EXPLAINABLE AI FOR METABOLIC ENGINEERING

Tijmen van Graft

4656687
Delft University of Technology
Date of Award 6-4-2023
Supervisors:
Thomas Abeel and P.H. van Lent

Abstract

Metabolic engineering is an important field in biotechnology, aimed at optimizing cellular processes to produce desired compounds. In this thesis, we focus on predicting the metabolome from the proteome, as understanding this relationship is crucial for understanding cellular metabolism. We investigate the usage of additional biological information like protein-protein interactions and cellular stoichiometry to improve the predictive performance of metabolome prediction models. We also employ explanation algorithms to gain key insights into the regulatory processes of a yeast cell.

We demonstrate the effectiveness of our approach by predicting the metabolic fold-change of multiple yeast kinase knockouts. Our results show that incorporating additional biological information does not significantly improve the accuracy of the metabolome prediction models. Furthermore, we identified enzymes that are relevant for all metabolites used in this study, which indicates the existence of a global set of regulatory enzymes.

Overall, our study shows that through careful manipulation of the limit amount of data decent performance can be expected when predicting the metabolome. We apply a broad spectrum of machine learning algorithms to identify optimal model architecture. The methods and insights presented in this thesis could be used for creating a general pipeline for predicting a broad spectrum of metabolites from the proteome.

Introduction

Beer brewery has started 13700-11700 BCE years ago with the help of wild yeasts fermenting a mixture of wheat and water (Liu et al., 2018). Later specific strains would be isolated that specialized in fermentation (Samuel, 1996) and bread making (Feldmann, 2012). Fermentation is a metabolic pathway that extract energy from carbohydrates in the absence of oxygen (Neijssel & Tempest, 1986). This process is available to yeasts and other microorganisms, which makes fermentation one of the oldest metabolic processes. Reactants, products, and intermediates of reactions are known as metabolites. The reactions and metabolites form a complex metabolic network of interdepend reactions of which segments can be labeled as pathways. Thus far 905 enzymes, 1577 reactions and 1226 have been identified in the yest metabolic network (King et al., 2016). The concentration of enzymes in a cell, along with the catalytic properties of the enzyme, often determines the rate of a metabolic reaction. Therefore, to control the rate at which a metabolite is consumed or produced a cell might increase or decrease the concentration of a particular enzyme (Robinson, 2015). The metabolism of natural occurring microorganism is not suited for the large-scale production of molecules of interests. By altering the metabolic network of these microorganisms, metabolic engineers can create efficient microbial cell factories that produce a molecule of interest in an economical and sustainable way (Lee et al., 2012). An application of a microbial cell factory would be the production of plastic from non-food carbon sources instead of relying on non-renewable substrates like oil or gas (Chae et al., 2017). The effectiveness of a microbial cell factory can be measured based on the titer (product concentration), yield (gram of product per gram of substrate), and productivity (gram of product per cell per unit time) for the target molecule (Oyetunde et al., 2019). A multitude of strategies exists that can help a metabolic engineer to improve the effectiveness of a cell's metabolism (Figure 1). For the optimization of metabolic flux, the expression levels of promoters, enzymes and regulatory elements can be controlled via altering the DNA of a target strain.

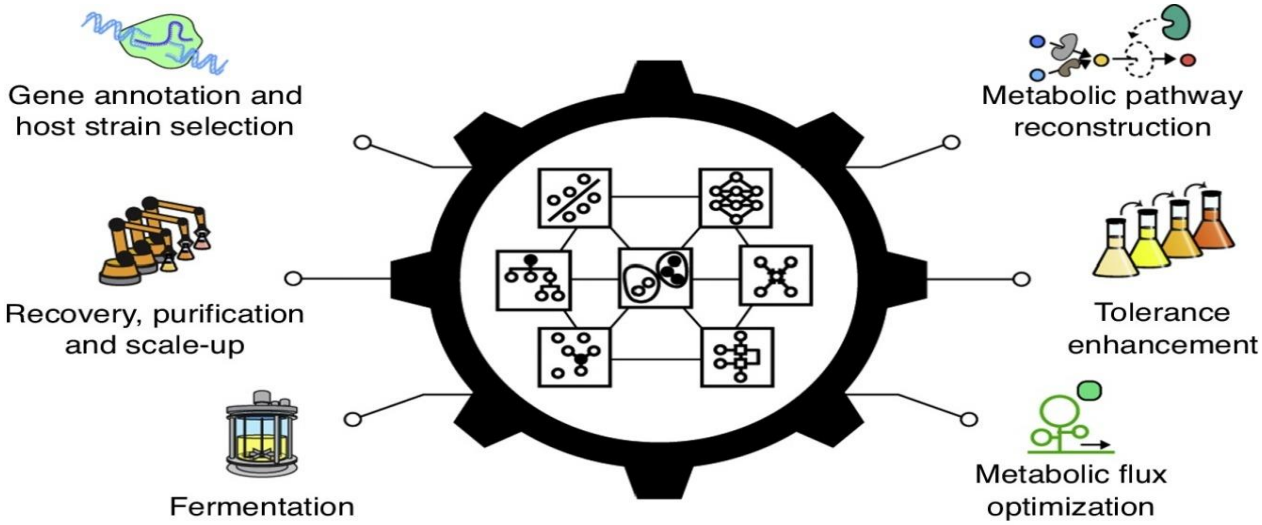


Figure 1: Overview of the variety of metabolic engineering strategies that can be applied to improve titer, yield and productivity (G. B. Kim et al., 2020)*. Gene annotation and host strain selection: to pick a target strain detailed information is required about the strain's characteristics* (H. U. Kim et al., 2016)*. Recovery, purification and scale up: finding an effective strategy to extract the target metabolites from the reactor without destroying the processes within* (Herzer et al., 2015)*. Fermentation: the hyperparameters like the optimal growth temperature (OGT) or optimal nutrient composition have a large impact on the production of target metabolites* (Masampally et al., 2019; Zheng et al., 2017)*. Metabolic pathway reconstruction: retro synthesis can be applied to create a novel pathway to creates complex molecules from simpler precursors* (Segler et al., 2018).

From this new artificial pathway metabolic engineers aim to find corresponding enzymes to introduce the pathway in microbial factories (Hadadi et al., 2019). *Tolerance enhancement: improves the cells resistance to metabolic and environmental stress* (Swinnen et al., 2017). *Metabolic flux optimization: Selected expression levels of promoters, enzymes and regulatory elements can be controlled via creating configurations of the expression levels* (Choi et al., 2019; Groher et al., 2019a; Jervis et al., 2019).

There have been plenty of successful applications of metabolic flux optimization. An example is the production optimization of *n*-butanol in yeast cells. *n*-butanol is bio sustainable replacement of gasoline due to its comparable energy density (Shi et al., 2016). For the efficient creation of *n*-butanol metabolic engineers overexpressed (increased the concentration of) keto-acid decarboxylases and alcohol dehydrogenase in *S. cerevisiae* (Shi et al., 2016). The overexpression of these enzymes, together with other edits lead to a 7-fold increase in the production of *n*-butanol compared to the original strain. The improved production of *n*-butanol required changes in natural occurring strains of the target organism. A prominent technology to make these changes is CRISPR/Cas9, which allows for precise edits in the DNA of the target strain (Patmanathan et al., 2018).

The number of possible changes a metabolic engineer can make in cellular metabolism is enormous. Metabolic pathways are interconnected networks and modifying one part of the pathway might have unexpected effects on other parts of the metabolic network (Nielsen & Keasling, 2016). Via the regulation and feedback mechanisms found in cellular metabolism, upregulation of one enzyme might trigger downregulating response of another pathway which further complicates engineering efforts (Nielsen, 2014). Acquiring large amounts of high-quality data has become easier through cheaper, more sophisticated, and accessible machinery. Information is acquired over different layers in a cell, these studies are often referred to as -omics studies. Genome scale models (GSM) are a popular mechanistic modelling approach to simulate the metabolism of a cell, often coverings thousands of metabolic reactions. This class of computational models offers a qualitative mapping of cellular metabolism, can help discover metabolic functions and guide metabolic engineers towards desired phenotypes (Monk et al., 2017). A GSM is created via combining information derived from multiple -omics studies, other biological information, and human knowledge (Chakdar et al., 2021). This approach ensures that GSMs are an accurate depiction of the cellular metabolism of a target organism, however creating high-quality GSMs is a costly process (Mendoza et al., 2019; Thiele & Palsson, 2010). An alternative approach to mechanistic modelling is data-driven modelling, where machine learning algorithms learn directly from experimental datasets (Zhang et al., 2020). This removes the need of having a priori information about a target organism. In one study the proteomics information of *s. cerevisiae* was used to directly predict the metabolite concentration of the cell after performing kinase knockouts (Zelezniak et al., 2018). The data-driven modelling approach can also more easily cover a lesser studied organism, as less expensive data is required to use this modelling approach.

Data-driven modelling relies on the quality and quantity of the experimental information available. Predicting the cellular phenotype is not trivial. Proteomic data, which is closely interacting with the metabolome, is scarcely available. Genomics data is readily and widely available, but the interaction with the metabolome is more indirect (Töpfer et al., 2015). We hypothesize that biological information that is closer to the metabolome has more predictive power than information that is further away, as cellular processes play a large in determining a cells phenotype (Burga & Lehner, 2012). Adding more biological information either via vertical integration (more -omics layers) or horizontal integration (more biological information from different sources) could mitigate this issue (Töpfer et al., 2015). Proteomics data could be enriched by horizontally integrating protein-protein interaction (PPIs) information. PPIs play a relevant role among different cellular functions, understanding the interactions improves the

understanding of cellular function (Shatnawi, 2015). The PPI information could be used as filter for proteomics data by creating samples that only consists of known interacting proteins, allowing machine learning models to directly learn a relationship between interacting proteins and metabolite concentrations. GSMs can be used in a data-driven fashion, as this model can be leveraged to select and rank reactions or genes that most likely affect the production of a metabolite or via directly incorporating their stoichiometry in a machine learning algorithm (Zhang et al., 2020).

With the increased volume of information available and increasing reliance on data-driven modelling more complex machine learning algorithms are developed or used in the field of bioinformatics. Algorithms that predict the up and down regulation of a set of enzymes for the optimal metabolite concentration, make it difficult to find the individual contribution of a particular enzyme (Yamamoto et al., 2023). Explainable AI could be applied to open the black box machine learning models to find the individual contribution of a particular enzyme, which helps understand the relationship between enzymes and metabolites (Belle & Papantonis, 2021). An explanation in the machine learning context aims to convey insights in how an algorithm makes a prediction. The aim of generating explanations is to increase trust in AI based systems. There exist many criteria to measure the goodness of an explanation, which makes it difficult to create sufficient explainable model, as most criteria have trade-offs between them (*Interpretable Machine Learning*, n.d.). The fidelity, the truthfulness of the explanation to the black-box model, and accuracy, the error between the explanation and prediction target, are the most interesting metrics for this study. For the generation of explanations two model-agnostic algorithms are commonly used: LIME and SHAP. LIME uses a local weighted linear model to find the most important features for an instance. The SHAP algorithm uses game theory to calculate the contribution of each feature to a prediction.

Now that we introduced the different components of this research, we can formulate the research question: “How can we accurately predict the metabolite concentration after intervention using explainable AI?”

Research objective

The aim of this research is to further investigate the relationship between the proteome and the metabolome using machine learning algorithms. With the help of explanation algorithms, we aim to project the relationship between the proteome and metabolome on a genome scale metabolic model.

This research objective is split into three questions.

1. What is an effective model architecture for predicting the precursor metabolites concentrations from the proteome?
2. What is the effect of including prior biological information on model performance of precursor metabolite concentration prediction?
3. How can explanation algorithms extract useful information from a trained machine learning model and project it on a genome scale metabolic model?

Answering these questions, we apply a variety of machine learning and data analysis techniques to look at this research from a computer science perspective. To answer the first question, we use a dataset of the proteome and metabolome data extracted from (Zelezniak et al., 2018) and apply a variety of machine learning techniques. For the second question, data from StringDB and the yeast stoichiometry are integrated to give additional biological context to the model. Finally for the third question we apply a variety of explanation algorithms to the machine learning models and rank them based on their usability.

Background

Metabolism is a set of chemical reactions that include the conversion of food energy into cellular energy, the conversion of food into building blocks for proteins, lipids, nucleic acids, and carbohydrates and the elimination of metabolic waste. These reactions are organized in metabolic pathways, where a chemical is transformed through several steps which are catalyzed by enzymes. A high-level abstract overview of these different components is organized are shown in (Figure 2).

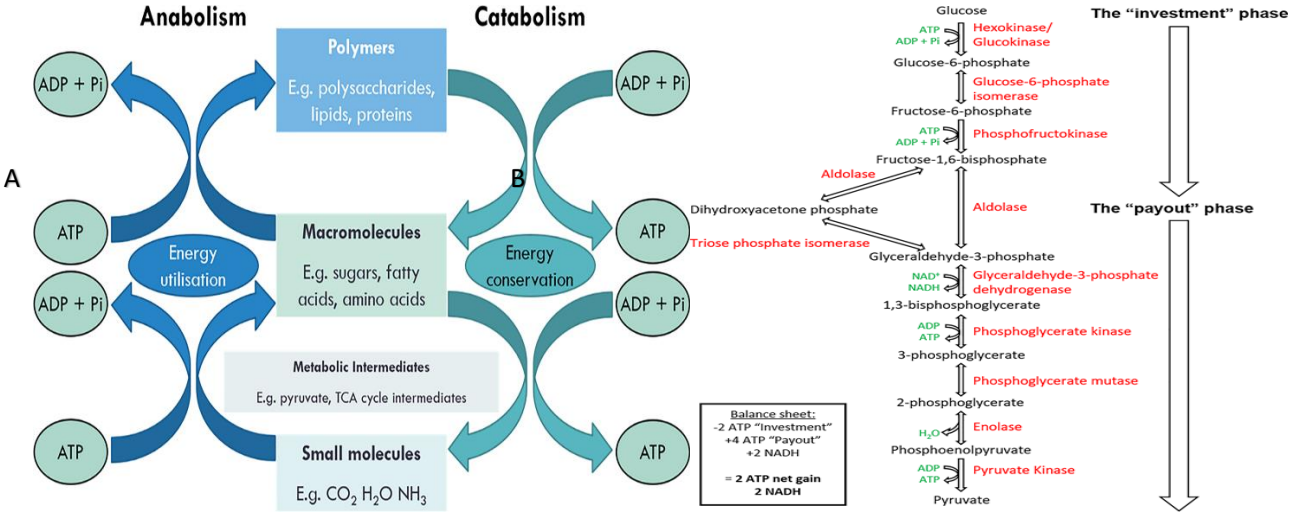


Figure 2: Simplification of the components of metabolism, both figures are adapted from (Judge & Dodd, 2020). The interplay between the different categories of metabolites and how they are transformed via anabolic and catabolic reactions are shown (Panel A). Right figure: the glycolysis pathway, which is available in a variety of organisms, is shown. The input to the glycolysis pathway is glucose which is transformed into pyruvate using a variety of reactions (Panel B). During the process a net gain of 2 ATP and 2 NADH is achieved, therefore this pathway can be classified as a catabolic pathway.

A cell can conceptually be divided into different layers (-omes) like genome, proteome, transcriptome, epigenome, or metabolome (Figure 3). With the rise of high throughput techniques, like array technologies, and high-throughput mass spectrometry the volume and quality of data increased. The development of algorithms for the movement, management, and integration of high-dimensional data made these datasets more available to researchers (Krassowski et al., 2020). The data from different omics studies can be combined to form a multi-omics study.

