Statistical analysis of Michaelis constant data from online databases

Teddy Groves and Areti Tsigkinopoulou

February 1, 2022

Contents

1 Introduction **2**

2 State of the art **3**

3 Method **4**
3.1 Data fetching 4
3.2 Data processing 4
3.2.1 Filtering 4
3.2.2 Grouping 5
3.3 Statistical model 5
3.4 Model validation 8
3.4.1 Computation 9
3.4.2 Comparison models 9
3.4.3 Graphical posterior predictive checks 10
3.4.4 **TODO** Cross validation 12
3.5 **DONE** Web app 12

4 Results **12**
4.1 Marginal distributions of interesting parameters 12
4.2 Comparison of estimated Kms with physiological metabolite concentrations 14
4.3 NADH vs NADPH 15
4.4 **TODO** Cofactor vs substrate 16
4.5 Which results to use? 16

5 TODO Case studies **17**

6 TODO Discussion **17**

7 References **17**

Abstract

We fit a range of Bayesian regression models to analyse data from the BRENDA and SABIO-RK databases pertaining to Michalis constants or Km parameters. We report the results and provide tools for reproducing them independently and using them in a systems biology workflow. We illustrate the intended use of our work with case studies.

1 Introduction

The choice of appropriate parameters for biological models has been a recurring issue for computational biologists in the recent years. As computational models become increasingly prominent in the current research [1]–[3] and lead the way to biological breakthroughs, the use of high quality parameter values is becoming essential. In view of these developments, there have been various efforts to address this issue by employing different strategies. Although the most popular approach is the estimation of parameters through optimization methods [4], [5], ensemble modelling strategies are rapidly gaining ground in the field of systems biology [6]–[9] along with the development of tools to define informative parameter priors [10], [11]. At the same time, machine learning methods have recently been gaining popularity as a tool for interpreting the large amount of existing biological data [12]–[14] and have also been employed for the prediction of parameter values (mostly Km values) in biological models [15]–[17].

By taking into the account the existing issues with model parameterisation and in an effort to address them, we have previously developed a standardized protocol for the definition of appropriate parameter lognormal distributions based on information retrieved from different sources and in accordance with the modeller's beliefs about the reliability of each experimental value [18]. This protocol was a significant step towards making ensemble modelling more accessible and promoting interdisciplinary collaborations. However, there were some remaining issues to be addressed. Firstly, the protocol should take into account the surrounding parameter landscape (phylogenetically related species, enzyme sub-classes etc.) in order to avoid having the prior distribution being too narrow or focused only on one particular area. Furthermore, it should avoid over-reliance on a limited set of experimental reports.

In order to resolve these problems and additionally make the protocol fully automated, we devised a novel method for generating appropriate prior distributions for kinetic parameters. Using previous work by Liebermeister and Klipp as a starting point [19], we developed a hierarchical Bayesian regression model in order to perform a statistical analysis of parameter reports from the BRENDA [20] and SABIO-RK [21] databases. The results of this analysis can be used in our existing protocol to rationalise parameter weights, or as part of another protocol. In order to make it easier to access and review the information in BRENDA, we also provided an online interface through which our model's results can easily be reviewed and extracted.

2 State of the art

The problem of modelling kinetic parameter data from online databases has previously been addressed in several studies.

[19] models logarithmic-scale measurements y_{ijk} of a Km value for substrate i, ec number j and organism k using the following ANOVA-style regression model:

$$y_i jk \sim N(\mu_i + \alpha_i j + \beta_i k, \sigma) \tag{1}$$

In this expression the term μ represents substrate-specific effects, whereas the terms α and β respectively represent substrate-ec number and substrate-organism interaction effects. σ represents the accuracy of the measurement apparatus.

The authors fit this measurement model to a subset of BRENDA data using maximum likelihood estimation, obtain out of sample predictions using leave-one-out cross validation and investigate patterns in the results.

This approach is limited by inability to incorporate prior information about the plausible values of the main effects or their general trends. In addition, the analysis produces point estimates: it does not attempt to fully capture the available information about Kms. This issue is particularly pertinent for systems biologists who need to construct informative prior distributions for kinetic models. For these users it is more important to know which possible Km values are ruled out by the data in BRENDA than it is to know which make that data most likely. Below we show that both of these issues with the

approach in [19] can be addressed by incorporating a hierarchical Bayesian component.

[16] and [17] predict kinetic parameters from BRENDA and other sources by taking into account protein and substrate structure information using neural networks. Exploiting this information allows for improved out of sample predictive performance compared to models that do not use it.

