Early antimicrobial resistance prediction

Using urinary proteomics and machine learning

Master's thesis in Engineering mathematics and computational science

EMELIE LEMANN

DEPARTMENT OF MATHEMATICAL SCIENCES

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2026
www.chalmers.se

Master's thesis 2026

Early antimicrobial resistance prediction

Using urinary proteomics and machine learning

EMELIE LEMANN

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Mathematical Sciences
Division of Applied Mathematics and Statistics
Chalmers University of Technology
Gothenburg, Sweden 2026

Early antimicrobial resistance prediction
Using urinary proteomics and machine learning
EMELIE LEMANN

© EMELIE LEMANN, 2026.

Supervisor: Annikka Polster, Department of Life Sciences
Examiner: Marija Cvijovic, Department of Mathematical Sciences

Master's Thesis 2026
Department of Mathematical Sciences
Division of Applied Mathematics and Statistics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2026

Early antimicrobial resistance prediction
Using urinary proteomics and machine learning
EMELIE LEMANN
Department of Mathematical Sciences
Chalmers University of Technology

Abstract

Antimicrobial resistance (AMR) is a growing global health challenge, requiring improved methods for rapid and accurate resistance prediction. This study investigates whether urinary proteomics data can be used to predict AMR phenotypes using machine-learning approaches.

Mass spectrometry-based proteomics data from urine samples associated with *E.coli* urinary tract infections were preprocessed, and analysed using multiple machine-learning frameworks. Both multi-label and single-label classification approaches were evaluated using stratified cross-validation, followed by feature selection and stability-based analysis to reduce dimensionality while improving interpretability. In addition, antibiotic-specific models were developed to account for variation in resistance prevalence and class imbalance across antibiotics.

The results show that proteomics-derived protein abundance profiles contain predictive information related to AMR phenotypes, although performance varied across antibiotics and modelling strategies. Single-label classification generally outperformed the multi-label framework in terms of stability and interpretability. Feature selection substantially reduced the dimensionality of the dataset while maintaining comparable predictive performance, suggesting that a limited subset of proteins captures a significant proportion of the predictive signal. However, class imbalance and small sample size limited the ability to reliably predict minority resistance phenotypes.

The findings demonstrate the potential of urinary proteomics combined with machine learning for AMR phenotype prediction, while highlighting key methodological challenges such as class imbalance, high dimensionality, and limited generalisability. These results support further development of proteomics-based predictive models and validation on larger, independent clinical cohorts.

Keywords: AMR, Proteomics, UTI, Machine learning, Classification, Feature selection, Targeted methods

Acknowledgements

I would like to express my gratitude to my supervisor, Annikka Polster, for her continuous guidance, support, and encouragement throughout this project. Her supervision has been essential, and her willingness to challenge me has contributed greatly to both the direction and the depth of my work. I am also grateful to my examiner, Marija Cvijovic, for her invaluable feedback and support along the way. Further, I would like to thank Roger Karlsson at Sahlgrenska University Hospital for providing the dataset and for taking the time to explain its structure and background. Special thanks go to PhD student Agata Marchi for the numerous discussions and insights throughout the project. Finally, I would like to thank the SysBio division at Chalmers for providing an inclusive and supportive working environment.

Emelie Lemann, Gothenburg, May 2026

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

AMR	Antimicrobial Resistance
AST	Antimicrobial Susceptibility Testing
CV	Cross-Validation
E.coli	Escherichia coli
LASSO	Least Absolute Shrinkage and Selection Operator
ML	Machine Learning
MS	Mass Spectrometry
MS/MS	Tandem Mass Spectrometry
RF	Random Forest
SVM	Support Vector Machine
TIMS	Trapped Ion Mobility Spectometry
TOF	Time-Of-Flight
UTI	Urinary Tract Infection

Contents

List of Figures

List of Tables

1

Introduction

Antimicrobial resistance (AMR) is one of the most pressing global challenges in modern healthcare. The rising prevalence of resistant bacterial infections complicates treatment decisions, increases hospitalization time, and contributes to higher healthcare costs and mortality [1]. Rapid and reliable identification of resistance profiles is therefore essential for guiding effective antibiotic therapy [2].

Current diagnostic workflows for urinary tract infections (UTIs) rely on culture-based antimicrobial susceptibility testing, which is time-consuming and delays the availability of resistance profiles. In routine clinical microbiology, urine samples constitute a major proportion of laboratory workload, although a large fraction of these samples do not yield clinically relevant culture results. These limitations have motivated interest in computational approaches that can support earlier sample stratification and predict antimicrobial resistance phenotypes from molecular data [3].