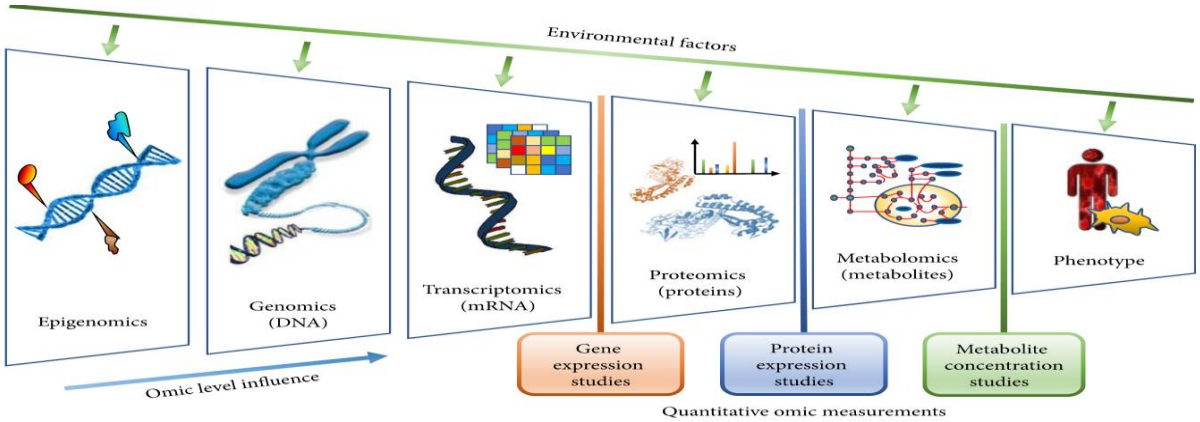


Figure 3: Different layers of multi-omics datasets (Angione, 2019). *Omics studies help to better understand cellular mechanisms, however a singular one give a very limited view of a mechanism, therefore integrating multiple omics can help us understand the*

mechanism more clearly. Environmental perturbations influence the complex interactions and feedback loops over several omic-layers. Epigenomics consists of epigenetic marker changes which do not involve DNA sequences, gene activity and expression. Genomics considers DNA which contains the genetic code of the cell. Transcriptomics describes the RNA encoded in the genome. Proteomics is the study of the proteins produced because of gene expression and posttranslational modifications. Finally, metabolomics stores the set of metabolites and metabolic reactions taking place in a cell.

Protein-protein interactions

Proteins play an important role in cellular metabolism. Proteins interact with each other, for example, the follicle-stimulating hormone travels from the brain to the ovary where it causes an egg release. These signals are received by receptor proteins that bind to a signalling molecule and initiate a physiological response. The STRING is a database that stores these interactions and can be used to model a protein-protein interaction (PPI) network database (*STRING: Functional Protein Association Networks*, n.d.).

The evidence for relationships between proteins is based on seven *evidence channels*. These channels can be grouped into three evidence topics: computational association, functional genomics, and consolidated knowledge. The final score of the interaction is normalized to be between zero and one, where a score closer to one indicates that STRING is more confident that this interaction is significant.

Although the STRING database contains a lot of information its content is still quite limited and must be combined with other PPI databases (not discussed here) to cover for example all interactions found in the CYC2008 catalogue (Nakajima et al., 2018). Finally, the STRING database is only able to show a limited disjoint network. This might influence for example clustering analysis as not all relevant protein interactions have been found.

Mechanistic modelling

Constraint-based modelling

Mathematical models have often been applied to analyze fluxes in a metabolic network. We will now discuss flux balance analysis (FBA). FBA aims to find a configuration of fluxes that optimize an objective function, like maximizing the growth rate. The stoichiometric coefficient of each reaction is stored in a tabulated form as a first step in the algorithm. This matrix has the shape n, the number of reactions, and m, the number of metabolites. For a metabolic network, n is larger than m creating $n - m$ degrees of freedom. This definition of the stoichiometric matrix allows for both reversible and irreversible reactions. Next upper and lower bounds are formulated based on literature or other sources. Furthermore, experimentally validated fluxes can be incorporated, as additional constraints to limit the degree of freedom in the system. Constraints reduce the available search space of the possible enzymatic concentrations (Figure 4).

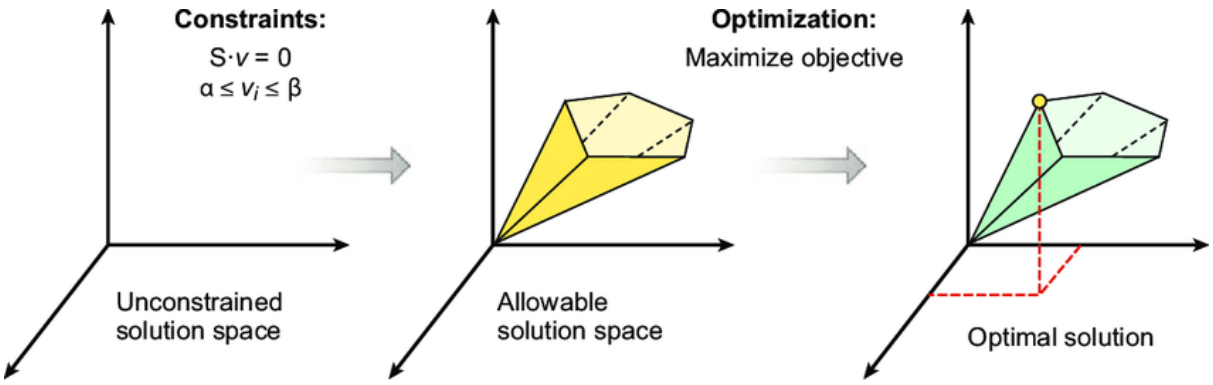


Figure 4: Constraint formulation allows the FBA algorithms to efficiently search the solution space for a configuration of fluxes that optimize an objective function. Often an upper and lower limit is given for each flux and the steady-state condition is used as a global constraint (Libourel & Shachar-Hill, 2008).

Determining the correct optimization function is important as the proposed distribution of fluxes is directly related to the optimization function. The target metabolite can be coupled with the growth objective function since growth is a realistic objective of a cell (Edwards et al., 1995). However not exactly knowing the optimization function is a major limitation of flux-balance analysis and can be defined as the knowledge-gap.

Kinetic based modelling

The flux is the rate at which metabolic products are created minus the rate at which they are consumed. Three factors determine the flux of a reaction: the activity level of the enzymes involved with a reaction, the properties of the enzyme and the concentration of reactants and products (Nielsen, 2003). We have seen that the enzyme concentration is regulated by gene expression, translation and post-translational protein modifications (Figure 3). The catalytic property of an enzyme is often fixed when determining the flux of a reaction. The concentration of reactants and products is determined by the rate of other reactions forming a dynamic system. Measuring fluxes directly is hard (Emwas et al., 2022). The gold-standard for measuring metabolic fluxes is via ^{13}C isotope analysis. Enzymatic reactions rearrange carbon atoms in a specific pattern; therefore, a labelled substrate can be used to find the contribution of specific pathways to intracellular fluxes (Antoniewicz, 2018). Since these measurements are often not available the steady-state assumption was created (Reimers & Reimers, 2016). In this assumption, we assume that the intracellular flux equals the extracellular flux, which hold under most environments. Based on the simulation of the Michealis-Menten equation, enzyme concentration can be altered which allows a metabolic engineer to observe the effect of changing that enzyme (Antolin & Cascante, 2021). Finding the values for the constants in the Michealis-Menten equation requires the measurement of the rate of products forming *in vivo* organisms.

Learning algorithms

Data-driven models use learning algorithms to learn a predictive model for a certain target variable. We introduce the basic concept behind the learning algorithm used for predicting the metabolome and show some relevant biological examples where the algorithm was successful in its prediction task.

Algorithm	**Description**	**Biological usage**
Elastic net	The elastic net algorithm combines LASSO and ridge regression in a	The elastic net algorithm can be used to predict the k_{cat} parameter of genome-

	parameterized fashion (Zou & Hastie, 2005). It is a linear model which can use the ordinary least squares algorithm to estimate the weight (β) for each variable (X). The calculated weights are penalized using the penalty function from ridge regression and LASSO. The contribution of each penalty function is determined using the λ parameter	scale metabolic models (Heckmann et al., n.d.). Using a diverse set of features extracted from the different datasets the elastic net was used to predict the k_{cat} of *in vitro* enzymes.
Random forest	Random forest uses a bagging strategy to combine multiple de-correlated trees and averages them together to improve predictive performance. The idea of bagging is to average many noisy, approximately unbiased models. Trees are ideal candidates for bagging as they can capture complex interactions in a dataset with sufficient depth and are relatively unbiased.	A random forest was applied to find the combination between sequence and biophysical parameters that maximize the dynamic range (DR) of a riboswitch (Groher et al., 2019b). Riboswitches are RNA sensors that regulate gene expression by interacting with the environment (Kavita & Breaker, 2023). Maximizing the DR is a complex task often executed in a trial-and-error fashion by genetic engineers, therefore having an algorithm that can predict the optimal riboswitch is vital (Groher et al., 2019b)
Support vector machine	Support vector machine aims to identify a hyperplane that optimally fits the dataset. It aims to maximize the distance between the hyperplane and a set of samples. A tube is created around the hyperplane, points that fall inside the area of the tube are considered well-predicted, while outside are considered poorly predicted. A kernel can be used to transform the linear-based algorithm into a non-linear-based algorithm.	Enzymes that stay active at high temperatures are relevant to the biochemical industry. To test the temperature resistance of an enzyme the optimal growth temperature (OGT) is often used as a measure of enzyme stability. However, the OGT needs to be experimentally validated and requires a temperature-controlled lab setting, which makes it infeasible to have the OGT measured for a large variety of species. The random forest algorithm can be used to predict the OGT of a specie based on the
Neural network	A neural network consists of three sets of neurons: input neurons, hidden neurons, and output neurons. The input of each neuron (x_i) is multiplied by a weight (w_{ij}) and summed up, then a bias is added (b_i) and finally passed through an activation function (act). The architecture of the neural network is determined by the size of the input layer, the size and depth of the hidden	Metabolic engineers created a new strain for *S. cerevisiae* that showed a 2.42-fold titer improvement in the production of violacein (Zhou et al., 2018). A neural network was used to suggest the expression levels for heterologous enzymes within the new strain, using the neural only (2%-5%) of the search spaces suggested by YeastFab had to be evaluated.

	layers, the number of outputs, the activation function and connectivity between neurons.	
Graph neural networks	Graph neural networks (GNNs) are a special form of neural network that works directly on graph-based input data. The tasks that can be solved by GNNs can be categorized into node-level prediction, edge-level prediction, and graph-level prediction. A GNN consists of multiple multi-layer-perceptron (MLPs), message-passing layers (MPLs) and pooling operations. A GNN aims to find a node/edge/graph level embedding that minimizes a loss function.	The metabolic state has often been linked with diseases like cancer. It is well understood that cancer rewires cellular metabolism to support rapid proliferation. A novel graph neural network architecture was used to predict the flux of each metabolite using scRNA data. A neural network was used to simulate the Michaelis-Menten equation for each metabolite, via message passing on the stoichiometric matrix the fluxes of all reactions were aggregated (Alghamdi et al., n.d.).

Explanations

We discussed the different machine learning algorithms that are to be used within this work. Here we will introduce the different explanation algorithms that can be used agnostically on the learning algorithms. Model agnostic explanation techniques are especially interesting as their flexibility allows for each integration with existing machine learning pipelines.

*Table 1: **Overview of metrics for the surrogate model used for evaluating generated explanations** (Interpretable Machine Learning, n.d.). The black-box model refers to the model that needs to be explained by a surrogate model. A black box might have some internal explanation components but those are not to be used by the surrogate explanation model.*

Metric	Description
Accuracy	When generating an explanation, we want the explanation to be as accurate to the real-world data as possible. Accuracy in this case means, can a surrogate model predict the testing dataset. The accuracy is measured relative to the black box model that needs to be explained.
Fidelity	Fidelity measures how well the surrogate model represents the black-box model. High fidelity is important because when the surrogate model does not represent the black box model, the explanation is useless. Fidelity can be measured on a global, local and instance level and depends on the explanation algorithm that is being used.
Consistency	Consistency refers to whether the explanation stays the same between different machine learning algorithms.
Stability	For similar instances given a fixed black box model does the explanation change relative to the change in prediction. A stable explanation can especially be useful in the context of metabolic engineering as we do not expect large variations in metabolite concentration with small fold-changes.
Comprehensibility	How well does the metabolic engineer understand the generated explanation. It is related to the size of the explanation and how well the target audience can understand the behaviour of the model.
Certainty	Does the explanation reflect the certainty of the model?

Degree of importance	Does the explanation reflect the importance of the feature? Can the metabolic engineer extract the most important enzymes from the explanation
Novelty	Is the prediction well supported by the training instances for which the model had high accuracy?
Representativeness	What is the scope of the explanation, does it encompass the entire model or a single instance?

Based on the given criteria available, the criteria that are most interesting for metabolic engineers are accuracy, fidelity, and degree of importance (Table 1). The accuracy measure is important as it measures the quality of the surrogate model. A low fidelity would indicate that the surrogate does not reflect the black box model that the metabolic engineer is working on when creating a new enzyme configuration. The degree of importance is important as there exist many different target enzymes and the explanation should limit to only the most important enzymes that led to a metabolite configuration.

Two important algorithms for model-agnostically generating explanations are the LIME and SHAP algorithms. Both algorithms can be used from generating instance-level explanations of a dataset. However, how the explanations are generated fundamentally differs between the algorithms.

For the explanation of an instance in a regression task, the LIME algorithm perturbates the instance and applies a weighting based on the distance to the original instance (Ribeiro et al., 2016). Then a linear model is fit on the perturbated dataset, from which the contribution of each feature can be estimated. The process requires only a single sample, which makes this algorithm effective on large datasets (Ribeiro et al., 2016). The LIME algorithm only achieves local fidelity as it only samples in the neighbourhood of the target instance. Due to the usage of a weighted linear model, normally implemented as LASSO regression, the number of features used in the explanation can be precisely determined. The SHAP algorithm uses an approximation algorithm to estimate the true Shapley values of a feature, as directly calculating the Shapley values would be an expensive operation. A Shapley value can be used to determine the contribution of a particular feature and is based on cooperative game theory (Lundberg et al., n.d.). For the efficient estimation of the Shapley, a subset of the possible coalition vectors is sampled from the coalition space. A priority is given to coalitions that give the most information of a feature. Based on the newly created samples a weighting function is applied that applies a higher weight to samples with the informative coalition. Finally, a weighted linear model is fitted on the dataset, where the coefficient of the linear model is the estimated Shapley values (Lundberg et al., n.d.). A unifying characteristic between the LIME and SHAP algorithms is both algorithms present the explanation as a linear model.

Results

In this section we will present the result of the analysis conducted on the proteome and metabolome dataset. This section is divided into three sections: first we will show the predictive performance on the baseline dataset using the four machine learning architectures. Next, we evaluate the performance on the datasets where additional biological information was included. Finally we present the result of explanation algorithms using the optimal setup from the previous two sections.

Support vector regressor as optimal model architecture for the baseline dataset

The baseline dataset was constructed to measure the predictive power of proteomics data. A variety of machine learning algorithms have been used to predict the relationship between the proteome and metabolome. We have evaluated the predictive power of the elastic net, Random Forest, Support Vector Regressor and multilayer perceptron, algorithm. For each machine learning algorithm, there is a trade-off between predictive power and the number of required samples for training. Since the baseline dataset has a relatively low number of samples (97), we expect that the less complex models have a better performance. We evaluate the baseline performance under the four different scenarios, the all metabolite, single metabolite and leave-one-metabolite-out and blind data preparation. First, a grid search of the free parameters of each model architecture was performed to find the optimal hyperparameters for the architecture (Methods 24). The set of parameters was evaluated using a 10-fold cross-validation strategy with the mean-squared error as a scoring function. The machine learning models are retrained 100 times using the optimal hyperparameters to account for the stochastic nature of the random forest and multi-layer perceptron, using different train-test splits. The mdAPE error is used as it is independent of the mean of the target variable, which makes it ideal to make a quantitative comparison between different groups.

