Computational approaches in infectious disease research: towards improved diagnostic methods

a dissertation presented
by
Defne Surujon
to
The Department of Biology

in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

Boston College
Chestnut Hill, Massachusetts
December 2020

Computational approaches in infectious disease research: towards improved diagnostic methods
Defne Surujon

Thesis advisor: Professor Tim van Opijnen Defne Surujon

Computational approaches in infectious disease research: towards improved diagnostic methods

Abstract

Due to overuse and misuse of antibiotics, the global threat of antibiotic resistance is a growing crisis. Three critical issues surrounding antibiotic resistance are the lack of rapid testing, treatment failure, and evolution of resistance. However, with new technology facilitating data collection and powerful statistical learning advances, our understanding of the bacterial stress response to antibiotics is rapidly expanding. With a recent influx of omics data, it has become possible to develop powerful computational methods that make the best use of growing systems-level datasets. In this work, I present several such approaches that address the three challenges around resistance. While this body of work was motivated by the antibiotic resistance crisis, the approaches presented here favor generalization, that is, applicability beyond just one context. First, I present ShinyOmics, a web-based application that allow visualization, sharing, exploration and comparison of systems-level data. An overview of transcriptomics data in the bacterial pathogen *Streptococcus pneumoniae* led to the hypothesis that stress-susceptible strains have more chaotic gene expression patterns than stress-resistant ones. This hypothesis was supported by data from multiple strains, species, antibiotics and non-antibiotic stress factors, leading to the development of a transcriptomic entropy based, general predictor for bacterial fitness. I show the potential utility of this predictor in predicting antibiotic susceptibility phenotype, and drug minimum inhibitory concentrations, which can be applied to bacterial isolates from patients in the near future. Predictors for antibiotic susceptibility are of great value when there is large phenotypic variability across isolates from the same species. Phenotypic variability is accompanied by genomic diversity harbored within a species. I address the genomic diversity by developing BFClust, a software package that for the first time enables pan-genome analysis with confidence scores. Using pan-genome level information, I then develop predictors of essential genes unique to certain strains and predictors for genes that acquire adaptive mutations under prolonged stress exposure. Genes that are essential offer attractive drug targets, and those that are essential only in certain strains would make great targets for very narrow-spectrum antibiotics, potentially leading the way to personalized therapies in infectious disease. Finally, the prediction of adaptive outcome can lead to predictions of future cross-resistance or collateral sensitivities. Overall, this body of work exemplifies how com-

putational methods can complement the increasingly rapid data generation in the lab, and pave the way to the development of more effective antibiotic stewardship practices.

Contents

List of figures

List of Tables

List of Abbreviations

19F	*Streptococcus pneumoniae* strain Taiwan-19F
A.b	*Acinetobacter baumannii*
A19F	Adapted 19F
AG	Adapted gene
AMR	Antimicrobial resistance
AMX	Amoxicillin
ARIBA	Antimicrobial Resistance Identification By Assembly
AST	Antibiotic susceptibility testing
AT4	Adapted T4
AUPRC	Area under the PR curve
AUROC	Area under the ROC curve
BFClust	Boundary Forest Clustering
BIRCH	Balanced iterative reducing and clustering using hierarchies
BPGA	Bacterial Pan Genome Analysis tool
CARD	Comprehensive antibiotic resistance database
CD-HIT	Cluster Database at High Identity with Tolerance
CDM	Chemically defined medium
CDS	Coding sequence
CEF	Cefepime
CFT	Ceftazidime
CFU	Colony forming units
CIP	Ciprofloxacin
COT	Cotrimoxazole
CWSI	Cell wall synthesis inhibitor
DE	Differential expression
DEG	Differentially expressed gene
DEL	Deletion
DIAMOND	Diagonal measurement of n-way drug interactions
DSI	DNA synthesis inhibitor
DT	Decision tree
dW	Fitness change
E.c	*Escherichia coli*
EG	Essential gene

GEN	Gentamicin
GLY	Glycine deprivation
H	Entropy
HGT	Horizontal gene transfer
IMI	Imipenem
INS	Insertion
IPD	Invasive pneumococcal disease
K.p	*Klebsiella pneumoniae*
KAN	Kanamycin
LIN	Linezolid
LR	Logistic regression
LVX	Levofloxacin
MA	Massachusetts
MCL	Markov clustering
MER	Meropenem
MIC	Minimum inhibitory concentration
ML	Machine learning
MLP	Multi layer perceptron
MOA	Mechanism of action
MOX	Moxifloxacin
MSE	Mean square error
NB	Naïve Bayes
NDARO	National Database of Antibiotic Resistant Organisms
NFDS	Negative frequency dependent selection
ODELAM	One-cell Doubling Evaluation of Living Arrays of Mycobacterium
PanOCT	Pan-genome ortholog clustering tool
PanPhlAn	Pangenome-based Phylogenomic Analysis
PATRIC	Pathosystems Resource Integration Center
PBCN	Pneumococcal Bacteremia Collection Nijmegen
PCA	Principal component analysis
PCR	Polymerase chain reaction
PEN	Penicillin
PETRI-Seq	Prokaryotic expression profiling by tagging RNA in situ and sequencing
PGAP	Prokaryotic Genome Annotation Pipeline
PIRATE	Pangenome Iterative Refinement and Threshold Evaluation
PopPUNK	Population Partitioning Using Nucleotide K-mers
PR	Precision-recall

PSI	Protein synthesis inhibitor
RAxML	Randomized Axelerated Maximum Likelihood
RefSeq	NCBI Reference Sequence Database
RF	Random forest
RIF	Rifampicin
RNA-Seq	RNA sequencing
ROC	Receiver-operator characteristic
RSI	RNA synthesis inhibitor
S.au	*Staphylococcus aureus*
S.pn-23F	*Streptococcus pneumoniae* strain 23F
S.pn-1	*Streptococcus pneumoniae* strain 1
*S.*Ty	*Salmonella enterica* serovar Typhimurium
SDMM	Semi-defined minimal medium
SNP	Single nucleotide polymorphism
SSE	Sum of squared errors
SVM	Support vector machine
T4	Streptococcus pneumoniae strain TIGR4
TET	Tetracycline
TFOE	Transcription factor over-expression
THY	Todd-Hewitt yeast media
Tn-Seq	Transposon-insertion sequencing
TOB	Tobramycin
URA	Uracil deprivation
URL	Uniform resource locator
VAL	Valine deprivation
VNC	Vancomycin
WGCNA	Weighted correlation network analysis

"I have come to believe that caring for myself is not self-indulgent. Caring for myself is an act of survival."
— Audre Lorde

Acknowledgments

I knew that writing a dissertation would be no easy task. However, I did not anticipate doing this during a global pandemic, as an international student in Trump's America, and while dealing with mental illness. I've had many amazing people and resources who supported me throughout the process, and I hope that I could return at least part of their kindness.

It goes without saying that I am grateful for my advisor Tim van Opijnen, who was the one to suggest I join his lab as a PhD student. I would like to acknowledge my committee, Ken Williams, Michelle Meyer, José Bento and Babak Momeni, for their scientific and advisory input throughout the past few years. In addition, I would like to thank Mohamad Sater for his scholastic guidance, and his constructive feedback, always acknowledging accomplishments, and encouraging me to go further.

I would like to thank the many people I've had the pleasure of working with, especially the undergraduate students Alexander Farrell, Nabil Ghazal and Jakob Weiss. All members of the van Opijnen lab, especially Federico Rosconi, Suyen Espinoza, Juance Ortiz-Márquez, Bharathi Sundaresh, Derek Thibault, Sophie Bodrog, Indu Warrier, Stephen Wood and Bimal Jana. They have been not only incredible lab-mates who provided valuable scientific input every time I needed it, but also friends who have been there to laugh, cry, run, vent, and celebrate together. I don't think I could have gone through this journey without Federico's wisdom, Suyen's compassion, Juance's sense of humor, and Bharathi's love of plants.

Boston College isn't the most welcoming environment to be in. I appreciate the help and support I've gotten from both IRIS, and the BCGEU-UAW. At times when I felt most helpless, these groups provided avenues to push for change and advocacy for basic human rights that shouldn't have even been up for discussion. I would also like to specifically thank Sam Dyckman, Matt Crum, Danny Beringer and Elise Gray, for being amazingly supportive friends, and equally amazing colleagues. You have provided a much needed network of support during grad school.

Thank you to Deniz Haznedar, Idil Kalaycıoğlu, Neşe İnanç, Kumru Dikmenli ("Golden Girls"), Jake Walters, Dan Diner, Ben Otoo, and everybody else who stayed in touch after years of not seeing each other face-to-face. You are all incredible human beings, and great friends. Thank you for keeping me sane.

I would like to give my special thanks to Bill Fleming. He has been there for the most

difficult parts of this journey, always believed in me, and gave me something to be happy about when I though it wasn't possible. Thank you for believing me when I couldn't, always grounding me when I was spiraling, and reminding me of what matters when I lost perspective.
Finally, and most importantly, I would like to acknowledge my family, especially my parents, Edna and Jos Surujon, who not only shaped who I am today, but also have provided me with everything I needed to pursue a world-class undergraduate and graduate education, even though it meant I would be away from them. I love you both, so much. And although he didn't get a chance to see me publish or pursue a PhD, I know my father would be proud. This dissertation is dedicated to his memory.

1

Introduction

Bacterial pathogens impose a large burden on human health, as causative agents of various types of infections. To combat these pathogens, a number of antibiotics have been discovered, developed and used as therapeutics. Discovery of novel antibiotics was at full speed mid-20^{th} century, which is often referred to as the "golden age of antibiotics". Through unnecessary exposure of bacterial populations to antibiotics (antibiotic overuse), and the use of an antibiotic that is ineffective against a particular pathogen (misuse), the prevalence of antibiotic resistant pathogens has been increasing[205,31]. Combined with a stall in novel antibiotic discovery, this has led many to think the golden age of antibiotics is over.

In this chapter, I describe the three major challenges that make antibiotic resistance a pressing threat. I summarize the work that has been done to address these challenges, describe where these approaches have room for improvement and how this thesis fits in the fight against antibiotic resistance.

1.1 Antimicrobial resistance related challenges

There are 3 major challenges that contribute to the emergence and spread of antibiotic resistance: a lack of rapid testing, treatment failure and evolution of resistance. The first challenge is the **lack of rapid testing**. Currently, the clinical gold standard is to use culture-based approaches in order to identify the infection-causing pathogen, and determine its antibiotic susceptibility profile. These culture-based methods are time consuming[104,27], and are often labor and resource intensive. Antibiotic susceptibility testing (AST) can take days, which causes a delay in the treatment of the infection. When an infection progresses aggressively such as in septic patients, every hour of delayed treatment can increase the mortality risk by 7.6%[109]. When fast intervention is critical, and AST data is not available, broad spectrum antibiotics are used to maximize the chances of eliminating the pathogen. This kind of untargeted approach exposes the entire human microbiota to antibiotics, increasing the likelihood of resistance appearing.

Treatment failure is the second antibiotic resistance related challenge. The prescription of an antibiotic without confirming its efficacy on the infection-causing pathogen can fail to clear the infection. In addition, treatment failure can occur due to tolerance, heteroresistance, multi-drug resistance. Antibiotic tolerance is the ability of bacteria to survive under antibiotic treatment even though they are not resistant[12]. A mixed population of tolerant and susceptible bacteria is referred to as persistent, and a mix of resistant and susceptible bacteria makes up a heteroresistant population. Tolerance and heteroresistance are difficult (if not impossible) to detect using microdilution plates, especially when the susceptible subpopulation far outnumbers the tolerant or resistant one. Since the detection of tolerance is

difficult, it is unclear how it can be best addressed. On the other hand, heteroresistance and multi-drug resistance can be addressed with multiple-drug combinations, however the most effective combination is difficult to identify due to the number of possible combinations.

The final challenge, and perhaps the most alarming, is the **evolution of resistance**. Bacteria can quickly acquire resistance to antibiotics, which makes the utility of antibiotics short-lived. This makes it less worthwhile to invest time, money and effort into new antibiotic discovery. Without new antibiotics, and existing ones becoming increasingly ineffective, treatment options for infections become severely limited.

These challenges have been difficult to address in the past, due to a limited understanding of the physiological effects of an antibiotic on a bacterial pathogen. The response triggered can be specific to the antibiotic-pathogen combination, requiring high-throughput data collection and sophisticated data analysis methods to gain a comprehensive understanding of antibiotic mechanisms of action and resistance.

There have been major advances in high-throughput data generation. Simultaneously, statistical learning approaches are being incorporated into microbiology research. Both of these factors allow for the 3 antibiotic resistance associated challenges to be addressed in various ways.

Existing literature focuses either on a single predictive task (e.g. predicting the effectiveness of drug combinations[195,209], prediction of susceptibility[180]); a single approach (e.g. genome-scale metabolic models[129,60]); or highlight one method and its general applications, without an antibiotic resistance focus (e.g. machine learning in microbiology[149]). Antibiotic resistance is a complex, multi-faceted problem; lack of rapid testing, treatment failure and evolution of resistance are related but separate challenges that need to be tackled si-

multaneously. For instance, identifying effective drug combinations without considering the potential evolutionary consequences would be a short-sighted approach that can lead to new problems in the future. With the increasing availability of large-scale multi-omics datasets and novel applications of sophistical statistical methods, it is possible to generate predictive tools that have complementary utility. Together, predictions on antibiotic susceptibility phenotype, treatment outcome, and evolutionary outcome can inform clinical decision making in a way that most effectively limits the spread of antibiotic resistance. In the next section I outline the experimental and mathematical advances that enable the generation and analysis of large scale, systems-level data. In the following sections I explore the ways in which each of the three main antibiotic resistance challenges are being addressed

1.2 Advances in high-throughput technology and statistical methods help address AMR-related challenges

The challenges outlined in the previous section are likely to be solved now, with technological and computational innovations being incorporated into infectious disease research. Omics data generation technology has been improving considerably. This allows for the identification of new biological information, characterization of new genes and pathways, and also feeds into computational prediction tools being developed. Here I highlight recent advances in high-throughput data generation that makes the training of complex, data-driven models possible. **I.** Whole genome sequencing is most popularly done using short read technologies such as Illumina[17]. Recently, long read technologies have made it possible to generate substantial portions of contiguous genomic sequences[152,95]. As additional

benefits, the PacBio platform allows the collection of methylome data[64], and the MinION device from Oxford Nanopore Technologies allows real-time data generation in field work[88]. **II.** Phenotypic screens that involve the generation of a mutant library is quickly being replaced by high throughput technologies such as Tn-Seq[198,199,122]. These newer methods have also been shown to be customized for specific purposes. For instance, microdroplet encapsulation of each mutant makes it possible to decouple the fitness effect of the mutation itself from the effect of cooperation or competition within the population of mutants[190]. **III.** Transcriptomics for bacterial pathogens has been possible with RNA-Seq for over a decade[145]. There are a number of variations of RNA-Seq for prokaryotes[155], including dual RNA-Seq to profile host and pathogen transcriptomes simultaneously[6], and PETRI-Seq to profile single cells[22]. It is also possible to use long read technologies to obtain full transcripts[80]. **IV.** Multi-omic data visualization and exploration is becoming increasingly critical in microbial research. Exploratory visualizations and analyses on a platform such as ShinyOmics[185] (described in Chapter 2) can lead to new hypotheses that can later be explored further.

New applications of statistical learning are becoming more prevalent in microbiology. Existing models (such as regression, SVM, random forest[85]) can be customized and trained on omics data, and can be used as predictors of clinical outcomes relevant to antibiotic resistant infectious bacteria. In the next few sections I summarize recent applications of statistical learning and omics-screens in this field, emphasizing the advancements in the past decade (Figure 1.1)

1.3 Addressing a lack of rapid testing

Determining the correct antibiotic to use for a specific infection can be challenging and/or time consuming. Therefore, clinicians often resort to the use of ineffective or broad-spectrum antibiotics before AST data is available. The current gold standard relies on methods that hinge on culturing isolates from a patient sample, and observing their growth characteristics under antibiotic selection. This is not only time consuming (e.g. when assaying slow-growing organisms such as *Mycobacterium tuberculosis*), but also require the use of large amounts of growth media, expensive antibiotics, and are extremely labor intensive when multiple drugs are being screened at multiple concentrations. Moreover, the results of culture based methods are highly variable, and dependent on factors such as the initial inoculation density[26]. While there exist protocols with regulatory-approval that leverage automation and increase reproducibility, these methods also require the sub-culturing of clonal isolates from patient samples which can miss variants in a non-homogenous population, and remain sensitive to the initial culture density[136]. Predictors of antibiotic sensitivity on the other hand, especially if they can be used on patient samples directly, would have the added advantage of not relying on an initial subculture step. Use of a rapid, point-of-care diagnostic predictor that can evaluate what antibiotic would be effective for a specific case has the potential of reducing the spread of antibiotic resistance, as predicted in theoretical models[194]. Determining the appropriate antibiotic treatment that is customized for one case can also reduce the risk of disrupting the microbiota of the patient, and risking secondary infections of for instance *Clostridium difficile*[74]. Instead of culture-based AST methods which are inherently time consuming, labor-intensive, and are not fully repro-

ducible, rapid diagnostics can rely on the prediction of antibiotic susceptibility from data that can be obtained much quicker. For instance, nucleic acids can be isolated from bacteria and sequenced on the order of hours. With the use of computational models that use the sequencing data and can distinguish between resistant and susceptible isolates, antibiotic susceptibility can be predicted much quicker than the culture-based AST protocols, which can take days to weeks.

1.3.1 Prediction of antibiotic susceptibility using genomic sequence data

Certain antibiotics have well-understood resistance mechanisms. The existence of a resistance-causing genetic element can be a strong predictor of the resistance phenotype. Rule-based methods use presence/absence of such resistance markers[130].

There are several well-characterized examples of the presence of an allele or a genetic element determining the antibiotic resistance phenotype of the organism. For instance, the presence of beta-lactamase-coding genes allows the bacterium to destroy beta-lactam antibiotics, thus conferring resistance to this class of drugs in multiple Gram-negative species[210]. SNPs can also confer resistance. A mutation in the *gyrA* gene that results in the amino acid change T86I on DNA gyrase, the target of fluoroquinolones, also confers resistance, by interfering with drug-target binding[62]. Since many such genetic determinants that explain different cases of resistance have been characterized, genomic sequences from clinical samples can be used to identify resistance[84].

Rule-based models have the advantage of being straightforward and easily interpretable. Some use a single "rule" to infer a single type of resistance. For instance, *Escherichia coli* resistance to Amoxicillin-Clavulanate can be predicted using beta-lactamase presence, pro-

moter mutations and copy number[49]. In a similar, targeted approach, the resistance-causing SNP in *gyrA* can be rapidly detected using a using mismatch amplification mutation assay in *Campylobacter jejuni*[74]. Broader rule-based methods rely heavily on curated databases that list the genotypic markers of resistance for each antibiotic e.g. CARD and Mykrobe[132,91]. For instance, 500 strains of *Staphylococcus aureus* have been classified as resistant or susceptible using such genetic presence/absence information[78]. A more comprehensive tool named ARIBA allows for a similar approach to be taken with a variety of pathogens[92]. While multiple predictions can be made simultaneously, these methods are inherently limited to resistance mechanisms that are well-studied and characterized. Nevertheless, with faster sequencing technology, whole genome sequence data becomes more widely available (e.g. NDARO, PATRIC) to train/test these predictors.

1.3.2 Data-driven prediction of antibiotic susceptibility using omics data

When the resistance mechanism may not be well-understood, generating a rule-based predictor from prior knowledge is not possible. In these cases, whole genome sequencing data can be used to train "black-box" type machine learning models, which learn the rules that associate with resistance phenotype, without a priori knowledge. With this approach it is possible to identify novel genetic markers that associate with resistance. For instance, VAMPr uses XGBoost models to predict susceptibility in an extensive dataset spanning 9 species and 29 antibiotics. The authors were able to identify which genetic features were prioritized in their predictive models, and found a set of well-characterized mutations, as well as some previously unknown genotype-phenotype associations[106].

Beyond genomics, transcriptomics, metabolomics, proteomics screens can also identify

potential markers of resistance phenotype. With the wider availability and faster collection of omics data, sophisticated statistical approaches can now be tested on newly generated, large datasets, yielding powerful predictors of antibiotic resistance. For instance, gene panel based approaches can use differential expression [14] or normalized expression [186,89] of a small set of genes and predict antibiotic susceptibility. Similar to VAMPr, the feature selection step in these approaches can uncover novel associations between phenotype and transcriptional regulation. Cell morphology after antibiotic exposure can also be used to distinguish resistant and susceptible strains, as has been demonstrated with *S. aureus* [150]. While all these methods rely on data collection *in vitro*, there are also ML models that are trained on the clinical history of the patient in order to determine the best treatment option [213]. These prediction methods show that it is possible to move away from culture-dependent susceptibility testing.

1.4 Addressing treatment failure

1.4.1 Identification of heterogeneous populations that can survive antibiotic treatment

If a pathogenic population is composed of susceptible and resistant sub-populations, antibiotic treatment will only eliminate the susceptible sub-population, and the surviving resistant sub-population may cause recurring infections. It is possible to re-purpose classical plating-based approaches to detect heteroresistance. For instance, colistin heteroresistance for *Acinetobacter baumannii* can be observed as colony growth on the zone of inhibition around and antibiotic disc. In this approach, the fraction of the population that is resistant can be estimated with the number of colonies observed in the zone of inhibition [166]. Similar to classical AST methods, this culture-dependent approach has the disadvantage of being

time consuming and resource-intensive. There are more granular approaches that involve microfluidics and single-cell probing of morphological, genetic and regulatory features. Using single cell morphological tracking, the ODELAM system allows for individual cells to be classified as resistant or susceptible[87]. Bacteria in a mixed population can also be encapsulated in microdroplets, in which digital PCR can be performed. This has successfully detected fluoroquinolone resistance conferring mutations in mixed *Legionella pneumophila* populations from patient respiratory samples[86]. If the characteristic transcriptomic signatures of resistant and susceptible strains are known, it may also be possible to perform PETRI-Seq[22] to classify single bacteria as resistant or susceptible.

1.4.2 Prediction of synergistic drug combinations

Multi-drug resistant strains may not be effectively cleared by a single drug, but can be better targeted using a combination of drugs. Certain combinations of antibiotics are known to act synergistically, where the combination treatment is more effective in stopping bacterial growth than expected. Multidrug combinations are recommended for tuberculosis patients[59], however the combination therapy is generally a one-size-fits-all solution that might include more antibiotics than necessary for a specific patient. Identifying the effective combinations of fewest drugs is crucial, however, due to the astronomical number of potential combinations, screening all combinations may be prohibitively expensive and time-consuming. It is therefore crucial to prioritize potentially efficacious drug combinations in a time-efficient manner.

DIAMOND is an approach that makes *in vitro* screens faster[42]. It does so by selecting a small subset of concentration combinations for a drug pair to be tested. While this makes

the synergy testing more resource-effective, and makes it possible to multiplex more tests at a time, DIAMOND still relies on bacterial culture. Alternatively, increased omics data collection and availability, better computational resources enable rapid screening for candidate combinations. In order to truly screen drug combinations in a high-throughput manner, it is necessary to use computational methods. A random forest classification approach was taken with *E. coli* and *Mycobacterium tuberculosis*, where the classifier was trained on combinations of phenotypic screen data coming from single-antibiotic exposure conditions[127,32]. In fact, in a retrospective patient outcome study, the level of synergysic interaction among drug pairs predicted *in silico* and validated *in vitro* correlated with infection clearance in patients[127]. It has yet to be seen whether newly identified drug pairs that don't yet have clinical outcome data will be efficacious in trials. Other than its predictive power, an advantage of random forest in this case is that it can report feature importance in the trained model. This information can be used in understanding the regulatory basis of drug interaction outcomes[127]. Campos and Zampieri have taken a systems biology approach, going beyond a single omics data type for training. They combine chemogenomic screens with metabolomic ones to predict synergy for new drug combinations[29]. With omics-level data collection becoming cheaper and easier, it is reasonable to expect an array of computational models trained on various combinations of such data in the near future.

1.5 Addressing adaptive evolution

Using drugs in combination might be a good solution to treat an individual infection, but may increase the use of antibiotics overall. Using multiple drugs for a single infection has the potential benefit of clearing the infection-causing pathogen, but also the risk of

exposing the commensal microbiota to multiple antibiotics. With increased exposure to multiple drugs, the likelihood that any one of the commensal bacteria develop resistance increases. Whether antibiotics are used individually or in combination, their use comes with the risk of selecting for resistant strains of pathogens. This has inspired the discovery of new antibiotics[116,119]. A good antibiotic needs to be effective now, but also in the future. Resistance can emerge soon after (or sometimes before) the antibiotic is introduced into clinical practice[31], making it crucial to predict bacterial adaptation to antibiotics[173,69].

1.5.1 Population genetics models explain the emergence and spread of resistance

The evolutionary trajectories followed by bacteria that acquire resistance can be complex. However, established models on population dynamics can explain how certain lineages fix in a population[143]. An increasing numbers of lab-directed adaptive evolution experiments, including Lenski's decades-long evolution experiment confirm and improve existing population dynamics models[114]. In the context of antibiotic resistance, the determinism in evolution can be seen in the similarity of genotypes in bacterial populations adapted to antibiotics in a morbidostat[193]. In this study, all five populations adapted under trimethoprim selection acquired mutations in the same gene, *dhfr*, which encodes the drug target.

In addition to lab-directed adaptation experiments, microbial evolution can be studied on a larger scale, using retrospective analysis of genomic sequences from isolates collected from patients. Rapid whole-genome sequencing of surveillance and outbreak populations facilitates large-scale retrospective studies on adaptation. For instance, the negative frequency dependent selection model was proposed to be governing the genetic content of *Streptococcus pneumoniae* strains from distinct populations[?]. This work was followed up with

a study showing the predictive ability of NFDS in determining which strains increase or decrease in their prevalence in a population[9].

Studies on real-world bacterial populations are done on samples that experience not only antibiotics, but also other selective pressures as well. Therefore, the findings on an *in vitro* population might not necessarily be concordant with those from an outbreak population. However, a study on both experimentally adapted and clinical strains of beta-lactam resistant *E. coli* identified common trajectories for mutations in beta-lactamase[82].

1.5.2 Data-driven models for a priori prediction of resistance

While population genetics models retrospectively explain the observed adaptive trajectory, it is also possible to make *a priori* predictions on resistant strains that haven't been observed yet. Several studies take a data-driven approach to identify which genes will acquire adaptive mutations as a population is continually exposed to antibiotics. Better computational resources (in terms of hardware and more efficient code) make it possible to train complex models to this end. For instance, a machine learning model trained on experiment metadata predicts adaptive outcome in *E. coli*, although with modest precision and recall[207]. A big challenge in this type of prediction is the class imbalance – that there are very few genes that acquire adaptive mutations. Therefore, the growing compendium of adaptive evolution experiments is likely to improve performance on such models by providing large datasets to train on.

1.5.3 Novel antimicrobial discovery

Antibiotic discovery has been driven by prospecting approaches that identify existing molecules that microbes use to kill competitors inhabiting the same niche. An example of this is the discovery of teixobactin in a screen of soil bacteria that were difficult or impossible to culture previously[119]. This approach is more recently being replaced by computer-aided design of novel antibiotic candidates. For instance, computer-aided search for antimicrobial peptides is becoming a promising avenue, with increasing ease of peptide synthesis[148]. It is also possible to train a model to learn the molecular features of an effective antibiotic, and screen through vast chemical libraries. This approach led to the recent discovery of halicin, which was validated to be effective against multiple clinically relevant pathogens[178].

1.6 Overview of this thesis

In this thesis, I present a body of work that addresses the three challenges of antibiotic resistance through innovative computational approaches, leveraging recent advances in high-throughput screens and statistical methods. In Chapter 2, I present ShinyOmics, a data visualization and analysis platform that allows convenient exploration of multi-omics data. By generating customizable plots, tables and comparisons across datasets, ShinyOmics has enabled novel hypotheses to be generated and later tested. One such hypothesis was transcriptomic entropy being a general predictor of fitness under antibiotic stress, which can be used as an alternative to AST. In Chapter 3, I validate this hypothesis, and show that quantifying entropy yields a simple fitness predictor that generalizes for a number of bacterial species, antibiotic conditions, and is even applicable to non-antibiotic conditions. In Chap-

ters 4 and 5, I focus on the *S. pneumoniae* pan-genome and population structure, and train predictors of gene essentiality and adaptability, addressing treatment failure and evolution towards antibiotic resistance. Chapter 4 presents BFClust, a pangenome clustering tool that is unique in its ability to report confidence scores on its output. Reliable, high-confidence clustering results are necessary for the cross-strain comparisons and phylogenetic analyses involved in the predictions in Chapter 5. The prediction of essential genes, especially in context-specific cases, has the potential to identify targets of narrow-spectrum antibiotics that can be used in a personalized fashion for each infection case, making treatment failure less likely. Finally, the predictions on gene adaptability presented in Chapter 5 incorporate short-term stress response data from the ancestral strain and pan-genomic data in an ensemble machine learning model, distinguishing them from existing approaches.

Overall, this thesis outlines a significant knowledge and code base that addresses the antibiotic resistance crisis from multiple angles. While the motivation behind the work presented here is addressing the antibiotic resistance crisis, the approaches I develop are meant to be generally applicable, therefore have the potential to contribute to other areas of research as well.

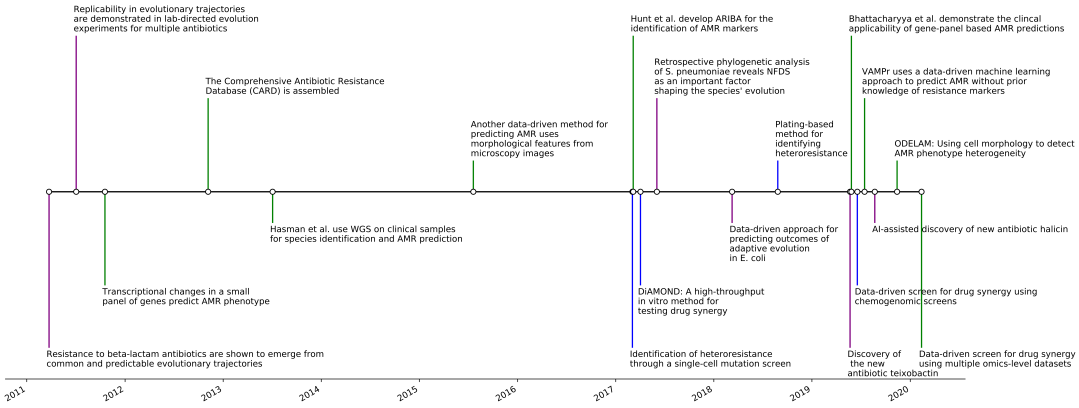

Figure 1.1: Notable studies in the past decade that address the three antibiotic resistance challenges are summarized on a timeline. Each annotated point represent one study cited in this chapter. The vertical bars are colored based on the specific challenge. Green: addressing the lack of rapid testing, blue: addressing treatment failure, purple: addressing the evolution of resistace

2

ShinyOmics: Collaborative Exploration of Omics-Data *

2.1 Background

Omics-profiling is becoming increasingly prevalent in many subfields in biology. For example, genome-wide transcriptomics have been used in studies of gene expression during embryonic stem cell differentiation, host-pathogen interactions, identification of biomarkers associated with antibiotic resistance and cancer disease progression[40,211,6,14,103,204,146,126]. Similarly, proteomic screens can identify proteins relevant for virulence, or cancer biomarkers[215,1,13,5]. Furthermore, phenotypic profiling using transposon insertion sequencing (Tn-Seq) in human pathogens has identified genes involved in colonization, infection, and intrinsic antibiotic resistance; and has been used in genetic interaction mapping[201,198,202,72,200,97].

Since genome-wide multi-omic profiling is paving the way to such varied and clinically

*Adapted from Surujon D, van Opijnen T. ShinyOmics: collaborative exploration of omics-data. BMC Bioinformatics. 2020 Jan 17;21(1):22. Author contributions: DS developed the application and wrote the paper. DS and TvO edited and approved the final manuscript.

relevant applications, considerable effort has gone into establishing analysis pipelines that process the resulting data. Tools such as DESeq2[125] and MAGenTA[133] are used for statistical analysis of differential gene expression and fitness changes respectively. However, the volume of the analyzed data can make interpretation and comprehensive evaluation non-trivial. Moreover, these tools often do not accommodate easy incorporation of metadata pertaining to genes and/or experimental conditions. This makes it time consuming and labor intensive to apply custom analysis protocols on each dataset, especially if the user has limited programming experience.

ShinyOmics offers several visualization and comparison options that are designed to assist in novel hypothesis generation, as well as data management, online sharing and exploration. Sharing a single URL can enable your collaborators and readers to interactively explore your datasets, and generate publication-quality figures. Moreover, ShinyOmics can be used as an interactive supplement accompanying research articles or presentations.

Existing tools for user-friendly data exploration and visualization include Stemformatics[38], Metascape[220], and mixOmics[153]. Stemformatics is an online portal that assembles gene expression data from stem cell datasets. While it provides an interactive visual interface, Stemformatics is tailored for stem cell research, and hosts a specific and focused dataset that does not expand to fields other than stem cell research. Metascape does allow users to supply their own datasets (often in the form of a gene list extracted from DE or other omics profiling data), and can merge information from public databases as well as perform functional enrichment and network analyses. The heavy dependence on well-curated annotation and information on public databases can be a limitation for researchers working with less well-characterized organisms, where these annotations may not be read-

ily available; or available to the user but not yet made public. Moreover, even though the user can provide gene lists extracted from different omics screens, these analyses are performed independently. mixOmics is an R package that allows the user to interact with and analyze their own (potentially unpublished) data with less reliance on public databases, and consider multi-omics data simultaneously. It provides multiple pipelines focused on dimensionality reduction and feature selection, which can be extremely valuable in determining what signatures are associated with for instance disease outcome. However, if a researcher's interests are more specific, e.g. asking what expression changes are observed for a specific set of genes, a more customizable platform may be better suited.

To complement existing tools, we present ShinyOmics, a browser-based interface that allows for customizable visualizations of genome-wide profiling data, incorporating user-supplied metadata from genes and experimental conditions, and network connectedness of genes. It is straightforward to swap out the existing datasets loaded in ShinyOmics with user-generated custom data; e.g. standard output from DESeq2 can directly be incorporated. This feature of ShinyOmics also facilitates data management and sharing; for example, a lab can host a fully interactive instance of ShinyOmics with their own data making it accessible to collaborators across the world through a URL. This creates a convenient alternative over transferring and describing a large number of spreadsheets and data files between labs. Moreover, ShinyOmics can be deployed with new data obtained in a research project, as an interactive supplement that can be included in a manuscript submission, or academic presentation. This chapter is a description of the application, and serves as a walkthrough of the kinds of preliminary analyses that can be done using it.

2.2 Implementation

ShinyOmics was developed in R version 3.4.3[189], using RStudio version 1.1.419[188]. Running the app locally requires the packages ggplot2[?] (v3.1.0), visNetwork[3] (v2.0.5), RColorBrewer[139] (v.1.1), igraph[48] (v1.2.2), heatmaply[71] (v.0.16.0), shinyHeatmaply[170] (v.0.1.0) and shiny[33] (v1.2.0).

An example of the app with data from[221,70,161] is available at[183]. The source code for the app and detailed usage notes can be accessed from the ShinyOmics GitHub repository[181]. Detailed usage notes are also provided in the aforementioned link.

There are three types of custom data that can be added; genome-wide profiling data, strain metadata, and network data. The main reference file for the app is "exptsheet.csv" under the "data" subdirectory. Any added experiment needs to be recorded in this file, with the corresponding profiling and metadata file locations specified. At minimum exptsheet.csv should have columns "Experiment", "Time", "Name", "DataFile", "Strain", and "MetadataFile". There can be as many additional columns as desired to record metadata of the experiments. For profiling data files, the standard output of DESeq2 can be directly transferred to the "data" directory. Alternatively, a file with at least the columns "Gene", "Value" (e.g. log2 fold change of expression), and "padj" can be provided. While the data source can be any organism or strain, eukaryotic datasets with tens of thousands of genes are likely to cause significant lag in the application loading. We therefore recommend, in the case of eukaryotic data, filtering the dataset (based on the number and quality of reads, or variability among replicates) and working with only a subset of a few thousand genes at most. There needs to be one metadata file per strain, and the minimum requirement for

each metadata file is one column labeled "Gene". Each metadata file can have as many columns as desired, all selectors on the app will adjust accordingly. Finally, the networks should be specified as edge tables, with two columns: "source" and "target", and be named "[Name]_Edges.csv" in the "data/networks/" subdirectory. The network statistics will be computed automatically.

When the app is first loaded in the browser, all data/metadata files and the experiment sheet will be screened and validated for the requirements mentioned above. If the files provided do not fit these specifications, pop up error messages will indicate what caused the validation to fail, in which file(s), and the app will load with no data.

2.3 Results

We provide a version of ShinyOmics pre-loaded with multi-omic data from two human pathogens; *Streptococcus pneumoniae* and *Mycobacterium tuberculosis*. The *S. pneumoniae* dataset includes Tn-Seq and RNA-Seq data from two strains (TIGR4 and 19F) that were exposed to 1x Minimum Inhibitory Concentration (MIC) of kanamycin (KAN), levofloxacin (LVX), rifampicin (RIF), vancomycin (VNC) and penicillin (PEN) for 2-4 hours[221]. Differential expression (DE) on the RNA-Seq data was evaluated as the fold change in transcript abundance comparing antibiotic conditions to a no-antibiotic control using DESeq2[125]. Fitness change (dW) on the Tn-Seq data was evaluated comparing antibiotic to no-antibiotic conditions as described in[97]. The *M. tuberculosis* dataset includes microarray data[70] and proteomics data[161] under hypoxic conditions over a span of up to 20 days of culture in vitro. In its current configuration there are four panels that allow for different types of visualization: Single Experiment, Comparison of 2 Experiments, Comparison of All Experiments,

and Network Visualization.

In ShinyOmics the first panel is designed to explore relationships between a value associated with all genes (e.g. DE, dW, protein abundance) and any other user supplied metadata (Figure 2.1). The user can include other genome-wide profile data (e.g. change in fitness, dW) in the metadata fields, or as a separate experimental data file. In the Single Experiment panel, DE is plotted against the selected metadata type. For instance, in the preloaded dataset, one can answer whether there are significant DE changes appearing in a specific cellular function, by selecting "Tag1" (primary functional tag of the gene) from the dropdown menu labelled "Variable" (Figure 2.1). The resulting scatter plot has each gene as a point, with the categorical variable "Tag1" on the x-axis and DE on the y-axis. The plot is faceted by timepoints, i.e. each timepoint in the selected experiment is a separate panel. The user can select which timepoints to display or hide using the checkboxes on the right. There are several visualization tuning options, such as changing the transparency of points, or in the case of categorical x-axis variables, adding some noise (or "jitter") to the x-coordinate of each point (such that individual points do not overlap) and/or superimposing a violin plot. It is also possible to display only a subset of genes by pasting a gene list in the text box ("Paste gene list"), subsetting the genes by a metadata variable ("Select genes by metadata variable"), or to select genes directly from the plot by dragging a rectangle to define a region of interest (or "brushing") the plot. The brushed genes will be displayed in the table below. Clicking anywhere on the plot will reset the brushing. In the example provided, it is possible to identify a set of genetic information processing genes that are upregulated drastically when *S. pneumoniae* is exposed to kanamycin (Figure 2.1). Kanamycin, an aminoglycoside, is a protein synthesis inhibitor that triggers the incorpo-

ration of erroneous amino acids during protein synthesis, leading to an accumulation of misfolded proteins [38]. In *S. pneumoniae* TIGR4, the Clp protease ATP-binding subunit (SP_0338) is upregulated 256-fold (Figure 2.1), indicating a response by this organism to alleviate the antibiotic stress through the destruction of misfolded proteins. This is accompanied by the simultaneous upregulation of chaperones *dnaK* and *grpE* (SP_0517 and SP_0516), whose function it is to repair denatured and misfolded proteins[160].

The Compare 2 Experiments panel allows for quick pairwise comparisons of experiments (Figure 2.2). Here, one can plot the DE of one experiment against another, for the timepoints that are in common in both experiments. There is a selector for the color of the points (e.g. one can color each gene by functional category, or any other metadata feature). The plot is brushable, similar to the Single Experiment panel. As an example, the DE of two antibiotics are compared in Figure 2.2. Vancomycin and penicillin are both cell wall synthesis inhibitors, and the transcriptomic changes in response to these antibiotics appear highly correlated, especially in the later timepoints (Figure 2.2). This global similarity in transcriptional profiles is unique to the PEN-VNC pair, and is not observed when comparing antibiotics of different classes. In contrast, at 90-minutes a group of genes are brushed (SP_0044-SP_0054, Figure 2.2) belonging to the category "Nucleotide metabolism" that turn out to be downregulated across most of the tested antibiotics, including the RNA synthesis inhibitor Rifampicin, and the DNA synthesis inhibitor Levofloxacin. This set of genes are part of the purine biosynthesis pathway, and their downregulation might point to a common antibiotic response in *S. pneumoniae* TIGR4.

It is also possible to see whether different systems under the same condition harbor similar responses using the Compare 2 Experiments panel. Comparison of Tn-Seq and

RNA-Seq data from *S. pneumoniae* antibiotic experiments and a comparison of microarray and proteomic data from *M. tuberculosis* shows a lack of similarity in the responses in the different screens (Figure 2.3). This is in accordance with previous findings that systems-level data are often quite distinct, and different systems should not be taken as substitutes of one another, but rather complementary parts of the organism as a whole[97,76].

