Metabolic Pathway Optimization with Data Driven Approaches

Yingxin Cao

A thesis
submitted in partial fulfillment of the
requirements for the degree of

Master of Science in Bioengineering

University of Washington

2018

Reading Committee:

Herbert M. Sauro, Chair

Hao Yuan Kueh

Program Authorized to Offer Degree:
Department of Bioengineering

University of Washington

Abstract

Metabolic Pathway Optimization with Data Driven Approaches

Yingxin Cao

Chair of the Supervisory Committee:
Associate Professor Herbert M. Sauro
Department of Bioengineering

The flux control coefficient (FCC) is a sensitivity coefficient that measures the percent change in flux as a result of a given percentage change in the activity of the enzyme. The higher the FFC the more controlling the step. However, the value for a flux control coefficient for a specific enzyme is determined by all the other enzymes in the pathway. Most kinetic rate laws for pathway models are generally nonlinear, which makes an analytical analysis virtually impossible. Previous studies have explored linearization under assumptions such as non-saturation, which limit all steps to the first order region, however these assumptions are not realistic for all situations.

Here we present a statistical approach to predict the probability distribution of dominance of each enzyme step in a linear section of metabolic flux, and in the meantime, to identify key system parameters that could maximally reduce the uncertainty of the distribution. Generalized linear models are applied to predict the probability of dominance of each step, while L1 regularization is applied to select system parameters that contribute most to the prediction. For better predictions, a neural network is used for the prediction of distribution of control coefficients based on the FCC summation property. The models are trained on synthetic datasets generated using fully reversible Michaelis-Menten kinetics. All parameters are randomly sampled from a maximum entropy distribution assuming no prior knowledge on the system. For a pathway up to 15 nodes, the results show over 90% accuracy in predicting

step with the largest control coefficient at the extreme regularization condition, where total enzyme, the equilibrium constant, and forward Michaelis-Menten constant are identified as key system parameters. Similar patterns can be generated for pathways with different number of nodes. The approach is also tested under noisy data, and shows higher accuracy when noise increase compared with numerical simulation. These results offer a means to determine the FCCs of a pathway given minimal information under noise. It will make it easier for metabolic engineers to target the most promising enzymatic steps to maximize pathway flux.

TABLE OF CONTENTS

Page

List of Figures iii

List of Tables iv

Chapter 1: Introduction 1
1.1 Metabolic Engineering 1
1.2 Uncertainty in Biological Systems 2
1.3 Supervised Machine Learning and Feature Selection 3
1.4 Synthetic Dataset 3
1.5 A Statistical Framework 4

Chapter 2: Background and Significance 6
2.1 Metabolic Control Analysis 6
2.2 Kinetics for Simulation of Cell Metabolic Networks 7
2.3 Statistical Learning Methods 8
2.3.1 Models and Frameworks 8
2.3.2 Feature Selection and Ranking 12
2.4 Significance of the Work 13

Chapter 3: Methods 15
3.1 Simulation Methods for Synthetic Dataset 15
3.2 Generalized Linear Model with Group Lasso 21
3.2.1 Multinomial Logistic Regression 21
3.2.2 Group Lasso 21
3.3 Neural Networks 22
3.4 Comparison Study Under Noise 24

Chapter 4: Results 26
4.1 Results of Linear Methods 26
4.2 Results of Neural Networks 30
4.3 Results under Uncertainty 33

Chapter 5: Discussions 35
5.1 Implications 35
5.1.1 Trade-off between Accuracy and Cost 35
5.1.2 Robustness Against Noise 36
5.1.3 Neural Networks as An Effective Approach to Enhance Performance . 36
5.2 Limitations and Future work 37

Chapter 6: Conclusions 40

Bibliography 41

LIST OF FIGURES

Figure Number		Page
2.1	Lasso v. Ridge on a two parameter case [8].	9
2.2	Architecture of an artificial neural network with two hidden layers.	11
3.1	Linear metabolic pathway with two boundary spices and $n+1$ nodes	15
3.2	Example of a 4-node model using Antimony Definition	17
3.3	Diagrammatic description of the algorithm used to generate data sample.	18
3.4	Snapshot of the control coefficient generated by simulations	19
3.5	Distributions of max control coefficients	20
3.6	Architecture of the multilayer neural network model	23
3.7	Diagram of steps for study under uncertainty	25
4.1	Heatmap visualization of GLM coefficients	27
4.2	Accuracy and number of nonzero parameters under increasing penalty	28
4.3	Values of coefficients under increasing penalty	29
4.4	Loss and accuracy during training process of NN	31
4.5	Confusion Matrix of NN model for 7-node pathway	32
4.6	Accuracy v. Noise for Numerical Simulation and NN	34
5.1	Architecture of a deeper NN	39

LIST OF TABLES

Table Number Page

3.1 Distributions of randomly sampled kinetic parameters 16

4.1 Prediction of test set using NN 31

ACKNOWLEDGMENTS

I would like to thank Dr. Herbert Sauro first. As my PI, he's been offering valuable insights and advice during my entire Master study, from which I gain valuable skills and research thoughts. I also would like to thank the other committee member Dr. Hao Yuan Kueh, whos been my instructor for my advanced systems biology class which eventually lead me into this research field.

I thank friends and folks from lab, who offered their selfless help during my study. You have been really patient as I firstly entered this field, and offered me support academically and emotionally.

At last, I thank my parents who have been supportive through my entire life. It was your supports that keep inspiring me to achieve more.

DEDICATION

to my dear parents, Yang and Yueju

Chapter 1

INTRODUCTION

1.1 Metabolic Engineering

Previous studies have shown that natural products have played an invaluable role in the drug discovery process, especially in the areas of cancer and infectious diseases[25]. Natural products synthesized by biological organisms through in vivo biochemical reactions are a rich source of valuable industrial chemicals[33]. However, these chemicals exists only at a low level of concentration, thus often make it hard for industrial use. *Metabolic engineering* aims at improving specific cellular properties by intentional optimizations of biochemical reaction steps. Applications of metabolic engineering includes increasing yield of chemical products produced by microorganisms, introducing new chemical products to the host cell, and etc[32]. Many aspects of metabolic engineering have yielded numerous scientific discoveries, as well as viable approaches for optimizing natural product biosynthesis. For example, characterization of components in individual pathways has facilitated the construction of hybrid pathways for efficient production; control in expression and regulation has enabled better fine-tuning intervention of pathways for more target product with less costs; and systems biology and metabolic modeling tools have resulted in increasing predictive and analytic capabilities[26].

Strategies of metabolic engineering generally involve increasing or decreasing metabolic flux through certain branch, which is often a complicated problem due to the robustness of natural biological systems. Substantial efforts have been made in order to explore the cellular metabolism and its regulation rules resulting in various experimental and theoretical approaches developed to solve the problem. A thorough understanding of molecular basis of metabolism provides researchers the capability to model the cellular systems with mathematical equations, which allows researchers to focus more on theoretical and analytical

studies to understand control behaviors over the cell metabolic network and achieve certain engineering purposes with intentional design instead of intuitions.

Metabolic Control Analysis (MCA) is one of the first theoretical frameworks built to analyze and interpret control and regulation behavior of cellular metabolic networks using kinetic characteristics of component enzymes at a system level [15, 1, 6]. As a conceptual approach, MCA offers quantitative prospectives on understanding the control of the system of interest by introducing and quantifying sensitivities of parameter to the responses of metabolic flux changes. MCA has been well-developed and received tremendous amount of attention since established, and has become a powerful tool in metabolic engineering. By looking at the sensitivity coefficients for each step, it's a simple and promising guideline for metabolic engineers to choose target steps to optimize.

1.2 Uncertainty in Biological Systems

As addressed by previous researches [16, 1], it requires kinetic information of all component enzymes to determine the sensitivity of metabolic flux in response to change of enzyme kinetics parameters for each steps, which means either we are calculating the sensitivity coefficients for each steps analytically or numerically, we are going to need kinetic informations from all the enzymes in the system. Although, nowadays biochemical technologies allow measurements of kinetic parameters and there are existing databases for enzymatic data [29], we are still faced with the problem of missing data, since the database can hardly cover all component enzymes in the network, and antecedent experimental conditions may not precisely reflect the real conditions that the enzymes are actually working at. Thus, to obtain exact quantitative information on control and regulation details of the metabolic network, it requires measurements of kinetic parameters in vivo which are often expensive in cost and come with noise.

1.3 Supervised Machine Learning and Feature Selection

Supervised machine learning is usually considered as a statistical predictive approach mapping input information X to output predictions y with a probabilistic based model. Typical tasks for supervised learning include classification and regression [24]. The training process generally involves optimization of an objective function that describes deviation of predicted labels from correct labels. The statistical approaches have several advantages, by tuning the objective function people can achieve different learning purposes including robustness against noise or sparsity in predictors [19]. Also, these learning approaches are often scalable to large datasets which are nowadays common cases for scientific research with advancements in measurement technologies and high performance scientific computing. After training processes, supervised learning models can produce probabilistic predictions for novel inputs, which can then provides valuable references for users.

Simultaneous multi-class feature selection focus on selecting a small subset of features from input variables that work simultaneously for all classes [10]. It is initially proposed to solve learning problem of high-dimensional dataset with limited training cases such as text classification problems when number of vocabularies is large and training samples are limited; it is also applied to biomedical studies where each feature corresponds to certain measurements that come with a cost thus expensive to acquire [3]. Multiple feature selection strategies and techniques are developed leveraging performance and computational cost[10, 3]. These methods provides feasible implementations to rank the input features based on how informative they are regarding to correct prediction, while in the experimental context could provide a guideline for optimal rank of experimental measurements that shall be taken in order to acquire the desired trade-off between accuracy and cost.

1.4 Synthetic Dataset

Concrete model represents a working hypothesis for biological systems[28]. Synthetic datasets are generated based on simulations of certain models. Accordingly, the datasets consist

with the hypothesis, thus, allow researchers to extract patterns of certain systems from the datasets if the models are built on correct set of assumptions.

With the advancements in high-level languages for simulation and high-performance scientific computing, synthetic datasets are easy to acquire and reproduce, making it an efficient and effective method for data-driven discovery of the system as well as hypothesis generating and testing[4, 30, 2].

Experimental validations are still needed, since assumptions are not always correct due to the complex nature of the biological systems. Unknown interactions or undiscovered properties of the molecules may severely disrupt the predictions of the models built on synthetic datasets. However, these negative results could potentially, on the other hand, lead to new biological discoveries of details of the system.

