The Journal of Open Source Software

KAT: A Python-based GUI for the analysis of enzyme kinetic data using both classical and complex models.

Evan R. Jones [1]

1 Department of Chemistry and Biochemistry, University of Oklahoma, United States

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor:

Submitted: 04 August 2025
Published: unpublished

Summary

Kinetics serves as the basis for understanding the key properties of an enzyme. These apparent properties can give vital information regarding interactions with substrate, energetics or speed of the reaction, and potential allosteric effects. These parameters often help guide novel therapeutic discovery and development. Often, kinetic assays are the simplest way to ascertain these key details of function, allowing properties to be compared between homologous and mutant enzymes. Kinetic data processing can be swiftly automated and fitted to a number of kinetic models, a necessary step for understanding enzyme mechanisms. A fully-functional GUI that accepts raw data and outputs both kinetic parameters and a graph of the fit not only significantly accelerates data analysis, but also readily normalizes treatment of data across replicas and experiments. The accessibility of this GUI can also facilitate more thorough examination of kinetic models, potentially revealing previously overlooked features.

Statement of need

KAT is a toolkit dedicated to parsing both fluorescence and absorbance kinetic data and fitting the data to several classical (Michaelis-Menten and Hill) (Goutelle et al., 2008; Johnson & Goody, 2011) and complex (Monod-Wyman-Changeux and Koshland-Nemethy-Filmer) (Koshland et al., 1966; Monod et al., 1965) models. A Python-based GUI allows for simple input of necessary data: CSV file with fluorescence or absorbance data, the substrate information (number of substrate concentrations, dilution factor, and maximum concentration), and a time window within which to calculate the velocity data. A built-in function automatically calculates the linear range of the assay data, which is a necessary parameter for many steady-state models. This is calculated through analysis of slope values less than 5% different on average, with variation weighted by signal strength. Once the user has provided the necessary data, s/he can easily fit the data to multiple models, allowing for the comparison of fits and kinetic parameters. Each of these models can also be used to analyze either replicate data or data arising from several mutations. These replica fitting functions output either averages and standard deviations for each kinetic parameter or side-by-side analysis for mutant data sets.

While many other tools effectively fit Michaelis-Menten kinetics to a data set, fewer fit data to the non-linear form of the Hill Equation, which adds the extra Hill coefficient parameter. Further, software like EnzFitter require a license and do not have complex models built-in, requiring the user to input complex algebraic equations by hand.(Leatherbarrow, 1988) Free software like EKA focus on enzyme inhibition, along with classical models, and serve as a teaching aid with its online-only implementation. (Mak et al., 2024) KAT integrates complex models directly and outputs easily-modifiable SVG graphs, as well as typical PNG-formatted graphs.

KAT utilizes both numerical solving (classical models) and optimization (complex models) using the lowest residual square sum for the determination of kinetic parameters. For fitting to

classical models, a standard 12 substrate concentrations is sufficient to provide substantial confidence in the model fitting; however, the use of complex models that solve for up to 6 parameters are challenging to fit when using under ~30 substrate concentrations. Therefore, several statistical techiniques have been implemented within KAT to test the confidence interval of the model fit and each parameter. Should one parameter fall outside these intervals, a warning message is displayed in the GUI alerting the user to potential poor confidence.

With an efficient method of analyzing raw enzyme kinetic data using a simple GUI, KAT will undoubtedly be useful for a wide-range of enzyme types and significantly standardize the fitting of data to both classical and complex kinetic models. The free and accessible nature of KAT empowers non-experts to make use of these informative characterization tools for studying enzymes, including in classroom settings.