We found that there is a significant difference between the different data strategies (one-way ANOVA $P < .001$). Further analysis showed that models trained under the leave-one-metabolite-out scenario performed significantly ($P < .001$) better than the models trained in other scenarios (Figure 5, Panel A). To investigate the optimal architecture for the baseline dataset, we need to filter the results based on the leave-one-metabolite-out scenario. We found a significant (one-way ANOVA $P < .001$) difference in predictive performance between the different machine learning algorithms. The distribution of errors for each learning algorithm is compared with a t-test and we found that the SVR algorithm performed significantly better than the other algorithms (Figure 5, Panel B). It must, however, be noted that for the *3pg;2pg* metabolite the Random Forest has a better performance and for the *dhap* metabolite the MLP algorithm has the same error.

The result indicates that there is a non-linear relationship between the fold-change in enzyme abundance and the fold-change in precursor metabolite concentration. We found that the Support Vector Regressor was the optimal model architecture for predicting this relationship. Since each dot effectively represents a separate train-test split, which is the same when running the leave-one-metabolite-out scenario. We can see that there is large variability in the model performance for both the Random Forest and MLP model architecture. This can only be attributed to the inherited randomness of those models as all other factors are fixed between the experiments.

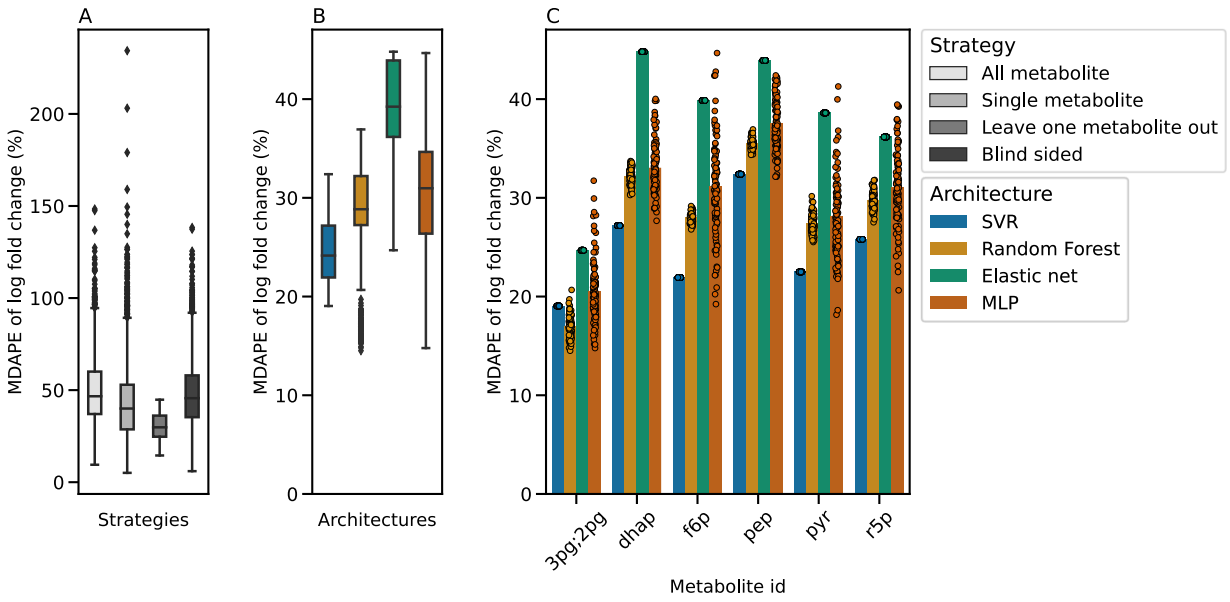


Figure 5: Result analysis of the baseline model. A boxplot of the model performance under different learning scenarios is shown for the baseline model (Panel A). Data presented in the leave-one-metabolite-out scenario reach optimal performance given the baseline dataset. In panel B we see the effect of the model architecture on the baseline dataset, given that the leave-one-metabolite-out scenario is used. Immediately we find that the SVR model architecture is the optimal architecture for this dataset. In panel C the model architecture per metabolite is shown in a bar plot using the leave-one-metabolite-out scenario. Again, we observe that the SVR is the optimal model architecture for predicting each metabolite.

Prior biological information does not improve model performance

Protein interactions play a large role in cellular metabolism (Pandey et al., 2017). We, therefore, hypothesized that creating a dataset that reflects those interactions could be beneficial in the effort of predicting precursor metabolic fold-change given a kinase knockout. We were interested in finding if adding this information would lead to better-performing models compared to the baseline dataset. We expect that adding PPI interactions could positively affect the model performance as through the construction of the dataset the number of instances is increased, the number of features is decreased, and data is enriched (Methods 26). To make a fair comparison between the baseline dataset and the PPI dataset the same model architecture was used for predicting the fold-change in the metabolome. Further, the same test instances were used for the final evaluation of the model.

In our analysis of the dataset incorporating PPI information, we found that the SVR architecture has the best performance. We evaluated the different scenarios and found that the leave one metabolite out approach led to significantly ($P < .01$) higher average performance as measured by the mdAPE (Figure 6, Panel A). The support vector regressor has a significantly ($P < .001$) better performance than the other algorithms on average over all the metabolites (Figure 6, Panel B). It must be noted that for the *3pg;2pg* metabolite the MLP has a slightly better performance than the SVR, however on all other metabolites the SVR is the best model architecture (Figure 6, Panel C).

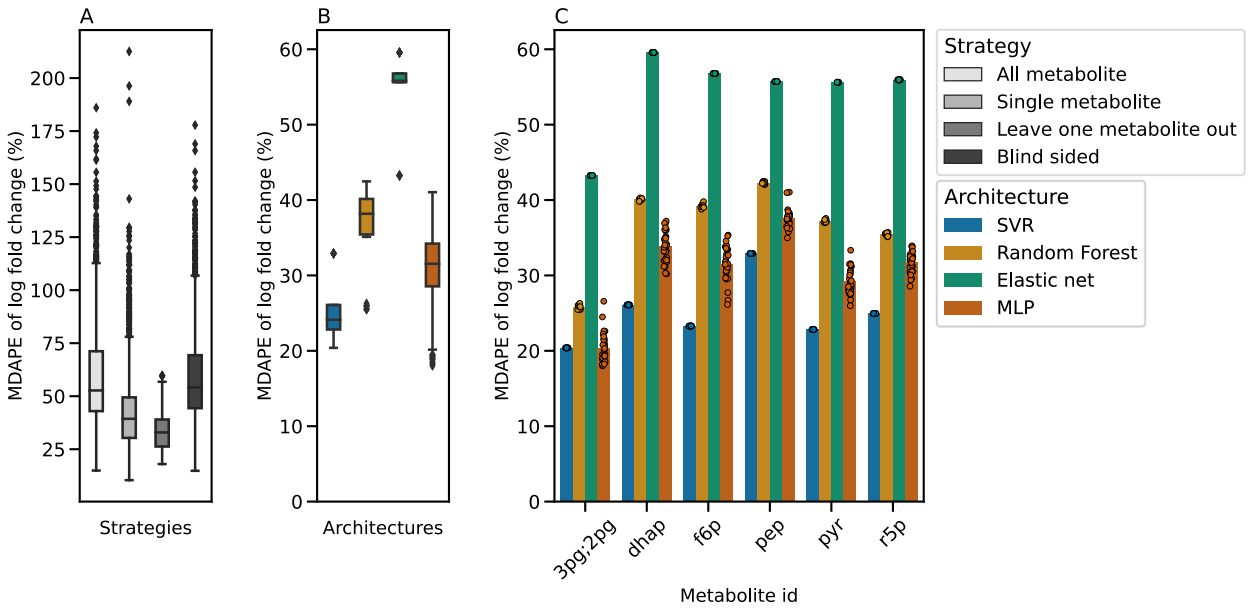


Figure 6: Performance overview of the machine learning algorithms trained on the PPI dataset. The leave-one-metabolite-out scenario has the best overall performance when compared to the other strategies (Panel A). We found that the SVR architecture has a better performance than the other trained algorithms on the PPI-transformed dataset (Panel B). This claim is further reinforced when we observe the architecture performance per metabolite, we find that the random forest architecture has the performance on all metabolites.

For the next experiment, we added stoichiometry to the dataset. Like adding PPI interactions, we expected that adding stoichiometry as additional information allows the model to extract information from direct relationships between enzymes and metabolites. This direct modelling approach could direct the data-driven models to a more biologically sound model. We aimed to identify if combining stoichiometry with the proteomics and metabolomics information would lead to a more biologically informative model. This might ultimately lead to a more accurate model in predicting the metabolomic fold-change given a perturbation in the proteome. First, the standard machine learning algorithms were evaluated using the four scenarios then due to the construction of the dataset we evaluated a graph neural network. For the graph neural network, we tested two architectures together with the four scenarios. As baseline model architecture Graph attention (GAT) layers are chosen, as these architectures perform better than other graph message passing layers (Veličkovi´veličkovi´c et al., n.d.). GATs make use of a learnable attention mechanism to extract useful information from the surrounding nodes.

Similar to the baseline and PPI datasets we find that data presented using the leave-one-metabolite-out scenario leads to a significant ($P < .001$) better result (Figure 7, Panel A). Using the leave-one-metabolite-out scenario as a filter we find that the SVR model architecture has significantly better performance (Figure 7, Panel B). Interestingly we see that the graph neural networks have the worst performance overall architecture. Finally, we compared the model performance per metabolite, and we observe that the F6P metabolite was best predicted by the model and a similar error was measured for the metabolites (Figure 7, Panel C).

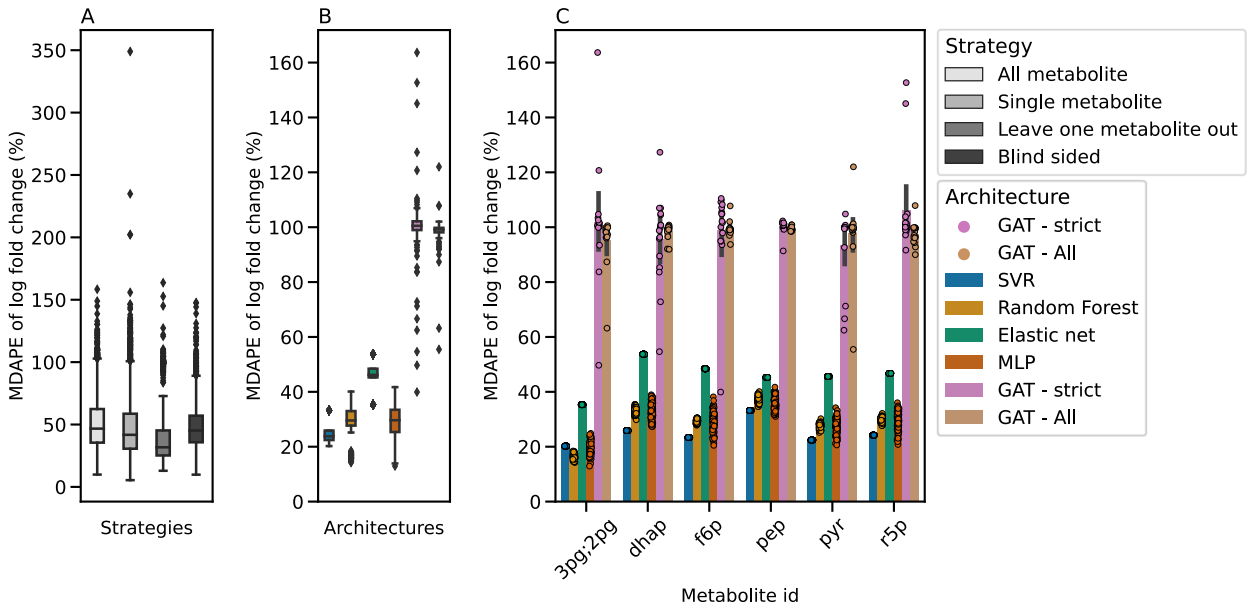


Figure 7: Performance overview of the model performance using stoichiometry as additional biological information. Information presented using the leave-one-metabolite-out scenario had a significantly higher performance (Panel A). We found that the SVR has the best average performance when compared to the other strategies (Panel B). Finally, when we zoom into the model performance per metabolite we find that the SVR model architecture still has the best average performance for all metabolites except for 3pg;2pg (Panel C).

Adding additional biological information following the procedures described for the PPI and both stoichiometry datasets did not result in the increased performance that was anticipated (Figure 8). Using the mdAPE as a model performance metric we found that the baseline dataset has the best-performing model when comparing the optimal model architectures from each dataset. The relative performance between the metabolites stayed similar between the different datasets. We still observe that the PEP metabolites were the hardest to predict in all datasets and the 3pg;2pg and F6P metabolites were more trivial to predict.

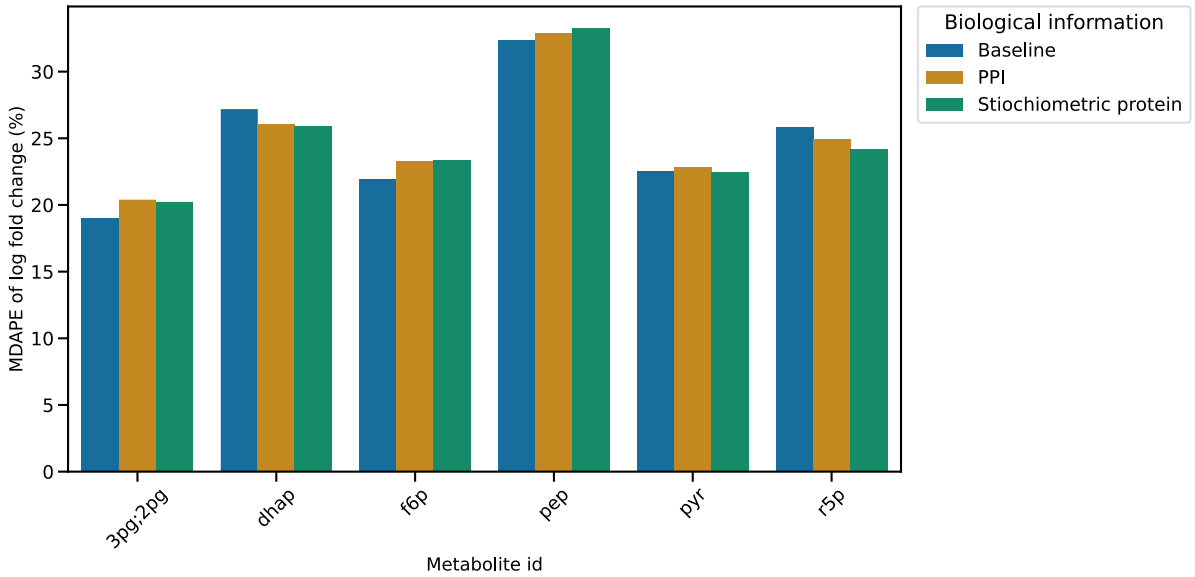


Figure 8: Global overview of the model performance using the mdAPE metric to rank the effect of adding biological information. Immediately we can observe that adding additional biological information does improve the model performance given the

evaluation metric. We observe that the error measured over the three datasets stays the same for each metabolite. To fairly compare the different datasets we pick the leave-one-metabolite-out scenario and SVR model architecture for each dataset.

Projecting explanations on yeast stoichiometry

Based on the analysis of the model performances, we found that the features of the baseline dataset were the best at predicting the metabolite fold-change. Using the SVR as a model architecture and leave-one-metabolite-out scenario we achieved the highest predictive performance. To exactly find the features that were learned from this dataset we applied the SHAP and LIME algorithms. Using both algorithms the top hundred most important features for each instance in the test set were determined. The surrogate models of SHAP and LIME were retrained ten times to ensure that important enzymes were consistently selected. To find if important enzymes are generally preserved between the algorithms, we first evaluated the overlap between the top twenty features. We found that there is at most 60% overlap between the two algorithms, indicating that the selection of an explanation algorithm has an impact on the final presented feature set (Figure 9). We identified that the amount of overlap between the two algorithms is dependent on the data preparation strategy. Furthermore, there is also a difference in the overlap between the different metabolites, notably the enzymes marked for phosphoenolpyruvate (PEP) seem especially well preserved over the three data preparation strategies.