However, this approach does not avoid the issues with lack of prior information and inaptness for downstream prior modelling that we highlight above. In addition, the need for information about protein structure limits the amount of kinetic parameters for which predictions can be obtained. Again taking the point of view of a systems biologist attempting to construct informative prior distributions, this is a severe problem as coverage is at least as important a consideration for this application as precision.

3 Method

3.1 Data fetching

We fetched data from the SABIO-RK [21] and BRENDA [22] databases using their publicly available APIs. We also fetched a list of all EC numbers from the Expasy database [23]. In the project repository, see the script `fetch_data.py` and the library file `fetching.py` for code used to fetch data and the directory `data/raw/` for the results.

In total we fetched EXACT NUMBER HERE raw BRENDA Km reports, EXACT NUMBER HERE SABIO-RK Km reports and EXACT NUMBER HERE Expasy EC numbers.

3.2 Data processing

We made several significant data processing choices. See the library file `data_preparation.py` for code used to implement these choices.

3.2.1 Filtering

For each dataset and kinetic parameter, we removed all reports which failed to satisfy any of the following conditions:

- The kinetic parameter value must be a number
- The literature reference must not be missing

- The substrate must be catalysed naturally by the enzyme
- The enzyme must be from a wild organism
- The temperature, if recorded, must be between 10 and 45 degrees C
- The pH, if recorded, must be between 5 and 9
- The organism must be have data from at least 50 separate study/biology (i.e. organism:substrate:ec4 for BRENDA or organism:substrate:enzyme for SABIO-RK) combinations that satisfy all the other conditions.

3.2.2 Grouping

Instead of modelling reports directly, we chose to group together reports with the same biology and study, treating the median log-scale km as a single observation. We took this decision because of the presence in both datasets of different kinds of study. In some cases - presumably when the aim of a study was to discover the sensitivity of a kinetic parameter to changes in conditions - many reports with the same enzyme, organism, substrate and study are available, with a range of different kinetic parameter values and different experimental conditions recorded in the `commentary` field. In other cases a study will report only a single value for one kinetic parameter.

Due to this discrepancy it seemed wrong to treat reports from better populated studies as equivalent to reports from more concise studies. While taking the median for a given study/biology combination before modelling destroys information, we judged that it would lead to more realistic results than treating each report as an observation, especially since we chose not to attempt to model the effects of experimental conditions.

3.3 Statistical model

We used a Bayesian hierarchical regression model to describe all data. Since Km parameters are constrained to be positive, we modelled them on natural logarithmic scale, using the same approach taken in [19]. Our model includes a global mean parameter μ, hierarchical substrate-specific intercept parameters a^{sub} and hierarchical intercept parameters $a^{enz:sub}$, $a^{ec4:sub}$ and $a^{org:sub}$ specific to interactions of substrate and enzyme, ec4 number and organism respectively. In addition we used a student-T measurement model with latent standard deviation and degrees of freedom σ and ν.

We chose assigned semi-informative prior distributions based on the pre-

experimental information. In particular, the prior for the measurement distribution degrees of freedom parameter ν follows the recommendation in [24] and is truncated below at one, reflecting our view that the measurement distribution should not be excessively heavy-tailed.

The full model specification in tilde notation is shown below. In these lines the symbol $should be understood as meaning" has the distribution".ST$, N and Γ represent the student-t, normal and gamma distributions respectively. Square brackets represent truncated probability distributions. For example $X\tilde{N}(0,1)[0,\infty]$ means that the variable X has a normal distribution truncated below at zero, also known as a "half-normal" distribution.

$$
\begin{aligned}
\ln y &\sim ST(\nu, \hat{\ln y}, \sigma) \\
\hat{\ln y} &= \begin{cases} \textbf{enz available :} \ \mu + a^{sub} + a^{org:sub} + a^{ec4:sub} + a^{enz:sub} \\ \textbf{otherwise :} \ \mu + a^{sub} + a^{org:sub} + a^{ec4:sub} \end{cases} \\
a^{sub} &\sim N(0, \tau^{sub}) \\
a^{org:sub} &\sim N(0, \tau^{org:sub}) \\
a^{ec4:sub} &\sim N(0, \tau^{ec4:sub}) \\
a^{enz:sub} &\sim N(0, \tau^{enz:sub}) \\
\mu &\sim N(-1, 2) \\
\nu &\sim \Gamma(2, 0.1)[1, \infty] \\
\sigma &\sim N(0, 2)[0, \infty] \\
\tau^{sub} &\sim N(0, 1)[0, \infty] \\
\tau^{org:sub} &\sim N(0, 1)[0, \infty] \\
\tau^{ec4:sub} &\sim N(0, 1)[0, \infty] \\
\tau^{enz:sub} &\sim N(0, 1)[0, \infty]
\end{aligned}
$$

