Integration of metabolic modelling with machine learning to identify mechanisms underlying antibiotic killing

by

Sarah Natalie Wright

B.S. Massachusetts Institute of Technology (2016)

Submitted to the Department of Biological Engineering in partial fulfillment of the requirements for the degree of Masters of Engineering in Biomedical Engineering at the MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2017

© Massachusetts Institute of Technology 2017. All rights reserved.

Signature redacted

Author ........ Department of Biological Engineering May 12, 2017

Signature redacted

Certified by .................. James J. Collins
Professor of Biological Engineering
Thesis Supervisor

Signature redacted

Accepted by .......... Mark Bathe
Chair, Graduate Program Committee in Biological Engineering
Integration of metabolic modelling with machine learning to identify mechanisms underlying antibiotic killing

by

Sarah Natalie Wright

Submitted to the Department of Biological Engineering on May 12, 2017, in partial fulfillment of the requirements for the degree of Masters of Engineering in Biomedical Engineering

Abstract

Microbial pathogens are becoming a pressing global health issue due to the rapid appearance of resistant strains, accompanied by slow development of new antibiotics. In order to improve these treatments and engineer novel therapies, it is crucial that we increase our understanding of how these antibiotics interact with cellular metabolism. Evidence is increasingly building that the efficacy of antibiotics relies critically on downstream metabolic effects, in addition to inhibition of primary targets. Here we present a novel computational pipeline to expedite investigation of these effects: we combine computational modelling of metabolic networks with data from experimental screens on antibiotic susceptibility to identify metabolic vulnerabilities that can enhance antibiotic efficacy. This approach utilizes genome-scale metabolic models of bacterial metabolism to simulate the reaction-level response of cellular metabolism to a metabolite counter screen. The simulated results are then integrated with experimentally determined antibiotic sensitivity measurements using machine learning. Following integration, a mechanistic understanding of the phenotype-level antibiotic sensitivity results can be extracted. These mechanisms further support the role of metabolism in the mechanism of action of antibiotic lethality. Consistent with current understanding, application of the pipeline to M. tuberculosis identified cysteine metabolism, ATP synthase, and the citric acid cycle as key pathways in determining antibiotic efficacy. Additionally, roles for metabolism of aromatic amino acids and biosynthesis of polyprenoids were identified as pathways meriting further investigation.

Thesis Supervisor: James J. Collins
Title: Professor of Biological Engineering
Acknowledgements

Foremost, I would like to express my sincere gratitude to Prof. James J. Collins and Dr. Jason Yang for their wonderful support and guidance during my time as a student. I am very thankful for the encouragement, insight, patience, and enthusiasm shown in helping me achieve my goals. I cannot imagine having a better advisor and mentor to work with during my degree.

I also want to thank the rest of the Collins lab. In particular, Meagan Hamblin for all the hard work she put into the experiments accompanying this project, and all those who eagerly shared their knowledge with me.

Last but not least, I would like to thank my family, friends, and loved ones. They have provided me with incredible support throughout my time at MIT, and have made it the past five years the experience of a lifetime.
Contents

CHAPTER 1 – INTRODUCTION ..............................................................................................................13
  1.1 GENOME-SCALE METABOLIC MODELS (GSMMs) ................................................................. 14
  1.2 OVERVIEW OF MODELLING AND MACHINE LEARNING STRATEGY ................................. 16

CHAPTER 2 – DEVELOPMENT OF METABOLIC MODELLING STRATEGY 19
  2.1 INTRODUCTION TO METABOLIC MODELLING ....................................................................... 19
  2.2 OVERVIEW OF METABOLIC MODELLING STRATEGY ................................................................. 19
    2.2.1 E. coli model and biomass objective ................................................................................ 19
    2.2.2 Gene Inactivity Moderated by Metabolism and Expression (GIMME) algorithm .......... 22
    2.2.3 Parsimonious Flux Balance Analysis (pFBA) algorithm .................................................. 24
    2.2.5 Sampling the solution space of GSMMs ......................................................................... 26
  2.3 RESULTS & DISCUSSION OF METABOLIC MODELLING IN E. COLI ...................................... 27
    2.3.1 Choice of pFBA as simulation algorithm ......................................................................... 27
    2.3.2 Results of metabolic modelling for E. coli .................................................................... 30
  2.4 CONCLUSIONS ............................................................................................................................. 30

CHAPTER 3 – INTEGRATION OF EXPERIMENTAL DATA USING MACHINE LEARNING ...................... 33
  3.1 INTRODUCTION TO DATA INTEGRATION STRATEGY ............................................................... 33
  3.2 COMPONENTS OF DATA INTEGRATION STRATEGY ................................................................... 33
    3.2.1 In vitro antibiotic susceptibility data from metabolite counter screen ........................... 33
    3.2.2 Preprocessing of modelling and in vitro datasets ............................................................. 34
    3.2.3 Machine learning algorithm I: Elastic Net with cross validation .................................. 34
    3.2.4 Machine learning algorithm II: Gradient Boosting regression .......................................... 36
    3.3.5 Hypergeometric statistics for subsystem analysis ............................................................ 37
  3.4 RESULTS & DISCUSSION ............................................................................................................ 38
    3.4.1 Data integration results ..................................................................................................... 38
    3.4.2 Biological insights from data integration ......................................................................... 43
  3.5 CONCLUSIONS ............................................................................................................................. 45

CHAPTER 4 – APPLICATION TO M. TUBERCULOSIS ......................................................................... 47
  4.1 INTRODUCTION ............................................................................................................................. 47
  4.2 METABOLIC MODELLING OF M. TUBERCULOSIS .................................................................... 48
    4.2.1 Selection of the iSM810 model of M. tuberculosis .............................................................. 48
    4.2.2 Initial conditions for metabolic modelling ......................................................................... 49
    4.2.3 Implementation of pFBA with iSM810 ............................................................................. 49
4.2.4 Summary of metabolic modelling results ........................................................................51

4.3 DATA INTEGRATION ........................................................................................................51

4.3.1 Collection of antibiotic susceptibility data for *M. tuberculosis* .........................51
4.3.2 Data integration results for *M. tuberculosis* ..........................................................54
4.3.3 Biological insights from data integration .................................................................54
4.4 CONCLUSIONS .............................................................................................................59

CHAPTER 5 – CONCLUSIONS AND FUTURE DIRECTIONS ..............................................61
List of Figures

Figure 1. Form of the stoichiometric matrix of a GSMM. Each column represents the stoichiometry of the reactions in the toy metabolic network on the left. Negative and positive values correspond to metabolites that are reactants and products respectively. Figure adapted from O'Brien et al., 2015. 15

Figure 2. Overview of the modelling and integration pipeline constructed. (1) Computational metabolic modelling of growth under different media conditions to create metabolite-specific models. (2) Integration of metabolic modelling results with experimental data to predict relationships between individual reactions and antibiotic susceptibility. These two steps are followed by analysis of the results to determine biological relevance. 17

Figure 3. Schematic representation of the GIMME Algorithm. The GIMME algorithm takes 3 inputs: a GSMM, gene expression data, and an objective function. It also requires two user-defined thresholds: a gene expression threshold to define which genes and their associated reactions are active and inactive, and a condition that the biomass objective must be maintained above a defined value. 23

Figure 4. Schematic of the Parsimonious Flux Balance Analysis Algorithm (pFBA). This algorithm optimizes the GSMM for maximum growth rate while minimizing total flux through the network. It first identifies pathways that result in a lower growth rate (purple) along with those that cannot carry a flux due to absence of necessary factors (blue). It then classifies genes and reactions into five categories: 1. essential, 2. pFBA optima, 3. enzymatically less efficient (ELE) 4. Metabolically less efficient (MLE) 5. pFBA no-flux. Shown is an example of the implementation with a toy metabolic network. Here, Gene A, classified as MLE, represents a suboptimal enzyme that reduces the growth rate if used. Gene B does not have a necessary precursor so is classified as pFBA no-flux. Genes E and F require two different enzymes to catalyze the same reaction which Gene D can do alone; therefore they are classified as ELE. Gene G is essential as there is no alternate pathway available. Genes C and D represent the most efficient pathway so are classified as pFBA optima. Figure adapted from Lewis et al., 2010. 25

Figure 5. Distribution of number of flux-carrying reactions after simulation with GIMME and pFBA. The y-axis refers to the number of unique simulations with a certain number of flux-carrying reactions predicted. Arrows indicate the number of flux-carrying reactions in the control condition: 455 and 451 respectively. The total 208 conditions are shown. 29
Figure 6. Network utilization of iJO1366 simulations by subsystem. Values are displayed as a percentage of the total number of reactions present in a subsystem. Essential reactions are reactions that carry a flux in all simulations. Accessible reactions are flux-carrying in $\geq 1$ simulations. Inaccessible reactions are never predicted to carry a flux in the simulated conditions. Results from all 208 simulations are included..............................................................31

Figure 7. Distribution of IC$_{50}$ values from *E. coli* metabolite counter-screen. 208 metabolic conditions are shown, with a bin size of 0.05. All values are normalized as described in section 3.2.2. The IC$_{50}$ of the control is equal to 9.56 $\mu$g/mL.......................................................39

Figure 8. Distribution of non-zero coefficients in the Elastic Net Solution for iJO1366. A total of 177 non-zero coefficients are shown with bin size of 0.02. Positive coefficients represent reactions predicted to be associated with increase the IC$_{50}$ of *E. coli*, while negative coefficients represent reactions predicted to be associated with greater susceptibility to ampicillin. ..................................................................................................................................................40

Figure 9. Distribution of non-zero importance scores in the Gradient Boosting solution for iJO1366. A total of 175 non-zero importance scores are shown coefficients are shown with bin size of 0.002. A higher importance score reflects that reaction is predicted to have a greater effect on the antibiotic susceptibility of *E. coli* to Ampicillin, positive or negative. .................................................................................................................................................................41

Figure 10. Network utilization of iSM810 simulations by subsystem. Values are displayed as a percentage of the total number of reactions present in a subsystems. Essential reactions are reactions that carry a flux in all simulations. Accessible reactions are flux-carrying in $\geq 1$ simulations. Inaccessible reactions are never predicted to carry a flux in the simulated conditions. Results from all 333 simulations are included..............................................................50

Figure 11. Differences in antibiotic susceptibility following metabolite counter screening in *M. bovis* BCG. Values are the log$_2$ transform of the fold change in IC$_{90}$. Higher values represent an increased IC$_{90}$ and therefore a decrease in antibiotic susceptibility. Results are shown for Ethambutol (ETH), Rifampicin (RIF), Isoniazid (INH), and Ciprofloxacin (CIP). Two control conditions are present: 1) the control wells that contained no cells, and 2) the wells containing 1% DMSO used to control for antibiotics and metabolites solubilized in DMSO......53

Figure 12. Distribution of non-zero importance scores from Gradient Boosting of iSM810 results. All importance scores are log-transformed and grouped with a bin size of 0.5 following transformation. The total number of non-zero importances are 126 (INH), 127 (ETH), 157 (RIF) and 124 (CIP). A higher importance score reflects that reaction is predicted to have a greater effect on the susceptibility of *M. tuberculosis* to the respective antibiotics, positive or negative. ......................................................................................................................................................55
Figure 13. Comparison of simulated reaction fluxes of ATP Synthesis and Biomass Growth Rate. Values are from metabolic modelling of *M. tuberculosis* using the iSM810 model. All 333 modelled metabolic conditions are shown. Points in the red group correspond to models that show increased metabolic activity with a minimal increase in growth rate.

List of Tables

Table 1. List of metabolites comprising the core biomass objective function for iJO1366. Model names represent abbreviations used in the GSMM. Letters in brackets refer to: [e] extracellular, [p] periplasmic, [c] cytoplasmic. GAM refers to the growth-associated maintenance requirement of ATP.

Table 2. P-values for E. coli subsystems following hypergeometric testing of results from Elastic Net and Gradient Boosting. Bold indicates p-values < 0.05. Values with * indicate that the result corresponds to under representation of reactions from the subsystem in the machine learning solutions. The table is sorted by the p-values form elastic net, with over represented subsystems first.

