Project Milestone 4 – Technical Brief

To: Avery D. Lion, President of NaturalCatalysts Inc.
From: Team 004-19 — Nathan Yao, Justin Schwartz, Patrick Mullen, Kaushik Karthikeyan
RE: Technical Brief: Analysis of Final Results and Recommendations
Date: April 30, 2021

Introduction

NaturalCatalysts has collected and provided us with 100 kinetic enzyme tests for their newest five next-generation enzymes and requested that we construct an algorithm in MATLAB to assess each enzyme's performance, create a detailed analysis and summary of the data and provide an error analysis that characterizes the approach to determine enzyme performance. We must draft a recommendation about what NaturalCatalysts can honestly and ethically claim to its customers about their new enzymes.

In our MATLAB program, each experiment on the CSV file which NaturalCatalysts has provided is passed into an algorithm that finds v_0 by first applying a moving-average data denoiser and a loop that uses least-squares regression to find a good-fitting second order polynomial model for a given data set. From the model, we are able to find the values of v_0 at various concentrations for an enzyme, which are then passed into a function that uses the Eadie-Hofstee equation to find K_m and V_{max} for the enzyme.

We made 3 critical decisions to improve parameter accuracy. The first decision we made was to use a quadratic polynomial instead of a straight line (as we did initially) to model a data set when finding the v_0 value. This is because the front part of a [P] (product concentration) vs time graph looks to have a parabolic-shape, and the linear model was not reliable in yielding consistent results. Because we are using least-squares regression and a quadratic polynomial can model the curve better than a straight line, this would allow us to further raise the r^2 threshold which boosts the overall accuracy of the model which we are taking a derivative of (and evaluating the derivative at an initial point in time to find v_0).

The second decision was to use the Eadie-Hofstee method instead of the Lineweaver-Burk or Hanes Woolf method to find the parameters K_m and V_{max}. This is because the Eadie-Hofstee method is easy to implement compared to the other approaches, does not have severe spacing issues like the Lineweaver-Burk method, and does not take a reciprocal of v_0, meaning it does not magnify small errors in the calculation of v_0 (and therefore produces more accurate results for K_m and V_{max}) (The Equations of Enzyme Kinetics, 2021).

The third decision we made was to find v_0 separately for an experiment and its duplicate. In our first design, we combined an experiment and its duplicate at each moment of time as an attempt to reduce the noise by taking an arithmetic average. But based on feedback and technical review, we realized that a test and its duplicate may be offset by a few seconds (as data collection is not perfect), and that taking an average between them would skew the results. Now instead of averaging a data set and obtaining only one v_0 value, we use both v_0 values at some given initial substrate concentration to find an even more accurate Michaelis-Menten model.

Parameter Identification Procedure

Our final algorithm sets out to accomplish the task of discovering the following three parameters. The slope of the tangent line of the product concentration versus vs time curve as time approaches zero, also known as v_0. The substrate concentration that allows for half of the maximum reaction velocity for the reaction velocity vs substrate concentration curve, also known as K_m. And the maximum reaction velocity reached as substrate concentration increases, also known as V_{max}.

Our program starts in the main function, which is essentially a large nested loop that continues to parse in the [P] vs time data for each substrate concentration for each enzyme into a function which finds v_0 for a

given data set. The v_0 function takes the data set and attempts to model a quadratic curve (forced through the origin) on the given data set. Initially, by least-squares regression, the model will be of very poor quality, because as time increases, the [P] vs time data set levels eventually level out (because there are no more products to produce by the enzymes). A loop is implemented to continue removing one data point from the end of the set until the quadratic curve fits the data with an r^2 value of above 0.995, which indicates a near-perfect fit. After obtaining the quadratic model, we take a manual derivative using differentiation power rule, and evaluate the derivative at an initial point in time (e.g., time at 1 second). This result yields v_0, the slope at the beginning of the [P] vs time curve. An experiment and its duplicate are parsed separately, yielding their individual v_0 values.

Knowing v_0 values and their corresponding initial substrate concentrations allows us to find K_m and V_{max} for the given enzyme by using the Eadie-Hofstee linearization technique. The Eadie-Hofstee equation takes the form $v_0 = V_{max} - K_m(v_0/[S])$. Where v_0/[S] is the dependent variable, v_0 is the independent variable, and the parameters K_m and V_{max} are found by taking the constants (negative K_m is the slope and V_{max} is the vertical axis intercept). Again, we chose this technique because of its non-severe spacing issues, ease of code implementation, and because it does not magnify errors in v_0 (The Equations of Enzyme Kinetics, 2021).