Mass spectrometry-based proteomics have enabled large-scale measurement of protein abundance directly from clinical samples. Since proteins represent functional downstream biological activity, proteomic profiles may capture molecular signatures associated with AMR mechanisms. In parallel, machine learning (ML) has become an important tool in biomedical research due to its ability to model complex patterns in high-dimensional data. The combination of proteomics and ML therefore offers a promising approach for improving resistance prediction and supporting future clinical decision-making [4].

1.1 Aim

The aim of this study was to investigate whether urinary proteomics data can be used to predict antimicrobial resistance phenotypes using ML. This will be addressed by transforming high-dimensional proteomics into predictive models through systematic preprocessing, comparison of different modelling strategies, and evaluation of feature selection approaches, with the goal of developing a final predictive pipeline.

1.2 Limitations

This study has several limitations, both in dataset and the analytical framework. The dataset consisted of a relatively small patient cohort, which limits generalisability and increases sensitivity to sampling variability. In addition, strong class imbalance was present for several antibiotics, making minority resistance phenotypes difficult to predict reliably. The study also focused exclusively on *E. coli*-associated UTIs, meaning that the findings may not directly generalise to other bacterial species, infection types, or clinical populations.

Proteomics data are characterized by high dimensionality, where thousands of protein features are measured from a relatively small number of samples, increasing the risk of overfitting. In addition, missing values are common due to stochastic sampling and low-abundance proteins falling below detection limits, which requires careful preprocessing [5].

Taken together, these factors limit generalisability and highlight the challenges of working with high-dimensional clinical proteomics data. The study therefore primarily serves as a proof-of-concept for applying ML to urinary proteomics for AMR.

2

Theory

This chapter presents the theoretical background of the study, covering both the clinical context and the ML principles relevant to the analysis. It introduces the biomedical setting of AMR and proteomics, as well as the core concepts required for modelling and evaluating high-dimensional biological data. Together, these foundations support the development of predictive models based on proteomics data.

2.1 Clinical and Biological Background

This section introduces the clinical and biological concepts relevant to this study, including AMR, UTIs, and proteomics-based analysis. Together, these concepts provide the biological context for the ML methods applied throughout the study.

2.1.1 Antimicrobial Resistance

AMR refers to the ability of microorganisms, such as bacteria, to survive or proliferate despite exposure to antimicrobial agents that would normally inhibit growth or eliminate the organism [6]. AMR has emerged as a major global public health challenge due to increasing resistance against commonly used antibiotics or other pathogen drugs, leading to prolonged infections, increased mortality, and reduced treatment options [1].

Bacterial resistance can arise through several biological mechanisms. Common mechanisms include modification of antibiotic target sites, reduced membrane permeability, active drug efflux, and enzymatic inactivation of antimicrobial compounds. Resistance traits may develop through spontaneous genetic mutations or through gene transfer between bacteria [7].

In clinical microbiology, antimicrobial susceptibility testing (AST) is routinely performed to determine whether bacterial isolates are susceptible or resistant to specific antibiotics. Susceptibility outcomes are commonly categorized into three classes: susceptible (S), intermediate (I), resistant (R). Susceptible isolates are expected to respond to standard antibiotic treatment, whereas resistant isolates are unlikely to respond sufficiently. Intermediate classifications represent isolates with reduced susceptibility or uncertain treatment outcomes [8].

Rapid identification of AMR phenotypes is clinically important for selecting effective antibiotic treatments. Delayed or inappropriate therapy may increase the risk of complications, contribute to the spread of resistant bacteria, and promote further resistance development. Conventional culture-based susceptibility testing methods are often time-consuming, motivating the development of computational and molecular approaches capable of predicting resistance phenotypes more rapidly [2].

2.1.2 Urinary Tract Infections

UTIs are among the most common bacterial infections worldwide and represent a substantial clinical burden. UTIs may affect different parts of the urinary tract, including the bladder and kidneys, and are commonly diagnosed using urine samples [9]. The most common causative pathogen of UTIs is *Escherichia coli* (*E.coli*), and is also the most associated with resistance [10].

Urine-based diagnostics are clinically advantageous because they provide relatively non-invasive sampling while containing information related to both bacterial infection and host response. Rapid identification of resistant pathogens in urine samples is particularly important for guiding antibiotic treatment decisions and reducing unnecessary use of broad-spectrum antibiotics [11].

Because treatment selection depends strongly on bacterial susceptibility profiles, improved prediction of AMR phenotypes could contribute to faster and more targeted therapeutic interventions.

2.1.3 Proteomics

Proteomics is the large-scale study of proteins expressed in a biological system, often referred to as the proteome. In clinical and biomedical research, proteomics is commonly used to measure the characteristics of the protome such as peptide expression, interactions and modifications of proteins [12].

