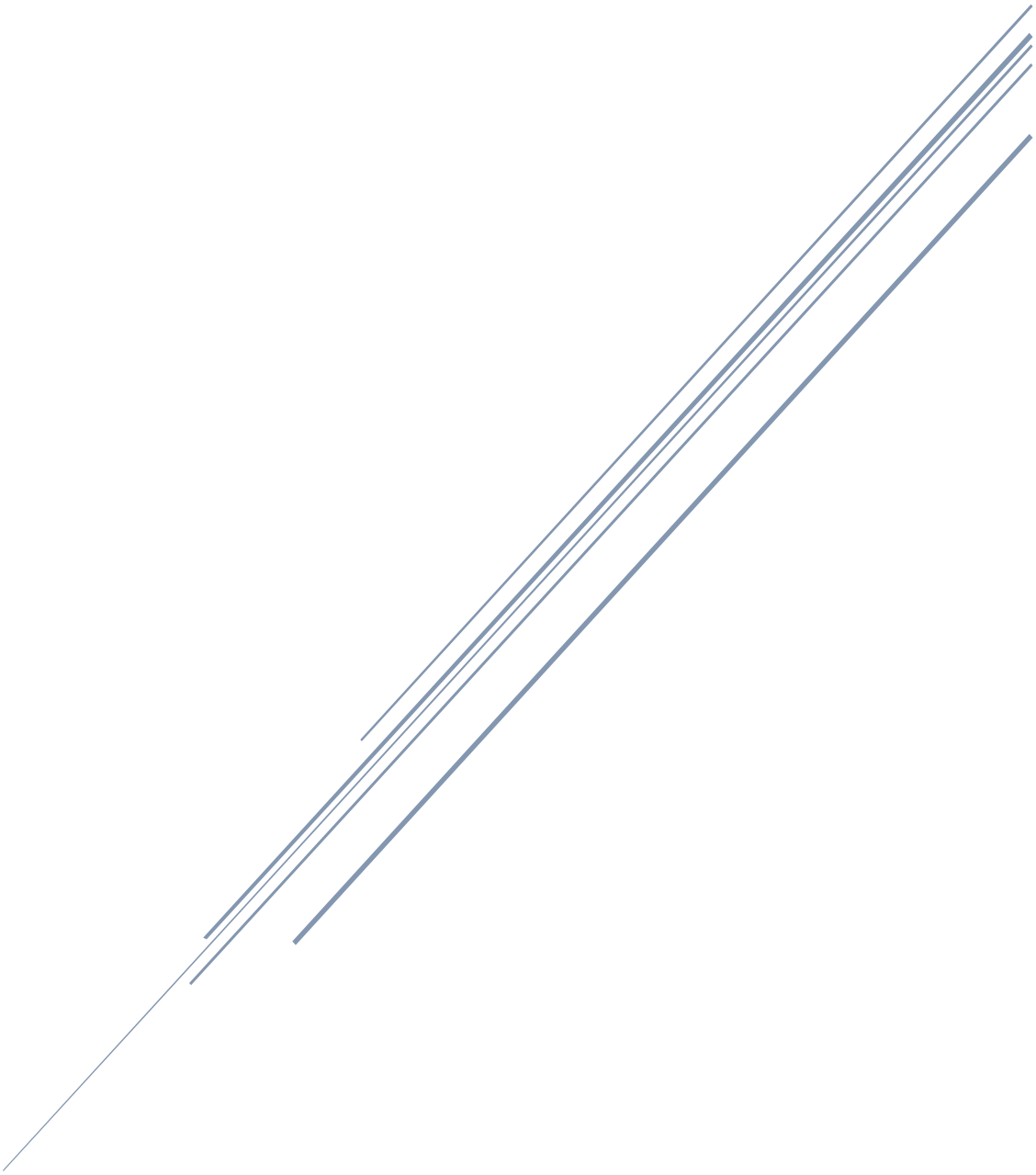
MATHEMATICS EXPLORATION

Determining the optimal timing for insulin injection to minimize glucose level variability after a meal in ideal conditions



Analysis and Approaches SL
Number of pages: 23

Contents

Introduction

Type 1 diabetes is a relatively common genetic disease affecting millions of people around the world, including me. It results in a malfunction of the pancreas and so the shutting down of insulin production which leads to higher glucose concentrations in the blood that are associated with severe long-term damage to many tissues if left untreated. Modern treatment has, however, tamed it into more of an inconvenience rather than a death sentence, which allows me to live essentially normal life. However, it is still not without inconveniences. One such thing I, and many other diabetics I have spoken with, have continuously struggled with is how to best time the injection of insulin with respect to a meal to minimize glucose variability, i.e., to keep its concentration in the ideal range, at all times as close as possible to that measured prior to eating.

This is something that has bothered me for almost the entire decade that I have soon lived with the condition as the behavior of glucose levels in response to eating, even if carbohydrates are accurately known, often seems rather unpredictable. Sometimes they rise way more than expected and other times they barely change, or just drop. Changes in everyday treatment are therefore primarily made with the method of trial and error, which is highly unsatisfactory for an engineering-oriented person like me. That is why I want to approach this problem much more systematically and come up with a method to mathematically determine the optimal timing using the wide range of mathematics we have learned on the IB to finally get some clarity on the issue and so potentially further improve the control of the disease for myself and possibly others as well. With further development it could be used by the very active diabetes open-source community to, for example, build an algorithm that automatically calculates the optimal injection times based on an individual's historical data, provided that it is sufficient. What I also like about the topic is that it is a concrete and useful real-world application of mathematics that inherently combines many of the most interesting branches of mathematics to me, such as statistics, modeling and calculus.

Aim and approach

My aim is to determine the optimal timing for insulin injection to minimize glucose level variability after a meal in ideal conditions. Implicitly, although beyond the scope of this exploration, a secondary aim of mine is also to develop a robust method for its determination in the hopes of providing a basis for such a feature in diabetes-related software. I will use my own data, since its collection requires certain exactitude and health data is generally considered very sensitive.

My initial idea was to try to get my glucose levels stable within an ideal range and deliberately manipulate them by certain dosages, while altering my routine to control other variables. However, I chose not to pursue this deliberate self-experimentation as it would have been on the gray area of the IB guidelines, and I wanted the method to work with noisier, more realistic data as well.

Instead, for the purposes of data collection I decided to just start writing down the amounts of both insulin and carbohydrates and their exact timings for over a month. From these I could then isolate the naturally occurring instances with the most stable ends, in which I had corrected the levels by 1 unit (0.01ml) of Fiasp insulin or 10.5 grams of carbohydrates from 5 “Siripiri” glucose tablets, which I used exclusively for all moderate corrections during the data collection period in order to carefully control both the amount and composition of the carbohydrates. The amounts were chosen based on my insulin to carbohydrates dosage ratio of 1 to 10, meaning that they should produce similar magnitudes of change, only in opposite directions. As the initial concentration of glucose has an effect on the extent of the effect of either carbohydrates or insulin, this will introduce more error and uncertainty but averaging multiple observations should give at least somewhat directive values.

I will then average this data on the magnitude and duration of the glucose level changes and model their behaviors after consumption of 10.5g carbohydrates and 1 unit of insulin (0.01ml) to get approximate curves and functions for both. They can then be combined into a single function, the

total area under which can be used as a proxy for glucose variability, in this case defined as any change in concentration, as it is a measure of absolute divergence from zero over time, giving a single, easily comparable number involving both the magnitude and time of glucose concentration change. I can then transform the insulin curve in the horizontal dimension to change its timing with respect to carbohydrate consumption. The optimal timing can be found as that transformation of the insulin curve, which produces the smallest area, the definite integral, under the combined curve.

Due to time constraints and the massive amounts of data processing necessary, I concluded that I had to program everything. However, my enthusiasm greatly outweighed my expertise. I had searched this excuse for long due to my fascination with computer science and software and now that the opportunity to learn it in a meaningful context finally presented itself, I was very excited to throw myself into it. I decided to learn Python, which is a relatively easy-to-learn but extremely powerful programming language that enabled me to develop and test the method simultaneously to collecting the data, which was a long and unpredictable process as I could not do it systematically, but only hope to collect good data while living normally. Continuously calculating all values of the dataset when it was still growing allowed me to take advantage of the law of large numbers without having to wait for the final results. Additionally, any errors I made could be corrected easily and I could continue to use and develop the algorithm even after finishing the project.