Mathematical Basis

Kinetic Models

Classical

Based on the simple model of enzyme catalysis, the Michaelis-Menten equation identifies a V_{max} and a K_M that define the maximum velocity and the Michaelis constant, which correlates to suitability of a given substrate for catalytic action by an enzyme, respectively.(Johnson & Goody, 2011) The Michaelis-Menten equation assumes that there are three states that are discreet and independent.

$$E + S \longleftrightarrow ES \longrightarrow E + P \tag{1}$$

where E is the enzyme, S is the substrate, and P is the product formed. After rearranging for the rate-constants that govern each transition, one solves for the complete Michaelis-Menten equation, where [S] is the substrate concentration and v is the initial steady-state velocity.

$$v = \frac{V_{max}[S]}{K_M + [S]} \tag{2}$$

The Hill equation is an extension of the Michaelis-Menten equation that accounts for cooperativity, which is often the result of allostery within the enzyme and is the effect where binding of one substrate molecule accelerates the binding of another. The Hill coefficient, or n, is a measure of the substrate dependence of the reaction progress, where values greater than 1 exhibit positive coorperativity (*i.e.* stronger binding of subsequent ligands) and values less than 1 exhibit negative cooperativity (*i.e* weaker binding of subsequent ligands). (Goutelle et al., 2008)

$$v = \frac{V_{max}[S]^n}{K_M^n + [S]^n} \tag{3}$$

Complex

While the Michaelis-Menten and Hill models often explain the majority of enzymes, the Hill equation in particular is limited to cooperativity within the enzyme, though allostery can also impact the catalysis steps in addition to cooperative substrate interactions. Monod, Wyman, and Changeux further expanded upon the equation and introduced two stages for an allosterically-regulated enzyme: a tensed or "T" state, where binding of substrate and catalysis is limited, and a relaxed or "R" state, where both are accelerated. These additional states each have a V and a K parameter. Further, there are two additional parameters (for a total of

6): L_0, or the cooperativity coefficient as a ratio of the enzyme in the T state vs. the R state, and n, or typically the number of allosteric sites in the enzyme.(Monod et al., 1965) These parameters combine to give the following Monod-Wyman-Changeux equation:

$$v = \frac{V_T L_0 (1 + \frac{[S]}{K_T})^n + V_R (1 + \frac{[S]}{K_R})^n}{L_0 (1 + \frac{[S]}{K_T})^n + (1 + \frac{[S]}{K_R})^n} \tag{4}$$

Although the Monod-Wyman-Changeux model accurately predicts the transitions between the T and R states, the model requires that the T and R states exist in discreet environments, where part of the enzyme cannot exist in both states at once. Since not all enzymes exhibit a complete transition from T to R states at once, Koshland, Nemethy, and Filmer developed a model that accounts for enzymes that exist in states T and R simultaneously by introducing a γ term. This term provides the basis for cooperativity in the enzyme. Additionally, the Koshland-Nemethy-Filmer model utilizes a term for each active site of the enzyme (denoted as i here).(Koshland et al., 1966) Upon summing all possible enzyme states (from unbound to bound) in each active site, i:

$$v = \frac{E_{total} \sum_{i=0}^{j} k_i \binom{j}{i} (\frac{[S]}{K_d})^i}{\sum_{i=0}^{j} \binom{j}{i} (\frac{[S]}{K_d})^i} \tag{5}$$

where j is the total number of active sites, E_{total} can be built into a V_{max} term, and $k_i = k_{basal} + (V_{max} - k_{basal})(\frac{i}{j})^\gamma$.

Due to the large number of parameters, these complex equations are difficult to assess the confidence of fit given an estimated starting guess ("Best-Fit") when the number of substrate concentrations is less than 30. Therefore, Cross-Validation of the solved parameters using the KFold technique with 10 splits is used.(Pedregosa et al., 2011) Further, if the number of substrate concentrations is below 30, Bayesian bootstrapping is implemented to assess the 99% confidence intervals of each parameter, and the "best-fit" data is tested to be within these confidence intervals. If a best-fit parameter falls outside of the 99% confidence interval, the cross-validation parameters are provided instead of "best-fit." Otherwise, the "best-fit" values are provided.