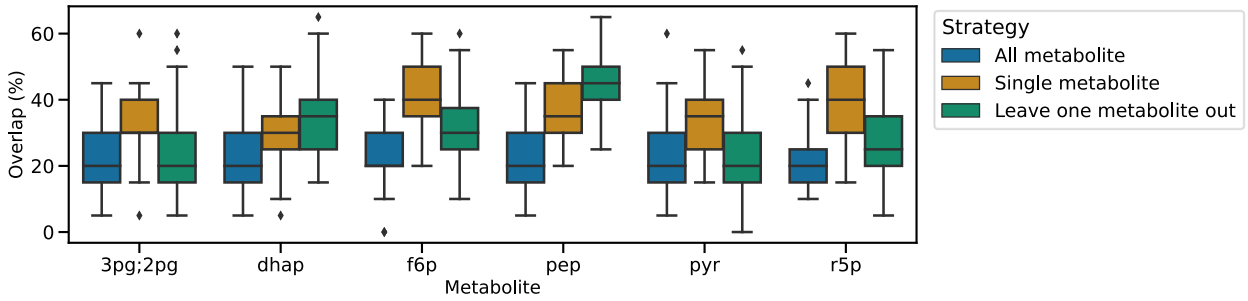


Figure 9: Overlap of the first twenty features between the enzymes that the LIME and SHAP algorithms predict. We find that there is at most 60% overlap between the SHAP and LIME algorithms in the leave-one-metabolite-out scenario, interestingly in the single metabolite strategy the SHAP and LIME algorithms tend to have the most overlap while for the leave-one-metabolite-out strategy, there is a moderated amount of overlap.

Since there is no 100% overlap between the enzymes marked as important by the SHAP and LIME algorithms, we can determine the optimal algorithms using the accuracy and fidelity scores (Validating the models and explanations). We find that the SHAP algorithm has a higher fidelity, but lower accuracy compared to the LIME algorithm given the baseline dataset, leave-one-metabolite-out data preparation scenario and SVR model architecture (Figure 10, Panel A, Panel B). This pattern is preserved across the different metabolites in the dataset. Network analysis on the explanation generated using the SHAP algorithm indicates that for the prediction of a metabolite, a specific set of enzymes is found to be important, however, some enzymes are globally important to all metabolites like the PRE1 enzyme (Figure 11).

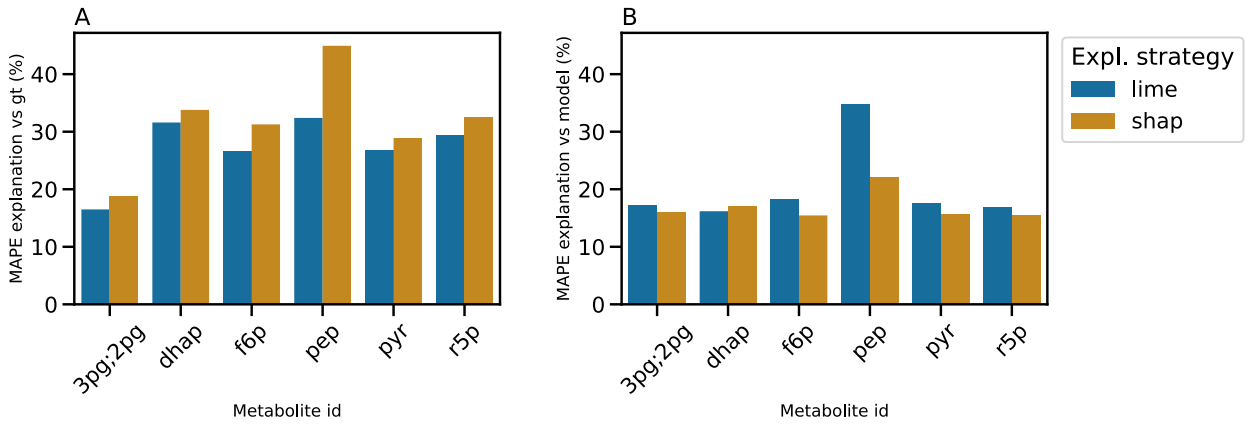


Figure 10: Metrics collected on the SHAP and LIME algorithm. For the baseline dataset, using the SVR model architecture and leave one metabolite out strategy the accuracy and fidelity of the explanation algorithm is calculated. We find that the LIME algorithm has an lower error when the algorithms predictions are compared to the groundtruth (accuracy, Panel A) and an higher error when the predictions are compared to the models' predictions (fidelity, Panel B).

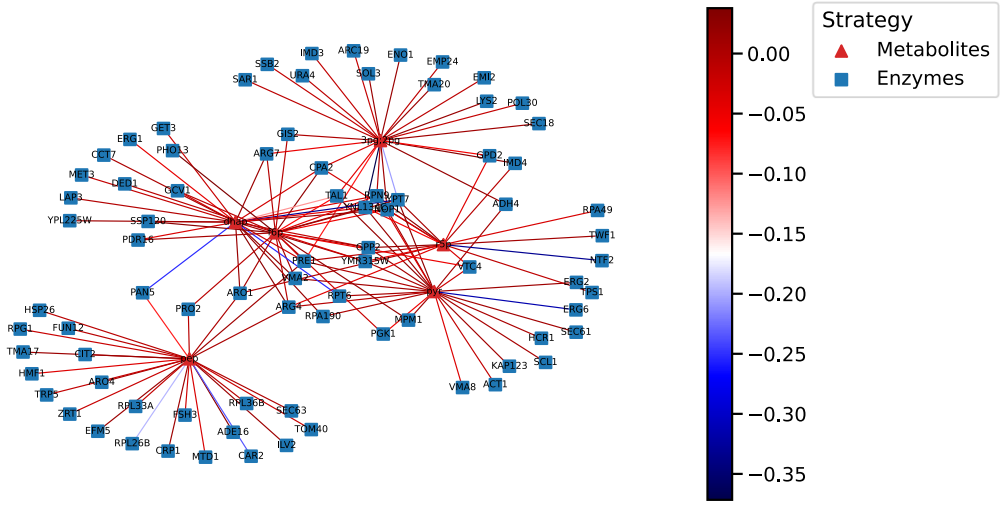


Figure 11: Graph constructed based on the top twenty most important enzymes found using the SHAP explanation algorithm. There is a limited set of enzymes that are associated with two or more metabolites. The edge colour is associated with the average SHAP value found for this edge for the different knockouts in the dataset. A notable example of an enzyme that is associated with multiple metabolites is PRE1. It is associated with all six metabolites and is a subunit in the 20S proteasome.

To allow metabolic engineers to use the explanations generated from the analysis pipeline, we created a strategy to project the contribution of enzymatic fold-changes on a reaction map. To evaluate the strategy, we investigated the effect of kinase knockout GCN2 on the pyruvate concentration. GCN2 is an enzyme associated with amino acid starvation (Wang et al., 2017). It reacts to uncharged tRNA accumulating in a cell and inhibits translation while activated, but simultaneously promotes translation of the set of mRNAs for amino acid biosynthesis and transport (Natarajan et al., 2001). We found that the GCN2 knockout led to a significantly higher fold-change in the first thirteen enzymes (Figure 12, Panel B). Notably, we found an enormous difference in concentration between the average fold-change after knockout for the SSE2 and HSP82 enzymes, both associated with heat shock responses (Hideyuki et al., 1993).

The concentration of pyruvate decreased with a fold-change of 1.44 (Figure 12, Panel C), using the support vector regressor retrained on the baseline dataset with a leave-one-metabolite-out scenario, we predicted a fold-change of 1.04. We then used the SHAP algorithm to find the 100 most important enzymes and projected them on the IMM904 yeast metabolic model.

First, we aimed to explain the fold-change through metabolic enzymes which play only a small role in predicting the fold-change of pyruvate (**Error! Reference source not found.**, Panel E). We find that only SEC61 was in the top twenty most changed enzymes when explaining the prediction using a SHAP waterfall plot (Figure 12, Panel D). Based on the SHAP explanation we identified that VTC4, is responsible for regulating the polyphosphate concentration in yeast (Tomashevsky et al., 2020), and PRE1, is responsible for protein degradation (Heinemeyer et al., 1991).

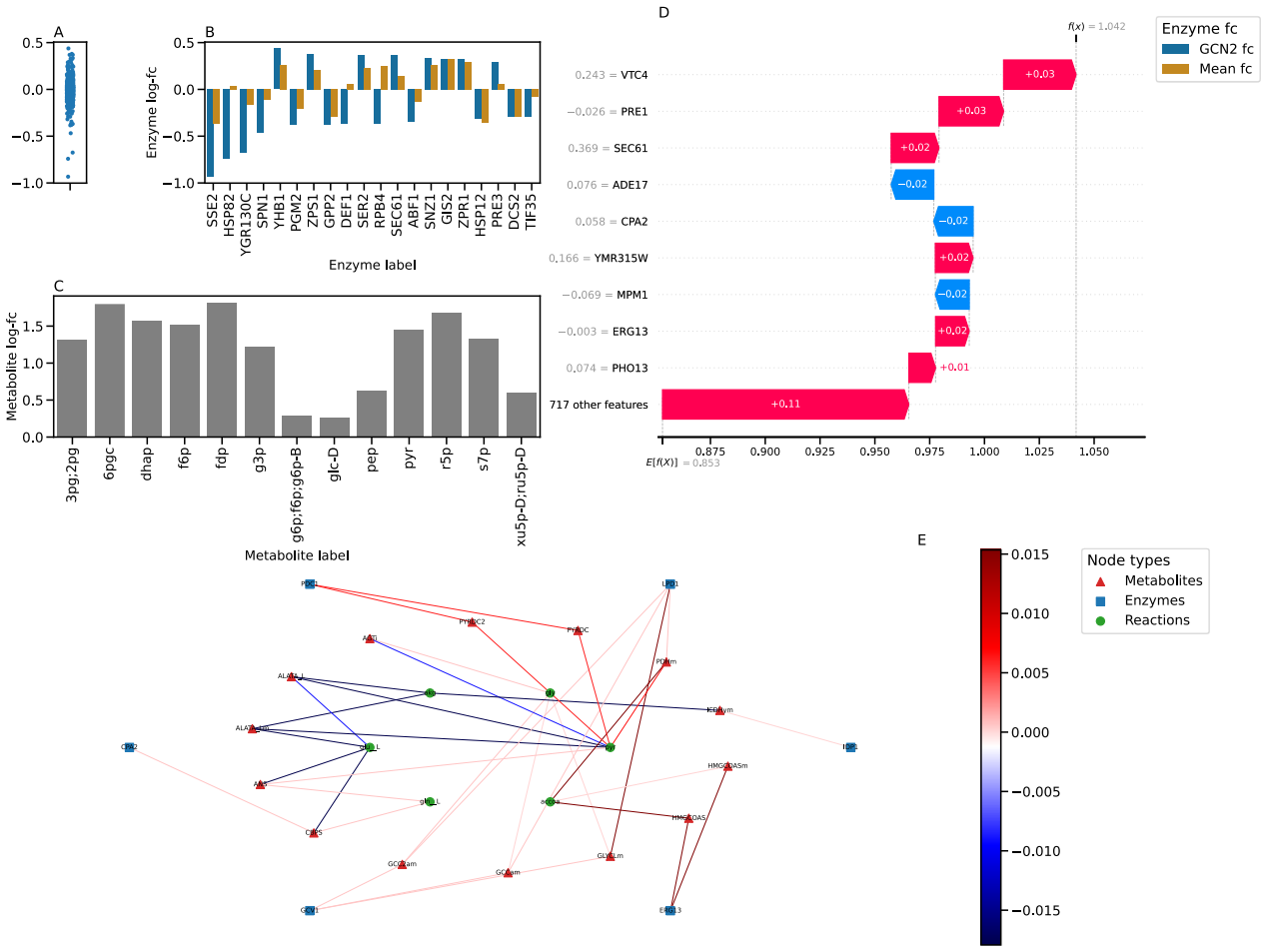


Figure 12: Strip plot of all enzyme fold-changes measured under the GCN2 knockout. We observe a large, centred split around a .5 up/down regulation (Panel A). In the top twenty most changing enzymes given the GCN2 knockout compared to the average fold-change of that enzyme, we found that most of the first thirteen enzymes have a significantly higher change when compared to their average fc (Panel B). In the metabolic fold-change of all measured metabolites of the GCN2 knockout, we observed that all measured metabolites express a decrease in concentration (Panel C). Waterfall explains the prediction of the pyruvate fold-change using the SVR, baseline dataset under the leave-one-metabolite-out scenario. The predicted fold-change was 1.042 while the measured fold-change was 1.44. The waterfall plot shows the buildup of the predicted fold-change, notably, we find that functional enzymes were the major contributors to the final prediction (Panel D). Graph analysis of the GCN2 knockout. Using the IMM904 yeast metabolic model, we could project the metabolic enzyme on the metabolic network of a yeast cell. We found that only a limited number of enzymes were able to be associated with the fold-changes in pyruvate. For visualization purposes, enzymes associated with reactions that consume or produce energy (ATP, ADP, AMP) are omitted (Panel E).

Discussion

Effective model architecture for predicting the metabolome

We aimed to investigate the optimal model architecture for the baseline dataset and hypothesized that probably the random forest algorithm would have the best performance. However, we found that the support vector regressor algorithm was the optimal algorithm for predicting the fold-change in the metabolome. This is in contrast with the expectation as in other related studies, tree-based algorithms are often found to be able to best predict the relationship between the proteome and metabolome (Zelezniak et al., 2018).

We hypothesized that the location in the glycolysis pathway has an impact on the model performance, as the signal measured for more downstream metabolites might be harder to measure. However, with closer inspection of the glycolysis pathway, we found that the position in the pathway was not related to the error of certain metabolites. Alternatively, we tried to explain the difference in predictive model performance per metabolite by investigating the distribution of training and testing samples, however since the leave-one-metabolite-out strategy was the optimal splitting strategy the difference in error could not be attributed to the train test split. Using this splitting strategy, the knockouts have been observed in the training dataset and the model is evaluated on the unobserved metabolites.

Next to answering the research questions we have introduced a novel way to extract more information from the limited number of instances available. Due to the different data pipelines adopted in this study, we found that the machine learning models trained were dependent on the way the instances were presented. This immediately also shows the largest shortcomings in this work, since there is only a limited number of samples available not the entire range of possible outcomes of all perturbations of the enzymes can be predicted. This is a limitation of using machine learning within metabolic engineering as machine learning algorithms are generally limited to the range of their training data.

Based on these findings, practitioners should consider increasing the number of samples such that more sophisticated machine learning algorithms can be applied that generalize better on more complex perturbations. The field of metabolic engineering can greatly benefit from machine learning as it would remove the need for making costly kinetic models, however, more diverse samples are needed to achieve the true potential of machine learning algorithms. Furthermore, the field can use a standardization of data transformation algorithms such that constraint-based, kinetic-based and machine-learning algorithms can be compared more fairly.

Additional biological information to improve model performance

The findings of this study suggest that adding additional biological information to the dataset does not improve the predictive performance of the learning algorithms. We measured the machine learning performance for predicting the metabolome when additional biological information was included. In total three separate experiments were conducted. PPI information was to reduce the feature space of the dataset. The stoichiometry of *S. cerevisiae* was used to construct the protein-protein bipartite graph on which proteomics and metabolomics could be projected. The two experiments were then compared to the baseline experiment, and we found that additional biological information did not significantly improve model performance.

We expected that integrating data from different datasets would improve the model performance as multi-omics studies often tend to have a higher model accuracy when compared to their

single omics counterparts (Yao et al., 2015). Ultimately the PPIs were only used for reducing the feature space and did not necessarily add new information to the model, therefore a comparable result using the baseline dataset could be achieved if the model was able to use filter irrelevant features. For the experiments conducted with stoichiometry as additional information we can conclude that there are too few samples to effectively use the GNNs as a model architecture. Therefore the SVR was selected as the optimal model architecture.

An interesting observation is that for the baseline, PPI and stoichiometry experiments we found that the leave-one-metabolite-out splitting strategy was the optimal splitting strategy. This suggests that the machine learning models can generalize the relationship between fold-changes in the proteome and fold-changes in the metabolome. Furthermore, we found that the SVR had a significantly better performance than the other learning algorithms.