To identify general patterns across many experimental conditions, the Compare All Experiments panel can be used (Figure 2.4). On the left of this panel, a heatmap shows all genes across all conditions, with optional dendrograms showing hierarchical clustering. The heatmap on the bottom is interactive, and shows only a user-specified set of genes, and conditions. On the right side of the panel, principal component analysis (PCA) results are visualized. The first scatter plot shows all experiments on any combination of the top 10 principal components. The user can select which components to plot, and a metadata variable to color the points by (e.g. in order to see whether the experiments are separated by antibiotic, one can select "AB" as the color variable in the pre-loaded dataset). For instance, Figure 2.4 shows clear separation of Rifampicin from the other 4 antibiotics. Rifampicin, being an RNA synthesis inhibitor, elicits the most dramatic changes in expression out of the 5 antibiotics included. The last plot shows the percent variance explained by each principal component. The informative components will be those that explain more of the variance in the data. A common way of selecting important components is to look for an 'elbow' in the last plot (i.e. a relatively clear point on a line where the slope changes drastically) and consider the components before the elbow[30].

In order to evaluate whether genes with for instance significant DE (DEGs) or dW are related to one another in a network context, the last panel (Network) allows visualization

of a user supplied network of genes. Common types of biologically meaningful networks include protein-protein interaction[162], transcription regulatory[61] metabolic[98] and genetic interaction[44] networks. Depending on the organism, these networks can be manually curated, inferred bioinformatically[90,171,217], or might already be experimentally mapped out. The preloaded metabolic networks were generated by Jensen et al.[97]. It is also important to keep in mind what kind of network is being used, in order to draw meaningful conclusions from the network analysis. For example, all DEGs localizing on a certain part of the transcription regulatory network may be a result of the DEGs belonging to the same regulon. However, the same phenomenon on a metabolic network may mean a specific metabolic pathway is being activated, which would imply a functional relationship between DEGs. The panel allows the user to select the experiment, timepoint and network, leading to DEGs marked on the network as red and blue nodes for up- and down-regulation respectively. On the example metabolic network of *S. pneumoniae* 19F (initially generated in[97]), the 120-minute VNC response is overlaid (Figure 2.5). It is possible to pick out numerous groups of interconnected genes that are up- or down-regulated together, although there are also examples of upregulated genes being adjacent to downregulated or non-DE genes. On the left, the network itself will be visualized in an interactive plot that allows zooming, selecting and dragging of nodes. On the right, a set of selectors allow for a custom scatter plot to be made, relating network characteristics of nodes (e.g. degree) to DE or any other metadata supplied by the user. As an example, network degree is plotted against sequence diameter (how variable the sequence is across multiple strains of *S. pneumoniae*), and genes are colored by whether or not they are essential in 19F (Figure 2.5), showing a lack of relationship between these variables. Similar to scatter plots in the other panels, this

plot is also brushable, and brushed points are displayed in the table below.

2.4 Conclusion

While genome-wide profiling can be incredibly valuable in a variety of applications, initial exploratory analysis of large datasets can be a daunting task. For instance, enumerating the DE of each gene with tools such as DESeq2 is a necessary but insufficient step in such analyses. ShinyOmics is a simple platform for facilitating initial exploratory analysis of omic-profiling data and hypothesis generating. The emphasis on relating genome-wide profiling to custom, user supplied metadata enables the user to make functional associations between any set of features of genes. Moreover, ShinyOmics serves as a convenient data management and sharing tool. Deploying an instance of ShinyOmics with data from a new study results in an interactive supplement for research articles or presentations. For example, a modified version of ShinyOmics accompanying a manuscript with the full antibiotic response dataset from[221] can be found at[184].

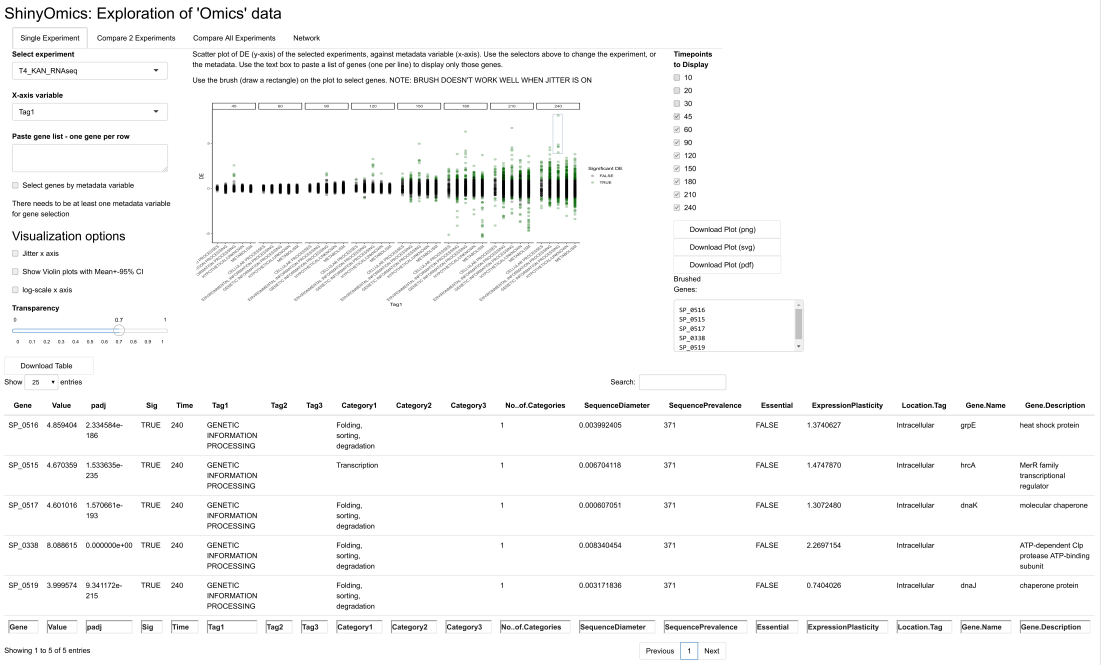

Figure 2.1: The tabs above allow the user to navigate to different panels. On the left, there is an experiment selector (where options are populated from the experiment sheet supplied by the user), a gene list selector (when empty, all genes are displayed), a variable selector, and several visualization customization options. Here, the T4 kanamycin ("T4_KAN") experiment is displayed as a scatterplot. Setting the x-axis variable to "Tag1" splits genes by functional Tag. 4 genes are brushed at timepoint 240 (blue rectangle), whose identity and metadata are displayed in the table (bottom).

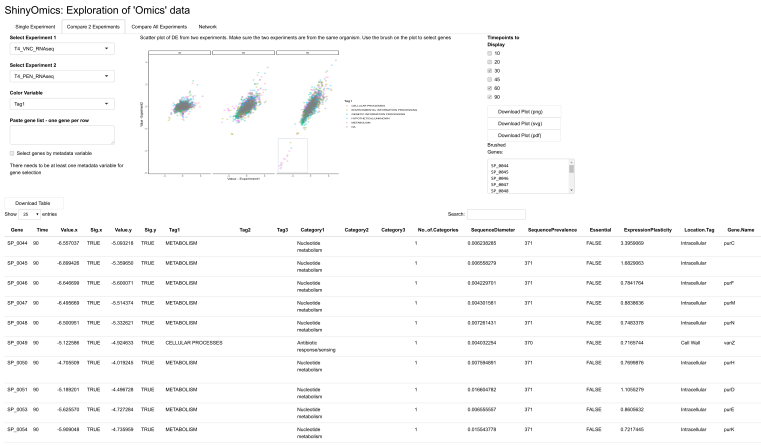

Figure 2.2: On the left are selectors for the two experiments to be compared, and a color variable. Here, DE from vancomycin (VNC) and the penicillin (PEN) are being compared for T4. Blue box on the plot indicates a set of brushed points. The table below the plot (cropped) displays all available information regarding the brushed points.

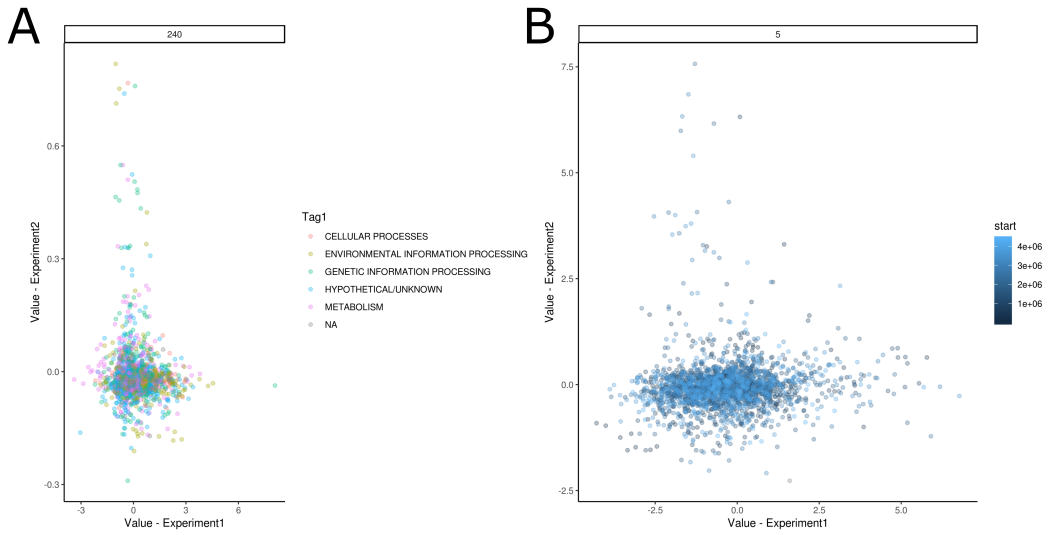

Figure 2.3: A. For the TIGR4 KAN experiment, RNA-Seq (Experiment 1) is plotted against Tn-seq (Experiment 2). B. For the *M. tuberculosis* hypoxia experiment, microarray data (Experiment1) is plotted against proteomics data (Experiment 2).

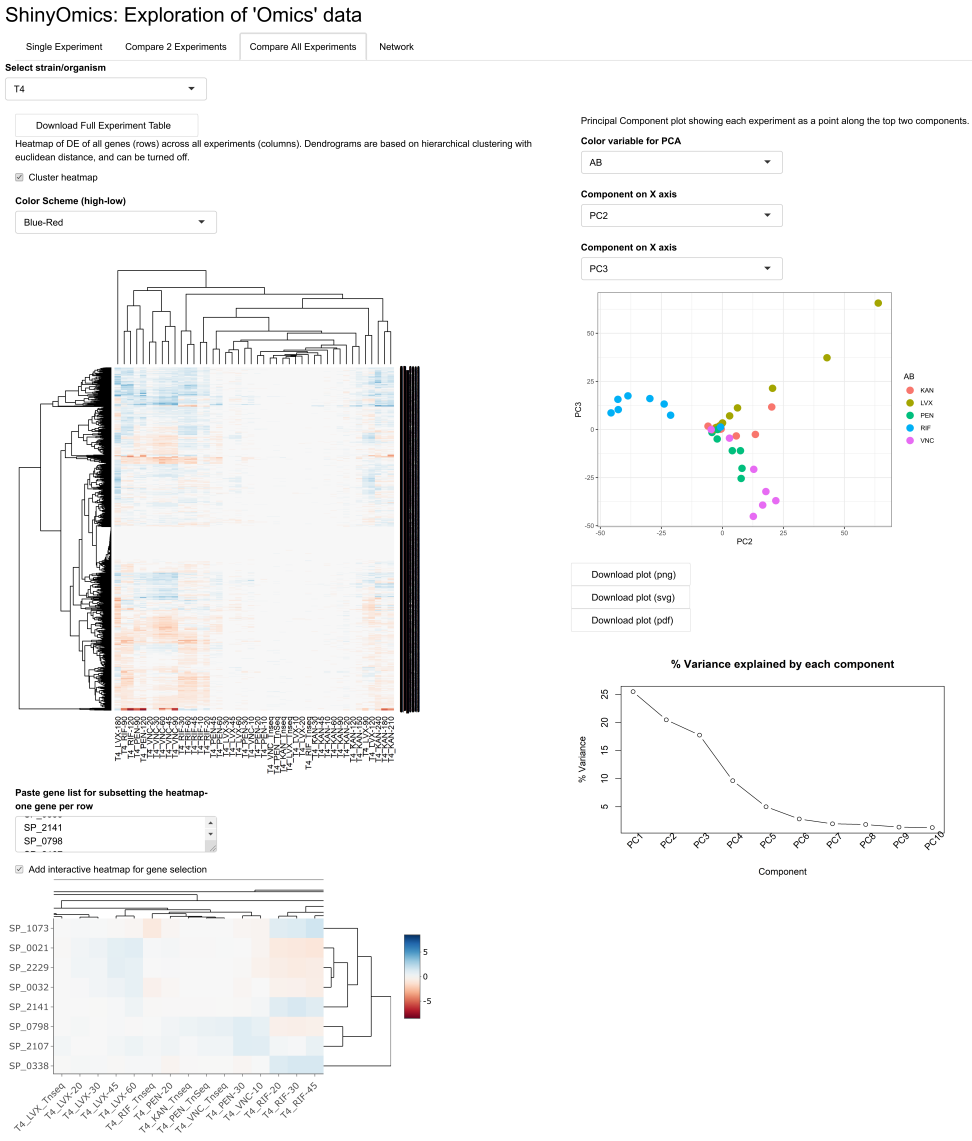

Figure 2.4: The heatmap shows DE of all experiments included in the experiment sheet for a specific strain (T4: TIGR4). The dendrogram on the heatmap and the PCA (colored by antibiotic) shows that the RNA synthesis inhibitor rifampicin (RIF) is most dissimilar to other antibiotics. AB: antibiotic. KAN: Kanamycin. LVX: Levofloxacin. VNC: Vancomycin. PEN: Penicillin.

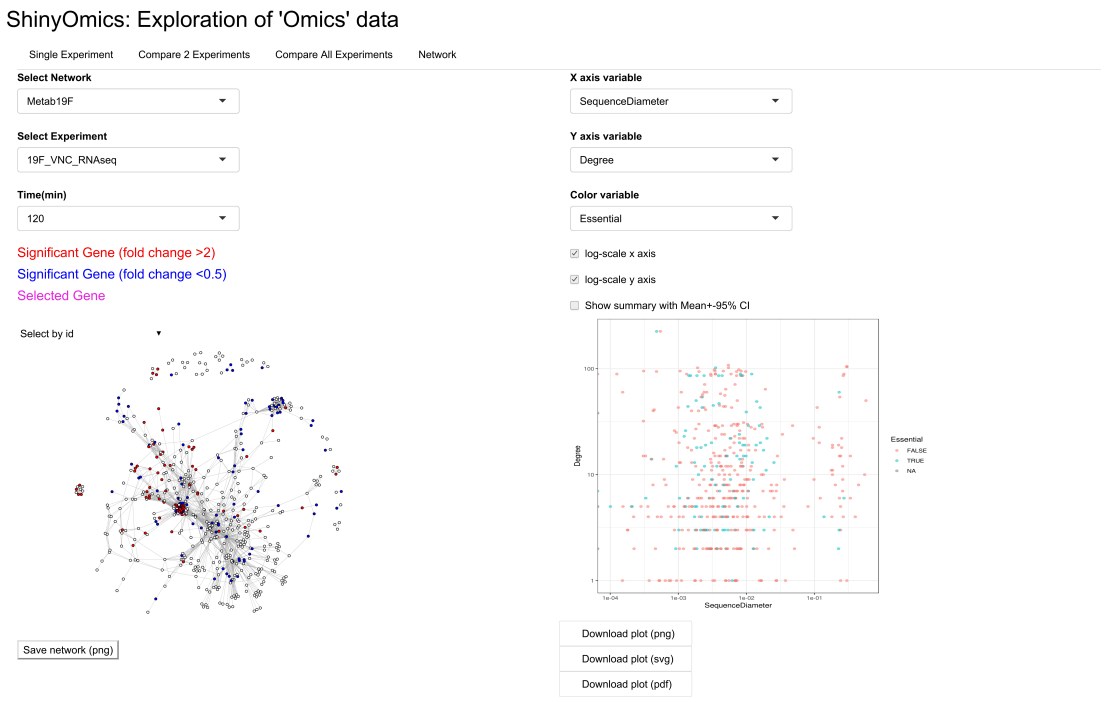

Figure 2.5: The selectors on the upper left allow the user to select a network to display, and a specific experiment and timepoint to overlay. Each gene is a node, and links are defined by the type of network used. The 19F Metabolic ("Metab19F") network has two genes linked, if their gene products participate in the same reaction, or subsequent reactions in the metabolism of 19F. In the Vancomycin experiment shown (at 120 minutes), significantly up- and down-regulated genes appear as red and blue nodes respectively. The selectors on the right help generate a scatter plot (lower right) that can relate network-related information (e.g. network degree) to metadata. In the example plot, degree is plotted against sequence diameter i.e. variability of homologous sequences across different strains of *S. pneumoniae.*

3

Entropy of a bacterial stress response is a generalizable predictor for fitness and antibiotic sensitivity *

3.1 Background

It is generally assumed that in order to overcome a stress, bacteria activate a response such as the stringent response under nutrient deprivation[15,124,34] or the SOS response in the presence of DNA damage[58,8]. Measuring the activation of a specific response, or genes associated with this response, can thereby function as an indicator of what type of stress is occurring in a bacterium. For instance, *lexA*, encoding a master regulator of

*Adapted from Zhu Z, Surujon D, Ortiz-Marquez JC, Huo W, Isberg RR, Bento J, et al. Entropy of a bacterial stress response is a generalizable predictor for fitness and antibiotic sensitivity. Nature Communications. 2020 Aug 31;11(1):4365. Author contributions: TvO devised the study. ZZ, JO, WH, and TvO performed the wet-lab experiments. DS performed the computational experiments. JB contributed to the key conceptual ideas. ZZ, DS, JO, WH, RI, JB, and TvO analyzed the data. ZZ, DS, JO, JB, and TvO. interpreted results and wrote the paper. ZZ, DS, JO, WH, RI, JB, and TvO. approved the final manuscript.

the SOS response in *Escherichia coli* and *Salmonella*[120,214], is upregulated in response to fluoroquinolones, indicative of the DNA damage resulting from this class of antibiotics[214]. Moreover, genes implicated in a stress response can help construct statistical models for predicting growth/fitness outcomes under that stress. For instance, gene-panels have been assembled from transcriptomic data to predict whether a bacterium can successfully grow in the presence of specific antibiotics[14,186,89,19,105]. This type of prediction of growth under antibiotic conditions can lead to point-of-care diagnostics that guide decisions on antibiotic prescription[216].

While methods that are based on a known stress-response or a gene-panel can be valuable in determining a bacterium's sensitivity to a stress, these methods have limited applicability: they only work for small sets of strains, species or environments. For instance, responses such as the stringent or SOS response are only well characterized in a small number of species, genes in a gene-panel may not be present in other strains or species, and responses are not necessarily regulated in the same manner in different strains or species[25,10]. This means that every time such an approach is applied to a new strain, species or condition, a new gene-panel needs to be assembled and validated, which requires the collection of large amounts of data for model training. In contrast, a universal stress response signature would allow for the development of a predictive model that would work for multiple species and conditions, without relying on collecting new data for different settings. While certain organisms may elicit a "general stress response", i.e. regulatory changes coordinated by the same mechanism in response to different types of stress, this general response has not been defined for many species, and it is still not clear to what extent the downstream transcriptional changes triggered under different stress factors overlap[79]. Until this point, there is

no generally agreed upon stress response signature that performs as a fitness predictor, with equal or better performance than the gene-panel approaches.

One possible key ingredient in building a universal predictor is to base a prediction not on specific genes, but rather on a bacterium's global response to stress. A global, genome-wide stress response can be captured on at least two organizational levels; RNA-Seq captures transcriptional changes, while transposon-insertion sequencing (Tn-Seq) characterizes the phenotypic importance of genes, i.e. a gene's contribution to fitness in a specific environment[97,198,201,200,199,202]. We have previously shown that when an organism is challenged with an evolutionarily familiar stress (i.e. one that has been experienced for many generations), it triggers a subtle response, whereas the response becomes more chaotic when the bacterium responds to a relatively unfamiliar stress, for instance antibiotics[97]. This suggests that the degree to which a bacterium is adapted to a specific stress may be predicted from the global response it elicits. It is possible to observe genome-wide differences between stress-susceptible and stress-resistant bacteria in data from previously published transcriptomic studies that mostly focus on gene-panel approaches. Specifically, in these data it can be observed that the number of differentially expressed genes, and the magnitude of changes in expression seem to be more dramatic in stress-susceptible strains than stress-resistant ones[14,186,89,19,105,206]. Therefore, if these are indeed characteristic differences between responses coming from stress-sensitive and stress-resistant bacteria, and these differences can be appropriately quantified, an opportunity would arise to define a universal method that can predict fitness for multiple species and conditions.

In this study we generate and analyze a substantial transcriptomic dataset for the bacterial pathogen *Streptococcus pneumoniae*. To validate our dataset, existing gene-panel approaches

are replicated and scrutinized as a point-of-comparison. Thereby, we first demonstrate that bacterial fitness under antibiotic or nutrient stress can be predicted by expression profiles from small gene-panels, while a separate panel can predict an antibiotic's mechanism of action. We highlight the limitations of these existing approaches by showing that gene-panels are sensitive to model parameters and the data they are trained on, and are limited to strains and species that share the same genes. With the goal to develop a general approach, we explore the observation that global transcriptional disorder seems to be a common stress feature in bacteria. It turns out that increasing disorder stems from an increasing loss of dependencies among genes (e.g. regulatory interactions). These dependencies manifest as correlations in gene expression patterns, and by accounting for these dependencies, the statistical definition of entropy can be used to accurately quantify the amount of disorder in the system. First, we show that when entropy is calculated using time-series RNA-Seq data and dependencies amongst genes are accounted for, stress-sensitive strains have higher entropy than stress-insensitive ones. This enables fitness predictions using a simple decision rule, where if entropy is either above or below a threshold, fitness is respectively low or high. Importantly, this entropy-based method achieves better performance in predicting fitness outcomes compared to existing gene-panel approaches. In order to simplify the approach, we show that entropy can be calculated using a single time-point, and does not necessarily require time-series data to achieve high accuracy. To highlight the universality of entropy, in addition to evaluating performance on a previously unseen test set, validation experiments are performed for 7 Gram-negative and Gram-positive pathogenic species, and the approach is applied to multiple published datasets. Moreover, we show that transcriptional entropy is correlated with the level of antibiotic sensitivity, enabling

MIC predictions. Overall, we develop a large new experimental dataset, and a species-independent fitness prediction method based on entropy. By carefully defining entropy, we illustrate that entropy does not simply capture large changes in expression, but instead builds upon a very intuitive notion of disorder, and enables predictions on bacterial fitness. We present gene-panel based methods as a baseline for comparison, and demonstrate that entropy-based methods perform better, are robust to parameter tuning, and can accommodate different amounts of data to enable fitness predictions. Most importantly, unlike gene-panels entropy-based predictions generalize to previously unseen settings, and to multiple pathogenic bacteria.

3.2 Materials and Methods

3.2.1 Bacterial strains, culture media and growth curve assays

S. pneumoniae strain TIGR4 (T4; NC_003028.3) is a serotype 4 strain originally isolated from a Norwegian patient47,48, Taiwan-19F (19F; NC_012469.1) is a multi-drug resistant strain[168,134] and D39 (NC_008533) is a commonly used serotype 2 strain originally isolated from a patient about 90 years ago[111]. Strain PG1 and PG19 were isolated from adults with pneumococcal bacteremia infection and included in the Pneumococcal Bacteremia Collection Nijmegen (PBCN)[45]. All *S. pneumoniae* gene numbers refer to the T4 genome. *E. coli* strain AR538, *Klebsiella pneumoniae* strain AR497 and *Salmonella enterica* subsp Typhimurium strain AR635 were clinical isolates obtained from the Center of Disease Control (CDC). *Staphylococcus aureus* strain MN6 was kindly provided by George Sakoulas (Center of Immunity, Infection and Inflammation, UCSD School of Medicine). Unless otherwise specified, *S. pneumoniae* strains were cultivated in Todd Hewitt medium with 5% yeast extract (THY)

with 5μL per mL oxyrase (Oxyrase, Inc) or on sheep's blood agar plates (Northeastern Laboratories) at 37^oC with 5% CO_2. *Acinetobacter baumannii*, *E. coli*. *K. pneumoniae*, *S. aureus* and *S*. Typhimurium were cultured in Mueller Hinton broth II (Sigma) at 37^oC with 220rpm constant shaking. RNA-Seq experiments of *S. pneumoniae* under nutrient-depletion and antibiotic conditions were performed in semi-defined minimal medium (SDMM)[97]. RNA-Seq experiments for *A. baumannii*, *S*. Typhimurium, *E. coli*. *K. pneumoniae*, and *S. aureus* were performed in Mueller Hinton broth II. Single strain growth assays were performed three times using 96-well plates by taking OD600 measurements on a Tecan Infinite 200 PRO plate reader.

3.2.2 Temporal RNA-Seq sample collection, preparation and analysis

In nutrient RNA-Seq experiments, T4, D39 and adapted D39 were collected at 30 and 90min after depletion of D39-essential nutrients. In the training set antibiotic RNA-Seq experiments, wild-type and adapted T4 or 19F were collected at 10, 20, 30, 45, 60, 90, 120min post-vancomycin, rifampicin or penicillin treatment. Additional time points at 150, 180, 210 and 240min were collected in levofloxacin and kanamycin experiments due to the slower transcriptional response. In the test set antibiotic RNA-Seq experiments, wild-type T4 and 19F were collected at 30 and 120min post-cefepime, ciprofloxacin, daptomycin or tetracycline treatment. Ciprofloxacin-adapted T4 and 19F were collected at 30 and 120min post-ciprofloxacin treatment. T4 was collected at 30 and 120min post-amoxicillin, ceftriaxone, imipenem, linezolid, moxifloxacin or tobramycin treatment. Wild-type strains were exposed to 1xMIC antibiotics; antibiotic-adapted strains were exposed to 1xMIC (i.e. same concentration as wild-type) and 1.5-2xMIC of the respective antibiotic. Cell pellets were

collected by centrifugation at 4000 rpm at $4^{o}C$ and snap frozen and stored at $-80^{o}C$ until RNA isolation with the RNeasy Mini Kit (Qiagen). 400ng of total RNA from each sample was used for generating cDNA libraries following the RNAtag-Seq protocol[169] as previously described[97]. PCR amplified cDNA libraries were sequenced on an Illumina NextSeq500 generating a high sequencing depth of 7.5 million reads per sample[83]. Raw sequencing data were converted to fastq files using the bcl2fastq software (v2.19, Illumina BaseSpace). RNA-Seq data was processed using an in-house developed analysis pipeline. In brief, raw reads are demultiplexed by 5' and 3' indices[169], trimmed to 59 base pairs, and quality filtered (96% sequence quality>Q14). Filtered reads are mapped to the corresponding reference genomes using bowtie2 (v2.2.6) with the –very-sensitive option (-D 20 –R 3 –N 0 –L 20 –i S, 1, 0.50)[110]. Mapped reads are aggregated by featureCount and differential expression is calculated with DESeq2 (v1.10.1)[118,125]. In each pairwise differential expression comparison, significant differential expression is filtered based on two criteria: $|log2foldchange| > 1$ and $adjustedp - value(padj) < 0.05$. All differential expression comparisons are made between the presence and absence of the antibiotic or nutrient at the same time point. The reproducibility of transcriptomic data was confirmed by an overall high Spearman correlation across biological replicates ($R > 0.95$). Furthermore, the consistent patterns we observe in DE for the training, test and validation experiments, as well as the similarity of DE from experiments using antibiotics with the same MOA, point to the high quality and reproducibility of our dataset. *n.b.* comparison of experiments can be done using the ShinyOmics supplement

3.2.3 Experimental evolution

D39 was used as the parental strain in nutrient-depletion evolution experiments; T4 and 19F were used as parental strains in antibiotic evolution experiments. Four replicate populations were grown in fresh chemically defined medium (CDM) with a decreasing concentration of uracil or L-Valine for nutrient adaptation populations, or an increasing concentration of ciprofloxacin, cefepime, levofloxacin, kanamycin, penicillin, rifampicin, or vancomycin for antibiotic adaptation populations. Four replicate populations were serial passaged in CDM or SDMM as controls to identify background adaptations in nutrient or antibiotic adaptation experiments, respectively. When populations were adapted to their nutrient or antibiotic environment, a single colony was picked from each experiment and checked for its adaptive phenotype by growth curve experiments.

3.2.4 Determination of relative minimal inhibitory concentration (MIC)

$1 - 5 \times 10^5$ CFU of mid-exponential bacteria in 100μL was diluted with 100μL of fresh medium with a single antibiotic to achieve a final concentration gradient of cefepime (T4: 0.008-0.8 μg per mL; 19F: 0.6-2.4 μg per mL), ciprofloxacin (*S. pneumoniae* strains: 0.125-4.0 μg per mL; other species: 0.0125-25 μg per mL), daptomycin (15-55 μg per mL), levofloxacin (0.1-2 μg per mL), kanamycin (35-250 μg per mL), penicillin (T4: 0.02-0.055 μg per mL, 19F: 1-4 μg per mL), rifampicin (0.005-0.04 μg per mL), tetracycline (T4: 4-18 μg per mL; 19F: 19-22 μg per mL); amoxicillin (0.01-0.16μg per mL), imipenem (0.0005-0.045μg per mL), ceftriaxone (0.0005-0.009μg per mL), linezolid (0.05-0.65μg per mL), tobramycin (35-255μg per mL), cotrimoxazole (0.5-7.5μg per mL); moxifloxacin (0.05-0.70μg per mL), and vancomycin (0.1-0.5 μg per mL) in 96-well plates. Each concentration was

tested in triplicate. Growth was monitored on a Tecan Infinite 200 PRO plate reader at $37^{o}C$ for 16 hours. MIC is determined as the lowest concentration that abolishes bacterial growth.

3.2.5 Selection of a gene panel for fitness prediction

Differential expression data from experiments from all experimental timepoints with time≥60min were assembled in R (v3.6.2). The data were split into training and test sets, yielding a training set of 138 and a test set of 19 experiments. Genes with incomplete data (e.g. genes unique to one strain) were omitted. The differential expression data was then scaled such that the values for each gene had mean = 0 and variance = 1. A binomial logistic regression model was fit to the training set with glmnet v3.0-2. In order to determine the appropriate value of the regularization parameter λ, 5-fold crossvalidation was performed on the training set, and mean squared error (MSE) of the crossvalidation set for each of the 5 folds was computed as a measure of classification error. The value of λ was selected to be the largest at which the MSE is within 1 standard deviation of the minimal MSE overall[68,108]. The heatmap of DE for this gene panel was generated using heatmaply (v1.0).

Evaluation of the gene panel's sensitivity to input data was done using another 5-fold crossvalidation strategy, where for each fold, the training portion includes 80% of the original training dataset. The model was fit with the same strategy as above, selecting the best λ for each fold.

Evaluation of the gene panel's sensitivity to λ was done using the standard output of the glmnet function.

For gene panels specific to a single MOA, the training and test sets were filtered to

include only experiments from that MOA. The model fitting procedure was the same for all gene panels that predict fitness. Performance statistics and visualization were done using plotmo (v3.5.6), caret (v6.0-85), PRROC (v1.3.1) and ggplot2(v3.2.1).

3.2.6 PCA and Trajectory clustering

For principal component analysis (PCA), differential expression (log2fold change of +/- antibiotic comparisons) data from all 255 experimental conditions (per time point per antibiotic from all experiments excluding CIP-validation set with *A. baumannii*, *E. coli*, *K. pneumoniae*, *S.* Typhimurium, *S. aureus*, *S. pneumoniae* serotype 1 and 23F strains) were assembled in R (v3.6.2). The function "prcomp" was used for PCA. Timepoints of the same experiment were connected to form trajectories. Since not all experiments are on the exact same time scale (e.g. KAN experiments extend to 240min whereas RIF experiments cover 120min), equivalent timepoints for each experiment were determined to be $(it_max)/6$ for $i = 1, 2, , 6$ and t_max being the latest time point available for the corresponding experiment. If a timepoint did not correspond to an existing RNA-Seq data point, this time point was inferred by linear interpolation of the existing trajectories. To cluster these trajectories, a trajectory-distance metric between two trajectories X and Y is defined as the sum of Euclidean distances ('dist', on the principal component coordinates) $_(i = 1)^6 dist(X_i, Y_i)$ of all timepoints i. All pairwise distances are computed for all pairs of trajectories included in the analysis (WT strains with low fitness, for PSI, DSI, CWSI and RSI). K-means clustering in MATLAB with $K = 4$ is used on the pairwise distances to cluster the trajectories.

3.2.7 Selection of a gene panel for MOA prediction

Differential expression (log2 fold change of drug/no drug comparison) data from all antibiotic experiments with low fitness outcome and time ≥ 60 minutes were assembled in R (v3.6.2). The data were split into training and test sets, yielding a training set of 39 and a test set of 15 experiments. Similar to the fitness gene panel data preparation, genes with incomplete data were omitted. A multinomial logistic regression model was fit to the training set with glmnet v3.0-2. The appropriate value of λ was selected using a similar crossvalidation scheme to the fitness gene panel: the largest λ at which the corssvalidation error is within 1 standard deviation of the minimal error overall. Visualization, and evaluation of the model's performance, sensitivity to input and λ were done as described in the "Selection of gene panel for fitness prediction" section above.

3.2.8 Gene set enrichment analysis

Gene panels for *S. pneumoniae* were evaluated for enrichment of functional categories, using a hypergeometric test, and Benjamini-Hochberg correction for multiple comparisons.

3.2.9 Quantifying entropy of transcriptomic data

Entropy (*H*) for a time-course experiment is defined as in Equation 3.1. The DE data for the timecourse is assembles into a single matrix *S*, where columns are individual genes, and rows are different time points. The covariances across all pairs of columns (i.e. genes) is computed using the 'cov' function in R (v3.6.2) to generate the covariance matrix Σ. Σ is then used as input for the 'glasso' function within the glasso package (v1.10), which generates a regularized covariance matrix (Σ_ϱ). Multiple values of ϱ are scanned between 0

and 5, and for each value of ϱ, the error on the training set was computed. The value of ϱ was determined to be that which minimized error. Using this value of ϱ, multiple values of threshold t were scanned within the range of entropy values within the training set. The value of t was determined to be that which maximized accuracy on the training set.

Entropy of a single timepoint (H_{stp}) is defined as in Equation 3.2. The variance (σ^2) of the whole-transcriptome DE distribution is computed using the 'var' function in R (v3.6.2). The threshold value t was determined by scanning the range of H_{stp} values in the training set, and finding the t that maximized accuracy on this dataset.

The predictive performance of all entropy models was evaluated on both the training and test sets using caret (v6.0-85), PRROC (v1.3.1); and visualized using ggplot2(v3.2.1).

3.3 Results

3.3.1 Existing methods have several limitations, and do not generalize

Previously, the expression levels of specific genes have been used to predict susceptibility of a specific species under a specific antibiotic stress[14,19,206]. In contrast, the goal here is to identify a general predictor of fitness (presence or absence of growth) that does not only work for a specific stress or species, but instead extends to as many previously unseen settings (i.e. species and conditions) as possible. We hypothesized that, in line with existing approaches, a gene-panel that predicts fitness could be generated. This panel, when trained on expression data coming from multiple stress conditions, would then predict bacterial fitness for any condition (rather than a specific condition). Importantly, we would thereby also be able to assess how sensitive such models are to input data and model parameters. Below we first show that gene-panel models indeed are highly sensitive to these factors

and thereby have limited generalizability. Subsequently, we develop an alternative approach using entropy, that is generalizable, robust, and condition-agnostic (i.e. applicable to many conditions).

To test the first hypothesis, whether a gene-panel model can be trained that predicts fitness for many different conditions, a large RNA-Seq dataset was generated for the human pathogen *Streptococcus pneumoniae*. To produce transcriptomic response profiles from multiple stress conditions, *S. pneumoniae* strains TIGR4 (T4) and Taiwan-19F (19F) were grown in the presence or absence of 1x the minimum inhibitory concentration (MIC) of 16 antibiotics representing 4 mechanisms of action (MOA). These include cell wall synthesis inhibitors (CWSI), DNA synthesis inhibitors (DSI), protein synthesis inhibitors (PSI) and RNA synthesis inhibitors ((RSI); Figure 3.1A). Each strain was exposed to each antibiotic for 2 to 4 hours and cells were harvested for RNA-Seq at various time points. As T4 and 19F are susceptible to most antibiotics used, the transcriptional profiles in the presence of antibiotics mostly represent cases of low fitness (Figure 3.1A, sensitive strain, 1xMIC). In order to find patterns that differentiate fitness outcomes, we generated adapted strains with increased fitness in the presence of antibiotics by serial passaging wildtype T4 and 19F in the presence of increasing amounts of antibiotics. Four independent adapted populations for each strain were selected on individual antibiotics. These adapted strains could grow in the presence of antibiotic at 1.5xMIC of the wildtype strain, albeit with a small growth defect. In parallel, RNA-Seq was performed on *S. pneumoniae* strains D39 and T4 in a chemically defined medium, and media from which either uracil, Glycine or L-Valine was removed, which are essential for D39 but not T4. This enabled the potential identification of a common stress signature that is shared between antibiotic exposure and

nutrient deprivation, and across multiple strains. Lastly, D39 was adapted to grow in the absence of each individual nutrient, after which RNA-Seq was repeated for adapted clones (It is possible to visualize and explore these data using a ShinyOmics[185] based app online at http://bioinformatics.bc.edu/shiny/ABX).

Transcriptome data was separated into a training set for parameter fitting, and a test set. The test set includes a completely different set of antibiotic conditions, to enable proper evaluation of model performance on previously unseen data. A condition-agnostic predictor of fitness was developed by fitting a regression model on the training set, which includes high and low fitness outcomes from 5 antibiotics (representing 4 MOAs), 3 nutrient depletion conditions, and from 3 *S. pneumoniae* strain backgrounds. Lasso-regularization was used in order to limit the number of features, thereby lowering the risk of overfitting the model (there are over 1500 genes in common for the 3 strains, therefore there are as many potential features that could be used)[68]. In order to avoid any bias in the selection of features, the regularization strength (λ) was automatically determined using crossvalidation analysis on the training data (Figure 3.1B)[68,108]. The resulting model (which contains 28 genes and an intercept) has an accuracy of 0.93 and 0.77 on the training and the unseen test set respectively (Figure 3.1C, Figure 3.2).

Fitness predictions that rely on the expression of specific genes are potentially influenced by the data used during training[206]. A model robust to input data would recover mostly the same features (i.e. genes) when small subsets of input are omitted during parameter fitting. In order to test the sensitivity of the regression model to input data, the same type of regression model was trained on 5 different subsets of the training dataset, each time omitting a different 20% of the data. The features included and their coefficients

varied greatly in these experiments (Figure 3.1D), with only 5 out of 28 genes in the model common to all iterations of model fitting. To assess sensitivity of the gene-panel to the regularization strength (i.e. λ), the same model was trained using different values for λ. While the coefficients of individual genes vary drastically (Figure 3.1E), the performance at different values of λ remains similar (Figure 3.1B, Figure 3.2B). This indicates that there are genes that contain similar information for classification purposes, and are interchangeable. Thus, we demonstrate that the gene-panel approach is sensitive not only to input data, but also to model parameters. An implication of this sensitivity is that the genes in a gene-panel that are selected in an automatic fashion can be influenced by how the model is trained. Therefore, interpreting these genes as the determinant biological factors for fitness can be problematic. Furthermore, enrichment analysis reveals there are no significantly enriched functional categories in this gene-panel (Figure 3.2E). This suggests that a gene-panel is not a suitable approach for developing a condition-agnostic model, since no specific common response to different stresses can be detected that separates low fitness cases from high fitness ones.

While a condition-agnostic gene-panel is sensitive to input data and model parameter λ, it remains to be seen whether condition-specific models suffer from the same issue as well. For three MOA's for which we generated data for multiple antibiotics (CWSI, DSI, and PSI), regularized regression models were trained, and the models' sensitivities to input data and λ were evaluated. In all 3 cases, the models change with input and λ, and show no enrichment for specific functional categories (Figure 3.3). In contrast, some published gene-panels have shown functional enrichment[19]. However, this is likely because the published gene-panels have been developed for single antibiotics. Therefore, the genes in those panels are highly

selective for the species-specific response that is triggered in a particular stress. In contrast, in this work, we identify predictors that differentiate high and low fitness cases for multiple stresses. The fact that there is no enrichment on our gene-panels is suggestive of a lack of a general response, characterized by a set of specific genes, that gets triggered under many different circumstances.

Besides a lack of functional enrichment, neither the MOA-specific nor the condition-agnostic gene-panels developed here include genes that are known direct-targets of the antibiotics used. Moreover, in addition to being sensitive to input data and regularization strength, the condition-agnostic fitness gene-panel is limited in its applicability to other species, as genes in this panel lack homologs in other Gram-positive as well as Gram-negative species (Figure 3.1F). In fact, this homology problem is a limitation of previously published gene-panels as well (Figure 3.1G). Gene-panel based models therefore not only require re-training for each new condition, but also when they are to be implemented for a new species. This shows that gene-panel approaches in general not only need to be applied and interpreted with caution, but there is also no good evidence to expect that they can be turned into a generalizable fitness predictor that is both species and condition-agnostic.