This model can be expressed as the following program in the probabilistic programming language Stan program [25]:

```
/* Extends the BLK model for more specific data from the SABIO-rk database */

data {
  int<lower=1> N;
```

```
  int<lower=1> N_train;
  int<lower=1> N_test;
  int<lower=1> N_biology;
  int<lower=1> N_substrate;
  int<lower=1> N_ec4_sub;
  int<lower=1> N_enz_sub;
  int<lower=1> N_org_sub;
  int<lower=1,upper=N_ec4_sub> ec4_sub[N_biology];
  int<lower=1,upper=N_org_sub> org_sub[N_biology];
  int<lower=0,upper=N_enz_sub> enz_sub[N_biology];
  int<lower=1,upper=N_substrate> substrate[N_biology];
  array[N_train] int<lower=1,upper=N_biology> biology_train;
  array[N_train] int<lower=1,upper=N> ix_train;
  array[N_test] int<lower=1,upper=N_biology> biology_test;
  array[N_test] int<lower=1,upper=N> ix_test;
  vector[N] y;
  int<lower=0,upper=1> likelihood;
}
parameters {
  real<lower=1> nu;
  real mu;
  real<lower=0> sigma;
  real<lower=0> tau_substrate;
  real<lower=0> tau_org_sub;
  real<lower=0> tau_ec4_sub;
  real<lower=0> tau_enz_sub;
  vector<multiplier=tau_substrate>[N_substrate] a_substrate;
  vector<multiplier=tau_org_sub>[N_org_sub] a_org_sub;
  vector<multiplier=tau_ec4_sub>[N_ec4_sub] a_ec4_sub;
  vector<multiplier=tau_enz_sub>[N_enz_sub] a_enz_sub;
}
transformed parameters {
  vector[N_biology] log_km = mu
    + a_substrate[substrate]
    + a_ec4_sub[ec4_sub]
    + a_org_sub[org_sub];
  for (b in 1:N_biology){
    if (enz_sub[b] > 1) log_km[b] += a_enz_sub[enz_sub[b]];
  }
}
```

```
model {
  if (likelihood){
    y[ix_train] ~ student_t(nu, log_km[biology_train], sigma);
  }
  nu ~ gamma(2, 0.1);
  sigma ~ normal(0, 2);
  mu ~ normal(-1, 2);
  a_substrate ~ normal(0, tau_substrate);
  a_ec4_sub ~ normal(0, tau_ec4_sub);
  a_enz_sub ~ normal(0, tau_enz_sub);
  a_org_sub ~ normal(0, tau_org_sub);
  tau_org_sub ~ normal(0, 1);
  tau_ec4_sub ~ normal(0, 1);
  tau_enz_sub ~ normal(0, 1);
  tau_substrate ~ normal(0, 1);
}
generated quantities {
  vector[N_test] llik;
  vector[N_test] yrep;
  for (n in 1:N_test){
    llik[n] = student_t_lpdf(y[ix_test[n]] | nu, log_km[biology_test[n]], sigma);
    yrep[n] = student_t_rng(nu, log_km[biology_test[n]], sigma);
  }
}
```

For the BRENDA data, we used the same model but removed the enzyme-substrate interaction parameters as BRENDA does not provide enzyme-specific information.

3.4 Model validation

We used a number of standard methods to test our models' validity, aiming to follow the method described in [26]. To verify the computation we used standard MCMC convergence metrics and fit our model to fake data. To assess model specification we performed graphical prior and posterior predictive checks and both approximate and exact cross validation, using some simple test models for comparison.

3.4.1 Computation

For each MCMC run we used arviz [27] to calculate the improved statistic following the method in [28] and to check for divergent transitions. If the statistic was sufficiently close to 1 (i.e. +/- 0.03) and there were no post-warmup divergent transitions we judged that the computation was likely to have been successful in the sense that the draws can be treated as samples from the target distribution.

To further validate the computation we also fit the model to fake data generated using the model assumptions and a plausible configuration of parameters. We used graphical posterior predictive checks to assess whether the true parameters were approximately recovered in the posterior distribution. See supplementary material for graphical posterior predictive checks with fake data.

3.4.2 Comparison models

Ideally we would compare the results of our models with previously published attempts to model the same data. However this was not practical in our case for two reasons. First, other results are typically derived from different raw datasets, and rely on different data filtering and summarising decisions. In particular, we have not previously seen the problem of whether and how to aggregate results from the same study addressed in detail. Second, the other results typically assessed model specification by comparing observed data to point predictions generated by their models. Applying this procedure to our models would give an incomplete picture, since a single point cannot adequately summarise the full posterior predictive distribution. In addition, it would not align with the priorities of our intended users, namely systems biologists constructing prior distributions for the purpose of ensemble modelling, who we imagine care more about the extremes of the distributions than central estimates.