We believe that adding additional biological information might have an impact when integrated with another strategy. In this study, we used biological information mostly to filter the existing dataset. However, in another setup, a model per class of biological information might be trained which can summarize the information found in that class. The summarization of each class might then be aggregated to make the final prediction for a perturbation.

Extracting information using explanation algorithms

During the experiments on explainability, we found that the SHAP algorithm has a better fidelity with slightly lower accuracy. This result led to the conclusion that the SHAP algorithm was the optimal explanation algorithm, for explaining the predictions resulting from the optimal models trained using the leave-one-metabolite-out splitting strategy on the baseline dataset. However, it must be noted that in the evaluation we found no significant difference between the algorithms meaning that SHAP was selected based on its slightly better performance on the fidelity metric. Using the LIME algorithm, a different explanation could be generated for each instance in the testing dataset.

We found that enzymes directly involved with the target metabolite were not the major contributors to finding prediction results. Rather we found that enzymes that were further away from a particular metabolite had a higher contribution to a particular concentration. This might be attributed to the regulatory interactions of the cell. For predicting the metabolic fold-change using the proteome they found that the dataset including enzyme two or three hops away from the target metabolite had the highest predictive power (Zelezniak et al., 2018). The results demonstrate that there is a small set of preserved enzymes that are important for multiple metabolites. These enzymes are related to some key regulatory functions like proteasome which is responsible for regulating the concentration of proteins (Tanaka, 2009).

GCN2 is preserved enzyme over a variety of species, it has been associated with the sensing of amino acid deprivation. When there are too few amino acids GCN2 would be activated and modulate amino acid metabolism, while reducing the production of proteins. In yeast any deficiency in amino acids triggers the activation of GCN2, it can therefore be argued that knocking GCN2 out is detrimental to cellular proliferation. As amino acids play a vital role in the survival of the cell, not being able to control the concentration of amino acids would be a large handicap for the cell. We found that the GCN2 knockout led to a decreased concentration in all measured metabolites, and a spike in the concentration of certain enzymes

We theorize due to the GCN2 knockout, the cell will consume all metabolites to produce enzymes as the regulatory process that limits this process has been knocked out. We predicted a fold-change of 1.04 for pyruvate, while the measured fold-change was 1.44. We found that the prediction could only limitedly be explained via metabolic enzymes. This might be attributed to not having included any metabolites directly associated with the amino acid metabolic pathway. We could however explain a large part of the prediction via the non-metabolic enzymes VTC4 and PRE1. VTC4 is responsible for the synthesis of *polyP,* which plays an important role in the cell cycle as a phosphate reservoir (Bru et al., 2016) and PRE1 is responsible for protein degradation. These two enzymes are two logical candidates for the algorithm to extract from the dataset and mark them as important.

It might be interesting to experimentally validate the interactions between the amino acid sensing enzyme (GCN2), a decreased concentration of the VTC4 enzyme and the slightly increased concentration of PRE1. We believe that understanding the interaction between these three enzymes might push the knowledge of the working of GCN2.

Conclusion

In this work, we addressed three major topics. First, we investigated what the optimal architecture for the baseline dataset would be then we explored the effect of adding prior biological information to the dataset and how that affected model performance, finally we investigated the effectiveness of generating an explanation for the machine learning models.

Based on the analysis of the model performance on the baseline dataset, it can be concluded that the support vector regressor was the optimal model architecture. This was attributed to the relatively low number of samples found within the dataset. In comparative analysis between the different datasets where prior biological information was added, we found that adding prior biological information does not improve model performance. This can be attributed to the fact that there is a limited number of data points available when adding stoichiometry. The PPI information did not seem to have additional predictive power for predicting the metabolome. Finally, we were able to thoroughly analyse the explanations by analyzing the enzymes that were marked as important. We found that a few preserved enzymes were important for a variety of metabolites.

In this work, we have investigated the usage of machine learning for generating an automatic prediction of the metabolite concentrations after enzyme perturbations. Explanations generated on these models can help guide metabolic engineers as they can directly see the effect of perturbations on the most important reactions within the model.

Methods

A high-level overview of the data analysis pipeline used in this thesis is reported (Figure 13). We show the internal steps of the pipeline with a concrete example.

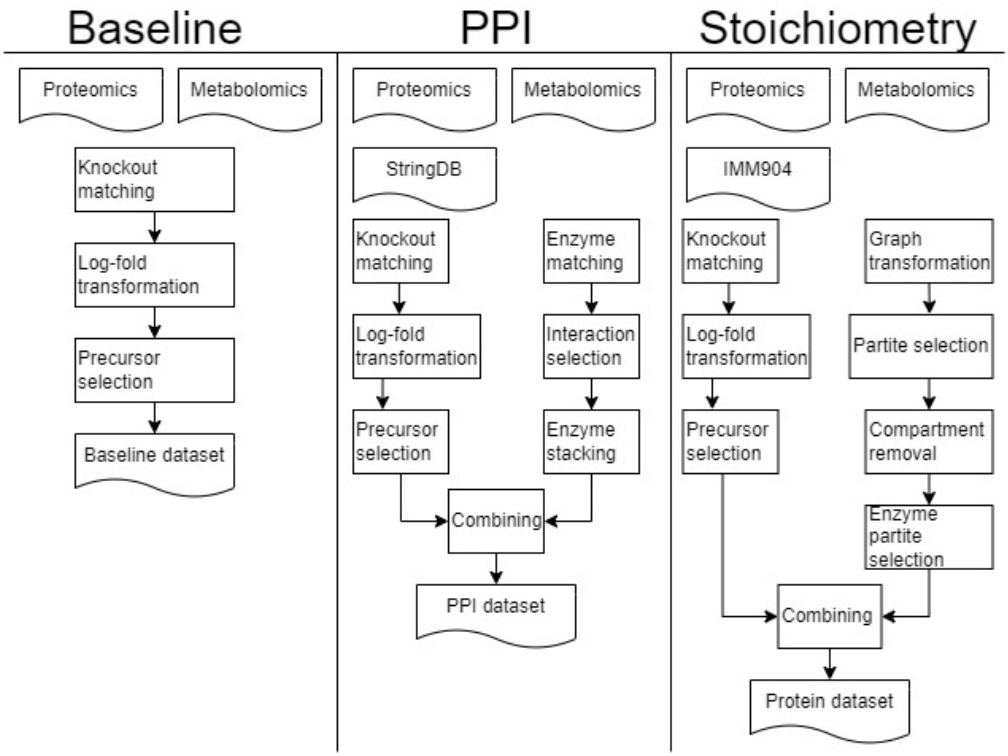


*Figure 13**: High-level overview of the data pipeline for each dataset generated**. A new pipeline aims to add additional biological information to the baseline dataset, such that we can evaluate the biological information separately. For the baseline dataset, the proteomics and metabolomics data are combined. In the PPI dataset, protein-protein interactions are used to filter the dataset. Stoichiometry information can be used to build protein-protein or protein-metabolite graphs on which graph neural networks can be trained. Each pipeline has its respective section within this chapter; however, the protein stoichiometry and metabolite stoichiometry are grouped as they follow a similar pattern.*

Once a dataset is constructed via the data processing pipeline a splitting strategy is applied to form a training and testing dataset. Because of the strategy the dataset might be replicated transformed or filtered, which creates unique scenarios in which the learning algorithms can be evaluated (Table 2).

*Table 2: **Overview of the splitting strategies used to generate the training and testing datasets** from the resulting dataset for each data processing pipeline. Each strategy represents a unique learning task for the model.*

Scenario name	Details
All metabolites	The dataset generated by the data pipeline is stratified according to the metabolite id. 30% of the dataset is used for generating the testing dataset.
Single metabolite	The dataset generated by the data pipeline is replicated for each metabolite in the dataset. Each replicated dataset is then filtered for one target metabolite. For each of the newly constructed datasets a train-test split with a testing percentage of 30% is performed. This will evaluate whether the model performance increases if the model was trained for an individual metabolite.

Leave one metabolite out	The dataset generated by the data pipeline is replicated for each metabolite in the dataset. A train-test split is used to split one metabolite into the test set while the others are used from the algorithm to learn on. This process is repeated for each metabolite within the dataset. With this strategy, we aim to find whether metabolism can be generalized within one learning algorithm and if this is possible predictions can be easily done for new metabolites.
Blindsided	The datasets generated in this data pipeline are similar to the ones created in the leave-one-metabolite-out scenario. However, the algorithms are only trained on a subset of knockouts and evaluated on the left-out knockouts.

First, the three data processing pipelines are explained step-by-step to give a deeper intuition into how each dataset is constructed. Then the validation of the models and explanations are further elaborated on, the final evaluation metric is introduced and how the final explanation is combined to form a singular explanation panel.

Constructing the baseline dataset

The baseline dataset is constructed from the proteomics and metabolomics datasets made available in a previous study where the predictive power of both datasets where researched (Zelezniak et al., 2018). The specification of the dataset is given in (Table 3). Proteomics is the large-scale study of the proteomes, which is the set of proteins produced in an organism. The proteome differs per cell and changes over time and can be used to evaluate the rates of protein production, which is related to reaction flux (*What Is Proteomics? | Proteomics*, n.d.). Metabolomics is the large-scale study of the metabolome and measures the concentration of metabolites within an organism (*What Is Metabolomics? | Metabolomics*, n.d.).

*Table 3: **The dimensions and features of the proteomics and metabolomics dataset**. The proteomics dataset has 98 samples and 728 features. The metabolomics dataset consists of the same 98 knockouts and 50 metabolites.*

Characteristic	**Dimensionality**
Phenotype-enzyme combinations	71148
Enzymes	728
Knockouts	97
Metabolites	50
Precursor metabolites	11
Precursor metabolites full sample	6

The proteome and metabolome datasets are measured for all kinase knockouts, this simplifies the merging operation between the two datasets. The datasets are merged according to the kinase knockout (Figure 14). Next to the kinase knockouts the enzyme and metabolite concentrations were also measured for a yeast wildtype (WT) strain. This WT is the baseline measurement for all kinase knockouts and can be used to find differentially expressed enzymes within the dataset. Learning algorithms require target variables to be generally within the same range of each other, else the model will have a hard time predicting the different concentrations for each sample. A log-fold transformation is applied to bring the proteomics abundance and metabolite concentration column-wise within the same range (Equation 1). After the data transformation, the eleven precursor metabolites available in the metabolomics datasets are selected as the target for the downstream learning algorithms.

$$sample_i = \log_2 sample_{wt} - \log_2 sample_i$$

Equation 1: The log-fold transformation applied for each sample within the dataset. A sample refers to a row spanning the proteomics and metabolomics dataset.

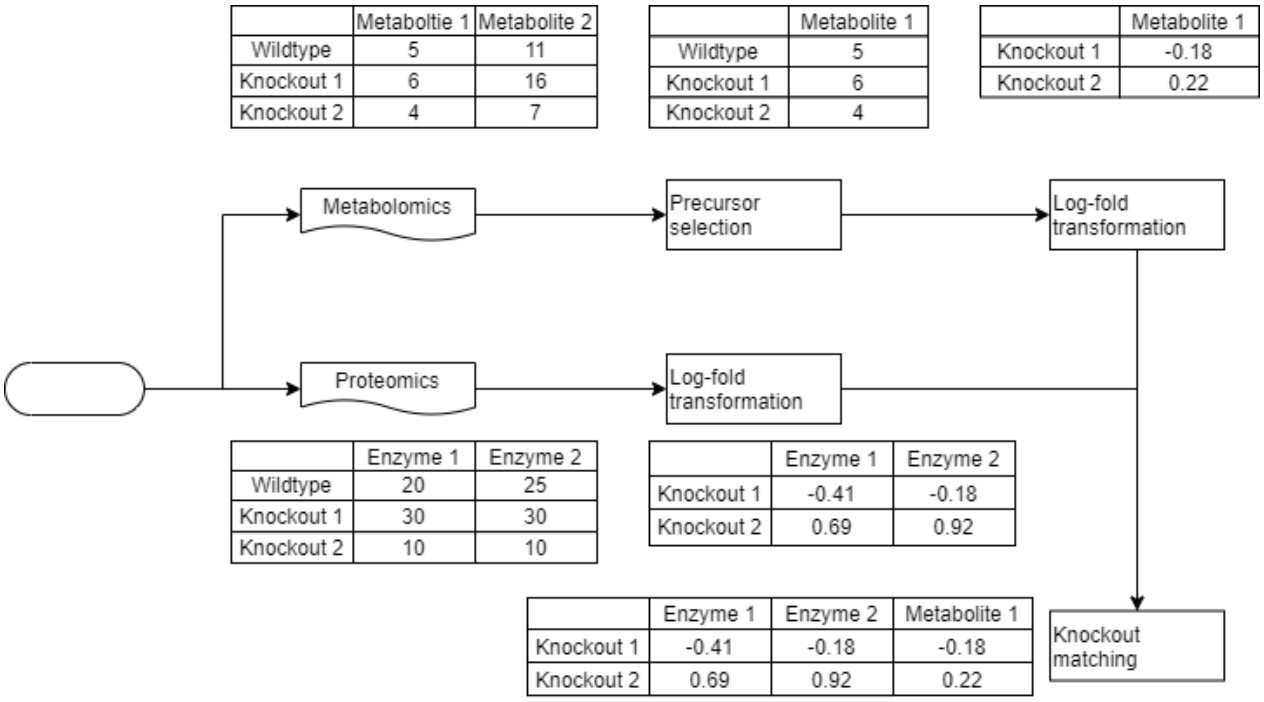


*Figure 14: **Combining proteomics and metabolomics information.** First, metabolite precursors were filtered from all metabolite concentrations followed by a log-fold transformation. Similarly, proteomics was also log-transformed.*

The dataset can now split according to the strategy used for training the learning algorithms. For the baseline dataset RandomForest, ElasticNet, SVM and multi-layer perceptron are selected as learning algorithms as these models have been proven to be successful for tabular datasets. Due to the limited number of samples available deep learning techniques are likely to be underperforming. The hyperparameters for the learning algorithms are specified in (Table 4).

*Table 4: **Hyperparameters of the training algorithms selected for the baseline dataset**. A grid-searching strategy is used to enumerate all possible combinations within a model architecture. This is a feasible approach due to the limited number of samples, a small number of hyperparameters and relatively fast learning algorithms. For each learning algorithm the feature scaling is used as a hyperparameter (MinMaxScaler, StandardScaler) additionally. In total 174 hyperparameter configurations are evaluated using a 10-fold cross-validation strategy.*

Hyperparameter space for the baseline dataset		
Model architecture	Parameter	Values
SVR	Kernel	RBF, Sigmoid
SVR	Gamma	Auto, scale
SVR	Epsilon	0.1, 0.01, 0.001, 0.0001
SVR	C	10, 100, 1000
Random Forest	Number of estimators	10, 25, 50, 75, 100
Random Forest	Criterion	Squared error, Friedman mse
Random Forest	Max depth	5, 10, 20
Elastic Net	L1 ratio	0.01, 0.25, 0.5, 0.75. 1
Elastic Net	Tol	0.01

MLP	Layer sizes	(128, 32, 32), (64, 32)
MLP	Batch size	2, 4, 8, 16

Once the optimal combination of hyperparameters is found for the baseline dataset, a new model is trained according to this set of parameters. This retraining process is repeated to evaluate the stability of the model. The optimal model, the model that minimizes the mean absolute error on the dataset, is stored for further analysis.