1.5 A Statistical Framework

Many disciplines has been benefiting from advancements in machine learning. Although, these statistical methods only offer probabilistic outcomes instead of exact results, however, under the situation that biological measurements are already noisy, expensive and time-consuming, robustness against noises and availability to make useful prediction with incomplete input information become unique metrics of this line of methods.

In this study, we purpose a framework predicting control of steps within a metabolic pathway based on synthetic datasets and statistical learning methods. Most modern machine learning methods are built on availability of large scale dataset, since novel models like neural networks contain a large number of parameters to fit. In our framework, synthetic datasets serve as a population of system built on given assumptions defined by the simulation conditions. And the statistical methods extract patterns from this population of system and make useful predictions based on patterns learned by the models. High performance simulation softwares and high level modeling languages allow us to easily and effectively generate large datasets that satisfy the requirements of machine learning methods also well-represent the properties of the population of the system. Machine learning models are then

trained on these synthetic datasets to find patterns of kinetic parameters of these specific systems that can be used to make useful predictions for novel samples as well as discover new properties of systems that are hard to be studied by traditional analytical approaches due to the complexity.

Chapter 2
BACKGROUND AND SIGNIFICANCE

In this chapter, concepts and theoretical backgrounds of methods used in this study will be reviewed. The chapter starts with concepts from *Metabolic Control Analysis* (MCA) framework and enzyme kinetics generally used to describe the cellular metabolism. Then, it moves on to background of the statistical learning approaches brought up in this project. At last, it ends with brief discussions of existing methods and novel points of the new approach.

2.1 Metabolic Control Analysis

Metabolic Control Analysis (MCA) is a theoretical framework bringing sensitivity analysis into biochemical models in order to quantitatively assess regulation power of each step. It was first developed by Henrik Kacser & Jim Burns [15] and Reinhart Heinrich & Tom Rapoport[12] independently in the 1970s and then be well-developed by many researchers in the following decades [7, 11, 17]. *Flux* of a metabolite through a pathway (J) can be represented by the net difference of forward reaction rate and reverse reaction rate (2.1).

$$J = v_f - v_r \tag{2.1}$$

Researches in biochemistry have provided thorough explanations for molecular basis of processes of metabolic networks, which allows researchers to describe reaction rates v in a mathematical form using mechanistic model determined by rate laws and kinetic parameters.

Flux control coefficient (FCC) is a sensitivity coefficient that measures the relative percent change in flux as a result of a given percentage change in the activity of the enzyme. Its mathematical representation is shown in equation 2.2, where J denotes the flux, and p denotes the parameter.

$$C_p^J = \frac{dJ}{dp}\frac{p}{J} = \frac{d\ln J}{d\ln p} \tag{2.2}$$

FCCs generally range from 0 to 1, and the *summation theorem* states that all control coefficients of all enzymes affecting a particular metabolic flux sum up to one as mathematically expressed in equation 2.3 [15].

$$\sum_{i=1}^{n} C_i^J = 1 \tag{2.3}$$

As in cellular metabolism each reaction is catalyzed by enzymes, a specific rate law can be written down and sensitivity coefficient of each parameters can be then calculated based on equation 2.2. Thus, FCC offers a rigorous quantitative measurement on the degree of control that each reaction step, or enzyme, has over the whole system of interest. Modern biotechnologies can achieve changes of parameters in the rate law, such as total enzyme, through certain genetic modifications, which makes FCCs good criterion when choosing the target parameters for engineering. Detailed introduction of kinetics used in this study is provided in next section.

2.2 Kinetics for Simulation of Cell Metabolic Networks

All models can be build based on elementary reactions with mass action kinetics, however, firstly, it would unnecessarily increase the complexity of the model, making analysis hard to perform; secondly, many rate constants for elementary reactions are unknown or unmeasurable [27]. Thus, making reasonable assumptions to approximate the reactions will simplify the model without losing too much precision.

Michaelis-Menten equation (2.5) is first derived by Michaelis and Menten in 1913[23] to describe the enzyme catalyzed reactions (2.4).

$$E + S \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} ES \xrightarrow{k_{cat}} E + P \tag{2.4}$$

$$v = \frac{V_m S}{K_d + S} \tag{2.5}$$

It is based on rapid equilibrium assumption that the binding process and dissociation process are at equilibrium. The assumptions hold for most cases, however, the irreversible rate law is not quite realistic in the context of cellular metabolism where products of previous reaction will be the reactant for the next step [27]. Some extend of the reversibility will have to exist to maintain the metabolites at a stable dynamic range. Thus, for metabolic pathway modeling, reversibility is taken into consideration by also including the reverse reaction rate at the second step. The reversible rate law take the form of equation 2.6, where V_f represents the max forward reaction rate; K_s and K_p are Michaelis-Menten constant values of substrate and product; K_{eq} is the equilibrium constant of the reversible reaction; S represents the concentration of the substrate and P represents the concentration of the product.

$$v = \frac{V_f/K_s \times (S - P/K_{eq})}{1 + S/K_s + P/K_p} \tag{2.6}$$

Here the reaction rate v represents the net rate of the reversible reaction, thus, in a pathway, it represents the flux J through that reaction step.

2.3 Statistical Learning Methods

2.3.1 Models and Frameworks

In *statistical decision theory*, for given input X, our goal is to seek a function $f(X)$ to predict y [8]. An objective function $L(y, f(X))$ is defined to penalize error of the prediction. Type of objective function differs as we are facing with different types of learning problems. Square error (2.7) is often used for regression problems, categorical cross entropy (2.8) is commonly used for classification problems.

$$L(y, f(X)) = (y - f(X))^2 \tag{2.7}$$

$$L(y, f(X)) = -\sum_{i=1}^{N} [y_i \log \mu_i + (1 - y_i) \log(1 - \mu_i)] \tag{2.8}$$

The objective function is a function of parameters. The learning process is to find a approximation of $f(x)$ that can make useful prediction. There has been complete theories

on linear methods. *Generalized linear model* (GLM) is one of the linear model commonly used for classification tasks (2.9).

$$
\begin{aligned}
Pr(C = j|X = x) &= \frac{\exp(\beta_{j0} + \beta_j^T x)}{1 + \sum_{l=1}^{N-1} \exp(\beta_{l0} + \beta_l^T x)} \quad , j = 1, 2, 3, ..., K-1. \\
Pr(C = K|X = x) &= \frac{1}{1 + \sum_{l=1}^{N-1} \exp(\beta_{l0} + \beta_l^T x)}
\end{aligned} \tag{2.9}
$$

Theories for its properties, regularization and optimization methods has been well-studied [8, 24]. According to the dataset and the learning problem, different types of regularization can be added. For example L1 regularization, also know as Lasso [35], is commonly used for sparsity purpose. It's realized by adding penalty to the parameters besides the error term (2.10).

$$
\hat{\beta} = \arg\min_{\beta}(\|Y - X\beta\|_2^2 + \lambda \sum_{j=1}^{n} |\beta_j|) \tag{2.10}
$$

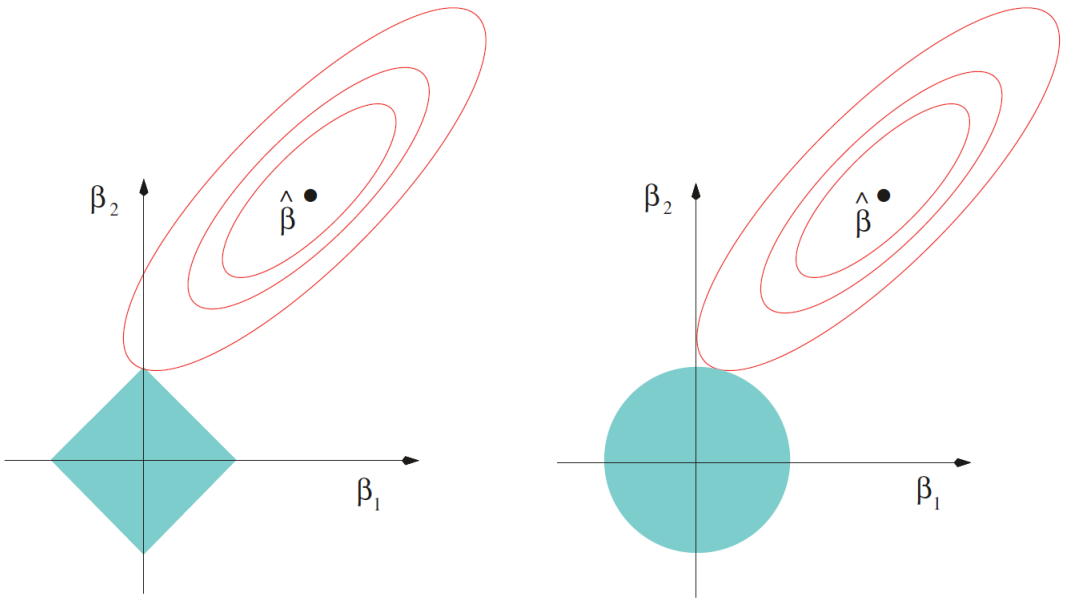

Figure 2.1: Lasso v. Ridge on a two parameter case [8].

By increasing λ to increase penalty, parameters for predictors that contribute less to the prediction will be pushed to zero. Lasso utilize $l1$-*norm* as penalty for parameters. As shown in figure 2.3.1, compared with $l2$-*norm* (Ridge regression), lasso is more likely to set parameters to zero.

Linear methods are easy to interpret and implement, however, for many real applications, linear methods cannot approximate $f(x)$ well enough to generate useful predictions. In order to add nonlinearity, methods like *smoothing*, *local regression*, *neural networks* are developed [8].

Artificial neural networks (NN) inspired by structure of biological neural systems recently have undergone significant developments and have become a powerful machine learning method in many fields of applications [9, 20]. Former work has proofed that multi-layer neural networks are a class of universal approximators [13, 14].