Due to this lack, we constructed two simple models purely for comparison with our main model. The first model, which we called the "really simple" model, removes all latent random variables from the main model's predictor, except for the global mean parameter μ. The second model, which we called the "simple" model, includes a single vector of hierarchical parameters at the finest available granularity, i.e. organism:substrate:ec4 for BRENDA data and organism:substrate:enzyme for SABIO-RK data. See supplemental information for full model specifications in the form of tilde notation and

Stan programs.

Our thinking was that the really simple model and simple models represent relative extremes of underfitting and overfitting. A well-specified model should be able to make better predictions than both, unless the data are very surprisingly uninformative.

In addition to these comparison models, we also compared our final model with a Bayesian version of the model in [19], which we obtained simply by removing the parameters $a^{enz:sub}$ and $\tau^{enz:sub}$ from our final model. We included this extra test in order to verify whether including enzyme-specific parameters was worthwhile.

3.4.3 Graphical posterior predictive checks

Figure 1 below shows our main model's marginal posterior predictive 1%-99% interval for each observation, alongside the observed value. Ideally exactly 98% of observations should be covered, and whether or not an observation is covered should not be systematically predictable.

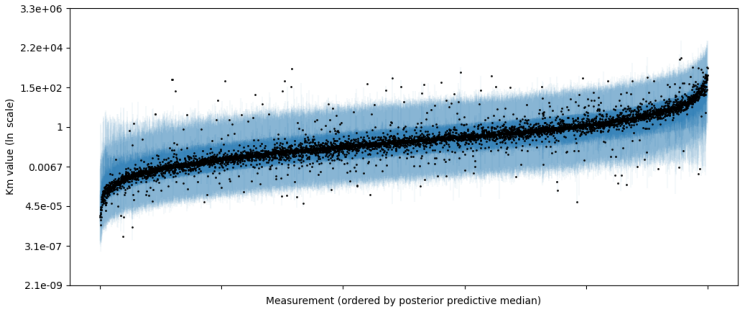

Figure 1: Marginal posterior predictive distributions for main model fit to SABIO Km data

Figures 2 and 3 show graphical posterior predictive checks for the really simple and simple models fit to the same dataset.

From the continuous line of black dots in figure 2 we can see that the posterior predictive intervals have approximately the same order as the observed data, suggesting that the model is likely overfitting, or at least not using any structural information from the data.

In contrast figure 3 shows that the really simple model is clearly underfitting

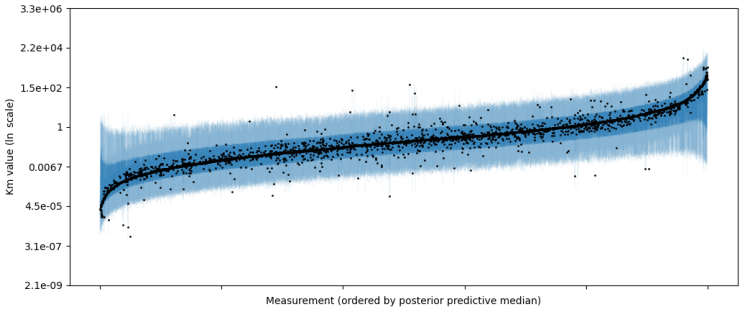

Figure 2: Marginal posterior predictive distributions for simple model fit to SABIO Km data

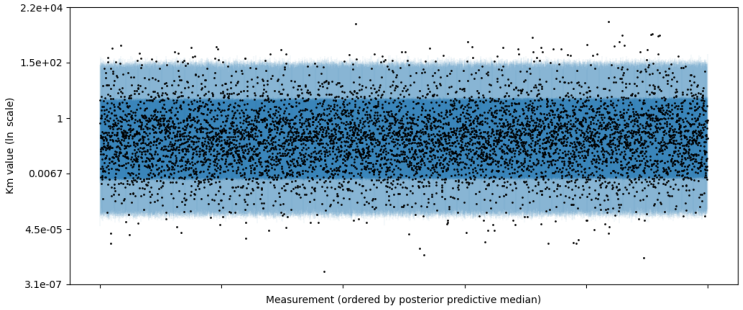

Figure 3: Marginal posterior predictive distributions for really simple model fit to SABIO Km data

the observed data - there is no relationship between the order of the black dots and the blue lines.

3.4.4 TODO Cross validation

For a quantitative compliment to the graphical posterior predictive checks, we used cross-validation to assess our models' out-of-sample predictive performance. For exploratory comparison we calculated each model's approximate leave-one-out expected log predictive density using the method set out in [29] and implemented in [27]. To address cases where diagnostics suggested that the approximate leave-one-out algorithm was likely to be unreliable we also carried out exact tenfold cross-validation.