3.3.2 Gene-level transcriptional responses are unique to the type of stress

We hypothesized that one of the reasons why it may be non-trivial to produce a condition-agnostic model is because the different conditions (i.e. MOA's of different antibiotics) trigger such distinct responses that it is unlikely to identify a common signature among them. To determine whether responses from different antibiotics that fall under the same MOA cluster together, principal component analysis (PCA) was performed on the com-

plete differential expression dataset. Each experiment is presented as one trajectory, connecting individual timepoints within that experiment (Figure 3.4A). *K*-means clustering of all experiments' trajectories showed that transcriptional responses to drugs within the same MOA tend to follow similar trajectories over time (Figure 3.4A, B).

To further analyze whether different MOA's trigger different responses, a multi-class logistic regression model was fit on the training dataset, and evaluated on the test set. If a simple classifier can successfully distinguish between different MOA's, this would imply that there are discriminating signals specific to each MOA. Similar to the fitness prediction, the regularization parameter was selected via a principled automatic procedure (without making any arbitrary decisions) to avoid overfitting (Figure 3.5A). This simple regression model is able to classify MOA's with an accuracy of 1 on the training set, and with only a single misclassification in the test set (Figure 3.4C, Figure 3.5D). Similar to our fitness panel, enrichment analysis of the 34 genes in this MOA panel reveals no significantly enriched functional categories (Figure 3.5E). While some of the genes in the panel are relevant to the action of specific antibiotics, it is not immediately evident how each individual gene is relevant for the classification. For instance, DNA gyrase A (SP_1219) appears in the MOA panel, and is a direct target of fluoroquinolones LVX and CIP, belonging to the class DSI. However, it is downregulated to a higher extent under both RSI compared to DSI stress, and thus does not have much discriminating power on its own (Figure 3.5D). Compared to the fitness prediction panel, the features in the MOA panel are more robust to parameter tuning (Figure 3.5B), and to input data (Figure 3.5C). This suggests that MOA prediction is an easier task than fitness prediction using existing gene-panel approaches. Previous studies have demonstrated it is possible to train a classifier that predicts MOA

from whole transcriptome data[93,24]. However, it was unclear whether MOA could be predicted from the expression of a few genes. Our model could therefore, for instance, be implemented to classify the MOA of novel antimicrobials, without having to profile the entire transcriptome.

3.3.3 Entropy as a measure of transcriptional disorder predicts fitness

While the practical application of the MOA model may be useful, the main goal of this work is to build a versatile toolbox for fitness predictions that does not have many parameters to tune, does not rely on specific genes, and therefore possibly has improved generalizability compared to gene-panel models. To accomplish this, we focused on the following observation that we made in the data presented in this work, as well as in previously published studies[19,105,206,101]: bacteria with low-fitness in a given condition trigger larger, and seemingly more chaotic gene expression changes than those with high fitness (Figure 3.6A, B). Specifically, the temporal response of the wildtype strain with low fitness shows an escalating response over time, with increasing and fluctuating transcriptional changes. In contrast the response of the adapted strain, with high fitness, is contained with only small changes in expression (Figure 3.6A). Since these characteristics can be observed for many different stress-types and species, it could possibly be turned into a generalizable predictor of fitness if appropriately captured. Importantly, these types of patterns in the data evoke statistical entropy, which is a well-established concept that captures the amount of disorder in a system (Figure 3.6B). Figure 3.6B shows 3 hypothetical scenarios. Genes in scenarios 1 and 2 have some sort of regulatory interaction, for instance because they are in the same operon. In scenario 2, the individual genes' expression patterns have differences in magni-

tude and direction, but all genes still have similar overall expression trajectories that co-vary. Therefore, the first 2 scenarios are illustrative of strong dependencies among genes. In contrast, scenario 3 highlights a more disordered pattern, and a lack of dependencies between genes, which results in this scenario's entropy being the highest. We hypothesized that with increasing amounts of stress (i.e. when the fitness of the bacterium is lowered), the bacterium experiences increasing amounts of dysregulation, resulting in a loss of dependencies in expression among genes. A loss of such dependencies results in more and more genes changing in expression independently (and perhaps seemingly randomly), resulting in an increase in entropy. Based on this idea, we aimed to quantify the amount of disorder in a transcriptomic response by computing entropy. To predict fitness, we then use a simple decision rule on a single feature, which avoids overfitting, where entropy higher than a threshold t predicts low fitness, and entropy lower than t predicts high fitness.

To calculate entropy on a transcriptomic dataset with multiple timepoints, we redefine the classical statistical concept of entropy (H) of a multivariate Gaussian distribution as follows:

$$H = ln(|\Sigma_\varrho|) \tag{3.1}$$

Where Σ is the empirical covariance matrix (Σ_ϱ is the empirical covariance of $gene_i$ and $gene_j$ computed from the time series data), and $|\Sigma|$ denotes the determinant of Σ[2,135,28,174]. Σ_ϱ is a graphical-lasso regularized Σ, where ϱ denotes the regularization strength.

Entropy is computed from experiments with multiple timepoints as follows. **1.** The temporal differential expression (DE) data is used to compute a gene-gene empirical covariance matrix Σ. **2.** Graphical lasso[67] is applied to Σ to obtain a regularized inverse of

this covariance matrix (Σ_ϱ^{-1}). The matrix Σ_ϱ^{-1} represents a network of dependencies of the regulatory interactions of the genes. **3.** The inverse of this matrix (Σ_ϱ) can then be used in Equation 3.1 to compute entropy (Figure 3.7).

It is important to note that, with the described approach, a high entropy response reflects large changes in magnitude in the transcriptome that come from independently responding genes. This means that large changes in magnitude can still result in low entropy, when changes in expression are synchronized among genes (Figure 3.6B). Synchronization thus comes from dependencies between genes, for instance due to regulatory interactions, which can vary based on the condition. Here, it is assumed that there is a sparse network of such dependencies (i.e. regulatory interactions), which are specifically determined for each experimental condition. These regulatory interactions for each experiment are inferred by computing a covariance matrix Σ from temporal DE data. The inverse of this covariance matrix (Σ^{-1}) is interpretable as the (condition-specific) regulatory interaction network, where gene pairs have a zero value on Σ^{-1} when their expression patterns are not directly dependent on each other. Like most biological networks, the condition-specific regulatory interaction network is expected to be sparse[191,51,70]. However, raw values on Σ^{-1} empirically measured using RNA-Seq data, are mostly non-zero, resulting in a dense network, potentially due to noise in data collection. Regularization is thereby applied on Σ^{-1} to estimate a de-noised, sparse network of interactions Σ_ϱ , more likely to represent real, biologically relevant regulatory dependencies.

Training of this multi time-point entropy model includes the determination of two parameters: regularization strength ϱ and threshold t. This is accomplished by first determining ϱ by 5-fold crossvalidation (on the training set), and then determining t for this selected

ϱ. ϱ at 1.5 minimizes crossvalidation error (Figure 3.6C), and using this value of ϱ on the full training set, results in a threshold t of 1066.25. This in turn yields an accuracy of 0.97 and 0.84 in the training and test sets respectively. Receiver-operator characteristic (ROC) curve analysis shows that entropy can effectively separate high and low fitness cases, with an area under the ROC (AUROC) curve of 0.99 and 0.91 for the training and test sets respectively (Figure 3.6D). Precision-Recall (PR) curve analysis reveals that entropy can detect high-fitness cases, with an area under the PR curve (AUPRC) of 0.99 and 0.98 for the training and test sets respectively (Figure 3.6E). Both ROC and PR analyses thus show much better performance of entropy compared to the gene-panel on the test set (Table 3.1). Unlike the gene-panel based fitness prediction models, the entropy model is robust to the selection of regularization strength ϱ. It is possible to set ϱ to be an extreme value and still get comparable performance to the model above (Figure 3.8). Here, two such extreme values are considered. For instance, if $\varrho = \infty$ (i.e. the co-variances among genes are ignored and genes' responses are assumed to be independent), entropy can be computed as the average of the logarithm of variances of all genes. In this case, the training and test set accuracies are 0.94 and 0.74 respectively, which is comparable to the fitness gene-panel. If, on the other extreme, $\varrho = 0$, i.e. entropy is computed directly on the non-regularized co-variance matrix, the model will over-correct for a dense network. In this case, the training and test set accuracies are 0.86 and 0.32 respectively. In this case, the poor performance on the test set is likely due entropy being sensitive to the number of experimental timepoints used. The training set (which is used for determining t) contains mostly experiments with 7 timepoints or more, whereas the test set contains experiments with only 2 timepoints For $\varrho = 0$ it appears that the value of t determined on the training set is inappropriate

for the test set. Yet the low-fitness experiments in the test set still have higher entropy than high-fitness experiments. Thus, a lower threshold for entropy could perform better on experiments with fewer timepoints. While the model is sensitive to extreme changes in regularization, this sensitivity is not as severe as the gene-panels, since the extreme value of $\varrho = \infty$ also yields a test set accuracy of 0.74, which is comparable to the gene-panel method with a 0.79 test set accuracy. That said, the entropy-based model operates with highest accuracy when biologically realistic assumptions are made, and ϱ is optimized.

3.3.4 A simpler model of entropy predicts fitness from a single timepoint

The time course experiments accurately capture a bacterium's survival in a test environment, but they are labor intensive and potentially expensive. In cases where temporal information may not be available or is prohibitively expensive to generate, computing covariance across genes is not possible. However, entropy can still be determined for a single-timepoint transcriptome profile as follows[112]:

$$H_{stp} = ln(\sigma^2) \tag{3.2}$$

Where σ^2 is the variance of the distribution of differential expression across genes for a single timepoint (Figure 3.9A, B). This simpler definition of entropy enables the approach to be applied even in settings where temporal transcriptional information cannot be obtained. Similar to the temporal models, a threshold t for entropy was determined automatically (in this case $t = 2.08$), which is the value that maximizes classification accuracy in the training set which contains data from multiple timepoints. Analogous to the temporal models, low fitness is associated with higher entropy compared to high fitness

conditions (Figure 3.9). The single-timepoint variant of entropy outperforms gene-panels: on the test set, the area under ROC curve is 0.88 for entropy, and 0.75 for the gene-panel (Figure 3.9D). Similarly, for the test set, the area under the PR curve is 0.96 for entropy, whereas for the gene-panel, it is 0.32 (Figure 3.9E). Moreover, the single timepoint variant of entropy can classify low and high fitness cases with an accuracy of 0.81 and 0.61 in the training and previously unseen test sets respectively (Figure 3.9F). However, our data shows that different antibiotics trigger responses in a time-dependent manner, which may lead to ambiguities in the entropy-based prediction of fitness for early timepoints for antibiotics that cause a slower response (e.g. KAN, Figure 3.9C). Therefore, predictions based on (slightly) later timepoints might result in improved accuracy. To test this, the training and test datasets were split into early (≤45 minutes of stress exposure) and late (≥60 minutes of exposure) timepoints. Two new thresholds for entropy were determined: $t_{early} = 0.94$ on the early timepoints and $t_{late} = 2.11$ on the late timepoints within the training data. On the early timepoints, t_{early} achieves an accuracy of 0.75 and 0.63 on the training and test sets respectively. On the later timepoints, t_{late} yields a high accuracy of 0.88 and 0.84 on the training and test datasets, only including 3 false positive predictions in the test data set (Figure 3.9G). This shows that entropy computed on data from later time points results in a higher predictive accuracy of fitness outcome than earlier time points (Figure 3.9G). Biologically this also makes sense, because while only some antibiotics trigger a clear response within 30-60 minutes after exposure, all antibiotics trigger an increasingly pronounced response as exposure times progress past 60 minutes. The time dependency of an antibiotic response thus makes it more difficult to accurately predict fitness using data from early timepoints. This time dependency would affect the gene-panel for fitness predictions as well. Even

though the gene-panel is trained and tested on only the later timepoints and has far poorer performance compared to entropy trained and tested on the same (late) timepoints. Moreover, entropy trained on early timepoints does only slightly worse than gene-panels trained on late timepoints, with only 3 additional misclassifications. This highlights that despite the time dependency of an antibiotic response, our new entropy-based approach can make predictions on at least two time frames, unlike gene-panels.

Overall, the entropy model (and its variants) has several advantages. First, it is based on a simple, and intuitive principle: large and independent changes in the transcriptome are indicative of dysregulation, and beyond a threshold predictive of low fitness. Second, it is possible to simplify the entropy-based model to accommodate less data (i.e. single timepoint transcriptome). Third, an entropy-based model has few parameters (at most 2 parameters need to be determined), and is therefore less likely to be overfit to data. Fourth, the model does not depend on the identity of specific genes, who may or may not be present in different strains/species. Fifth, the model could be easily applied to other data types (e.g. proteomics, metabolomics). Therefore, an entropy-based model is more likely than a gene-panel based approach to be generalizable to previously unseen conditions and species.

3.3.5 Entropy-based predictions generalize across species and conditions

To test if the entropy-based approach is indeed generalizable and successfully predicts fitness for other *S. pneumoniae* strains and other species, a new RNA-Seq dataset was generated under CIP exposure for *S.* Typhimurium, *S. aureus*, *E. coli*, *K. pneumoniae* and two additional *S. pneumoniae* strains representing serotypes 1 and 23F. These five species represent both Gram-negative and Gram-positive bacteria and cover a wide range of CIP MICs (Figure

3.10A). Since the single-timepoint variant of entropy is the most practical (in terms of data collection and cost), the generalizability of entropy to previously unseen species was evaluated using this model. RNA-Seq was performed at 120 minutes post exposure to 1μg per mL of CIP. The overall response characteristics are similar to what was observed for *S. pneumoniae*, with 120 minutes exposure to 1μg per mL CIP triggering expression changes with higher variance from bacterial cultures having low fitness (*S.* Typhimurium and *S. pneumoniae* serotype 1), compared to those with high fitness (*S. pneumoniae* serotype 23F, *E. coli* and *K. pneumoniae*) (Figure 3.10B). Single-timepoint entropy was computed for the transcriptome of each of these previously unseen isolates. Importantly, with the original threshold of 2.08, which was determined during model training with data from *S. pneumoniae* in Figure 3.10, fitness outcomes could be predicted for the new organisms with 100% accuracy, indicating that the single-timepoint entropy measure, which uses the least amount of data compared to other variants of entropy, is a species-independent generalizable feature for fitness outcome.

Furthermore, the entropy measurement of each strain was found to be inversely proportional to the CIP MIC (Figure 3.10C), consistent with transcriptional disruption being proportional to stress sensitivity. The correlation between entropy and CIP sensitivity in Figure 3.10C (left panel) therefore implies that the antibiotic sensitivity of other species could be predicted from its transcriptomic entropy. To test this, entropy was calculated for *A. baumannii* isolates that are either low (ATCC 17978) or high (LAC-4) virulence, by collecting RNA-Seq profiles after 120 minute exposure to 1 μg per mL of CIP. Using a linear regression model, the CIP MICs of the *A. baumannii* strains were predicted to be 0.04 and 10.45μg per mL, which are proximate to the measured MIC's of 0.07 and 8.5μg per mL for

ATCC 17978 and LAC-4, respectively (Figure 3.10D). This demonstrates that entropy is not simply a binary indicator of fitness outcomes. Even when using a single timepoint i.e. the least amount of transcriptomic information, entropy can be applied to determine the antibiotic sensitivity level for new unseen species that were not in any training data.

To further validate the approach, data from Bhattacharyya *et al.*[19] was used. In this RNA-Seq dataset, susceptible and resistant strains from 3 species were exposed to 3 different antibiotics (2 of which were not present in our dataset). Again, by using the entropy threshold of 2.08 (obtained above through training on the *S. pneumoniae* data) susceptible strains with low fitness are successfully separated from resistant strains with high fitness (Figure 3.10E).

Finally, to explore the applicability of entropy beyond nutrient and antibiotic stress, entropy-based fitness classification was performed on a published collection of 193 *Mycobacterium tuberculosis* transcription factor over-expression (TFOE) strains[154]. Upon TFOE, these strains exhibit fitness changes, ranging from severe growth defects to small growth advantages[128]. Over-expression of a single transcription factor can thereby exert stress on the bacterium that can result in different fitness outcomes. By calculating entropy from whole-genome microarray data collected from each TFOE strain, it is possible to distinguish strains based on their fitness levels at an accuracy of 0.78, using a newly trained entropy threshold for this dataset (Figure 3.10F). This result compares favorably with a much more complicated approach involving the integration of each TFOE transcriptional profile into condition-specific metabolic models[154]. Overall, these data clearly highlight the strength of entropy, which has the potential to be utilized as a generalizable fitness prediction method for both antibiotic and non-antibiotic stress, and a large variety of bacterial

strains and species.

3.4 Discussion

A major goal of this work is to determine if there is a quantifiable feature that can accurately predict bacterial fitness in an environment, independent of strain, species or the type of stress. To be generalizable, the selected feature needs to be common across species and environments. By generating a large experimental dataset and analyzing published ones, we show that such a feature exists, namely transcriptomic entropy, which quantifies the level of transcriptional disorder while a bacterium is responding to the environment. It is important to realize that entropy is not simply a measure of large magnitude changes in the transcriptome. Instead, entropy takes into account condition-specific transcriptional dependencies among genes, and quantifies the amount of independent changes. The underlying assumption is that gene expression patterns lose underlying dependencies and become more stochastic with increasing amounts of stress. The difference between simple measures of magnitude changes and more controlled measures of entropy is illustrated in Figure 3.6B. We show that entropy is a flexible, and generalizable predictor of bacterial fitness in a variety of different environments, it can be used with time-course data or single-timepoint data, and can even be used to predict the MIC of an antibiotic. This study demonstrates how entropy-based predictive models can be implemented in several ways, by using different amounts of data, resulting in different types of predictions. Even using a single timepoint, it is possible to predict both fitness as a binary outcome, as well as the MIC of an antibiotic (Figure 3.10D), highlighting entropy as a very flexible framework that can be adapted to different settings.

We use current gene-panel based approaches for two reasons: 1) To search for a gene-panel that would capture a general stress-response (if it exists), and thus would represent a set of genes and associated regulatory changes coordinated by the same mechanisms in response to different types of stress. The existence of such a general response has been mostly connected to the manner in which *rpoS* responds to stress in *E. coli* and a small number of other species. However, it is largely unclear which genes respond downstream of *rpoS*, whether this response is accompanied by stress-specific responses, to what extent these transcriptional changes overlap across species and in response to different types of stress[79]. Moreover, if such a general stress response exists widely across species, it is unclear whether there is any predictive information to be extracted from it. Importantly, we were unable to identify such a gene-panel within the dataset we generated for *S. pneumoniae* and other species, as well as in the published datasets we explored; 2) As a point of comparison for our entropy-based approach. This comparison highlights that an entropy-based approach yields better performance than a gene-panel based approach, and has at least 3 additional advantages over existing gene-panel approaches: a) It is independent of specific genes, whereas gene-panels focus entirely on specific genes. This might lead researchers to interpret genes present in a particular panel as those most relevant to the stress response. However, caution should be taken in the interpretation of these gene panels, because it turns out that the genes that appear in these panels are strongly influenced by model parameters (λ) and input data (Figure 3.1). b) An entropy-based method has few (at most 2) parameters, and therefore does not risk overfitting (unlike gene-based approaches, where there is at least one parameter per each transcriptionally measured gene). c) The entropy method generalizes across different antibiotic and non-antibiotic conditions, and across

different species. This is not the case for gene-panel based methods, which can only make predictions on the same conditions as the data they were trained on (i.e. one model is predictive for a specific species and a specific antibiotic). And even though a gene-panel may only use expression of a limited number of genes to predict fitness, and may therefore seem to be relatively easy to implement in a clinical setting, each new antibiotic-species combination requires the collection of an entirely new training dataset. This makes gene-panel approaches costly. Although in this paper we focus mainly on accuracy of fitness predictions, there are additional biological insights to be gleaned from the data presented in this work. For instance, the inverse covariance matrix from Equation (1) represents a network that reveals regulatory interactions among genes. The covariance network inference using graphical-lasso regularization presented here is to the best of our knowledge an improvement upon other methods (e.g. WGCNA[217]), which will be explored in depth in future work. Thus, it is possible that the networks generated in this work will be applicable in other ways, e.g. in the identification of novel regulators, their targets, or the prediction of transcriptional changes that follow a perturbation.

By demonstrating the feasibility of predictions of fitness outcomes and antibiotic sensitivity, we envision several possibilities of integrating entropy-based predictions in a clinical diagnostic setting. Currently, AST is often performed using culture-based methods. These methods may take days and even weeks for slow-growing species such as *M. tuberculosis*[96], delaying diagnosis and treatment in clinical settings. Therefore, it is desirable to be able to predict the fitness outcome of such slow-growing species as early as possible, for instance using RNA expression data. Another potential application of our entropy-based fitness predictions is monitoring an active infection *in vivo*. Performing transcriptome profiling

and predicting the fitness of the infectious agent directly in its host environment would allow for monitoring of disease progression, and determining if and when treatment is necessary. Simultaneously profiling the pathogen and the host using dual RNA-Seq[212,6], and predicting the fitness of both could also be valuable in assessing the state and progression of an infection.

Admittedly, direct implementation of RNA-Seq in diagnostic tests might not (yet) be practical, as RNA-Seq experiments still remain relatively expensive, labor-intensive and time-consuming. In particular, time-course experiments such as those included in this study increase in cost linearly with an increasing number of time points. However, the advances in technology are likely to reduce cost much more drastically than a linear model, as is observed for many sequencing approaches. To implement temporal entropy, it is important to recognize that more timepoints will yield better results. However, even 2 timepoints gives robust results. The most economic approach would clearly be the single-timepoint model, which has comparable performance to the temporal models, with the only disadvantage that it lacks possible insights that could be gleaned from the covariance networks temporal entropy is based on. With the advent of real-time sequencing technologies, such as Oxford Nanopore Technologies, the speed of data collection may soon be improved significantly. Additionally, a transcriptome can be sub sampled by monitoring conserved genes across species. In this scenario, transcriptional entropy can be obtained via more economical gene expression technologies, such as NanoString nCounter[73] or the Luminex platform[55]. To conclude, we present an approach that uses entropy to predicting fitness independently of gene-identity, gene-function, and type of stress. This approach can be applied as a fundamental building block for generalizable predictors of fitness and MICs for Gram-

positive and negative species alike, and thereby possibly improve clinical decision-making.

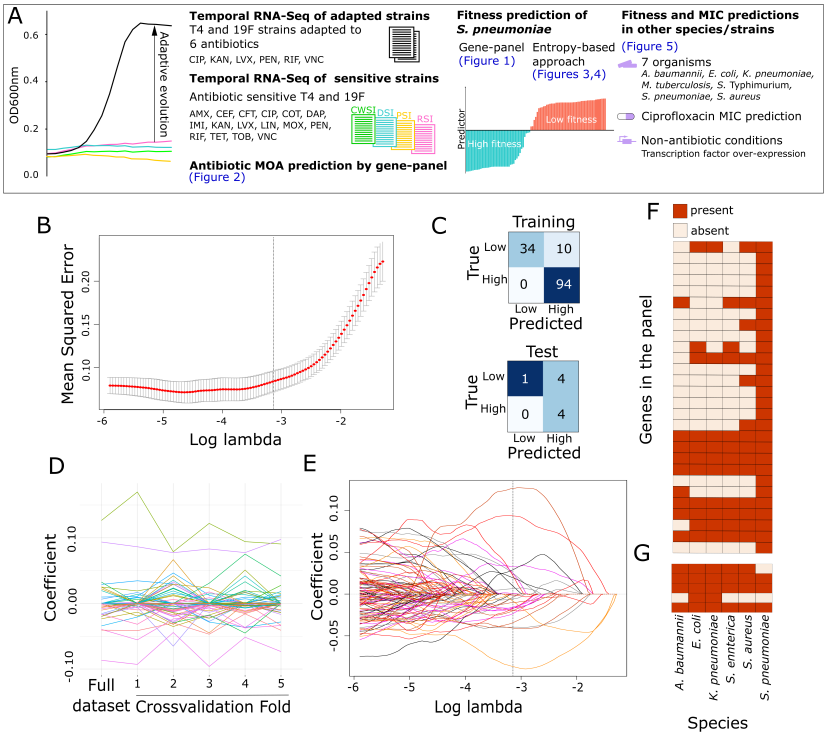

Figure 3.1: (A) Project setup and overview. Wildtype and adapted strains of *S. pneumoniae* are exposed to multiple antibiotics, belonging to 4 different classes, and their fitness outcomes in each condition is determined by growth curves. Temporal RNA-Seq data is used to train models that predict the MOA of an antibiotic, and the fitness outcome of a strain using gene-panel approaches. The concept of entropy is developed expanding predictions to MIC and fitness for other strains and species in the presence of antibiotics and in non-antibiotic conditions. (B) A gene-panel for fitness prediction is generated by a regularized logistic regression model fit on differential expression data from the training set. The selected value of λ = 0.0428 is shown as a dashed line, resulting in 28 genes in this panel. Red points and error bars represent mean ± standard deviation of error across $n = 5$ crossvalidation folds. (C) Prediction performance of the fitness gene-panel is shown as confusion matrices for the training (top) and test (bottom) datasets. The gene-panel generates 10 and 4 false positives, and an overall accuracy of 0.93 and 0.77 in the training and test data sets respectively. (D) Coefficients of individual features (i.e. genes) are plotted for the model trained on the full dataset, and 5 crossvalidation training folds, where 20% of the data is omitted during model fitting. The gene-panel is highly affected by training data, indicated by many genes having nonzero coefficients on some folds, but not others. Only 5 out of the 28 genes in the fitness gene-panel are maintained as predictors in the regression models across all folds. (E) Each gene's coefficient is plotted as an individual line, against varying values of λ. The gene panel is highly affected by λ, indicated by the nonmonotonic increase or decrease in the coefficient in each gene. In fact, there are many genes that have nonzero coefficients only for a small range of λ. Dashed line depicts the selected value of λ as in B. (F) The presence and absence of each of the 28 genes in the *S. pneumoniae* fitness panel is highly variable across 5 Gram-positive and Gram-negative species. (G) A published *E. coli* CIP sensitivity panel11 also suffers from a lack of conservation across the same group of species.

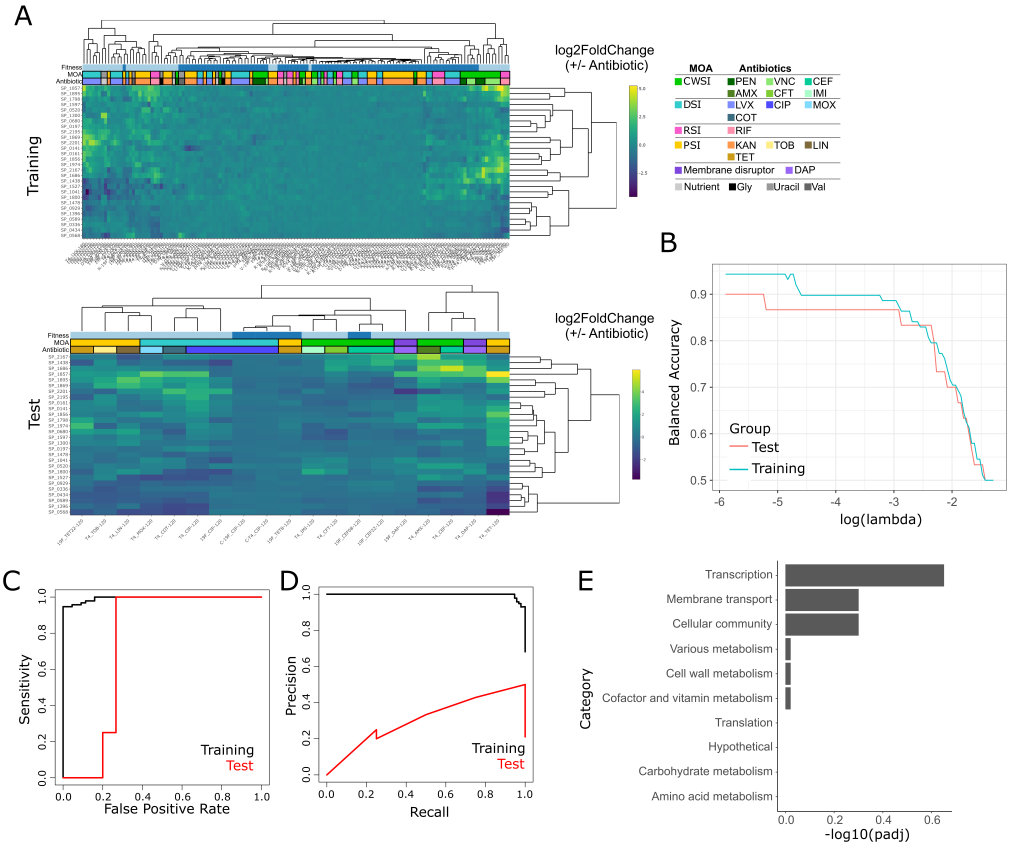

Figure 3.2: (A)Heatmaps show the differential expression (log2FoldChange) of each gene in the panel in each of the 19 stress conditions. Each row is a gene in the panel, and each column is a different experiment (experimental timepoints are separate columns). Top: Training set data, Bottom: Test set data. The top bar above the heatmap shows the observed fitness outcome (light blue: low fitness, dark blue: high fitness). The middle and bottom bars above the heatmaps indicate the MOA and identity of the stress respectively. Dendrograms on the top and side of the heatmaps show hierarchical clustering of the columns and rows respectively. (B) Balanced accuracy of the regression model is similar for training and test sets at different values of λ. For λ < 0.05, both train and test set accuracies are > 0.85, despite the models selecting different sets of genes (Figure 3.1E). (C) Receiver-operator characteristic (ROC) curve for the fitness gene-panel. The area under the curve is 0.99 and 0.75 for the training and test sets respectively. (D) Precision-Recall (PR) curve for the fitness gene-panel. The area under the curve is 0.99 and 0.31 for the training and test sets respectively. (E) No functional category is enriched in the fitness gene-panel. For each category present in the gene-panel, a hypergeometric test was performed, and the resulting p-value is adjusted for false discoveries (padj). No category had padj<0.01.

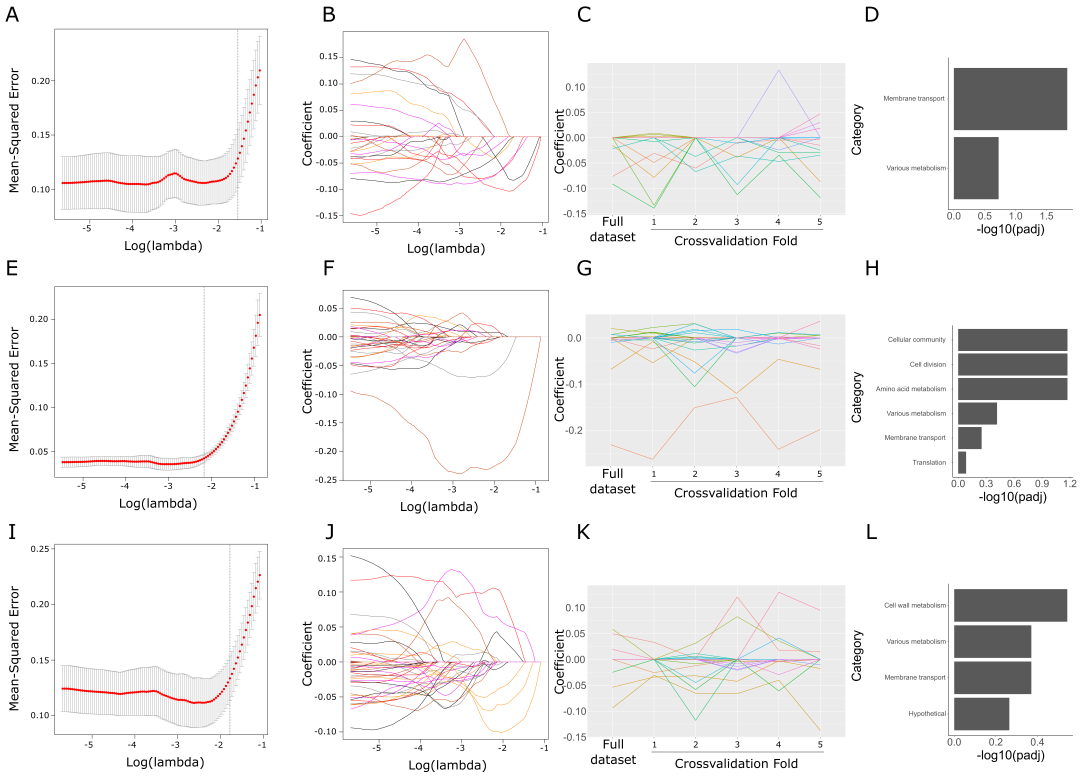

Figure 3.3: (A-D) CWSI-specific panel. (E-H) DSI-specific panel. (I-L) PSI-specific panel. A, E, I show the crossvalidation analysis that determine the value of lambda (as in Figure 3.1B). The selected lambda is shown as the dashed line. Red points and error bars represent mean ±standard deviation of error across $n = 5$ crossvalidation folds. B, F, J show the coefficients of each gene changing depending on lambda. C, G, K show the coefficients of each gene changing with different input data used. Full dataset: coefficients obtained when the regression model is trained on all available training data for a specific MOA. Crossvalidation fold: coefficients obtained when the model is trained on 80% of the available training data. D, H, L show enrichment analysis of each gene-panel predicting fitness specific to CWSI, DSI, PSI respectively (similar to Figure 3.2E). There are no functional categories with padj<0.01.

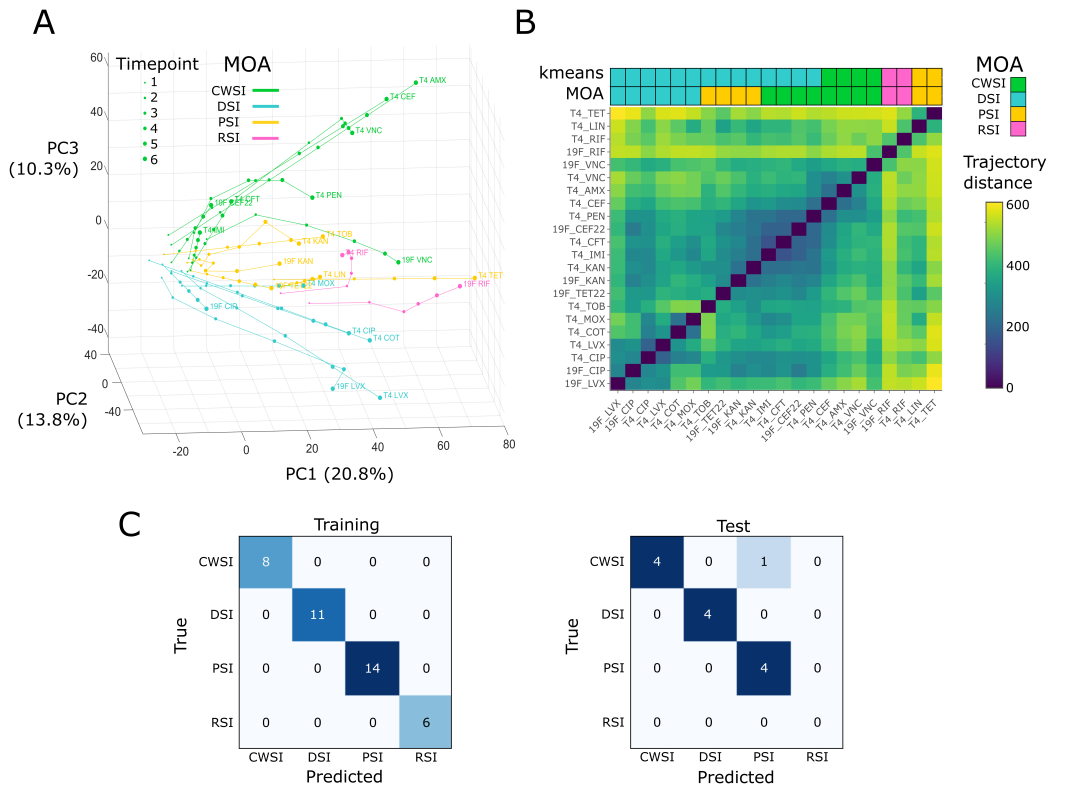

Figure 3.4: (A) Principal component analysis (PCA) on differential expression datasets from sensitive *S. pneumoniae* strains T4 and 19F grown in the presence of 16 different antibiotics at 1xMIC depicts antibiotic responses as temporal transcriptional trajectories. Each line describes the trajectory of one of one strain in the presence of a CWSI (AMX, CEF, CFT, IMI, PEN, VNC), DSI (CIP, COT, LVX, MOX) PSI (KAN, LIN, TET, TOB) or RSI (RIF). Trajectories for each strain are largely grouped based on their MOA, and grouped-trajectories become more distinct over time. The size of each data point increases with the time of antibiotic exposure; each trajectory is split into 6 timepoints, e.g. for an experiment that spans 120' each point indicates a 20' increment. Abbreviations are as in Figure 3.1. (B) In order to quantify the separation of the PCA trajectories by an antibiotic's MOA, pairwise distances between PCA trajectories were computed (see Methods). Pairs of transcriptional trajectories obtained using drugs within the same MOA tend to have smaller distances than pairs obtained using drugs with different MOA's. *K*-means clustering of the trajectory distances groups the trajectories mostly by MOA, although some PSI and CWSI trajectories are grouped with DSI ones. The top and bottom bars above the heatmap show the *K*-means clustering result, and the real MOA of each trajectory respectively, which have 64% agreement. (C) Confusion matrices indicating the performance of the gene-panel that predicts MOA. This panel was generated using a multi-class regression model and consists of 34 genes. The gene-panel correctly predicts the MOA on all training set data and only misclassifies a single experiment on the previously unseen test dataset, showing the different MOA's being easily distinguishable with simple gene-based methods.

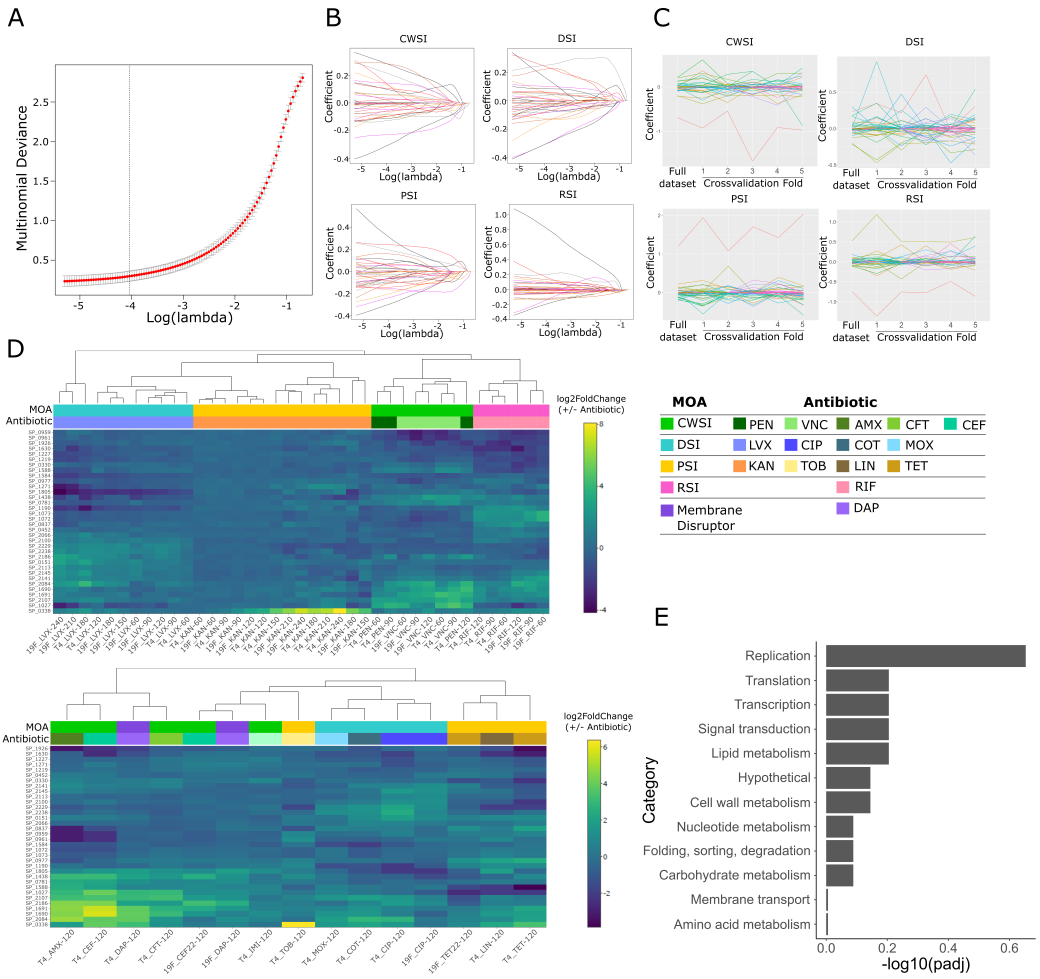

Figure 3.5: (A) Crossvalidation analysis was applied to determine the best value of lambda on the multi-class regression model that predicts MOA. Unlike the 2-class models, error is evaluated as multinomial deviance. Otherwise, lambda is determined the same way as in Figure 3.1B. Red points and error bars represent mean ±standard deviation of error across $n = 5$ crossvalidation folds. (B) Coefficients of each gene for each class (i.e. MOA, shown as separate sub-panels) change monotonically as lambda is decreased. This is indicative of the genes being more consistent than the gene-panels that predict fitness (Figure 3.1E, Figure 3.3B, F, J). (C) Coefficients of each gene, for each class (sub-panels) are affected by input data. Analysis similar to that done in Figure 3.1D and Figure 3.3C, G, K. (D) Heatmaps show differential expression (log2FoldChange) of each gene in the MOA gene-panel (rows) in each experimental condition (columns). The bars directly above the heatmaps show the MOA and the antibiotic. Top panel: training set. Bottom panel: test set. Dendrograms above heatmaps show hierarchical clustering of the experiments. (E) Functional category enrichment analysis was done similarly to Figure 3.2E. There are no categories with padj<0.01.