Data collection and results

Collecting the data was very easy as the only thing I had to change in my daily routine was just to remember to write down the amounts of insulin and carbohydrates I normally injected and consumed. I regularly wear a Dexcom G6 continuous glucose monitoring (CGM) sensor, held in place by an adhesive patch in my arm, attached to which is a transmitter that reads and transmits the glucose concentration values measured from my interstitial fluid to my cellphone in real time at 5-

minute intervals, which then produces a continuous curve, the values of which I could download as a CSV-file from the Dexcom Clarity website. This corresponds to systematic sampling, which is the only method available for any real-time glucose monitoring. A limitation of this is the rather long time between datapoints obtained with current technology, that only allows us to look at the very general trends, but on the upside, it makes data processing much easier and faster while filtering out noise. There is uncertainty in the sensor measurements, but this may be neglected as it is relatively small (Dexcom, Inc., 2021; Danne, 2017) and most likely systematic which should thus not have any effect on the results as only the general trends, rather than the values themselves, are of key interest. Below is a sample of the formatted CGM data used in the exploration, showing some glucose values and all event types of interest from the data collection interval between 27.10.2021 and 1.12.2021.

Index	Timestamp (YYYY-MM-DDThh:mm:ss)	Event Type	Event Subtype	Glucose Value (mmol/L)	Insulin Value (u)	Carb Value (grams)	Duration (hh:mm:ss)	Glucose Rate of Change (mmol/L/min)	Transmitter Time (Long Integer)	Transmitter ID
206	2021-10-27T17:05:31	EGV	NaN	6.2	NaN	NaN	NaN	NaN	8832673.0	8PBGYQ
207	2021-10-27T17:06:00	Carbs	NaN	NaN	NaN	10.0	NaN	NaN	NaN	NaN
208	2021-10-27T17:10:30	EGV	NaN	6.4	NaN	NaN	NaN	NaN	8832973.0	8PBGYQ
209	2021-10-27T17:15:30	EGV	NaN	6.9	NaN	NaN	NaN	NaN	8833273.0	8PBGYQ
210	2021-10-27T17:20:30	EGV	NaN	7.4	NaN	NaN	NaN	NaN	8833573.0	8PBGYQ
...	...	...	...	...	...	...	...	...	...	...
10171	2021-11-30T08:52:55	EGV	NaN	10.8	NaN	NaN	NaN	NaN	2362573.0	8J44L8
10172	2021-11-30T08:57:55	EGV	NaN	10.6	NaN	NaN	NaN	NaN	2362873.0	8J44L8
10173	2021-11-30T08:59:00	Insulin	Fast-Acting	NaN	1.0	NaN	NaN	NaN	NaN	NaN
10174	2021-11-30T08:59:00	Insulin	Long-Acting	NaN	20.0	NaN	NaN	NaN	NaN	NaN
10175	2021-11-30T09:02:56	EGV	NaN	10.0	NaN	NaN	NaN	NaN	2363173.0	8J44L8

Table 1 – A sample of formatted CGM data from the interval 27.10.2021 - 1.12.2021, showing glucose values, carbohydrate consumption and insulin injections

Table 1 shows glucose concentrations with 5-minute intervals as well as the manually entered events affecting it and their details. During data collection, I used two different transmitters and at least three different sensors and various ampoules of insulin injected manually with a NovoPen Echo, all with (not-too-noticeably) varying levels of efficacy and error, that might or might not add up to a greater overall error.

The total number of data points accumulated during the data collection period was 10 440, which would make for a rather long table. Therefore, they are better presented in a graph such as figure 1 below, which shows a continuous, interpolated trendline for glucose concentration (in blue) as well as each instance where 10.5 carbohydrates were consumed in the form of 5 “Siripiri” glucose tablets (in green) or 1 unit (0.01ml) of Fiasp (fast-acting) insulin was injected (in red).

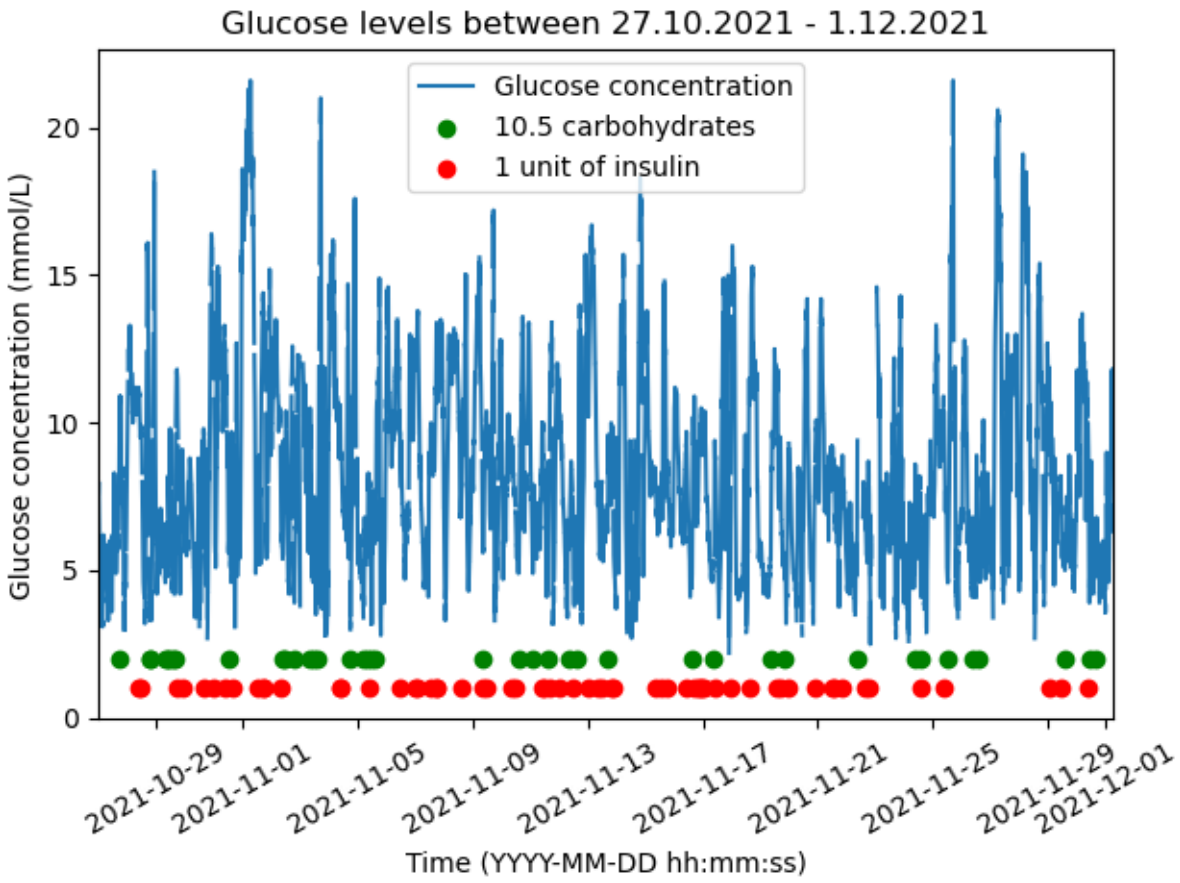
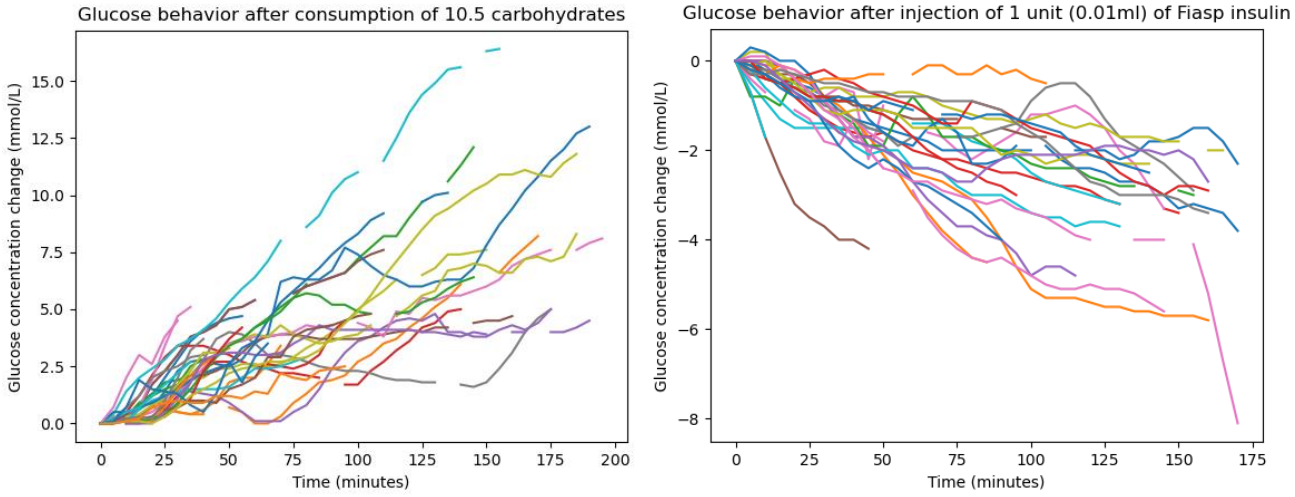