An example of multi-layer fully connected feed-forward neural network is shown in figure 2.2. Yellow nodes represent input variable vector X. Gray nodes represent variables of hidden layers X_j, and the blue node indicate the output variable y which is generally a probability or a probability distribution for classification task or a specific output value for regression task. B_i matrices are weight matrices contain trainable weights mapping variables from previous layer to the next. Besides linear transformation between layers, activation functions can be added to increase nonlinearity. Commonly used activation functions include tanh (2.11), Rectified linear unit (ReLU) (2.12), sigmoid (2.13), and etc.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \tag{2.11}$$

$$f(x) = \begin{cases} 0 \ for \ x < 0 \\ x \ for \ x > 0 \end{cases} \tag{2.12}$$

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{2.13}$$

Number of nodes in the input and output layers are fixed depending on the dataset and the machine learning task, however, number of nodes in hidden layers can be changed by the

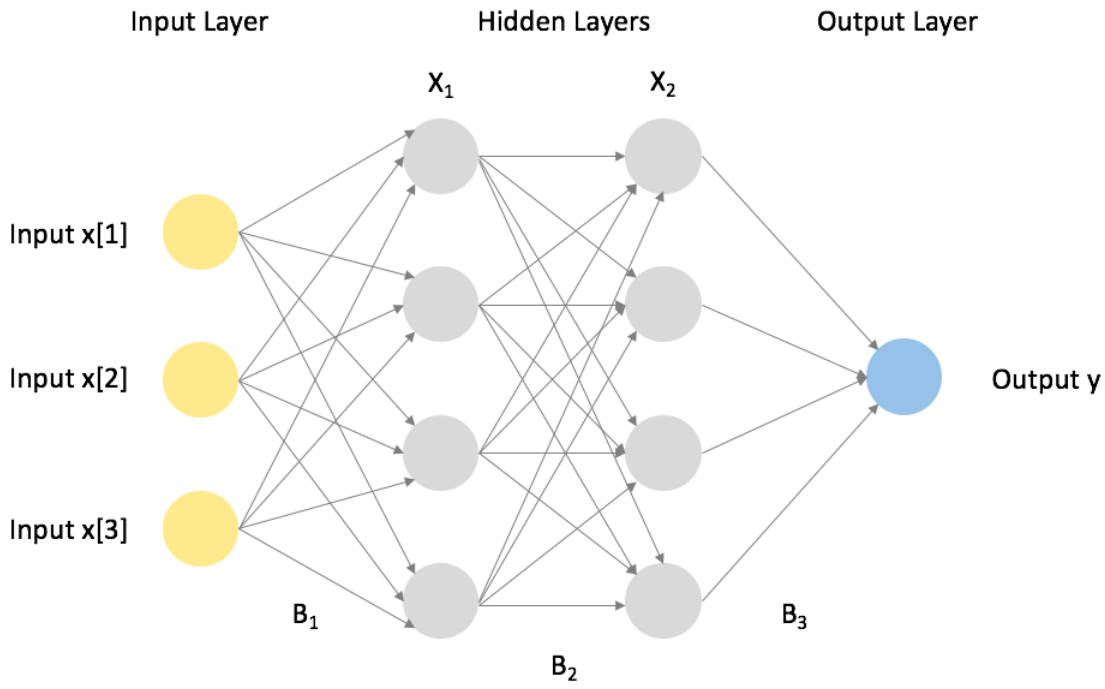

Figure 2.2: Architecture of an artificial neural network with two hidden layers.

user giving flexibility to build a good model to make useful prediction for the learning task. The NN models will be trained on a training set, during which the weight matrices will be optimized so that the model will have a better fitting to the data. The optimization process is based on *backpropagation* (BP) algorithm which utilizing *chain rule* for differentiation to backpropagate the error from objective function to the parameters using gradient decent update rule described in equation 2.14. β is the weight of the model that being updated during each iteration i of training. E represents the error measured by objective function and λ is the learning rate that define the stepsize for each iteration of optimization.

$$\beta_{i+1} = \beta_i + \lambda \frac{\partial E}{\partial \beta_i} \tag{2.14}$$

2.3.2 Feature Selection and Ranking

Simultaneous multi-class feature selection is one of important subareas of machine learning. In the context of biological application, since each feature usually request a series of experimental measurements, it help to make optimal trade-offs between prediction and cost by selecting and ranking features based on their contribution to correct prediction. Traditional feature selection strategies includes *filter methods*, *wrapper methods*, and *embedded methods* [10].

- *Filter methods* use several principle criteria for variable ranking. Scores for all variables are pre-calculated based on the criterion before training. Commonly used criteria include *correlation* (2.15), *mutual information* (2.16), and etc. Filter methods are computationally cheap but not effective compared to other methods.

$$corr(i) = \frac{cov(x_i, y)}{\sqrt{var(x_i) \cdot var(y)}} \tag{2.15}$$

$$\begin{aligned} I(y, x_i) &= H(y) - H(y|x_i) \\ &= -\sum_y p(y) \log(p(y)) + \sum_{x_i} \sum_y p(x_i, y) \log(p(y|x_i)) \end{aligned} \tag{2.16}$$

- *Wrapper methods* use prediction model as a block box to evaluate the score of each subset of features. This class of methods are good in terms of performance, however, computationally expensive especially when applied to high-dimensional datasets. Since every combination of features needs to be fitted to the model, it becomes a NP-hard search problem and makes this method less efficient.

- *Embedded methods* combine the feature selection process with the training process. This class of methods reduce computational cost yet have better performance than *filter methods* thus become a viable and popular choice for many feature selection problems. Commonly used approaches include L1 regularization (2.10), Recursive Feature Elimination (RFE), and etc.

2.4 Significance of the Work

Values of *flux control coefficient* is informative for metabolic engineers to optimize a metabolic pathway in order to increase metabolic flux. Theoretical and computational work has been done to investigate patterns of control through kinetic properties of the pathway. In [16], Kacser, et al. rigorously proved that in order to calculate control coefficient, elasticities information of all enzymes in the system is required. Using this theory, Snell et al. used elasticities of enzymes in the serine pathway determined the control coefficients [31]. In [36], Wang, et al. firstly proposed a computational framework fitting control coefficients using steady state information. However, it requires steady states of all metabolites and the method is sensitive to noise which commonly occurs in biological measurements.

In this project, a framework using statistical approach to predict step with highest control coefficient is proposed. The statistical models are trained on synthetic datasets generated by reversible Michaelis-Menten kinetics (2.6) and return a probabilistic prediction of the dominant step of the control. Feature selection approaches are applied to the model during the training process providing a rank of the features in terms of their contribution to correct prediction. Taking advantages of the predictive approaches, our framework is less compu-

tationally expensive in terms of prediction and more robust to noisy data. The ranking generated by feature selection technique identifies key kinetic information that contributes to the prediction thus offers metabolic engineers a way of leveraging prediction with cost. This framework can be easily generated to linear segments of metabolic pathway with any given number of nodes.

Chapter 3

METHODS

This chapter provides details of the methods as well as implementations for the framework we proposed in this study. The chapter starts with the construction of synthetic dataset, and then, moves on to the statistical learning and feature selection approaches.

3.1 Simulation Methods for Synthetic Dataset

The datasets used to train the predictive model are generated by numerical simulations. The project focus on linear segments of metabolic pathway based on reversible Michaelis-Menten kinetics. The networks take the from of figure 3.1, where S_i indicates metabolites, species with '$'s are boundary species and for convenience of description we define each metabolite as a node of the network. The network can have any given number of nodes. The metabolic flux stars from the source (the first boundary species) of the pathway to the sink (the last boundary species). For kinetics, here we make slight modification of equation 2.6, and take

$$\$S_0 \longleftrightarrow S_1 \longleftrightarrow S_2 \longleftrightarrow ... \longleftrightarrow S_{n-1} \longleftrightarrow \$S_n$$

Figure 3.1: Linear metabolic pathway with two boundary spices and $n + 1$ nodes

the form of equation 3.1, where E_{Total} indicates total amount of the enzyme and k_{cat} indicates the rate constant for forward reaction. This form of expression allows us to calculate flux

control coefficients (CCs) of total enzyme (E_{Total_i}) of each step.

$$\begin{aligned} v &= \frac{V_f/K_s \times (S - P/K_{eq})}{1 + S/K_s + P/K_p} \\ &= \frac{E_{Total} \times k_{cat}/K_s \times (S - P/K_{eq})}{1 + S/K_s + P/K_p} \end{aligned} \tag{3.1}$$

Assuming no prior knowledge on the system, we randomly sample all the kinetic parameters from uniform distributions except the equilibrium constants K_{eq}s are uniformly distributed on a log scale. Readers can see table 3.1 for details of sampling scheme.

Table 3.1: Distributions of randomly sampled kinetic parameters

Parameter	Notation	Distribution	Range
e_i	Total Enzyme	Uniform	$(0.01, 100)$
K_{P_i}	Backward Michaelis-Menten Constant	Uniform	$(0.01, 100)$
$\log(K_{eq_i})$	Equilibrium Constant	Uniform	$(0.01, 100)$
k_{cat_i}	Forward Reaction Rate	Uniform	$(0.01, 100)$
K_{S_i}	Forward Michaelis-Menten Constant	Uniform	$(0.01, 100)$

All the numerical computations are performed on *Tellurium*[4], a Python-based platform for modeling in systems and synthetic biology. It uses *Antimony*[30] as model definition language which is human-readable and -writable. An example of Antimony model used in generating dataset can be found in figure 3.1.

Based on the kinetic parameters randomly generated by the scheme described above, the control coefficients are numerically calculated by calculating steady state change with respect to slight changes in total enzyme (E_{Total_i}) through simulations. Steady state check is performed for each run of the simulation to make sure the control coefficient is calculated when the system reached steady state. A diagrammatic description of the algorithm used to generate one sample is shown in 3.3. The algorithm will run $100,000$ times to construct the entire dataset for each network with n nodes.

```
J0: $S0 -> S1; (e0 * kcat0/Ks0) * (S0 - S1/Keq0) / (1 + S0/Ks0 + S1/Kp0);
J1: S1 -> S2; (e1 * kcat1/Ks1) * (S1 - S2/Keq1) / (1 + S1/Ks1 + S2/Kp1);
J2: S2 -> $S3; (e2 * kcat2/Ks2) * (S2 - S3/Keq2) / (1 + S2/Ks2 + S3/Kp2);
S0 = 14.0;
S1 = 0.0;
S2 = 0.0;
S3 = 3.0;
e0 = 95.6076380375;
e1 = 87.6814878929;
e2 = 61.283808257;
Kp0 = 12.8119631217;
Kp1 = 16.4217510142;
Kp2 = 0.871020612875;
Keq0 = 0.0752446813909;
Keq1 = 0.0618083011984;
Keq2 = 1.5527881969;
kcat0 = 33.2908671763;
kcat1 = 76.4675351307;
kcat2 = 42.4190902369;
Ks0 = 98.2902904769;
Ks1 = 8.55095884724;
Ks2 = 18.0226782237;
```