A common method for analyzing complex protein sample is mass spectromy (MS). Mass spectrometry (MS) works by first ionizing molecules and converting them into the gas phase. The ions are then separated in the mass analyzer according to their mass-to-charge ratio (m/z), producing a mass spectrum with peaks representing different molecules [13]. In proteomics, proteins are usually broken into peptides before analysis ("bottom-up" MS). These peptides can then be fragmented further in tandem mass spectrometry (MS/MS), where the fragment patterns help identify the peptide sequences and, ultimately, the original proteins [4].

The resulting MS/MS spectra are matched against theoretical peptide sequences derived from protein databases in a process known as peptide-spectrum matching. This enables identification of peptides, which are subsequently mapped back to their originating proteins, allowing inference of protein-level abundances [13].

In this study, proteomic measurements were generated using a timsTOF-based mass spectrometry platform. The timsTOF system combines trapped ion mobility spec-

trometry (TIMS) with time-of-flight (TOF) analysis, that allows more rapid and efficient analysis [14].

2.2 Machine Learning

This section introduces the ML concepts and statistical learning principles relevant to this study. Particular focus is placed on supervised classification, preprocessing, different classifications strategies and algorithms, and model evaluation.

2.2.1 Preprocessing

Preprocessing is applied to raw data before analysis or classification in order to improve data quality and reduce noise. Real-world data often contain inconsistencies, noise, or variables with different scales, making preprocessing an important step before training ML models [15].

Log transformation

Log transformation is commonly used to reduce skewness and stabilize variability in data, especially when outliers are present. By compressing extreme observations, the transformation can reduce the influence of outliers and improve the validity of statistical analyses [16].

Normalization

Median normalization centers the data around the median value of each sample:

$$\tilde{y}_{ij} = y_{ij} - \text{median}(y_i) \tag{2.1}$$

where y_{ij} is the value of variable j in sample i, and $\tilde{y}_{ij}$ is the normalized value. Median normalization is preferred over mean normalization when outliers are present, since the median is more robust to extreme values [17].

Scaling

Z-score normalization, or scaling, standardizes data by subtracting the mean and dividing by the standard deviation:

$$Z = \frac{X - \mu(X)}{\sigma(X)} \tag{2.2}$$

where $\mu(X)$ is the mean and $\sigma(X)$ is the standard deviation of the variable X [15].

This transformation produces data with mean 0 and standard deviation 1, making variables with different scales easier to compare and analyze.

2.2.2 Supervised Learning

Supervised learning refers to a model that is trained on samples with annotated class labels, it simply allows systems to learn from the information provided by the data in order to make predictions [4]. Supervised learning can either be a classification or a regression problem, where classification is categorical and regression is numerical [18]. The fundamentals of supervised ML can be described as using a dataset:

$$\mathcal{D} := \{(\mathrm{y}_i, \mathbf{x}_i), i = 1, \ldots, N\}, \tag{2.3}$$

where $\mathbf{x} = (x_1, \ldots, x_p)$ are features and y is the target.

The goal of supervised learning is to find a function f that maps features to the target:

$$(x_1, \ldots, x_p) \xrightarrow{f} y \tag{2.4}$$

A central challenge in applying supervised ML techniques to proteomics is to construct models that performs well and without overfitting, due to the high dimensionality [5]. Proteomics datasets are typically characterized by high dimensionality, meaning that the number of measured variables substantially exceeds the number of available samples.

2.2.3 Classification Strategies

Single-Label Classification

Single-label classification is the most common classification problem [19]. Classification is a method where a model, based on its supervised learning on known classes, predicts the class of new samples without annotations [4]. In classification problems, the response variable y is categorical.

The aim is to construct a classifier $C(\mathbf{x})$ that assigns an observation with feature vector $\mathbf{x}$ to one of J possible classes, where $J = 2$ in binary classification problems [18]. Unlike regression problems, where prediction accuracy is commonly evaluated using the mean squared error (MSE), classification problems require a different objective since the outputs are discrete class labels rather than numerical values. Instead, the quality of a classifier is measured using the mean classification error (MCE), defined as:

$$\mathrm{MCE}(C) = \mathbb{E}\left[I(y \neq C(\mathbf{x}))\right], \tag{2.5}$$

where $I(\cdot)$ is the indicator function, taking the value 1 when the statement is true and 0 otherwise [18].

A common probabilistic approach to classification is based on the conditional probability:

$$p_j(\mathbf{x}) = \mathbb{P}(y = j \mid \mathbf{x}), \tag{2.6}$$

which represents the probability that an observation belongs to class j given the feature vector $\mathbf{x}$ [18].

The optimal classification rule, known as the Bayes classifier, assigns an observation to the class with the highest conditional probability:

$$C(\mathbf{x}) = \arg\max_j \ p_j(\mathbf{x}). \tag{2.7}$$

The Bayes classifier minimizes the mean classification error and therefore provides the theoretically optimal classification rule [18].