The results were as follows:

RESULTS OF CROSS VALIDATION COMPARISON

3.5 DONE Web app

In order to make our results accessible to the systems biology community, we wrote a web app that presents them in an easy to use and actionable format. This can be found at URL.

The user can choose a dataset, organism, substrate and enzyme and is then presented with a KDE summary of the corresponding marginal posterior distribution and some summary statistics describing it, which can be used to inform a choice of priors.

For each biological category there is an option to choose "Unknown" - in this case the relevant marginal posterior is calculated dynamically.

The app also provides links from which tables summarising all the marginal posteriors can be downloaded.

The app was written using Streamlit [30].

4 Results

4.1 Marginal distributions of interesting parameters

We begin by looking at the marginal distributions of the model's standard deviation parameters. Comparing these indicates which effects varied the most. The comparison in figure 4 shows that the most important hierarchical

parameter is the substrate effect, followed by the ec4/substrate interaction, the enzyme/substrate interaction and finally the organism/substrate interaction. All of the distributions tend to be greater than that of the estimated measurement error `sigma`, suggesting that our model is able to distinguish all the effects from measurement noise. The substrate and ec4:substrate taus tend to be the greatest, suggesting that these are the most important predictors.

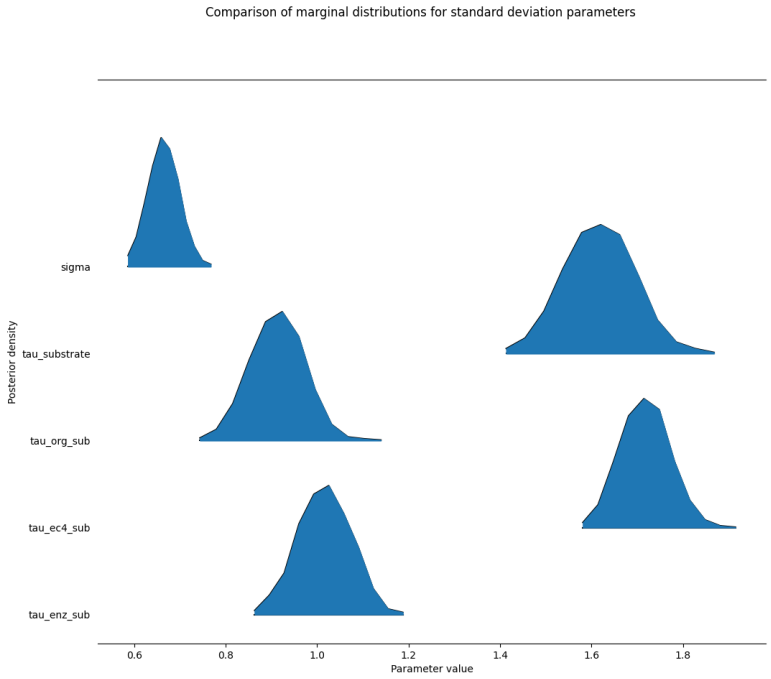

Figure 4: Comparison of hierarchical standard deviation parameters

Figure 5 compares the marginal distributions of km parameters from our best SABIO-RK and BRENDA models. The marginal posterior distributions in the BRENDA model tend to be narrower, reflecting the larger input dataset.

What else to put here? Distributions of ln kms?

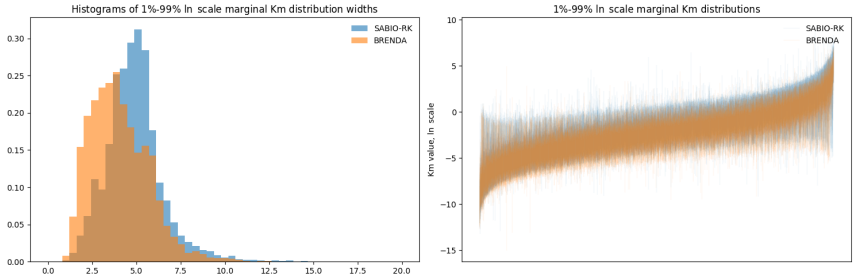

Figure 5: Comparison of measured and modelled Km parameter distributions in BRENDA and SABIO-RK

4.2 Comparison of estimated Kms with physiological metabolite concentrations

Several sources suggest that the Km parameter for a particular organism, substrate and enzyme should have the same order of magnitude as the typical physiological concentration of that substrate. ANOTHER REFERENCE HERE? For example [19] write

> . . . we may hypothesize that KM values are adjusted to the order of magnitude of the substrate concentration. If this is the case and if a metabolite exhibits a particularly high concentration in a certain organism, then all corresponding KM values should also tend to be increased.