Figure 3.2: Example of a 4-node model using Antimony Definition

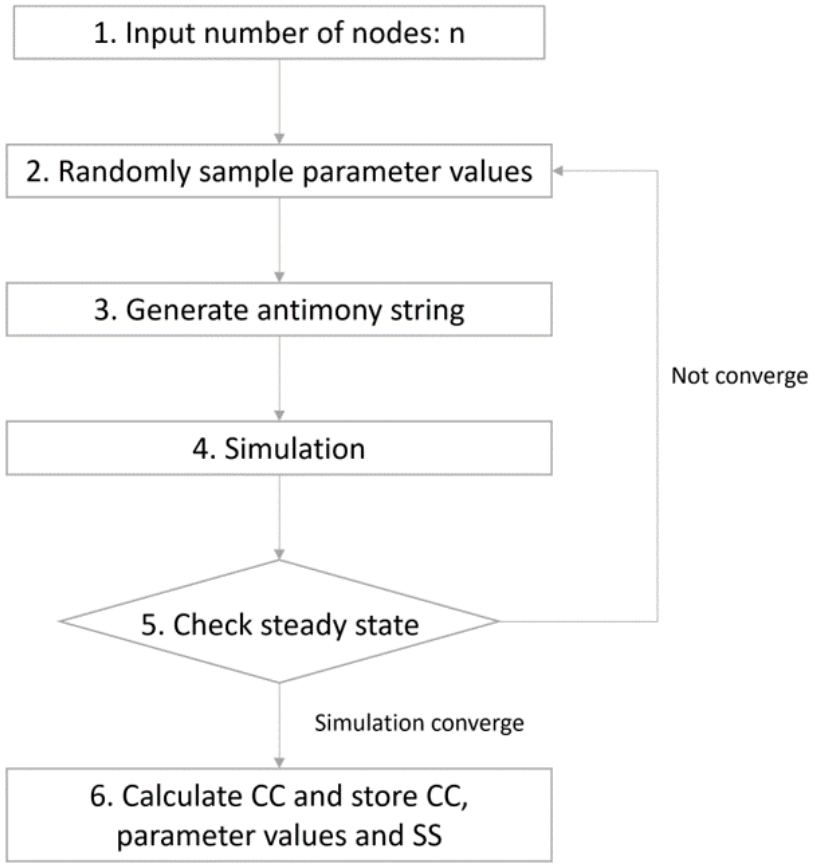

Figure 3.3: Diagrammatic description of the algorithm used to generate data sample.

Distribution of control coefficients of first 300 samples of a 5-node model dataset is shown in 3.4. Each row indicates one sample, each column indicates one reaction step, and the color depth indicates the values of control coefficients. Also, distributions of max control coefficients of 5, 7, 10, and 15 nodes are displayed in figure 3.5.

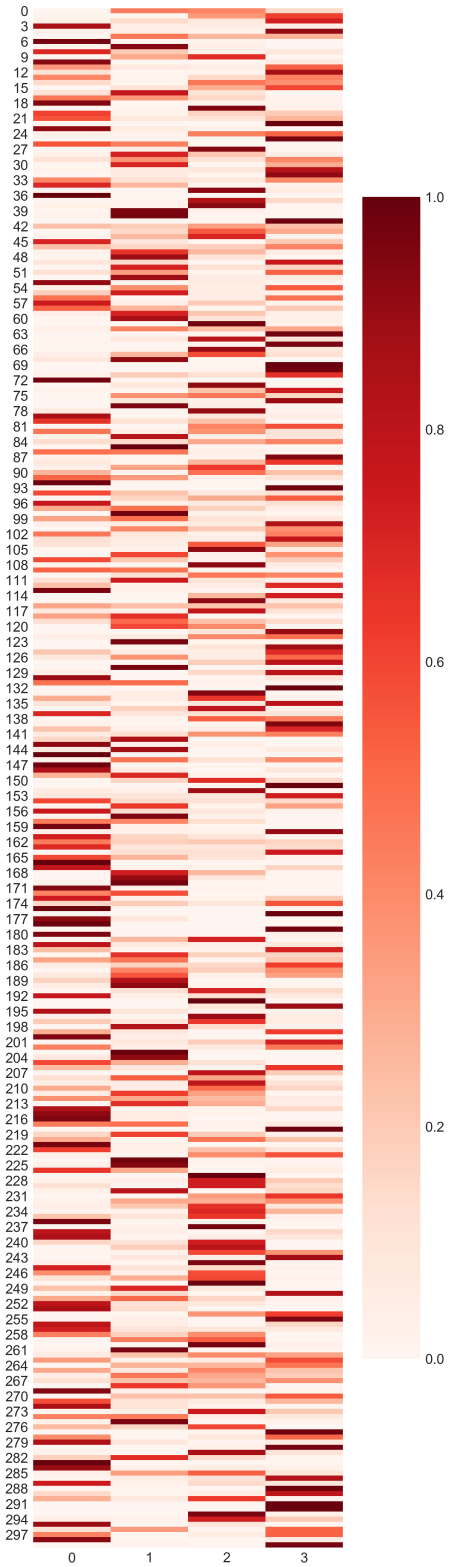

Figure 3.4: Snapshot of the control coefficient generated by simulations

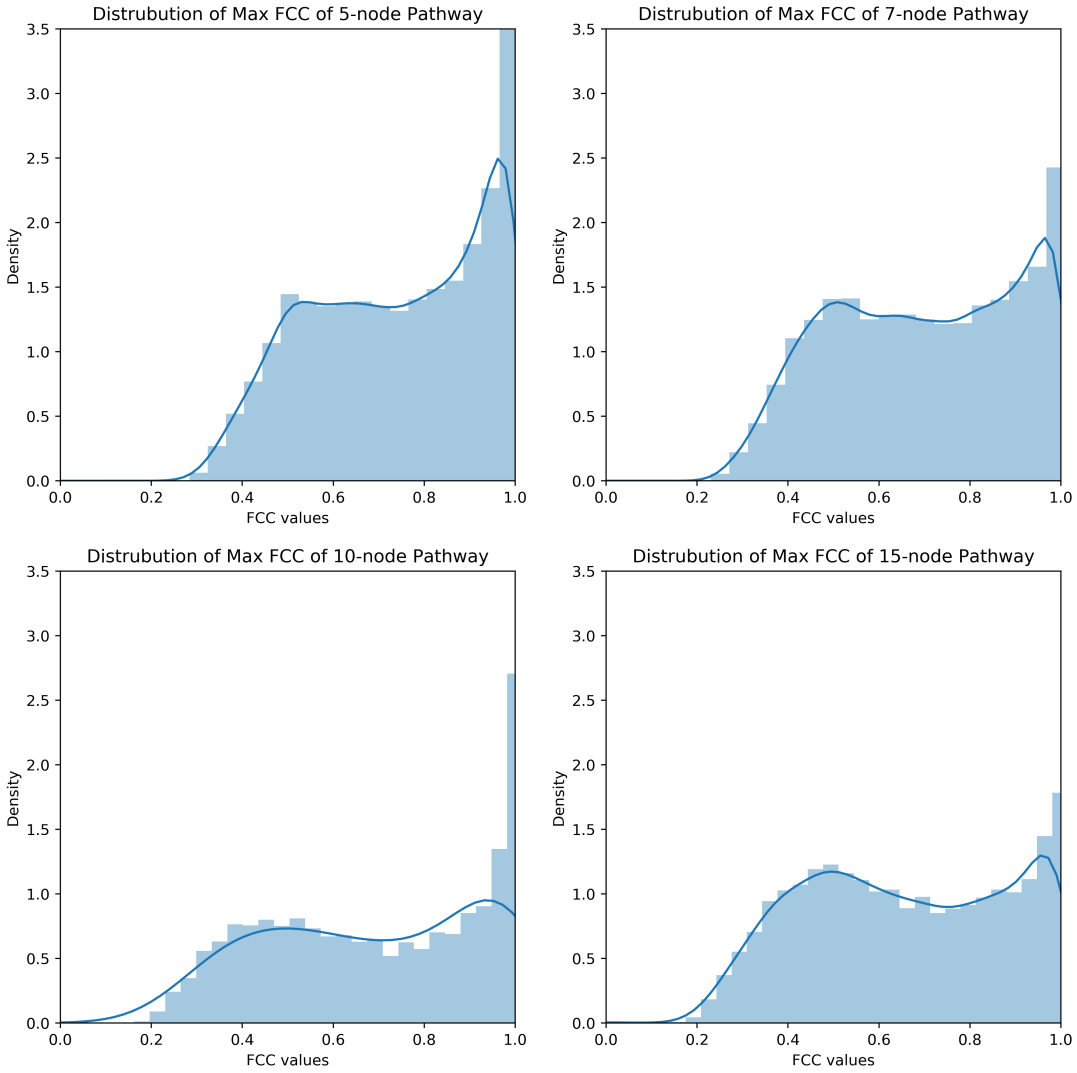

Figure 3.5: Distributions of max control coefficients

3.2 Generalized Linear Model with Group Lasso

3.2.1 Multinomial Logistic Regression

In this study, we are using statistical methods to predict the probability distribution of dominance in flux control. Our goal is to predict the probability of each step being the dominant step. Thus, this problem can be formed as a multi-class classification problem, which takes kinetic information of a pathway as input, and outputs the estimated probability distribution of being the dominant step. *Multinomial logistic regression*, also known as *softmax regression*, is a generalization of *generalized linear model* (GLM) to multi-class prediction problems. For a K-class classification problem, the multinomial logistic regression model forms $K-1$ log-odds as described in equation 2.9.

3.2.2 Group Lasso

Another goal of our project is to get a rank of the parameters that contribute to the prediction of control coefficient. Since all the inputs require experimental measurements, sparsity is also desired besides accuracy. Thus, for both computational efficiency and performance, we applied *embedded feature selection* methods to the linear model in order to acquire feature ranking. Lasso (2.10) is one of the commonly used method for feature selection with GLM. However, for our problem, we want to select key parameters among all kinetic parameters in the model, thus, *group lasso*[22], an extension of traditional lasso method to perform variable selection on predefined groups is applied. It is an intermediate penalty method between $l1$ and $l2$ *norm*. The estimator for group lasso is defined in equation 3.2, where $\mathcal{I}_g$ indicates the group ($g = 1, 2, ..., N$) that variable belongs to. Same kinetic parameters for different enzymes are grouped together.

$$\hat{\beta} = \arg\min_{\beta}(\|Y - X\beta\|_2^2 + \lambda \sum_{g=1}^{N} \|\beta_{\mathcal{I}_g}\|_2) \tag{3.2}$$

In this project, group lasso is implemented with `Pyglmnet`, a Python implementation of elastic-net regularization. A series of penalties are added by changing the value of λ. Ac-

curacy and number of non-zero parameters are plotted as the penalty changes. Also, the feature ranking can be acquired by observing the tendency of each group of parameters going toward zero when penalty is increased.