In the binary classification setting, where $y \in \{0, 1\}$, the conditional probability can be written as:

$$p(\mathbf{x}) = \mathbb{P}(y = 1 \mid \mathbf{x}), \tag{2.8}$$

while

$$1 - p(\mathbf{x}) = \mathbb{P}(y = 0 \mid \mathbf{x}). \tag{2.9}$$

An observation is classified into class 1 if:

$$p(\mathbf{x}) > 0.5, \tag{2.10}$$

and into class 0 otherwise [18].

Multi-Label Classification

Classification problems are commonly formulated as single-label classification problems. However, some classification tasks are more ambiguous and require other approaches and multi-label classification was developed to address such problems. In multi-label learning, each observation can be associated with multiple labels at the same time [20].

A common approach to multi-label classification is to decompose the problem into multiple binary classification tasks, where one classifier is trained independently for each label [19]. The mapping is similar to Eq. (2.4), and can in this case be described using m target functions:

$$f(\mathbf{x}) = \Big(f_1(\mathbf{x}), f_2(\mathbf{x}), \ldots, f_m(\mathbf{x})\Big), \tag{2.11}$$

where each f_j is learned by fitting a separate classifier:

$$f_j : \mathbf{x} \to y_j, \quad j = 1, \ldots, m. \tag{2.12}$$

2.2.4 Classification Algorithms

Random Forest

Random Forest (RF) is an ensemble classifier that can handle large number of features and is efficient on high-dimensional datasets [4]. The RF classifier consists

of multiple decision trees trained independently. For each tree, a random subset of K features is selected and used for training, and this process is repeated to build a forest of trees. Each tree votes for a class, and the final prediction is given by majority vote. The Figure 2.1 illustrates a RF classifier.

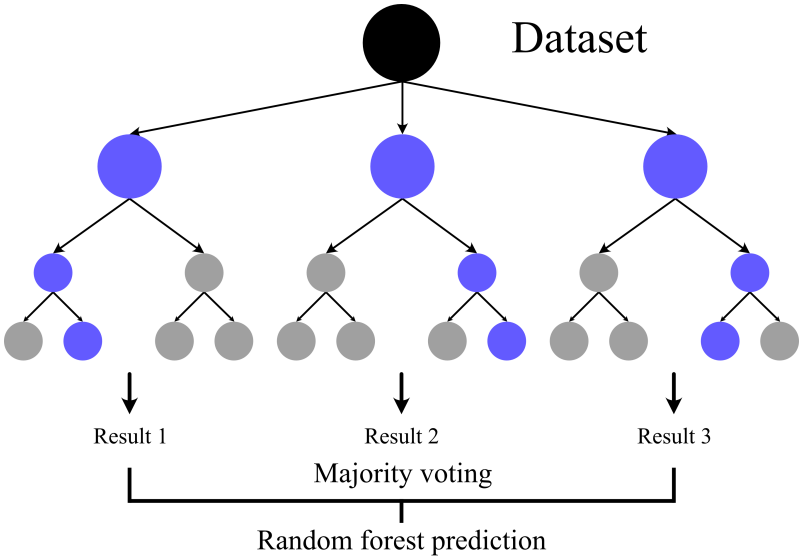

Figure 2.1: A RF classifier with multiple decision trees. The figure is adapted from Widodo et al. (2022) [22].

Logistic Regression

A common way to model the unknown conditional probabilities in classification is to use parametric approaches. Among these, logistic regression is one of the most widely used methods. Logistic regression models the conditional probability of belonging to a class by transforming a linear predictor through a nonlinear sigmoid function. For the binary case, where $y \in \{0, 1\}$, the probability is defined as in Eq. (2.8). This probability is modeled as:

$$p(x) = \frac{e^{x^\top \beta}}{1 + e^{x^\top \beta}}, \tag{2.13}$$

where $x^\top \beta = \beta_0 + x_1\beta_1 + \cdots + x_p\beta_p$ is a linear combination of the input features [18].

The logistic function ensures that predicted probabilities are always restricted to the interval $[0, 1]$, unlike simpler linear probability models which may produce invalid probability values outside this range. For this reason, logistic regression is preferred for classification [18].

A classification decision is then made using a probability threshold, the same as in Eq. (2.10):

$$\hat{y} = \begin{cases} 1 & \text{if } p(x) > 0.5, \\ 0 & \text{otherwise.} \end{cases} \tag{2.14}$$

Support Vector Machines

Support Vector Machine (SVM) is a supervised learning method for classification that constructs a decision boundary, called a hyperplane, which separates data points of different classes [23].