In order to test whether this is the case, we obtained data about some physiological metabolite concentrations from SABIO-RK and compared these with the corresponding Km measurements and marginal posterior distributions from our model. Figure 6 shows the results.

The graph on the left shows that our data somewhat support the first part of the hypothesis: the physiological concentrations are distributed similarly to the posterior Km samples. The graph on the right suggests that ln scale Kms tend to be 1-2 ln mM lower than their corresponding physiological concentration reports, give or take about 5 ln mM. In other words, there is only a fairly weak relationship between a particular Km and its corresonding physiological concentration.

This is reflected in our model's marginal posterior distribution for the parameter $\tau^{org:sub}$, which concentrates closer to zero than any of the other hierarchical standard deviation parameters CHECK THIS IS TRUE, as can

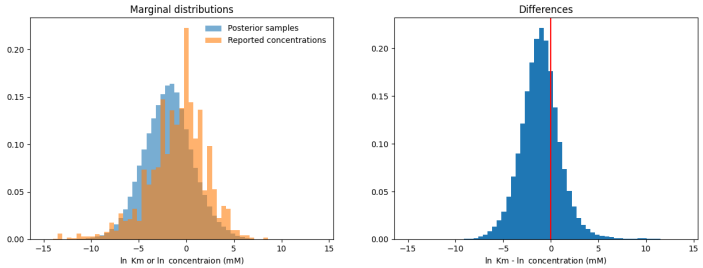

Figure 6: Comparison of model predictions with physiological metabolite concentrations from HMDB

be seen in figure 4

4.3 NADH vs NADPH

Figure 7 shows histograms of measured Km parameters from the SABIO-RK dataset for NADH and NADPH binding enzymes alongside histograms of our final model's posterior samples for these parameters. We can see that the measurements for NADPH tended to be higher, and this difference is also present in the posterior distributions.

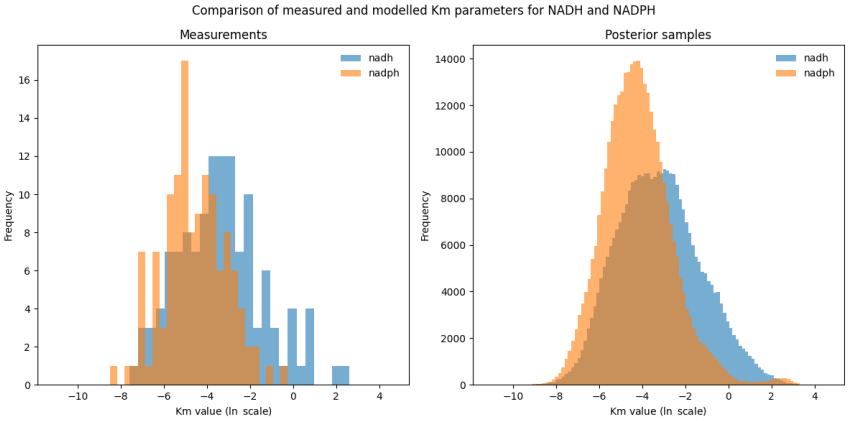

Figure 7: Comparison of modelled and measured Km parameters for NADH and NADPH

4.4 TODO Cofactor vs substrate

We tested whether there tended to be a systematic difference between the km of an enzyme's main substrate and that of a cofactor.

4.5 Which results to use?

Our results make it possible to compare the SABIO-RK and BRENDA datasets, allowing practitioners to make more informed modelling decisions.

The SABIO-RK dataset provides more specific information than the BRENDA datase. UNIPROT ids are available for many Km measurements, whereas the most specific enzyme-level information in the BRENDA dataset is the EC4 number. In addition, in contrast to BRENDA, the SABIO-RK dataset provides information about substrate, literature reference, enzyme, temperature, pH and strain type in interoperable formats and without the need for potentially error-prone string operations.

As a result of this extra information, the model fit to the SABIO-RK data is more specific, allowing modellers to obtain priors that reflect measurements of the exact enzyme they wish to model while still appropriately taking into account measurements of related enyzmes. In addition, the SABIO model excludes some potential sources of bias, such as reports from non-wildtype strains and atypical conditions that are not accounted for in the BRENDA model.

The SABIO dataset is worse than the BRENDA dataset in one respect, however: at the time of writing there are far more reports in the BRENDA dataset. The model fit to the BRENDA dataset is therefore able to take into account more information than is available to the SABIO-RK model.

There are therefore potentially conflicting factors to take into account when deciding which dataset to use in a given case. A further complication is that, due to the lack of interoperable identifiers in the BRENDA dataset, it is difficult to directly compare the which yields better predctions.