Combining protein-protein-interactions with proteomics

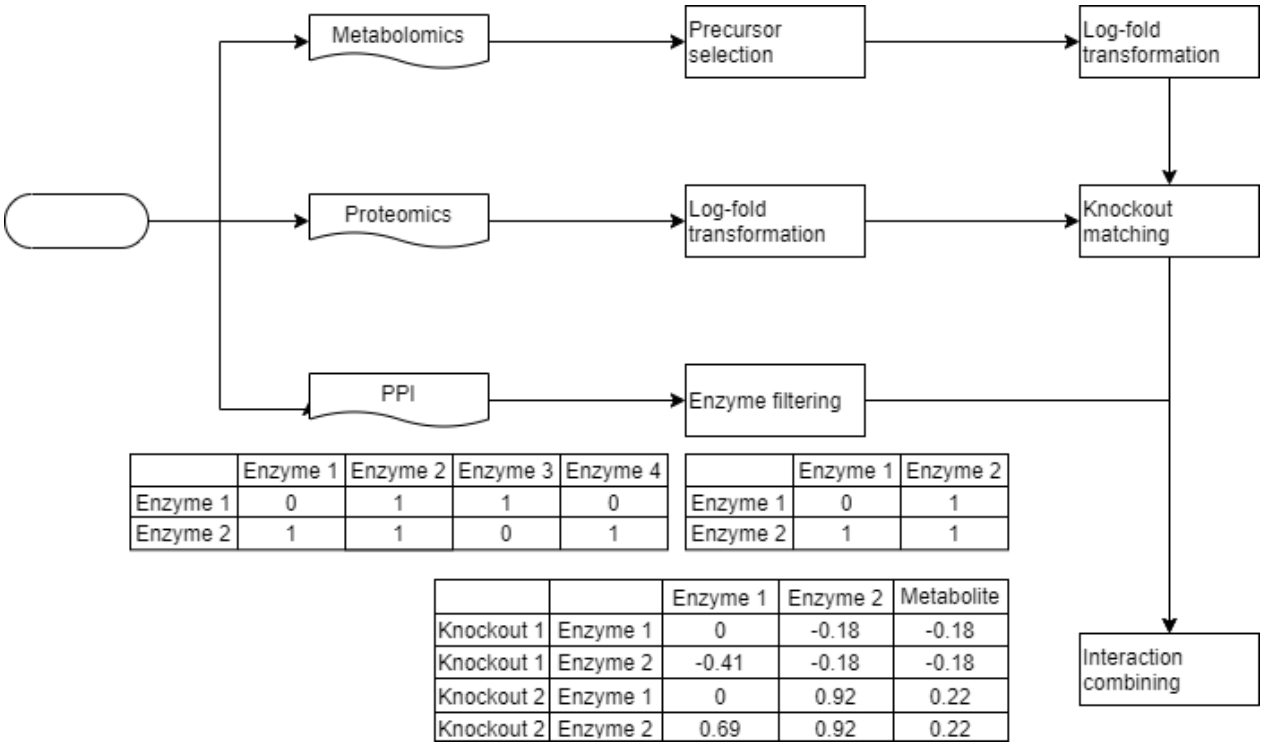
The protein-protein-interaction information is extracted from the STRING database (Szklarczyk et al., 2015), which is a rich database containing experimentally validated interactions and predicted interactions. For each enzyme in the proteomics dataset, its interaction network is downloaded via the STRING API. As species, *S. cerevisiae* is selected to filter interactions that were not predicted or measured for the dataset. A string containing the interacting information is returned, which is saved for further processing.

The protein-protein interactions gathered from the STRING database need to be transformed to fit the proteomics datasets. A graph of interacting proteins for each enzyme in the proteomics dataset is constructed. Graphs with less than thirty interactions are removed to make sure that enough non-zero feature values are available for downstream algorithms. The proteomics dataset is then replicated for each of the available enzymes left over in the filtered list. Enzymes that are no longer represented within one of the graphs in the list of graphs are also removed to remove noise in the final dataset. For each sample in the constructed knockout-enzyme matrix, the fold-change of the interacting enzymes is stored, which creates the final populated data frame.

The balance between the number of informative features and samples is an important property of a dataset, as it impacts the model's predictive power. Adding the PPI information reduced the number of available features while increasing the number of artificial samples. This should improve machine learning performance as the models have an increased number of samples (Azhar & Thomas, 2019). However, due to the increased number of zero values in the dataset, the quality of the data has decreased. The dataset is then split according to the training strategy used for training the algorithm (Figure 15).

The hyperparameters used for training the algorithms are shown in (Table 4). This set of hyperparameters consists of the most influential parameters for their respective algorithms. A standard grid search is applied to find the optimal combination of hyperparameters for each model. The models are trained the minimize the mean squared error (MSE) metric on the training dataset.

Once the model has been trained for each combination of hyperparameters the optimal model architecture and parameters are chosen via the mean absolute error (MAE) on the testing dataset. For each metabolite individually, when the model was trained using the single metabolite or leave-one-metabolite-out strategy else the globally best model is selected using the MAE. Using the optimal set of hyperparameters the model is retrained to evaluate the hyperparameter stability and the final prediction for each sample is generated. The best-performing model according to the testing dataset is stored for the generation of explanations.



	Enzyme 1	Enzyme 2	Enzyme 3	Enzyme 4
Enzyme 1	0	1	1	0
Enzyme 2	1	1	0	1

	Enzyme 1	Enzyme 2
Enzyme 1	0	1
Enzyme 2	1	1

		Enzyme 1	Enzyme 2	Metabolite
Knockout 1	Enzyme 1	0	-0.18	-0.18
Knockout 1	Enzyme 2	-0.41	-0.18	-0.18
Knockout 2	Enzyme 1	0	0.92	0.22
Knockout 2	Enzyme 2	0.69	0.92	0.22

Figure 15: Example of the PPI data transformation pipeline. The metabolomics and proteomics datasets are transformed with the same procedures given in the baseline data pipeline, the resulting example from the knockout matching is therefore used. This dataset is then combined with the adjacency matrix extracted from the PPI database forming the PPI dataset.

Adding stoichiometric biological information

The stoichiometry of the yeast metabolism is extracted from the iMM904 genome-scale metabolic model (Zomorrodi & Maranas, 2010). The stoichiometry is encoded in a matrix where an integer number indicates the consumption or production of a metabolite in a reaction. For each reaction, a list of associated enzymes was constructed when the model was produced. All metabolites within the dataset also have been annotated with the compartment to which the metabolite belongs. The stoichiometric matrix can be viewed as a large tripartite graph where one set of nodes represents metabolites, another set of node's reactions and the final set of nodes enzymes associated with the reaction. Edges are created between the metabolite and reaction nodes if the metabolite is present in the reaction and edges between the reaction and enzymes are created when the enzyme catalyzes the reaction (Figure 16).

Like the other tiers the proteomics and metabolomics are first scaled based on the log-fold transformation, nan-values are to be removed and finally the metabolomics dataset is filtered based on the list of precursors. Since we want to embed the stoichiometric information into the proteomics and metabolomics datasets, the stoichiometric information needs to be transformed to fit the target dataset. The stoichiometric information consists of the relationship between enzymes, reactions and metabolites and is often structured as a metabolite-reaction matrix and reaction-enzyme matrix.

The metabolites from the stoichiometric matrix contain a compartment component, since the used dataset does not contain such a component, it is removed. We assume that the fold-change of the enzymes and metabolites is independent of the compartment. The enzymes that are not available in the proteomics dataset are removed from the stoichiometric matrix to further clean the matrix.

Constructing tabular data

To stay in line with the other tiers we have to create a tabular dataset based on the stoichiometry information of the yeast cell. We have decided to use the enzymes that are in the stoichiometric matrix as a filter for the proteomics dataset. Using the stoichiometric enzymes removes all signalling enzymes as those are not in the matrix. The algorithms are then trained using the hyperparameters described in Table 2.

Constructing a graph neural network

Since the stoichiometric information is encoded in a graph the data can also be used to train a graph neural network. Here we will discuss how the data is prepared for training the graph neural network.

To each reaction within the matrix, the set of associated enzymes is assigned as a proxy. This creates an enzyme-metabolite matrix, which can be represented as a bipartite graph. The protein-metabolite bipartite graph is replicated for each knockout and to each node, the fold-change is assigned given the specific knockout. Then we can create the graph-level prediction task has the benefit of the algorithm not necessarily having to be retrained if a new target metabolite is introduced as the structure of the graph does not change. The node-level prediction task can be transformed into a graph-level prediction task via the projection of the protein nodes. Two nodes that share a common metabolite are connected via edge creating a densely connected graph. The fold-change of the metabolites is stored as a vector with the newly created graph.

The dataset created from the stoichiometric matrix has only a single feature for the downstream algorithm to learn on, together with the limited number of samples this could lead to deteriorated results for the algorithm. To tackle the first problem embeddings generated based on the node2vec algorithm are concatenated to each node in both datasets. A small embedding size of 32 has been selected as the optimal size of the embeddings, as the graphs in both datasets are small yet complex. The node2vec embeddings can be exploited by the downstream algorithm, as the representation of the graph has already been encoded. This approach has been used before for node level prediction task and graph level prediction tasks. For the node level prediction task on the Arvix benchmark dataset algorithms using the node2vec embeddings performed better than the algorithms without and for the graph level prediction task a similar result was observed when predicting the label of a protein from the PPI dataset (Arsov & Mirceva, 2019; Dalmia & Gupta, n.d.).

A last transformation needs to be applied before the data can be used by the downstream analyzation algorithm. For the metabolite-protein dataset we remove all non-precursor metabolites from the graph, together with all the unconnected enzymes creating the precursor-protein dataset. For the protein dataset we remove all protein nodes associated with non-precursor metabolites, creating the filtered-protein dataset. After this transformation two different dataset exists that can be used for training the analyzation algorithm.

To make a fair comparison between the different biological tiers considered in this study, the three different splitting strategies are also applied on this dataset. However, since the data is now encoded as a graph the original algorithm of the splitting the data by strategy cannot be applied, therefore a tier specific algorithm had to be created.

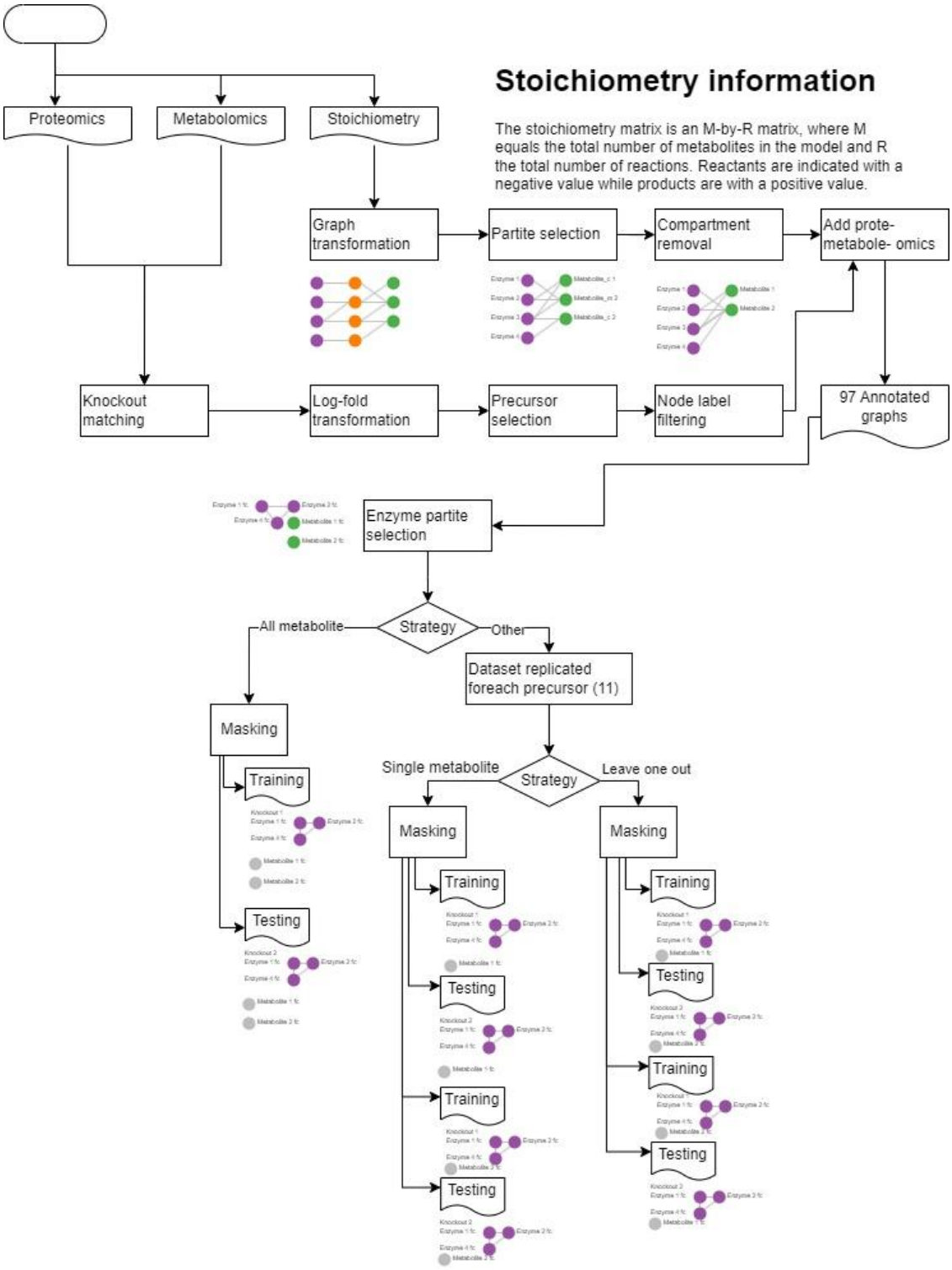


Figure 16: ***Graphical depiction of data pipeline for adding stoichiometry information****. The three datasets are first individually processed then combined to form the 97 annotated graphs. After the task has been selected the node embedding is added to each respective node based on the graph topology for that task. Then for selected training strategy the datasets are generated based on the final graph instances. This prevents having to rerun the entire transformation pipeline.*

For all datasets generated according to the strategy, a hyperparameter search (Table 5) is performed making use of Raytune async hyperband optimization. This optimization approach allows for the early termination of models according to the reported loss function, which leads to an increased number of hyperparameter configurations that can be tested. The most promising model is trained for three hundred epochs if it is not terminated via early stopping. The model results, together with the model configuration are stored and ranked according to the performance metrics. The configuration of the optimal model is then used to generate 16 new models to evaluate the stability of the configuration, the best from retraining is then reported as the optimal model.

*Table 5: **Hyperparameters for the GNN that is trained based on the stoichiometry-derived datasets**. The hyperparameters are used for training the graph attention layers, which are used for graph-level optimization.*

Parameter	Value
Batch size	2, 4, 8
Learning rate	0.1, 0.05, 0.01, 0.001
SGD momentum	0.5, 0.8, 0.9
Scheduler gamma	0.995, 1
Embedding size	8, 16, 32, 64, 128
Attention heads	1, 2, 3, 4
Layers	1, 3, 5, 7

Validating the models and explanations

Correctly validating the performance of the model is important to create a ranking for the learning algorithms. To validate the models two approaches were used a scale-based validation and a scale-free validation. The scale-based validation uses the mean of the target variable (metabolic fold-change) to create a metric, while the scale-free validation is independent of the target variable. The scale-based validation is used during the hyperparameter search for each model as there is no direct comparison between the different metabolites that are in the dataset. To rank each hyperparameter configuration the mean squared error (MSE) is used, as this metric has proven to be a good balance between penalizing bad predictions and overestimating the model's badness (Botchkarev, 2018). When comparing models over different datasets or different architectures the scale-free median-absolute percentage error (mdAPE) is used (Equation 2). This metric is it is insensitive to outliers and relatively easy to interpret, furthermore the mdAPE can be compared between the different metabolites allowing one to make a ranking on how well a model predicted a certain group of metabolites.

$$mdAPE = median\left(\frac{\left|y_{true} - y_{pred}\right|}{y_{true}}\right) * 100\%$$

Equation 2: Using the mdAPE we can compare different populations that have different means as the mdAPE is independent of the mean of the target variable.

To give a fairer indication of the performance of the machine learning algorithm, a total of one hundred random train test splits are made for each tier-scenario combination. With this excessive amount of repeats, we can give a true indication of the average performance of a model architecture. We used a randomized train-test split where within one iteration the seed is fixed, which allows the comparison of the algorithms within one iteration. As the knockouts between the different tier-scenario combinations are fixed.