Consider a dataset similar to Eq. (2.3) consisting of feature vectors $\mathbf{x}_i \in \mathbb{R}^p$ and corresponding class labels $y_i \in \{-1, +1\}$. In the linear case, SVM assumes that the classes can be separated by a linear decision boundary in the feature space. SVM assumes that the classes can be separated by a linear decision boundary in the input space. In the linear case, the decision function is defined as a linear model:

$$f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x} + b, \tag{2.15}$$

The points for which $f(\mathbf{x}) > 0$ are assigned to class $+1$, while points for which $f(\mathbf{x}) < 0$ are assigned to class -1. So basically this classification rule assigns a class label based on the sign of the decision function:

$$y = \text{sign}(f(\mathbf{x})). \tag{2.16}$$

The hyperplane divides the feature space into two half-spaces corresponding to the two classes, and is defined by:

$$\mathbf{w}^\top \mathbf{x} + b = 0. \tag{2.17}$$

If the data is linearly separable, the SVM finds the hyperplane that maximizes the margin between the two classes. This is formulated as the optimization problem:

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 \quad \text{subject to} \quad y_i(\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \quad \forall i. \tag{2.18}$$

The constraint ensures that all training samples are correctly classified and lie outside the margin boundaries [23].

2.2.5 Hyperparameters

Hyperparameters are settings in a ML model that are chosen before training, and is therefore not learned from the data. They control how the algorithm behaves, and the model's performance can depend heavily on them. [24] Examples of hyperparameters n_estimators, min_samples_leaf and C, where n_estimators describes number of trees, min_samples_leaf the number of samples required in a leaf node and C is regularization strength [25][26].

2.2.6 Feature Selection

Proteomics datasets typically contain thousands of protein features while only including a limited number of patient samples. This high feature-to-sample ratio

increases the risk of overfitting, where ML models learn noise or sample-specific patterns rather than biologically meaningful relationships [27].

Feature selection aims to reduce dimensionality by identifying informative variables while removing redundant or irrelevant attributes [4]. Reducing the number of features may improve computational efficiency, data quality, and data understanding [28]. In biomedical applications, feature selection is particularly important because selected proteins could be potential biomarkers [4].

Filter Methods

Filter methods select features before training a model by using general properties of the data, independent of any learning algorithm. They are called filters because they remove irrelevant features and keep only the most promising ones. These methods are fast and general because they do not depend on how a specific model works. [29]

A common filter approach is based on statistical hypothesis testing. The idea is to test whether a feature is useful for distinguishing classes. We define the hypotheses as H_0, the feature is not important, and H_a, the feature is important for distinguishing classes. In a t-test, which is used for binary classification, the means of a feature in two classes are computed. A significant p-value is computed, which measures the probability of observing such a difference under the assumption that H_0 is true. [29]

Embedded Methods

Embedded methods combine ideas from both filter and wrapper methods. [30] Like filter methods, they evaluate the importance of features based on some criterion, but like wrapper methods, they generate feature subsets which are used to train a learning algorithm, and the best feature subset is selected. LASSO and elastic net are two embedded methods, which are described based on Fonti and Belitser (2017) [30].

LASSO

One widely used embedded method is LASSO (Least Absolute Shrinkage and Selection Operator), a regularization method used in linear regression that performs both parameter estimation and feature selection. It works by minimizing the squared error while adding an L1 regularization penalty on the magnitude of the coefficients. L1 regularization refers to a penalty based on the absolute values of the model coefficients, which encourages sparsity by shrinking some coefficients exactly to zero.

The LASSO optimization problem is:

$$\hat{\beta}(\lambda) = \arg\min_{\beta} \left(\frac{1}{n} \|Y - X\beta\|_2^2 + \lambda \|\beta\|_1 \right) \tag{2.19}$$

where:

$$\|Y - X\beta\|_2^2 = \sum_{i=1}^{n}(Y_i - (X\beta)_i)^2, \quad \|\beta\|_1 = \sum_{j=1}^{k}|\beta_j| \tag{2.20}$$

and $\lambda \geq 0$ controls the strength of the regularization term.

A larger λ increases shrinkage of coefficients toward zero, while $\lambda = 0$ reduces the method to ordinary least squares regression.

The key property of LASSO is that the L1 penalty can force some coefficients to become exactly zero:

$$\hat{\beta}_j(\lambda) = 0 \tag{2.21}$$

This means that the corresponding features are removed from the model, making LASSO a powerful embedded feature selection method.

Elastic Net

Elastic Net is an extension of LASSO that combines L_1 and L_2 (Ridge) regularization:

$$\hat{\beta}_{EN} = \arg\min_{\beta}\left(\frac{1}{n}\|Y - X\beta\|_2^2 + \lambda_1\|\beta\|_1 + \lambda_2\|\beta\|_2^2\right) \tag{2.22}$$

2.2.7 Model Evaluation

Hold-Out test

The hold-out test is a simple method for evaluating a predictive model by splitting the dataset into two separate sets, a training set and a testing set. The model is trained on the training set and then evaluated once on the testing set, which is treated as unseen data. A common split is, for example, 80% of the data for training and 20% for testing. [31]

Stratified K-fold Cross-Validation

Cross-validation (CV) is a method used to evaluate how well a predictive model generalises to unseen data by repeatedly splitting the dataset into training and testing parts. [32] In k-fold CV, the dataset is divided into k equal, or nearly equal, parts called folds. In each iteration, one fold is used as the test set, while the remaining $k-1$ folds are used for training. This process is repeated k times so that each fold is used exactly once as the test set. The final performance is obtained by averaging the results over all folds [31]. This approach provides a more reliable estimate of model performance, reduces dependence on a single train-test split, and helps reduce overfitting by evaluating the model on multiple different subsets of the data [32].