Table 3. Summary statistics on *M. tuberculosis* models. Total number of unique metabolites, reactions, genes and exchange reactions are shown.

Table 4. P-values for *M. tuberculosis* subsystems following hypergeometric testing of results from Gradient Boosting. Bold indicates p-values < 0.05. Values with * correspond to under representation of reactions from the subsystem in the machine learning solution. Results for Isoniazid (INH), Ethambutol (ETH), Rifampicin (RIF), and Ciprofloxacin (CIP) are shown.
Chapter 1 – Introduction

Microbial pathogens continue to be a major health issue due to antibiotic resistance and persistence against current treatments. Many bacterial species now have strains that have been identified as multi-drug resistant. These species include *Mycobacterium tuberculosis* (TB) which infects one third of the world’s population and causes approximately 2 million deaths annually (CDC, 2017). Others include *Staphylococcus aureus* and *Pseudomonas aeruginosa* which are major causes of life-threatening hospital acquired infections. In addition to resistance due to mutations to the bacterial genome, many species also avoid cell death during treatment with antibiotics by entering a persistent state. These persister cells are characterized by very low metabolic activity and correspondingly low uptake of compounds from the microenvironment. Most antibiotics target cell growth processes but these are rendered ineffective against persister cells because the cells are not actively growing. Exacerbating these challenges is the fact that development of new antibiotics is diminishing (Brown and Wright., 2016). Therefore, it is critical to increase our understanding of the factors contributing to resistance and persistence so that new therapeutics can be developed.

Traditional understanding is that antibiotics act by inhibiting essential processes in the cell, thus leading to cell death. However, there is a growing body of evidence suggesting that the efficacy of antibiotics crucially depends on downstream metabolic processes in addition to direct inhibition. For example, many antibiotics have been shown to increase production of reactive oxygen species (ROS) which then contribute to cell death (Kohanski et al., 2007). Further investigation into these relationships is imperative to understanding the action of current antibiotics, and for investigation of novel methods for treating infections.

In this thesis, I used a systems approach to develop a pipeline to integrate computational metabolic modeling data with experimental measurements of antibiotic susceptibility. *In vitro* experiments were performed using addition of metabolites to the media in order to systematically perturb cellular metabolism. The response of the bacteria to antibiotics was then used to measure the effect of the perturbations on antibiotic sensitivity. These metabolic effects were estimated by genome-scale modelling, and the datasets were combined using machine learning to generate hypotheses about the biological mechanisms underlying the *in vitro* results. The goal of creating this pipeline was to streamline the process of using the comprehensive information contained in genome-scale metabolic models (GSMMs) to interpret experimental results. This streamlined systems approach makes revealing biological insights more accessible to researchers who do not have a strong background in computational techniques. The pipeline was developed using *E. coli*, but was constructed to be generalizable to other organisms. Following development, I successfully applied the approach to *M. tuberculosis*, which generated biological insights into the relationship between metabolic processes and the action of drugs used for treatment of *M. tuberculosis* infections.
1.1 Genome-scale metabolic models (GSMMs)

GSMMs are reconstructions of the known biochemical reactions and gene-associations of an organism’s metabolism. There are three components required for simulation: an $S$ matrix, reaction constraints, and an objective function. The models use the stoichiometry of the reactions to characterize the metabolic network, and this is then represented by the $S$ matrix that relates all unique metabolites in the model to all reactions (Figure 1). In addition to the $S$ matrix, each reaction is constrained by the maximum rate at which the reaction can proceed based on enzyme kinetics, uptake and secretion rates, and/or thermodynamic constraints (Price et al., 2003). For any simulation, the steady-state flux predicted for a reaction must fall within these constraints. Irreversible reactions have a lower bound of zero, and reversible reactions have upper and lower bounds with non-zero values. These constraints can be modified, allowing for modelling of different conditions. The final component needed for simulation is an objective function. This is a reaction within the model that is optimized when linear programming is applied to the GSMM. The objective is usually set as the biomass function – a pseudo-reaction that is used to predict the growth rate of a cell by including accurate proportions of all necessary precursors for biomass. In addition to the components necessary for simulation, GSMMs contain a wealth of other information about the metabolic network. This includes metabolic subsystems and all known gene-reaction associations. Subsystems classify the reactions based on the major metabolic pathways that they partake in, such as membrane metabolism, nucleotide biosynthesis, and energy metabolism. Gene-reaction associations are useful for quickly relating reaction results to appropriate genes and they also allow genes to be used as constraints.

The simplest approach for characterizing the state of the metabolic network is Flux Balance Analysis (FBA). It takes the $S$ matrix, upper bounds, lower bounds, and objective function as inputs and produces a predicted steady-state flux for every reaction in the model as well as the optimal growth rate. This approach assumes that there is no net production or consumption of mass within the system.

FBA maximizes $Z = c \cdot v$ \hspace{1cm} (Eq.1)

subject to

$S \cdot v = 0$ \hspace{1cm} (Eq.2)

$lb \leq v \leq ub$ \hspace{1cm} (Eq.3)

where $Z$ is the value of the biomass objective, $c$ is the stoichiometric representation of the biomass reaction, $v$ is the vector of predicted reaction fluxes, $S$ is the model’s stoichiometric matrix, $lb$ is the vector of the lower bounds, and $ub$ is the vector of upper bounds. Eq.2 represents the mass balance. In addition to FBA, a range of techniques have been developed to produce the CobraToolbox (The openCOBRA Project, 2016) for working with GSMMs. These tools allow for
the creation of unique models for given conditions; for instance a model can be created to simulate addition of a metabolite to the media. These unique models can then be used to interpret experimental results. This project utilizes a number of these techniques and then combines the results with experimental data to create biologically meaningful results.

Figure 1. Form of the stoichiometric matrix of a GSMM. Each column represents the stoichiometry of the reactions in the toy metabolic network on the left. Negative and positive values correspond to metabolites that are reactants and products respectively. Figure adapted from O’Brien et al., 2015.
1.2 Overview of modelling and machine learning strategy

The overall pipeline can be divided into two major steps: Metabolic Modelling and Data Integration (Figure 2). In the metabolic modelling step, a GSMM is used to create an array of models, each of which simulates the addition of a different metabolite to the media. To create these unique models, the initial conditions set for the GSMM are first adjusted to match the media composition. Then, a simulation algorithm is used to maximize the growth rate. This produces a new model which predicts the importance of each reaction under the given conditions. From this predicted metabolic state, the flux vectors from FBA are gathered and used to assess the variation between each of the metabolite-specific models. Once this data is collected, the data integration step can be performed. Data integration additionally requires an experimental data set comprised of measurements of antibiotic susceptibility for cells grown with each of the simulated metabolites. The goal of combining the two datasets, simulated and experimental, is to identify the metabolic reactions responsible for changes in the antibiotic susceptibility. These relationships are revealed by implementing machine learning algorithms which discover how much of the variation in the experimental data can be explained by variation in each individual reaction, and weigh the reactions accordingly. By analyzing the reactions identified, it is then possible to generate hypotheses about biological processes. Chapter 2 describes the development of the metabolic modelling strategy and this is followed by the development of the data integration strategy in Chapter 3. Once the pipeline was constructed using E. coli, it was then applied to M. tuberculosis in Chapter 4.
Figure 2. Overview of the modelling and integration pipeline constructed. (1) Computational metabolic modelling of growth under different media conditions to create metabolite-specific models. (2) Integration of metabolic modelling results with experimental data to predict relationships between individual reactions and antibiotic susceptibility. These two steps are followed by analysis of the results to determine biological relevance.
Chapter 2 – Development of Metabolic Modelling Strategy

2.1 Introduction to metabolic modelling

In order to model the metabolic network under different conditions it is necessary to create context-specific simulations. Genome-scale metabolic models (GSMMs) contain all known information about the metabolic network of an organism. When investigating particular conditions much of this information is extraneous, so simulation methods condense the model to reveal the parts of the network relevant to the context. In particular, I was interested in investigating which reactions are predicted to be active when growth in the presence of different media metabolites is simulated. From the several algorithms available for this purpose I chose to investigate the use of two: Gene Inactivity Moderated by Metabolism and Expression (GIMME), and Parsimonious Flux Balance Analysis (pFBA). Both were implemented to compare their ability to produce context-specific models that showed variation in utilization of the metabolic network in response to changes in the computational media.

GIMME and pFBA were chosen as the candidate algorithms based on their demonstrated ability to simulate in vitro conditions and their ease of implementation. With the goal of creating a pipeline that will be useful to other researchers with limited computational knowledge, it was important to choose openly sourced methods. Both of these methods exist within the COBRA Toolbox and can be used with Matlab (Mathworks) or Python. In addition, they have reasonable requirements for computational power, input data, and degree of user interaction. Other methods require access to a computing cluster, complex function inputs, or much larger datasets (Blazier and Papin, 2012).

2.2 Overview of metabolic modelling strategy

2.2.1 E. coli model and biomass objective

The iJO1366 model of E. coli, published by Orth et al. in 2014, represents the most comprehensive GSMM of not only E. coli, but also of any organism (Feist and Palsson, 2008). For these reasons it was chosen as the model for development of this computational pipeline. The first reconstruction of the metabolic network of E. coli K-12 MG1655 was created in 2000. Since then it has been updated using known genomic and biochemical information, both curated and newly discovered. The construction of the iJO1366 model took a previously updated E. coli GSMM and added to the network based on analysis of additional genetic information from all E. coli strains as well as
results of further experimentation. It includes 1366 genes, 2251 metabolic reactions, and 1136 metabolites, with the reactions sorted into 37 distinct subsystems of metabolism. From the commercially available metabolite libraries used for experimentation, I was able to identify 208 unique metabolites also present in the iJO1366 model with corresponding transport reactions describing exchange between the extracellular space and the interior of the cell. These 208 metabolites represent the “simulated conditions” for which uptake can be accurately estimated for *E. coli* using the iJO3166 model.

For simulation, the “core” biomass function was used as the objective function (Table 1). This reaction is the commonly used measure of growth rate in GSMMs and encodes depletion of biomass precursors at rates based on experimentally determined values (Feist and Palsson., 2010). Each metabolite in the reaction has units of mmol/gDW (mmol per gram cell dry weight) and the flux through the reaction has units of h\(^{-1}\) to represent the exponential growth rate of the organism. Components of the reaction include amino acids, cofactors, membrane lipids, inorganic ions, DNA, RNA and growth-associated maintenance (GAM) of ATP. The “core” reaction contains only essential components of growth, as opposed to the “wild-type” reaction which includes all wild-type cellular components of *E. coli*. The “core” reaction was chosen as it produces the most accurate results for growth on minimal media conditions (Orth et al., 2014). As part of the creation of iJO1366, this function was updated to reflect the current knowledge of the growth requirements of *E. coli*. 

20
Table 1. List of metabolites comprising the core biomass objective function for iJO1366. Model names represent abbreviations used in the GSMM. Letters in brackets refer to: [e] extracellular, [p] periplasmic, [c] cytoplasmic. GAM refers to the growth-associated maintenance requirement of ATP.

