Analysis, 17(3-4), 339-360. doi: 10.1016/j.jfca.2004.03.011

[23] Reid, K., Baron, K., & Zee, P. (2014). Meal timing influences daily caloric intake in healthy adults. Nutrition Research, 34(11), 930-935. doi: 10.1016/j.nutres.2014.09.010