Stratified k-fold CV is a variant of k-fold CV in which the folds are constructed so that each fold preserves the same class proportions as the full dataset. In other words, each fold maintains the original distribution of the classes by using stratified sampling instead of random splitting. This is especially useful for classification problems with imbalanced classes, since it ensures that every fold is representative of the overall data distribution [22].

Performance Metrics

In a binary classification task, instances are typically predicted as either positive or negative. A positive label often indicates the presence of an illness or resistance , while a negative label indicates no deviation from the baseline. Each prediction can therefore be classified into one of four categories: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). A TP is a correctly predicted positive case, a TN is a correctly predicted negative case, a FP is a negative instance incorrectly predicted as positive, and a FN is a positive instance incorrectly predicted as negative [33].

From these values, several standard evaluation metrics are derived. Accuracy measures the proportion of correctly classified instances among all predictions:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.23}$$

It provides an overall measure of performance but can be misleading in imbalanced datasets [33].

Precision and recall focus on the positive class. Recall is the fraction of correctly identified positive cases:

$$\text{Recall} = \frac{TP}{TP + FN} \tag{2.24}$$

Precision is the fraction of correctly predicted positive cases among all predicted positives:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{2.25}$$

The F1-score combines precision and recall using their harmonic mean:

$$F1 = \frac{2 \cdot \text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{2.26}$$

For multiclass problems, different averaging strategies are used. The macro F1-score computes the F1-score independently for each class and then averages them equally:

$$F1_{macro} = \frac{1}{N} \sum_{i=1}^{N} F1_i \tag{2.27}$$

where

$$F1_i = \frac{2 \cdot \text{Precision}_i \cdot \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i} \tag{2.28}$$

The weighted F1-score accounts for class imbalance by weighting each class by its support:

$$F1_{weighted} = \sum_{i=1}^{N} w_i F1_i \tag{2.29}$$

where

$$w_i = \frac{TP_i + FN_i}{\sum_{j=1}^{N}(TP_j + FN_j)} \tag{2.30}$$

Macro F1 treats all classes equally, while weighted F1 gives more importance to frequent classes, making it more suitable when class distributions are imbalanced [34].

Confusion Matrix

A confusion matrix is a table used to evaluate the performance of a classification model by comparing true labels with predicted labels. It summarizes correct and incorrect classifications and serves as the basis for many evaluation metrics such as accuracy, sensitivity, and false-positive rate [32].

The matrix contains information about actual and predicted classes, where predicted classes are typically shown along the horizontal axis and actual classes along the vertical axis. Each entry of the matrix indicates how many instances of a given actual class are classified into a given predicted class [22]. In this way, it directly shows the number of correct and incorrect predictions made by the model, which can also be expressed as a percentage.

2.3 Reproducibility and Reliability

Reproducibility is an important consideration in biomedical ML research, particularly in studies involving complex datasets, such as high-dimensional data, multimodal data integration, and datasets with limited sample sizes. In addition, certain

ML algorithms contain inherent stochasticity during model training, while the conditions under which stochastic methods should be applied are not yet fully established. The design of such studies is therefore essential for ensuring reliable and reproducible results [27].

Data Leakage

Data leakage occurs when information from validation or test datasets unintentionally influences model training. Leakage can lead to artificially inflated performance estimates and poor generalization to unseen data [35].

Pre-processing leakage is a common source of leakage in ML. For example, during feature selection, if features are selected using the full dataset prior to train–test splitting, information from the test data becomes indirectly incorporated into the model. To avoid leakage, all test data should remain completely unseen until the final evaluation stage [35].

Leakage prevention is especially important in proteomics datasets because the number of measured features substantially exceeds the number of samples. Without careful validation procedures, ML models may appear highly predictive while failing to generalize to independent datasets [36].

3

Methods

This chapter describes the methodological framework used to investigate whether urinary proteomics data can be used to predict AMR phenotypes using ML. The workflow consists of four main components: data collection and preprocessing, development of classification frameworks, model evaluation and validation, and feature selection with stability analysis. An initial exploratory phase was used to compare different modelling strategies, followed by the construction of a final predictive pipeline based on stable feature selection and antibiotic-specific optimisation. Together, these steps define a structured approach for transforming high-dimensional proteomic measurements into predictive models of AMR.