<table>
<thead>
<tr>
<th>Substrates</th>
<th>Coef. (mmol/gDW)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Protein</strong></td>
<td></td>
</tr>
<tr>
<td>ala-L[c]</td>
<td>L-Alanine</td>
</tr>
<tr>
<td>arg-L[c]</td>
<td>L-Arginine</td>
</tr>
<tr>
<td>asn-L[c]</td>
<td>L-Asparagine</td>
</tr>
<tr>
<td>asp-L[c]</td>
<td>L-Aspartate</td>
</tr>
<tr>
<td>cys-L[c]</td>
<td>L-Cysteine</td>
</tr>
<tr>
<td>glt-L[c]</td>
<td>L-Glutamine</td>
</tr>
<tr>
<td>glu-L[c]</td>
<td>L-Glutamate</td>
</tr>
<tr>
<td>gly[c]</td>
<td>Glycine</td>
</tr>
<tr>
<td>his-L[c]</td>
<td>L-Histidine</td>
</tr>
<tr>
<td>ile-L[c]</td>
<td>L-Isoleucine</td>
</tr>
<tr>
<td>leu-L[c]</td>
<td>L-Leucine</td>
</tr>
<tr>
<td>lys-L[c]</td>
<td>L-Lysine</td>
</tr>
<tr>
<td>met-L[c]</td>
<td>L-Methionine</td>
</tr>
<tr>
<td>phe-L[c]</td>
<td>L-Phenylalanine</td>
</tr>
<tr>
<td>pro-L[c]</td>
<td>L-Proline</td>
</tr>
<tr>
<td>ser-L[c]</td>
<td>L-Serine</td>
</tr>
<tr>
<td>thr-L[c]</td>
<td>L-Threonine</td>
</tr>
<tr>
<td>trp-L[c]</td>
<td>L-Tryptophan</td>
</tr>
<tr>
<td>tyr-L[c]</td>
<td>L-Tyrosine</td>
</tr>
<tr>
<td>vaL[c]</td>
<td>L-Valine</td>
</tr>
<tr>
<td><strong>DNA</strong></td>
<td></td>
</tr>
<tr>
<td>datp[c]</td>
<td>dATP</td>
</tr>
<tr>
<td>dctp[c]</td>
<td>dCTP</td>
</tr>
<tr>
<td>dgtp[c]</td>
<td>dGTP</td>
</tr>
<tr>
<td>dtt[p]</td>
<td>dTTP</td>
</tr>
<tr>
<td><strong>RNA</strong></td>
<td></td>
</tr>
<tr>
<td>ctp[c]</td>
<td>CTP</td>
</tr>
<tr>
<td>gtp[c]</td>
<td>GTP</td>
</tr>
<tr>
<td>utp[e]</td>
<td>UTP</td>
</tr>
<tr>
<td><strong>Murein</strong></td>
<td>murein5px4p[p]</td>
</tr>
<tr>
<td>linked murein units</td>
<td>0.019</td>
</tr>
<tr>
<td><strong>LPS</strong></td>
<td></td>
</tr>
<tr>
<td>kdo2lipid4[e]</td>
<td>KDO(2)-lipid IV</td>
</tr>
<tr>
<td><strong>Lipid</strong></td>
<td></td>
</tr>
<tr>
<td>pe160[c]</td>
<td>phosphatidylethanolamine (dihexadecanoyl, n-C16:0)</td>
</tr>
<tr>
<td>pe160[p]</td>
<td>phosphatidylethanolamine (dihexadecanoyl, n-C16:0)</td>
</tr>
<tr>
<td>pe161[c]</td>
<td>phosphatidylethanolamine (9-enoyl, n-C16:1)</td>
</tr>
<tr>
<td>pe161[p]</td>
<td>phosphatidylethanolamine (9-enoyl, n-C16:1)</td>
</tr>
<tr>
<td><strong>Ions</strong></td>
<td></td>
</tr>
<tr>
<td>2fe2s[c]</td>
<td>[2Fe-2S] iron-sulfur cluster</td>
</tr>
<tr>
<td>4fe4s[c]</td>
<td>[4Fe-4S] iron-sulfur cluster</td>
</tr>
<tr>
<td>bmocogdp[c]</td>
<td>bis-molybdopterin guanine dinucleotide</td>
</tr>
<tr>
<td><strong>GAM</strong></td>
<td></td>
</tr>
<tr>
<td>atp[c]</td>
<td>ATP</td>
</tr>
<tr>
<td>h2o[c]</td>
<td>H2O</td>
</tr>
<tr>
<td><strong>Products</strong></td>
<td></td>
</tr>
<tr>
<td>adp[c]</td>
<td>ADP</td>
</tr>
<tr>
<td>h[c]</td>
<td>H+</td>
</tr>
<tr>
<td>p[c]</td>
<td>Phosphate</td>
</tr>
<tr>
<td>pni[c]</td>
<td>Diphosphate</td>
</tr>
</tbody>
</table>
2.2.2 Gene Inactivity Moderated by Metabolism and Expression (GIMME) algorithm

The goal of the GIMME algorithm is to integrate gene expression data into metabolic modelling to increase the accuracy of simulations. GIMME (Figure 3) uses gene expression data to create a context-specific network that maximizes the biomass objective of the GSMM. An expression threshold is supplied by the user that predicts whether a reaction is active or inactive based on gene-reaction associations. First, the model is optimized using Flux Balance Analysis (FBA) to determine the maximum biomass value. Then, inactive reactions are removed to reduce the model and the objective is maximized again. A target range for biomass set as a percentage (commonly 90%) of the maximum value from FBA. The resulting model is then tested to see if it achieves an objective value within the target range and if it fails it is deemed nonfunctional and a set of inactive reactions are added back to the model restore the biomass objective value. An inconsistency score is calculated based on the product of the distance from the expression threshold and the necessary flux for each reaction that is reactivated. The set of reactions to be reactivated is chosen using linear programming to minimize the inconsistency score. The resulting model produced by GIMME represents the metabolic state that is a best fit for the expression data given the applied constraints (Becker and Palsson, 2008). All reactions are classified as “Expressed,” “Unexpressed,” or “Unknown” based on the mRNA data, and reactions removed from or added back to the model are classified as “Downregulated” or “Upregulated” respectively. In addition to classifying the reactions, GIMME also generates a context-specific model that contains only the information relevant to the conditions simulated.

GIMME was implemented with the iJO1366 model using the createTissueSpecificModel method in the COBRA Toolbox. Computation was done in Matlab (Mathworks) with the Gurobi 6.0.4 solver. Gene expression data was collected from microarrays performed on E. coli cells grown in MOPS minimal media using Affymetrix GeneChip E. coli 2.0 chips (Dwyer et al., 2014). Expression profiles were background adjusted and normalized using Robust Multiarray Averaging (Bolstad et al., 2003). MOPS minimal media was chosen for data collection because the absolute concentrations of all components are known. The threshold for active genes was set to 60%, thus keeping genes with expression levels in the top 60%. The binary prediction of active versus inactive genes was used as the input for GIMME. In addition, the threshold on the objective was set to maintain growth within 90% of the optimum. Simulations were performed with these parameters to create metabolite-specific models.
Figure 3. Schematic representation of the GIMME Algorithm. The GIMME algorithm takes 3 inputs: a GSMM, gene expression data, and an objective function. It also requires two user-defined thresholds: a gene expression threshold to define which genes and their associated reactions are active and inactive, and a condition that the biomass objective must be maintained above a defined value.
2.2.3 Parsimonious Flux Balance Analysis (pFBA) algorithm

Parsimonious flux balance analysis (Figure 4) is a linear programming optimization algorithm for GSMMs that performs two functions: (1) optimization of the biomass objective function, and (2) minimization of total reaction flux through the GSMM (Feist et al., 2007). In doing so it assumes that under conditions of exponential growth there is selection for the fastest-growing strains that use the smallest amount of resources.

Reactions are sorted into five gene-associated classes: (1) essential for growth, (2) part of the optimal pFBA solution, (3) part of enzymatically less-efficient pathways, (4) part of metabolically less efficient pathways, or (5) cannot carry a flux under the given conditions. In the first step the optimal growth rate is calculated by setting the objective to the biomass function and performing FBA. This step also identifies essential reactions. Second, flux variability analysis (FVA) is used with no biomass constraint to identify reactions that cannot carry a flux. Third, with the lower bound of the biomass objective set to the identified optimal solution, FVA is again performed to identify enzymatically less-efficient pathways. Fourth, linear programming is used to minimize the absolute total flux through all reactions with the biomass value fixed at the optimum. Fifth, the minimum total flux is imposed as an additional constraint and FVA is again performed to identify alternate pathways to the optimal solution. The final outputs of the pFBA algorithm are a context-specific model and the classifications.

The COBRA Toolbox implementation of pFBA was used for simulation using default options. Computation was done in Matlab (Mathworks) with the Gurobi 6.0.4 solver. Flux through all reactions, including those without a gene-association, was minimized using the default tolerance of 1E-6. Metabolite-specific models were produced for each of the 208 metabolites being investigated.
Figure 4. Schematic of the Parsimonious Flux Balance Analysis Algorithm (pFBA).
This algorithm optimizes the GSMM for maximum growth rate while minimizing total flux through the network. It first identifies pathways that result in a lower growth rate (purple) along with those that cannot carry a flux due to absence of necessary factors (blue). It then classifies genes and reactions into five categories: 1. essential, 2. pFBA optima, 3. enzymatically less efficient (ELE) 4. Metabolically less efficient (MLE) 5. pFBA no-flux. Shown is an example of the implementation with a toy metabolic network. Here, Gene A, classified as MLE, represents a suboptimal enzyme that reduces the growth rate if used. Gene B does not have a necessary precursor so is classified as pFBA no-flux. Genes E and F require two different enzymes to catalyze the same reaction which Gene D can do alone; therefore they are classified as ELE. Gene G is essential as there is no alternate pathway available. Genes C and D represent the most efficient pathway so are classified as pFBA optima. Figure adapted from Lewis et al., 2010.
2.2.4. Initial conditions and outputs of simulation

To simulate *E. coli* metabolism in the MOPS minimal media used in the *in vitro* screen, the initial conditions of the GSMM were set to only allow exchange reaction flux of the compounds in the media, and oxygen. Metabolites that are not rate-limiting are effectively unconstrained, and thus in the simulation were set with an upper bound of 1000 and a lower bound of -1000. Rate-limiting metabolites are given lower bounds that reflect physiological limits in mmol/h. Note that exchange reactions in iJO1366 are defined as “metabolite ↔” meaning that a negative value represents uptake of the metabolite. A unique constraint was imposed for each of the conditions to be simulated. The corresponding exchange reaction was given a lower bound of -10 mmol/h and an upper bound of 0, allowing uptake of the metabolite to be included in the solution. The lower bound value of -10 is the standard value used for glucose exchange when simulating growth on media containing glucose at 10mM. For consistency across simulations, this value was chosen as the standard for growth on alternate carbon sources, nitrogen sources, and sulfur sources.

Following simulation with pFBA and GIMME, flux balance analysis (FBA) was performed on the resulting models. The “core” biomass function was maximized to give the set of reactions contributing to the optimal solution. The number of reactions with a non-zero flux was defined as the network-utilization. The distribution in network-utilization of simulated conditions was used as an unsupervised indicator of the ability of the algorithms to simulate variation in the metabolic network in response to the presence of the given metabolites.

2.2.5 Sampling the solution space of GSMMs

The results of GSMM simulations do not necessarily represent unique flux states due to the underdetermined nature of the models. Instead, they represent a space of possible flux rates. For accuracy, it is necessary to uniformly sample this solution space to get the most likely flux state for each reaction in the network. To complete this step I used optGpSampler, a sampling tool specifically designed for sampling solutions of GSMMs (Megchelenbrink et al., 2014). This algorithm is an improvement on the sampling algorithm in the COBRA Toolbox. Both are based on Artificial Centering Hit-and-Run (ACHR), but the optGpSampler implementation is up to 40 times faster and has better convergence for models containing more than 500 reactions.

Hit-and-run is a Markov Chain Monte Carlo method that produces a chain of consecutive sample points. It starts from a point $x_0$ within the sample space. Then a direction $u_0$ is chosen using a uniform distribution and step size $\lambda_0$ is selected such that the sampler remains within the constraints of the solution space. The next point $x_1$ is determined by stepping $\lambda_0$ in the direction $u_0$, with this process being repeated until convergence (Smith, 1984). Artificial Centering improves hit-and-run by reducing the limitation that hit-and-run disproportionally sample points near the boundary (Kaufman and Smith, 1998). ACHR uses the points sampled so far to estimate an ‘artificial’ center of the solution space, and then uses a randomly chosen direction from a
previously sampled point to this artificial center. This allows the sampler to more easily explore elongated directions within the solution space. To achieve this, ACHR requires a warm-up phase to generate a set of previous points with uniform distribution for direction selection.