In our judgment the best approach given these circumstances is to favour the SABIO-RK results in cases where they provides a lot of information for a given parameter or when the information in BRENDA is too coarse or likely to be biased. In other cases the BRENDA results should be preferred.

Our results make it easier to assess which of these two scenarios apply. The information available for a given km is captured by the marginal posterior

distribution for that parameter - if the marginal posterior distribution for a parameter in the SABIO model is very narrow, this means that the SABIO dataset contains a lot of information about that parameter. In order to assess the liklihood of the BRENDA dataset being too coarse for a parameter whose enzyme shares an EC number, the marginal prior distributions of the common parameters can be compared. If they are very different, it is likely better to use the more specific SABIO results. Finally, in order to assess the likelihood of bias in the BRENDA data, our webapp provides links to the source documents for any parameter measurement.

5 TODO Case studies

6 TODO Discussion

7 References

[1] A. J. Lopatkin and J. J. Collins, "Predictive biology: Modelling, understanding and harnessing microbial complexity," *Nature reviews. microbiology*, vol. 18, no. 9, pp. 507–520, Sep. 2020, doi: 10.1038/s41579-020-0372-5.

[2] S. Khoshnaw, R. Salih, and S. Sulaimany, "Mathematical Modelling for Coronavirus Disease (COVID-19) in Predicting Future Behaviours and Sensitivity Analysis," *Mathematical modelling of natural phenomena*, vol. 15, May 2020, doi: 10.1051/mmnp/2020020.

[3] M. Tokic, V. Hatzimanikatis, and L. Miskovic, "Large-scale kinetic metabolic models of Pseudomonas putida KT2440 for consistent design of metabolic engineering strategies," *Biotechnology for biofuels*, vol. 13, p. 33, 2020, doi: 10.1186/s13068-020-1665-7.

[4] A. Khodayari and C. D. Maranas, "A genome-scale Escherichia coli kinetic metabolic model k-ecoli457 satisfying flux data for multiple mutant strains," *Nature communications*, vol. 7, no. 1, Dec. 2016, doi: 10.1038/ncomms13806.

[5] P. A. Saa and L. K. Nielsen, "Formulation, construction and analysis of kinetic models of metabolism: A review of modelling frameworks," *Biotechnology advances*, vol. 35, no. 8, pp. 981–1003, Dec. 2017, doi: 10.1016/j.biotechadv.2017.09.005.

[6] A. Tsigkinopoulou, S. M. Baker, and R. Breitling, "Respectful Modeling: Addressing Uncertainty in Dynamic System Models for Molecular Biology,"

Trends in biotechnology, vol. 35, no. 6, pp. 518–529, Jun. 2017, doi: 10.1016/j.tibtech.2016.12.008.

[7] F. Hussain, C. J. Langmead, Q. Mi, J. Dutta-Moscato, Y. Vodovotz, and S. K. Jha, "Automated parameter estimation for biological models using Bayesian statistical model checking," *Bmc bioinformatics*, vol. 16, no. 17, p. S8, Dec. 2015, doi: 10.1186/1471-2105-16-S17-S8.

[8] X. Zhu, M. Welling, F. Jin, and J. Lowengrub, "Predicting simulation parameters of biological systems using a Gaussian process model," *Statistical analysis and data mining: The asa data science journal*, vol. 5, no. 6, pp. 509–522, 2012, doi: 10.1002/sam.11163.

[9] S. van Mourik, C. ter Braak, H. Stigter, and J. Molenaar, "Prediction uncertainty assessment of a systems biology model requires a sample of the full probability distribution of its parameters," *Peerj*, vol. 2, p. e433, Jun. 2014, doi: 10.7717/peerj.433.

[10] P. Saa and L. K. Nielsen, "A General Framework for Thermodynamically Consistent Parameterization and Efficient Sampling of Enzymatic Reactions," *Plos computational biology*, vol. 11, no. 4, p. e1004195, Apr. 2015, doi: 10.1371/journal.pcbi.1004195.

[11] S. Mukherjee and T. P. Speed, "Network inference using informative priors," *Proceedings of the national academy of sciences*, vol. 105, no. 38, pp. 14313–14318, Sep. 2008, doi: 10.1073/pnas.0802272105.

[12] R. E. Baker, J.-M. Peña, J. Jayamohan, and A. Jérusalem, "Mechanistic models versus machine learning, a fight worth fighting for the biological community?," *Biology letters*, vol. 14, no. 5, p. 20170660, May 2018, doi: 10.1098/rsbl.2017.0660.

[13] C. Xu and S. A. Jackson, "Machine learning and complex biological data," *Genome biology*, vol. 20, no. 1, p. 76, Apr. 2019, doi: 10.1186/s13059-019-1689-0.