3.1 Data and Preprocessing

The dataset used in this study was obtained from Sahlgrenska University Hospital and consisted of mass spectrometry–based urinary proteomics data. The cohort initially comprised 104 patients, from which protein abundance profiles were generated using timsTOF-based proteomic workflows applied to clinical urine samples. Protein identifications were obtained using pathogen-specific databases, enabling the detection of both host-derived and pathogen-associated proteins. Corresponding AMR phenotypes were available for 93 of the initially included patients, providing matched proteomic and clinical outcome data for supervised learning.

Several exclusion criteria were applied to ensure a clinically homogeneous study population. Patients who had undergone organ transplantation and were receiving immunosuppressive treatment were excluded due to their potential influence on urinary proteomic profiles. In addition, samples in which *Klebsiella pneumoniae* was identified as the dominant pathogen were removed to maintain focus on *E.coli*–associated infections. Following these exclusions, the final dataset consisted of 85 patients with complete proteomic and AMR profiles.

To formulate the task as a binary classification problem, intermediate susceptibility classes (I) were merged with resistant classes (R). This grouping was performed to reduce class sparsity and align with clinically conservative interpretation of reduced susceptibility. Susceptibility labels represented resistant (R) and susceptible (S) outcomes. Five antibiotics (Trimetoprim, Ciprofloxacin, Ceftazidim, Tobramycin,

Piperacillin/Tazobactam) were selected for analysis based on class representation, requiring the minority class to contain at least eight samples.

3.1.1 Preprocessing

Raw peptide intensity data were imported and processed using Python. Peptide intensity values were converted to numerical format, and invalid entries were mapped to missing values (NaN). Peptides containing exclusively zero or missing intensity values across all samples were removed, as they were considered non-informative for quantitative analysis.

To account for peptides mapping to multiple proteins, intensity values were first adjusted by dividing each peptide signal by the number of associated proteins, reducing bias introduced by shared peptide assignments. Peptide-level intensities were then aggregated to the protein level by computing the median across all peptides mapped to the same protein group.

Protein intensities were then log-transformed to stabilize variance and improve comparability across samples. To reduce systematic technical variation, median-centering normalization was applied across samples while preserving relative biological differences. The normalization was performed according to Eq. (2.1). Missing values were imputed using a small constant value (10^{-6}) to approximate signals below detection limits while minimizing distortion of relative abundance patterns. Where appropriate, additional feature scaling, performed according to Eq. (2.2), was applied for specific modelling and visualization purposes.. The final preprocessing pipeline resulted in a dataset comprising 15,336 protein features across all samples. This confirms the high-dimensional nature of the problem, where the number of predictors substantially exceeds the number of observations.

3.2 Overall Modelling Strategy

An iterative modelling framework was developed to evaluate the predictive performance of urinary proteomics for AMR classification. Given the high dimensionality of the data, limited sample size, and heterogeneous resistance patterns across antibiotics, a stepwise approach was used to progressively refine model complexity and feature representation.

The analysis began with a multi-label classification approach to assess whether resistance phenotypes could be jointly predicted within a single model. This was compared to independent single-label models, where each antibiotic was treated as a separate binary classification task. Subsequently, feature selection methods were introduced to reduce dimensionality and reduce the risk of overfitting, followed by stability-based selection of consistently informative protein features across CV folds. Finally, based on the observed variability in predictive performance across antibiotics, antibiotic-specific modelling considerations were introduced. This included adapting decision thresholds to better reflect differences in class distribution and biological differences between antibiotics.

3.3 Machine Learning Framework

This study evaluated two complementary classification strategies for predicting AMR from urinary proteomics data: multi-label and single-label classification. All models were trained and evaluated using identical feature matrices to ensure comparability.

3.3.1 Multi-Label Classification

A multi-label classification approach was first implemented to jointly predict resistance phenotypes across the five selected antibiotics. A multi-output RF model was used, where each output node corresponded to a specific antibiotic. The response matrix was structured such that each column represented a binary resistance label (susceptible or resistant), and each row corresponded to an individual patient sample.

3.3.2 Single-Label Classification

In the single-label setting, separate binary classifiers were trained independently for each antibiotic. This formulation treats each resistance phenotype as an individual prediction task, without assuming dependencies between antibiotics. The approach was included to evaluate whether antibiotic-specific models provide improved performance and stability compared to joint multi-label prediction.

3.3.3 Classification Algorithms

Three supervised learning algorithms were evaluated and compared: RF, Logistic Regression, and SVMs. All models were trained within the single-label classification framework and evaluated using identical CV and performance metrics to ensure comparability. Default hyperparameters from the Scikit-learn implementation were used for the comparison. Later in the analysis, the hyperparameters shown in Table 2.1 were used for the RF classifier.