The COBRA Toolbox’s gpSampler (Schellenberger and Palsson, 2009) is an implementation of ACHR with a modified warm-up phase because a uniform set of directions is not appropriate for the irregular solution space of GSMMs. The gpSampler algorithm uses linear programming to select $T$ warm-up points for sampling. The process has two stages: (1) $2n$ warm-up points are created by optimizing the flux rate of each reaction, and (2) the remaining $T-2n$ points are generated by random assignment of weights to the fluxes that should be optimized. Because of the use of linear programming, the warm-up points all reside on the boundary of the solution space. To prevent sampling being biased towards the boundary, gpSampler uses a linear transformation to move the points into the interior of the solution space. Following this is the sampling stage where each warm-up point is used to generate $T$ chains of length $k$ (a provided step count). The end points of each chain are returned as the samples.

OptGpSampler (Megchelenbrink et al., 2014) uses the first step of gpSampler’s warm-up procedure, but does not perform the linear-programming step due to computation time and a tendency for the weighted points to bias the algorithm due to a limited set of optimal solutions. In the sampling step it only selects a sample at each $k$ iterations and uses $p$ processors to generate chains in parallel. This results in fewer chains used in total, but each of them is longer by a factor of $T/p$. Overall, the optGpSampler algorithm provides a more efficient method for sampling the solution space while maintaining a similar performance in simulation results.

For each of the sampled conditions, optGpSampler was used with the Gurobi 6.0.4 solver to generate 10,000 samples using 500 steps and 4 parallel processors. The median of the 10,000 samples was taken to represent the most likely flux state for each reaction.

2.3 Results & Discussion of Metabolic Modelling in E. coli

2.3.1 Choice of pFBA as simulation algorithm

Implementation of GIMME and pFBA with iJO1366 yielded network utilization results indicating that both algorithms were able to reveal metabolic diversity under the simulated conditions (Figure 5). GIMME gave a range of 21 reactions carrying flux around the control of 455 and a standard deviation of 4.02 reactions. pFBA gave a range of 27 flux-carrying reactions around the control of 451 and a standard deviation of 4.89 reactions. In order to maintain the unsupervised nature of the approach, the identity of the reactions subject to variation was not investigated at this stage. Overall, pFBA gave more varied flux states in the resulting models.

Therefore, pFBA was chosen as the algorithm for metabolic modelling as it was expected to better match the diversity in the in vitro experiments. pFBA also has three major advantages
over GIMME: (1) it is more widely applicable to organisms where expression data is difficult to procure or gene-reaction associations are not well characterized, (2) it does not rely on the moderate association between mRNA and protein levels, and (3) it does not involve the imposition of arbitrary constraints.

For *E. coli*, the relationships between genes and reactions are very well understood - 94% of protein-associated reactions having gene associations in the iJO1366 model. In addition, the ease of working with *E. coli* makes collecting gene expression data straightforward. However for other organisms, such as TB, limited knowledge of the genome and experimental challenges make algorithms requiring expression data intractable.

Moreover, the correlation between mRNA levels and protein abundance is not strict (Ideker et al., 2001; Griffin et al., 2002). Therefore, determining the relevance of reactions based almost entirely on mRNA abundance, as in GIMME, may lead to questionable results. Many factors such as post-translational modifications, post-transcriptional regulation, and enzyme kinetics affect the mRNA-protein relationship. Furthermore, these processes vary between pathways, making their effects on the relationship inconsistent across the metabolic network (Wessely et al., 2011).

Finally, the GIMME algorithm imposes threshold on the gene expression data to decide which reactions are active. The arbitrary nature of this threshold is unavoidable. In this study the threshold was chosen as 60% based on the number of reactions “Upregulated” as the threshold varied. These are reactions that should have been removed based on the expression data but were required for a functional model resulting in the gene expression constrain being overridden. The integrity of the gene expression constraint is therefore decreased with more “Upregulated” reactions. For iJO1366 the relationship between the number of “Upregulated” reactions and the threshold imposed followed an S-shaped curve (data not shown). The 60% threshold corresponds to the end of the lag-phase of this curve, therefore it imposes the strictest expression constraint on the model with a minimal number of “Upregulated” reactions.

Algorithms exist that attempt to address these limitations of integrating gene expression data but they are inappropriate for this application due to the type of data required and the difficulty of implementation. By comparison, pFBA avoids these issues by not relying on gene expression data or arbitrary thresholds. Despite not taking *in vitro* data as an input, it has been demonstrated that computed growth rates using pFBA are consistent with data from *E. coli* and that the flux-carrying reactions in the resulting models are supported by transcriptomic and proteomic data (Lewis et al., 2010). Therefore, pFBA is valid for simulating cell metabolism under different conditions. The flux-carrying reactions in the resulting simulated models correspond to the “essential” and “pFBA optima” categories produced by pFBA.
Figure 5. Distribution of number of flux-carrying reactions after simulation with GIMME and pFBA. The y-axis refers to the number of unique simulations with a certain number of flux-carrying reactions predicted. Arrows indicate the number of flux-carrying reactions in the control condition: 455 and 451 respectively. The total 208 conditions are shown.
2.3.2 Results of metabolic modelling for *E. coli*

Sampling was performed on the models derived from pFBA simulations. Variation of non-zero flux values for reactions was less than $1 \times 10^{-6}$, and in cases where the network utilization itself varied, this occurred in less than 5% of samples. No reactions showed multiple modes. Therefore, the median value was taken to represent the most likely value for each reaction.

With the final simulation data, analysis of the network utilization within subsystems was performed (Figure 6). Overall, using the metabolic modelling of 11.5% (208/1805) of the metabolites present in iJO1366 resulted in 39.3% (1015/2583) of the total reactions being predicted to be active in response to at least 1 metabolite condition. The remaining reactions were never predicted to carry flux so were classified as inaccessible by metabolic modelling. Of the active reactions, 28.5% (289/1015) carried flux in all simulated conditions. These reactions were classified as essential because they represent essential processes for growth and are always part of the optimal solution. The remaining 71.5% (726/1015) of active reactions showed variability in whether they were predicted to carry a flux between the conditions and so were classified as accessible by metabolic modelling. Subsystems with a direct relationship to the simulated metabolites, such as amino acid systems and alternate carbon metabolism, showed very little essentiality, but a high incidence of accessible reactions. As expected, some subsystems show very low accessibility. This can be attributed to the lack of relevant metabolites screened. Furthermore, reactions associated with less efficient pathways and non-growth associated processes would not be identified using pFBA so fall into the inaccessible categories.

2.4 Conclusions

Utilizing pFBA and optGpSampler I was able to successfully create the metabolic modelling stage of the overall pipeline and implement it with the iJO1366 model of *E. coli*. It is clear that the model shows variation in the flux-carrying reactions that the metabolic network uses to achieve optimal growth under the different simulated conditions. Though only 11.5% of the metabolites present in iJO1366 were simulated, the modelling was able to access 39.3% of the reactions. Furthermore, the modelling was also successful in reducing the dimensionality of the GSMM by identifying the optimal metabolic state. This result will allow meaningful conclusions to be drawn about the underlying biological mechanisms present once integrated with in-vitro data. These conclusions hold across both algorithms implemented, giving confidence to the ability of simulation to produce variation in response to the change of a single metabolite in the media. The strategy was designed to be easily deployable to any GSMM of interest so will be practical for my application to *M. tuberculosis*, and for future work with other organisms.
in ≥ 1 simulations. Accessible reactions are never predicted to carry a flux in the simulated conditions. Results from all 208 simulations are included.

Accessory reactions that carry a flux in all simulations. Accessible reactions are flux-carrying displayed as a percentage of the total number of reactions present in a subsystem. Essential values are

**Figure 6. Network utilization of 10136 simulations by subsystem.**
Chapter 3 – Integration of Experimental Data using Machine Learning

3.1 Introduction to data integration strategy

In order to extract biological insights about bacterial phenotypes from the *in vitro* measurements and metabolic modelling, it is necessary to combine these datasets using regression. In this chapter the simulated reaction fluxes from the metabolic modelling described in Chapter 2 will be used as regression inputs, and the *in vitro* measurements of antibiotic susceptibility will be used as the regression output. The measurements taken are the IC$_{50}$S for each metabolite condition, defined as the concentration of antibiotic needed to inhibit 50% of growth. Machine learning was used for integration of the two datasets so that the results are generated in an unbiased manner.

Two different machine learning algorithms for regression were implemented with the data generated for *E. coli*: Elastic Net and Gradient Boosting. Elastic Net is a linear regression technique and was chosen because the individual relationships between reaction fluxes and IC$_{50}$ values are most closely represented as linear. In addition, Elastic Net capably handles the sparseness of the metabolic modelling data. Gradient Boosting is an ensemble regression technique that is also powerful with underdetermined models and is designed to work with data that has weak relationships between variables. It is often used in biological applications such as fitting transcriptomic data.

As with the simulation techniques in Chapter 2, these algorithms were implemented on open source platforms, namely the scikit learn package for Python. By implementing both of these methods I was able to extract information about which reactions from the simulated models impact whether *E. coli* is likely to be more or less susceptible to antibiotics. This allows for generation of hypotheses that can either be confirmed from literature or investigated further.

3.2 Components of data integration strategy

3.2.1 *In vitro* antibiotic susceptibility data from metabolite counter screen

*In vitro* experiments provide antibiotic susceptibility data in terms of the IC$_{50}$. To collect this data, a metabolite counter screen was performed with Ampicillin. *E. coli K12 MG1655* cells were grown in MOPS minimal medium with 0.2% glucose and Ampicillin was added to the *E. coli* cells
at varying concentrations and then dispensed into PM1-4 metabolite compound plates (Biolog). After 4 hours, optical density measurements were taken to assess growth. IC\textsubscript{50}s were determined for each metabolite-antibiotic combination by fitting the Hill equation to optical density measurements in Matlab (Mathworks). Results from replicates were combined before fitting, with the exclusion of outliers, and the IC\textsubscript{50} values were extracted.

3.2.2 Preprocessing of modelling and \textit{in vitro} datasets

Flux values for reactions in the simulated results were preprocessed to make them appropriate for use in machine learning. First, reactions directly related to the imposed conditions were removed from the model. These reactions represent the initial conditions, rather than perturbations to the network so should not be treated as independent variables as they would bias the results. In contrast, transport reactions of the minimal media components vary in response to the presence of unrelated metabolites, so these reactions were retained.

Second, the pFBA algorithm converts all reversible reactions into separate forward and reverse components. In many cases this resulted in a single reaction being present twice in the data, or a backward reaction being present with a positive flux value. All backward components were transformed to negative values and results from forward and backward reactions were combined to ensure single representation of every reaction. The directionality of exchange reactions was also reversed from “metabolite↔” to “←+metabolite” for consistency across all reversible reactions.

Finally, the simulated data was scaled using the MaxAbsScaler function (Scikit-Learn, 2017). This function treats each reaction individually and scales the data based on the maximum absolute value for each reaction. The maximum absolute value is set to 1.0 and the remaining data is scaled accordingly between 0.0 and ±1.0. This method does not center the data so the significant sparsity that is present in the modelling data set was preserved. In addition, it maintains the sign of all values in the data. IC\textsubscript{50} data from the metabolite screen was normalized to the ratio:

\[
\frac{\text{IC}_{50,\text{metabolite}}}{\text{IC}_{50,\text{control}}} \quad (\text{Eq.4})
\]

Data from each of the four compound plates was normalized based on the results of the control well in each plate, hence controlling for variation between the plates.

3.2.3 Machine learning algorithm I: Elastic Net with cross validation

Elastic Net is a linear regression algorithm that combines the properties of two other linear regression methods: Ridge Regression, and Least Absolute Shrinkage Selection Operator (LASSO)
Regression. Both of these methods are extensions of basic linear regression, as follows. Linear regression builds a model from supplied data by creating a linear combination of the inputs (Eq.5).

\[
y(x, w) = c_0 + c_1x_1 + c_2x_2 + \cdots + c_nx_n \tag{Eq.5}
\]

\[
\min_c ||cx - y||_2^2 \tag{Eq.6}
\]

For this application, each value in \(x\) corresponds to a reaction in the metabolic modelling data, and \(w\) is the measured IC50 data. The coefficients are determined by the machine learning algorithm by minimizing the least squares (Eq.6), where \(c\) is the vector of all coefficients. Key assumptions are that the input variables have a Gaussian distribution, that the input variables are related to the output variable, and that the input variables are not highly correlated with each other (i.e. low collinearity).