Figure 1 – Graph of glucose concentration against time on the interval 27.10.2021 – 1.12.2021 with consumption of 10.5 carbohydrates and injection of one unit of insulin highlighted in green and red respectively

The variables affecting the behavior of glucose levels are numerous: carbohydrate content and their form (sugars, starches, fiber) in the food, the amount and quality of fats and other nutrients in the food, amount, effectiveness and qualities (such as duration of action) of insulin used, amount of long-acting insulin (in this case 16 to 20 units of Levemir twice a day) in the body, the extent of functionality of the pancreas, insulin resistance and metabolism of the body, the behavior of the liver and other biological processes, time between meals and injections, duration of meals, time of day (phase of circadian clock), overlapping insulin, injection site, the initial glucose level, its direction and rate of change, amount of sleep, exercise, stress and the physical environment and uncertainty and error in the measuring devices as well as the mechanical nature and errors of the injector or its needles. As there is no way to control all these variables, the investigation must be limited to a very approximate, ideal case, the exploration of which, however, may reveal a hint of a universal method which may eventually allow taking these additional factors into account.

To keep as many of these variables as constant as possible, I only considered the instances of consumption of 10.5 carbohydrates and 1 unit of insulin from the moment of logging to three hours forward. This three-hour interval is chosen based on both personal experience and official time of action data from the manufacturer Novo Nordisk (2019), which states that the insulin dosage can be taken anywhere from the start of the meal to 20 minutes after beginning the meal. Within this 3-hour interval, the peak effect is reached with all dosages and the insulin concentration has had time to significantly decrease according to the data, while the probability of other events falling onto the inspected interval is reduced. The same interval is used for both for realistic comparison. Below are graphs of the min-max and max-min behaviors of the glucose levels after carbohydrate consumption (Fig. 2, green in Fig. 1) and insulin injection (Fig. 3, red in Fig. 1) in the three-hour interval,

transformed to start from the origin by plotting them on the same domain starting from zero and by transforming them by $f(x) - f(0)$ to visually compare their relative behaviors:



Figures 2 & 3 – Graphs of interpolated glucose behavior after carbohydrate consumption (left) and insulin injection (right)

From this three-hour interval I programmed the algorithm to choose the data points from the first minimum to the first maximum in the case of carbohydrate consumption and first maximum to first minimum in the case of insulin injection. This approach may be a key weakness, but one made to offset the effect of other variables such as prior "momentum" of the glucose levels and later correction events. It also makes the data much easier and cleaner to process, compare and model.

One final key assumption made for ease of modeling is that of stable ends. It is assumed that the change in glucose levels after an event is distinct and permanent. The ideal case is thus the absence of any variables, which should result in the glucose concentration staying constant on a certain level, which in this case is zero for the purpose of modeling change, and only be affected by the selected event increasing or decreasing it. After the effect has been manifested, i.e., the first maximum or minimum has been reached, it is again free from influences and should thus stay constant on the level it rose or dropped to. This is obviously never the case in real life, even under laboratory conditions due to bodily functions alone but is something that significantly eases the modeling process by allowing the averaging of values of curves of different lengths.

Statistical processing of data

As I wanted to make the algorithm entirely automatic, data selection, processing and filtering had to be based on simple statistical procedures rather than hand-picking. Therefore, the data was filtered in terms of both the duration of the change in glucose concentration as well as its end magnitude by finding and discarding outliers, defined as all values $1.5 \cdot IQR$ greater or smaller than Q_3 or Q_1 respectively, where IQR is the interquartile range and Q_3 and Q_1 are the upper and lower quartiles, enclosing the IQR, which holds 50% of the data within it. Due to the attempt to keep it simple, understandable and concise, this is most likely where the majority of the potential improvements to the algorithm lie as, for example, outliers could be defined differently and the data selected does not currently account for many of the variables mentioned in the data collection and results section such as initial magnitudes, rates of change and proximity of other events.

Below is a table of the data in ascending order, resulting from filtering to the first extreme, used to calculate the quartiles for both duration and end magnitude for both carbohydrate consumption and insulin injection with resulting outliers highlighted in red.

Durations to reach first maximum for carbohydrates consumed (minutes)	15, 20, 20, 20, 20, 25, 25, 35, 40, 40, 40, 45, 45, 45, 45, 45, 55, 55, 60, 60, 60, 65, 65, 80, 85, 85, 85, 100, 115, 140, 150, **160**
End magnitudes for carbohydrates consumed (mmol/L)	0.1, 0.1, 0.3, 0.7, 1.0, 1.0, 1.0, 1.4, 1.9, 2.6, 2.7, 2.7, 2.8, 2.8, 2.9, 3.0, 3.0, 3.1, 3.4, 3.9, 3.9, 4.0, 4.2, 4.3, 4.7, 5.1, 5.7, 6.6, 7.6, **10.1**, **12.1**, **16.4**
Durations to reach first minimum for insulin injected (minutes)	10, 10, 15, 15, 15, 15, 15, 15, 20, 20, 20, 20, 20, 20, 25, 30, 30, 35, 35, 40, 50, 50, 50, 50, 50, 55, 55, 60, 60, 75, 80, 85, 90, 90, 90, 90, 90, 100, 105, 110, 110, 135, 170
End magnitudes for insulin injected (mmol/L)	-7.6, -6.0, -5.7, -4.9, -4.8, -4.5, -4.5, -4.4, -4.2, -4.2, -3.5, -3.2, -3.0, -2.7, -2.4, -2.2, -2.1, -1.9, -1.9, -1.8, -1.7, -1.7, -1.7, -1.7, -1.7, -1.6, -1.4, -1.4, -1.0, -0.9, -0.8, -0.7, -0.5, -0.4, -0.3, -0.3, -0.3, -0.1, -0.1, -0.1, -0.1, 0.0, 0.0

Table 2 – Durations and end magnitudes for carbohydrates consumed and insulin injected in ascending order for calculating quartiles (Bolded red values signify **to-be outliers** after processing)

These data can be summarized as follows:

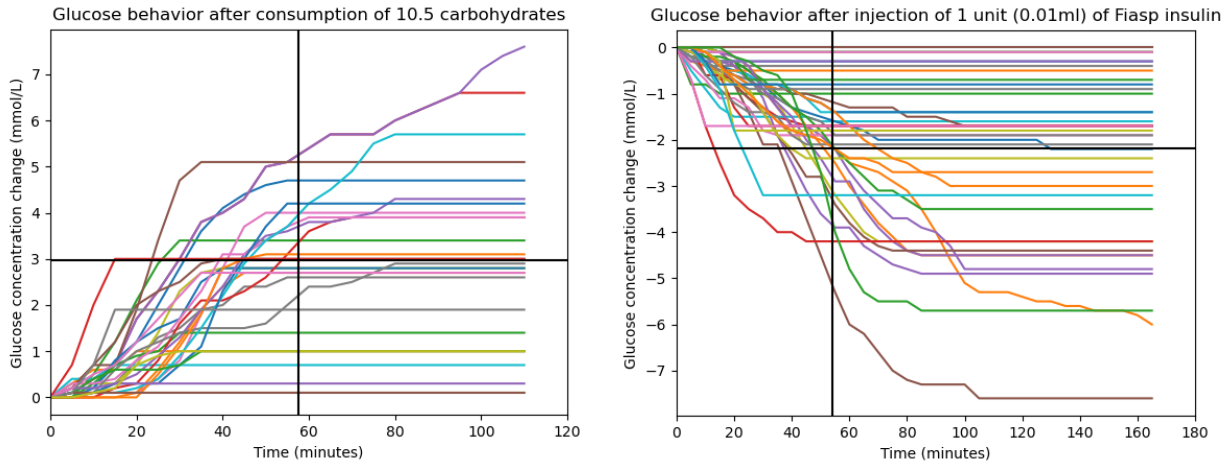
	Consumption of 10.5 carbohydrates		Injection of 1 unit of insulin	
	Duration (minutes)	**End magnitude (mmol/L)**	**Duration (minutes)**	**End magnitude (mmol/L)**
Lower fence ($Q_1 - 1.5 \cdot IQR$)	-35.00	-3.088	-85.00	-8.000
Minimum value	15.00	0.100	10.00	-7.600
Lower quartile $Q_1 = \frac{n+1}{4}^{\text{th}}$ value	36.25	1.525	20.00	-3.500
Median $Q_2 = \frac{n+1}{2}^{\text{th}}$ value	50.00	3.000	50.00	-1.700
Upper quartile $Q_3 = \frac{3(n+1)}{4}^{\text{th}}$ value	83.75	4.600	90.00	-0.500
Maximum value	160.00	16.400	170.00	0.000
Upper fence ($Q_3 + 1.5 \cdot IQR$)	155.00	9.213	195.00	4.000
$IQR = Q_3 - Q_1$	47.50	3.075	70.00	3.000
Mean $\overline{x} = \frac{\sum x}{n}$	57.58	2.983	54.07	-2.186