3.3 Neural Networks

As mentioned in Chapter 2, many prediction functions $f(x)$ can not be well-approximated by a linear model, in the meantime, multi-layer neural networks (NNs) have been proofed to be a class of universal approximators. Thus, in this project, NN is applied as an approach for nonlinear transformation of input features in order to get better predictions. Structure of the NN used in this study is shown in figure 3.6. A batch normalization layer is added to accelerate the training process [34]. The model contains three fully connected layers with `tanh` activation function (2.11) between layers. All input kinetic parameters are flatten into a vector as input layer. Inputs only contain parameters selected by the linear model, and NN here only servers as an approach to enhance prediction performance. For final output, a softmax layer (3.3) is added so that the output layer has the summation property which satisfy the form of a probability distribution.

$$P(y = k|x) = \frac{e^{x^T \omega_k}}{\sum_{i=1}^{K} e^{x^T \omega_i}} \tag{3.3}$$

The implementation utilize `Sequential()` module of Keras[5], a Python-based a high-level neural networks API running on top of Tensorflow[21]. *Categorical cross-entropy* (2.8) is used for objective function, and Adam optimizer[18] is used for optimization of the NN. A 3-fold cross validation is used during training process. In the meantime, in order to prevent overfitting, aggressive dropouts are applied. Mini-batchsize of 32 is used, as in practice, small batchsize has the advantages of faster training speed, thus the training processes may converge faster to the optimal.

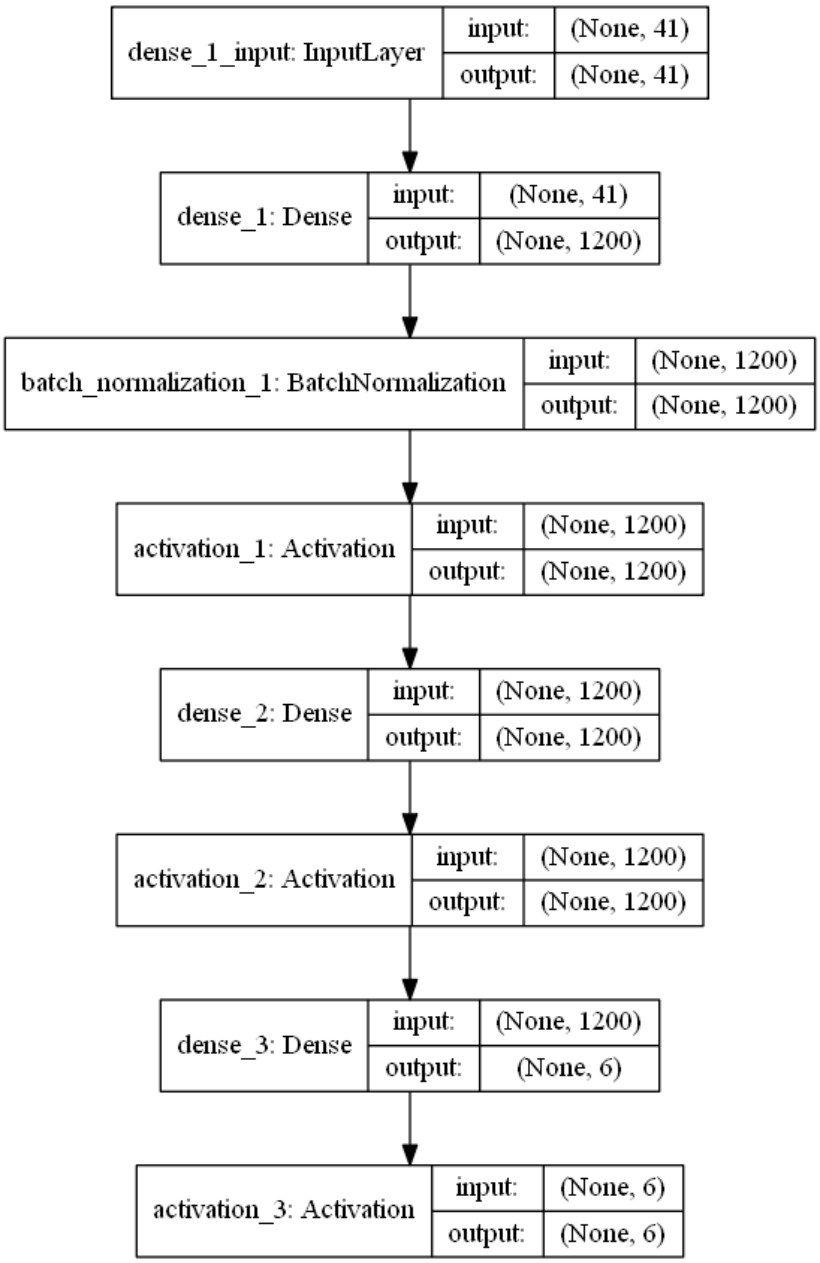

Figure 3.6: Architecture of the multilayer neural network model

3.4 Comparison Study Under Noise

Biological data are always noisy. In this section, methods for a study on model performance against noisy data is introduced. In order to demonstrate the robustness of our predictive model against noises, we added random noise drawn from uniform distribution to the original dataset. To quantify amount of the noise, we define *noise* as a percentage number ranging from 0 to 100%, $noise \in (0, 1)$; and then multiply the original data with a random number drawn from uniform distribution centered at 1 and with random interval define by *noise*, as shown in equation 3.4.

$$p = p \times \epsilon \tag{3.4}$$
$$\epsilon \sim U(1 - \frac{1}{2}noise, 1 + \frac{1}{2}noise), noise \in [0, 2)$$

Numerical computation results under noisy input parameters are used as benchmark. The noise for numerical simulation is added by the same way for predictive models, and control coefficients are numerically calculated with noisy parameters. Then, the index of reaction with largest control coefficient is returned as the result for each sample, and accuracy is calculated based on the results. Results for both predictive method and numerical simulation method are plotted and compared to each other with different amount of noise. A diagrammatic description of this method is shown in figure 3.7.

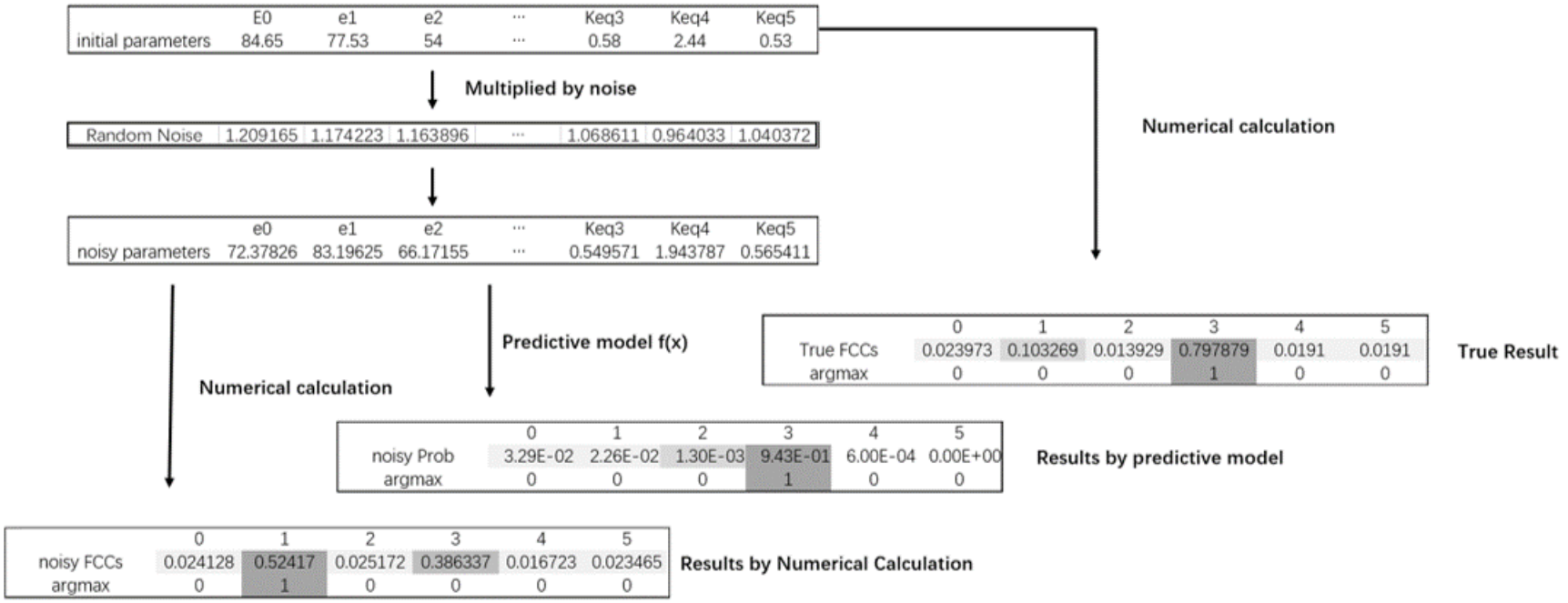

Figure 3.7: Diagram of steps for study under uncertainty

Chapter 4

RESULTS

This chapter describes computational results of this study in detail, includes visualization of the models and regularizations, feature ranking, training processes, and performance of our methods.

4.1 Results of Linear Methods

Figure 4.1 shows fitted coefficient values of the linear models of a five-node pathway which contains four reactions simulated by reversible Michaelis-Menten kinetics (2.6). The coefficients of GLM are visualized by heatmap where the color depth indicates the value of a coefficient.

From top to bottom, there are six different penalty values arranged in an increasing order. The colorbars are fixed, so it is clear to see how coefficients of GLMs changes in response to penalty increase. The sparsity increases as a result of increasing penalty and it is obvious to see that parameters are changing at different rates as penalty increases. When the penalty becomes extremely large, as shown at the bottom of the figure, all the coefficients are set to zero and the model will return the most frequent class for prediction.

As a result of increasing penalty and fewer nonzero coefficients, we are inevitably loosing power in the predictions since all kinetics parameters contain information on the network and reducing number of input feature will, without doubt, compromise the predictions. In figure 4.2, accuracy of the best prediction model under certain penalty and the number of features are plotted against increasing values of penalty.