Ridge regression builds on linear regression by modifying the loss function to minimize the complexity of the model (Eq.7), where \(\alpha > 0\) is the shrinkage parameter as described below. The complexity of the model is defined as the sum-squared value of the coefficients in the resulting model \(\alpha||c||_2^2\), also called the L2-norm. Therefore, Ridge Regression penalizes large coefficients and the higher the value of \(\alpha\), the more robust the model is to collinearity of the input variables. To reach a solution, Ridge Regression uses the loss function described in Eq.7. Generally, ridge regression favors models that assigns a non-zero coefficient to the majority of the input variables.

\[
\min_c ||cx - y||_2^2 + \alpha||c||_2^2 \tag{Eq.7}
\]

By comparison, LASSO regression tends towards solutions with fewer input variables in the solution. LASSO also uses a modification to the loss function as shown in Eq.8. It minimizes the least-squares penalty with \(\alpha||c||_1\) added, where \(\alpha\) is again the shrinkage parameter and \(||c||_1\) is the L1-norm of the coefficient vector.

\[
\min_c \frac{1}{2N} ||cx - y||_2^2 + \alpha||c||_1 \tag{Eq.8}
\]

The value of \(\alpha\) in both Ridge and LASSO regression is determined using cross-validation, a technique commonly used to fit tuning parameters. Cross-validation takes input data with \(S\) measurements and uses a proportion \((S-1)/S\) of the data for training, while using all data for assessing the performance of the model. For this application the number of input measurements (208) is scarce compared to the number of input variables (666). Therefore, all data points are used \((S=N, \text{ where } N \text{ is the total number of measurements})\). This leave-one-out method means the regression algorithm iteratively fits the data with one of the input variable removed, and uses the results to optimize \(\alpha\).
Elastic Net combines the features of Ridge and LASSO regression. It minimizes the complexity in terms of both the number of non-zero coefficients in the solution and the magnitude of these coefficients (Eq. 9). In doing so it penalizes the model using both the L1-norm and L2-norm. The combination of the L1 and L2 norms is controlled using the so called L1-ratio, also represented as $\rho$. A higher L1-ratio results in the algorithm behaving more closely to LASSO regression, and with a lower L1-ratio it behaves more like Ridge Regression. Cross-validation is again used to set the parameters $\alpha$ and $\rho$.

$$\min_{\mathbf{c}} \frac{1}{2N} \| \mathbf{c} \mathbf{x} - \mathbf{y} \|^2_2 + \alpha \rho \| \mathbf{c} \|_1 + \frac{\alpha(1-\rho)}{2} \| \mathbf{c} \|_2^2$$  \text{ (Eq. 9)}

Elastic Net regression with cross validation was implemented for this project using the ElasticNetCV function from scikit learn (Scikit-Learn 0.17.1). A screen of the L1-ratio was performed by iteratively performing the regression, increasing the resolution of values within a narrower range around the previously set value. This iterative process allowed a high resolution screen to be conducted without exceeding the computational performance of a standard computer. The tuning of the $\alpha$ parameter was done by ElasticNetCV using 100 $\alpha$'s automatically selected for each L1-ratio. The processed and scaled metabolic modelling data was used as the input data, with the normalized IC$_{50}$ acting as the output data. Following implementation of the algorithm, the coefficients for each reaction were extracted in order to assess their contribution to the variation seen in the IC$_{50}$ data.

3.3.4 Machine learning algorithm II: Gradient Boosting regression

Gradient Boosting regression is an ensemble method for machine learning. Ensemble methods combine predictions from base estimators to improve the robustness of the overall model. Boosting methods are a further subset of these methods that build the estimators sequentially to reduce the bias of the combination of the estimators, thereby creating a more accurate ensemble model.

Gradient Boosting is a generalization of the classification boosting algorithm AdaBoost, and both will be introduced here. To achieve regression it involves three elements: (1) a loss function, (2) weak estimators, and (3) an additive model. Boosting is a process that generates stronger models from weak estimators where a weak estimator is defined as a model whose performance is slightly better than random chance. Such an algorithm is appropriate because of the under-determined nature of this machine learning application. The basis of boosting algorithms is to filter out observations that can already be predicted by the weak estimators and then adding additional weak estimators to predict the remaining observations.

AdaBoost uses decision trees with a single split and weights observations by assigning more weight to observations that are difficult to classify with the weak estimator. Then new estimators are added sequentially to train on the more difficult observations. Final predictions are
then made by combining the individual predictions from the decision trees and choosing the most common relationships. Use of AdaBoost is restricted to classification problems.

Gradient Boosting recasts AdaBoost in a statistical framework to allow use for regression. It is redefined as a numerical optimization problem where minimization of a loss function is used as the objective. Estimators are added to the model using gradient descent. This is done in a stage-wise manner so that additional estimators are added without changing those already in the model. Regression trees are used as estimators and are constructed in a greedy manner, with the best split being chosen dynamically based on the specified loss function. The estimators used in gradient boosting generally use 4-8 splits, compared to the single split in AdaBoost. However, the size of the individual estimators remains constrained so that each remains weak on its own. They are added to the model one at a time using a variation on gradient descent that parameterizes the estimator, then modifies the parameters to reduce the residual loss. The estimator is then added to the model in order to improve the ensemble model. Estimators are added sequentially until a specified maximum number is reached or the loss goes below a specified level.

The scikit learn implementation of Gradient Boosting Regression (sklearn.ensemble.GradientBoostingRegressor) was used to fit modelled reaction fluxes to the E. coli IC$_{50}$ values. Least squares regression was used as the loss function, with a maximum of 100 estimators. Default values of other input parameters were used. GradientBoostingRegressor assigns an ‘importance score’ between 0.0 and 1.0 to each feature in the data based on its contribution to the variation in output values. These values were used to assess the relevance of individual reactions with respect to the variation in IC$_{50}$ values.

### 3.3.5 Hypergeometric statistics for subsystem analysis

Following implementation of the machine learning algorithms, the high-level results were statically analyzed using subsystems. Each reaction in a GSMM is assigned to a high-level subsystem of metabolism. These subsystems group reactions based their involvement in major metabolic processes such as central carbon metabolism, lipid biosynthesis and nucleotide salvage. The goal of implementing hypergeometric testing at the subsystem level is to identify which major metabolic pathways are implicated in affecting antibiotic susceptibility. For each reaction assigned a non-zero coefficient or importance score in machine learning, the subsystem was identified from the original iJO1366 model. Hypergeometric statistics were then used to determine subsystems that were enriched in the data. The significance of the number of reactions from each subsystem was computed using the hypergeometric probability density function:

$$y = f(x|M, K, n) = \frac{{K \choose x} (M-K) \choose (n-x)}{{M \choose n}}$$

(Eq.10)

This distribution models the number of successes $x$, in a fixed sample size $n$, drawn without replacement from a finite population $M$, where $K$ is the number of items with the characteristic of interest in the population. For each subsystem $S$ these parameters are defined as follows:
$M$ is the total number of flux-carrying reactions identified by metabolic modelling, after processing.  
$K$ is the number of reactions in $M$ that belong to subsystem $S$.  
$n$ is the number of non-zero coefficients/importance scores identified by machine learning.  
$x$ is the number of reactions in the set $n$ that belong to subsystem $S$.  

Both $M$ and $n$ are fixed for all subsystems, whereas $K$ and $x$ vary between the subsystems. The output $y$ is the p-value of the observed values of $x$. A p-value $< 0.05$ indicates that reactions from a subsystem are significantly over or under represented in the machine learning results compared to random chance. An over-represented subsystem is likely to be important in determining antibiotic susceptibility. This provides a high-level analysis of the machine learning results in addition to the reaction-level results.

### 3.4 Results & Discussion

#### 3.4.1 Data integration results

The metabolite counter screen of *E. coli* with Ampicillin showed variation between the different conditions (Figure 7). Normalized IC$_{50}$ values show a distribution around the control value of 1.0 with a minimum of 0.48, a maximum of 3.2, and a standard deviation of 0.257. The maximum value represents a 3-fold increase in the IC$_{50}$ and the minimum value represents a 2-fold decrease in the IC$_{50}$. Therefore, variation exists in both the metabolic modelling data (Chapter 2) and the antibiotic susceptibility data. Both Elastic Net and Gradient Boosting were successful in integrating these two data sets and predicting which reactions are associated with the variation seen *in vitro*.

Elastic Net returned a solution that included 28% (177/666) of the reactions. Coefficients of the reactions showed a range of 0.58, a maximum value of 0.29 and a minimum of -0.29, and a standard deviation of 0.072 (Figure 8). Gradient Boosting returned a solution with 26% (175/666) reactions having a non-zero importance score. The importance scores ranged from <1.0E-6 to 0.052, with a standard deviation of 0.0085 (Figure 9). Comparison of the two sets of identified reactions reveals that 38% of reactions identified are present in both sets. Elastic Net and Gradient Boosting each have different advantages and limitations with respect to this machine learning problem so the approaches and results can be considered complementary. Subsystem analysis of the reactions revealed that a number of subsystems were significantly over-represented (Table 2). These include nucleotide salvage, the citric acid cycle, and valine/leucine/isoleucine metabolism.
Figure 7. Distribution of IC\textsubscript{50} values from \textit{E. coli} metabolite counter-screen. 208 metabolic conditions are shown, with a bin size of 0.05. All values are normalized as described in section 3.2.2. The IC\textsubscript{50} of the control is equal to 9.56 \textmu g/mL.
Figure 8. Distribution of non-zero coefficients in the Elastic Net Solution for iJO1366. A total of 177 non-zero coefficients are shown with bin size of 0.02. Positive coefficients represent reactions predicted to be associated with increase the IC_{50} of *E. coli*, while negative coefficients represent reactions predicted to be associated with greater susceptibility to ampicillin.
Figure 9. Distribution of non-zero importance scores in the Gradient Boosting solution for iJO1366. A total of 175 non-zero importance scores are shown with bin size of 0.002. A higher importance score reflects that reaction is predicted to have a greater effect on the antibiotic susceptibility of *E. coli* to Ampicillin, positive or negative.
The limitations of each algorithm as they relate to these results will be briefly discussed. As previously described, Elastic Net, by virtue of being a linear regression technique relies on a series of assumptions. While Elastic Net improves on simple linear regression by reducing the effect of deviation from these assumptions it is important to note the degree to which the assumptions are met in this study. First, analysis of the individual correlations between each reaction and the IC\textsubscript{50} data confirm that the relationships are best modelled as linear due to a lack of non-linear (exponential, logarithmic etc.) patterns. Therefore, the assumption of linear relationships is valid. Second, the assumption that the input variables are normally distributed is not strongly upheld by the metabolic modelling data. In general, for essential reactions the variation across conditions around the mean can be approximated as Gaussian. However, for the reactions that are not used in every simulation, a significant number of conditions give a zero value, with only non-zero values showing variation about a mean. These reactions do not show a Gaussian distribution, and this is a limitation for application of Elastic Net. Third, it is expected that many of the reactions do not have a direct effect on antibiotic susceptibility and thus not all inputs are related to the output. However, Elastic Net captures this characteristic by having a penalty associated with the number of parameters in the solution, so this does not act as a substantial limitation. The fact that the number of reaction included in the solution is <30% shows that the algorithm is able to exclude reactions that do not have a relationship to antibiotic susceptibility. Fourth, it is expected that collinearity exists within the metabolic modelling data, further limiting the applicability of a linear model. Because the data is generated from a GSMM, many reactions are not independent of each other by definition. While providing some control for collinearity, Elastic Net cannot avoid the limitation presented by a lack of collinearity. Despite these limitations, use of Elastic Net is informative as it provides quantitative values for the direct relationship between each reaction and the antibiotic susceptibility. When combined with the sign of the values, this allows for generation of hypotheses that include the magnitude and directionality of the relationship.