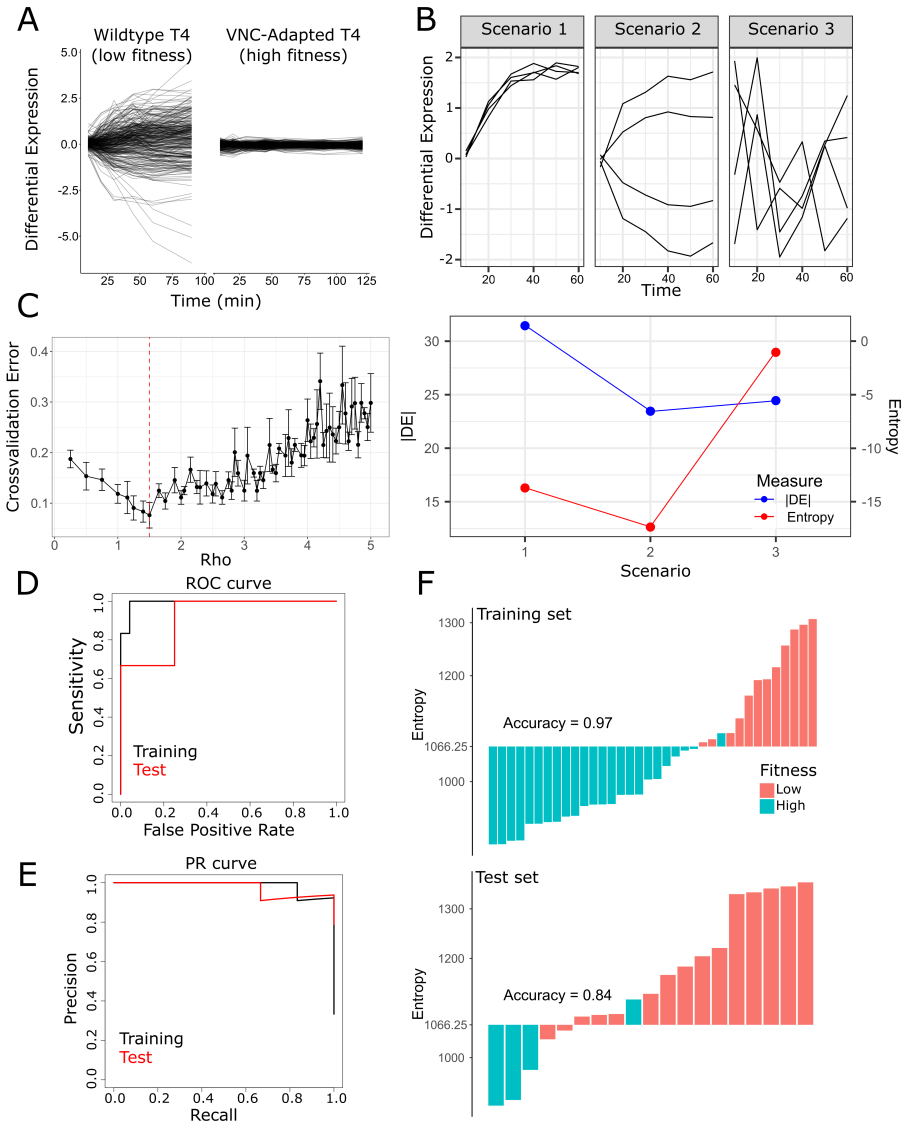
A
Wildtype T4
(low fitness)
VNC-Adapted T4
(high fitness)
Differential Expression
Time (min)
B
Scenario 1
Scenario 2
Scenario 3
Differential Expression
Time
C
Crossvalidation Error
Rho
|DE|
Entropy
Measure
|DE|
Entropy
Scenario
D
ROC curve
Sensitivity
False Positive Rate
Training
Test
E
PR curve
Precision
Recall
Training
Test
F
Training set
Entropy
Accuracy = 0.97
Fitness
Low
High
Test set
Entropy
Accuracy = 0.84

Figure 3.6: (A) Depiction of the transcriptomic response of wildtype T4 and VNC-adapted T4 in response to 1xMIC-wt of VNC. Differential expression (DE) of each gene over time is represented as a line. The response of the wildtype is more disordered than the adapted-response, and has higher entropy. (B) Entropy captures disorder in a transcriptome and not simply high-magnitude changes. The top panel shows 3 hypothetical scenarios, where DE of four individual genes are tracked over time. In scenarios 1 and 2, the individual genes are dependent on each other and follow similar transcriptional trajectories. In scenario 3, dependencies are largely absent and the overall changes in DE seem much more disordered. In the bottom panel, magnitude changes (blue, quantified as the sum of absolute DE), and entropy (red) for the 3 scenarios are compared. While the largest changes in magnitude are in scenario 1, both scenario 1 and 2 have relatively low entropy, due to dependencies among genes. In scenario 3 overall DE is similar to the other two scenarios, but the magnitude changes have lost much of their dependency and have become disordered, resulting in high entropy. (C) Selection of regularization parameter ϱ. 5-fold crossvalidation was used to determine the best choice of ϱ. Error (1-accuracy) is reported as the mean ± standard deviation across 5 folds. The value of ϱ that minimizes the mean crossvalidation error is determined to be 1.5 (red dashed line). (D) Performance of temporal entropy-based fitness prediction is shown as receiver-operator characteristic (ROC) curves plotting the sensitivity against the false positive rate across a range of thresholds for training (black) and test (red) datasets. The area under the ROC (AUROC) curve shows how well the predictor can separate high and low fitness. The AUROC is 0.89 and 0.94 for the training and test set respectively. (E) Performance of temporal entropy-based fitness prediction is shown as Precision-Recall (PR) curves plotting precision against recall across a range of thresholds for training (black) and test (red) datasets. The area under the PR curve (AUPRC) shows how well the predictor can detect high fitness cases. The AUPRC is 0.88 and 0.98 for the training and test set respectively. (F) Entropy of all experiments in the training (top panel) and test (bottom panel) sets. Each experiment is represented as an individual bar, colored according to the experimentally determined fitness outcome. Bars above the entropy threshold (Entropy = 1066.25) are predicted to be low fitness and bars below the threshold are predicted to be high fitness. Both training and test sets score very well with an accuracy of 0.97 and 0.84 respectively.

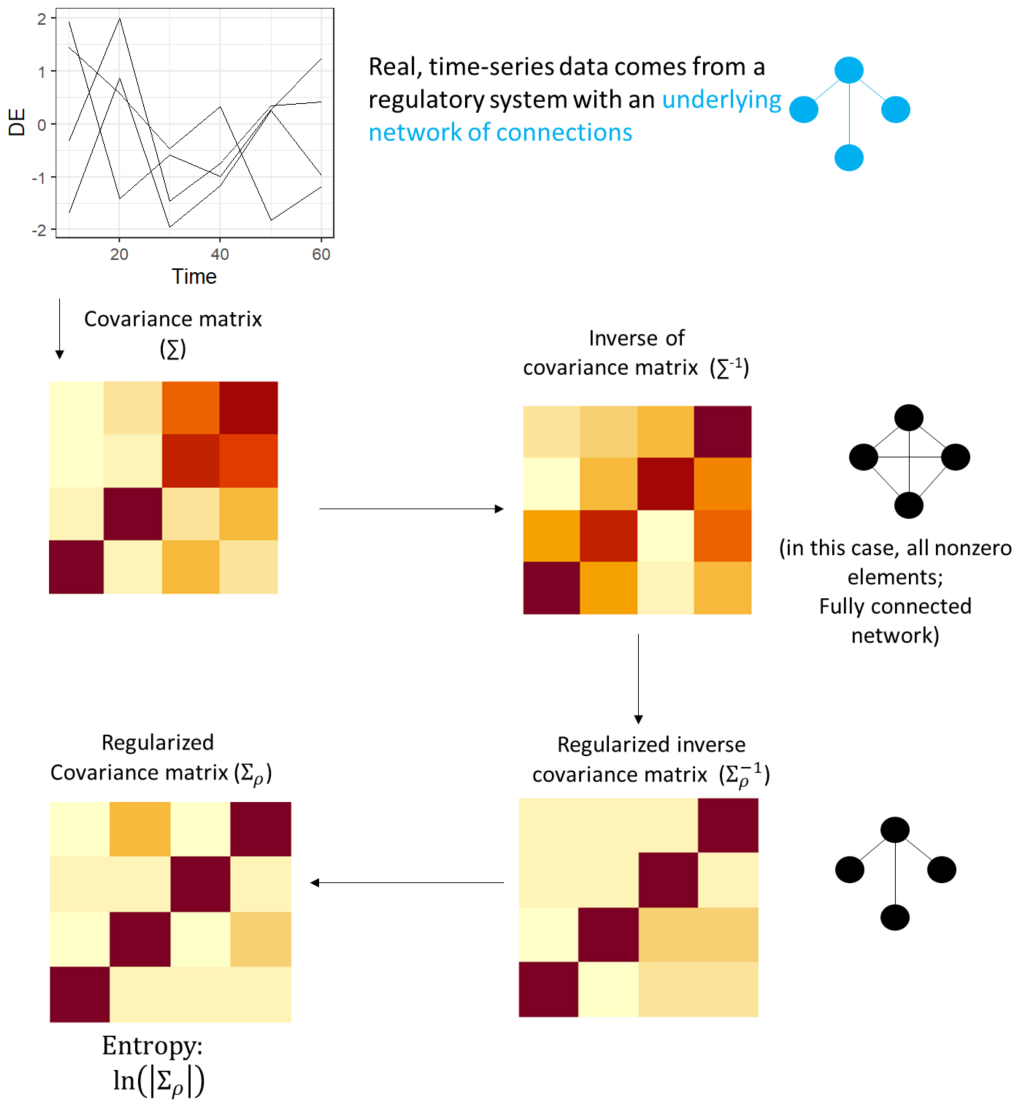
DE
2
1
0
-1
-2
20
40
60
Time
Real, time-series data comes from a regulatory system with an underlying network of connections
Covariance matrix (Σ)
Inverse of covariance matrix (Σ^{-1})
(in this case, all nonzero elements; Fully connected network)
Regularized inverse covariance matrix (Σ_ρ^{-1})
Regularized Covariance matrix (Σ_ρ)
Entropy: $\ln(|\Sigma_\rho|)$

Figure 3.7: The observable DE patterns are assumed to be influenced by condition-specific networks of interactions among genes. These interactions are unknown, but can be inferred from the covariances among genes. Entropy quantifies disorder on a transcriptome, taking into account these interactions. In order to achieve this, we first compute the covariance matrix (Σ) across genes, and take its inverse (Σ^{-1}). The support of this inverse covariance matrix yields a dense network, which is the "uncorrected" version of the real coexpression network. Since the real network is assumed to be sparse, we apply graphical lasso to retrieve a sparse network Σ_{ϱ}^{-1}, and invert the resulting matrix (Σ_{ϱ}). Entropy is defined as the logarithm of the determinant of this matrix.

Model	Group	Cohen's Kappa	AUROC	AUPRC	Sensitivity	Specificity	Balanced Accuracy	F1
Gene Panel	Training	0.8224	0.993956	0.997277	1	0.772727	0.886364	0.949495
Gene Panel	Test	0.5366	0.75	0.314235	1	0.733333	0.866667	0.666667
Temporal entropy (rho = ∞)	Training	0.875	0.982639	0.967558	0.958333	0.916667	0.9375	0.958333
Temporal entropy (rho = ∞)	Test	0.4571	0.933333	0.980821	1	0.666667	0.833333	0.615385
Temporal entropy (rho = 0)	Training	0.6512	0.913194	0.862189	1	0.583333	0.791667	0.90566
Temporal entropy (rho = 0)	Test	0.0608	0.85	0.944888	1	0.133333	0.566667	0.380952
Temporal entropy (rho = 1.5)	Training	0.9388	0.993056	0.986079	0.958333	1	0.979167	0.978723
Temporal entropy (rho = 1.5)	Test	0.5649	0.916667	0.97502	0.75	0.866667	0.808333	0.666667
Entropy (single timepoint)	Training	0.5417	0.790149	0.771126	0.97351	0.5125	0.743005	0.872404
Entropy (single timepoint)	Test	0.2963	0.875	0.963406	1	0.5	0.75	0.516129
Entropy (single timepoint; early)	Training	0.4291	0.70516	0.709462	0.947368	0.444444	0.695906	0.824427
Entropy (single timepoint; early)	Test	0.24	0.8	0.90758	0.75	0.6	0.675	0.461538
Entropy (single timepoint; late)	Training	0.7001	0.897727	0.861196	0.957447	0.704545	0.830996	0.913706
Entropy (single timepoint; late)	Test	0.6275	1	1	1	0.8	0.9	0.727273

Table 3.1: Model: name of the model. Group: training or test set.

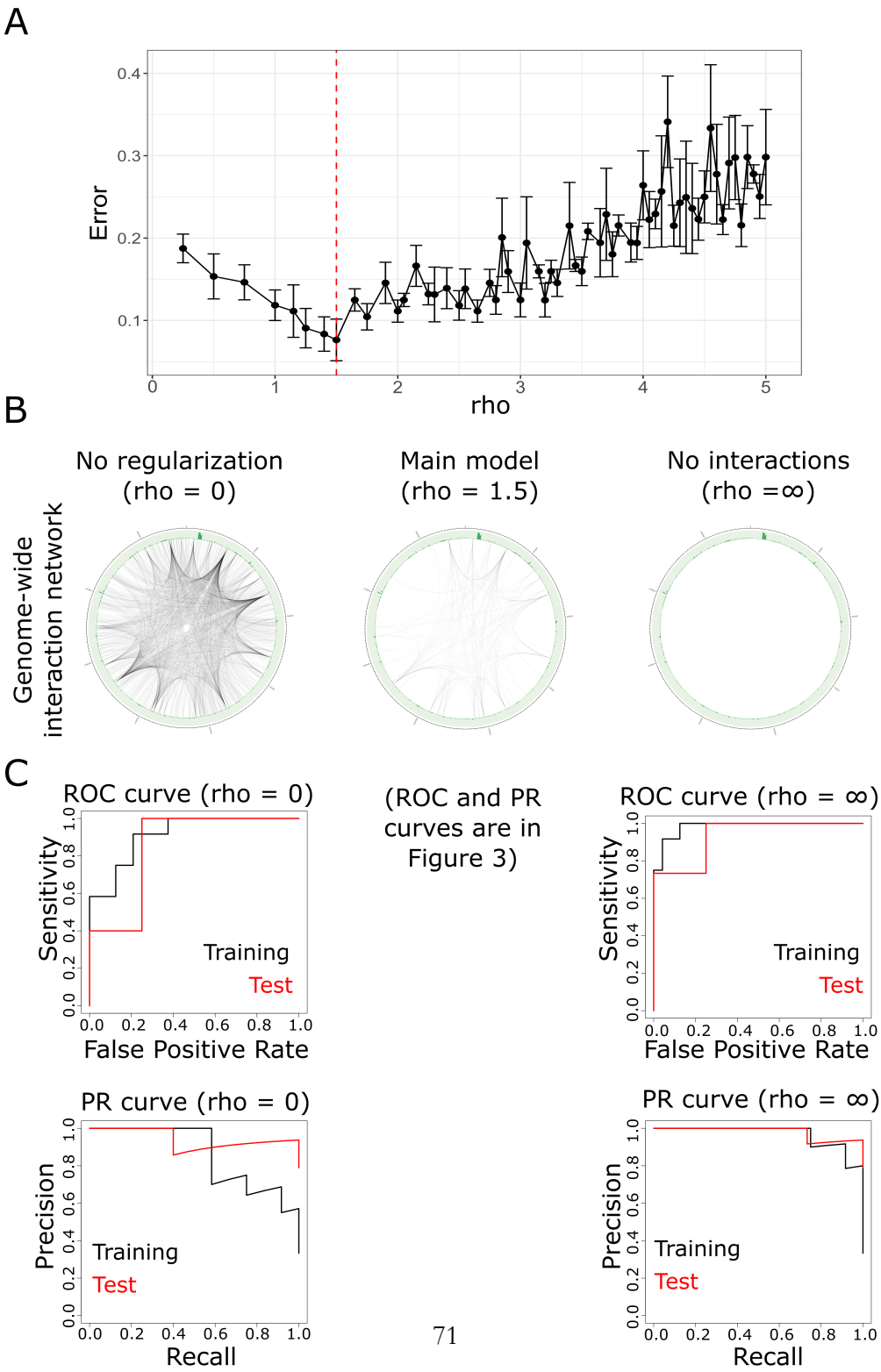
A
Error
rho
B
No regularization
(rho = 0)
Main model
(rho = 1.5)
No interactions
(rho =∞)
Genome-wide
interaction network
C
ROC curve (rho = 0)
Sensitivity
False Positive Rate
Training
Test
(ROC and PR
curves are in
Figure 3)
ROC curve (rho = ∞)
PR curve (rho = 0)
Precision
Recall
PR curve (rho = ∞)

Figure 3.8: (A) (Same as Figure 3.6C) To test whether entropy was sensitive to regularization parameter rho, two extreme values of rho were used, as opposed to the optimal value of $\varrho = 1.5$ determined based on crossvalidation error. Black points and error bars represent mean ± standard deviation of error across $n = 5$ crossvalidation folds. (B) For $\varrho = 0$ (corresponding to no regularization, and a dense network of gene-to-gene interactions), and for $\varrho = \infty$ (corresponding to no interactions, and an empty network), the resulting networks for wildtype T4 exposed to VNC are shown. (C) These two extreme models were evaluated using ROC and PR curves, and resulted in areas under the curve ≥0.85 for all cases in both PR and ROC, for training and test datasets.

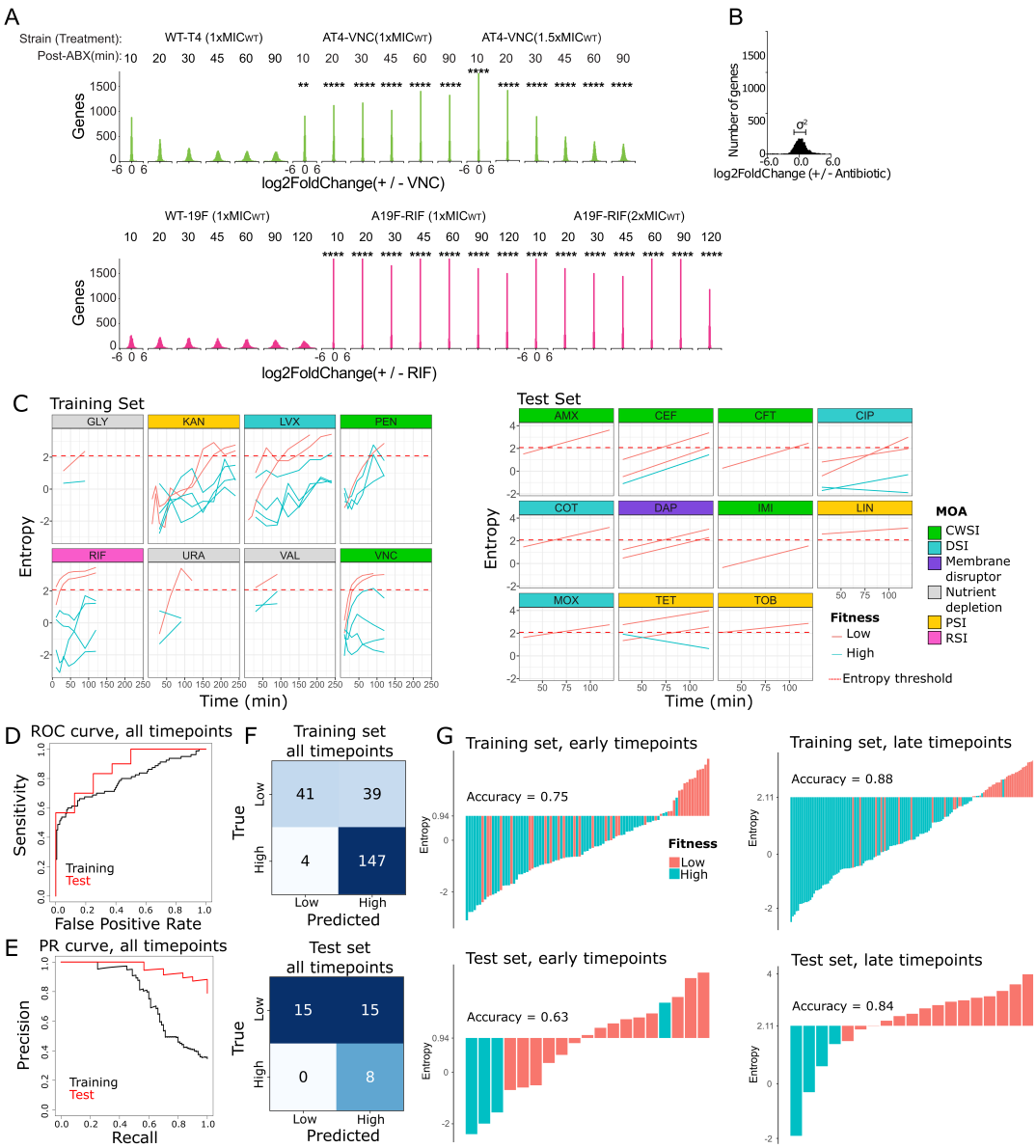
A
Strain (Treatment):
WT-T4 (1xMICWT)
AT4-VNC(1xMICWT)
AT4-VNC(1.5xMICWT)
Post-ABX(min):
Genes
log2FoldChange(+ / - VNC)
WT-19F (1xMICWT)
A19F-RIF (1xMICWT)
A19F-RIF(2xMICWT)
log2FoldChange(+ / - RIF)
B
Number of genes
log2FoldChange (+/- Antibiotic)
C
Training Set
GLY
KAN
LVX
PEN
RIF
URA
VAL
VNC
Entropy
Time (min)
Test Set
AMX
CEF
CFT
CIP
COT
DAP
IMI
LIN
MOX
TET
TOB
Fitness
Low
High
Entropy threshold
MOA
CWSI
DSI
Membrane disruptor
Nutrient depletion
PSI
RSI
D
ROC curve, all timepoints
Sensitivity
False Positive Rate
Training
Test
E
PR curve, all timepoints
Precision
Recall
F
Training set all timepoints
True
Predicted
41
39
4
147
Test set all timepoints
15
15
0
8
G
Training set, early timepoints
Accuracy = 0.75
Training set, late timepoints
Accuracy = 0.88
Test set, early timepoints
Accuracy = 0.63
Test set, late timepoints
Accuracy = 0.84

Figure 3.9: (A) Genome-wide differential expression (indicated as log2FoldChange Antibiotic/NDC (no drug control)) shows significantly wider distributions in antibiotic-sensitive strains (wtTIGR4 and wt19F) compared to antibiotic-adapted strains in the presence of VNC and RIF, respectively in a two-sided Kolmogorov-Smirnov test. $** : 0.0001 < p < 0.001; *** : p < 0.0001$. (B) Entropy for a single time point is defined as the log-transformed variance of the distribution of differential expression across genes for a specific timepoint. (C) Single time point entropy is calculated from differential expression of all genes in experiments in the training (left panels) and test (right panels) datasets at each time point and plotted against time post-stress exposure. Dashed red line indicates the entropy threshold (2.08) for the single-timepoint entropy predictions of fitness. (D, E) The performance of the single time-point entropy-based fitness prediction (applied to all timepoints, ranging from 10' to 240') is shown as ROC (D) and PR (E) curves. The area under the ROC curve is 0.79 and 0.88 for training and test sets respectively. The area under the PR curve is 0.77 and 0.96 for training and test sets respectively. (F) Confusion matrix of single time-point entropy-based fitness prediction of the training (top panel) and test (bottom panel) datasets, highlights a good performance, but shows that there are a relatively large number of false positives. (G) Entropy values of individual experiments in the training (top) and test (bottom) sets, separated by time. Left and right panels show early (≤45 minutes) and late (> 45 minutes) timepoints respectively. It turns out that most false positive predictions in panel F come from early timepoints due to a lack in transcriptional changes within the first 45 minutes after antibiotic exposure. In contrast, antibiotic exposure longer than 45 minutes (late timepoints) leads to a clear separation of high and low fitness and high accuracy in training and test data sets.

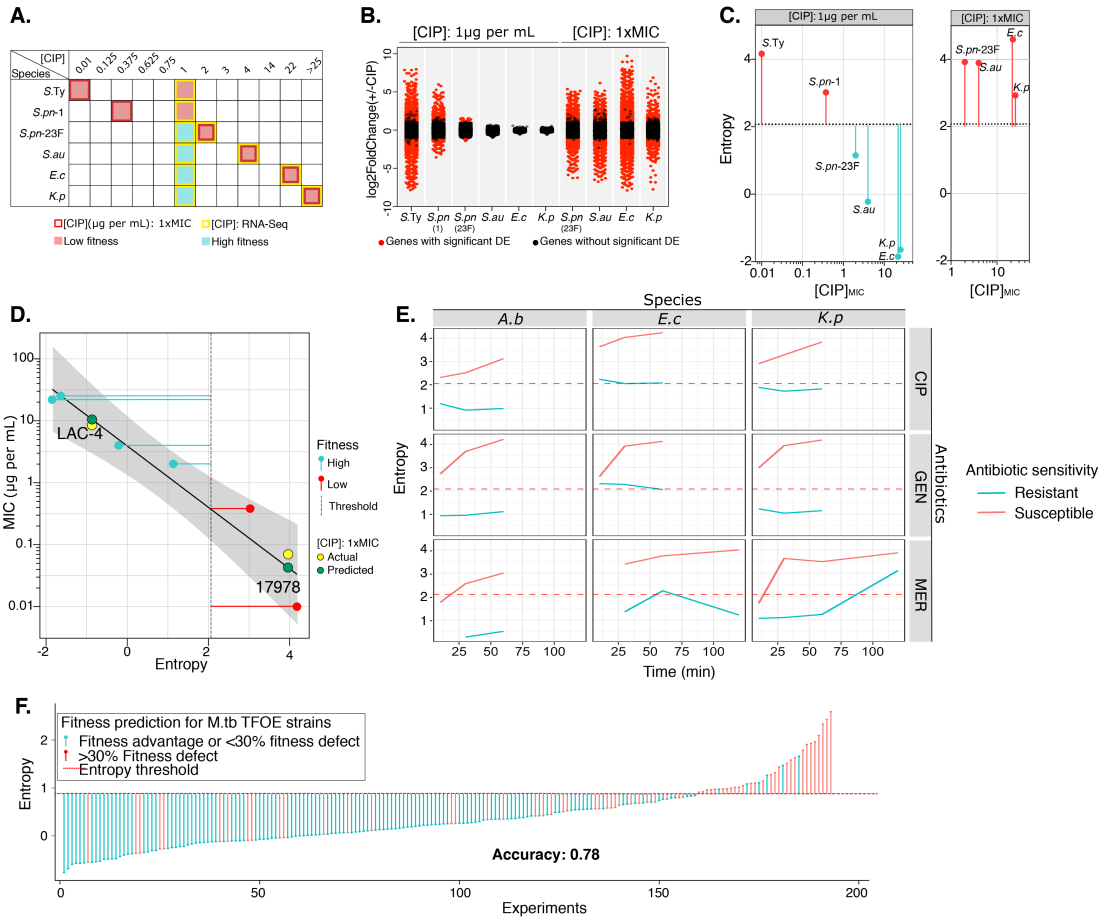
A.
[CIP]
Species
0.01
0.125
0.375
0.625
0.75
1
2
3
4
14
22
>25
S.Ty
S.pn-1
S.pn-23F
S.au
E.c
K.p
[CIP](µg per mL): 1xMIC
[CIP]: RNA-Seq
Low fitness
High fitness
B.
[CIP]: 1µg per mL
[CIP]: 1xMIC
log2FoldChange(+/-CIP)
S.Ty
S.pn (1)
S.pn (23F)
S.au
E.c
K.p
Genes with significant DE
Genes without significant DE
C.
[CIP]: 1µg per mL
[CIP]: 1xMIC
Entropy
[CIP]MIC
D.
MIC (µg per mL)
LAC-4
17978
Entropy
Fitness
High
Low
Threshold
[CIP]: 1xMIC
Actual
Predicted
E.
Species
A.b
E.c
K.p
CIP
GEN
MER
Antibiotics
Time (min)
Antibiotic sensitivity
Resistant
Susceptible
F.
Fitness prediction for M.tb TFOE strains
Fitness advantage or <30% fitness defect
>30% Fitness defect
Entropy threshold
Accuracy: 0.78
Experiments

Figure 3.10: (A) Six strains representing 5 species are ranked from low to high CIP minimal inhibitory concentrations (MIC_{CIP}) tested by growth curve assays. The multi-species CIP RNA-Seq is performed at two CIP concentrations: 1) 1µg per mL for all 6 strains corresponding to 2 low fitness outcomes (red squares) and 4 high fitness outcomes (cyan squares); 2) MIC_{CIP} for strains that are insensitive to 1µg per mL of CIP, i.e. *S. pneumoniae* serotype 23F, *S. aureus* UCSD Mn6, *E. coli* AR538, and *K. pneumoniae* AR497, corresponding to 4 additional low fitness outcomes. The number of genes that change in expression upon exposure to 1µg per mL CIP (|log2FoldChange| > 1 and p-adj < 0.05) as well as their change in magnitude is inversely correlated to their CIP sensitivity (B) and their entropy (C). Additionally, strains with MIC_{CIP} higher than 1µg per mL revert to triggering a large number of differential expression genes (B) and a high entropy (C) at their respective $1 \times MIC_{CIP}$. (D) Using a linear regression model (black line; error band: 95% confidence interval for the regression), MIC's are predicted for *A. baumannii* strains ATCC 17978 and LAC-4 based on their entropy at 1µg per mL of CIP. The predicted (green datapoint) and measured (yellow datapoint) MIC for the two strains are highly accurate indicating that entropy can be used as a quantitative predictor. (E) Further validation of the generalizability of the single time-point entropy approach on published expression data11. The universal entropy threshold of 2.08 trained on our *S. pneumoniae* data, was successfully used to predict fitness outcomes of susceptible and resistant strains from 3 species in the presence of 3 different antibiotics. Importantly, six of the species-antibiotic combinations (GEN-*A.b/E.c/K.p* and MER-*A.b/E.c/K.p*) were not present in our datasets, which highlights the universality and generalizability of the entropy based approach. *A.b: A. baumannii, E.c: E.coli, K.p: K. pneumoniae*. (F) Entropy calculated from transcriptional profiles of 193 *M. tuberculosis* transcription factor over-expression (TFOE) strains from reference [154] separates strains with a >30% fitness defect upon TFOE induction (red) from strains with a fitness advantage or <30% fitness defect upon induction (cyan). At the threshold of 0.71 (red dotted line), fitness outcomes are correctly predicted at an accuracy of 0.78.

4

Boundary-Forest Clustering: Large-Scale Consensus Clustering of Biological Sequences *

4.1 Background

Most bacterial species harbor large amounts of sequence diversity. For example, any given strain of the human respiratory bacterial pathogen *Streptococcus pneumoniae* has about 2,100 genes in its genome, but two strains can differ by the presence or absence of hundreds of genes. In fact, the core genome (the genes shared by all strains) is estimated to be anywhere between 15-50% of the pangenome (the entire genetic repertoire of the species, thought to contain between 5,000-10,000 genes)[54,46,203]. In species such as *S. pneumoniae* where there

*Adapted from Surujon D, Ghazal N, Weiss J, Bento J, van Opijnen T. Boundary-Forest Clustering: Large-Scale Consensus Clustering of Biological Sequences. PLOS Computational Biology. *Under review*. Author contributions: DS, JB and TvO conceptualized the study, DS, NG, and JW performed computational experiments, DS and JB wrote the manuscript, DS, JB and TvO edited the manuscript. All authors approved the final manuscript.

is a large amount of genetic diversity, phylogenetic studies or studies that compare multiple strains first necessitate identifying which genetic elements are the same across the different strains.

Establishing gene correspondence is often achieved by orthologue clustering, which groups orthologues of the same gene based on sequence similarity. An ideal orthologue clustering method is scalable, accurate, allows cluster augmentation (the addition of new sequences to a clustered set, without changing the initial clustering), and assigns a confidence score to the clusters it outputs. Earlier approaches for orthologue clustering such as PanOCT[65] and PGAP[219], involve all-against-all sequence comparisons, which compares each sequence to all other sequences in the dataset, and uses all of these comparisons to cluster. With such an approach, the number of comparisons increase quadratically with the number of data points, making these methods inapplicable for large datasets. PopPUNK is a more recent tool that also performs all pairwise comparisons[113]. However, PopPUNK is designed to cluster strains rather than coding sequences, for downstream population structure analysis. Since its use case is different than orthologue clustering tools, we have excluded PopPUNK in our comparisons. Other approaches such as CD-HIT[117] and Usearch UCLUST[56] require the user to choose a sequence similarity threshold for the clusters. These direct threshold methods ensure that sequences that are more dissimilar than the threshold do not appear in the same cluster, and are extremely fast. CD-HIT has been used for pan-genome clustering for different microbial species[203,102,165], while UCLUST is the default clustering algorithm in the Bacterial Pan Genome Analysis tool (BPGA)[35], which is also used for multiple species' pan-genome analysis[75,176,222,121,18]. Importantly, when using direct-threshold methods, the correct value of the threshold may be difficult or impos-

sible to determine, and an incorrectly chosen threshold value directly impacts clustering accuracy.

An alternative to direct-threshold methods are network-based methods, such as spectral clustering or Markov clustering (MCL)[85,57]. These methods represent each sequence as a node in a network, and sequences are connected to one another according to how similar they are. The resulting network can then be partitioned into clusters based on its topology. Since generating the network requires all-against-all comparisons, these methods also do not scale out-of-the-box. To overcome this challenge, four newer software solutions for pan-genome clustering, PanX[53], Roary[147], PIRATE[16] and Panaroo[192] first use a representative selection step – which reduces the redundancy in, and the size of, the dataset by grouping extremely similar sequences together. For each group, a representative sequence is picked, and the representatives are then clustered using MCL or alternative network-based approaches. The cluster membership for the representatives is then extrapolated to all sequences.

There are multiple strategies for representative selection. For instance, PanX separates consecutive input sequences into groups, then performs clustering within each one of these groups, and finally, selects one representative from each cluster from each group. Alternatively, Roary, PIRATE and Panaroo use CD-HIT as their representative selection method[147,16,192]. In either case, only a single set of representatives is selected, and there is no guarantee that this set best represents the whole dataset, which is a critical limitation. Two additional challenges for pan-genome clustering are a lack of cluster augmentation, and a lack of confidence scores on the clustering output. Currently CD-HIT and Panaroo are the only clustering tools that enable cluster augmentation, while no software produces

confidence scores, which are important in evaluating the ambiguity in the clustering results.

To overcome these challenges, we developed BFClust and made available a MATLAB and a python implementation. BFClust uses a Boundary-Forest as a representative selection step, resulting in multiple sets of representatives that are stored. Each set of representatives is then clustered using MCL, yielding a clustering ensemble. A final consensus clustering step yields a single clustering solution from the ensemble. This approach has 2 main advantages: **1.** multiple sets of representatives and consensus clustering enable calculation of confidence scores; **2.** storing the Boundary-Forest enables quick cluster augmentation.

In this work, we evaluate the performance of 7 clustering methods (including hierarchical, K-means, spectral and MCL), and show that network-based methods such as MCL outperform others. BFClust using MCL is then compared to UCLUST, CD-HIT, PanX, Roary, PIRATE and Panaroo, which highlights that BFClust and PanX have high accuracy and robustness to noise when evaluated on a synthetic dataset generated in silico with known cluster assignments. In real pan-genome datasets, BFClust identifies clusters with low confidence scores, even in the core genome. Since such clusters most likely do not represent real orthologues, the confidence score can thus serve as a means to filter clustering results, only retaining unambiguous clusters. To the best of our knowledge, BFClust is the only clustering solution that produces confidence scores, offers automatic cluster augmentation, and updates confidence scores during cluster augmentation.

4.2 Materials and Methods

4.2.1 Minigenome sequence sets

Nucleotide sequences spanning the first 10 annotated CDS sequences from S. pneumoniae strain TIGR4 were selected (nucleotides 1-27310, spanning locus tags: SP_0001-SP_0010) and used as an initial synthetic "minigenome". Each minigenomes dataset contains 50 copies of these 10 genes, where random independent nucleotide mutations are allowed at a rate *r*. The mutation rate *r* is equal to the probability that one nucleotide is replaced with a different random nucleotide. We generated 100 such nucleotide-based "minigenomes" datasets, namely, 10 datasets for each of 10 different values of *r*, ranging from $r = 0$ to $r = 0.4$. As BFClust uses amino acid sequences by default, the nucleotide sequences for each gene were translated into amino acid sequences. To use Roary and panX, the nucleotide sequences and CDS annotations were converted into GFF3 and genbank files respectively.

4.2.2 Synthetic *Escherichia coli* datasets

In order to test performance on a more realistic dataset, where the ground truth is known, synthetic pangenome datasets were generated using simurg[63]. In particular, each pangenome dataset included 10 strains, each having a modified genome, sampled from the reference *E. coli* strain K-12 (ASM584v2). The core genome was set to be 2000 genes, with gene gain and loss probabilities in the accessory genomes being 10^{-8} and 10^{-11} respectively. A total of 10^9 generations were simulated, and 10 organisms were selected from each simulation to form the synthetic pangenome set. For substitution rates, values ranging from 10^{-12} to

10^{-7} per site per generation were used.

4.2.3 *Streptococcus pneumoniae* datasets

The "RefSeq" dataset ($N = 23$) contains 21 annotated chromosome sequences from the RefSeq database[144] and 2 strains our lab uses in its studies: BHN97[157] and 22F-CT (CDC Pneumococcal surveillance isolate). The "MA" dataset ($N = 616$) is a set of isolates from[46], collected from children between 2000-2007 from Massachusetts. The "Nijmegen" dataset ($N = 350$) includes isolates from invasive pneumococcal disease (IPD) patients in Nijmegen, Netherlands, collected between 2001-2011[45]. The "Maela" dataset ($N = 348$) is comprised of scaffold-level assemblies of carriage isolates collected from the Maela refugee camp in Thailand between 2007-2010[37].

Existing CDS annotations on these genomes were used as the input sequences to be clustered. Since the Nijmegen dataset did not have annotations, the contig fasta files were annotated using Prokka[163]. The genomes were then converted to genbank format, with *dnaA* as the first coding sequence using custom in-house scripts. The translated sequences of all CDS annotations were then extracted into a fasta file for each dataset using Biopython[41]. When necessary, the genbank files were converted to GFF3 for use with Roary, PIRATE and Panaroo. The GFF files retain CDS start and end coordinates on the chromosome/contig.

4.2.4 Prochlorococcus dataset

The assembled, annotated genomes from Biller *et al.*[20] were downloaded as annotated genbank records. They were then converted into GFF and fasta records for use with different

software.

4.2.5 Boundary-Forest

Within BFClust, a large sequence dataset is reduced to a set of representative sequences using Boundary-Forests. For each input dataset, 10 randomized read orders are generated. The sequences are read in these orders and 10 different Boundary-Trees are constructed as described in[131]. Briefly, the first sequence that is read is placed as the root node, and the second as its child. For each subsequent sequence read, it is compared to the root node, and all its children using Smith-Waterman distance[172]. If the sequence being processed is within a pre-determined distance similarity threshold t of a node already on the tree, then this node on the tree becomes its representative. This means that the sequence being processed is marked with the identity of the representative, and is not included in the tree. Otherwise, the sequence is compared to the current node, and all its children, and added to the tree as a child of the node that it is closest to. Most of the input sequences are not included in the tree and are simply associated with a representative on the tree. Boundary-Trees contain 2% of the original input sequences, making the clustering of large numbers of (e.g. 1 million) sequences possible. By default, the sequence distance similarity threshold is 0.1 and each node is allowed up to 10 children. We found that the clustering results on the minigenomes dataset were not altered when these parameters were changed.

BFClust was developed in MATLAB R2017b and the source code is available under the MIT license here. In order to make BFClust available without the need of proprietary compilers, a python version was also developed, under the MIT license, found here.

4.2.6 Clustering

An all-against-all pairwise comparison is done on the representative sequences obtained from each Boundary-Tree to construct a distance matrix S. For each of the following methods, excluding MCL, and each of the 10 replicate trees, a custom range number of clusters K is considered. In the clustering of *S. pneumoniae* pangenomes, a range of $K = 3000, 3200, 3400, , 6000$ clusters is used.

Hierarchical Clustering: an agglomerative hierarchical cluster tree is generated using Ward's linkage[100] on S. Then, the representative sequences are split into K clusters.

***K*-means Clustering:** S is clustered using Lloyd's algorithm[123], with K-means++ for cluster center initialization[7]. This is an approach to partition sequences into K clusters, by iteratively selecting K cluster centroids, assigning points to their closest centroids, and updating the centroids based on the new cluster assignments.

***K*-means Vectorized:** Since K-means is commonly applied to vector data in Euclidean space, we extract from S, vectors in Euclidean space. For this, we first generate the symmetric matrix M, where $M_{ij} = \frac{S_{1j}^2 + S_{i1}^2 - S_{ij}^2}{2}$. Then, the eigenvalue decomposition $M = UVU^T$ is computed, where U is orthogonal and V is diagonal. The product $U\sqrt{V}$ gives Euclidean coordinates for all data points. For the vectorized K-means algorithm, we use the same kmeans function, but with $U\sqrt{V}$ as input instead of S.