To generate explanations for the baseline dataset SHAP and LIME are applied to the final model, and for each instance within the testing dataset, an explanation is generated. The stability of the explanations is evaluated by measuring the consistency between the explanation's strategies. For both the SHAP and LIME algorithm the most interesting non-zero interacting proteins are generated. An explanation algorithm is used to evaluate the stability of explanations by generating a ranking for each feature in the dataset and comparing the list between the different samples.

The surrogate models created by the explanation strategies had to be evaluated based on their truthfulness with the original model. A common practice to evaluate the truthfulness of a model is to use the fidelity score of the surrogate model. The fidelity score can be applied to each explanation algorithm, which allows the ranking of the explanation algorithm based on its truthfulness. Next to truthfulness the explanations also need to be accurate, the prediction of all explanations can be calculated by summing the relative contribution of each feature. This prediction is then compared to the ground truth indicating the accuracy of the explanation algorithm. Finally, the explanations were used for generating a visualization that helps a metabolic engineer. This was not a trivial task and it was decided that visualizing the interplay between the metabolites, reactions and enzymes was the most optimal way of visualizing the explanation.

Bibliography

Alghamdi, N., Chang, W., Dang, P., Lu, X., Wan, C., Gampala, S., Huang, Z., Wang, J., Ma, Q., Zang, Y., Fishel, M., Cao, S., & Zhang, C. (n.d.). *A graph neural network model to estimate cell-wise metabolic flux using single cell RNA-seq data*. https://doi.org/10.1101/2020.09.23.310656

Angione, C. (2019). *Human Systems Biology and Metabolic Modelling: A Review-From Disease Metabolism to Precision Medicine*. https://doi.org/10.1155/2019/8304260

Antolin, A. A., & Cascante, M. (2021). AI delivers Michaelis constants as fuel for genome-scale metabolic models. *PLOS Biology*, *19*(10), e3001415. https://doi.org/10.1371/JOURNAL.PBIO.3001415

Antoniewicz, M. R. (2018). A guide to 13C metabolic flux analysis for the cancer biologist. *Experimental & Molecular Medicine 2018 50:4*, *50*(4), 1–13. https://doi.org/10.1038/s12276-018-0060-y

Arsov, N., & Mirceva, G. (2019). *Network Embedding: An Overview*.

Azhar, M. A., & Thomas, P. A. (2019). Comparative Review of Feature Selection and Classification modeling. *2019 6th IEEE International Conference on Advances in Computing, Communication and Control, ICAC3 2019*. https://doi.org/10.1109/ICAC347590.2019.9036816

Belle, V., & Papantonis, I. (2021). Principles and Practice of Explainable Machine Learning. *Frontiers in Big Data*, *4*, 39. https://doi.org/10.3389/FDATA.2021.688969/BIBTEX

Botchkarev, A. (2018). Performance Metrics (Error Measures) in Machine Learning Regression, Forecasting and Prognostics: Properties and Typology. *Interdisciplinary Journal of Information, Knowledge, and Management*, *14*, 45–76. https://doi.org/10.28945/4184

Bru, S., Martínez-Laínez, J. M., Hernández-Ortega, S., Quandt, E., Torres-Torronteras, J., Martí, R., Canadell, D., Ariño, J., Sharma, S., Jiménez, J., & Clotet, J. (2016). Polyphosphate is involved in cell cycle progression and genomic stability in Saccharomyces cerevisiae. *Molecular Microbiology*, *101*(3), 367–380. https://doi.org/10.1111/MMI.13396

Burga, A., & Lehner, B. (2012). Beyond genotype to phenotype: why the phenotype of an individual cannot always be predicted from their genome sequence and the environment that they experience. *The FEBS Journal*, *279*(20), 3765–3775. https://doi.org/10.1111/J.1742-4658.2012.08810.X

Chae, T. U., Choi, S. Y., Kim, J. W., Ko, Y. S., & Lee, S. Y. (2017). Recent advances in systems metabolic engineering tools and strategies. *Current Opinion in Biotechnology*, *47*, 67–82. https://doi.org/10.1016/J.COPBIO.2017.06.007

Chakdar, H., Hasan, M., Pabbi, S., Nevalainen, H., & Shukla, P. (2021). High-throughput proteomics and metabolomic studies guide re-engineering of metabolic pathways in eukaryotic microalgae: A review. *Bioresource Technology*, *321*. https://doi.org/10.1016/J.BIORTECH.2020.124495

Choi, K. R., Jang, W. D., Yang, D., Cho, J. S., Park, D., & Lee, S. Y. (2019). Systems Metabolic Engineering Strategies: Integrating Systems and Synthetic Biology with Metabolic Engineering. *Trends in Biotechnology*, *37*(8), 817–837. https://doi.org/10.1016/J.TIBTECH.2019.01.003

Dalmia, A., & Gupta, M. (n.d.). *Towards Interpretation of Node Embeddings*. https://doi.org/10.1145/3184558.3191523

Edwards, J. S., Covert, M., & Palsson, B. (1995). Metabolic modelling of microbes: the flux-balance approach. In *Environmental Microbiology* (Vol. 4, Issue 3). Varner and Ramkrishna.

Emwas, A. H., Szczepski, K., Al-Younis, I., Lachowicz, J. I., & Jaremko, M. (2022). Fluxomics - New Metabolomics Approaches to Monitor Metabolic Pathways. *Frontiers in Pharmacology*, *13*, 299. https://doi.org/10.3389/FPHAR.2022.805782/BIBTEX

Feldmann, H. (2012). Yeast: Molecular and Cell Biology: Second Edition. *Yeast: Molecular and Cell Biology: Second Edition*. https://doi.org/10.1002/9783527659180

Groher, A. C., Jager, S., Schneider, C., Groher, F., Hamacher, K., & Suess, B. (2019a). Tuning the Performance of Synthetic Riboswitches using Machine Learning. *ACS Synthetic Biology*, *8*(1), 34–44. https://doi.org/10.1021/ACSSYNBIO.8B00207

Groher, A. C., Jager, S., Schneider, C., Groher, F., Hamacher, K., & Suess, B. (2019b). Tuning the Performance of Synthetic Riboswitches using Machine Learning. *ACS Synthetic Biology*, *8*(1), 34–44. https://doi.org/10.1021/ACSSYNBIO.8B00207/ASSET/IMAGES/LARGE/SB-2018-00207F_0011.JPEG

Hadadi, N., MohammadiPeyhani, H., Miskovic, L., Seijo, M., & Hatzimanikatis, V. (2019). Enzyme annotation for orphan and novel reactions using knowledge of substrate reactive sites. *Proceedings of the National Academy of Sciences of the United States of America*, *116*(15), 7298–7307. https://doi.org/10.1073/PNAS.1818877116/SUPPL_FILE/PNAS.1818877116.SD01.XLSX

Heckmann, D., Lloyd, C. J., Mih, N., Ha, Y., Zielinski, D. C., Haiman, Z. B., Desouki, A. A., Lercher, M. J., & Palsson, B. O. (n.d.). *Machine learning applied to enzyme turnover numbers reveals protein structural correlates and improves metabolic models*. https://doi.org/10.1038/s41467-018-07652-6

Heinemeyer, W., Kleinschmidt, J. A., Saidowsky, J., Escher, C., & Wolf, D. H. (1991). Proteinase yscE, the yeast proteasome/multicatalytic-multifunctional proteinase: mutants unravel its function in stress induced proteolysis and uncover its necessity for cell survival. *The EMBO Journal*, *10*(3), 555–562. https://doi.org/10.1002/J.1460-2075.1991.TB07982.X

Herzer, S., Bhangale, A., Barker, G., Chowdhary, I., Conover, M., O'Mara, B. W., Tsang, L., Wang, S. Y., Krystek, S. R., Yao, Y., & Rieble, S. (2015). Development and scale-up of the recovery and purification of a domain antibody Fc fusion protein-comparison of a two and three-step approach. *Biotechnology and Bioengineering*, *112*(7), 1417–1428. https://doi.org/10.1002/BIT.25561

Hideyuki, M., Takayoshi, K., Hozumi, T., Dai, H., Tokichi, M., & Chikako, T. (1993). Isolation and characterization of SSE1 and SSE2, new members of the yeast HSP70 multigene family. *Gene*, *132*(1), 57–66. https://doi.org/10.1016/0378-1119(93)90514-4

Interpretable Machine Learning. (n.d.). Retrieved January 30, 2023, from https://christophm.github.io/interpretable-ml-book/

Jervis, A. J., Carbonell, P., Taylor, S., Sung, R., Dunstan, M. S., Robinson, C. J., Breitling, R., Takano, E., & Scrutton, N. S. (2019). *SelProm: A Queryable and Predictive Expression Vector Selection Tool for Escherichia coli*. https://doi.org/10.1021/acssynbio.8b00399

Judge, A., & Dodd, M. S. (2020). Metabolism. *Essays in Biochemistry*, *64*(4), 607–647. https://doi.org/10.1042/EBC20190041

Kavita, K., & Breaker, R. R. (2023). Discovering riboswitches: the past and the future. *Trends in Biochemical Sciences*, *48*(2), 119–141. https://doi.org/10.1016/J.TIBS.2022.08.009

Kim, G. B., Kim, W. J., Kim, H. U., & Lee, S. Y. (2020). Machine learning applications in systems metabolic engineering. *Current Opinion in Biotechnology*, *64*, 1–9. https://doi.org/10.1016/J.COPBIO.2019.08.010

Kim, H. U., Charusanti, P., Lee, S. Y., & Weber, T. (2016). Metabolic engineering with systems biology tools to optimize production of prokaryotic secondary metabolites. *Natural Product Reports*, *33*(8), 933–941. https://doi.org/10.1039/C6NP00019C

King, Z. A., Lu, J., Dräger, A., Miller, P., Federowicz, S., Lerman, J. A., Ebrahim, A., Palsson, B. O., & Lewis, N. E. (2016). BiGG Models: A platform for integrating, standardizing and sharing genome-scale models. *Nucleic Acids Research*, *44*(D1), D515–D522. https://doi.org/10.1093/NAR/GKV1049

Krassowski, M., Das, V., Sahu, S. K., & Misra, B. B. (2020). State of the Field in Multi-Omics Research: From Computational Needs to Data Mining and Sharing. *Frontiers in Genetics*, *11*, 1598. https://doi.org/10.3389/FGENE.2020.610798/BIBTEX

Lee, J. W., Na, D., Park, J. M., Lee, J., Choi, S., & Lee, S. Y. (2012). Systems metabolic engineering of microorganisms for natural and non-natural chemicals. *Nature Chemical Biology 2012 8:6*, *8*(6), 536–546. https://doi.org/10.1038/nchembio.970

Libourel, I. G. L., & Shachar-Hill, Y. (2008). Metabolic flux analysis in plants: From intelligent design to rational engineering. *Annual Review of Plant Biology*, *59*, 625–650. https://doi.org/10.1146/ANNUREV.ARPLANT.58.032806.103822

Liu, L., Wang, J., Rosenberg, D., Zhao, H., Lengyel, G., & Nadel, D. (2018). Fermented beverage and food storage in 13,000 y-old stone mortars at Raqefet Cave, Israel: Investigating Natufian ritual feasting. *Journal of Archaeological Science: Reports*, *21*, 783–793. https://doi.org/10.1016/j.jasrep.2018.08.008

Lundberg, S. M., Allen, P. G., & Lee, S.-I. (n.d.). *A Unified Approach to Interpreting Model Predictions*. https://github.com/slundberg/shap

Masampally, V. S., Pareek, A., & Runkana, V. (2019). Cascade Gaussian Process Regression Framework for Biomass Prediction in a Fed-batch Reactor. *Proceedings of the 2018 IEEE Symposium Series on Computational Intelligence, SSCI 2018*, 128–135. https://doi.org/10.1109/SSCI.2018.8628937

Mendoza, S. N., Olivier, B. G., Molenaar, D., & Teusink, B. (2019). A Systematic Assessment Of Current Genome-Scale Metabolic Reconstruction Tools. *BioRxiv*, 558411. https://doi.org/10.1101/558411

Monk, J. M., Lloyd, C. J., Brunk, E., Mih, N., Sastry, A., King, Z., Takeuchi, R., Nomura, W., Zhang, Z., Mori, H., Feist, A. M., & Palsson, B. O. (2017). iML1515, a knowledgebase that computes Escherichia coli traits. *Nature Biotechnology 2017 35:10*, *35*(10), 904–908. https://doi.org/10.1038/nbt.3956

Nakajima, N., Hayashida, M., Jansson, J., Maruyama, O., & Akutsu, T. (2018). *Determining the Minimum Number of Protein-Protein Interactions Required to Support Known Protein Complexes*.

Natarajan, K., Meyer, M. R., Jackson, B. M., Slade, D., Roberts, C., Hinnebusch, A. G., & Marton, M. J. (2001). Transcriptional Profiling Shows that Gcn4p Is a Master Regulator of Gene Expression during Amino Acid Starvation in Yeast. *Molecular and Cellular Biology*, *21*(13), 4347–4368. https://doi.org/10.1128/MCB.21.13.4347-4368.2001/ASSET/CC975E8D-9FE1-4CBF-B454-F61BDA501562/ASSETS/GRAPHIC/MB1310194010.JPEG

Neijssel, O. M., & Tempest, D. W. (1986). Fermentation Products: Physiological and Bioenergetic Considerations. *Biotechnology: Potentials and Limitations*, 83–97. https://doi.org/10.1007/978-3-642-70535-9_7

Nielsen, J. (2003). It Is All about Metabolic Fluxes. *Journal of Bacteriology*, *185*(24), 7031–7035. https://doi.org/10.1128/JB.185.24.7031-7035.2003

Nielsen, J. (2014). Synthetic Biology for Engineering Acetyl Coenzyme A Metabolism in Yeast. *MBio*, *5*(6). https://doi.org/10.1128/MBIO.02153-14

Nielsen, J., & Keasling, J. D. (2016). Engineering Cellular Metabolism. *Cell*, *164*(6), 1185–1197. https://doi.org/10.1016/J.CELL.2016.02.004

Oyetunde, T., Liu, D., Martin, H. G., & Tang, Y. J. (2019). Machine learning framework for assessment of microbial factory performance. *PLoS ONE*, *14*(1). https://doi.org/10.1371/JOURNAL.PONE.0210558

Pandey, A. v., Henderson, C. J., Ishii, Y., Kranendonk, M., Backes, W. L., & Zanger, U. M. (2017). Editorial: Role of protein-protein interactions in metabolism: Genetics, structure, function. *Frontiers in Pharmacology*, *8*(NOV). https://doi.org/10.3389/FPHAR.2017.00881

Patmanathan, S. N., Gnanasegaran, N., Lim, M. N., Husaini, R., Fakiruddin, K. S., & Zakaria, Z. (2018). CRISPR/Cas9 in Stem Cell Research: Current Application and Future Perspective. *Current Stem Cell Research & Therapy*, *13*(8), 632–644. https://doi.org/10.2174/1574888X13666180613081443

Reimers, A. M., & Reimers, A. C. (2016). The steady-state assumption in oscillating and growing systems. *Journal of Theoretical Biology*, *406*, 176–186. https://doi.org/10.1016/J.JTBI.2016.06.031

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *NAACL-HLT 2016 - 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Demonstrations Session*, 97–101. https://doi.org/10.48550/arxiv.1602.04938

Robinson, P. K. (2015). Enzymes: principles and biotechnological applications. *Essays in Biochemistry*, *59*, 1. https://doi.org/10.1042/BSE0590001

Samuel, D. (1996). Investigation of Ancient Egyptian Baking and Brewing Methods by Correlative Microscopy. *Science (New York, N.Y.)*, *273*(5274), 488–490. https://doi.org/10.1126/SCIENCE.273.5274.488

Segler, M. H. S., Preuss, M., & Waller, M. P. (2018). Planning chemical syntheses with deep neural networks and symbolic AI. *Nature 2018 555:7698*, *555*(7698), 604–610. https://doi.org/10.1038/nature25978