Gradient Boosting has some similar limitations in this study. First, it also assumes that the input variables are independent of each other. In addition, the importance scores extracted do not provide information on the sign of the coefficients and thus the direction of the relationships between reactions and the IC\textsubscript{50}, which limits the hypotheses that can be generated from the method. Aside from these limitations, Gradient Boosting is expected to better handle the sparsity of the reaction data. This is because it does not need the input variables to be normally distributed. In addition, because it is designed to work with weak estimators, it is more appropriate for use with underdetermined problems. This analysis is underdetermined because the number of modelled conditions is considerably less that the number of reactions.

Overall, the purpose of integrating the modelled metabolic flux rates and the \textit{in vitro} antibiotic susceptibility measurements is to generate hypotheses for further investigation. Therefore, the predictions from machine learning are informative, despite the limitations in application of the techniques to this problem. Furthermore, the results of both algorithms can be
used in combination for generation of these hypotheses. First, the two methods have a different foundation and have different limitations, thus it is likely that some biologically important results will be picked up by one method and not the other based on the structure of the data. Second, if a reaction is identified by both algorithms, there is a strong likelihood that it has a significant contribution to the output. Finally, Elastic Net maintains directionality so the results can be used to inform the direction of relationships identified by Gradient Boosting.

3.4.2 Biological insights from data integration

The results of the machine learning suggest a number of subsystems that could be responsible for the observed variation in antibiotic susceptibility (Table 2). From hypergeometric testing using $M = 666$, the most significant over-represented pathways that are consistent across both methods are Valine/Leucine/Isoleucine metabolism, Arginine/Proline metabolism, Glycine/Serine metabolism, and nucleotide salvage. In contrast, glycerophospholipid metabolism, cell envelope biosynthesis, and cofactor/prosthetic group biosynthesis were revealed as significantly under-represented. In order to gain insight into the new hypotheses generated, I investigated these pathways in relation to what is already known about Ampicillin treatment in *E. coli*.

It has been shown that Leucine and Serine can have a significant effect on the resistance of *E. coli* to beta-lactams such as Ampicillin. In a study by Boulac et al. in 1992, it was shown that presence of L-leucine or L-serine in the media conferred resistance to beta-lactam action in otherwise antibiotic-sensitive *E. coli* cells. Furthermore, it was found that the effect could be reversed by introduction of isoleucine and valine. Integration of the metabolic modelling data with the measured responses to ampicillin identifies reactions such as the leucine transaminase reactions as being associated with higher IC$_{50}$ values. Therefore, these results are consistent with what is known about *E. coli*'s response to L-leucine and encourage further investigation of individual reactions within the leucine and serine metabolic subsystem, as well as other related amino acids.

Furthermore, nucleotide salvage pathways have been associated with changes in antibiotic susceptibility in *E. coli*. In one study it was shown that treatment with antibiotics resulted in increased expression of genes associated with both nucleotide salvage and purine/pyrimidine biosynthesis (Shaw et al., 2003). Another study similarly identified UMP biosynthesis as contributing to *E. coli* survival in the presence of ampicillin, and more specifically reactions mediated by the *pyrB* gene (Garavaglia et al., 2012). In the iJO1366 model, the *pyrB* gene is associated with the aspartate carbamoyltransferase (ASPCT) reaction from the purine/pyrimidine biosynthesis subsystem. ASPCT was identified by both Elastic Net and Gradient Boosting in this project. Furthermore, the UMP kinase (UMPK) reaction from the nucleotide salvage pathway was among the reactions identified by Gradient Boosting. Therefore, the methods used in this project were again able to identify biologically relevant reactions.
Table 2. P-values for *E. coli* subsystems following hypergeometric testing of results from Elastic Net and Gradient Boosting. Bold indicates p-values < 0.05. Values with * indicate that the result corresponds to under representation of reactions from the subsystem in the machine learning solutions. The table is sorted by the p-values form elastic net, with over represented subsystems first.

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Elastic Net p-value</th>
<th>Gradient Boosting p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nucleotide Salvage Pathway</td>
<td>3.5e-5</td>
<td>0.04</td>
</tr>
<tr>
<td>Histidine Metabolism</td>
<td>4.2e-5</td>
<td>0.27</td>
</tr>
<tr>
<td>Threonine and Lysine Metabolism</td>
<td>4.5e-4</td>
<td>0.23</td>
</tr>
<tr>
<td>Glycine and Serine Metabolism</td>
<td>5.1e-4</td>
<td>0.01</td>
</tr>
<tr>
<td>Arginine and Proline Metabolism</td>
<td>2.4e-3</td>
<td>*5.7e-4</td>
</tr>
<tr>
<td>Valine, Leucine, and Isoleucine Metabolism</td>
<td>0.008</td>
<td>3.3e-4</td>
</tr>
<tr>
<td>Citric Acid Cycle</td>
<td>0.07</td>
<td>0.07</td>
</tr>
<tr>
<td>Cysteine Metabolism</td>
<td>0.07</td>
<td>0.17</td>
</tr>
<tr>
<td>Methionine Metabolism</td>
<td>0.07</td>
<td>0.26</td>
</tr>
<tr>
<td>Tyrosine, Tryptophan, and Phenylalanine Metabolism</td>
<td>0.09</td>
<td>0.09</td>
</tr>
<tr>
<td>Other</td>
<td>0.10</td>
<td>0.28</td>
</tr>
<tr>
<td>Transport, Inner Membrane</td>
<td>0.13</td>
<td>*4.2e-4</td>
</tr>
<tr>
<td>Oxidative Phosphorylation</td>
<td>0.13</td>
<td>0.13</td>
</tr>
<tr>
<td>Exchange</td>
<td>0.15</td>
<td>0.09</td>
</tr>
<tr>
<td>Membrane Lipid Metabolism</td>
<td>0.15</td>
<td>0.17</td>
</tr>
<tr>
<td>Glutamate Metabolism</td>
<td>0.16</td>
<td>0.43</td>
</tr>
<tr>
<td>Purine and Pyrimidine Biosynthesis</td>
<td>0.17</td>
<td>*1.0e-5</td>
</tr>
<tr>
<td>Pentose Phosphate Pathway</td>
<td>0.19</td>
<td>0.19</td>
</tr>
<tr>
<td>Transport, Outer Membrane</td>
<td>0.27</td>
<td>0.26</td>
</tr>
<tr>
<td>Anaplerotic Reactions</td>
<td>0.28</td>
<td>0.10</td>
</tr>
<tr>
<td>Alanine and Aspartate Metabolism</td>
<td>0.31</td>
<td>0.31</td>
</tr>
<tr>
<td>Folate Metabolism</td>
<td>0.34</td>
<td>0.15</td>
</tr>
<tr>
<td>Cofactor and Prosthetic Group Biosynthesis</td>
<td>*1.2e-14</td>
<td>*1.2e-4</td>
</tr>
<tr>
<td>Cell Envelope Biosynthesis</td>
<td>*0.02</td>
<td>*0.05</td>
</tr>
<tr>
<td>Glycolysis/Gluconeogenesis</td>
<td>*0.02</td>
<td>0.001</td>
</tr>
<tr>
<td>Lipopolysaccharide Biosynthesis / Recycling</td>
<td>*0.02</td>
<td>*0.11</td>
</tr>
<tr>
<td>Glycerophospholipid Metabolism</td>
<td>*0.04</td>
<td>*0.05</td>
</tr>
<tr>
<td>Alternate Carbon Metabolism</td>
<td>*0.05</td>
<td>*4.7e-6</td>
</tr>
<tr>
<td>Inorganic Ion Transport and Metabolism</td>
<td>*0.17</td>
<td>*0.12</td>
</tr>
<tr>
<td>Transport, Outer Membrane Porin</td>
<td>*0.18</td>
<td>*0.15</td>
</tr>
<tr>
<td>Pyruvate Metabolism</td>
<td>*0.29</td>
<td>*0.30</td>
</tr>
<tr>
<td>Nitrogen Metabolism</td>
<td>*0.40</td>
<td>*0.40</td>
</tr>
<tr>
<td>Murein Biosynthesis</td>
<td>*0.54</td>
<td>0.39</td>
</tr>
<tr>
<td>Glyoxylate Metabolism</td>
<td>*0.73</td>
<td>*0.74</td>
</tr>
<tr>
<td>Methylglyoxal Metabolism</td>
<td>*1.00</td>
<td>*1.00</td>
</tr>
<tr>
<td>Murein Recycling</td>
<td>*1.00</td>
<td>*1.00</td>
</tr>
<tr>
<td>tRNA Charging</td>
<td>*1.00</td>
<td>*1.00</td>
</tr>
</tbody>
</table>
Conspicuously, the method did not significantly identify reactions associated with cell wall synthesis, the target of Ampicillin. In fact, this subsystem was significantly under-represented. However, this is not unexpected based on both the modelling and experimental conditions. The metabolic modelling does not attempt to model the action of the Ampicillin, and the metabolites screened did not contain direct precursors of cell wall synthesis. Therefore, it is expected that the modelling does not produce a high degree of variation among these pathways compared to subsystems such as central carbon metabolism where metabolites are directly supplied in the screen. Because machine learning techniques use variation in the input variables to determine those that are part of the solution, these important metabolites were not emphasized. While cell wall synthesis reactions are required for growth, the modelling does not directly perturb them so they are not identified by the techniques used. However, the direct effects of an antibiotic are not the sole contributors to cell death. As discussed, there are various downstream effects and this method is powerful for identifying these less obvious factors, and the results suggest that there may be effects on antibiotic susceptibility from a greater range of amino acids than has been described so far.

3.5 Conclusions

Using both Elastic Net and Gradient Boosting I was able to successfully integrate the metabolic modelling results with the experimental data of *E. coli* susceptibility to Ampicillin. Both datasets showed variation based on the presence of different metabolites in the media, so machine learning was able to make predictions about which reactions might be responsible for the changes in antibiotic susceptibility. Both algorithms predicted that approximately 30% of reactions in the metabolic modelling data contribute to the antibiotic susceptibility. As discussed, the application of machine learning to this problem does have limitations. These limitations vary between algorithms so the two methods do not select the same overall set of reactions; however, when using them to generate hypotheses, these machine learning algorithms can be treated as complementary approaches. Furthermore, despite the limitations these methods were able to identify reactions with known biological importance. This gives confidence for investigating other relationships identified and helps validate the use of this overall approach as an unsupervised method for hypothesis generation. Overall, I was able to successfully develop a strategy for predicting underlying biological mechanisms using machine learning by piloting the methods in *E. coli*. With the entire pipeline constructed, I was then able to proceed to my major goal of investigating *M. tuberculosis* metabolism.
Chapter 4 – Application to 
*M. tuberculosis*

4.1 Introduction

*M. tuberculosis* is difficult to treat and emergence of drug resistant strains is increasing. Therefore, there is a need to increase our understanding of how currently available antibiotics interact with cellular metabolism. Furthermore, there is a need to investigate new ways of causing or promoting cell death to overcome limitations of current treatments. Traditional experimental methods for investigating cellular metabolism such as transcriptomics fail when applied to *M. tuberculosis* due to the difficulty of performing experiments, meaning that alternative methods such as the analysis pipeline presented here are critical to solving the challenge posed by *M. tuberculosis*. *E. coli* was used for the development of the simulation and data integration steps because its metabolic network is very well understood. However, the overall goal of this approach is application to clinically significant pathogens, such as *M. tuberculosis*. In doing so, our understanding of this important global pathogen will be increased, allowing for enhanced engineering of novel strategies for treatment.

For implementation of the pipeline described in Chapters 2 & 3, a genome-scale metabolic model (GSMM) of *M. tuberculosis* was selected for metabolic modelling, and a pilot counter-screen using 35 metabolites and four different clinically relevant antibiotics was performed to generate the antibiotic susceptibility data. For this pilot screen, *Mycobacterium bovis BCG* cells were used in place of *M. tuberculosis* cells to streamline the screen. These species have a very high degree of genetic similarity and *M. bovis BCG* is accepted in scientific literature as a substitute for *M. tuberculosis* (Gordon et al., 1999). From the results I was able to produce significant biological insights for *M. tuberculosis*. 
4.2 Metabolic modelling of *M. tuberculosis*

4.2.1 Selection of the iSM810 model of *M. tuberculosis*

To start the modelling process, a genome-scale metabolic model (GSMM) had to be selected. There are a number of similarly comprehensive GSMMs for *M. tuberculosis*, and models from three authors were considered: Ma et al., 2015 (iSM810), Garay et al., 2015 (Garay), and Rienksma et al., 2014 (sMtb). An overview of the model contents is provided in Table 3.