It is clear to see that the accuracy and number of selected features both drop as penalty increases, however, the number of selected features drops significantly faster than the ac-

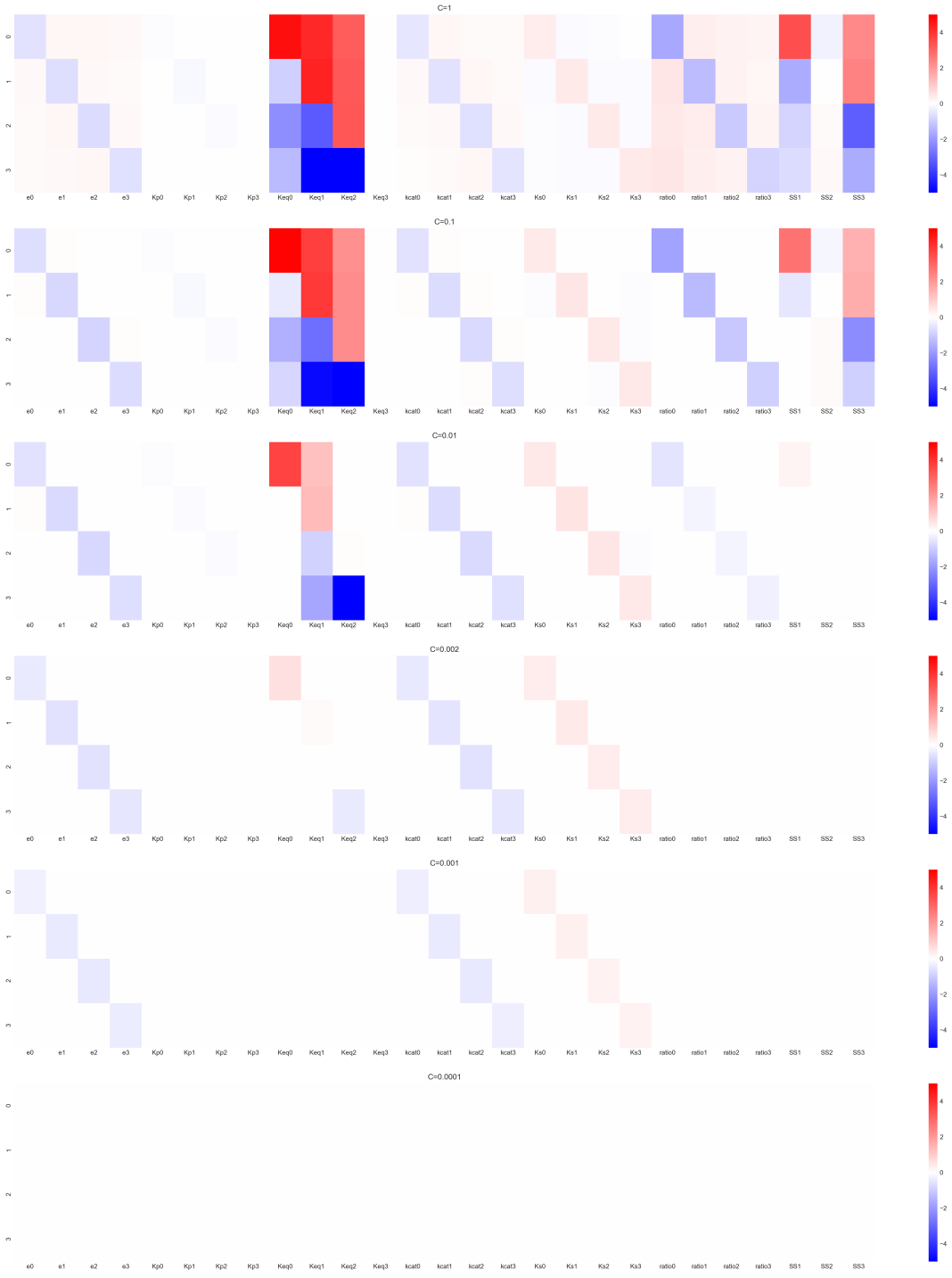

Figure 4.1: Heatmap visualization of GLM coefficients

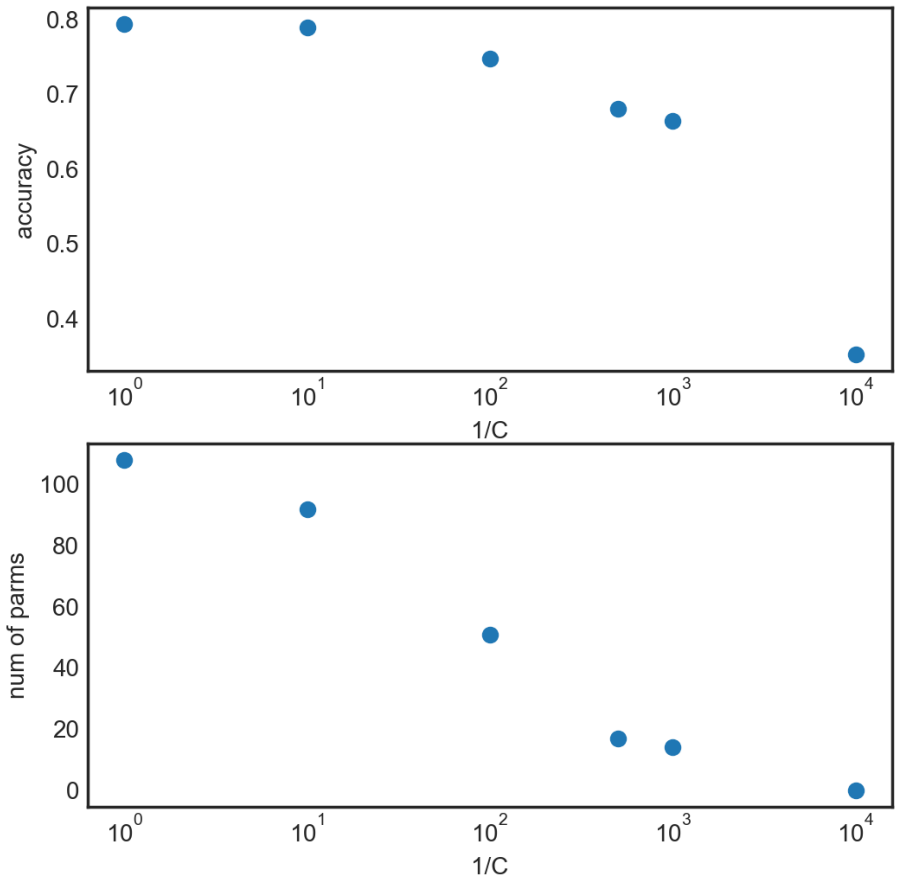

Figure 4.2: Accuracy and number of nonzero parameters under increasing penalty

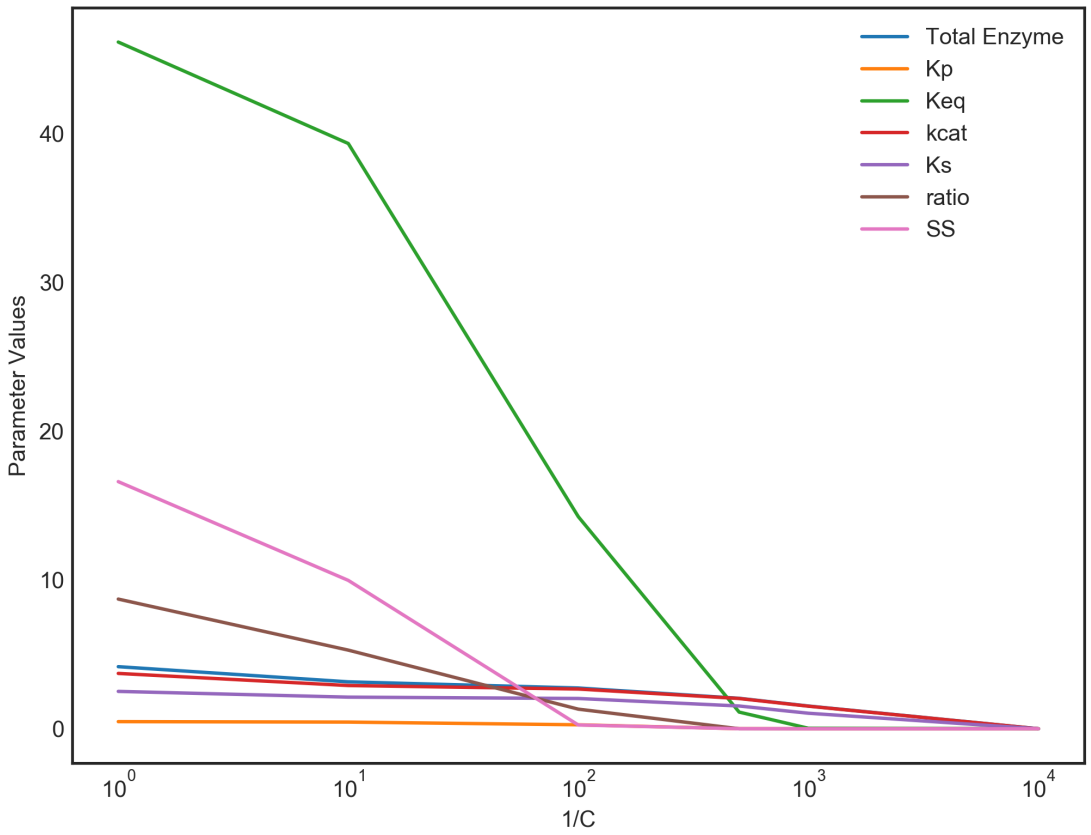

Figure 4.3: Values of coefficients under increasing penalty

curacy in certain regions of penalty. From figure 4.2, we can see that around penalty at $1/C = 10^2$, we can reduce half amount of inputs with only slight drop in prediction.

By comparing the rate that groups of coefficients are pushed to zero, a ranking of grouped features can be acquired. As described in previous chapter, in order to identify the key parameters that contributes to the prediction of control, we used group lasso method and group same kinetic parameters of different enzymes together. In figure 4.3, values of parameters summed over within groups are shown as penalty increase.

Figure 4.3 is another visualization of model coefficients. Same patterns can be observed

in figure 4.1, but here we have a more straightforward view. We can see that coefficients drop as penalty increase, but with different rate. Coefficients of steady state value and equilibrium constant variables have high values at the beginning but soon drop to zero as penalty increase. In the contrast, coefficients of variables for forward MichaelisMenten constant K_s, forward reaction rate k_{cat}, and total enzyme E_{Total} remain at moderate level through the whole process. Here we can have a feature ranking in terms of prediction with GLM, $E_{Total} > k_{cat} > K_s > K_{eq} > Steady\ States$.

4.2 Results of Neural Networks

The predictions with GLM are not satisfying as family-wise error for multinomial logistic regression will increase as number of nodes increase. Here NN as a nonlinear transformation of feature to enhance prediction power is implemented. A typical training process of a 10-node network is shown as an example. The values of objective function (also known as loss function) and accuracy of both training and validation set is shown in figure 4.4. As we can see from the training process, overfitting problem is still exist despite that certain amount of regularization is added to the model. The prediction result is shown in figure 4.5 as a confusion matrix, where x-axis represents predicted labels and y-axis represents correct labels. Good predictions generally mean that most points fall on the diagonal line where the predicted label is the same as correct label.