Results

Again, the substrate concentration that allows for half of the maximum reaction velocity for the reaction velocity vs substrate concentration curve is known as K_m. The maximum reaction velocity reached as substrate concentration increases is known as V_{max}. As seen in Table 1 in Appendix: Figures and Tables, the sum of squared errors (SSE, which is the difference between the actual and predicted values) for the Michaelis-Menten models of all NextGen enzymes is very small (ranging from 0.00003 to 0.00109), which indicates that our method of finding v_0 is reliable in producing consistent and accurate results. The high quality of our results can be observed when analyzing our many figures, such as Tables 2 and 3 in Appendix, which show a well-fitting Michaelis-Menten model and a precise v_0 line. To view additional figures for all NextGen enzymes, run our MATLAB main function.

Interpretation

The error of our model ranges from $3x10^{-5}$ $(\mu M/s)^2$ to $2.67x10^{-3}$ $(\mu M/s)^2$, and it comes from our model line we use to determine v_0 not being a perfect fit to the data. Our method of using a This would cause inaccuracies in our v_0 calculations, despite having a high threshold for the goodness of fit evaluation we calculate for the quadratic regression. We believe our model on NextGen enzymes is accurate because of our tests on PGO-X50 benchmarking data. Our algorithm's calculations are fairly accurate to the benchmarking data for PGO-X50. Our algorithm determined v_0's, K_m, and V_{max} parameters to have a sum of squares error of 0.0144 $(\mu M/s)^2$, as opposed to the benchmarking data having a sum of squares error of 0.0048 $(\mu M/s)^2$.

The manufacturing performance is generally measured by cost, quality, flexibility, and dependability. Our model fares well in these factors. In terms of quality, our model does a fairly good job. Although the results for a few variables might be slightly inaccurate, overall the model paints an accurate picture of the data. Considering the fact that we have not hard coded our algorithms, it is also relatively flexible. In terms of manufacturing consistency, the dataset we received was not greatly consistent. It had a fair amount of outliers. However, our model ensured that we took these outliers into account and hence provides a consistent output.

References

10.2: The Equations of Enzyme Kinetics. (2021). Retrieved 27 April 2021, from https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Map%3A_Physical_Chemistry_for_the_Biosciences_(Chang)/10%3A_Enzyme_Kinetics/10.2%3A_The_Equations_of_Enzyme_Kinetics

"The Effect of Substrate Concentration on Enzyme Activity." *London's Global University*, www.ucl.ac.uk/~ucbcdab/enzass/substrate.htm

Appendix: Figures and Tables

Table 1

Results for V_{max} and K_m From Our Final Algorithm

Enzyme	Final Algorithm		
	Enzyme Parameters		SSE $(\mu M/s)^2$
	V_{max} (μM/s)	K_m (μM)	
NextGen-A	0.845	138.360	0.00062
NextGen-B	0.766	329.512	0.00267
NextGen-C	1.089	174.560	0.00003
NextGen-D	1.438	278.806	0.00006
NextGen-E	1.476	148.534	0.00109

Table 2

Comparing PGO-X50 Benchmarking Data with Our Algorithm

Parameter (μM/s)	PGO-X50 Reference Values	M4_Algorithm
v_{0_1}	0.025	0.026
v_{0_2}	0.049	0.053
v_{0_3}	0.099	0.104
v_{0_4}	0.176	0.182
v_{0_5}	0.329	0.335
v_{0_6}	0.563	0.548
v_{0_7}	0.874	0.800
v_{0_8}	1.192	1.082
v_{0_9}	1.361	1.216
$v_{0_{10}}$	1.603	1.492
V_{max}	1.806	1.539
K_m (μM)	269.74	217.374
SSE $(\mu M/s)^2$	0.0048	0.01442

Figure 1

Example of Michaelis Menten Plot for NextGen-A

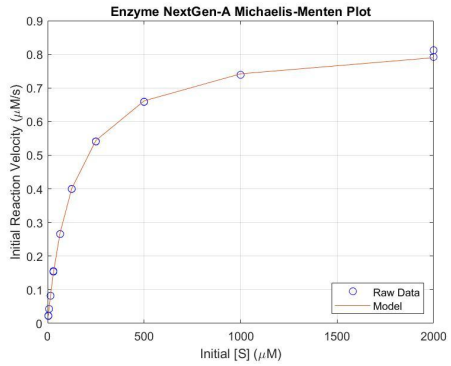

Figure 3

Example of [P] vs Time Curve and Corresponding v_0 Lines