Spectral Clustering: The distance matrix S is transformed into an unweighted adjacency matrix W by applying a Gaussian kernel, and thresholding. Then, the graph Laplacian (L) and L's eigenvalue decomposition is computed. The top eigenvectors are then clustered using the standard kmeans function. We consider three variants of spectral clustering. One just as described before, which we call SpectralNN, one where L is normalized

as in[167], which we call SpectralSM (for Shi-Malik), and one where L is normalized as in[140], which we call SpectralNJW (for Ng-Jordan-Weiss).

Markov Clustering (MCL): Similar to Spectral clustering, MCL also uses the adjacency matrix W. W is column-normalized to yield a stochastic matrix. Then a series of expansion (taking matrix power)-inflation (taking element-wise power)-renormalization steps are applied iteratively on this matrix until the resulting matrix does not change. The nonzero elements of the diagonal correspond to attractor nodes. Each attractor, together with all its neighbors in W form a cluster[197].

The run parameters used with each clustering software can be found at the BFClust GitHub repository Data in Figures 2, 5, and 6 were obtained using the MATLAB implementation of BFClust, whereas data in Figures 3 and 4 were generated with the python implementation.

4.2.7 Error and selection of best number of clusters

In cases where the ground truth is not known, we use the sum of squared errors (SSE) as a measure of clustering quality. SSE is defined as follows:

$$SSE(K) = \sum_{i=1}^{K} \sum_{j=1}^{|c_i|} |x_j - m_i|^2$$

Where K is the total number of clusters, c_i is the i'th cluster, and $|c_i|$ is the number of elements in c_i. m_i is the medioid (sequence that has the smallest total distance to all other points within the cluster), and x_j is the j'th element in c_i. We compute SSE for a user-defined range of K values. The most appropriate number of clusters is determined to be the elbow point, or the point of maximal curvature, of the SSE vs K curve. We detect this point by

finding the value of K where the second derivative of $SSE(K)$ is maximized.

4.2.8 Consensus clustering

In order to aggregate the replicate Boundary-Forest clustering results, consensus clustering is used[179]. First the cluster assignments are extended such that each point that was excluded from the Boundary-Tree gets the cluster assignment of its representative on the tree. This is done for the 10 Boundary-Trees, generating a feature vector of 10 clustering assignments for each sequence, for each clustering method. We then use K-medioids clustering, a scalable method, to cluster these feature vectors. For the number of clusters, we use the mode of the best number of clusters from each tree. The feature vectors associated with each sequence is stored for later use, in cluster augmentation.

4.2.9 Cluster augmentation

Given an existing set of clustered sequences, and a new set of sequences, cluster augmentation assigns the new sequences to the closest existing cluster. The new sequences can be processed directly, or the used can choose to do a round of representative selection to reduce the size of the new dataset. A set of representatives is selected from the input sequences by constructing a Boundary-Tree. The representative sequences are then run through the existing Boundary-Forest that was generated when the first set of sequences were clustered. Each representative sequence in the new set traverses each tree in the Boundary-Forest, starting from the root node, by moving to the closest child node. In each tree, the representative is assigned the same cluster as the node it has the smallest Smith-Waterman distance to. This results in as many cluster assignments as the number of

trees in the forest. These cluster assignments are taken as a vector, having the same length as the existing feature vector of clustering assignments prior to consensus clustering. The closest existing cluster for each new sequence is determined by **1.** finding the vector v in the list of stored feature vectors that is closest to the new cluster assignment vector, and **2.** assigning to the new sequence the same cluster as that of vector v.

4.2.10 Matching of two clustering partitions

In order to compare two clustering results, or to compare the misclustering error against a known ground truth, we apply the Hungarian matching algorithm [138]. Briefly, for clustering A and clustering B, if we have n and m clusters respectively, we generate an empty cost matrix M: a $(n+m)\times(n+m)$ matrix of zeros, with each row representing a cluster in A, and each column representing a cluster in B. The $(i,j)^{th}$ entry in this matrix is the dissimilarity cost between cluster i from clustering A and cluster j from clustering B. The entries on the upper left $n \times m$ section of M, i.e. $M(1:n, 1:m)$, are populated with the total number of mismatches between clusters i and j from clustering A and B respectively. That is, the sum $|A_i \setminus B_j| + |B_j \setminus A_i|$, where $|S|$ denotes the size of a set S. The block $M(n+1:n+m, 1:m)$ represents the costs of clusters in B having no representatives in A. Each column in this block is populated with $|B_j|$ for cluster j. Similarly, $M(1:n, 1+m:n+m)$ is populated with $|A_i|$. Finally, $M(n+1:n+m, 1+m:n+m)$ only has 0 cost. We use this sum of costs to be the error between two clusterings (or a clustering and the ground truth, when the ground truth is known).

4.2.11 Overlap of two clustering partitions

We define the overlap between clustering partitions $C1$ and $C2$ on the same dataset as the fraction of clusters in $C1$ that are conserved in $C2$. In other words, if a cluster in $C1$ has all its members in the same cluster in $C2$ (with possibly other sequences included in this cluster in $C2$), it counts towards the overlap. Note that this overlap measure is not symmetrical (i.e. $Overlap(C1, C2)$ is not necessarily the same as $Overlap(C2, C1)$).

4.2.12 Confidence scores

We use definitions of item and cluster confidence scores similar to those defined by Monti *et al.*[137]. For a dataset of size N, that has been clustered on T Boundary-Trees, we define a consensus matrix M which is a $N \times N$ matrix, where $M(i,j)$ is the proportion of times that items i and j have appeared in the same cluster. The item consensus for item i belonging to cluster k is defined as $c_i(k) = \frac{1}{|k|} \sum_{j \in k} M(i,j)$ i.e. the average consensus between i and other items belonging to the same cluster. Similarly, the cluster consensus for cluster k is defined as $c_k = \frac{1}{|k|^2} \sum_{i,j \in k} M(i,j)$ i.e. the average consensus between all pairs of items in cluster k.

4.3 Results

4.3.1 Algorithm Overview

Clustering of sequences using BFClust has three major steps: **1.** representative selection i.e. reducing redundancy in the input data using Boundary-Forest; **2.** clustering of each set of representatives associated with each Boundary-Tree into an ensemble of clustering solu-

tions; and **3.** deriving a consensus clustering from this ensemble of solutions (Figure 4.1). Once a consensus clustering is obtained, each cluster is assigned a cluster confidence score, and each amino acid sequence is given an item consensus score, based on the agreement of the clustering produced using the different Boundary-Trees.

A naïve way to cluster all sequences from many bacterial genomes would be to look at all-vs-all pairwise sequence comparisons. Since all-vs-all pairwise comparisons require a computational effort that scales quadratically ($O(N^2)$ comparisons) with the number of sequences (N), it is beneficial to apply a representative selection scheme such that a group of extremely similar sequences is represented by a single sequence. We achieve this by constructing a Boundary-Forest (see Appendix for pseudocode). In a Boundary-Forest, n Boundary-Trees are constructed, with $n = 10$ as the default size of the forest. Before constructing each Boundary-Tree, the order of sequences is randomized. The Boundary-Tree is constructed by placing the first sequence as the root, and the second sequence as its child. Then, each subsequent sequence is compared to the root node and its children. If the Smith-Waterman distance[172] between the incoming sequence and a node in the Boundary-Tree is smaller than a pre-set threshold t, the incoming sequence is represented by this node, and omitted from the tree. If the incoming sequence is not within the threshold of the root node or its children, we select the node (among the parent and children being compared to the incoming sequence) with smallest distance to the incoming sequence. If the newly selected node also has children, we repeat the comparison, moving down the tree until a representative is found that is sufficiently close to the incoming sequence. If such a node is found, we assign this node as the representative of the incoming sequence, and start processing the next incoming sequence. If no node within distance t is found on the

tree, the new sequence is added as a child of a leaf in the tree. To control the breadth of the tree, the maximum number of children allowed for each node is limited (with the parameter "MaxChild"). Note that below, we explore the sensitivity of the approach to these parameters. Since any Boundary-Tree that is constructed is sensitive to the order in which the sequences are read, a single tree is not guaranteed to capture a set of representatives that leads to highly accurate downstream clustering. Therefore, multiple Boundary-Trees (the Boundary-Forest) are used, which can be thought of as multiple 'opinions' on what representative sequences should be chosen. Once the sequence set is reduced to n sets of representatives, stored as a forest of n trees, the pairwise distances are computed within each set of representatives, and well-established clustering algorithms are applied.

After clustering the representatives, the cluster assignments of the representatives are extended to the full dataset. This is a necessary step for comparing the clustering output to the ground truth, comparing two clustering outputs to each other, and for consensus clustering, as these actions are performed on the full dataset, and not on the representatives. During the construction of each Boundary-Tree, each sequence is assigned a representative (or is itself a representative) based on sequence similarity. Cluster extension from the representatives to the full dataset is done by assigning each sequence the cluster of its representative. The representatives of each Boundary-Tree are used to produce one clustering output, the whole Boundary-Forest thus leading to an ensemble of possible clustering outputs. Consensus clustering across the clustering ensemble is then applied, combining the clustering output obtained from each tree, to improve accuracy. Finally, BFClust compares how the different clustering outputs in the ensemble contribute to the consensus clustering, and using the differences in these contributions it assigns an item confidence score to

the membership of each sequence to its consensus cluster, and a cluster confidence score to the existence of each cluster.

4.3.2 Boundary-Forest reduces redundancy in the sequence set

In order to evaluate whether Boundary-Forest effectively reduces an input dataset into a small set of representatives by removing redundant sequences, we studied how much this step reduces the size of the dataset, how this reduction depends on the algorithm's parameters, and how, in turn, this affects downstream clustering accuracy. We generated a small test dataset ('minigenomes'), with 500 sequences of varying length (ranging from 65 to 1170 amino acids). This dataset has 50 noisy copies of 10 genes, and therefore 10 inherent sequence clusters. The noise is independent, random changes in the nucleotide sequence with probability 0.01 per nucleotide. Since BFClust uses amino acid sequences by default, the perturbed nucleotide sequences were then translated into perturbed amino acid sequences *in silico*. As the changes introduced to the sequences are random, 10 replicate sequence sets with the same mutation probability were generated. Figure 4.2A shows how the size of the Boundary-Tree constructed from this dataset is robust to two parameters that are crucial in constructing the Boundary-Tree: MaxChild and the sequence similarity threshold t. A detailed description of all parameters used in BFClust is provided in S1 Appendix. While a drastically small threshold value ($t = 0.01$) results in larger trees (which is intuitive, since with a smaller similarity threshold, fewer sequences can be represented by the same node), the size of the tree is robust to a large range of t and MaxChild values. Once a tree is generated, applying downstream clustering still may require all pairwise comparisons on the representatives. However, the number of pairwise comparisons are

now greatly reduced; for example, a tree generated with $MaxChild = 10$ and $t = 0.1$ has 15 nodes (Figure 4.2A), which requires only $\binom{15}{2} = 105$ pairwise comparisons for clustering, versus $\binom{500}{2}$ 125,000 when clustering the entire dataset. Importantly, the construction of each Boundary-Tree also requires relatively few sequence comparisons itself: for example, 4500 comparisons are sufficient to generate the Boundary-Tree, when $MaxChild > 2$ (Figure 4.2B). The trees constructed are also relatively shallow (Figure 4.2C), meaning that the addition of a new sequence to an existing forest will require very few sequence comparisons (the number of comparisons grows with the depth of the tree) and will thus be very fast, which is relevant when we later discuss clustering augmentation. This is in line with the results reported by the creators of Boundary-Forest, where the depth of Boundary-Trees was shown to depend logarithmically on the number of data points for multiple datasets[131]. Importantly, applying the full BFClust pipeline with varying the parameters MaxChild and t did not alter the clustering output, and the recovered clusters using any combination of these parameters (in the ranges presented in Figure 4.2) were identical to the ground truth and resulted in no error in clustering output.

There are alternatives to representative selection that involve simpler algorithms than Boundary-Forest; two approaches we discuss here are a random sampling and a naïve sampling strategy. Random sampling is the selection of representatives randomly from the full set of sequences. This is not a viable strategy, as it does not guarantee that all clusters will be selected. For example, on the set of 500 sequences with 10 known clusters, a random selection is likely to include at least one representative from all 10 clusters only when a sufficiently large number of sequences (e.g. $N = 50$) are randomly selected. In this case, a 10-fold reduction in the number of representatives (compared to the full sequence set)

may seem promising. However, in real pan-genome datasets, this reduction might result in hundreds of thousands of representative sequences, which would still be prohibitively numerous for downstream clustering. Moreover, in real sequence sets it is not clear how many sequence clusters are present, and random sampling risks missing smaller clusters. Thus, estimating the number of random samples to be selected such that all clusters will be represented is difficult, if not impossible.

In the second, naïve sampling scheme, sequences are read in a random order, and the first sequence is placed into a 'representatives' group. Each incoming sequence is then compared to the existing representatives, and if no representative closer than a threshold t is found, the incoming sequence is added to the representatives group. Both CD-HIT and UCLUST apply this naïve sampling strategy. On the small 500-sequence minigenomes dataset, the number of representatives selected by the naïve sampling and Boundary-Tree are similar (Table 4.1). However, cluster augmentation, which is a key advance we present below, requires the comparison of new sequences to all of the representatives in the case of naïve sampling, while in the Boundary-Tree, the traversal of a much smaller subset of representatives is required. The number of comparisons on a Boundary-Tree is dependent on the tree depth, and the number of children each node has, which is limited with the MaxChild parameter. With MaxChild = 10, the estimated number of possible comparisons in the Boundary-Tree for new sequences would be at worst 10 × (tree depth), whereas in the naïve scheme it would be equal to the current size of the representatives set. The advantage of the Boundary-Forest becomes apparent when a larger sequence set is considered. For instance, 20 *S. pneumoniae* strains were selected from the RefSeq database, and the coding sequences were subjected to naïve sampling and Boundary-Tree sampling. While the num-

ber of representatives in the Boundary-Tree is about twice as large as the representatives picked with naïve sampling, the trees are shallow. The number of comparisons needed to process a new sequence in the Boundary-Tree is 90, which is about 35-fold smaller than the comparisons using the naïve representative set (3265). Therefore, we conclude that the extra effort at the beginning of constructing the Boundary-Forest results in more efficient sample processing as the sequence dataset grows larger.

4.3.3 BFClust can compute cluster confidence scores

The consensus clustering step across the Boundary-Forest not only reduces error, but it also allows confidence estimation for the existence of each cluster, and for the membership of each sequence in its consensus cluster. By comparing the clustering done using the representatives on each Boundary-Tree, it is possible to measure how frequently a cluster has the same members, and use this value as an estimate of cluster confidence. We define a cluster confidence score (for each cluster) and an item confidence score (for each sequence), and include both sets of values in the BFClust output. Both values depend on the consensus index[137]. The consensus index of a pair of sequences *i* and *j* is the number of times that they appear together in the same cluster across n Boundary-Trees, divided by the total number *n* of Boundary-Trees used. The item confidence score for item i is the average consensus index between i and all other members of the same consensus cluster. The cluster confidence score is the average consensus index between all pairs of items within the same consensus cluster (Figure 4.3). Both scores take a value between 0 and 1, and a score of 1 indicates perfect agreement of cluster memberships across the Boundary-Forest.

4.3.4 Cluster augmentation: addition of new sequences to an existing clustering.

A major advantage of the BFClust algorithm is that it stores the Boundary-Forest containing representatives from all previously processed input sequences. This allows BFClust to add new sequences to an existing clustering/partition while being able to update the confidence scores without much computational work. To achieve this, a cluster augmentation method is implemented (see Figure 4.4A for a schematic overview). A set of incoming input sequences can either be used as-is, or optionally are reduced to a new set of representatives by constructing a new Boundary-Tree. These new sequences (or representatives) are then run through each tree in the existing forest (corresponding to the already clustered set of sequences), and for each new representative, a close match on each of the 10 trees is identified. The cluster membership associated with each tree is extracted for these close matches from the previously computed clustering. Each new sequence is assigned the same clustering membership as that of the corresponding close match within each tree. This results in a vector of cluster assignments for each new sequence. Afterwards, the vectors composed of the cluster memberships for the new input representatives from each tree are turned into a consensus cluster assignment using a nearest neighbor search on the cluster assignments of the datapoints in the existing dataset. If an initial representative selection step was used, the consensus clustering on each input representative is then extended to the entire input set, using the same procedure of cluster extension during *de novo* clustering.

The runtime of de novo clustering and cluster augmentation scales tractably with increasing number of data points (Figure 4.4B). For *de novo* clustering, increasing numbers of strains of *S. pneumoniae* were included. For cluster augmentation, the 5 additional strains were augmented onto the already-clustered data. The runtime remains the same, regardless

of the size of the existing clustered dataset. (Figure 3B). Memory usage on the other hand does depend on the size of the existing dataset (Figure 4.4C).

4.3.5 Comparison and benchmarking with existing methods

We selected four existing sequence clustering methods to compare BFClust against. The first is Usearch UCLUST[56], a very fast and scalable algorithm that is also the basis of several other pangenome phylogenetic analysis pipelines such as SaturnV[66], PanPhlAn[159] and BPGA[35]. Second, we consider CD-HIT, another scalable software that has been used directly in pan-genome analysis[102,165] and for representative selection in other pipelines such as Roary[147], PIRATE[16] and Panaroo[192]. These three tools are also included, as they allow pan-genome analysis of several hundred genomes at once, and therefore have been utilized in recent studies[151,158]. Finally, PanX[53] is included, another recent software that can be used with hundreds of genomes. First, the runtime and memory usage of the four software tools was compared to BFClust (Figure 4.5A-B). The direct-threshold methods UCLUST and CD-HIT are orders of magnitude faster than the other methods and have a small memory footprint. On the other hand, methods that employ network-based clustering take far longer, and use more memory. With 250 input genomes, which corresponds to 500,000 coding sequences to be clustered, BFClust has a larger runtime and memory footprint (Figure 4.5A-B, Table 4.2). The increased memory use can be explained by the ensemble clustering approach BFClust employs, which is critical for computing confidence scores. The time and memory use, while higher in BFClust, do scale similarly to most other modern clustering tools.

In order to compare the sensitivity to noise of our approach to existing methods for

biological sequence clustering, we generated an extended set of test sequences. Here, we made 10 replicates of small pangenomes of *Escherichia coli*. The reference strain K-12 was evolved *in silico*, with both *de novo* mutations and lateral gene transfer events allowed[63]. The *de novo* mutation rate was varied between 10^{-12} and 10^{-7} mutations per nucleotide per generation. The clustering output was compared to the known cluster assignments for each mutation rate. For a mutation rate of 10^{-10} per nucleotide per generation, BFClust, PanX and PIRATE are the three tools that have high overlap with the real cluster assignments. In contrast, Roary and Panaroo have less overlap with the real clusters at low substitution rates (e.g. 10^{-11}) and UCLUST and CD-HIT have a very sharp drop-off (Figure 4.5C, Table 4.2). The location of this drop off depends on the threshold chosen by the user; more stringent thresholds would result in the accuracy to drop at a smaller substitution rate. Based on this, we recommend using PIRATE, PanX or BFClust when a high amount of variation is expected in the sequence data, either due to genetic variation, or noise from error-prone sequencing technologies[95]. An overall comparison of all 7 approaches are summarized in Table 4.2.

4.3.6 Clustering of real pan-genomes

To demonstrate the applicability of BFClust beyond synthetic datasets, several real pan-genome *S. pneumoniae* datasets were explored. *S. pneumoniae* is a naturally competent, opportunistic human pathogen that is known to have a relatively large pan-genome, partially shaped by recombination events[54,46]. Since in a real pan-genome, the ground truth for clustering is unknown, it is not possible to compute clustering error. Therefore, in this section we compare different clustering methods to each other and see whether they yield consis-

tent outputs, in addition to exploring the cluster confidence scores generated by BFClust. Previously, core and pan-genome analyses using Roary had revealed that across different datasets of pneumococcal isolates, the core genome is not conserved, and the size of the pan-genome is not the same across datasets[203]. However, it is unclear whether this is a consequence of the datasets (which come from different populations that are also geographically separated) and/or an artifact of the clustering method used. In order to avoid any bias associated with a specific dataset, we compiled 4 datasets in this study: **1.** RefSeq (closed, chromosomal genomes, n=20)[144] **2.** Maela (annotated contigs from a Thai refugee camp, n=348)[37]; **3.** Nijmegen (annotated contigs from a Dutch hospital, n=350)[45]; and **4.** MA (annotated contigs from surveillance data from Massachusetts, n=616)[46]. Despite being the smallest dataset, the RefSeq set is the most diverse, as these strains have collection dates and countries of origin that vary the most. The Nijmegen dataset is comprised of pneumococcal isolates from invasive pneumococcal disease patients, whereas the MA and Maela datasets are collections of pneumococcal isolates from healthy individuals (i.e. carriage isolates).

First, we evaluated whether the core and accessory genome profiles detected by each method are consistent. A reasonable expectation for a given tool is that it produces similar core and pan-genome size estimates for the 3 larger datasets (MA, Maela, Nijmegen). This expectation is met by all methods but Roary, which shows a big discrepancy in the core and pan genome size across these datasets (Figure 4.6A, B). Relative to the other methods, it appears that Roary underestimates the core genome size, and over-estimates the pan-genome size (Figure 4.6A, B). In comparison, BFClust and PanX both find a larger core genome and a smaller accessory genome compared to the other methods, whereas UCLUST and

CD-HIT find a similarly sized core genome, but a larger accessory genome compared to BFClust and PanX (Figure 4.6).

In order to compare the agreement between clustering methods on a given dataset, we computed cluster overlap: the proportion of clusters generated by one method that are fully contained within another cluster generated by a second method (note that this measure is sensitive to the direction of comparison; agreement of method A with B is not necessarily the same as the agreement of B with A; see Methods). Interestingly, on the same datasets, CD-HIT and UCLUST had the highest agreement, as determined by cluster overlap (Figure 4.6C). BFClust and PanX were also in high agreement. Roary appears to have poor agreement with other methods in one direction, which could be attributed to the fact that it is producing many more clusters, fewer of which end up in the core genome. This is potentially a consequence of Roary using CD-HIT for the first step of selecting representative sequences, as both were sensitive to noise.

Clustering of pan-genome sequences can be the first step of phylogenetic analyses. For instance, the SNPs within the core genome can be used to generate phylogenetic trees and make conclusions on population structure[46]. In these analyses, it is essential that the clustering is unambiguous; incorrect clustering would potentially lead to misleading conclusions. We therefore computed the cluster confidence scores for each cluster obtained using BFClust, on each of the 4 *S. pneumoniae* datasets. The majority of the clusters had a score near 1, indicating very little ambiguity in the clustering output (Figure 4.7). Specifically, we observe high-confidence clustering in the core genome; the mean score for core genome clusters is > 0.999 (and the median score $= 1$) in all 4 datasets. In the 3 larger datasets (Maela, MA and Nijmegen), we observe that the lower scoring clusters are mainly in the

accessory genome, shared by less than a third of the strains included. In all datasets, there exists a single cluster with a much lower score than the average, present in the majority (and in some datasets all) of the strains included (marked in red in Figure 4.7). This cluster is comprised mostly of sequences of very short length (30 amino acids), annotated as hypothetical genes. It is unclear whether these short sequences are artifacts of sequencing errors, annotation errors, incomplete genome assembly or a combination of these factors.

In order to extend our analysis beyond a single species, a set of Prochlorococcus genomes were clustered, and the results of the 7 methods were compared. This dataset comes from a marine cyanobacterium species complex with much greater genomic diversity than *S. pneumoniae*[20]. BFClust identified 5703 clusters, 939 of which belonged to the core genome. These values are similar to the core and pan genome sizes reported using PanX[53], as well as those found here using PanX and PIRATE (Figure 4.8A). The agreement across methods is within the same range as the *S. pneumoniae* datasets (Figure 4.8B). However, the overall consensus scores of clusters are lower in this dataset, potentially due to the high amount of diversity in this collection of strains (Figure 4.8C). This indicates that the solutions found by any one method may have uncertainty, however methods that do not employ a consensus clustering step might give the user a false sense of security in the correctness of their output. The uncertainty in the clustering output is potentially a result of the large amount of diversity of this species complex. In fact, all methods' outputs suggest that Prochlorococcus has a large, open pangenome (Figure 4.8A). This is not surprising, as this species complex occupies and is adapted to marine niches that vary in light intensity, nutrients available and temperature. Given the low confidence scores, clustering results in this highly diverse dataset should be interpreted with the increased uncertainty in mind.

4.4 Discussion

When clustering a set of sequences from a bacterial pan-genome, there are multiple options regarding the software/algorithm to choose from. We observed that direct-threshold methods such as UCLUST and CD-HIT are extremely fast, have the advantage of scalability, but they often do poorly in terms of accuracy and sensitivity to noise (Figure 4.5). They also require the user to select a sequence similarity threshold, assuming all sequence clusters have similar sequence diversity, which is not always true. Different genetic elements are subject to different selective pressures, and therefore sequence conservation/diversification may be associated with multiple factors, e.g. essentiality[99], rendering the use of a single threshold problematic. Therefore, it is more advisable to first reduce the dataset into a smaller representative set (potentially using these faster methods) and then apply a more rigorous clustering method.

For the selection of representative sequences, we propose the use of Boundary-Forest, which is supported by existing numerical experiments showing the improved accuracy and speed of Boundary-Forest compared to other algorithms[131]. Its implementation is also very simple (see pseudocode in Appendix). The inclusion of multiple trees in the forest and downstream application of consensus clustering reduces errors, and results in BFClust being highly tolerant to noise, especially when used with network-based downstream methods such as MCL. Furthermore, the use of multiple Boundary-Trees makes it possible to compute confidence scores. Saving a copy of the shallow Boundary-Trees allows rapid cluster augmentation without having to alter the existing clustering assignments, which is highly desirable for consistency. Moreover, augmentation can be done while updating the

clustering confidence scores. This makes BFClust the only pan-genome clustering method that can generate such a cluster confidence score, both during de novo clustering and during cluster augmentation. These added features do come with a small cost of runtime and memory – they are both an order of magnitude larger than most other tools, but scale similarly (Figure 4.5). The majority of the time is spent on Boundary-Tree generation, and more specifically during the sequence comparisons. The current implementation of BFClust uses non-compiled python functions. Use of compiled and speed-optimized software such as the NCBI BLAST+ suite can potentially increase speed during the Boundary-Tree generation step, as it is approximately 1000x faster than the current implementation. The increased memory use can be addressed by using swap files on computers where solid state drive space can supplement random access memory.

The cluster augmentation strategy implemented in BFClust (and in CD-HIT) is distinct from online clustering methods, which update the clustering, as new data points become available. This can potentially change the cluster memberships of the already-clustered dataset. BFClust on the other hand performs cluster augmentation by using a K-nearest neighbor search to find a cluster in the existing dataset that is a close match of the incoming sequence, without altering the existing clustering. This *K*-nearest neighbors search could potentially be replaced by a *K*-means or *K*-medioids clustering on the full combined set of already-clustered and incoming sequences, in order to make BFClust more similar to an online method. However, it is not clear whether the same *K* value (total number of clusters, which was used in the initial clustering) would apply to the full set with the incoming sequences included. BIRCH[218] and stream clustering[81] are two examples of online clustering algorithms, however it is not known whether they would apply well to biological sequences,

as they are non-network-based methods. The BFClust strategy has a number of advantageous features that can be explored further. Since each of the trees generated in BFClust has a small depth, the number of comparisons one needs to make for a new sequence set is relatively small (tree depth $\times$ 10 trees). Thus, this method offers a framework that makes the rapid integration of new clinically important isolates and their sequences possible. In the same vein, it is possible to quickly compare the clustering results of two different datasets (e.g. isolates of the same species of bacteria, collected from different geographical locations) by running one set through the Boundary-Forest of the other. Moreover, the networks that are generated as intermediate steps in clustering may harbor novel data that remains unexplored in this work. For instance, it may be possible to extract additional information from the network connectivity of sequences, and infer evolutionary trajectories of different genes under differing selective pressures[36].

In conclusion, UCLUST and CD-HIT may not be best suited for pan-genome clustering, as they depend on a user-supplied similarity threshold. UCLUST, CD-HIT, Roary and Panaroo are sensitive to noise in the data. Nevertheless, the speed of UCLUST and CD-HIT make these methods attractive alternatives to BLAST when querying large datasets. Overall, BFClust and PanX are pan-genome orthologue clustering methods that are in high agreement, and can tolerate noise in the sequence dataset. However, when the dataset is highly diverse, we observed that there is a lot of uncertainty in the clustering results, as shown by the low consensus scores on the Prochlorococcus dataset (Figure 4.8B). Panaroo and PIRATE have lower runtimes and memory requirements as an advantage. PanX has the advantage of informative and interactive visualizations, whereas BFClust has the added features of estimating confidence scores. Moreover, most pan-genome clustering

methods (with the exception of CD-HIT and Panaroo) do not readily allow cluster augmentation, and to the best of our knowledge, no previous clustering method enables cluster augmentation while being able to output confidence scores. Confidence scores are crucial in pan-genome clustering, as they allow the researcher to avoid using ambiguous clusters (i.e. clusters with a low score) in downstream analyses and interpretation. With the confidence score of BFClust, such clusters can automatically be detected and excluded from further analysis.

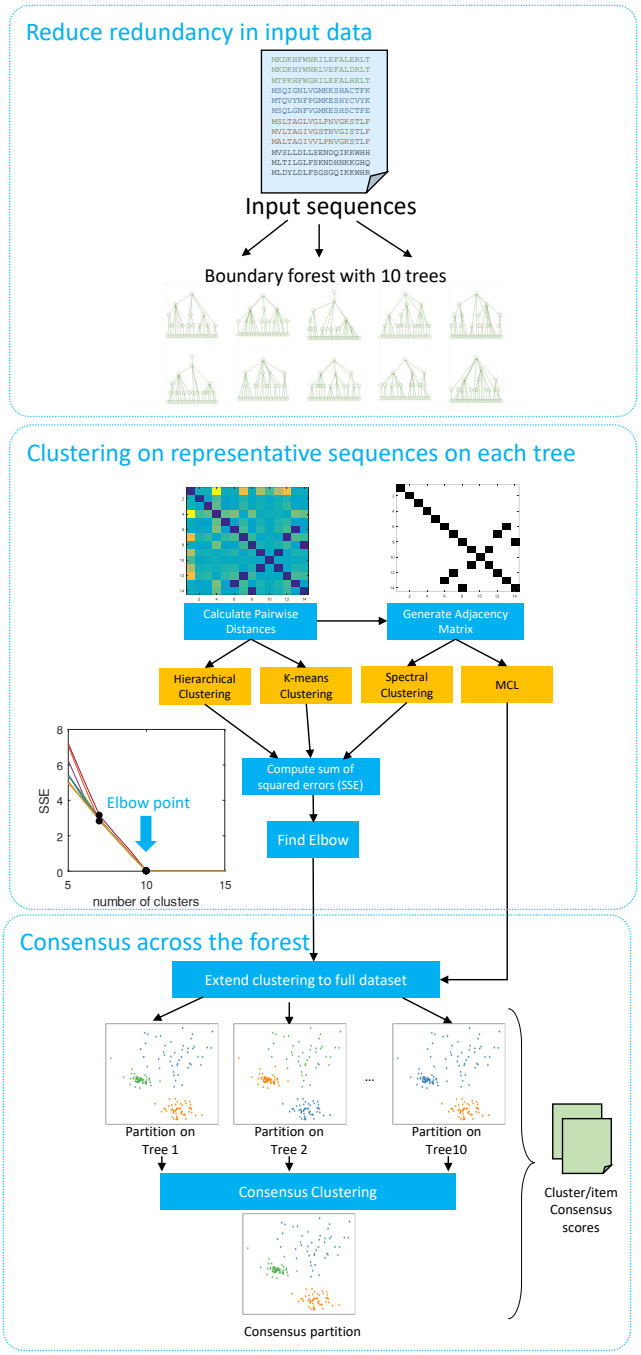
Reduce redundancy in input data
Input sequences
Boundary forest with 10 trees
Clustering on representative sequences on each tree
Calculate Pairwise Distances
Generate Adjacency Matrix
Hierarchical Clustering
K-means Clustering
Spectral Clustering
MCL
Compute sum of squared errors (SSE)
Find Elbow
SSE
Elbow point
number of clusters
Consensus across the forest
Extend clustering to full dataset
Partition on Tree 1
Partition on Tree 2
Partition on Tree10
Consensus Clustering
Consensus partition
Cluster/item Consensus scores

Figure 4.1: From the input sequences, multiple sets of representatives are selected using Boundary-Forest. Each set of representatives is stored as a Boundary-Tree. This reduces a large input dataset to a small set of representative sequences in the forest. Then, representatives on each tree are clustered using MCL. For comparison purposes, the following alternative algorithms were tested: Hierarchical, 2 variants of K-means, and 3 variants of Spectral clustering. Once representative sequences on each tree are clustered, the cluster assignments are extended to the full input sequence set, producing a clustering ensemble i.e. one clustering output associated with each set of representatives. A consensus clustering step is then used to take the clustering ensemble across the trees and produce a single clustering solution, as well as confidence scores. Cluster consensus scores are calculated for each cluster, and item consensus scores are calculated for each sequence within each cluster.

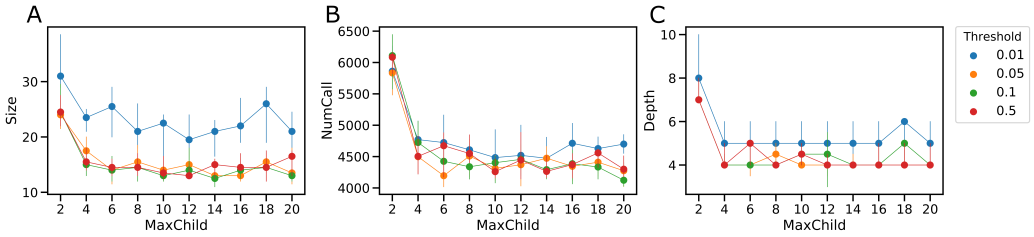

Figure 4.2: Boundary-Trees were generated from a 500-sequence dataset, in order to select representatives. The trees are small, shallow and quickly constructed. MaxChild: maximum number of children allowed for one node. Threshold: sequence similarity threshold, below which a sequence is assigned the tree node as a representative. (A) The size (number of nodes) of the Boundary-Tree (B) Number of calls made to the sequence comparison function (C) The depth of the resulting tree, dependent on MaxChild and Threshold. Overall, the tree depth/size/number of calls made to construct the tree are robust to user defined parameters MaxChild and threshold. Points are the mean ± standard deviation for 10 replicates.

Dataset	N	Representatives (Naïve Sampling)	Representatives (Boundary-Tree)	Tree depth	BT comparisons
minigenomes	500	15.5±5.6	13.2±2.0	4.5±0.8	45
RefSeq	42010	3264.7±5.4	6579.8±78.3	9±0.82	90

Table 4.1: Comparison of naïve sampling and Boundary-Trees as representative selection methods. Representatives were selected from two datasets (minigenomes, a synthetic small sequence set; and RefSeq, sequences from 20 *S. pneumoniae* strains present in the RefSeq database), using either naïve sampling or Boundary-Trees. N: number of sequences in the dataset. Representatives: number of representatives selected with naïve sampling or Boundary-Tree. Tree depth: depth of a Boundary-Tree. Mean±standard deviation of 10 replicate sets of representatives are reported. BT comparisons: expected number of comparisons that will be made during cluster augmentation using Boundary-Trees (note this value is the same as tree depth multiplied by the number of children allowed, which is 10 by default).

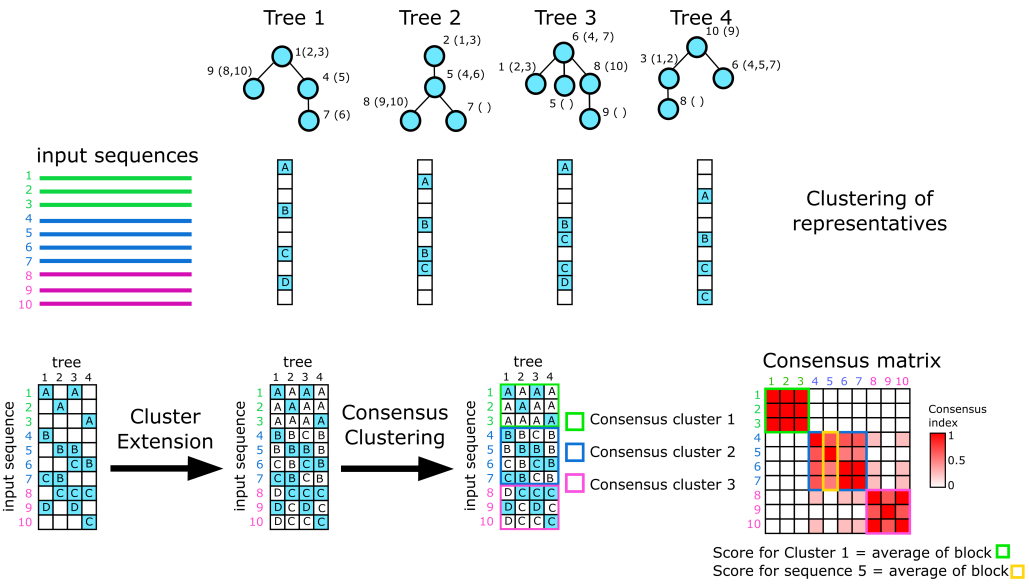

Figure 4.3: In this example, 10 sequences coming from 3 clusters are processed using BFClust. First, the Boundary-Forest is generated (for the sake of example, we use a forest with only 4 trees). In each tree, the nodes are marked by the identity of the representative sequence, and which other sequences they represent (in parentheses). E.g. in tree 1, the root node is sequence 1, and it represents sequences 2 and 3. The representatives on the Boundary-Trees (highlighted in blue) are then clustered, resulting in clusters A-C or A-D, and the clustering output is then extended to the full dataset. After cluster extension, consensus clustering is performed, using each row of cluster assignments as the input. To compute confidence scores, a consensus matrix is generated, where each entry is the consensus index between two input sequences, which is the overlap of cluster assignments between these two sequences across the forest. The score of a cluster (i.e. cluster confidence score) is the average consensus index among all pairs of sequences within that consensus cluster (e.g. the score of consensus cluster 1 is the average of the block outlined in green, which is 1 in this example). The score of a sequence (i.e. item confidence score) is the average consensus index between this sequence, and all other sequences belonging to the same consensus cluster (e.g. the score of sequence 5 is the average of the block outlined in yellow, which is 0.688)

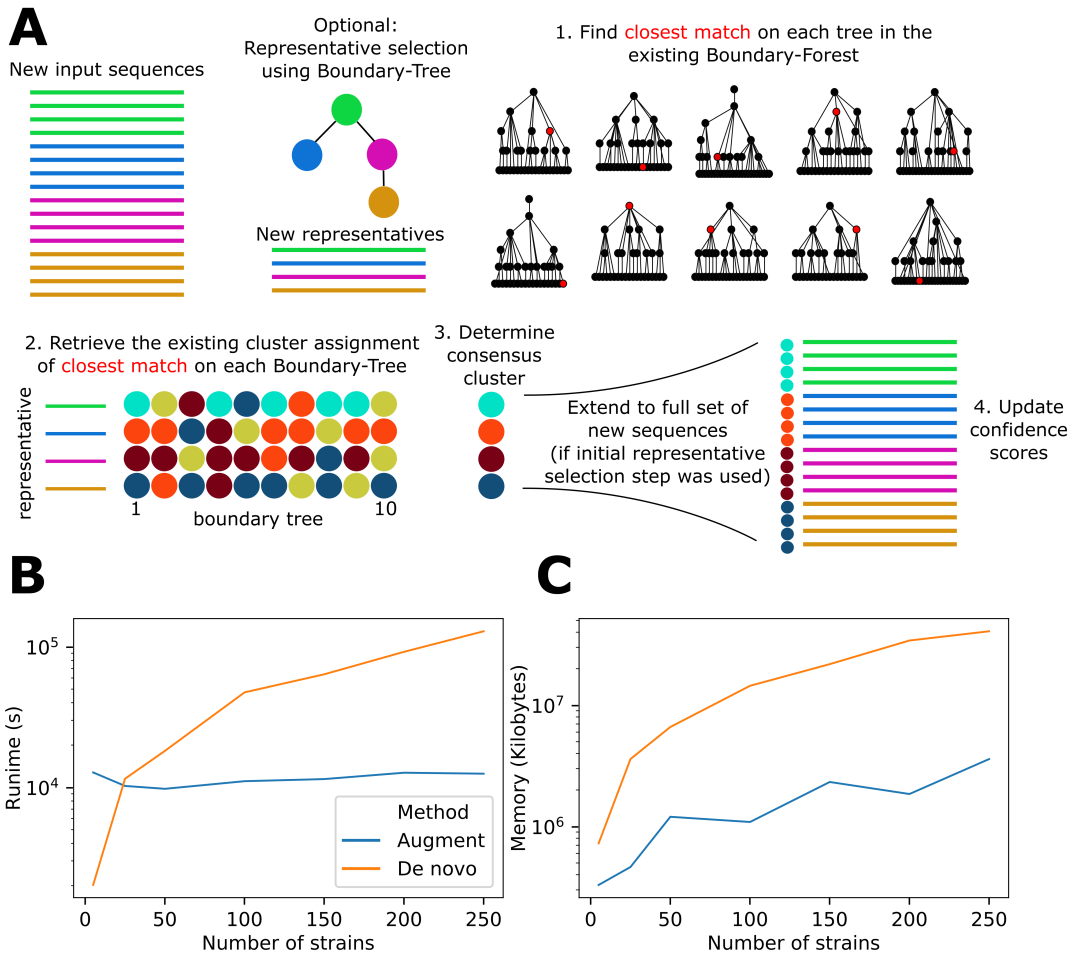

Figure 4.4: (A) Cluster augmentation method overview/schematic. The incoming sequences are either processed as-is, or they can be reduced to a small set of representatives using a Boundary-Tree. The new sequences (or representatives) are compared to the existing Boundary-Forest associated with the already clustered dataset. A close match in each tree, for each input sequence is found (red nodes). The cluster assignments of these closest matches are retrieved, and a consensus cluster assignment is computed using a nearest neighbor search. If representative selection is used, the consensus clusters assigned to the new representatives are extended to the full input dataset. The cluster assignments of the new sequences, as well as updated confidence scores for both the existing and new sequences are produced as the output. (B) Cluster augmentation is faster than clustering de novo. Runtime of clustering sequences *de novo* (orange), or cluster augmentation onto an already clustered set (blue). For augmentation, N genomes were clustered de novo, and the runtime for the augmentation 5 new genomes (10,000 sequences) is reported. (B) Cluster augmentation uses less memory than clustering de novo.