Model Fitting

Using Classical Models

For fitting data to a classical model, a simple minimization of residual sum of squares is performed using a data-driven starting guess. This starting guess uses an average of the three largest velocity values for V_{max} and the substrate concentration at half V_{max} as a guess for K_M. If the Hill model is used, a Hill coefficient of 2 is used as the starting guess for n. `sympy.nsolve` is used to numerically solve the three partial derivatives, $\frac{\partial Q}{\partial V_{max}}$, $\frac{\partial Q}{\partial K_M}$, $\frac{\partial Q}{\partial n}$, of the following equation:(Meurer et al., 2017)

$$Q = \sum_{s=1}^{S} (V_{data} - \frac{V_{max} s}{K_M + s})^2 \tag{6}$$

for the Micaelis-Menten model, or

$$Q = \sum_{s=1}^{S} (V_{data} - \frac{V_{max} s^n}{K_M^n + s^n})^2 \tag{7}$$

for the Hill model. High confidence in the initial fit results from the low number of parameters verses substrate concentrations tested combined with robust empirical estimates from the raw data.

Using Complex Models

Unlike fitting the classical kinetics models, the complex models have too many parameters to effiently and accurately numerically solve the partial derivatives of the extended residual sums. Therefore, `scipy.optimize.minimize` is implemented to minimize the loss function (defined similarly as above in (6) and (7)). The minimization is bound by typical biological constraints, where V_T, V_R, K_T, and K_R and bound between 0 and 1,000, L_0 is bound between 0.001 and 500, and N is bound between 0.5 and 14 for the Monod-Wyman-Changeux equation. Similarly for the Koshland-Nemethy-Filmer model, V_{max}, K_d, and k_{basal} are bound between 0 and 10,000 and γ is bound between -50 and 50. Due to the unstable structure of fitting relatively few data points to complex models, `scipy.optimize.differential_evolution` is first used to help identify global minima. (Virtanen et al., 2020)

Acknowledgements

I acknowledge contributions from Dr. Christina Bourne and the University of Oklahoma Department of Chemistry and Biochemistry for their resources.

References

Goutelle, S., Maurin, M., Rougier, F., Barbaut, X., Bourguignon, L., Ducher, M., & Maire, P. (2008). The hill equation: A review of its capabilities in pharmacological modelling. *Fundamental & Clinical Pharmacology*. https://doi.org/10.1111/j.1472-8206.2008.00633.x

Johnson, K., & Goody, R. (2011). The Original Michaelis Constant: Translation of the 1913 Michaelis-Menten Paper. *Biochemistry*. https://doi.org/10.1021/bi201284u

Koshland, D. E., Nemethy, G., & Filmer, D. (1966). Comparison of experimental binding data and theoretical models in proteins containing subunits. *Biochemisty*. https://doi.org/10.1021/bi00865a047

Leatherbarrow, R. (1988). Enzfitter: A non-linear regression data analysis program for IBM PC. *Journal of American Chemical Society*.

Mak, D. A., Dunn, S., Coombes, D., Carere, C. R., Allison, J. R., Nock, V., Hudson, A. O., & Dobson, R. C. J. (2024). Enzyme kinetics analysis: An online tool for analyzing enzyme initial rate data and teaching enzyme kinetics. *Biochemistry and Molecular Biology Education*, *52*(3), 348–358. https://doi.org/https://doi.org/10.1002/bmb.21823

Meurer, A., Smith, C. P., Paprocki, M., Čertík, O., Kirpichev, S. B., Rocklin, M., Kumar, A., Ivanov, S., Moore, J. K., Singh, S., Rathnayake, T., Vig, S., Granger, B. E., Muller, R. P., Bonazzi, F., Gupta, H., Vats, S., Johansson, F., Pedregosa, F., … Scopatz, A. (2017). SymPy: Symbolic computing in Python. *PeerJ Computer Science*, *3*, e103. https://doi.org/10.7717/peerj-cs.103

Monod, J., Wyman, J., & Changeux, J. P. (1965). ON THE NATURE OF ALLOSTERIC TRANSITIONS: A PLAUSIBLE MODEL. *Journal of Molecular Biology*. https://doi.org/10.1016/s0022-2836(65)80285-6

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830.

Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., Burovski, E., Peterson, P., Weckesser, W., Bright, J., Van Der Walt, S. J., Brett, M., Wilson, J., Millman, K. J., Mayorov, N., Nelson, A. R. J., Jones, E., Kern, R., Larson, E., ... Vázquez-Baeza, Y. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nat Methods*, *17*(3), 261–272. https://doi.org/10.1038/s41592-019-0686-2