The sMtb model is the most comprehensive model in terms of the number of metabolites and reactions; however, it lacks accompanying supplementary information describing the details of the model. The iSM810 and Garay models contain a similar amount of information about the metabolic network of *M. tuberculosis*. However, the iSM810 model has been explicitly demonstrated to accurately predict changes in growth in response to different metabolites in the media, while the Garay model has not. In addition, the supplementary information for the iSM810 model is complete. Therefore, iSM810 model was chosen as the GSMM for *M. tuberculosis*.

Table 3. **Summary statistics on *M. tuberculosis* models.** Total number of unique metabolites, reactions, genes and exchange reactions are shown.

<table>
<thead>
<tr>
<th>Author</th>
<th>Ma et al., 2015</th>
<th>Garay et al., 2015</th>
<th>Rienksma et al., 2014</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model Name</td>
<td>iSM810</td>
<td>N/A</td>
<td>sMtb</td>
</tr>
<tr>
<td>Metabolites</td>
<td>724</td>
<td>754</td>
<td>1047</td>
</tr>
<tr>
<td>Reactions</td>
<td>938</td>
<td>965</td>
<td>1192</td>
</tr>
<tr>
<td>Genes</td>
<td>810</td>
<td>760</td>
<td>915</td>
</tr>
<tr>
<td>Exchange Reactions</td>
<td>126</td>
<td>124</td>
<td>121</td>
</tr>
</tbody>
</table>
4.2.2 Initial conditions for metabolic modelling

For the in vitro screen, Middlebrook 7H9 media with OADC supplementation was used. Therefore, the initial conditions of the iSM810 model were set to only allow uptake of the metabolites present in this media, and oxygen. As with E. coli, metabolites that are not rate-limiting were given an upper bound of 1000 mmol/h and a lower bound of -1000 mmol/h. Rate-limiting metabolites were given lower bounds that reflect physiological limits in mmol/h. The definition of exchange reactions in iSM810 differs from the E. coli iJO1366 model: they are defined as “↔ metabolite[c]” where metabolite[c] refers to cytoplasmic metabolite. None of the M. tuberculosis models to date include transport through the periplasm, so exchange reactions are defined as directly into the cytoplasm. This is in contrast to E. coli models that define exchange reactions as into the extracellular space, with corresponding reactions describing transport through the periplasm into the cytoplasm.

For each metabolite condition simulated, the relevant exchange reaction’s upper and lower bounds were set to 10 and 0 mmol/h respectively. Again these bounds were set to reflect experimental concentrations. However, not all metabolites had inbuilt exchange reactions. In order to simulate the greatest range of conditions I decided to create additional exchange reactions in an attempt to simulate the perturbations these metabolites might produce. This was possible for iSM810 because it was not necessary to also create periplasmic transport reactions which often have H or ATP requirements. Furthermore, addition of new exchange reactions did not result in unwarranted disruption to the overall network in contrast to iJO1366, where attempting to add new exchange reactions to iJO1366 caused significant disruption, and in some cases changed the growth rate by an order of magnitude.

A total of 333 metabolite conditions were simulated, as well as the control condition. This set represents the intersection of the 724 metabolites present in the model and the metabolites that were able to be sourced commercially for completion of the metabolite screen.

4.2.3 Implementation of pFBA with iSM810

Metabolite-specific models were produced for each of the 333 metabolites being investigated using Parsimonious Flux Balance Analysis (pFBA). Aside from the change in directionality of the exchange reactions, this was implemented by applying the method constructed using iJO1366. Again, the objective function used was the inbuilt biomass production reaction in iSM810. Following creation of the individual models, optGpSampler was used as described in Chapter 2 to retrieve the most likely flux state for each reaction after flux-balance analysis (FBA). The total numbers of flux-carrying reactions in the solutions were used to assess the variation in network utilization across the simulated conditions (Figure 10).
Figure 10. Network utilization of iSM810 simulations by subsystem. Values are displayed as a percentage of the total number of reactions present in a subsystem. *Essential* reactions are reactions that carry a flux in all simulations. *Accessible* reactions are flux-carrying in ≥ 1 simulations. *Inaccessible* reactions are never predicted to carry a flux in the simulated conditions. Results from all 333 simulations are included.
Overall, the metabolic modelling of 46% (333/724) of the metabolites present in iSM810 resulted in 72% (854/1186) of the total reactions being predicted to be active in response to at least 1 metabolite condition. The total reactions include the 248 exchange reactions added to the model. Excluding these reactions we see 66% (623/938) reactions with variation. The remaining reactions were never predicted to carry flux so were determined to be inaccessible by metabolic modelling. Of the active reactions, 28% (175/623) carried flux in all simulated conditions. These reactions were essential as they represent essential processes for growth and are always part of the optimal solution. The remaining 72% (491/623) of active reactions showed variability between the conditions in whether they were predicted to carry a flux and so were accessible by metabolic modelling.

4.2.4 Summary of metabolic modelling results

Utilizing the methods I developed, I was able to successfully model 333 unique metabolite conditions using the iSM810 model. As with E. coli, the results clearly show that this modelling produces variation in the reactions used to achieve optimal biomass production in response to the presence of different metabolites. With 46% of the metabolites present in iSM810, I was able to access 66% of the reactions in the original model. Analysis of the subsystems of these reactions showed that major metabolic processes such as cell wall synthesis, central carbon metabolism, nucleotide biosynthesis and amino acid metabolism had high levels of accessibility. From the results of this metabolic modelling I was able to identify a set of 35 metabolites for a pilot metabolic screen in M. bovis BCG. The pilot set of metabolites was chosen to be commercially inexpensive and to perturb diverse parts of the metabolic network. M. bovis BCG is commonly used as a substitute for experimentation because M. tuberculosis is very difficult to work with and all experiments with it must be performed at Biosafety Level 3. By comparison, M. bovis BCG is Biosafety Level 2 and much easier to perform experiments with. It is very metabolically similar and has an equivalently sized genome, thus it closely approximates M. tuberculosis.

4.3 Data integration

4.3.1 Collection of antibiotic susceptibility data for M. tuberculosis

As with E. coli, a metabolite counter screen was performed in order to collect antibiotic susceptibility data. For the pilot set, M. bovis BCG cells were used in place of M. tuberculosis cells. Four different antibiotics clinically used for the treatment of M. tuberculosis infection were screened: Isoniazid (INH), Ethambutol (ETH), Ciprofloxacin (CIP) and Rifampicin (RIF). These drugs target mycolic acid synthesis, cell wall synthesis, DNA replication, and RNA synthesis respectively. The 35 metabolites identified from metabolic modelling were used.
An *M. bovis* BCG strain that constitutively expresses GFP2 was cultured in Middlebrook 7H9 medium with 0.05%(v/v) Tyloxapol and 10%(v/v) OADC. For screening, cultures were grown to stationary phase then back-diluted in media to a target optical density (OD600) of 0.05. Metabolites and antibiotics were prepared to a concentration of 10 mM in either water or DMSO depending on solubility. Antibiotics were diluted in either water (INH, ETH, CIP) or DMSO (RIF) depending on solubility limit. Aliquots of one metabolite per well were loaded onto 384-well assay plates in each column, followed by serial drug dilutions and then cultures of *M. bovis* BCG. Controls included no treatment, one well per metabolite without drug, and a column of drug gradient without an added metabolite. Plates were incubated for 7 days, following which the fluorescence in each well was measured. The concentration at which 90% growth inhibition occurred (IC$_{90}$) was the target output measurement. Statistical analysis in Prism 7.0a of the growth across the serial dilutions was used to determine the IC$_{90}$ for each metabolite condition. Outliers due to technical issues (fluorescence of the metabolite itself, for example) were excluded in the final results.

The metabolite counter screen of *M. bovis* BCG with INH, ETH, CIP and RIF showed variation in growth when supplied with the different metabolites (Figure. 11). Comparison of the results across the four antibiotic treatments reveals that INH, CIP and RIF produce similar responses, but ETH shows a different pattern. This suggests that the role of metabolism in contributing to antibiotic efficacy is specific to the antibiotic used. Normalized IC$_{90}$ values (fold changes) show the following distributions around the control value of 1.0: INH) max = 2.3, min = 0.88, σ = 0.29; ETH) max = 4.2, min = 0.50, σ = 0.9; CIP) max = 1.8, min = 0.32, σ = 0.35; RIF) max = 1.5, min = 0.39, σ = 0.23. Overall, variation existed in both the antibiotic susceptibility data and the metabolic modelling data, so I could proceed with data integration.
Figure 11. Differences in antibiotic susceptibility following metabolite counter screening in *M. bovis* BCG. Values are the log2 transform of the fold change in IC50. Higher values represent an increased IC50 and therefore a decrease in antibiotic susceptibility. Results are shown for Ethambutol (ETH), Rifampicin (RIF), Isoniazid (INH), and Ciprofloxacin (CIP). Two control conditions are present: 1) the control wells that contained no cells, and 2) the wells containing 1% DMSO used to control for antibiotics and metabolites solubilized in DMSO.
4.3.2 Data integration results for *M. tuberculosis*

The results of the metabolite counter screen were combined with the metabolic modelling results using the Elastic Net and Gradient Boosting regression methods, as described in Chapter 3. First, the metabolic modelling results were preprocessed to remove the newly added exchange reactions and to convert the reactions back to their reversible forms. Data was also scaled using the MaxAbsScaler function from scikit-learn. Second, the IC\textsubscript{90} data was normalized to the IC\textsubscript{90} of the control condition. For each of the four antibiotic screens, the data was normalized based on the corresponding control conditions. Third, the preprocessed data sets were integrated using the two machine learning algorithms. The metabolic modelling data again acted as the input data and the IC\textsubscript{90} data was used as the output data. Coefficients and importance scores of the individual reactions were extracted from Elastic Net and Gradient Boosting respectively, and used to assess their contribution to the variation seen in the antibiotic susceptibility.

Because this task is very underdetermined with 35 conditions and 623 reactions, Elastic Net was not able to be meaningfully implemented. For each of the antibiotics the algorithm was only able to identify a trivial number of reactions. However, gradient boosting performed successfully. Its implementation gave a solutions that included 18.9\%, 19.1\%, 23.6\%, and 18.6\% of the 623 total reactions for INH, ETH, CIP, and RIF respectively. This is less than the 26\% identified for *E. coli* and the importance scores assigned (Figure 12) by the algorithm are significantly lower than those assigned for *E. coli* (Figure 9). However, it was still possible to extract information about the subsystems represented by reactions with non-zero importance scores. The 22 subsystems $S$ were analyzed using hypergeometric statistics with $M = 623$ for iSM810. Results from this testing are displayed in Table 4. This statistical approach is more limited for application to *M. tuberculosis* because the subsystems are more coarsely defined and the data is much more underdetermined compared to *E. coli*.

4.3.3 Biological insights from data integration

From the results of data integration, a number of hypotheses can be generated regarding the contribution to antibiotic susceptibility of both individual reactions and larger pathways. Hypergeometric analysis of the subsystems suggests that polyprenoids biosynthesis, central carbon metabolism, and amino acid metabolism are over-represented in the solutions and so are important in determining antibiotic susceptibility (Table 4). Transport processes and lipid biosynthesis are revealed as significantly under-represented implying that they are not important. To investigate the results further, I looked at the reactions identified and found that 16 reactions were identified in all four cases and over 100 were identified in at least 2 cases. Many of these reactions have significant biological importance.
Figure 12. Distribution of non-zero importance scores from Gradient Boosting of iSM810 results. All importance scores are log-transformed and grouped with a bin size of 0.5 following transformation. The total number of non-zero importances are 126 (INH), 127 (ETH), 157 (RIF) and 124 (CIP). A higher importance score reflects that reaction is predicted to have a greater effect on the susceptibility of *M. tuberculosis* to the respective antibiotics, positive or negative.
Table 4. P-values for *M. tuberculosis* subsystems following hypergeometric testing of results from Gradient Boosting. Bold indicates p-values <0.05. Values with * correspond to under representation of reactions from the subsystem in the machine learning solution. Results for Isoniazid (INH), Ethambutol (ETH), Rifampicin (RIF), and Ciprofloxacin (CIP) are shown.