Table 3 – Statistical summary of the data where Q_1 is the lower quartile, Q_2 is the median, Q_3 is the upper quartile, IQR is the interquartile range, $\bar{x}$ is the mean of observations and n is the number of data points considered

Table 3 shows some of the key characteristics of carbohydrate consumption and insulin injection data, them being mainly its central tendencies, the medians and means, as well as measures of dispersion such as the interquartile range. It also defines the lower and upper fences, values under and over which are considered outliers. For this exploration, they are considered satisfactory as they remove the most extreme outliers. However, as is apparent from, for example, the lower fences for duration being negative minutes, they are not quite realistic and tight enough as I would already consider too little or too quick a change an outlier whereas the algorithm passes them as valid values. Therefore, a further improvement would be to either gather a lot more data and hope for the interquartile ranges to automatically tighten or filter the data with more specific conditions.

Decimal quartiles were handled according to the method detailed on the Brilliant wiki according to which the integer and fractional parts are separated and “the positive difference of the integer observation and its next observation multiplied by the fractional value” is added to the integer number observation. (Brilliant Worldwide, Inc., 2021) The graphs below show the filtered values with

their ends extended from the maximum or minimum point reached up to the length of the instance with the longest duration:



Figures 4 & 5 – Extended graphs of filtered data for carbohydrate consumption (left) and insulin injection (right)

The straight black lines in figures 4 and 5 denote the means on each axis; the vertical one being the mean of duration and the horizontal being the mean of glucose concentration after the change. At their intersections, they divide the graphs into four rectangles of which the one enclosed by the mean lines and the axes will be the section modeled for each.

To construct the average curve filling up the rectangle, I rounded up the duration means up to 60 and 55 to still include the data point right after the duration mean and then used those as the limits of the model as the aim is to find the optimal timing. The curves shorter than it were extended with the final value in accordance with the assumption of stable ends so that each point's magnitude could be averaged across all curves. Averaging the magnitudes of glucose concentration changes to an accuracy of three decimals (to retain enough relevant information for the shape of the curve) at each point on the time interval from zero up to the duration mean gives the following values:

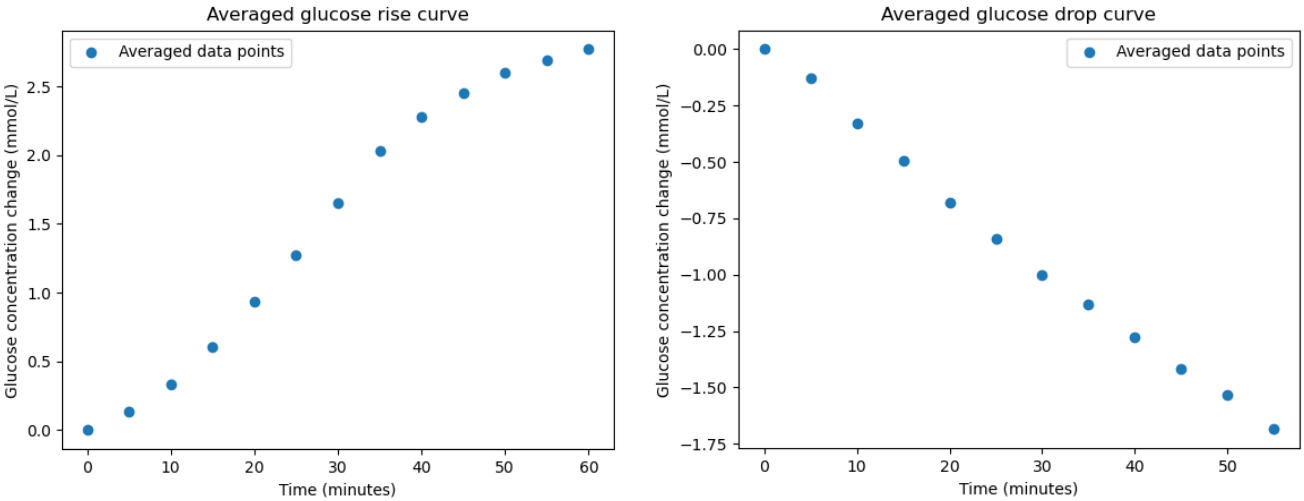
Glucose concentration change (mmol/L)	0.000, 0.138, 0.334, 0.603, 0.934, 1.272, 1.655, 2.028, 2.276, 2.448, 2.597, 2.686, 2.769
Time (minutes)	0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55, 60

Table 4 – Average glucose concentrations against time as a result of consumption of 10.5 carbohydrates

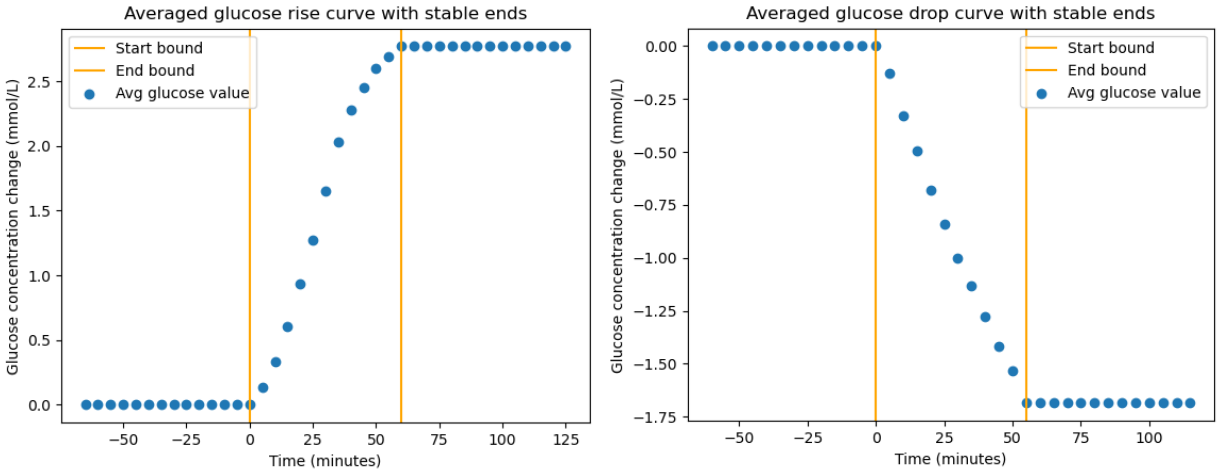
Glucose concentration change (mmol/L)	0.000, -0.128, -0.328, -0.493, -0.679, -0.842, -1.000, -1.130, -1.279, -1.419, -1.535, -1.686
Time (minutes)	0, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 55

Table 5 – Average glucose concentrations against time as a result of injection of 1 unit of insulin

Producing the scatter plots below:



Figures 6 & 7 – Averaged glucose concentration curves for carbohydrate consumption (left) and insulin injection (right)



Figures 8 & 9 – Averaged glucose concentration curves for carbohydrate consumption (left) and insulin injection (right) with extended, stable ends

Figures 8 and 9 show data sets extended from both ends with the same number of data points as their length, to incorporate the assumption of stable ends, setting the context of the change and so making their interpretation and modeling easier.

Modeling glucose concentration changes after consumption of 10.5 carbohydrates

Looking at figures 6 and 8, some general requirements for an appropriate model can be inferred. From figure 6 it can be seen that this function must have an initially gradually increasing slope that reaches its maximum at around the middle point and then gradually decreases to a constant value,

creating a loose S-shape. Figure 8 shows the requirement of two distinct horizontal asymptotes that confine the change and make it permanent when free of other influences.