Shatnawi, M. (2015). Review of Recent Protein-Protein Interaction Techniques. *Emerging Trends in Computational Biology, Bioinformatics, and Systems Biology: Algorithms and Software Tools*, 99–121. https://doi.org/10.1016/B978-0-12-802508-6.00006-5

Shi, S., Si, T., Liu, Z., Zhang, H., Lui Ang, E., & Zhao, H. (2016). Metabolic engineering of a synergistic pathway for n-butanol production in Saccharomyces cerevisiae. *Nature Publishing Group*. https://doi.org/10.1038/srep25675

STRING: functional protein association networks. (n.d.). Retrieved May 9, 2022, from https://string-db.org/

Swinnen, S., Henriques, S. F., Shrestha, R., Ho, P. W., Sá-Correia, I., & Nevoigt, E. (2017). Improvement of yeast tolerance to acetic acid through Haa1 transcription factor engineering: Towards the underlying mechanisms. *Microbial Cell Factories*, *16*(1), 1–15. https://doi.org/10.1186/S12934-016-0621-5/FIGURES/5

Szklarczyk, D., Franceschini, A., Wyder, S., Forslund, K., Heller, D., Huerta-Cepas, J., Simonovic, M., Roth, A., Santos, A., Tsafou, K. P., Kuhn, M., Bork, P., Jensen, L. J., & von Mering, C. (2015). STRING v10: protein-protein interaction networks, integrated over the tree of life. *Nucleic Acids Research*, *43*(Database issue), D447–D452. https://doi.org/10.1093/NAR/GKU1003

Tanaka, K. (2009). The proteasome: Overview of structure and functions. *Proceedings of the Japan Academy. Series B, Physical and Biological Sciences*, *85*(1), 12. https://doi.org/10.2183/PJAB.85.12

Thiele, I., & Palsson, B. (2010). A protocol for generating a high-quality genome-scale metabolic reconstruction. *Nature Protocols 2010 5:1*, *5*(1), 93–121. https://doi.org/10.1038/nprot.2009.203

Tomashevsky, A., Kulakovskaya, E., Trilisenko, L., Kulakovskaya, T., Fedorov, A., & Eldarov, M. (2020). *Role of VTC4 in Stress Response and Regulation of Inorganic Polyphosphate Levels in Yeast*. https://doi.org/10.20944/PREPRINTS202010.0484.V1

Töpfer, N., Kleessen, S., & Nikoloski, Z. (2015). Integration of metabolomics data into metabolic networks. *Frontiers in Plant Science*, *6*(FEB), 49. https://doi.org/10.3389/FPLS.2015.00049/BIBTEX

Veličkovi´veličkovi´c, P., Cucurull, G., Casanova, A., Romero, A., Lì, P., & Bengio, Y. (n.d.). *GRAPH ATTENTION NETWORKS*.

Wang, D., Akhberdi, O., Hao, X., Yu, X., Chen, L., Liu, Y., & Zhu, X. (2017). Amino Acid Sensor Kinase Gcn2 Is Required for Conidiation, Secondary Metabolism, and Cell Wall Integrity in the Taxol-Producer Pestalotiopsis microspora. *Frontiers in Microbiology*, *8*(SEP), 1879. https://doi.org/10.3389/FMICB.2017.01879

What is metabolomics? | Metabolomics. (n.d.). Retrieved February 5, 2023, from https://www.ebi.ac.uk/training/online/courses/metabolomics-introduction/what-is/

What is proteomics? | Proteomics. (n.d.). Retrieved February 5, 2023, from https://www.ebi.ac.uk/training/online/courses/proteomics-an-introduction/what-is-proteomics/

Yamamoto, Y., Yamada, R., Matsumoto, T., & Ogino, H. (2023). Construction of a machine-learning model to predict the optimal gene expression level for efficient production of d-lactic acid in yeast. *World Journal of Microbiology and Biotechnology*, *39*(3), 1–10. https://doi.org/10.1007/S11274-022-03515-X/FIGURES/7

Yao, Q., Xu, Y., Yang, H., Shang, D., Zhang, C., Zhang, Y., Sun, Z., Shi, X., Feng, L., Han, J., Su, F., Li, C., & Li, X. (2015). Global Prioritization of Disease Candidate Metabolites Based on a Multi-omics Composite Network OPEN. *Scientific RepoRts |*, *5*, 17201. https://doi.org/10.1038/srep17201

Zelezniak, A., Vowinckel, J., Capuano, F., Messner, C. B., Demichev, V., Polowsky, N., Mülleder, M., Kamrad, S., Klaus, B., Keller, M. A., & Ralser, M. (2018). Machine Learning Predicts the Yeast Metabolome from the Quantitative Proteome of Kinase Knockouts. *Cell Systems*, *7*(3), 269-283.e6. https://doi.org/10.1016/J.CELS.2018.08.001

Zhang, J., Petersen, S. D., Radivojevic, T., Ramirez, A., Pérez-Manríquez, A., Abeliuk, E., Sánchez, B. J., Costello, Z., Chen, Y., Fero, M. J., Martin, H. G., Nielsen, J., Keasling, J. D., & Jensen, M. K. (2020). Combining mechanistic and machine learning models for predictive engineering and optimization of tryptophan metabolism. *Nature Communications 2020 11:1*, *11*(1), 1–13. https://doi.org/10.1038/s41467-020-17910-1

Zheng, Z. Y., Guo, X. N., Zhu, K. X., Peng, W., & Zhou, H. M. (2017). Artificial neural network – Genetic algorithm to optimize wheat germ fermentation condition: Application to the production of two anti-tumor benzoquinones. *Food Chemistry*, *227*, 264–270. https://doi.org/10.1016/J.FOODCHEM.2017.01.077

Zhou, Y., Li, G., Dong, J., Xing, X. hui, Dai, J., & Zhang, C. (2018). MiYA, an efficient machine-learning workflow in conjunction with the YeastFab assembly strategy for combinatorial optimization of heterologous metabolic pathways in Saccharomyces cerevisiae. *Metabolic Engineering*, *47*, 294–302. https://doi.org/10.1016/J.YMBEN.2018.03.020

Zomorrodi, A. R., & Maranas, C. D. (2010). Improving the iMM904 S. cerevisiae metabolic model using essentiality and synthetic lethality data. *BMC Systems Biology*, *4*(1), 1–15. https://doi.org/10.1186/1752-0509-4-178/FIGURES/5

Zou, H., & Hastie, T. (2005). Regularization and variable selection via the elastic net. *J. R. Statist. Soc. B*, *67*(2), 301–320.

Appendix A

Figure 17: Toy example of how the strategy can be used to transform the baseline dataset. For this example, three enzymes and three metabolites are used to show all possible combinations. In total seven different splits are constructed.

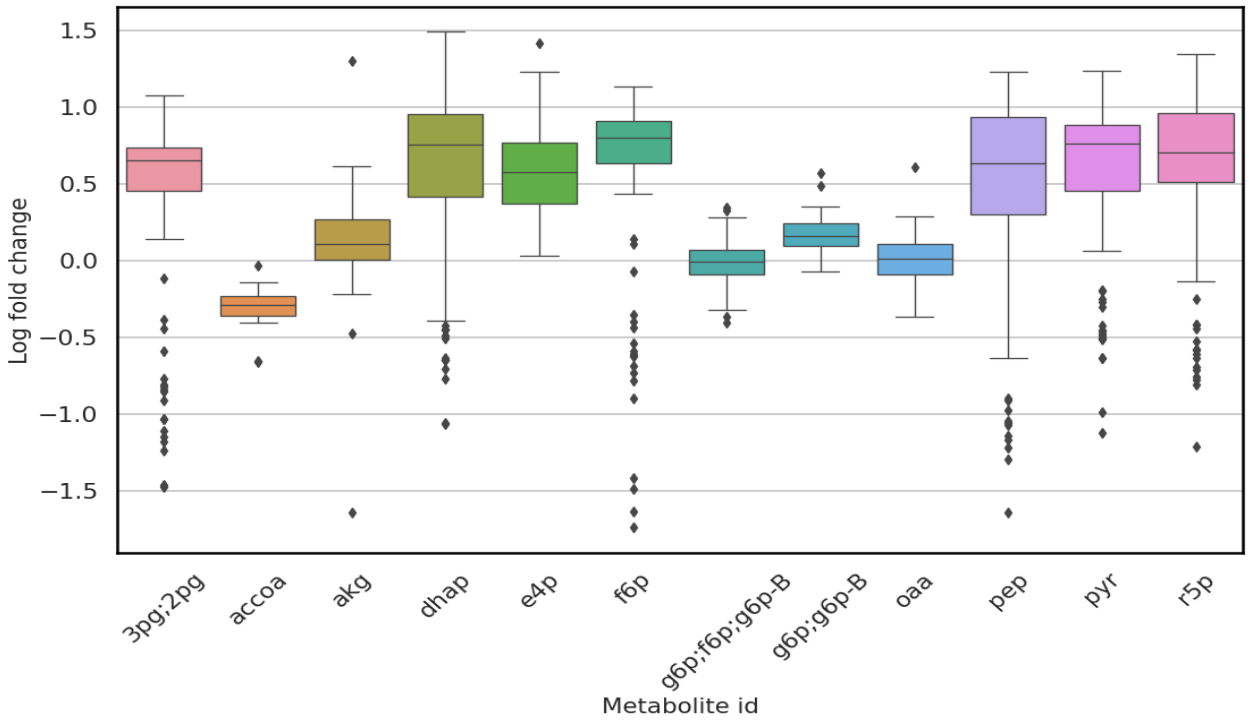


Figure 18: Boxplot showing the metabolite values after using the log-fold transformation. The fold change is centered between the -1.5 and 1.5 for each of the 12 precursor metabolites. Which makes model training relatively stable when using the All-metabolite strategy.

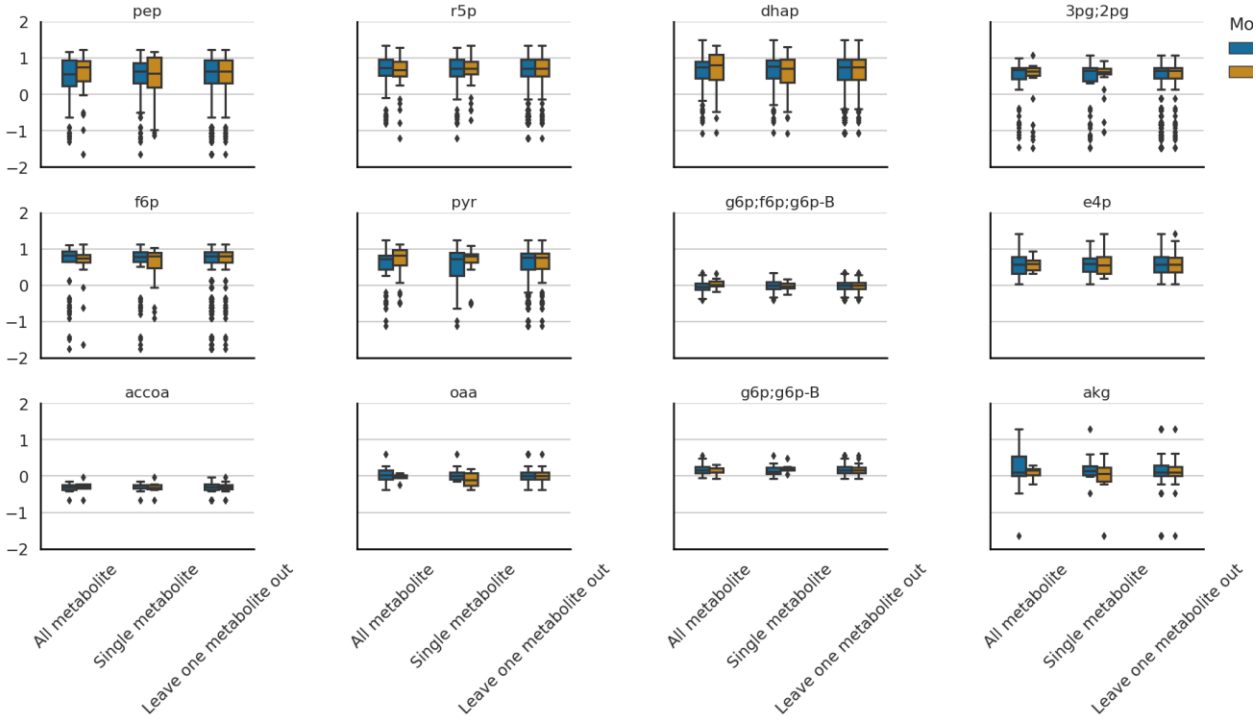


Figure 19: Distribution of metabolic-fold change between the training and testing dataset given the different splitting strategies. We observe that the testing dataset is within the range of the training dataset which allows us to the metrics used for evaluating the learning algorithms correctly.

To further explore why there is a large difference between F6P and PEP, we ran additional analysis per knockout to find, which configurations that model find hard to generalize on. We observe that the fold-change of F6P metabolite has a smaller domain than the fold-change of PEP in the testing dataset given the single metabolite strategy.

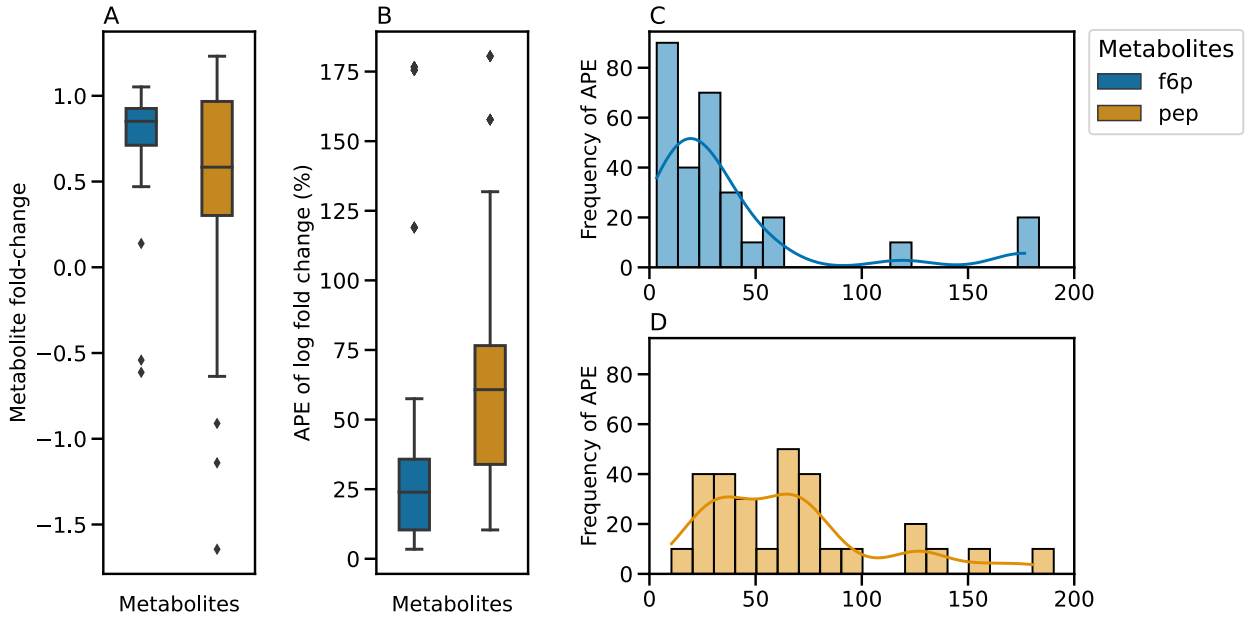


Figure 20: In-depth comparison between F6P and PEP. The metabolite fold-change, prediction targets for the models, are shown in Panel A. F6P has a smaller fold-change range within the test dataset than PEP which could indicate that the train-test split was not correctly balanced. The absolute percentage error is used for the boxplot in panel B. We reinforce the idea that the error of f6p is lower than the error of PEP. The frequency of APE is shown in panel C and D, we see that the error decreases gradually for the f6p metabolite while for the PEP metabolite the frequency of errors stay relatively the same.