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>INH</th>
<th>ETH</th>
<th>RIF</th>
<th>CIP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Polyprenoids biosynthesis</td>
<td>0.003</td>
<td>0.04</td>
<td>0.12</td>
<td>0.01</td>
</tr>
<tr>
<td>Central carbon metabolism</td>
<td>0.003</td>
<td>0.10</td>
<td>1.6E-4</td>
<td>0.01</td>
</tr>
<tr>
<td>Amino acid metabolism</td>
<td>0.013</td>
<td>0.07</td>
<td>0.07</td>
<td>0.10</td>
</tr>
<tr>
<td>Cholesterol degradation</td>
<td>0.05</td>
<td>*0.001</td>
<td>*0.001</td>
<td>0.06</td>
</tr>
<tr>
<td>Nucleotide biosynthesis</td>
<td>0.05</td>
<td>0.12</td>
<td>0.04</td>
<td>0.08</td>
</tr>
<tr>
<td>Cofactor biosynthesis</td>
<td>0.09</td>
<td>0.09</td>
<td>0.06</td>
<td>0.10</td>
</tr>
<tr>
<td>Folate Metabolism</td>
<td>0.10</td>
<td>*0.42</td>
<td>0.42</td>
<td>0.36</td>
</tr>
<tr>
<td>Sulfate metabolism</td>
<td>0.16</td>
<td>*0.32</td>
<td>0.20</td>
<td>0.13</td>
</tr>
<tr>
<td>Energy metabolism</td>
<td>0.19</td>
<td>0.30</td>
<td>0.03</td>
<td>0.002</td>
</tr>
<tr>
<td>Other</td>
<td>0.25</td>
<td>0.05</td>
<td>0.20</td>
<td>0.23</td>
</tr>
<tr>
<td>Transport reaction</td>
<td>*0.004</td>
<td>*0.004</td>
<td>*0.001</td>
<td>*0.02</td>
</tr>
<tr>
<td>Lipid biosynthesis</td>
<td>*0.03</td>
<td>*0.03</td>
<td>*0.04</td>
<td>*0.002</td>
</tr>
<tr>
<td>beta oxidation of fatty acids</td>
<td>*0.07</td>
<td>*0.04</td>
<td>*0.04</td>
<td>*0.10</td>
</tr>
<tr>
<td>Cell wall synthesis</td>
<td>*0.08</td>
<td>0.11</td>
<td>*0.004</td>
<td>*0.17</td>
</tr>
<tr>
<td>Alternate carbon metabolism</td>
<td>*0.23</td>
<td>*0.22</td>
<td>*0.19</td>
<td>*0.21</td>
</tr>
<tr>
<td>Pyruvate metabolism</td>
<td>*0.49</td>
<td>*0.42</td>
<td>0.42</td>
<td>*0.54</td>
</tr>
<tr>
<td>Glycogen metabolism</td>
<td>*0.49</td>
<td>*0.42</td>
<td>*0.42</td>
<td>*0.54</td>
</tr>
<tr>
<td>Fatty acid metabolism</td>
<td>*0.62</td>
<td>*0.56</td>
<td>*0.56</td>
<td>*0.66</td>
</tr>
<tr>
<td>Peptidoglycan metabolism</td>
<td>*0.62</td>
<td>*0.56</td>
<td>*0.56</td>
<td>*0.66</td>
</tr>
<tr>
<td>Mycobactin synthesis</td>
<td>*0.79</td>
<td>*0.75</td>
<td>0.24</td>
<td>*0.81</td>
</tr>
<tr>
<td>Membrane metabolism</td>
<td>*0.79</td>
<td>*0.75</td>
<td>*0.75</td>
<td>*0.81</td>
</tr>
<tr>
<td>Nitrogen metabolism</td>
<td>*0.79</td>
<td>*0.75</td>
<td>*0.75</td>
<td>*0.81</td>
</tr>
</tbody>
</table>
The most interesting result from these 16 reactions is that three of them are reactions in the TCA cycle. This is also consistent with the results of the hypergeometric testing which reveals central carbon metabolism as a subsystem over-represented in all four antibiotic conditions. Two of the TCA cycle reactions are associated the enzyme aconitase which catalyzes the stereo-specific isomerization of citrate to isocitrate via cis-aconitate and has been linked to antibiotic efficacy (Nandakumar et al., 2013), and the third is associated with fumarate hydratase which catalyzes the reversible hydration of fumarate to malate. The TCA cycle is a core part of metabolism, and variation in these reactions is expected with variation in growth rate. However, simulated variation of these reactions is much greater than the simulated variation in biomass production. Furthermore, upregulation of the TCA cycle has been demonstrated to increase antibiotic susceptibility of bacteria, especially with regard to tolerant cells. It has been shown that tolerant Pseudomonas aeruginosa cells grown with fumarate, a TCA cycle metabolite, are able to be killed with greater efficacy (Meylan et al., 2017). Upregulation of the TCA cycle leads to an increase in respiration which results in the tolerant cells showing an increase in uptake of antibiotic and an increase in reactive oxygen species (ROS) which lead to cell damage (Dwyer et al., 2014). The results of these studies suggest that metabolites, such as fumarate, could lead to an equivalent effect in M. tuberculosis.

Along the same theme, ATP synthase was also identified by machine learning. Comparing modelled fluxes for ATP synthase and biomass growth rates reveals that the metabolic simulations separate into two groups (Figure 13). The expected result is that increases in metabolic activity, represented by ATP synthase activity, will be accompanied by an increase in growth. However, these results indicate a set of metabolites that increase metabolic activity with a minimal increase in growth rate. These are of great interest for further investigation as they represent possible methods of reactivating persistent M. tuberculosis without stimulating growth of the pathogen. Consistent with results regarding the TCA cycle, fumarate is the metabolite with the highest ATP synthase activity in this second group.

A third result from the 16 top reactions has also been shown to have biological relevance. The reaction catalyzed by cystathionine γ-synthase, an enzyme involved in cysteine metabolism and sulfur assimilation, has been implicated in increasing M. tuberculosis sensitivity to isoniazid (INH). In experiments growing M. tuberculosis in the presence of cysteine, the cells show much greater cell death (Bhave et al., 2007; Vilchèze et al., 2017). The mechanism of this effect has again been linked to an increase in respiration and the resulting damage from ROS which is known to contribute to cell death (Lobritz et al., 2015; Dwyer et al., 2014; Kohanski et al., 2007). The fact that this reaction was identified for all antibiotics suggests that this effect may not be restricted to isoniazid action.
Figure 13. Comparison of simulated reaction fluxes of ATP Synthesis and Biomass Growth Rate. Values are from metabolic modelling of \textit{M. tuberculosis} using the iSM810 model. All 333 modelled metabolic conditions are shown. Points in the red group correspond to models that show increased metabolic activity with a minimal increase in growth rate.
Other subsystems revealed via hypergeometric testing include polyprenoids biosynthesis which is critical for cell wall formation. As two of the tested antibiotics inhibit cell wall synthesis, this is not surprising. While polyprenoids synthesis pathways and corresponding genes have been described for *M. tuberculosis*, little has been done to investigate the effect on antibiotic efficacy of perturbing these reactions. Overall, these results show that the method applied here is able to identify known factors that contribute to antibiotic susceptibility. Therefore, hypotheses regarding other reactions and pathways that arise based on this study, such as polyprenoids biosynthesis, warrant further investigation.

4.4 Conclusions

Integrating the metabolite-specific models generated from the iSM810 model of *M. tuberculosis* with the experimental measurements returned several significant biological insights, demonstrating that the strategies for metabolic modelling and data integration developed are generalizable to different organisms. Creation of 333 metabolite-specific models from the iSM810 model of *M. tuberculosis* metabolism gave very good coverage of the network. With 46% of the metabolites in the model simulated, variation in 71% of the reactions in the model was produced. Therefore, the success of the simulation strategy is not restricted to *E. coli*.

Data integration was similarly successful. The small pilot IC₉₀ data set likely resulted in the lower percent of reactions being identified – 19% for *M. tuberculosis* versus 26% for *E. coli*. Despite the fact that the hypergeometric testing did not identify many subsystems as over-represented, analysis of the reactions consistently present in the solutions indicated important processes in the metabolic network. The activity of the TCA cycle has been linked to antibiotic sensitivity in other organisms and these results strongly suggest that it plays a similar role in *M. tuberculosis*. This is a significant finding as it represents a potential target for addressing the major challenge of tolerant *M. tuberculosis* cells in patients. Furthermore, the results suggest that there may be a range of ways to increase metabolic activity of the cells without stimulating growth. This is important as activation of respiration leads to an increase in reactive oxygen species (ROS) that cause cell damage. Other reactions identified by machine learning as contributing to changes in antibiotic sensitivity included tyrosine and phenylalanine metabolism, as well as synthesis pathways for polyprenoids and nucleotides. Investigation into these reactions could lead to new biological understanding of the metabolic pathways associated with antibiotic killing in *M. tuberculosis*.

Retention of both regression algorithms proved important in generating predictions from the small pilot set of experimental results. Data from the full metabolite screen, with 10x the number of metabolites, is likely to allow Elastic Net to reveal meaningful relationships between the data sets. In addition, the larger data set will also make the results of Gradient Boosting more robust. Then these two sets of predictions can be analyzed in a complementary manner, as for *E. coli*.
coli in Chapter 3. Furthermore, a larger data set is expected to increase the number of reactions identified as contributing to the IC₉₀, thus making the use of hypergeometric statistics more powerful. Finally, the full screen will be performed with *M. tuberculosis* cells rather than *M. bovis BCG* which will eliminate any species-specific effects. Overall, the implementation of this strategy with the full screen will likely return even stronger biological insights.
Chapter 5 – Conclusions and Future Directions

In this thesis I have outlined the development of a widely applicable pipeline to ease integration of metabolic modelling data with experimental measurements of cell growth. Growth in the presence of different metabolites can be simulated for any organism that has a genome-scale metabolic (GSMM) reconstruction. Here I have simulated both the iJO1366 model of *E. coli* and the iSM810 model of *M. tuberculosis*. From these simulations I was able to successfully estimate the metabolic state of these organisms in response to a wide range of metabolites. I was then able to combine the modelling results with measured antibiotic susceptibilities from experimental metabolite counter screens. Following completion of both steps of the computational pipeline I was able to make predictions about biological mechanisms that affect antibiotic efficacy.

Of particular note were the results for *M. tuberculosis*, where a number of important metabolic processes were successfully identified. Reactions of the TCA cycle were identified from all four antibiotic conditions, which aligns with recent studies regarding the downstream effects of antibiotics in a variety of species. In addition, cysteine metabolism was implicated which has also been shown to potentiate antibiotic killing. The consistency of these results with previously documented experiments encourages potential roles for other reactions revealed by my approach, such as tyrosine and phenylalanine metabolism. Overall, this approach can be used as a powerful tool to infer relationships between cellular metabolism and antibiotic action. This means it could be very useful for improving our understanding of how antibiotics lead to cell death, and how metabolic and environmental conditions affect antibiotic efficacy. In addition, it could be used to identify additional targets and methods for drug discovery. All of these applications are critical for combatting the increasing danger posed by microbial pathogens.

This approach could be expanded in the future to include simulations of knock-out experiments (KOs). In order to validate the hypotheses generated it is necessary to perform follow up experiments, many of which would take the form of KOs. It would be useful to simulate these experiments first before making a decision to proceed. This is especially true for species such as *M. tuberculosis* where knock out experiments require a significant investment of resources. Because of the gene-reaction associations in GSMMs, it is possible to simulate KOs by implying a zero flux condition on the corresponding reaction. Methods exist within the CobraToolbox for this, so addition of this step to the pipeline would not significantly increase the complexity of the pipeline. In addition to the results produced from metabolite counter-screens, this approach could be generalized to simulate KO experiments with the goal of understanding downstream metabolic responses to a given knock out.
References


64