The generalized logistic function

$$F(t) = A + \frac{K - A}{(C + Qe^{-B(t-M)})^{\frac{1}{v}}}$$

where F in this case refers to the glucose concentration as a function of time (the independent variable, t) that starts from A, the left (in this case lower) asymptote, rising to K (or $A + \frac{K-A}{C^{\frac{1}{v}}}$ if $C \neq 1$), the right (in this case upper) asymptote at growth rate B, while $v > 0$ affects near which asymptote the maximum growth occurs, Q is related to the value of $F(0)$, C typically takes a value of 1 and M is the t value of the sigmoid's midpoint, is a sigmoid function that fulfills both the requirements of stable ends (horizontal asymptotes) and the gradually changing rate of change that is at its peak around midway through the transition, closely resembling the S-shape of the data. Also known as the Richard's curve, it was originally developed for growth modeling as an extension of the simpler logistic function (Wikipedia contributors, 2021), which makes it very fitting for modeling glucose concentration changes. A logistic function gives a great fit as well, but the generalized version offers the most flexibility for curve fitting due to its larger number of parameters, providing more flexible curves, which is better in the context of the algorithm as the symmetry forced by a logistic function may be subject to change with additional data and overfitting is not really an issue as the datapoints are already averaged and follow a distinct shape. Hence, as long as the function produced is integrable, the more parameters, the closer the smooth fit, the better.

The fitting itself was done with SciPy's curve fit function (The SciPy community, 2021) utilizing the Levenberg-Marquardt algorithm (The SciPy community, 2021; Wikipedia contributors, 2021), which is an iterative procedure to minimize the sum of squares of residuals, the in-sample prediction errors:

$$S = \sum_{i=1}^{m} r_i^2$$

where S is the sum of squares of the residuals, m is the number of datapoints, r_i is the ith residual given by $r_i = y_i - f(x_i, \beta)$ where y_i is the actual y value and $f(x_i, \beta)$ is the y value predicted by the model curve that takes the arguments of x_i, the independent variable and β, the model curve's parameters, and i is the iteration number up to m. The minimum value of S occurs where the gradient, its derivative with respect to β, is zero. Rather simplistically expressed; the parameters, β are then refined iteratively according to the Levenberg-Marquardt algorithm to find this minimum, based on initial guesses for the parameters, necessary to reduce the unknowns to just the independent variable, that must already be somewhat close to the final solution. In other words, the curve fit function only adjusted, sharpened so to say, the parameters already guessed by me. The initial guesses for the parameters were as follows:

$\boldsymbol{A = 0}$ was already set by me, since there is no change in glucose concentration at $t = 0$.

$\boldsymbol{K \approx 2.769}$ as this is the average magnitude of the glucose concentrations at $t = 60$.

$\boldsymbol{Q = C = v = 1}$ as this is the case of the logistic function, which appeared to be a close enough approximation and was much easier to evaluate by hand.

$\boldsymbol{M = 30}$ as this is the t value of the sigmoid's midpoint.

$\boldsymbol{B \approx 0.0992}$ as this was visually the closest to the steepness of the scatter plot in figure 6 acquired from calculations involving the real averaged data points. Using all the values for the variables defined above and substituting the point (10, 0.334) into the generalized logistic function gave:

$$F(10) = 0.334 = 0 + \frac{2.769 - 0}{(1 + 1e^{-B(10-30)})^{\frac{1}{1}}}$$

which is equivalent to the logistic function $f(x) = \frac{L}{(1+e^{-k(x-x_0)})}$ where L is the curve's maximum value, k is the logistic growth rate or steepness of the curve and x_0 is the x value of the sigmoid's midpoint. Solving this gave:

$$0.334 = \frac{2.769}{(1 + e^{-B(10-30)})} = \frac{2.769}{(1 + e^{20B})}$$

$$(1 + e^{20B}) = \frac{2.769}{0.334}$$

$$e^{20B} = \frac{2.769}{0.334} - 1 \approx 7.290$$

$$\ln 7.290 = 20B$$

$$B = \frac{\ln 7.290}{20} \approx 0.0992$$

Calling the curve fit function with these rough estimates gives:

$\boldsymbol{A = -0.008139493980971038}$	$\boldsymbol{B = 0.10056191535315151}$
$\boldsymbol{K = 2.794321220615598}$	$\boldsymbol{M = 25.39343305999737}$
$\boldsymbol{C = 1.002164676526231}$	$\boldsymbol{v = 0.5723571840990965}$
$\boldsymbol{Q = 0.5229628037686772}$	

Table 6 – Curve fit parameters based on the Levenberg-Marquardt algorithm

Which then produces the following curve:

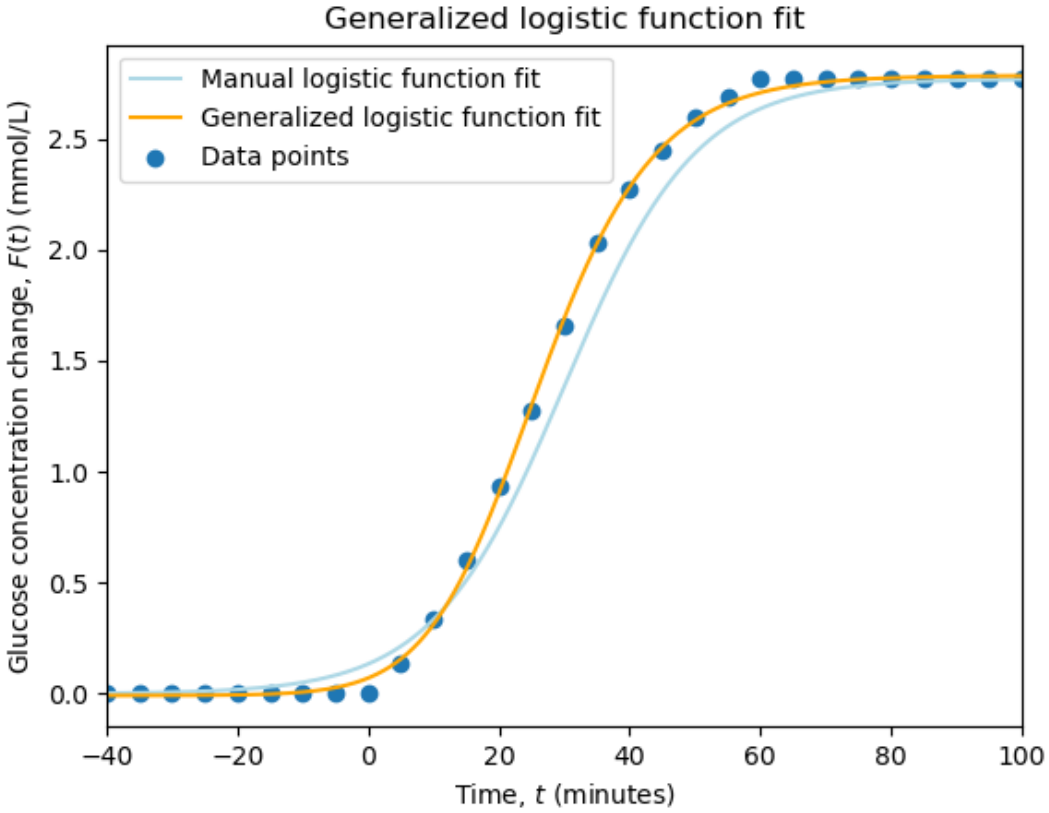


Figure 10 – Generalized logistic function fit with parameters based on the Levenberg-Marquardt algorithm (orange) with logistic function guess values as reference (light blue)

No elaborate testing is necessary as the purpose of the fit function (orange) was to follow the already averaged points as closely as possible, which it quite unmistakably does, as demonstrated in figure 10. The guess values (producing the function in light blue) were not too far off either but clearly skewed the curve slightly to the right. The additional parameters coupled with close guesses allowed the algorithm to make simultaneous adjustments and thus perform an extremely close fit for the function $F(t)$ with values approximated to three decimal places (for clarity):

$$F(t) = -0.008 + \frac{2.794 - (-0.008)}{(1.002 + 0.523e^{-0.101(t-25.393)})^{\frac{1}{0.572}}}$$

Modeling glucose concentration changes after injection of 1 unit of insulin

Figure 7 displays much more linear characteristics than figure 6 albeit with arguably a negligibly subtle kink in the middle. Due to the assumption of stable ends as demonstrated in figure 9, however, a simple linear function would not suffice as it would be continuous and thus never settle onto a certain level. Therefore, a piecewise linear function, which is essentially a collection of sub-functions only defined for certain domain intervals, is required to include the stable ends:

$$G(t) = \begin{cases} a(t + t_{tr}) + b & t < t_0 \\ c(t + t_{tr}) + d & t_0 \leq t \leq t_1 \\ e(t + t_{tr}) + h & t > t_1 \end{cases}$$

where G refers to the glucose concentration as a function of time, t_0 and t_1 are the breakpoints confining the sub-functions to the domain intervals up to, between and after them, a, c and e are the slopes of the linear sub-functions and b, d and h (instead of f or g not to be confused as a function) are their y-intercepts when the parameter for horizontal transformations, t_{tr}, is zero. Their values can be set by hand relatively easily:

$\boldsymbol{t_0 = 0}$ as this is defined to be the start time of the change.