As we can see from the results, early stop or stronger regularization may need implementing as overfitting is still a problem during training. The total accuracy for this example is 0.94. The NN perform well on predicting pathways where the control concentrate on first few nodes. When the largest control coefficients occur on the last few reactions, the prediction result drops.

For generalization to multiple nodes, the results can be found in table 4.1. NN models have steady performance on pathways with different number of nodes.

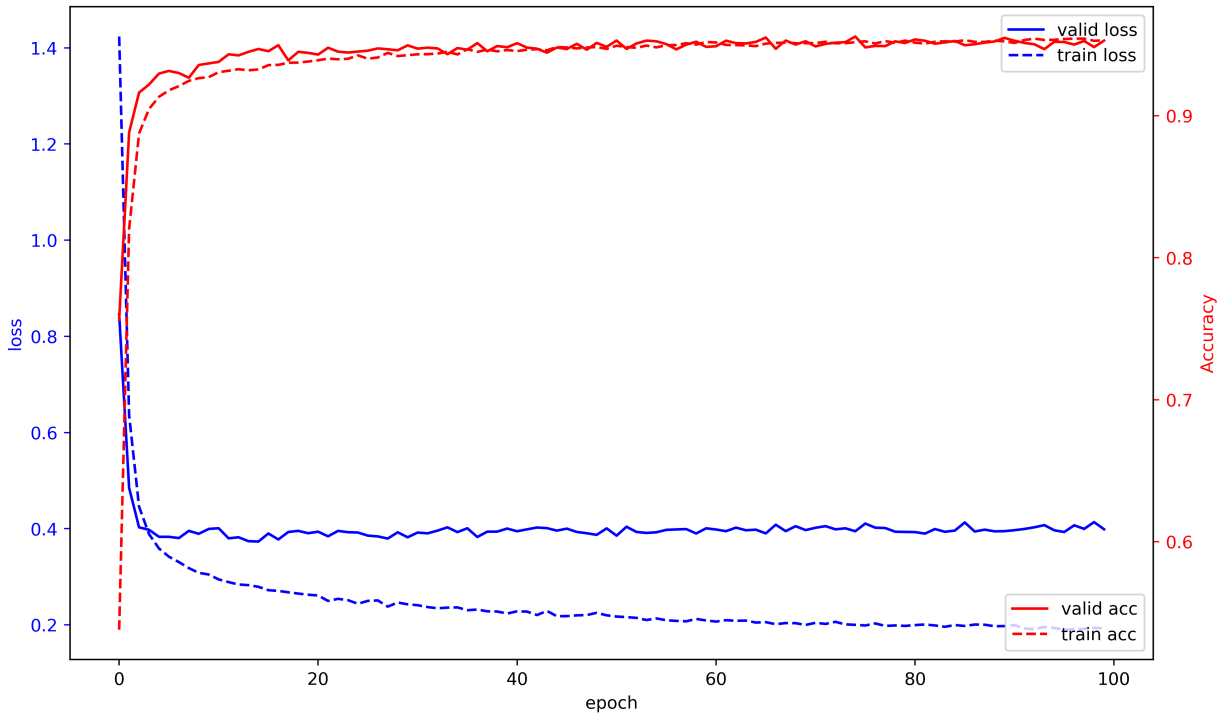

Figure 4.4: Loss and accuracy during training process of NN

Table 4.1: Prediction of test set using NN

Number of Nodes	5	6	7	10	15
Accuracy	0.785	0.879	0.898	0.935	0.878

Figure 4.5: Confusion Matrix of NN model for 7-node pathway

4.3 Results under Uncertainty

As described in previous chapter, a study with noisy input data is carried out by manually including random errors into the original dataset. Quantification of amount of noises has been defined in equation 3.4. In figure 4.6, accuracies of numerical simulation and our current best predictive model, neural network with architecture shown in figure 3.6 with all kinetic parameters inputted, are plotted against different amount of noises. As we can see from the plot, the accuracy of numerical simulation starts at 100% of accuracy and decreases sharply as small amount of noises are added; on the contrast, the NN starts at 96% of accuracy and drops relatively slower than the numerical simulation method. The accuracy of NN also has a significant decrease as the noise goes to 80% percent, where original information is damaged considerably.

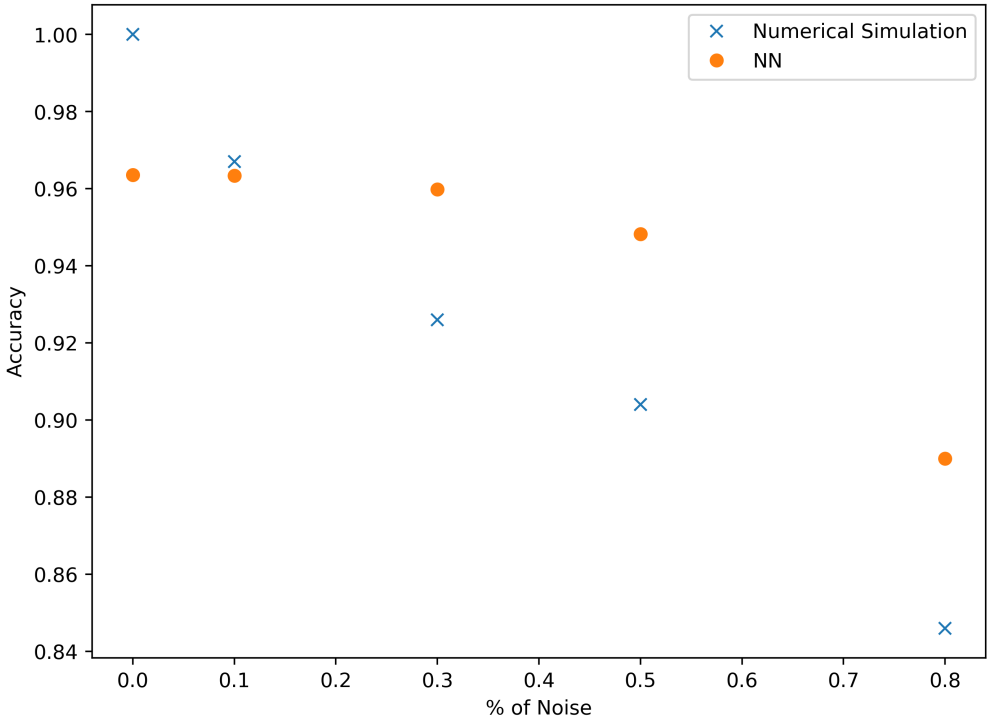

Figure 4.6: Accuracy v. Noise for Numerical Simulation and NN

Chapter 5
DISCUSSIONS

This chapter comprehensively discuss the results as well as approaches of this project. The discussion focus on implications of the results, limitations of the work and possible future work that could been down due to limitations on approaches and time.

5.1 Implications

5.1.1 Trade-off between Accuracy and Cost

The statistical learning approach offers us anther way of optimizing metabolic pathway. Taking advantages of well-studied GLM method, we are allowed to combine the learning process with feature selection technique using embedded feature selection strategy. It offers a means to have an optimal trade-off between accuracy and experimental cost with easy implementation and acceptable computation time. Feature ranking study with GLM shows that we have a chance to reduce number of input features without severely compromising prediction power, since as described above and shown in figure 4.2, as the penalty increases, the accuracy for the predictive model drops slower than the rate that number of input features are eliminated, which gives us a region to optimize the cost with accuracy. We attribute most of this phenomena to the optimized feature ranking (shown in figure 4.3) generated by Group lasso, as the optimization of the coefficients is a convex optimization problem, thus the optimal values of coefficients under given penalty will always be reached.

This work may also be done analytically by making linearization with certain assumptions such as [16], however, with increasing number of nodes in the network, the mathematical analysis is harder to perform, and the assumptions cannot hold for all situations. Thus, the predictive method, on the other hand, shows flexibility to generate to different size of

networks, and acceptable performance for pragmatic uses.

5.1.2 Robustness Against Noise

Biological measurements are always noisy. Kinetic parameters measured by experiments are not necessarily the real values of enzymes working in vivo. Thus, even though we are able to measure all kinetic parameters from the experiments and calculate the FCC through numerical methods, we are still facing with potential errors caused by noises in measurements.

Under the assumption that experimental measurements are centered at true values, here a study simulating the noisy measurement is carried out by sampling noise from random distributions and vary the input kinetics parameters accordingly. With this formulation, noise can be quantified by properties of the distribution, thus, another trade-off determined by potential noise number can be made between two methods. As described in the result chapter, numerical calculated results are sensitive to noise, even at a lower level. The accuracy could drop sharply even with 10% noise. On the other hand, the statistical framework is more robust. Even under the condition that all parameters can be measured, meaning there's no experimental cost limitation thus no cost-accuracy trade-offs need to be made, the statistical framework still have potentially better performance when the expected experimental noise is larger than 10%. Although, this result may be related to the way we are formulating this problem, since predicting the probability of the step with max control coefficient doesn't require accurate calculation of exact values of FCCs, predicting the step with max FCC is also meaningful as the distribution of max control coefficients shown in figure 3.5, under fully randomizations, most pathways contain a dominant step with most control.

5.1.3 Neural Networks as An Effective Approach to Enhance Performance

As shown in the snapshot of the dataset figure 3.4, the distribution of max controlling step is pretty randomized, thus the prediction problem is multi-class and not easy based on the dataset. For prediction power, NN as a nonlinear transformation approach enable probabilistic prediction with useful accuracy. As described in Chapter 2 & 3, by adding

nonlinear activation functions between layers, NN as a class of universal approximators, could perform nonlinear transformations between layers, which offers better approximation of prediction function $f(X)$ for output Y. For the same set of inputs, NN significantly increases the accuracy compared to GLM, make this statistical framework useful for the trade-off with experimental cost without losing too much prediction accuracy.

NN allows flexible design of hidden layers, in both number of nodes and number of layers. With the fact that optimization of NN is not a convex problem, design of NN is more like a line of art, with no theoretical foundation. During the practice, we discovered that for our problem, unlike NN used for computer vision, number of nodes per layer is more important than depth of the network, as with deeper networks, the non-convex optimizations perform worse. As a possible interpretation, more nodes in one layer means more transformations of the data thus may create useful features or combinations of features that make enhancive contributions to prediction. A network with deeper layers is used, shown in figure 5.1, but only with $60\% - 80\%$ of accuracy.