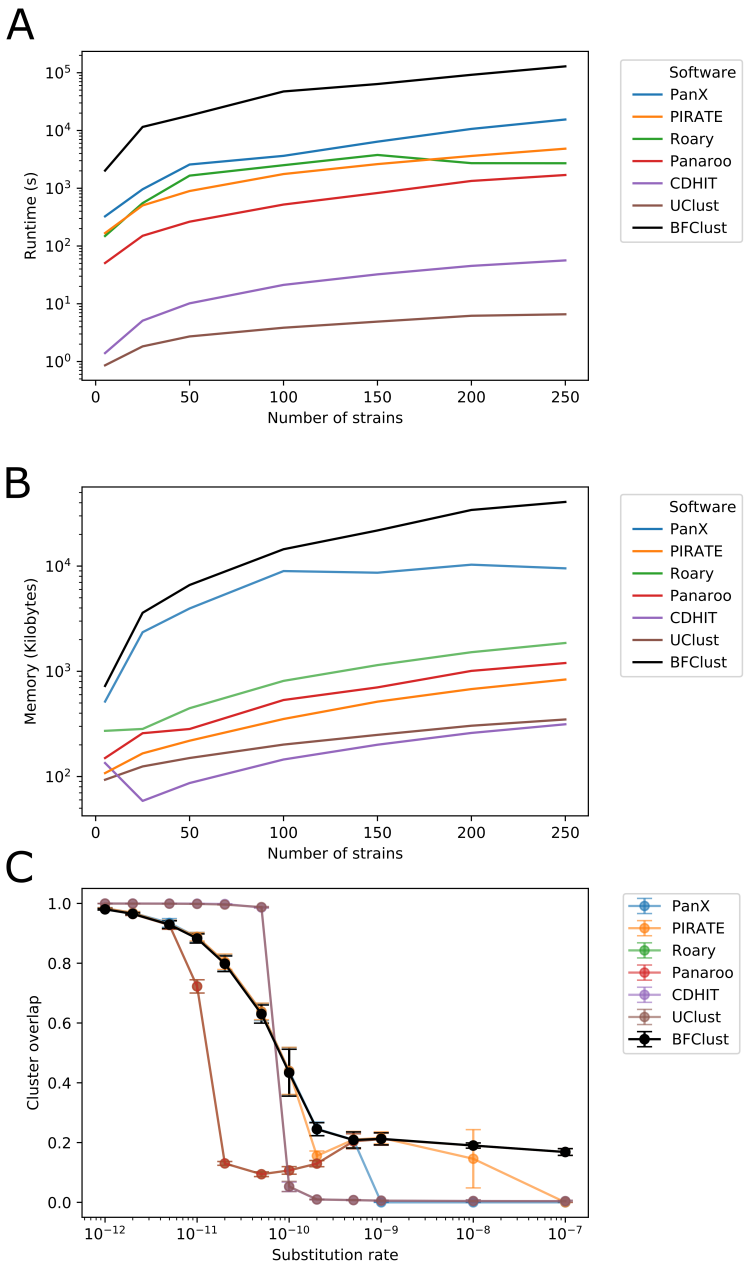

Figure 4.5: (A) Runtime in seconds of each method, as a function of dataset size (number of *S. pneumoniae* genomes). (B) Memory usage of each method as a function of dataset size (C) Sensitivity to noise of each method. Relative error against known clusters increases for all methods with increasing amount of mutations in the data. Mean ± standard error of 4 replicates are shown by the error bars. Roary and Panaroo appear as overlapping points, as do CD-HIT and Uclust.

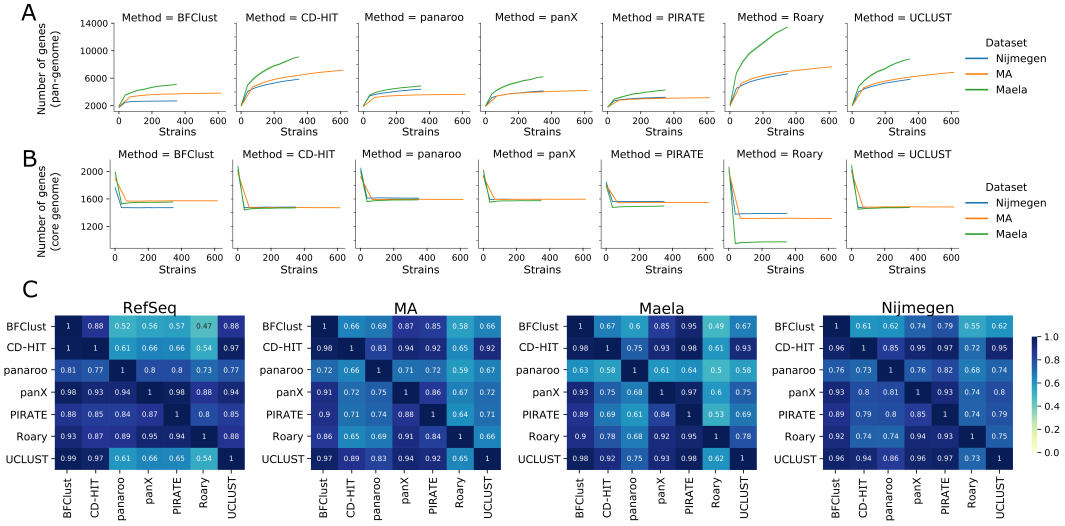

Figure 4.6: (A) Pan-genome size (total number of genes in the pan-genome) as a function of number of strains considered. (B) Core genome size (total number of genes common across strains) as a function of strains considered.(C). Cluster overlap (see methods) between different methods for each dataset. For (A) and (B) mean ± standard error of 10 replicates are shown by the line and error bands.

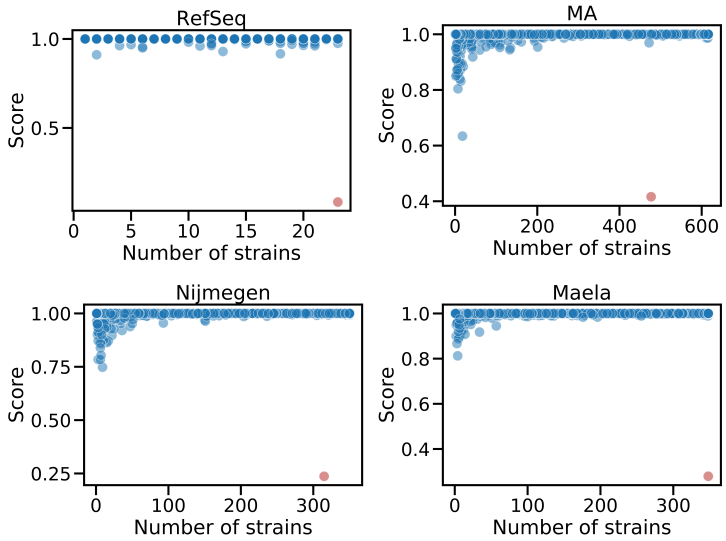

Figure 4.7: Cluster confidence scores for each cluster found using BFClust for 4 *S. pneumoniae* datasets, plotted against the number of strains that share the cluster. In general, the clusters with lower scores appear in the accessory genomes, and are not shared by many strains. There is one cluster within the core genome of each dataset with a low score (red clusters).

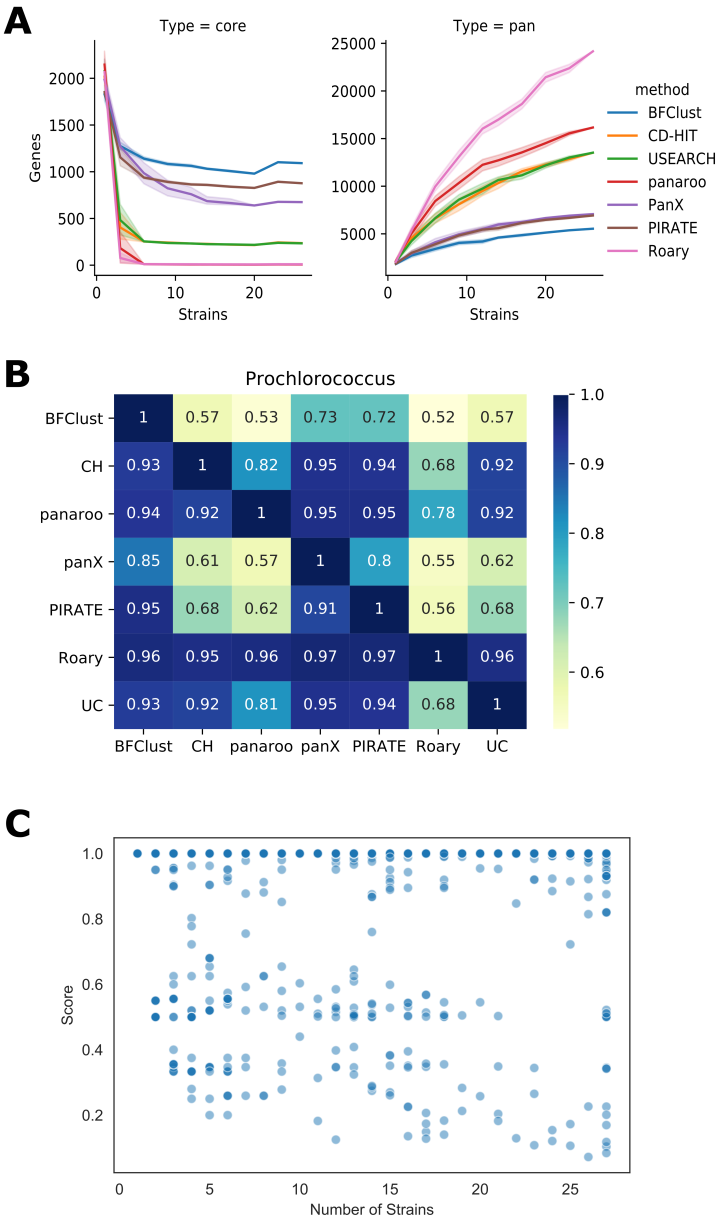

Figure 4.8: (A) Core and pan-genome sizes as a function of number of strains considered. (B) Cluster overlap between each pair of methods (C) Cluster confidence scores for each cluster found using BFClust show many low-confidence clusters.

Method	Runtime (min)	Memory (Gb)	Cluster overlap	Reference	Representative selection	Cluster Augmentation	Confidence score	Network-based clustering
BFClust	2149	40.71	0.419	This work	✓	✓	✓	✓
PanX	258	9.52	0.424	(19)	✗[a]	✗	✗	✓
Roary	45.2	1.86	0.100	(19)	✓	✗	✗	✓
PIRATE	80.7	0.84	0.425	(19)	✓	✗	✗	✓
Panaroo	28.3	1.20	0.100	(19)	✓	✓	✗	✓
CD-HIT	0.94	0.31	0.042	(19)	✓	✓	✗	✗
UCLUST	0.11	0.35	0.045	(19)	✓	✗	✗	✗

Table 4.2: Comparison of software tools applied to pan-genome-wide orthologue clustering. Runtime: time it takes to run each method on 250 genomes (in minutes). Memory: maximum memory usage on the same dataset (in megabytes). Cluster overlap: fraction of clusters in the ground truth that are fully contained within a single cluster identified by the tool, on the synthetic *E. coli* set with substitution rate = 10^{-10}. Representative selection: whether the clustering strategy reduces the input dataset to a small set of representatives before/during clustering. Cluster augmentation: whether the method provides an automatic procedure for adding new sequences to an existing clustering partition. Confidence score: whether the clustering algorithm returns a clustering consensus score as an output. Network-based clustering: whether the method used network-based clustering strategies (MCL).

[a] PanX uses a divide-and-conquer strategy to process large datasets, where batches of 50 genomes are clustered at one time. Representatives from each cluster are selected in each batch. Since representative selection is done after clustering a relatively large set of sequences, we consider this strategy substantially different than other representative selection strategies.

5

Contribution of Population Structure to Predictions of Gene Essentiality and Adaptability *

5.1 Background

Widespread antibiotic use (and misuse) contributes to the emergence and spread of antibiotic resistance. Ways of combating the growing resistance crisis include the development of personalized therapies, new antibiotics, and getting ahead of the curve by predicting future evolutionary trajectories of bacteria likely to acquire resistance. The effectiveness of an antibiotic and future emergence of resistance for a given drug can be predicted using statistical models[21,221,14,89]. In this study, several models are trained for identifying potential

*Contributions: DS devised the study. FR performed the Tn-Seq experiments and gene essentiality determination. KZ and JO performed the experimental evolution and whole genome sequencing experiments. DS designed and led all computational analyses. DS, NG, JW and FR performed statistical and bioinformatic analysis. TvO provided funding and supervision.

narrow-spectrum drug targets or predicting which genes acquire adaptive mutations.

5.1.1 The role of population structure in making predictions

In this chapter, we use the term population structure to describe the genetic differences across strains in the global pneumococcal population. Population structure potentially includes pertinent information for making predictions relevant to mitigating the antibiotic resistance crisis. Predictions on a related context are possible using population structure information. A recent neighbor typing approach can predict the antibiotic susceptibility profile of an isolate based on the susceptibility phenotypes of its closest relatives, with the assumption that genetically similar isolates will have similar antibiotic susceptibility profiles[21]. In this work, the predictive contribution of population structure is explored in the context of gene essentiality and adaptability.

5.1.2 Predicting gene essentiality

It may be possible to identify promising antibiotic targets by studying and predicting the essentiality of genes, as antibiotics often target essential processes in the bacterium. For instance, broad-spectrum antibiotics target processes essential to a large number of species, including Gram-positive and Gram-negative bacteria. An example is tetracycline, which inhibits protein synthesis[39]. However, some genes' essentiality can be context-specific, changing based on the genetic background of a strain. These strain-specific essential genes offer attractive targets for ultra-narrow-spectrum antibiotics i.e. those that can even target different strains or lineages within a species. Therefore, in this work, predictors of gene essentiality in a given genetic background are developed.

5.1.3 Explaining gene essentiality

For strain-specific essential genes, whether a gene is essential in one given strain may depend on the presence or absence of certain genetic elements. For instance, when a *itrA* is absent in *Acinetobacter baumannii*, the *wza-wzb-wzc* operon becomes non-essential, whereas this operon is essential in the presence of *itrA*[11]. The protein product of *itrA* is involved in the earlier steps of capsule biosynthesis, where toxic intermediates are produced. *wza-wzb-wzc* function in later steps, processing the intermediates and exporting them to the surface. Thus, the strain-specific essentiality of the *wza-wzb-wzc* operon can be explained by the accumulation of dead-end metabolites toxic to the bacterium when *itrA* is present[11]. In this work, we attempt to identify new putative interactions of this nature. This is done by training regression models that use gene presence/absence as explanatory variables, and output essentiality for a given gene in different genetic contexts.

5.1.4 Predicting gene adaptability

It is likely that adaptive evolution is also a genetic-context-dependent process, as many experiments have shown the replicability of the order in which mutations are acquired[193,115,82]. Thus, we hypothesized that a predictor of adaptive outcome can be improved using phylogenetic information. This may result in the *a priori* prediction of the antibiotic-adapted strain's genotype, and potentially predict cross-resistance/collateral sensitivity that may arise in the future.

In this work, we address the possibility that population structure may contain relevant information that could aid in the prediction of essential genes (EG) and adapted genes (AG). Presented here are 3 approaches for gene essentiality and adaptation prediction. The

potential contribution of population structure to both gene essentiality and adaptability is evaluated, and it appears that population structure has a stronger contribution in the prediction of EGs. The predictive models presented here will be a first step in developing more personalized therapeutic approaches, and predicting the future adaptive trajectories of clinical strains, based on genomic sequence information.

5.2 Materials and Methods

5.2.1 *Streptococcus pneumoniae* datasets

Two *S. pneumoniae* datasets were used in this study. PG350 is a collection of 350 isolates, obtained at a single hospital in the Netherlands, from invasive pneumococcal disease patients[45]. PGall is a much broader dataset, with 208 genomic sequences coming from 7 distinct sources. The number of strains from each source is roughly the same, in order to avoid any bias through over-representation of one sources. The full strain list and their respective sources can be found here

5.2.2 Phylogenetic analysis

For either *S. pneumoniae* dataset, variant calling was performed using snippy (with default parameters), using TIGR4 as the reference genome. A core genome alignment was generated with snippy-core[164]. A maximum likelihood tree was generated from this core genome alignment using RAxML[175], with GTRGAMMA as the nucleotide substitution model. The core alignment was processed with gubbins[47], in order to remove putative recombination sites, yielding a maximum likelihood tree on the core genome SNPs. Alignments with and

without recombination were converted to pairwise distance matrices across strains using snp-dists[?] .

5.2.3 Transposon-insertion sequencing and gene essentiality determination

Tn-seq experiments were performed by first constructing 6 independent transposon-insertion libraries in each strain. The libraries were then grown in THY followed by DNA extraction and sequencing library preparation as described[198]. Raw reads were processed with the MAGenTA[133] pipeline followed by custom processing[182] to generate wig files. These resulting wig files were analyzed using TRANSIT[50], using the Binomial method to quantify statistical significance. The posterior probabilities for essentiality reported by TRANSIT are thresholded to make the essentiality call.

5.2.4 Rule-based model predicting gene essentiality

For a given strain S and a given gene g whether g is essential in strain S was predicted as follows. First, $S\dagger$, the strain with the closest genetic distance to S, is found based on the SNP distances. If the ortholog of g is absent in $S\dagger$, it is assumed to be non-essential in $S\dagger$ and g in S is predicted to be non-essential as well. Otherwise, $g\dagger$ is the ortholog of g in the genome of $S\dagger$. The essentiality of g in S is predicted to be the same as that of $g\dagger$ in $S\dagger$. As the model does not require parameter training, its performance was evaluated using leave-one-out crossvalidation. Code for this predictor can be found here.

5.2.5 Regression models that identify putative genetic interactions with strain-specific essential genes

For each strain-specific essential gene, the posterior probabilities for essentiality across all 17 strains were obtained using TRANSIT[50]. For each of these genes, a separate model was trained that predicts its posterior probability in a given strain. The input variables for these models were the presence or absence of accessory genes. There is a high number of accessory genes (>2500), which are potential input features for each model. In order to limit the number of features and avoid overfitting, logistic regression models were trained using lasso regularization[221]. The strength of regularization was determined using 5-fold crossvalidation across the entire dataset for each model, as the point where average minimum squared error on the crossvalidation set is minimized. Model training was done in R, using glmnet v3.0-2. Code for regression models can be found here.

5.2.6 *In vitro* adaptive evolution and whole genome sequencing

T4 and 19F were used as parental strains in antibiotic evolution experiments. Four replicate populations were grown in fresh CDM with an increasing concentration of ciprofloxacin, cefepime, levofloxacin, kanamycin, penicillin, rifampicin, or vancomycin for antibiotic adaptation populations. Four additional replicate populations were serial passaged in SDMM as controls to identify background adaptations in the lab culture conditions without antibiotics. From the adapted populations, a single colony was picked from each experiment and checked for its adaptive phenotype by growth curve experiments.

5.2.7 Data driven adapted gene predictor model

Our machine learning approach for AG prediction relies on supervised binary classification methods. The objects to be classified are genes under specific stress conditions; in other words, we classify each gene-condition pair. We used T4 Kanamycin, 19F Kanamycin, T4 Levofloxacin, 19F Levofloxacin, T4 Rifampicin, 19F Rifampicin, T4 Daptomycin, T4 Cefepime, T4 Vancomycin, 19F Vancomycin, D39 Uracil and D39 Valine as the training set (22582 data points); and the T4 Penicillin, T4 Ciprofloxacin and 19F Ciprofloxacin experiments as an independent test set (5738 data points).

The features included are as follows: differential expression at an early (30min) and a late (90-240 min) timepoint (drug/no drug comparison), fitness change (drug/no drug comparison), whether the gene is essential, gene prevalence (number of strains that share a homolog of the gene), gene sequence conservation (average pairwise distance between amino acid sequences of this gene's homologs), gene expression plasticity (variance of expression across diverse environment; as described in[97]), mechanism of action of stress, and gene functional category. The categorical features (such as mechanism of action and gene functional category) are one-hot encoded and all data is standardized such that all scalar features have mean=0 and variance=1. Genes with missing data are omitted.

Six different models were trained on the training set using scikit-learn v0.20.2. Hyperparameter tuning was carried out using GridSearchCV, with 5-fold stratified crossvalidation. The selected models are then evaluated on the test set.

The parameter grid searched and other training details for each model can be found in the supplementary code.

5.2.8 Consensus classification

A consensus classifier was used to enhance the performance of the AG classification. We use a voting classifier, which weighs each of the six models above equally. This model computes a probability of being in the positive class (AG) as the proportion of classifiers that predict the gene to be in the positive class. This probability is then thresholded at 0.5 (i.e. a majority voting classifier). The consensus classifier's performance is also reported on the training and test sets.

5.3 Results

5.3.1 Population structure of *S. pneumoniae*

A collection of *S. pneumoniae* strains from 7 sources were assembled, with the aim of capturing the global genomic diversity of this species. The sources include carriage isolates, and isolates from invasive pneumococcal disease; samples from Europe, Asia, Africa, North America; and samples collected over a span of decades. First, a phylogenetic tree was generated using the core genome SNPs of this diverse dataset (Figure 5.1A). The different datasets were not entirely segregated on the phylogenies, with the exception of the isolates from Malawi. The fact that all isolates from this population appear nearly clonal could potentially be explained with the increased transmission rate in this region[187]. However, when inferred recombination events are removed, the datasets appear to separate more (Figure 5.1B). It is possible that these individual datasets, collected mostly from restricted geographies, are mostly distinguished by their core genomes, and the diversity in their accessory genomes (which is shaped in part by recombination events) is somewhat similar across the

datasets.

There appear to be a few isolates that are separated from the rest of the isolates by a long branch (Figure 5.1A). As these strains are part of the pre-1974 dataset, and considering how this long branch disappears when recombination is removed (Figure 5.1B), it is possible that there was a large recombination event separating these older isolates from the more recent ones, collected mostly in the 2000s.

5.3.2 Essential Genes can be predicted using a simple rule-based model

The first goal of this work was to generate a predictor of gene essentiality using population-level information. We hypothesized that strains that are genotypically more similar will also have similar gene essentiality profiles (essentialomes). In order to test this, a group of 17 strains with Tn-Seq data were considered. Pairwise distances of strains were computed either in terms of SNP distance, or in terms of essentialome distance. Essentialome distance is defined as the proportion of genes that have different essentialities between two strains. There is a modest correlation between SNP distance and essentialome distance (Figure 5.2A), suggesting that this relationship can potentially be used in predicting gene essentiality.

The ground truth in the case of these predictions come from the posterior probabilities computed by Transit, where essential genes have a normalized score near 1 and nonessential genes have a score near 0. Ambiguous cases often have an intermediate score ear 0.5 and are determined to be "Uncertain". Correct predictions in the rule-based for essential and nonessential genes correspond to those with very high and very low scores (Figure 5.3), and are fairly unambiguous. A majority of genes that are incorrectly predicted as essential

do have high z-scores, but are called as uncertain by Transit. Similarly, incorrect predictions of non-essential genes are mostly in the "uncertain" category, but have low z-scores (Figure 5.2B, 5.3B). This suggests that these incorrect predictions for uncertain cases might in fact be correct, however, the essentiality call by Transit was ambiguous due to experimental variability or low Tn-Seq library saturation.

Presented here is a simple, rule-based predictor of gene essentiality. The potential practical use of this model is prediction of gene essentiality in previously unseen strains. This predictor uses the assumption that a gene essential in one strain will be essential in a genetically related strain. In order to predict whether a gene in a given strain is essential, a second strain is selected such that is has the smallest SNP distance to the query strain. If the ortholog of the query gene in this second strain is present, the predictor returns the essentiality call of the ortholog. If the orthologue is absent, it returns "non-essential". This model has an overall accuracy of 92.3% (Defined as total number of correct predictions divided by total number of predictions; confusion matrix shown in Figure 5.2B). As this model is rule based, it does not require any parameter tuning, and operates as a "one size fits all" solution to gene essentiality. However, it does not provide any mechanistic insights as to why a gene becomes essential only in certain contexts.

5.3.3 Essential Gene classification with data driven models can reveal new hypotheses

It is possible to train more detailed models specific to single clusters of orthologous genes. These can potentially be more informative on genetic interactions that influence essentiality. In this section, regression models are trained on accessory gene presence/absence data, in order to classify the essentiality of a different gene. A separate model is trained for each

strain-specific gene. The purpose here is not necessarily to find the best predictive model, but rather potential interactions that explain what causes a gene to be essential in different contexts. Regression models are favored for their interpretability; each explanatory variable (in this case genes) has a coefficient that represents its importance in the classifier. Genes with higher magnitude coefficients have more influence in the classification, and therefore are more likely to have a biologically meaningful relationship with the gene whose essentiality is being predicted. A positive coefficient for a gene indicates that the presence of this gene predicts essentialiy, whereas a negative coefficient indicates the presence is associated with non-essentiality of the strain-specific essential gene. A total of 9 models that had high goodness-of-fit (defined as minimum squared error < 0.1) are included in Table 5.1.

Among this list of gene pairs that potentially include novel interactions, there are 3 possible explanations as to how the explanatory feature results in the essentiality of the strain-specific essential gene. **1.** *glnA* essentiality is associated with 2 transposases. This suggests that a transposable element, when present is rendering the *glnA* gene essential. It may be that the transposable element has been inserted within a gene with redundant function to *glnA*. **2.** Competence system regulator encoding genes *comX1* and *comX2* are associated with multiple strain-specific essential genes; *glnA*, *rplV*, *rpsO*. 2 of these strain-specific essential genes are ribosomal proteins. The competence system has been shown to activate the production of chaperone proteins and proteases in response to various stresses. It is possible that when the competence associated regulators are absent, misfolded proteins cannot be effectively cleared. Thus, accurate translation by the ribosome becomes essential. **3.** RNAse HIII becomes essential when arginosuccinate synthase is absent. The link between the two could be the stalling of ribosomes when arginine synthesis is impaired due

to the absence of arginosuccinate synthase. RNAse H proteins are responsible for degrading RNA that is hybridized to DNA, especially the RNA primers that establish Okazaki fragments during lagging strand replication. When a ribosome stalls, and RNA polymerase continues transcription, the exposed mRNA fragment in between the two has a chance to hybridize with its template DNA strand, and RNAse HIII might be involved in the removal of such hybridizations.

5.3.4 Population structure does not overlap with resistance phenotype

Prior to using population structure data in predictions of adapted genes, the antibiotic resistance phenotypes of isolates coming from the Nijmegen dataset were overlaid on the phylogenetic tree of the same isolates (Figure 5.4). If population structure had a direct influence on how likely a strain is to acquire resistance, resistant strains would be grouped within specific lineages. However, the acquisition of resistance does not appear to be strictly lineage-dependent. Although there are some small clusters where resistance is enriched, overall, resistant strains appear roughly evenly dispersed across the tree. There are also multiple groups of nearly clonal isolates that differ in their antibiotic susceptibility profile (Figure 5.4), suggesting that the emergence of resistance might be governed by other factors such as a single SNP. Therefore, alternative approaches were prioritized over using population structure in the prediction of adapted genes.

5.3.5 Adapted Gene prediction is possible with omics data

Mutations from in-vitro adapted populations were identified through whole genome sequencing of 67 adapted populations (55 experimental and 12 control populations) that

were carried out in 3 *S. pneumoniae* strains, under various antibiotic and nutrient conditions (adapted strains included are the same as in[221]). Adaptive mutations in this dataset are defined as those that reach a >50% frequency in a population evolving in a specific environment and are absent or at <10% frequency in the control condition. Most adaptive mutations appear in coding sequences (referred to as adapted genes, or AGs) as single nucleotide polymorphisms (SNPs) (Figure 5.5A) and are often functionally related to the action of the stress (Supplemental Table 1). For example, adaptive evolution in rifampicin leads to a single AG (*rpoB*) which is the target of this antibiotic. Additionally, strain-specific adaptive patterns emerge during evolution, such as that TIGR4 has more AGs than 19F, and AGs in TIGR4 belong to a more diverse set of functional categories often including capsule metabolism (Figure 5.5B).

Predicting which genes will acquire adaptive mutations is not a trivial task. Pan-genome wide association studies can identify which mutations are linked to resistance phenotype, by considering existing strains. However, they cannot predict AGs. In order to make such predictions, it is important to identify a characteristic of AGs that separate them from non-AGs. We initially hypothesized that the phylogenetic tree constructed from the sequence of a single adapted gene would have a distinctive feature, namely, the susceptible and resistant strains would be separated on the tree. This hypothesis is based on the observation of known resistance-associated alleles of certain genes e.g. S81Y in DNA gyrase[62]. The trees constructed from AGs did not overlap with resistance phenotypes, failing to support this initial hypothesis (Supplementary Dataset 1). This is possibly due to genetic variability within the AG other than the adaptive mutation. The only exception was *pbp2X* (Supplementary Dataset 1), where resistant isolates were those with clearly divergent gene se-

quences. Since resistance phenotype could not be readily explained from the full sequence of the adapted genes, other types of features were considered next.

In order to train a model that predicts whether a given gene, under a given condition will acquire adaptive mutations, a dataset of 41 features is assembled that either pertain to the stress response of the ancestral strain (e.g. RNA-Seq, Tn-Seq), the evolvability of the genome (e.g. sequence conservation) or stress-type (e.g. antibiotic MOA) (Figure 5.5C). Importantly, these features do not immediately reveal any easily recognizable patterns that distinguish AG from non-AGs; and the large amount of data available makes it challenging for a human to pick out such patterns (Figure 5.5C). Therefore, 6 different types of supervised machine learning models were trained. In order to improve performance, all 6 models are combined using a majority voting scheme (consensus model)[52]. All models but logistic regression performed extremely well on the training set (Figure 5.5D). The consensus model correctly identifies 5 out of 16 AGs in the previously unseen test set, with only 4 false positives (non-AGs that are predicted as AGs) and 10 false negatives (AGs that are predicted as non-AGs).

The 5 true positive AGs in the test set are also present as AGs in the training set. One explanation for this overlap could be that the classifier simply picks the same AGs for any experiment with a given MOA. However, if this were the case, all genes observed in the CWSI and DSI experiments in the training set would have been predicted as AG, which would have introduced 15 additional false positives. The consensus model therefore does not simply memorize which genes have appeared as an AG before. To determine which features are most relevant to the consensus model, each feature was omitted one by one from the training and test data, and performance of the consensus model as measured by

Cohen's kappa score (quantifying the performance compared to a random guess that takes into account the prevalence of each class) was re-evaluated. While the "strain" feature information appears to slightly hinder the performance (its omission improves predictions with a decrease in false positives in the test set) (Figure 5.5F), features "MOA" and "gene category" are the most critical, as their omission results in the loss of 3 and 5 true positives respectively. However, simply picking genes of a certain category depending on the MOA would still result in many false positives (as many as there are genes in the relevant category e.g. replication would be the relevant category for DSI).

While our consensus model performs fairly well on making true positive predictions, it is equally important that a model can minimize false positive predictions. By combining the six individual classifiers, the number of false positives is reduced from several hundred (e.g. in logistic regression, decision tree and support vector machine) to 4, suggesting a significant improvement in prediction performance. To determine whether the consensus model suffers from high bias or high variance, a learning curve was computed, tracking performance (Cohen's kappa score) with increasing number of data points (Figure 5.5G). Since the training performance is near perfect regardless of the number of samples used, it is unlikely that the consensus model is suffering from high bias. The cross-validation set performance does improve with increasing the number of data points used to train the model, which is a characteristic of overfit models. Therefore, inclusion of more data points for model training is likely to result in improved performance.

5.4 Discussion

Our initial analysis of the global population structure of *S. pneumoniae* indicated that when recombination events are considered, i.e. when the accessory genome has more influence, isolates collected from different geographies and contexts are more mixed. In contrast, when recombination events are inferred and removed, isolates from different contexts are better separated on the phylogenetic tree (Figure 5.1B). This is in line with the finding that the accessory genomes across different datasets share similarities, which are maintained by negative frequency dependent selection[43].

As population structure has been used to make antibiotic susceptibility predictions, we hypothesized that it could be used to make other types of predictions, such as gene essentiality and adaptability. Population structure appears to contain predictive information for gene essentiality (Figure 5.2), but we were unable to utilize population structure in AG predictions. This is possibly due to the adaptation dataset used here being much more limited in the number of background strains used (3 strains) compared to the essentialome dataset, which had 17 strains. Perhaps future adaptation experiments on more strain backgrounds will reveal whether and how population structure plays a role in predictions of AGs.

For EG predictions, two approaches are presented. The first is similar to the neighbor typing approach used for resistance phenotype predictions[21]. While this approach generated a single model that applies to all genes, which did not require model training, it sacrifices interpretability for practicality. In contrast, though the regression models are not as practical (a model is trained for each strain-specific essential gene, and with few strains it is difficult to evaluate performance), they can point in the right direction when it comes

to discovering new interactions.

Feature selection using lasso regression results in interpretable models with few features. However, the interpretation of these explanatory features requires care, as discussed for similar regression models in Chapter 3. These features are not necessarily the cause of the essentiality/non-essentiality of the query gene; the true explanatory relationship might include another gene whose presence/absence correlates with the selected feature. During regularization, if two genes have the same presence/absence pattern across strains due to synteny or random chance, they have an equal chance at getting selected in the regularized model. Nevertheless, it is straightforward to identify correlating genes, therefore the selected features provide a starting point for looking for relevant interactions between genes.

Similar to the regression models for EG prediction, AG prediction was also done using gene-centric models. This made it possible to readily use Tn-Seq, RNA-Seq, gene essentiality and conservation data. However, this coarse approach does not address multiple mutations that can emerge in a single gene, or mutations in intergenic regions. A future improvement can be making more granular predictions on each codon or nucleotide. This would also increase the number of data points used in the training and test sets, as there are orders of magnitude more codons/nucleotides than there are genes in a given genome. The learning curve in Figure 5.5G suggests that the final consensus model is overfit, which can be addressed with more datapoints, or by using simpler models.

Overall, this work introduces different approaches for EG and AG predictions, utilizing population structure information when possible and appropriate. By showing that it can possibly be used to make EG predictions, we expand the utility of population structure data

beyond making antibiotic susceptibility predictions[21]. As discussed above, there are multiple ways these approaches can be improved and built upon. This work presents a starting point for improved predictive approaches relevant for infection control, and suggests that population structure data offers predictive power in different contexts not explored here.

Model	Strain Specific Essential Gene Cluster	Strain Specific Essential TIGR4 homolog	Strain Specific Essential Functional Annotation	Explanatory Feature Gene Cluster	Explanatory Feature TIGR4 homolog	Explanatory Feature Functional Annotation	Explanatory Feature Coefficient
1	1189	SP_0502	glutamine synthetase glnA	1489	NA	IS5 family transposase	0.027411
1	1189	SP_0502	glutamine synthetase glnA	1711	SP_0055	phosphoribosyl-aminoimidazole carboxylase	-0.3728
1	1189	SP_0502	glutamine synthetase glnA	2219	SP_0695	ThiF family adenylyltrans-ferase	0.023791
1	1189	SP_0502	glutamine synthetase glnA	286	NA	IS30-like element ISSpn8 family transposase	0.372227
1	1189	SP_0502	glutamine synthetase glnA	289	NA	hypothetical protein	-0.01797
1	1189	SP_0502	glutamine synthetase glnA	2893	NA	ABC transporter ATP-binding protein	-0.02226
1	1189	SP_0502	glutamine synthetase glnA	65	SP_2006	sigma-70 family RNA polymerase sigma factor ComX2	-0.02442
2	1629	SP_1298	pApA phospho-diesterase	1011	SP_1251	McrB family protein	-0.06041
2	1629	SP_1298	pApA phospho-diesterase	360	NA	alpha-glycosidase	0.375144
3	2315	SP_2009	Ribosomal protein rpmG1	1711	SP_0055	phosphoribosyl-aminoimidazole carboxylase	-0.03012
3	2315	SP_2009	Ribosomal protein rpmG1	1973	NA	hypothetical protein	0.027878
3	2315	SP_2009	Ribosomal protein rpmG1	329	SP_0109	lactococcin 972 family bacteriocin	-0.05042
4	2343	SP_1178	Glutaredoxin-like protein nrdH	1073	NA	phospho-sugar mutase	-0.02165
4	2343	SP_1178	Glutaredoxin-like protein nrdH	1305	SP_0260	nucleotidyltransferase family protein	0.05492
4	2343	SP_1178	Glutaredoxin-like protein nrdH	908	SP_0312	glycoside hydrolase family 31 protein	0.05894
5	290	SP_0214	Ribosomal protein rplV	65	SP_0014	sigma-70 family RNA polymerase sigma factor comX1	0.175285
6	483	SP_1999	Catabolite control protein cepA	226	SP_1042	CopG family transcriptional regulator	0.30479
	567	SP_0403	ribonuclease HIII	1008	NA	circular bacteriocin, circularin A/uberolysin family	0.024461
7	567	SP_0403	ribonuclease HIII	2386	SP_0113	argininosuccinate synthase	-0.43278
8	643	SP_2107	4-alpha-glucanotransferase malQ	1044	SP_1560	YbbR-like domain-containing protein	-0.1449
9	904	SP_1626	Ribosomal protein rpsO	219	NA	hypothetical protein	-0.11322
9	904	SP_1626	Ribosomal protein rpsO	65	SP_0014	sigma-70 family RNA polymerase sigma factor ComX1	0.021872

Table 5.1: 9 high performing regression models for strain specific essentiality. For both the strain-specific essential genes, and their explanatory features, their ortholog gene cluster, locus tag of the TIGR4 ortholog (when available), and functional annotation is listed. Coefficient can be interpreted as the relative importance of the explanatory gene in the regression model. Only explanatory features with |coefficient|>0.01 are included for brevity.

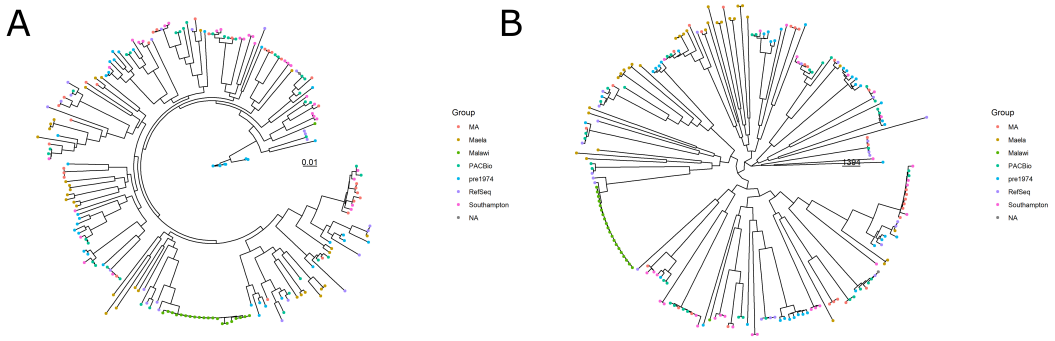

Figure 5.1: Phylogenetic trees with (A) and without (B) recombination were generated based on SNP data and show the global population with multiple sources. The strains are colored by their source, and all sources but Malawi have strains distributed evenly across the tree.