$\boldsymbol{t_1 = 55}$ as this is the duration mean for the change rounded up to the closest 5.

$\boldsymbol{a} = \boldsymbol{e} = \mathbf{0}$ as the stable ends are horizontal lines with slopes of 0, meaning that they stay constant.

$\boldsymbol{b} = \mathbf{0}$ as this is defined to be the start magnitude of the change.

$\boldsymbol{h} \approx -\mathbf{1.686}$ as this is the average magnitude of the glucose concentrations at $t = 55$.

$\boldsymbol{c} \approx -\mathbf{0.0307}$ from calculating the overall slope over the interval $0 \leq t \leq 55$:

$$m = \frac{y_1 - y_0}{t_1 - t_0} = \frac{-1.686 - 0}{55 - 0} = \frac{-1.686}{55} = -0.0307$$

$\boldsymbol{d} = \mathbf{0}$ as the second sub-function starts from zero and hence its $(G(t) =)\ y$-intercept is zero.

$\boldsymbol{t_{tr}}$ **= 0** initially as this is only added as a parameter for the curve fitting algorithm.

As can be seen in the graph below as the light blue curve, these values produce a relatively good fit already, although the curve is shifted to the right in relation to most of the datapoints just a little bit, suggesting that another function allowing for curvature might be incrementally better. For the purposes of this exploration, however, a linear approximation will suffice as it is already so close. To ensure the best possible fit, however, I decided to run the algorithm detailed above only for t_{tr} and c with the variables defined above as initial guesses producing the orange curve:

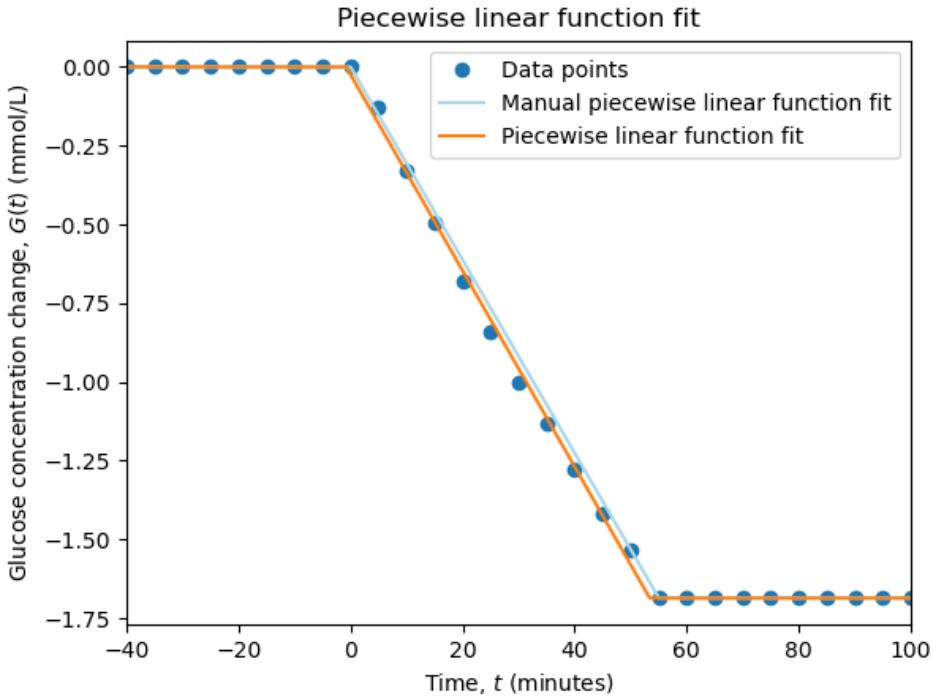


Figure 11 – Piecewise linear function fit with parameters based on the Levenberg-Marquardt algorithm with guess values defined above as reference

As a result, the curve is shifted left by the amount of $|t_{tr}| \approx 0.796$ and its gradient steepened slightly to -0.0311, necessitating the moving of the breakpoint, t_1 to $55 - 0.796 = 54.204$. This produces the fit function $G(t)$ with values approximated to three decimal places (for clarity):

$$G(t) = \begin{cases} 0 & t < 0 \\ -0.0311t & 0 \leq t \leq 54.204 \\ -1.686 & t > 54.204 \end{cases}$$

Finding the optimal timing of insulin injection to minimize glucose variability

As already discussed in the aim and approach section, the optimal timing for insulin injection to minimize glucose variability can be found as the horizontal transformation of the piecewise linear function representing the change in glucose concentration in response to insulin injection, that minimizes the total area under the curve produced by the function combined of the model functions. This area is a proxy for glucose variability due to being a measure of divergence from zero over time.

Combining the resulting functions from separately modeling both the glucose concentration as a function of time in response to consumption of 10.5 carbohydrates from 5 "Siripiri" tablets and injection of 1 unit (0.01ml) of Fiasp insulin produces the following combined function:

$$F(t) + G(t) = -0.008 + \frac{2.794 - (-0.008)}{(1.002 + 0.523e^{-0.101(t-25.393)})^{\frac{1}{0.572}}} + \begin{cases} 0 & t < 0 \\ -0.0311t & 0 \leq t \leq 54.204 \\ -1.686 & t > 54.204 \end{cases}$$

The total area under the curve produced by this function, visualized in the graph below, can be found from integrating the function between some limits to produce its definite integral. The limits of -115 and 115 were chosen based on the sum of the mean times, the durations of the changes, because with such extreme horizontal transformations the functions' changing parts no longer overlap and the area will therefore not decrease in either direction.

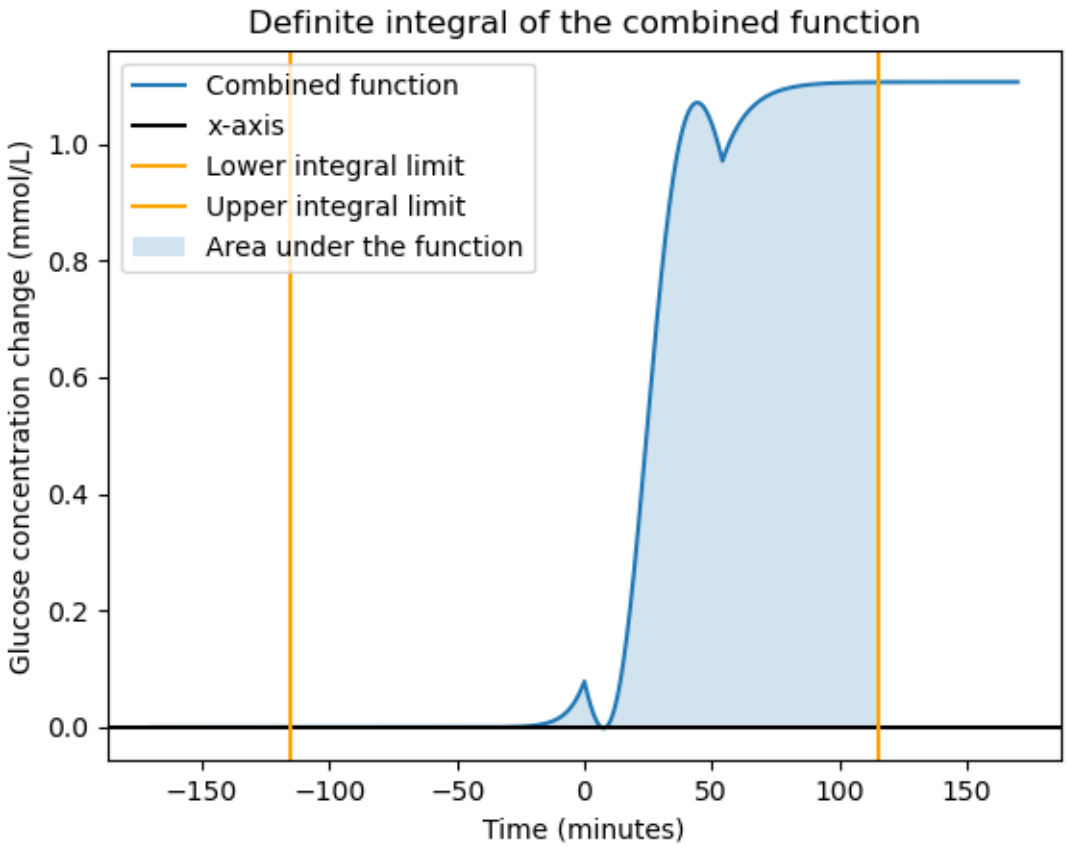


Figure 12 – Area under the combined function

This total area is given by the absolute value of the definite integral of the combined function:

$$\int_{-115}^{115} |F(t) + G(t + t_{tr})|\, dt$$

$$= \int_{-115}^{115} \left| -0.008 + \frac{2.794 - (-0.008)}{(1.002 + 0.523e^{-0.101(t-25.393)})^{\frac{1}{0.572}}} + \begin{cases} 0 & t + t_{tr} < 0 \\ -0.0311(t + t_{tr}) & 0 \leq t + t_{tr} \leq 54.204 \\ -1.686 & t + t_{tr} > 54.204 \end{cases} \right| dt$$

where t_{tr} is the horizontal transformation of the piecewise linear function.