[14] M. Zitnik, F. Nguyen, B. Wang, J. Leskovec, A. Goldenberg, and M. M. Hoffman, "Machine Learning for Integrating Data in Biology and Medicine: Principles, Practice, and Opportunities," *An international journal on information fusion*, vol. 50, pp. 71–91, Oct. 2019, doi: 10.1016/j.inffus.2018.09.012.

[15] S.-M. Yan, D.-Q. Shi, H. Nong, and G. Wu, "Predicting Km values of beta-glucosidases using cellobiose as substrate," *Interdisciplinary sciences,*

computational life sciences, vol. 4, no. 1, pp. 46–53, Mar. 2012, doi: 10.1007/s12539-012-0115-z.

[16] A. Kroll, M. K. M. Engqvist, D. Heckmann, and M. J. Lercher, "Deep learning allows genome-scale prediction of Michaelis constants from structural features," *Plos biology*, vol. 19, no. 10, p. e3001402, Oct. 2021, doi: 10.1371/journal.pbio.3001402.

[17] F. Li *et al.*, "Deep learning based kcat prediction enables improved enzyme constrained model reconstruction." Cold Spring Harbor Laboratory, p. 2021.08.06.455417, Aug. 2021. doi: 10.1101/2021.08.06.455417.

[18] A. Tsigkinopoulou, A. Hawari, M. Uttley, and R. Breitling, "Defining informative priors for ensemble modeling in systems biology," *Nature protocols*, vol. 13, no. 11, pp. 2643–2663, Nov. 2018, doi: 10.1038/s41596-018-0056-z.

[19] S. Borger, W. Liebermeister, and E. Klipp, "Prediction of enzyme kinetic parameters based on statistical learning," *Genome informatics*, vol. 17, no. 1, pp. 80–87, 2006, Accessed: Oct. 15, 2021. [Online]. Available: `https://pure.mpg.de/pubman/faces/ViewItemOverviewPage.jsp?itemId=item_1585340`

[20] "BRENDA - SOAP access help." Accessed: Nov. 18, 2021. [Online]. Available: `https://www.brenda-enzymes.org/soap.php#Turnover_Number`

[21] U. Wittig *et al.*, "SABIO-RK database for biochemical reaction kinetics," *Nucleic acids research*, vol. 40, no. D1, pp. D790–D796, Jan. 2012, doi: 10.1093/nar/gkr1046.

[22] A. Chang *et al.*, "BRENDA, the ELIXIR core data resource in 2021: New developments and updates," *Nucleic acids research*, vol. 49, no. D1, pp. D498–D508, Jan. 2021, doi: 10.1093/nar/gkaa1025.

[23] S. Duvaud, C. Gabella, F. Lisacek, H. Stockinger, V. Ioannidis, and C. Durinx, "Expasy, the Swiss Bioinformatics Resource Portal, as designed by its users," *Nucleic acids research*, vol. 49, no. W1, pp. W216–W227, Jul. 2021, doi: 10.1093/nar/gkab225.

[24] M. A. Juárez and M. F. J. Steel, "Model-Based Clustering of Non-Gaussian Panel Data Based on Skew-t Distributions," *Journal of business & economic statistics*, vol. 28, no. 1, pp. 52–66, Jan. 2010, doi: 10.1198/jbes.2009.07145.

[25] B. Carpenter *et al.*, "Stan: A Probabilistic Programming Language," *Journal of statistical software*, vol. 76, no. 1, pp. 1–32, Jan. 2017, doi: 10.18637/jss.v076.i01.

[26] A. Gelman *et al.*, "Bayesian Workflow," *Arxiv:2011.01808 [stat]*, Nov. 2020, Accessed: Nov. 05, 2020. [Online]. Available: `http://arxiv.org/abs/2011.01808`

[27] R. Kumar, C. Carroll, A. Hartikainen, and O. Martin, "ArviZ a unified library for exploratory analysis of Bayesian models in Python," *Journal of open source software*, vol. 4, no. 33, p. 1143, Jan. 2019, doi: 10.21105/joss.01143.

[28] A. Vehtari, A. Gelman, D. Simpson, B. Carpenter, and P.-C. Bürkner, "Rank-Normalization, Folding, and Localization: An Improved R for Assessing Convergence of MCMC (with Discussion)," *Bayesian analysis*, vol. 16, no. 2, pp. 667–718, Jun. 2021, doi: 10.1214/20-BA1221.

[29] A. Vehtari, A. Gelman, and J. Gabry, "Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC," *Statistics and computing*, vol. 27, no. 5, pp. 1413–1432, Sep. 2017, doi: 10.1007/s11222-016-9696-4.

[30] "Streamlit The fastest way to build and share data apps." Accessed: Feb. 01, 2022. [Online]. Available: `https://streamlit.io/`