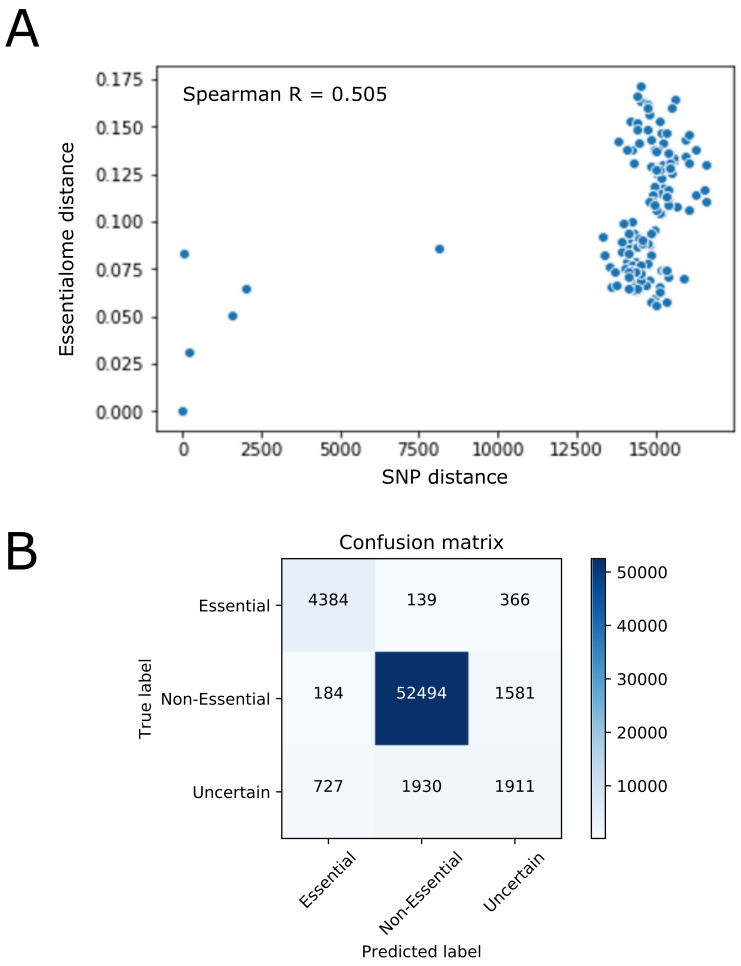

Figure 5.2: (A) Genomic distance (as measured by number of SNPs) and Essentialome distance (defined as proportion of genes with different essentiality calls) have a positive correlation. (B) Confusion matrix of the leave-one-out validation of the general EG predictor. The true label for whether a gene is essential comes from the Transit analysis of Tn-Seq data. Most misclassification errors are made for the “Uncertain” group.

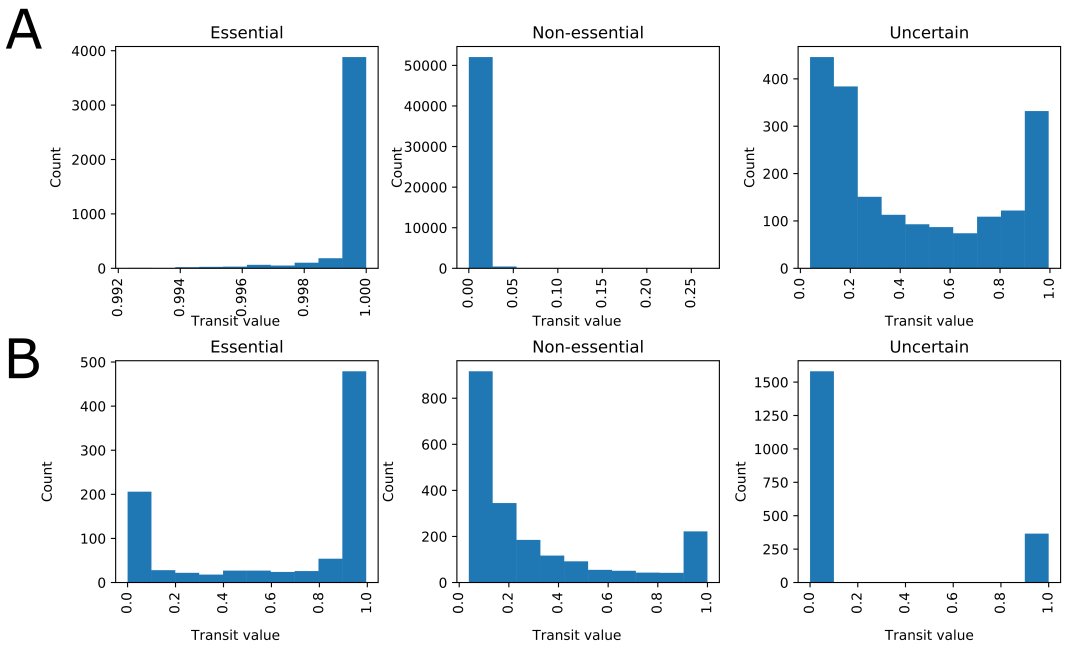

Figure 5.3: Histograms for correct (A) and incorrect (B) predictions show the distribution of the z-score value computed by Transit on experimental data. Histograms are split according to predicted class (essential, non-essential, uncertain). Correctly predicted essential and nonessential genes have high and low values respectively (A). Genes incorrectly predicted as essential often come genes with high z-scores, and genes incorrectly predicted as non-essential often have low z-scores. This is consistent with their predicted label, despite their true label (as reported by Transit) being mostly uncertain (Figure 5.2).

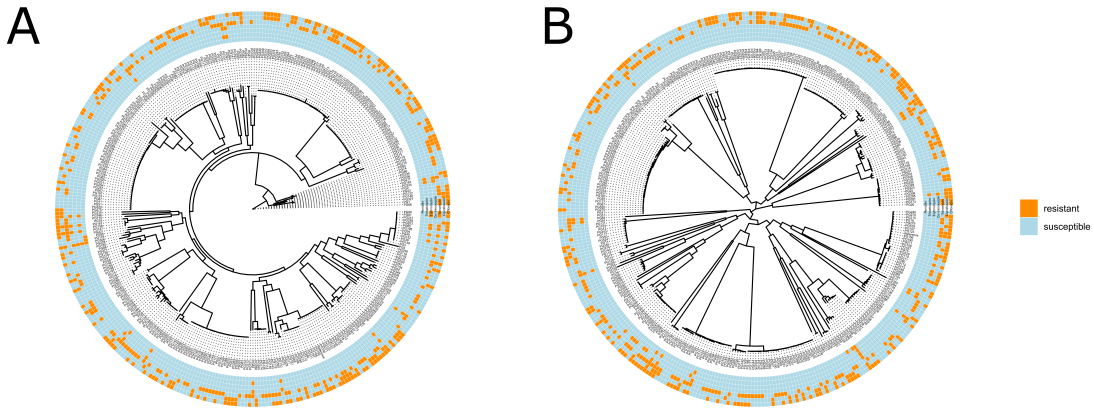

Figure 5.4: Phylogenetic trees of 350 strains from IPD patients in the Netherlands (A) with or (B) without recombination. Each ring around the phylogenetic tree represents the susceptibility of isolates to a different antibiotic.

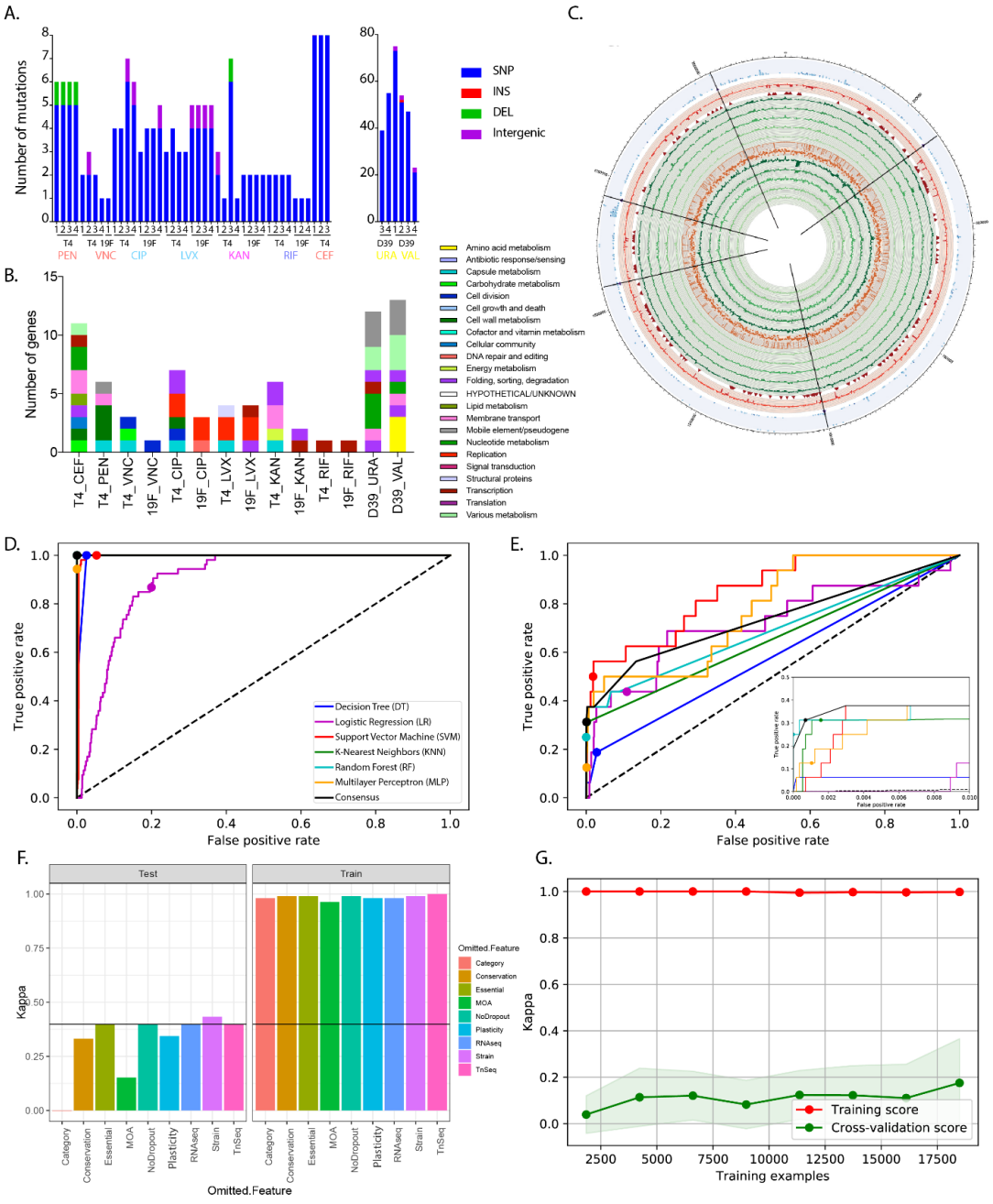
A.
Number of mutations
SNP
INS
DEL
Intergenic
PEN
VNC
CIP
LVX
KAN
RIF
CEF
URA
VAL
B.
Number of genes
T4_CEF
T4_PEN
T4_VNC
19F_VNC
T4_CIP
19F_CIP
T4_LVX
19F_LVX
T4_KAN
19F_KAN
T4_RIF
19F_RIF
D39_URA
D39_VAL
Amino acid metabolism
Antibiotic response/sensing
Capsule metabolism
Carbohydrate metabolism
Cell division
Cell growth and death
Cell wall metabolism
Cofactor and vitamin metabolism
Cellular community
DNA repair and editing
Energy metabolism
Folding, sorting, degradation
HYPOTHETICAL/UNKNOWN
Lipid metabolism
Membrane transport
Mobile element/pseudogene
Nucleotide metabolism
Replication
Signal transduction
Structural proteins
Transcription
Translation
Various metabolism
C.
D.
True positive rate
False positive rate
Decision Tree (DT)
Logistic Regression (LR)
Support Vector Machine (SVM)
K-Nearest Neighbors (KNN)
Random Forest (RF)
Multilayer Perceptron (MLP)
Consensus
E.
True positive rate
False positive rate
F.
Test
Train
Kappa
Omitted.Feature
Category
Conservation
Essential
MOA
NoDropout
Plasticity
RNAseq
Strain
TnSeq
G.
Kappa
Training examples
Training score
Cross-validation score

Figure 5.5: (A) Most observed adaptive mutations are single nucleotide polymorphisms in coding sequences (SNP). Each bar indicates a replicate population. INS: insertion, DEL: deletion in coding sequences. (B) Category distribution of adapted genes (AGs) in each experimental evolution experiment. (C) Data types included in the AG prediction are overlaid on the *S. pneumoniae* TIGR4 chromosome for the VNC experiment. From innermost to outermost tracks: Green bar plots: differential expression of each gene in the parental (inner 5 plots, increasing time points going from innermost to outermost plot) and adapted (outer 5 plots) strains. Orange dot plot: sequence conservation. Orange line plot: sequence prevalence. Red arrows: essential genes. Red bars: fitness change. Purple arrows and black wedges: AGs. Blue dots: frequency of mutation (different shades indicate different replicate populations). (D-E)Receiver-operator characteristic curve for all models evaluated on the training set (D) and test set (E) Selected models are marked with solid dots (E). Inset in (E) shows a zoomed-in region corresponding to the false positive rate of 0-0.01, where the consensus model outperforms all models but random forest. (F) MOA and Category are critical for AG prediction. The consensus model performance is re-evaluated in the absence of each feature in the training and test sets. NoDropout: performance when all features are included. Kappa: Cohen's Kappa score. Horizontal line: performance in NoDropout. (G) Learning curve for the consensus model. Cohen's Kappa score of the model is plotted with increasing number of data points for the training (red) and cross validation (green) sets. Lines and bands indicate mean and standard deviation of accuracy of 10 random splits of the data respectively.

6

Conclusion

This thesis addresses the three challenges outlined in the introduction chapter by taking a systems-biology approach, leveraging the increased availability of multi-omics data (Figure 6.1). In Chapter 2, ShinyOmics is presented as a software solution for the management, rapid exploration and easy sharing of multi-omics datasets. This tool makes it much easier to make comparisons across strains, conditions, and even different omics-screens, and facilitates hypothesis generation. In Chapter 4 I present BFClust, a pan-genome ortholog clustering tool that is unique in reporting confidence scores on its output. Thus, these two chapters and associated software tools enable data pre-processing and exploration that is necessary for addressing the three antibiotic resistance challenges, and form the basis on which the rest of the work presented here rests. A lack of rapid testing is addressed by developing a predictor of bacterial fitness under antibiotic stress, based on transcriptomic entropy in Chapter 3. This approach is unique in its generality, and can be applied to many species, antibiotics, and even non-antibiotic conditions. I address treatment failure, which can be a consequence of antibiotic therapy without testing for susceptibility

and strain-specific resistance, by developing a model that predicts gene essentiality in a strain-specific manner in Chapter 5. When identified, strain-specific essential genes can be used as antimicrobial targets for novel, ultra-narrow-spectrum antibiotics. Development of such antibiotics would allow for more options for treatment, as well as personalized treatment in infectious disease. In this same chapter, I also present an ensemble model that predicts the genotypic changes that are likely to be acquired during evolution towards antibiotic resistance. This type of genotypic predictor can be applied in predictions of the cross-resistance or collateral-sensitivity phenotypes of strains that have yet to arise through antibiotic selection.

6.1 Lack of rapid testing

Existing predictors of antibiotic susceptibility phenotype are often specific to a pathogen-antibiotic pair. More general predictors are preferred if data collection for a new species-antibiotic pair is not feasible. Transcriptomic entropy, as described in Chapter 3, is one such general predictor[221], that relies on the intuitive idea that when an organism is experiencing stress it cannot overcome (e. g. a susceptible pathogen challenged with an antibiotic), its gene expression patterns overall will be more chaotic. Entropy has been demonstrated to work for previously unseen species and antibiotics, and has the potential to extend well beyond antibiotic stress[221]. Moreover, its demonstrated generality makes the entropy-based predictor less likely to be confounded by factors such as population structure and limitations in dataset size and diversity[206]. An alternative way to address this problem is to use population structure as the central feature to predict resistance phenotype. Břinda *et al.* have developed a method that can infer phylogenetic lineage based on genomic kmer

content, and makes a prediction on the phenotype based on the phenotype of other members of that lineage[21]. While it can achieve high performance, this is contingent on having a comprehensive reference database of genomic information, since the predictions on a given isolate rely on finding genotypically similar isolates and lineages.

It is possible to apply entropy-based fitness predictors in the context of and active infection, where there is stress imposed on the pathogen by the host immune system (Figure 6.1). Conversely, during an active infection, the host can also experience stress from inflammation. Thus, entropy has the potential to be applied to the pathogen and host simultaneously in order to monitor an active infection. Moreover, both entropy and marker-based predictors that are condition-specific, have the potential to be applied to any other kind of omics data.

There are also possibilities for alternatives for current AST that do not rely on making predictions. The main drawbacks of existing AST methods is the time, labor, and cost of the assays. These culture-based assays can be miniaturized to be performed in microfluidics devices, reducing the amount of reagents necessary by orders of magnitude, as well as reducing the time to acquiring the result, as growth can be closely monitored under a microscope (Figure 6.1)

6.2 Treatment failure

One way of combating treatment failure is to incorporate personalized medicine in infectious disease treatment. This can be accomplished through the use of antibiotics that are narrow-spectrum, targeting specific strains instead of affecting the entire microbiota with broad-spectrum antibiotics that may or may not be effective against the infection caus-

ing agent. In Chapter 5, targeting strain-specific essential genes is proposed as a starting point for developing ultra narrow-spectrum antibiotics. While predicting and targeting strain-specific essential genes is a potential avenue for developing personalized treatments, treatment failure can be addressed through multiple avenues, which can take promising future directions.

The effectiveness of single drugs is reduced as resistance becomes more widespread. However, the combination of multiple drugs can be more than the sum of its parts. The vast number of potential combinations is narrowed down by computational predictions of successful combinations, which are later validated. This approach is in its infancy in microbiology, with lots of room for new development. Specifically, there is a body of cancer literature that focus on data-driven models for predicting synergistic combination treatments[196], which can be applied to microbiology.

While addressing treatment failure with the use of multi-drug combinations is an option, it poses an increased risk due to exposing a patient's microbiota to more antibiotics. An advantage of computational screens that help identify drug synergy is that they can screen any combination of antibiotic and non-antibiotic stress. Instead of multiple drug combinations, predictions of antimicrobial efficacy can be made on drug-metabolic stress combinations[177]. Finding effective antimicrobial combinations of non-antibiotic stresses would reduce the overall use of antibiotics. With reduced exposure, the likelihood of adaptive evolution towards drug resistance is also lowered.

Treatment failure is caused not only by mixed populations of resistant and susceptible strains, but also persisters[12]. Persistence can be detected through biphasic kill curves, but it may be possible to train data-driven models that can predict persistence. For example,

the *hipA* gene in *Escherichia coli* is associated with a high persistence phenotype[107], which can be used as a genetic feature in a rule-based model for persistence prediction. There may well be transcriptomic or metabolomic signatures as well that can be utilized similarly.

Lastly, the unique interactions between the host and the pathogen can provide information that is specific for each infection case (Figure 6.1). It may soon be possible to compute entropy on the host and pathogen simultaneously during an active infection. We can hypothesize that an infection that is being cleared (e.g. after antibiotic treatment), the pathogen's entropy rises, and the host - whose stress is alleviated through the clearing of the pathogen - demonstrates a decrease in entropy. Whereas when an antibiotic treatment is not successful, the host is under sustained stress and demonstrates high entropy, while the pathogen has high fitness and thus low entropy. One can imagine testing these hypotheses by performing dual-RNA-Seq on the host and pathogen at several timepoints post treatment.

6.3 Evolution of antibiotic resistance

In Chapter 5, I present an ensemble model that predicts whether a gene acquires adaptive mutations in a given adaptation experiment. While there are relatively few similar studies that focus on predicting the genotype of an adapted, antibiotic resistant strain, I anticipate these types of predictions becoming continually improved as new omics and adaptive evolution data are collected. The existing approaches focus on *de novo* mutations, however horizontal gene transfer (HGT) also plays an important role in the acquisition of resistance. Predictive models in the future that take HGT into account would therefore be more comprehensive, and potentially have improved performance.

Another factor that plays an important role in adaptive evolution is epistasis, i.e. interactions between genetic elements. Epistasis has been known to constrain evolution[23,77,4,156]. It is also possible to use epistatic interactions and fitness landscapes to predict which evolutionary trajectories are most likely[208], and use this information to predict collateral sensitivity/cross-resistance[142]; or even to control/steer evolutionary trajectories[141,94]. Epistatic interactions between genetic elements form a network, which can be incorporated into the ensemble model presented in Chapter 5, possibly improving performance (Figure 6.1).

6.4 The path forward

Improving technology, both experimentally and computationally, is paving the way to new diagnostic, prognostic and predictive approaches. The implementation of the ongoing research in clinical settings will allow for the determination of the most effective antibiotic treatment, reducing the misuse of antibiotics. Predictive models on resistance and novel antimicrobial design enables addressing resistance that might emerge in the future, in addition to expanding the molecular armory against pathogens. By addressing the antibiotic resistance crisis through multiple angles, antibiotic stewardship practices can be improved and novel therapeutic strategies can be developed simultaneously. The contents of this thesis, as well as the works cited, demonstrate the explosion in the amount of data, and how it can quickly be turned into valuable knowledge. While the prevalence of antibiotic resistance is increasing, there are also significant advancements in infectious disease diagnostics, epidemiology and evolutionary biology, partially due to the current COVID-19 pandemic. Through interdisciplinary work bringing together high-throughput screen development, systems biology, statistics and bioinformatics, it is very much possible that the

antibiotic resistance crisis can be mitigated.

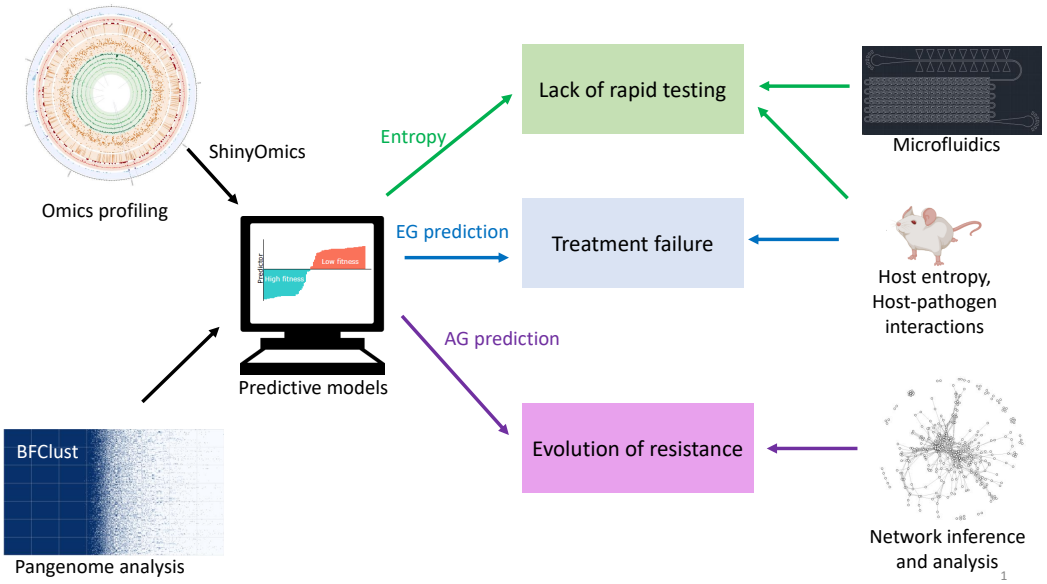

Figure 6.1: The focus of this thesis is the development of software and statistical models that predict several outcomes that pertain to antibiotic resistance. ShinyOmics and BFClust (Chapters 2 and 4 respectively) provide the link between data generation and predictive model development. These two tools provide data pre-processing, management, and quality control steps that are crucial for developing high-performing predictive models. The three main types of predictive models developed in this thesis are fitness prediction using entropy (Chapter 3), essential gene prediction using population structure (Chapter 5), and the prediction of adapted genes (Chapter 5). These models each address a different antibiotic resistance challenge. It is also possible to expand beyond the models in this thesis, and address the same challenges through improved and/or alternative techniques such as microfluidics assay development, examining the host-pathogen interactions, and studying genetic interaction networks.

A

Supplementary Information for Boundary-Forest Clustering: Large-Scale Consensus Clustering of Biological Sequences

A.1 Glossary of Terms

Cluster augmentation addition of new sequences to an already clustered set of sequences. New sequences are assigned a cluster based on their similarity to existing clusters. The existing clusters are not altered.

Direct-threshold methods clustering methods that rely on a single sequence similarity threshold that directly impact the clustering (e.g. CD-HIT, UCLUST). Sequences that are more similar than the threshold value are allowed to be in the same cluster, whereas a pair

of sequences that are more dissimilar than the threshold cannot be in the same cluster. These methods do have other parameters that can be refined, but the software authors do provide default and recommended values for those other parameters. We use these default values in all of our experiments.

Representative selection A step performed before or during clustering, that reduces redundancy in the full dataset. Sequences that are extremely similar to each other are grouped and only one sequence (the representative) from each group is used in clustering.

Representative (sequence) A sequence that is selected during the representative selection step.

Clustering ensemble a collection of clustering results for the same input data. For instance, in our method, the clustering done on the representatives of each Boundary-Tree result in a collection of 10 clustering outputs.

Consensus clustering a means to combine a clustering ensemble and output a single clustering result

Cluster confidence score a value assigned to each cluster in the consensus clustering output that is between 0 and 1, and that represents the relative agreement between clustering outputs across an ensemble.

Item confidence score a value assigned to each item (i.e. sequence) in the consensus clustering output that is between 0 and 1, and that represents the relative agreement between clustering outputs across an ensemble.

Random sampling A representative selection strategy where representatives are selected at random.

Naïve sampling A representative selection strategy where the set of representatives is

built by sequentially reading the input dataset, and adding to the representatives any new sequence that is sufficiently dissimilar (defined by a threshold) to the currently selected representatives.

Consensus index The consensus index of a pair of sequences *i* and *j* (belonging to the same consensus cluster) is the number of times that they appear together in the same cluster associated to one of n Boundary-Tree, divided by the number n of Boundary-Trees used.

Cluster extension Extrapolating the cluster assignments of the representatives to the full dataset.

A.2 Boundary Forest Pseudocode

Data: **seqs** (the set of sequences, N = number of sequences)

t (sequence similarity threshold)

maxChild (maximum number of children allowed on each node)

Result: **bt** (Boundary-Tree)

shuffle seqs;

add seqs[1] as the root node of bt;

add seqs[2] to the bt as the child of the root node;

for *query in seqs[3: N]* **do**

  while *True* **do**

    currentNode ← root node of bt;

    if *dist(query, currentNode)* < *t* **then**

      query's representative ← currentNode;

      break;

    else

      children ← children of currentNode;

      closestChild ← $argmin_{v \in children} dist(query, v)$;

      smallestDist ← $min_{v \in children} dist(query, v)$;

      if *dist(query, currentNode)* < *smallestDist && size(children)* < *maxChild*

        then

          add query as a child to currentNode;

      else

          currentNode ← closestChild;

      end

    end

  end

end

Where *dist(x,y)* calculates the Smith-Waterman distance between sequences *x* and *y*

References

[1] Adam, B.-L., Qu, Y., Davis, J. W., Ward, M. D., Clements, M. A., Cazares, L. H., Semmes, O. J., Schellhammer, P. F., Yasui, Y., Feng, Z., and et al. Serum protein fingerprinting coupled with a pattern-matching algorithm distinguishes prostate cancer from benign prostate hyperplasia and healthy men. *Cancer Research 62*, 13 (Jul 2002), 3609–3614.

[2] Ahmed, N., and Gokhale, D. Entropy expressions and their estimators for multivariate distributions. *IEEE Transactions on Information Theory 35*, 3 (May 1989), 688–692.

[3] Almende B. V., Benoit Thieurmel, R. T. *visNetwork: Network Visualization using "vis.js" Library*. Aug 2019.

[4] Angst, D. C., and Hall, A. R. The cost of antibiotic resistance depends on evolutionary history in escherichia coli. *BMC Evolutionary Biology 13* (Aug 2013), 163.

[5] Anjo, S. I., Santa, C., and Manadas, B. Swath-ms as a tool for biomarker discovery: From basic research to clinical applications. *PROTEOMICS 17*, 3–4 (2017), 1600278.

[6] Aprianto, R., Slager, J., Holsappel, S., and Veening, J.-W. Time-resolved dual rna-seq reveals extensive rewiring of lung epithelial and pneumococcal transcriptomes during early infection. *Genome Biology 17*, 198 (Sep 2016).

[7] Arthur, D., and Vassilvitskii, S. K-means++: The advantages of careful seeding. In *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms* (2007), SODA '07, Society for Industrial and Applied Mathematics, p. 1027–1035. event-place: New Orleans, Louisiana.

[8] Au, N., Kuester-Schoeck, E., Mandava, V., Bothwell, L. E., Canny, S. P., Chachu, K., Colavito, S. A., Fuller, S. N., Groban, E. S., Hensley, L. A., and et al. Genetic composition of the bacillus subtilis sos system. *Journal of Bacteriology 187*, 22 (Nov 2005), 7655–7666.

[9] Azarian, T., Martinez, P. P., Arnold, B. J., Qiu, X., Grant, L. R., Corander, J., Fraser, C., Croucher, N. J., Hammitt, L. L., Reid, R., and et al. Frequency-dependent selection can forecast evolution in streptococcus pneumoniae. *PLOS Biology 18*, 10 (Oct 2020), e3000878.

[10] Baharoglu, Z., and Mazel, D. Sos, the formidable strategy of bacteria against aggressions. *FEMS Microbiology Reviews 38*, 6 (Nov 2014), 1126–1145.

[11] Bai, J., Dai, Y., Farinha, A., Tang, A. Y., Syal, S., Vargas-Cuebas, G., Surujon, D., Isberg, R. R., Opijnen, T. v., and Geisinger, E. Essential gene analysis in acinetobacter baumannii by high-density transposon mutagenesis and crispr interference. *bioRxiv* (Sep 2020), 2020.09.15.299016.

[12] Balaban, N. Q., Helaine, S., Lewis, K., Ackermann, M., Aldridge, B., Andersson, D. I., Brynildsen, M. P., Bumann, D., Camilli, A., Collins, J. J., and et al. Definitions and guidelines for research on antibiotic persistence. *Nature Reviews Microbiology 17*, 7 (Jul 2019), 441–448.

[13] Bantscheff, M., Lemeer, S., Savitski, M. M., and Kuster, B. Quantitative mass spectrometry in proteomics: critical review update from 2007 to the present. *Analytical and Bioanalytical Chemistry 404*, 4 (Sep 2012), 939–965.

[14] Barczak, A. K., Gomez, J. E., Kaufmann, B. B., Hinson, E. R., Cosimi, L., Borowsky, M. L., Onderdonk, A. B., Stanley, S. A., Kaur, D., Bryant, K. F., and et al. Rna signatures allow rapid identification of pathogens and antibiotic susceptibilities. *Proceedings of the National Academy of Sciences of the United States of America 109*, 16 (Apr 2012), 6217–6222.

[15] Battesti, A., and Bouveret, E. Acyl carrier protein/spot interaction, the switch linking spot-dependent stress response to fatty acid metabolism. *Molecular Microbiology 62*, 4 (2006), 1048–1063.

[16] Bayliss, S. C., Thorpe, H. A., Coyle, N. M., Sheppard, S. K., and Feil, E. J. Pirate: A fast and scalable pangenomics toolbox for clustering diverged orthologues in bacteria. *GigaScience 8*, 10 (Oct 2019).

[17] Bentley, D. R., Balasubramanian, S., Swerdlow, H. P., Smith, G. P., Milton, J., Brown, C. G., Hall, K. P., Evers, D. J., Barnes, C. L., Bignell, H. R., and et al. Accurate whole human genome sequencing using reversible terminator chemistry. *Nature 456*, 7218 (Nov 2008), 53–59.

[18] Bhardwaj, T., and Somvanshi, P. Pan-genome analysis of clostridium botulinum reveals unique targets for drug development. *Gene 623* (Aug 2017), 48–62.

[19] Bhattacharyya, R. P., Bandyopadhyay, N., Ma, P., Son, S. S., Liu, J., He, L. L., Wu, L., Khafizov, R., Boykin, R., Cerqueira, G. C., and et al. Simultaneous detection of

genotype and phenotype enables rapid and accurate antibiotic susceptibility determination. *Nature Medicine 25* (Nov 2019), 1–7.

[20] Biller, S. J., Berube, P. M., Berta-Thompson, J. W., Kelly, L., Roggensack, S. E., and Awad, L. e. a. Genomes of diverse isolates of the marine cyanobacterium prochlorococcus. *Scientific Data 1*, 11 (Sep 2014), 140034.

[21] Břinda, K., Callendrello, A., Ma, K. C., MacFadden, D. R., Charalampous, T., Lee, R. S., Cowley, L., Wadsworth, C. B., Grad, Y. H., Kucherov, G., and et al. Rapid inference of antibiotic resistance and susceptibility by genomic neighbour typing. *Nature Microbiology 5* (Feb 2020), 1–10.

[22] Blattman, S. B., Jiang, W., Oikonomou, P., and Tavazoie, S. Prokaryotic single-cell rna sequencing by in situ combinatorial indexing. *Nature Microbiology 5* (May 2020), 1–10.

[23] Blount, Z. D., Borland, C. Z., and Lenski, R. E. Historical contingency and the evolution of a key innovation in an experimental population of escherichia coli. *Proceedings of the National Academy of Sciences 105*, 23 (Jun 2008), 7899–7906.

[24] Boshoff, H. I. M., Myers, T. G., Copp, B. R., McNeil, M. R., Wilson, M. A., and Barry, C. E. The transcriptional responses of mycobacterium tuberculosis to inhibitors of metabolism novel insights into drug mechanisms of action. *Journal of Biological Chemistry 279*, 38 (Sep 2004), 40174–40184.

[25] Boutte, C. C., and Crosson, S. Bacterial lifestyle shapes stringent response activation. *Trends in Microbiology 21*, 4 (Apr 2013), 174–180.

[26] Brook, I. Inoculum Effect. *Reviews of Infectious Diseases 11*, 3 (05 1989), 361–368.

[27] Burnham, C.-A. D., Leeds, J., Nordmann, P., O'Grady, J., and Patel, J. Diagnosing antimicrobial resistance. *Nature Reviews Microbiology 15*, 11 (Nov 2017), 697–703.

[28] Cai, T. T., Liang, T., and Zhou, H. H. Law of log determinant of sample covariance matrix and optimal estimation of differential entropy for high-dimensional gaussian distributions. *Journal of Multivariate Analysis 137* (May 2015), 161–172.

[29] Campos, A. I., and Zampieri, M. Metabolomics-driven exploration of the chemical drug space to predict combination antimicrobial therapies. *Molecular Cell 74*, 6 (Jun 2019), 1291–1303.e6.

[30] Cangelosi, R., and Goriely, A. Component retention in principal component analysis with application to cdna microarray data. *Biology Direct 2*, 1 (Jan 2007), 2.

[31] CDC. Antibiotic resistance threats in the united states. Tech. rep., U.S. Department of Health and Human Services, CDC; 2019, Atlanta, GA, 2019.

[32] Chandrasekaran, S., Cokol-Cakmak, M., Sahin, N., Yilancioglu, K., Kazan, H., Collins, J. J., and Cokol, M. Chemogenomics and orthology-based design of antibiotic combination therapies. *Molecular Systems Biology 12*, 5 (May 2016), 872.

[33] Chang, W., Cheng, J., Allaire, J. J., Xie, Y., McPherson, J., jQuery Foundation, jQuery contributors, jQuery UI contributors, and Otto, M. *shiny: Web Application Framework for R*. Apr 2019.

[34] Chatterjee, A., Saranath, D., Bhatter, P., and Mistry, N. Global transcriptional profiling of longitudinal clinical isolates of mycobacterium tuberculosis exhibiting rapid accumulation of drug resistance. *PLOS ONE 8*, 1 (Jan 2013), e54717.

[35] Chaudhari, N. M., Gupta, V. K., and Dutta, C. Bpga- an ultra-fast pan-genome analysis pipeline. *Scientific Reports 6*, 24373 (Apr 2016).

[36] Cheng, S., Karkar, S., Bapteste, E., Yee, N., Falkowski, P., and Bhattacharya, D. Sequence similarity network reveals the imprints of major diversification events in the evolution of microbial life. *Frontiers in Ecology and Evolution 2* (2014), 72.

[37] Chewapreecha, C., Harris, S. R., Croucher, N. J., Turner, C., Marttinen, P., Cheng, L., Pessia, A., Aanensen, D. M., Mather, A. E., Page, A. J., and et al. Dense genomic sampling identifies highways of pneumococcal recombination. *Nature genetics 46*, 3 (Mar 2014), 305–309.

[38] Choi, J., Pacheco, C. M., Mosbergen, R., Korn, O., Chen, T., Nagpal, I., Englart, S., Angel, P. W., and Wells, C. A. Stemformatics: visualize and download curated stem cell data. *Nucleic Acids Research 47*, D1 (Jan 2019), D841–D846.

[39] Chopra, I., and Roberts, M. Tetracycline antibiotics: Mode of action, applications, molecular biology, and epidemiology of bacterial resistance. *Microbiology and Molecular Biology Reviews 65*, 2 (Jun 2001), 232–260.

[40] Cloonan, N., Forrest, A. R. R., Kolle, G., Gardiner, B. B. A., Faulkner, G. J., Brown, M. K., Taylor, D. F., Steptoe, A. L., Wani, S., Bethel, G., and et al. Stem cell transcriptome profiling via massive-scale mrna sequencing. *Nature Methods 5*, 7 (Jul 2008), 613–619.

[41] Cock, P. J. A., Antao, T., Chang, J. T., Chapman, B. A., Cox, C. J., Dalke, A., Friedberg, I., Hamelryck, T., Kauff, F., Wilczynski, B., and et al. Biopython: freely available python tools for computational molecular biology and bioinformatics. *Bioinformatics 25*, 11 (Jun 2009), 1422–1423.

[42] Cokol, M., Kuru, N., Bicak, E., Larkins-Ford, J., and Aldridge, B. B. Efficient measurement and factorization of high-order drug interactions in mycobacterium tuberculosis. *Science Advances 3*, 10 (Oct 2017), e1701881.

[43] Corander, J., Fraser, C., Gutmann, M. U., Arnold, B., Hanage, W. P., Bentley, S. D., Lipsitch, M., and Croucher, N. J. Frequency-dependent selection in vaccine-associated pneumococcal population dynamics. *Nat Ecol Evol 1* (Dec 2017), 1950.

[44] Costanzo, M., VanderSluis, B., Koch, E. N., Baryshnikova, A., Pons, C., Tan, G., Wang, W., Usaj, M., Hanchard, J., Lee, S. D., and et al. A global genetic interaction network maps a wiring diagram of cellular function. *Science 353*, 6306 (Sep 2016), aaf1420.

[45] Cremers, A. J. H., Mobegi, F. M., de Jonge, M. I., van Hijum, S. A. F. T., Meis, J. F., Hermans, P. W. M., Ferwerda, G., Bentley, S. D., and Zomer, A. L. The post-vaccine microevolution of invasive streptococcus pneumoniae. *Scientific Reports 5* (Oct 2015), 14952.

[46] Croucher, N. J., Finkelstein, J. A., Pelton, S. I., Mitchell, P. K., Lee, G. M., Parkhill, J., Bentley, S. D., Hanage, W. P., and Lipsitch, M. Population genomics of post-vaccine changes in pneumococcal epidemiology. *Nature Genetics 45*, 6 (Jun 2013), 656–663.

[47] Croucher, N. J., Page, A. J., Connor, T. R., Delaney, A. J., Keane, J. A., Bentley, S. D., Parkhill, J., and Harris, S. R. Rapid phylogenetic analysis of large samples of recombinant bacterial whole genome sequences using gubbins. *Nucleic Acids Research 43*, 3 (Feb 2015), e15–e15.

[48] Csardi, G., and Nepusz, T. The igraph software package for complex network research. *InterJournal Complex Systems* (2006), 1695.

[49] Davies, T. J., Stoesser, N., Sheppard, A. E., Abuoun, M., Fowler, P., Swann, J., Quan, T. P., Griffiths, D., Vaughan, A., Morgan, M., and et al. Reconciling the potentially irreconcilable? genotypic and phenotypic amoxicillin-clavulanate resistance in escherichia coli. *Antimicrobial Agents and Chemotherapy 64*, 6 (2020).

[50] DeJesus, M. A., Ambadipudi, C., Baker, R., Sassetti, C., and Ioerger, T. R. Transit - a software tool for himar1 tnseq analysis. *PLOS Computational Biology 11*, 10 (Oct 2015), e1004401.

[51] di Bernardo, D., Thompson, M. J., Gardner, T. S., Chobot, S. E., Eastwood, E. L., Wojtovich, A. P., Elliott, S. J., Schaus, S. E., and Collins, J. J. Chemogenomic profiling on a genome-wide scale using reverse-engineered gene networks. *Nature Biotechnology 23*, 33 (Mar 2005), 377–383.

[52] Dietterich, T. G. *Ensemble Methods in Machine Learning*, vol. 1857. Springer Berlin Heidelberg, 2000, p. 1–15.

[53] Ding, W., Baumdicker, F., and Neher, R. A. panx: pan-genome analysis and exploration. *Nucleic Acids Research 46*, 1 (Jan 2018), e5.

[54] Donati, C., Hiller, N. L., Tettelin, H., Muzzi, A., Croucher, N. J., Angiuoli, S. V., Oggioni, M., Dunning Hotopp, J. C., Hu, F. Z., Riley, D. R., and et al. Structure and dynamics of the pan-genome of streptococcus pneumoniae and closely related species. *Genome Biology 11* (2010), R107.

[55] Dunbar, S. A. Applications of luminex® xmaptm technology for rapid, high-throughput multiplexed nucleic acid detection. *Clinica Chimica Acta 363*, 1 (Jan 2006), 71–82.

[56] Edgar, R. C. Search and clustering orders of magnitude faster than blast. *Bioinformatics 26*, 19 (Oct 2010), 2460–2461.

[57] Enright, A. J., Van Dongen, S., and Ouzounis, C. A. An efficient algorithm for large-scale detection of protein families. *Nucleic Acids Research 30*, 7 (Apr 2002), 1575–1584.