For the horizontal transformations, I generated a total of 13 800 values corresponding to changes of one second on a symmetrical interval of $2 \cdot$ carbohydrate mean times $+ 4 \cdot$ insulin mean times (from -170 to 170 minutes), which I observed to be computationally feasible, albeit slow, but also to hold both the minimum and maximum areas, producing a beautiful scatter plot of the values of the definite integral, the area under the curve, against the horizontal transformation of the

piecewise linear function when areas of the combined function are computed for each horizontal transformation using SciPy's quad function for numerical integration, the method with the best balance of speed and accuracy for such a costly computation, even for a computer:

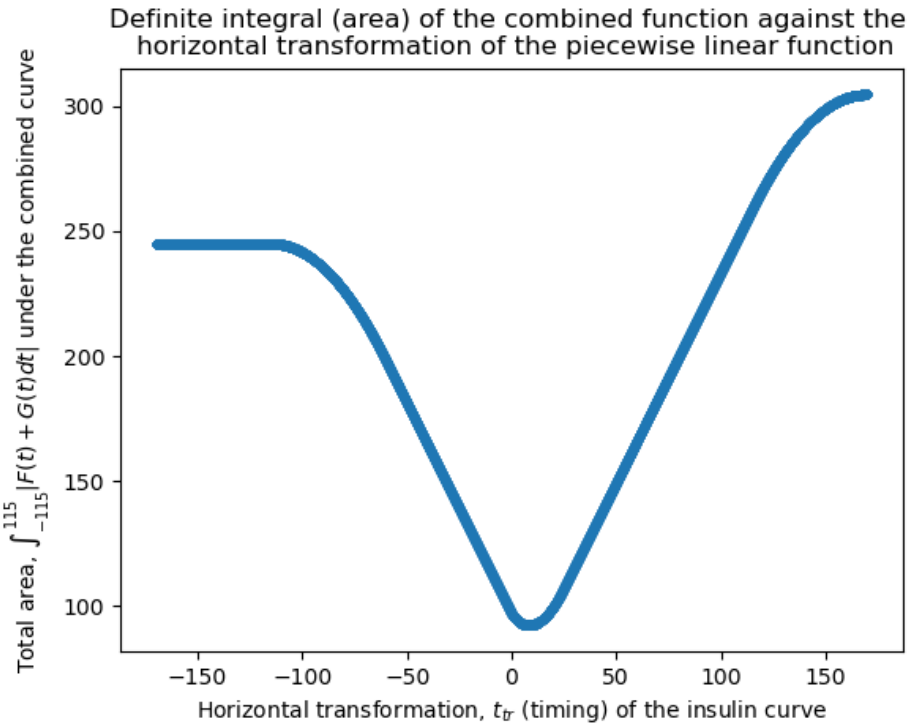


Figure 13 – Scatter graph of the total area under the combined function $F(t) + G(t + t_{tr})$ against the horizontal transformation of $G(t)$

This could most likely be modeled by either some piecewise function or by an asymmetric gaussian function achieved by combining a gaussian function with a logistic function: $ae^{\frac{(x-b)^2}{2e^2}} + \frac{L}{1+e^{k(x-x_0)}}$ which could then be differentiated in terms of *x* to find the minimum, $\frac{dy}{dx} = 0$ AND $\frac{d^2y}{dx^2} > 0$, but doing this with the dataset generated would not make much sense as the modeling itself would require knowing the minimum point, which can be found by just iterating through all the computed values and choosing the smallest one.

The minimum area is 92.309, which is achieved when $t_{tr} \approx 8.193$. As this value is positive, the curve resulting from insulin injection is transformed horizontally to the left by this amount, corresponding to a timing of the insulin of 8 minutes and 12 seconds before eating (as $0.193 \cdot 60 \approx 12$). The graph below shows the total area under the combined function before and after the transformation resulting in the minimum value of the definite integral:

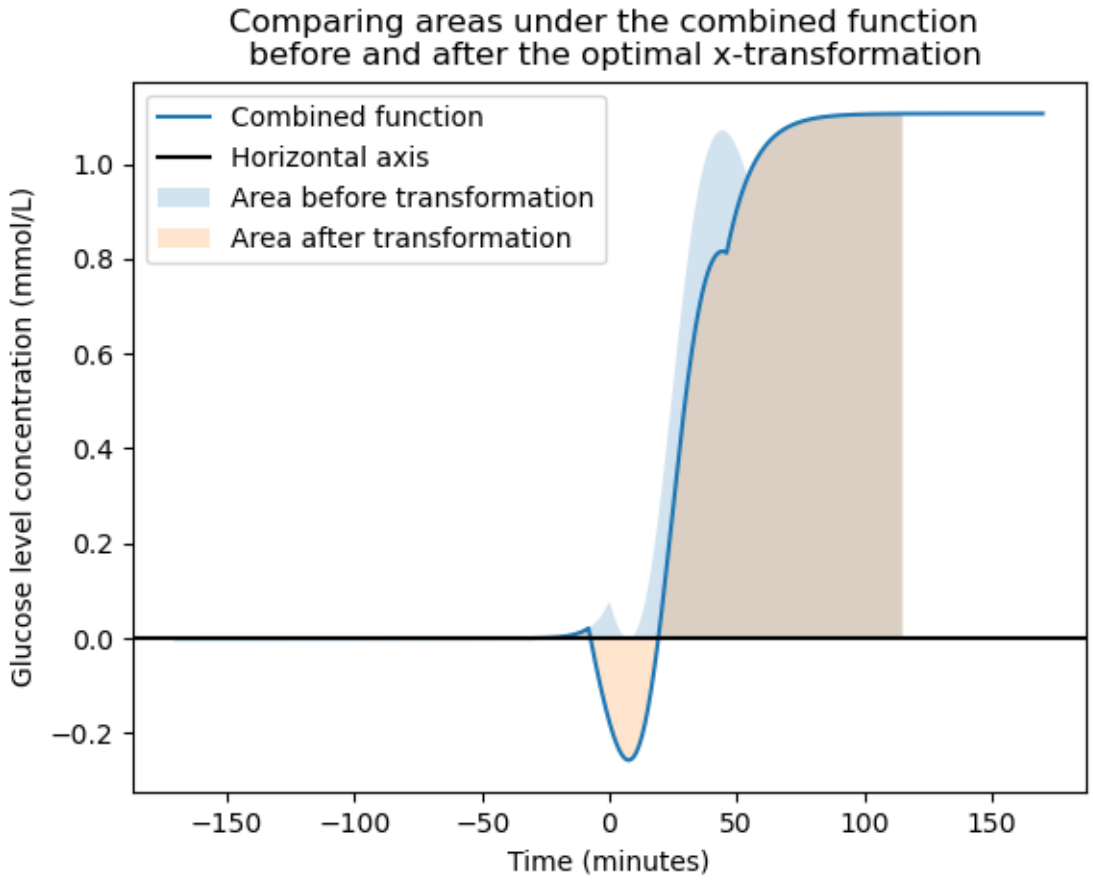


Figure 14 – Total area under the combined curve with $t_{tr} = 0$ in blue compared to the area with $t_{tr} = 8.193$ in orange

Analysis and evaluation

As can be seen in figure 14, the difference in area under the curve is rather small; in fact, only 96.928 (area with $t_{tr} = 0$) $-$ 92.309 (area with $t_{tr} \approx 8.193$) $=$ 4.619, implying that for convenience's sake, the entire 8-minute interval before eating is rather optimal. What is especially noticeable, however, is that whereas the area with $t_{tr} = 0$ in blue stays above the horizontal axis essentially throughout, the area with $t_{tr} = 8.193$ initially dips below the horizontal axis. To investigate this dip in more detail, we must look at a wider interval of horizontal transformations.