5.2 Limitations and Future work

Our goal of this project is to predict control by predicting the reaction step with largest flux control coefficient (FCC). Thus, we have to assume that FCC is a good standard for quantifying control. However, FCC doesn't provide accurate prediction for large changes in enzyme activities as equation 2.2 doesn't hold as changes of enzyme activities become large., which is one of the potential drawback of the study.

Due to the use of synthetic dataset. we have to make assumptions that our synthetic dataset is a good representation of the real situation going on within the cell, which means terms of E_{Total} has to be a representation of effective concentration of enzyme. Also, as defined by the kinetics (2.6), we have to have orthogonality within our network, which means all enzymes are not interacting with each other and the concentration of metabolites won't affect the effect amount of enzyme. And for the noisy data, the assumption that noisy observations are centered at true values are not always true as some experiments are carried

out under false conditions.

From the approach side, the $l1$ regularization used for feature selection only work well on GLM. However, the results from GLM may be inconsistent with the NN models since NN involves nonlinear transformation of features. Variational dropouts for feature ranking in NN is now an active research filed, which could be implemented for this problem in the future.

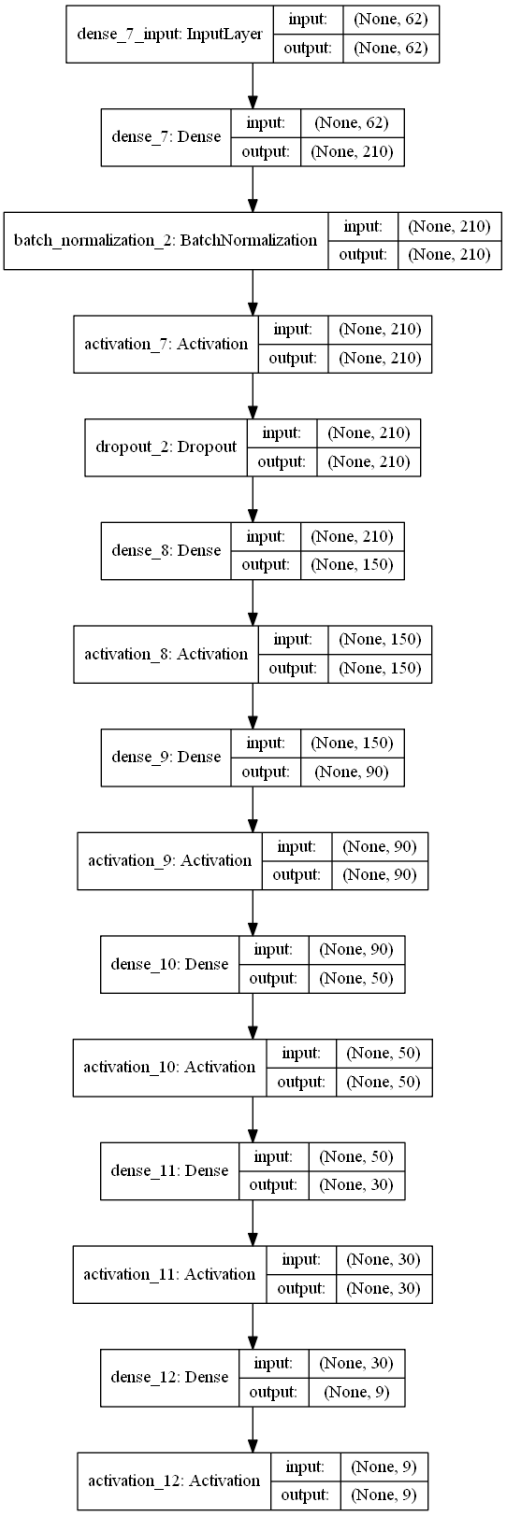

Figure 5.1: Architecture of a deeper NN

Chapter 6

CONCLUSIONS

In this project, we developed a statistical frame work to predict the max controlling step in a linear metabolic pathway given kinetic information as input. The statistical approach is built and tested on a synthetic dataset generated by numerical simulations, and reaches 90% of accuracy given partial kinetic information of the pathway. Our approach is able to provide a feature ranking of the input kinetic information, thus provide a trade-off between prediction accuracy and cost. The accuracy is also tested under different amount of noise. Under the condition of full input of kinetic parameters, statistical approach predicting the max controlling step is more robust against noise compared with numerical calculation as benchmark method. This framework can be generalized to any given number of nodes, and exhibits steady performance. This framework can sever as a tool making useful prediction as well as providing guidance on measurements that shall be taken in order to help metabolic engineer to target the most promising step given limited information.

BIBLIOGRAPHY

[1] Lilia Alberghina and Hans V Westerhoff. *Systems biology: definitions and perspectives*, volume 13. Springer Science & Business Media, 2007.

[2] Frank T Bergmann and Herbert M Sauro. SBW-a modular framework for systems biology. In *Proceedings of the 38th conference on Winter simulation*, pages 1637–1645. Winter Simulation Conference, 2006.

[3] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers and Electrical Engineering*, 40(1):16–28, 2014.

[4] Kiri Choi, J Kyle Medley, Caroline Cannistra, and K Matthias. Tellurium : A Python Based Modeling and Reproducibility Platform for Systems Biology. *bioRxiv*, pages 1–27, 2016.

[5] François Chollet and Others. Keras. \url{https://keras.io}, 2015.

[6] David Fell and Athel Cornish-Bowden. *Understanding the control of metabolism*, volume 2. Portland press London, 1997.

[7] David A FELL and Herbert M SAURO. Metabolic control and its analysis. *The FEBS Journal*, 148(3):555–561, 1985.

[8] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*, volume 1. Springer series in statistics New York, 2001.

[9] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

[10] Isabelle Guyon and André Elisseeff. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research (JMLR)*, 3(3):1157–1182, 2003.

[11] Vassily Hatzimanikatis and James E Bailey. MCA has more to say. *Journal of theoretical Biology*, 182(3):233–242, 1996.

[12] Reinhart Heinrich and Tom A. Rapoport. A Linear SteadyState Treatment of Enzymatic Chains: General Properties, Control and Effector Strength. *European Journal of Biochemistry*, 42(1):89–95, 1974.

[13] Kurt Hornik. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2):251–257, 1991.

[14] Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, 1989.

[15] Henrik Kacser and JA34 Burns. The control of flux. In *Symp. Soc. Exp. Biol.*, volume 27, pages 65–104, 1973.

[16] Henrik Kacser and James A Burns. The molecular basis of dominance. *Genetics*, 97(3-4):639–666, 1981.

[17] Boris N Kholodenko and Hans V Westerhoff. Metabolic channelling and control of the flux. *FEBS letters*, 320(1):71–74, 1993.

[18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[19] J Nathan Kutz. *Data-driven modeling & scientific computation: methods for complex systems & big data*. Oxford University Press, 2013.

[20] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015.

[21] Mart\'\in˜Abadi, Ashish˜Agarwal, Paul˜Barham, Eugene˜Brevdo, Zhifeng˜Chen, Craig˜Citro, Greg˜S.˜Corrado, Andy˜Davis, Jeffrey˜Dean, Matthieu˜Devin, Sanjay˜Ghemawat, Ian˜Goodfellow, Andrew˜Harp, Geoffrey˜Irving, Michael˜Isard, Yangqing Jia, Rafal˜Jozefowicz, Lukasz˜Kaiser, Manjunath˜Kudlur, Josh˜Levenberg, Dandelion˜Mané, Rajat˜Monga, Sherry˜Moore, Derek˜Murray, Chris˜Olah, Mike˜Schuster, Jonathon˜Shlens, Benoit˜Steiner, Ilya˜Sutskever, Kunal˜Talwar, Paul˜Tucker, Vincent˜Vanhoucke, Vijay˜Vasudevan, Fernanda˜Viégas, Oriol˜Vinyals, Pete˜Warden, Martin˜Wattenberg, Martin˜Wicke, Yuan˜Yu, and Xiaoqiang˜Zheng. {TensorFlow}: Large-Scale Machine Learning on Heterogeneous Systems, 2015.

[22] Lukas Meier, Sara Van De Geer, and Peter Bühlmann. The group lasso for logistic regression. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 70(1):53–71, 2008.

[23] L Michaelis and M L Menten. Die Kinetik der Invertinwirkung. *Biochem Z*, 49(February):333–369, 1913.

[24] Kevin P Murphy. *Machine Learning, a Probabilistic Perspective*. The MIT Press, 2012.

[25] David J Newman, Gordon M Cragg, and Kenneth M Snader. Natural products as sources of new drugs over the period 1981- 2002. *Journal of natural products*, 66(7):1022–1037, 2003.

[26] Lauren B Pickens, Yi Tang, and Yit-Heng Chooi. Metabolic engineering for the production of natural products. *Annual review of chemical and biomolecular engineering*, 2:211–236, 2011.

[27] Sauro. *Enzyme Kinetics for Systems Biology*, volume 500. 2011.

[28] Herbert M Sauro. *Systems Biology: Introduction to Pathway Modeling*. Ambrosius Publishing, 2016.

[29] Maurice Scheer, Andreas Grote, Antje Chang, Ida Schomburg, Cornelia Munaretto, Michael Rother, Carola Söhngen, Michael Stelzer, Juliane Thiele, and Dietmar Schomburg. BRENDA, the enzyme information system in 2011. *Nucleic acids research*, 39(suppl_1):D670—-D676, 2010.

[30] Lucian P Smith, Frank T Bergmann, Deepak Chandran, and Herbert M Sauro. Antimony: a modular model definition language. *Bioinformatics*, 25(18):2452–2454, 2009.

[31] Keith Snell and David A Fell. Metabolic control analysis of mammalian serine metabolism. *Advances in enzyme regulation*, 30:13–32, 1990.

[32] Gregory Stephanopoulos. Metabolic Fluxes and Metabolic Engineering. *Metabolic Engineering*, 1(1):1–11, 1999.

[33] Xinxiao Sun, Xiaolin Shen, Rachit Jain, Yuheng Lin, Jian Wang, Jing Sun, Jia Wang, Yajun Yan, and Qipeng Yuan. Synthesis of chemicals by metabolic engineering of microbes. *Chem. Soc. Rev.*, 44(11):3760–3785, 2015.

[34] Christian Szegedy, Scott Reed, Dumitru Erhan, Dragomir Anguelov, and Sergey Ioffe. Scalable, high-quality object detection. *arXiv preprint arXiv:1412.1441*, 2014.

[35] Robert Tibshirani. Regression Selection and Shrinkage via the Lasso, 1996.

[36] Liqing Wang, Inanç Birol, and Vassily Hatzimanikatis. Metabolic control analysis under uncertainty: framework development and case studies. *Biophysical journal*, 87(6):3750–3763, 2004.