[58] Erill, I., Campoy, S., and Barbé, J. Aeons of distress: an evolutionary perspective on the bacterial sos response. *FEMS Microbiology Reviews 31*, 6 (Nov 2007), 637–656.

[59] Falzon, D., Schünemann, H. J., Harausz, E., González-Angulo, L., Lienhardt, C., Jaramillo, E., and Weyer, K. World health organization treatment guidelines for drug-resistant tuberculosis, 2016 update. *European Respiratory Journal 49*, 3 (Mar 2017).

[60] Fang, X., Lloyd, C. J., and Palsson, B. O. Reconstructing organisms in silico: genome-scale models and their emerging applications. *Nature Reviews Microbiology 18* (Sep 2020), 1–13.

[61] Fang, X., Sastry, A., Mih, N., Kim, D., Tan, J., Yurkovich, J. T., Lloyd, C. J., Gao, Y., Yang, L., and Palsson, B. O. Global transcriptional regulatory network for escherichia coli robustly connects gene expression to transcription factor activities. *Proceedings of the National Academy of Sciences 114*, 38 (Sep 2017), 10286–10291.

[62] Fàbrega, A., Madurga, S., Giralt, E., and Vila, J. Mechanism of action of and resistance to quinolones. *Microbial biotechnology 2*, 1 (Jan 2009), 40–61.

[63] Ferrés, I., Fresia, P., and Iraola, G. simurg: simulate bacterial pangenomes in r. *Bioinformatics 36*, 4 (Feb 2020), 1273–1274.

[64] Flusberg, B. A., Webster, D. R., Lee, J. H., Travers, K. J., Olivares, E. C., Clark, T. A., Korlach, J., and Turner, S. W. Direct detection of dna methylation during single-molecule, real-time sequencing. *Nature Methods 7*, 66 (Jun 2010), 461–465.

[65] Fouts, D. E., Brinkac, L., Beck, E., Inman, J., and Sutton, G. Panoct: automated clustering of orthologs using conserved gene neighborhood for pan-genomic analysis of bacterial strains and closely related species. *Nucleic Acids Research 40*, 22 (Dec 2012), e172–e172.

[66] Freschi, L., Vincent, A. T., Jeukens, J., Emond-Rheault, J.-G., Kukavica-Ibrulj, I., Dupont, M.-J., Charette, S. J., Boyle, B., and Levesque, R. C. The pseudomonas aeruginosa pan-genome provides new insights on its population structure, horizontal gene transfer, and pathogenicity. *Genome Biology and Evolution 11*, 1 (Jan 2019), 109–120.

[67] Friedman, J., Hastie, T., and Tibshirani, R. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics 9*, 3 (Jul 2008), 432–441.

[68] Friedman, J., Hastie, T., and Tibshirani, R. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software 33*, 1 (2010), 1–22.

[69] Furusawa, C., Horinouchi, T., and Maeda, T. Toward prediction and control of antibiotic-resistance evolution. *Current Opinion in Biotechnology 54* (Dec 2018), 45–49.

[70] Galagan, J. E., Minch, K., Peterson, M., Lyubetskaya, A., Azizi, E., Sweet, L., Gomes, A., Rustad, T., Dolganov, G., Glotova, I., and et al. The mycobacterium tuberculosis regulatory network and hypoxia. *Nature 499*, 7457 (Jul 2013), 178–183.

[71] Galili, Tal, O'Callaghan, Alan, Sidi, Jonathan, Sievert, and Carson. heatmaply: an r package for creating interactive cluster heatmaps for online publishing. *Bioinformatics 34* (Oct 2017), 1600–1602.

[72] Gallagher, L. A., Shendure, J., and Manoil, C. Genome-scale identification of resistance functions in pseudomonas aeruginosa using tn-seq. *mBio 2*, 1 (Mar 2011), e00315–10.

[73] Geiss, G. K., Bumgarner, R. E., Birditt, B., Dahl, T., Dowidar, N., Dunaway, D. L., Fell, H. P., Ferree, S., George, R. D., Grogan, T., and et al. Direct multiplexed measurement of gene expression with color-coded probe pairs. *Nature Biotechnology 26*, 33 (Mar 2008), 317–325.

[74] Gerding, D. N., File, T. M., and McDonald, L. C. Diagnosis and treatment of clostridium difficile infection (cdi). *Infectious diseases in clinical practice (Baltimore, Md.) 24*, 1 (Jan 2016), 3–10.

[75] Ghattargi, V. C., Gaikwad, M. A., Meti, B. S., Nimonkar, Y. S., Dixit, K., Prakash, O., Shouche, Y. S., Pawar, S. P., and Dhotre, D. P. Comparative genome analysis reveals key genetic factors associated with probiotic property in enterococcus faecium strains. *BMC Genomics 19*, 652 (Sep 2018).

[76] Ghazalpour, A., Bennett, B., Petyuk, V. A., Orozco, L., Hagopian, R., Mungrue, I. N., Farber, C. R., Sinsheimer, J., Kang, H. M., Furlotte, N., and et al. Comparative analysis of proteome and transcriptome variation in mouse. *PLOS Genetics 7*, 6 (Jun 2011), e1001393.

[77] Gong, L. I., Suchard, M. A., and Bloom, J. D. Stability-mediated epistasis constrains the evolution of an influenza protein. *eLife 2* (May 2013), e00631.

[78] Gordon, N. C., Price, J. R., Cole, K., Everitt, R., Morgan, M., Finney, J., Kearns, A. M., Pichon, B., Young, B., Wilson, D. J., and et al. Prediction of staphylococcus aureus antimicrobial resistance by whole-genome sequencing. *Journal of Clinical Microbiology 52*, 4 (Apr 2014), 1182–1191.

[79] Gottesman, S. Trouble is coming: Signaling pathways that regulate general stress responses in bacteria. *Journal of Biological Chemistry 294* (Jun 2019), P11685–11700.

[80] Grünberger, F., Knüppel, R., Jüttner, M., Fenk, M., Borst, A., Reichelt, R., Hausner, W., Soppa, J., Ferreira-Cerca, S., and Grohmann, D. Exploring prokaryotic transcription, operon structures, rrna maturation and modifications using nanopore-based native rna sequencing. *bioRxiv* (May 2020), 2019.12.18.880849.

[81] Guha, S., Meyerson, A., Mishra, N., Motwani, R., and O'Callaghan, L. Clustering data streams: Theory and practice. *IEEE Transactions on Knowledge and Data Engineering 15*, 3 (Jun 2003), 515–528.

[82] Guthrie, V. B., Allen, J., Camps, M., and Karchin, R. Network models of tem β-lactamase mutations coevolving under antibiotic selection show modular structure and anticipate evolutionary trajectories. *PLOS Computational Biology 7*, 9 (Sep 2011), e1002184.

[83] Haas, B. J., Chin, M., Nusbaum, C., Birren, B. W., and Livny, J. How deep is deep enough for rna-seq profiling of bacterial transcriptomes? *BMC Genomics 13*, 1 (Dec 2012), 734.

[84] Hasman, H., Saputra, D., Sicheritz-Ponten, T., Lund, O., Svendsen, C. A., Frimodt-Moller, N., and Aarestrup, F. M. Rapid whole-genome sequencing for detection and characterization of microorganisms directly from clinical samples. *Journal of Clinical Microbiology 52*, 1 (Jan 2014), 139–146.

[85] Hastie, T., Tibshirani, R., and Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Second Edition*. Springer Science and Business Media, Aug 2009.

[86] Hennebique, A., Bidart, M., Jarraud, S., Beraud, L., Schwebel, C., Maurin, M., and Boisset, S. Digital pcr for detection and quantification of fluoroquinolone resistance in legionella pneumophila. *Antimicrobial Agents and Chemotherapy 61*, 9 (2017).

[87] Herricks, T., Donczew, M., Mast, F. D., Rustad, T., Morrison, R., Sterling, T. R., Sherman, D. R., and Aitchison, J. D. Odelam, rapid sequence-independent detection of drug resistance in isolates of mycobacterium tuberculosis. *eLife 9* (May 2020), e56613.

[88] Hoenen, T., Groseth, A., Rosenke, K., Fischer, R. J., Hoenen, A., Judson, S. D., Martellaro, C., Falzarano, D., Marzi, A., Squires, R. B., and et al. Nanopore sequencing as a rapidly deployable ebola outbreak tool. *Emerging Infectious Diseases 22*, 2 (Feb 2016), 331–334.

[89] Horinouchi, T., Suzuki, S., Kotani, H., Tanabe, K., Sakata, N., Shimizu, H., and Furusawa, C. Prediction of cross-resistance and collateral sensitivity by gene expression profiles and genomic mutations. *Scientific Reports 7*, 14009 (Oct 2017).

[90] Huang, L., Liao, L., and Wu, C. H. Protein-protein interaction network inference from multiple kernels with optimization based on random walk by linear programming. In *2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)* (Nov 2015), p. 201–207.

[91] Hunt, M., Bradley, P., Lapierre, S. G., Heys, S., Thomsit, M., Hall, M. B., Malone, K. M., Wintringer, P., Walker, T. M., Cirillo, D. M., and et al. Antibiotic resistance prediction for mycobacterium tuberculosis from genome sequence data with mykrobe. *Wellcome Open Research 4* (Dec 2019), 191.

[92] Hunt, M., Mather, A. E., Sánchez-Busó, L., Page, A. J., Parkhill, J., Keane, J. A., and Harris, S. R. Ariba: rapid antimicrobial resistance genotyping directly from sequencing reads. *Microbial Genomics 3*, 10 (Sep 2017).

[93] Hutter, B., Schaab, C., Albrecht, S., Borgmann, M., Brunner, N. A., Freiberg, C., Ziegelbauer, K., Rock, C. O., Ivanov, I., and Loferer, H. Prediction of mechanisms of action of antibacterial compounds by gene expression profiling. *Antimicrobial Agents and Chemotherapy 48*, 8 (Aug 2004), 2838–2844.

[94] Iram, S., Dolson, E., Chiel, J., Pelesko, J., Krishnan, N., Güngör, z., Kuznets-Speck, B., Deffner, S., Ilker, E., Scott, J. G., and et al. Controlling the speed and trajectory of evolution with counterdiabatic driving. *bioRxiv* (Dec 2019), 867143.

[95] Jain, M., Olsen, H. E., Paten, B., and Akeson, M. The oxford nanopore minion: delivery of nanopore sequencing to the genomics community. *Genome Biology 17*, 1 (Nov 2016), 239.

[96] Jenkins, S. G., and Schuetz, A. N. Current concepts in laboratory testing to guide antimicrobial therapy. *Mayo Clinic Proceedings 87*, 3 (Mar 2012), 290–308.

[97] Jensen, P. A., Zhu, Z., and van Opijnen, T. Antibiotics disrupt coordination between transcriptional and phenotypic stress responses in pathogenic bacteria. *Cell Reports 20*, 7 (Aug 2017), 1705–1716.

[98] Jeong, H., Oltvai, Z. N., and Barabási, A.-L. Prediction of protein essentiality based on genomic data. *Complexus 1*, 1 (2003), 19–28.

[99] Jordan, I. K., Rogozin, I. B., Wolf, Y. I., and Koonin, E. V. Essential genes are more evolutionarily conserved than are nonessential genes in bacteria. 8.

[100] Jr, J. H. W. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association 58*, 301 (Mar 1963), 236–244.

[101] Kaneko, K., Furusawa, C., and Yomo, T. Universal relationship in gene-expression changes for cells in steady-growth state. *Physical Review X 5*, 1 (Feb 2015), 011014.

[102] Kavvas, E. S., Catoiu, E., Mih, N., Yurkovich, J. T., Seif, Y., Dillon, N., Heckmann, D., Anand, A., Yang, L., Nizet, V., and et al. Machine learning and structural analysis of mycobacterium tuberculosis pan-genome identifies genetic signatures of antibiotic resistance. *Nature Communications 9*, 1 (Dec 2018).

[103] Khaledi, A., Schniederjans, M., Pohl, S., Rainer, R., Bodenhofer, U., Xia, B., Klawonn, F., Bruchmann, S., Preusse, M., Eckweiler, D., and et al. Transcriptome profiling of antimicrobial resistance in pseudomonas aeruginosa. *Antimicrobial Agents and Chemotherapy 60*, 8 (Aug 2016), 4722–4733.

[104] Khan, Z. A., Siddiqui, M. F., and Park, S. Current and emerging methods of antibiotic susceptibility testing. *Diagnostics 9*, 2 (May 2019).

[105] Khazaei, T., Barlow, J. T., Schoepp, N. G., and Ismagilov, R. F. Rna markers enable phenotypic test of antibiotic susceptibility in neisseria gonorrhoeae after 10 minutes of ciprofloxacin exposure. *Scientific Reports 8*, 11606 (Aug 2018).

[106] Kim, J., Greenberg, D. E., Pifer, R., Jiang, S., Xiao, G., Shelburne, S. A., Koh, A., Xie, Y., and Zhan, X. Vampr: Variant mapping and prediction of antibiotic resistance via explainable features and machine learning. *PLOS Computational Biology 16*, 1 (Jan 2020), e1007511.

[107] Korch, S. B., and Hill, T. M. Ectopic overexpression of wild-type and mutant hipa genes in escherichia coli: Effects on macromolecular synthesis and persister formation. *Journal of Bacteriology 188*, 11 (Jun 2006), 3826–3836.

[108] Krstajic, D., Buturovic, L. J., Leahy, D. E., and Thomas, S. Cross-validation pitfalls when selecting and assessing regression and classification models. *Journal of Cheminformatics 6*, 1 (Mar 2014), 10.

[109] Kumar, A., Roberts, D., Wood, K. E., Light, B., Parrillo, J. E., Sharma, S., Suppes, R., Feinstein, D., Zanotti, S., Taiberg, L., and et al. Duration of hypotension before initiation of effective antimicrobial therapy is the critical determinant of survival in human septic shock. *Critical Care Medicine 34*, 6 (Jun 2006), 1589–1596.

[110] Langmead, B., and Salzberg, S. L. Fast gapped-read alignment with bowtie 2. *Nature Methods 9*, 44 (Apr 2012), 357–359.

[111] Lanie, J. A., Ng, W.-L., Kazmierczak, K. M., Andrzejewski, T. M., Davidsen, T. M., Wayne, K. J., Tettelin, H., Glass, J. I., and Winkler, M. E. Genome sequence of avery's virulent serotype 2 strain d39 of streptococcus pneumoniae and comparison

with that of unencapsulated laboratory strain r6. *Journal of Bacteriology 189*, 1 (Jan 2007), 38–51.

[112] Lazo, A., and Rathie, P. On the entropy of continuous probability distributions (corresp.). *IEEE Transactions on Information Theory 24*, 1 (Jan 1978), 120–122.

[113] Lees, J. A., Harris, S. R., Tonkin-Hill, G., Gladstone, R. A., Lo, S. W., Weiser, J. N., Corander, J., Bentley, S. D., and Croucher, N. J. Fast and flexible bacterial genomic epidemiology with poppunk. *Genome Research 29*, 2 (Feb 2019), 304–316.

[114] Lenski, R. E. Experimental evolution and the dynamics of adaptation and genome evolution in microbial populations. *The ISME Journal 11*, 1010 (Oct 2017), 2181–2194.

[115] Levin-Reisman, I., Ronin, I., Gefen, O., Braniss, I., Shoresh, N., and Balaban, N. Q. Antibiotic tolerance facilitates the evolution of resistance. *Science 355*, 6327 (Feb 2017), 826–830.

[116] Lewis, K. Platforms for antibiotic discovery. *Nature Reviews Drug Discovery 12*, 5 (May 2013), 371–387.

[117] Li, W., and Godzik, A. Cd-hit: a fast program for clustering and comparing large sets of protein or nucleotide sequences. *Bioinformatics 22*, 13 (Jul 2006), 1658–1659.

[118] Liao, Y., Smyth, G. K., and Shi, W. featurecounts: an efficient general purpose program for assigning sequence reads to genomic features. *Bioinformatics 30*, 7 (Apr 2014), 923–930.

[119] Ling, L. L., Schneider, T., Peoples, A. J., Spoering, A. L., Engels, I., Conlon, B. P., Mueller, A., Schäberle, T. F., Hughes, D. E., Epstein, S., and et al. A new antibiotic kills pathogens without detectable resistance. *Nature 517*, 7535 (Jan 2015), 455–459.

[120] Little, J. W., and Mount, D. W. The sos regulatory system of escherichia coli. *Cell 29*, 1 (May 1982), 11–22.

[121] Liu, G., Kong, Y., Fan, Y., Geng, C., Peng, D., and Sun, M. Whole-genome sequencing of bacillus velezensis ls69, a strain with a broad inhibitory spectrum against pathogenic bacteria. *Journal of Biotechnology 249* (May 2017), 20–24.

[122] Liu, X., Gallay, C., Kjos, M., Domenech, A., Slager, J., van Kessel, S. P., Knoops, K., Sorg, R. A., Zhang, J.-R., and Veening, J.-W. High-throughput crispri phenotyping identifies new essential genes in streptococcus pneumoniae. *Molecular Systems Biology 13*, 5 (2017), 931.

[123] Lloyd, S. Least squares quantization in pcm. *IEEE Transactions on Information Theory 28*, 2 (Mar 1982), 129–137.

[124] Lopez, J. M., Dromerick, A., and Freese, E. Response of guanosine 5'-triphosphate concentration to nutritional changes and its significance for bacillus subtilis sporulation. *Journal of Bacteriology 146*, 2 (May 1981), 605–613.

[125] Love, M. I., Huber, W., and Anders, S. Moderated estimation of fold change and dispersion for rna-seq data with deseq2. *Genome Biology 15*, 12 (Dec 2014), 550.

[126] Lowe, R., Shirley, N., Bleackley, M., Dolan, S., and Shafee, T. Transcriptomics technologies. *PLOS Computational Biology 13*, 5 (May 2017), e1005457.

[127] Ma, S., Jaipalli, S., Larkins-Ford, J., Lohmiller, J., Aldridge, B. B., Sherman, D. R., and Chandrasekaran, S. Transcriptomic signatures predict regulators of drug synergy and clinical regimen efficacy against tuberculosis. *mBio 10*, 6 (Dec 2019).

[128] Ma, S., Morrison, R., Hobbs, S. J., Farrow-Johnson, J., Rustad, T. R., and Sherman, D. R. Network stress test reveals novel drug potentiators in mycobacterium tuberculosis. *bioRxiv* (Sep 2018), 429373.

[129] Mack, S. G., Turner, R. L., and Dwyer, D. J. Achieving a predictive understanding of antimicrobial stress physiology through systems biology. *Trends in Microbiology 0*, 0 (Mar 2018).

[130] Mahfouz, N., Ferreira, I., Beisken, S., von Haeseler, A., and Posch, A. E. Large-scale assessment of antimicrobial resistance marker databases for genetic phenotype prediction: a systematic review. *Journal of Antimicrobial Chemotherapy 75*, 11, 3099–3108.

[131] Mathy, C., Derbinsky, N., Bento, J., Rosenthal, J., and Yedidia, J. The boundary forest algorithm for online supervised and unsupervised learning. *arXiv:1505.02867 [cs, stat]* (May 2015). arXiv: 1505.02867.

[132] McArthur, A. G., Waglechner, N., Nizam, F., Yan, A., Azad, M. A., Baylay, A. J., Bhullar, K., Canova, M. J., De Pascale, G., Ejim, L., and et al. The comprehensive antibiotic resistance database. *Antimicrobial Agents and Chemotherapy 57*, 7 (Jul 2013), 3348–3357.

[133] McCoy, K. M., Antonio, M. L., and van Opijnen, T. Magenta: a galaxy implemented tool for complete tn-seq analysis and data visualization. *Bioinformatics 33*, 17 (Sep 2017), 2781–2783.

[134] McGee, L., McDougal, L., Zhou, J., Spratt, B. G., Tenover, F. C., George, R., Hakenbeck, R., Hryniewicz, W., Lefévre, J. C., Tomasz, A., and et al. Nomenclature of major antimicrobial-resistant clones of streptococcus pneumoniae defined by the pneumococcal molecular epidemiology network. *Journal of Clinical Microbiology 39*, 7 (Jul 2001), 2565–2571.

[135] Misra, N., Singh, H., and Demchuk, E. Estimation of the entropy of a multivariate normal distribution. *Journal of Multivariate Analysis 92*, 2 (Feb 2005), 324–342.

[136] Mittman, S. A., Huard, R. C., Della-Latta, P., and Whittier, S. Comparison of bd phoenix to vitek 2, microscan microstrep, and etest for antimicrobial susceptibility testing of streptococcus pneumoniae. *Journal of Clinical Microbiology 47*, 11 (Nov 2009), 3557–3561.

[137] Monti, S., Tamayo, P., Mesirov, J., and Golub, T. Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data. *Machine Learning 52*, 1 (Jul 2003), 91–118.

[138] Munkres, J. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics 5*, 1 (1957), 32–38.

[139] Neuwirth, E. *RColorBrewer: ColorBrewer Palettes*, 2014. R package version 1.1-2.

[140] Ng, A. Y., Jordan, M. I., and Weiss, Y. *On Spectral Clustering: Analysis and an algorithm*. MIT Press, 2002, p. 849–856.

[141] Nichol, D., Jeavons, P., Fletcher, A. G., Bonomo, R. A., Maini, P. K., Paul, J. L., Gatenby, R. A., Anderson, A. R. A., and Scott, J. G. Steering evolution with sequential therapy to prevent the emergence of bacterial antibiotic resistance. *PLOS Computational Biology 11*, 9 (Sep 2015), e1004493.

[142] Nichol, D., Rutter, J., Bryant, C., Hujer, A. M., Lek, S., Adams, M. D., Jeavons, P., Anderson, A. R. A., Bonomo, R. A., and Scott, J. G. Antibiotic collateral sensitivity is contingent on the repeatability of evolution. *Nature Communications 10*, 1 (Jan 2019), 1–10.

[143] Nowak, M. A. *Evolutionary Dynamics: Exploring the Equations of Life*. Harvard University Press, Sep 2006. Google-Books-ID: YXrIRDuAbE0C.

[144] O'Leary, N. A., Wright, M. W., Brister, J. R., Ciufo, S., Haddad, D., McVeigh, R., Rajput, B., Robbertse, B., Smith-White, B., Ako-Adjei, D., and et al. Reference sequence (refseq) database at ncbi: current status, taxonomic expansion, and functional annotation. *Nucleic Acids Research 44*, D1 (Jan 2016), D733–745.

[145] Oliver, H. F., Orsi, R. H., Ponnala, L., Keich, U., Wang, W., Sun, Q., Cartinhour, S. W., Filiatrault, M. J., Wiedmann, M., and Boor, K. J. Deep rna sequencing of l. monocytogenes reveals overlapping and extensive stationary phase and sigma b-dependent transcriptomes, including multiple highly transcribed noncoding rnas. *BMC Genomics 10* (Dec 2009), 641.

[146] Ozsolak, F., and Milos, P. M. Rna sequencing: advances, challenges and opportunities. *Nature reviews. Genetics 12*, 2 (Feb 2011), 87–98.

[147] Page, A. J., Cummins, C. A., Hunt, M., Wong, V. K., Reuter, S., Holden, M. T. G., Fookes, M., Falush, D., Keane, J. A., and Parkhill, J. Roary: rapid large-scale prokaryote pan genome analysis. *Bioinformatics 31*, 22 (Nov 2015), 3691–3693.

[148] Porto, W. F., Irazazabal, L., Alves, E. S. F., Ribeiro, S. M., Matos, C. O., Pires, l. S., Fensterseifer, I. C. M., Miranda, V. J., Haney, E. F., Humblot, V., and et al. In silico optimization of a guava antimicrobial peptide enables combinatorial exploration for peptide design. *Nature Communications 9*, 11 (Apr 2018), 1490.

[149] Qu, K., Guo, F., Liu, X., Lin, Y., and Zou, Q. Application of machine learning in microbiology. *Frontiers in Microbiology 10* (2019), 827.

[150] Quach, D., Sakoulas, G., Nizet, V., Pogliano, J., and Pogliano, K. Bacterial cytological profiling (bcp) as a rapid and accurate antimicrobial susceptibility testing method for staphylococcus aureus. *EBioMedicine 4* (Jan 2016), 95–103.

[151] Raven, K. E., Reuter, S., Gouliouris, T., Reynolds, R., Russell, J. E., Brown, N. M., Török, M. E., Parkhill, J., and Peacock, S. J. Genome-based characterization of hospital-adapted enterococcus faecalis lineages. *Nature microbiology 1*, 3 (Mar 2016).

[152] Rhoads, A., and Au, K. F. Pacbio sequencing and its applications. *Genomics, Proteomics and Bioinformatics 13*, 5 (Oct 2015), 278–289.

[153] Rohart, F., Gautier, B., Singh, A., and Cao, K.-A. L. mixomics: An r package for 'omics feature selection and multiple data integration. *PLOS Computational Biology 13*, 11 (Nov 2017), e1005752.

[154] Rustad, T. R., Minch, K. J., Ma, S., Winkler, J. K., Hobbs, S., Hickey, M., Brabant, W., Turkarslan, S., Price, N. D., Baliga, N. S., and et al. Mapping and manipulating the mycobacterium tuberculosis transcriptome using a transcription factor overexpression-derived regulatory network. *Genome Biology 15*, 11 (Nov 2014), 502.

[155] Saliba, A.-E., C Santos, S., and Vogel, J. New rna-seq approaches for the study of bacterial pathogens. *Current Opinion in Microbiology 35* (Feb 2017), 78–87.

[156] Salverda, M. L. M., Dellus, E., Gorter, F. A., Debets, A. J. M., Oost, J. v. d., Hoekstra, R. F., Tawfik, D. S., and Visser, J. A. G. M. d. Initial mutations direct alternative pathways of protein evolution. *PLOS Genetics 7*, 3 (Mar 2011), e1001321.

[157] Sandgren, A., Albiger, B., Orihuela, C. J., Tuomanen, E., Normark, S., and Henriques-Normark, B. Virulence in mice of pneumococcal clonal types with known invasive disease potential in humans. *The Journal of Infectious Diseases 192*, 5 (Sep 2005), 791–800.

[158] Schmid, M., Muri, J., Melidis, D., Varadarajan, A. R., Somerville, V., Wicki, A., Moser, A., Bourqui, M., Wenzel, C., Eugster-Meier, E., Frey, J. E., Irmler, S., and Ahrens, C. H. Comparative genomics of completely sequenced lactobacillus helveticus genomes provides insights into strain-specific genes and resolves metagenomics data down to the strain level. *Frontiers in Microbiology 9* (2018), 63.

[159] Scholz, M., Ward, D. V., Pasolli, E., Tolio, T., Zolfo, M., Asnicar, F., Truong, D. T., Tett, A., Morrow, A. L., and Segata, N. Strain-level microbial epidemiology and population genomics from shotgun metagenomics. *Nature Methods 13*, 5 (May 2016), 435–438.

[160] Schröder, H., Langer, T., Hartl, F. U., and Bukau, B. Dnak, dnaj and grpe form a cellular chaperone machinery capable of repairing heat-induced protein damage. *The EMBO Journal 12*, 11 (Nov 1993), 4137–4144.

[161] Schubert, O. T., Ludwig, C., Kogadeeva, M., Zimmermann, M., Rosenberger, G., Gengenbacher, M., Gillet, L. C., Collins, B. C., Röst, H. L., Kaufmann, S. H. E., and et al. Absolute proteome composition and dynamics during dormancy and resuscitation of mycobacterium tuberculosis. *Cell Host and Microbe 18*, 1 (Jul 2015), 96–108.

[162] Schwikowski, B., Uetz, P., and Fields, S. A network of protein–protein interactions in yeast. *Nature Biotechnology 18*, 12 (Dec 2000), 1257–1261.

[163] Seemann, T. Prokka: rapid prokaryotic genome annotation. *Bioinformatics 30*, 14 (Jul 2014), 2068–2069.

[164] Seemann, T. snippy: Rapid haploid variant calling and core genome alignment, 2020.

[165] Seif, Y., Kavvas, E., Lachance, J.-C., Yurkovich, J. T., Nuccio, S.-P., Fang, X., Catoiu, E., Raffatellu, M., Palsson, B. O., and Monk, J. M. Genome-scale metabolic reconstructions of multiple salmonella strains reveal serovar-specific metabolic traits. *Nature Communications 9*, 3771 (Sep 2018).

[166] Sherman, E. X., Wozniak, J. E., and Weiss, D. S. *Methods to Evaluate Colistin Heteroresistance in Acinetobacter baumannii*. Methods in Molecular Biology. Springer, 2019, p. 39–50.

[167] Shi, J., and Malik, J. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence 22*, 8 (Aug 2000), 888–905.

[168] Shi, Z.-Y., Enright, M. C., Wilkinson, P., Griffiths, D., and Spratt, B. G. Identification of three major clones of multiply antibiotic-resistant streptococcus pneumoniae in taiwanese hospitals by multilocus sequence typing. *Journal of Clinical Microbiology 36*, 12 (Dec 1998), 3514–3519.

[169] Shishkin, A. A., Giannoukos, G., Kucukural, A., Ciulla, D., Busby, M., Surka, C., Chen, J., Bhattacharyya, R. P., Rudy, R. F., Patel, M. M., and et al. Simultaneous generation of many rna-seq libraries in a single reaction. *Nature Methods 12*, 44 (Apr 2015), 323–325.

[170] Sidi, J., and Galili, T. *shinyHeatmaply: Deploy 'heatmaply' using 'shiny'*, 2020. R package version 0.2.0.

[171] Skwark, M. J., Croucher, N. J., Puranen, S., Chewapreecha, C., Pesonen, M., Xu, Y. Y., Turner, P., Harris, S. R., Beres, S. B., Musser, J. M., and et al. Interacting networks of resistance, virulence and core machinery genes identified by genome-wide epistasis analysis. *PLOS Genetics 13*, 2 (Feb 2017), e1006508.

[172] Smith, T., and Waterman, M. Identification of common molecular subsequences. *Journal of Molecular Biology 147*, 1 (Mar 1981), 195–197.

[173] Sommer, M. O. A., Munck, C., Toft-Kehler, R. V., and Andersson, D. I. Prediction of antibiotic resistance: time for a new preclinical paradigm? *Nature Reviews Microbiology 15*, 11 (Nov 2017), 689–696.

[174] Srivastava, S., and Gupta, M. R. Bayesian estimation of the entropy of the multivariate gaussian. In *2008 IEEE International Symposium on Information Theory* (Jul 2008), p. 1103–1107.

[175] Stamatakis, A. Raxml version 8: a tool for phylogenetic analysis and post-analysis of large phylogenies. *Bioinformatics 30*, 9 (May 2014), 1312–1313.

[176] Stevens, M. J. A., Tasara, T., Klumpp, J., Stephan, R., Ehling-Schulz, M., and Johler, S. Whole-genome-based phylogeny of bacillus cytotoxicus reveals different clades within the species and provides clues on ecology and evolution. *Scientific Reports 9*, 1984 (Feb 2019).

[177] Stokes, J. M., Lopatkin, A. J., Lobritz, M. A., and Collins, J. J. Bacterial metabolism and antibiotic efficacy. *Cell Metabolism 30*, 2 (Aug 2019), 251–259.

[178] Stokes, J. M., Yang, K., Swanson, K., Jin, W., Cubillos-Ruiz, A., Donghia, N. M., MacNair, C. R., French, S., Carfrae, L. A., Bloom-Ackermann, Z., and et al. A deep learning approach to antibiotic discovery. *Cell 180*, 4 (Feb 2020), 688–702.e13.

[179] Strehl, A., and Ghosh, J. Cluster ensembles – a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research 3*, 583–617.

[180] Su, M., Satola, S. W., and Read, T. D. Genome-based prediction of bacterial antibiotic resistance. *Journal of Clinical Microbiology 57*, 3 (Mar 2019).

[181] Surujon, D. Shinyomics, 2019.

[182] Surujon, D. Misc. data processing scripts, 2020.

[183] Surujon, D. Shinyomics: Exploration of 'omics' data, 2020.

[184] Surujon, D. Streptococcus pneumoniae antibiotic stress response, 2020.

[185] Surujon, D., and van Opijnen, T. Shinyomics: collaborative exploration of omics-data. *BMC bioinformatics 21*, 1 (Jan 2020), 22.

[186] Suzuki, S., Horinouchi, T., and Furusawa, C. Prediction of antibiotic resistance by gene expression profiles. *Nature Communications 5* (Dec 2014), 5792.

[187] Swarthout, T. D., Fronterre, C., Lourenço, J., Obolski, U., Gori, A., Bar-Zeev, N., Everett, D., Kamng'ona, A. W., Mwalukomo, T. S., Mataya, A. A., and et al. High residual carriage of vaccine-serotype streptococcus pneumoniae after introduction of pneumococcal conjugate vaccine in malawi. *Nature Communications 11*, 11 (May 2020), 2222.

[188] Team, R. *RStudio: Integrated Development Environment for R*. RStudio, Inc., 2015.

[189] Team, R. C. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, 2017.

[190] Thibault, D., Jensen, P. A., Wood, S., Qabar, C., Clark, S., Shainheit, M. G., Isberg, R. R., and van Opijnen, T. Droplet tn-seq combines microfluidics with tn-seq for identifying complex single-cell phenotypes. *Nature Communications 10*, 11 (Dec 2019), 5729.

[191] Thieffry, D., Huerta, A. M., Pérez-Rueda, E., and Collado-Vides, J. From specific gene regulation to genomic networks: a global analysis of transcriptional regulation in escherichia coli. *BioEssays 20*, 5 (1998), 433–440.

[192] Tonkin-Hill, G., MacAlasdair, N., Ruis, C., Weimann, A., Horesh, G., Lees, J. A., Gladstone, R. A., Lo, S., Beaudoin, C., Floto, R. A., and et al. Producing polished prokaryotic pangenomes with the panaroo pipeline. *Genome Biology 21*, 1 (Jul 2020), 180.

[193] Toprak, E., Veres, A., Michel, J.-B., Chait, R., Hartl, D. L., and Kishony, R. Evolutionary paths to antibiotic resistance under dynamically sustained drug selection. *Nature Genetics 44*, 1 (Jan 2012), 101–105.

[194] Tuite, A. R., Gift, T. L., Chesson, H. W., Hsu, K., Salomon, J. A., and Grad, Y. H. Impact of rapid susceptibility testing and antibiotic selection strategy on the emergence and spread of antibiotic resistance in gonorrhea. *The Journal of Infectious Diseases 216*, 9 (Nov 2017), 1141–1149.

[195] Tyers, M., and Wright, G. D. Drug combinations: a strategy to extend the life of antibiotics in the 21st century. *Nature Reviews Microbiology 17*, 33 (Mar 2019), 141–155.

[196] Vakil, V., and Trappe, W. Drug combinations: Mathematical modeling and networking methods. *Pharmaceutics 11*, 5 (May 2019), 208.

[197] Van Dongen, S. Graph clustering via a discrete uncoupling process. *SIAM J. Matrix Anal. Appl. 30* (Jan 2008), 121–141.

[198] van Opijnen, T., Bodi, K. L., and Camilli, A. Tn-seq: high-throughput parallel sequencing for fitness and genetic interaction studies in microorganisms. *Nature Methods 6*, 10 (Oct 2009), 767–772.

[199] van Opijnen, T., and Camilli, A. Genome-wide fitness and genetic interactions determined by tn-seq, a high-throughput massively parallel sequencing method for microorganisms. *Current Protocols in Microbiology 19*, 1 (2010), 1E.3.1–1E.3.16.

[200] van Opijnen, T., and Camilli, A. A fine scale phenotype–genotype virulence map of a bacterial pathogen. *Genome Research 22*, 12 (Dec 2012), 2541–2551.

[201] van Opijnen, T., and Camilli, A. Transposon insertion sequencing: a new tool for systems-level analysis of microorganisms. *Nature Reviews Microbiology 11*, 7 (Jul 2013), 435–442.

[202] van Opijnen, T., Dedrick, S., and Bento, J. Strain dependent genetic networks for antibiotic-sensitivity in a bacterial pathogen with a large pan-genome. *PLOS Pathogens 12*, 9 (Sep 2016), e1005869.

[203] van Tonder, A. J., Bray, J. E., Jolley, K. A., Jansen van Rensburg, M., Quirk, S. J., Haraldsson, G., Maiden, M. C. J., Bentley, S. D., Haraldsson, s., Erlendsdóttir, H., and et al. Genomic analyses of >3,100 nasopharyngeal pneumococci revealed significant differences between pneumococci recovered in four different geographical regions. *Frontiers in Microbiology 10* (2019), 317.

[204] Veer, L. J. v. t., Dai, H., Vijver, M. J. v. d., He, Y. D., Hart, A. A. M., Mao, M., Peterse, H. L., Kooy, K. v. d., Marton, M. J., Witteveen, A. T., and et al. Gene expression profiling predicts clinical outcome of breast cancer. *Nature 415*, 6871 (Jan 2002), 530.

[205] Ventola, C. L. The antibiotic resistance crisis. *Pharmacy and Therapeutics 40*, 4 (Apr 2015), 277–283.

[206] Wadsworth, C. B., Sater, M. R. A., Bhattacharyya, R. P., and Grad, Y. H. Impact of species diversity on the design of rna-based diagnostics for antibiotic resistance in neisseria gonorrhoeae. *Antimicrobial Agents and Chemotherapy 63*, 8 (Aug 2019).

[207] Wang, X., Zorraquino, V., Kim, M., Tsoukalas, A., and Tagkopoulos, I. Predicting the evolution of escherichia coli by a data-driven approach. *Nature Communications 9*, 1 (Sep 2018), 3562.

[208] Weinreich, D. M., Delaney, N. F., DePristo, M. A., and Hartl, D. L. Darwinian evolution can follow only very few mutational paths to fitter proteins. *Science 312*, 5770 (Apr 2006), 111–114.

[209] Weinstein, Z. B., Bender, A., and Cokol, M. Prediction of synergistic drug combinations. *Current Opinion in Systems Biology 4* (Aug 2017), 24–28.

[210] Weldhagen, G. F. Integrons and beta-lactamases–a novel perspective on resistance. *International Journal of Antimicrobial Agents 23*, 6 (Jun 2004), 556–562.

[211] Westermann, A. J., Förstner, K. U., Amman, F., Barquist, L., Chao, Y., Schulte, L. N., Müller, L., Reinhardt, R., Stadler, P. F., and Vogel, J. Dual rna-seq unveils noncoding rna functions in host–pathogen interactions. *Nature 529*, 7587 (Jan 2016), 496–501.

[212] Westermann, A. J., Gorski, S. A., and Vogel, J. Dual rna-seq of pathogen and host. *Nature Reviews Microbiology 10*, 9 (Sep 2012), 618–630.

[213] Yelin, I., Snitser, O., Novich, G., Katz, R., Tal, O., Parizade, M., Chodick, G., Koren, G., Shalev, V., and Kishony, R. Personal clinical history predicts antibiotic resistance of urinary tract infections. *Nature Medicine 25*, 7 (Jul 2019), 1143–1152.

[214] Yim, G., McClure, J., Surette, M. G., and Davies, J. E. Modulation of salmonella gene expression by subinhibitory concentrations of quinolones. *The Journal of Antibiotics 64*, 11 (Jan 2011), 73–78.

[215] Yimer, S. A., Birhanu, A. G., Kalayou, S., Riaz, T., Zegeye, E. D., Beyene, G. T., Holm-Hansen, C., Norheim, G., Abebe, M., Aseffa, A., and et al. Comparative proteomic analysis of mycobacterium tuberculosis lineage 7 and lineage 4 strains reveals differentially abundant proteins linked to slow growth and virulence. *Frontiers in microbiology 8* (May 2017), 795–795.

[216] Zeitler, K., and Narayanan, N. The present and future state of antimicrobial stewardship and rapid diagnostic testing: Can one ideally succeed without the other? *Current Treatment Options in Infectious Diseases 11*, 2 (Jun 2019), 177–187.

[217] Zhang, B., and Horvath, S. A general framework for weighted gene co-expression network analysis. *Statistical applications in genetics and molecular biology 4*, 17 (2005).

[218] Zhang, T., Ramakrishnan, R., and Livny, M. Birch: An efficient data clustering method for very large databases. *SIGMOD Rec. 25*, 2 (Jun 1996), 103–114.

[219] Zhao, Y., Wu, J., Yang, J., Sun, S., Xiao, J., and Yu, J. Pgap: pan-genomes analysis pipeline. *Bioinformatics 28*, 3 (Feb 2012), 416–418.

[220] Zhou, Y., Zhou, B., Pache, L., Chang, M., Khodabakhshi, A. H., Tanaseichuk, O., Benner, C., and Chanda, S. K. Metascape provides a biologist-oriented resource for the analysis of systems-level datasets. *Nature Communications 10*, 1 (Apr 2019), 1–10.

[221] Zhu, Z., Surujon, D., Ortiz-Marquez, J. C., Huo, W., Isberg, R. R., Bento, J., and van Opijnen, T. Entropy of a bacterial stress response is a generalizable predictor for fitness and antibiotic sensitivity. *Nature Communications 11*, 11 (Aug 2020), 4365.

[222] Zou, Y., Xue, W., Luo, G., Deng, Z., Qin, P., Guo, R., Sun, H., Xia, Y., Liang, S., Dai, Y., and et al. 1,520 reference genomes from cultivated human gut bacteria enable functional microbiome analyses. *Nature Biotechnology 37*, 2 (Feb 2019), 179–185.

This thesis was typeset using LaTeX, originally developed by Leslie Lamport and based on Donald Knuth's TeX. A template that can be used to format a PhD thesis with this look and feel has been released under the permissive mit (x11) license, and can be found online at github.com/suchow/Dissertate or from its author, Jordan Suchow, at suchow@post.harvard.edu.