The maximum improvement from the area at $t_{tr} = 0$ is a 4.619 decrease. Suppose we can tolerate a similar worsening from the default starting point $t_{tr} = 0$, recommended by the nurses and doctors, usually even in the presence of the 20-minute leeway according to the manufacturer. This gives us a value of $96.928 + 4.619 =$ 101.547 for the area under the curve, which is then the threshold of

tolerance. The interval of horizontal transformations producing an area less than or equal to this is from -2.772 to 21.325 as can be seen in figure 13. Notably, the negative transformation (to the right) is much smaller than the 20 minutes promised by the manufacturer. Similarly, the transformation to the left is much more than expected, resulting in the dosing interval essentially being inverted. The graph on the right compares the areas and curves of maximum tolerance (orange and green) to the official dose timing interval recommended by the manufacturer (light blue and blue). It can be seen in figure 15 that something is not quite right with the dosing interval being inverted as the recommended interval would produce the largest area increases.

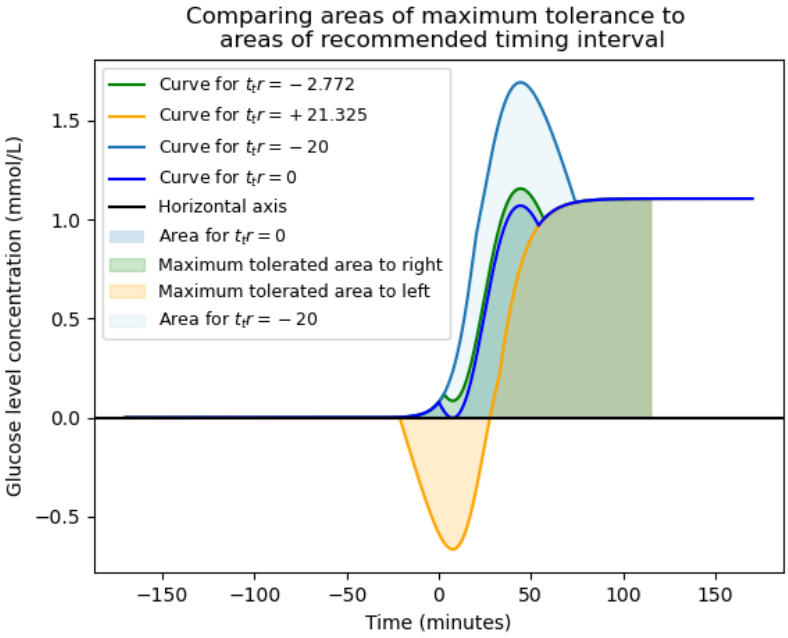


Figure 15 – Comparison of areas of maximum tolerance to areas of recommended timing interval

My method clearly favors transformations to the left even at the cost of dips below the horizontal axis, which leads to greater fluctuation of the glucose concentration below and above the initial concentration, which may lead the person interpreting the values in real time to make hasty correction decisions, leading to amplified fluctuation. This effect would be even worse for slower carbohydrates and larger amounts of insulin as the dip would be larger. This raises a question of the appropriateness of using the area under the curve as a measure of glucose variability. It expresses and minimizes the absolute divergence from zero but does so by increasing fluctuation, which would perhaps be a more relevant measure of variation based on this.

The fact that the horizontal transformations to the right produce much larger areas than those to the left, as seen in figure 15, might also signal of issues created by the min-max and max-min approaches. It may, for example, be the case that it takes the body longer to register carbohydrates than insulin, which would justify the 20-minute dosing window after eating. As this delay is entirely neglected in the models that only start from the first observed effect, these phenomena might go entirely unnoticed, therefore contributing to the error.

It can also be noticed that my $\frac{1}{10}$ ratio of insulin to carbohydrates might not quite hold as the level on which the combined curve settles, is higher than the level from where it starts. This is due to the difference in the means of magnitude of change, which are ≈ -2.2 for 1 unit of insulin and ≈ 3.0 for 10.5 carbohydrates from 5 'Siripiri" glucose tablets, which may be another factor for why the method favors the dips. The values are, however, brought down by the very small values that do not get filtered and the fact that there appear to be feedback loops that weaken the effect of insulin at higher concentrations and the effect of glucose at lower concentrations, which are the conditions under which these corrections are made, makes them somewhat unreliable.

Therefore, as it is difficult to better control the variables in data collection, improvements could be made in data selection and filtering by using more sophisticated statistical methods to, for example, carefully tighten the range of data selected. Delay to first effect must also be taken into account in future development by, for example, utilizing the stable ends to push the changing part of the function further so as to make 0 always denote the timing of an event rather than the first observed effect, or by throwing both assumptions out entirely and collecting much more data to hopefully even out the effect of other variables naturally. To better consider the up-and-down fluctuation in addition to just divergence over time from the horizontal, the object of minimization could also be, for example, the range of the combined function (maximum y-value – minimum y-value), which, with current assumptions, would give t_{tr} to be approximately 0.

Conclusion

Despite having to learn programming with python, using related software and libraries and understanding some new mathematics from scratch simultaneously to collecting data and finding some biases in my method and flaws in my assumptions in the end, I did manage to accomplish my aim to the extent it was defined. Neglecting the seconds as excessive precision, the optimal timing for insulin injection to minimize glucose level variability for me personally after a meal in ideal conditions is about 8 minutes before the meal. Due to the improvement being rather marginal compared to the recommendation, however, it might be safe to say that the optimum is rather an interval from no earlier than 8 minutes prior to beginning the meal, although this might differ for longer meals and slower carbohydrates. I might nevertheless take my finding to the next doctor's appointment and ask them what do they think of it for me. Although the result is too weak for anything really actionable for the various reasons mentioned above, it did importantly remind me of the fact that although fast, Fiasp should still be used as close to the beginning of a meal as possible, if not a few minutes prior, which is something I have not always followed closely enough.

Although in the end not quite as definitive and actionable as I would like, this works as great basis for further development and learning. In the next iteration, I will aim to cut down the assumptions, try the alternative approach of minimizing the range and find ways for the method to work with variable amounts of carbohydrates and insulin. In the course of this exploration,however, I learned a lot of mathematics, programming and even writing, especially modeling, data processing and visualization and got very interested in their various aspects and forms. It was a very humbling experience, once again returning me to the beginner's mindset, as I basically was one in all aspects of the exploration. It showed me how much there is still to learn, understand and discover while still showing me a glimpse of the tremendous power and potential mathematics has to offer for all problem solving from the mundane to life-altering, leaving me inspired to learn and explore a lot more.

Bibliography

Brilliant Worldwide, Inc. (2021). *Interquartile Range (IQR) | Brilliant Math & Science Wiki*. Brilliant. Retrieved December 7, 2021, from https://brilliant.org/wiki/data-interquartile-range/

Danne, T. (2017, December 1). *International Consensus on Use of Continuous Glucose Monitoring*. Diabetes Care. Retrieved December 5, 2021, from https://care.diabetesjournals.org/content/40/12/1631

Dexcom, Inc. (2021). *Dexcom CGM, Continuous Glucose Monitor Technology*. Dexcom. Retrieved December 5, 2021, from https://provider.dexcom.com/dexcom-cgm

Novo Nordisk. (2019, December). *Fiasp insulin aspartion injection 100 units/mL*. https://www.novo-pi.com/fiasp.pdf

The SciPy community. (2021). *scipy.optimize.curve_fit — SciPy v1.7.1 Manual*. SciPy. Retrieved December 8, 2021, from https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.curve_fit.html

The SciPy community. (2021). *scipy.optimize.least_squares — SciPy v1.7.1 Manual*. SciPy. Retrieved December 8, 2021, from https://docs.scipy.org/doc/scipy/reference/generated/scipy.optimize.least_squares.html#scipy.optimize.least_squares

Wikipedia contributors. (2021, November 15). *Levenberg–Marquardt algorithm*. Wikipedia. Retrieved December 8, 2021, from https://en.wikipedia.org/wiki/Levenberg%E2%80%93Marquardt_algorithm

Wikipedia contributors. (2021, November 24). *Generalised logistic function*. Wikipedia. Retrieved December 8, 2021, from https://en.wikipedia.org/wiki/Generalised_logistic_function

Appendices