Table 3.1: Hyperparameter tuning for RF classifier. The descriptions are based on the Scikit-learn documentation [25]

Hyperparameter	Value	Description
n_estimators	1000	Number of trees in the forest
min_samples_leaf	5	Minimum samples required in a leaf node
max_features	"sqrt"	Features considered when splitting a node
max_depth	5	Maximum depth of each tree
class_weight	"balanced"	Adjust class weights for imbalance

3.4 Validation Strategy and Model Evaluation

Model evaluation was conducted in two distinct stages: an initial exploratory evaluation phase and a final model validation phase. In the initial phase, all classification strategies, including multi-label and single-label models as well as feature selection approaches, were evaluated exclusively using stratified 5-fold CV. This phase was used for comparative model assessment and selection of the most suitable modelling framework. No independent test set was used at this stage.

Following this exploratory analysis, a final modelling pipeline was constructed. For this final model, the dataset was split into a training set (70%) and an independent hold-out test set (30%). Within the training set, stratified 5-fold CV was applied to perform feature selection and identify stable protein features. Feature selection was therefore fully nested within the training data to avoid information leakage from the test set.

After identifying stable features, a final RF classifier was trained on the full training set using only these selected features. The performance of the final model was then evaluated once on the independent hold-out test set, providing an unbiased estimate of generalisation performance.

3.4.1 Performance Metrics

Model performance was evaluated using accuracy, macro-averaged F1-score, and weighted F1-score. Accuracy provides a global measure of classification performance but is sensitive to class imbalance, which is present across several antibiotic resistance phenotypes in this dataset. To address this limitation, macro F1-score was used to assess performance equally across classes, independent of their frequency, while weighted F1-score accounts for class imbalance by weighting each class according to its support. Together, these metrics provide a balanced evaluation of predictive performance under imbalanced class distributions.

3.5 Feature Selection

Feature selection was performed to reduce the high dimensionality of the proteomics dataset and to improve model generalisation and interpretability. Given the large number of protein features relative to the number of samples, multiple complementary feature selection strategies were evaluated.

3.5.1 Feature Selection Approaches

Three feature selection strategies were considered. First, a missingness-based filtering step was applied to remove proteins with a high proportion of missing or unreliable intensity values. This step reduced noise and ensured that analyses were performed on consistently observed features.

Second, univariate statistical filtering was performed using t-tests to identify proteins significantly associated with AMR phenotypes. Features were ranked according to their statistical association with the outcome, and subsets of the most informative proteins were retained for model training.

Third, embedded feature selection methods were applied using regularised linear models, specifically LASSO (L1 regularisation) and Elastic Net (combined L1 and L2 regularisation). These methods perform feature selection during model fitting by shrinking less informative coefficients towards zero, enabling simultaneous classification and dimensionality reduction.

3.5.2 Stable Feature Selection

To improve robustness and reduce variability in selected features, feature selection was embedded within a stratified 5-fold CV framework, enabling assessment of feature stability across different training folds. Proteins that were consistently selected across all CV splits were defined as stable features, reflecting reproducible predictive signals rather than fold-specific artefacts.

Due to the resulting number of stable features, the top 30 proteins per antibiotic were selected for further analysis based on their selection frequency and consistency. Finally, a RF classifier was trained using only the stable feature set derived exclusively from the training data. The final modelling workflow is summarised in Figure 3.1.

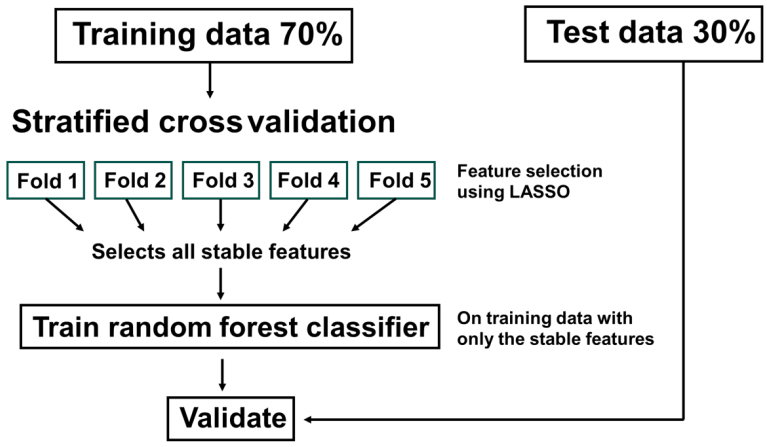

Figure 3.1: ML pipeline for the final model

3.6 Antibiotic-Specific Model Optimisation

Due to class imbalance and variation in resistance prevalence across antibiotics, a common decision threshold was found to be suboptimal for certain outcomes. To address this, antibiotic-specific decision thresholds were defined prior to final model training. The decision thresholds are defined according to Eq. (2.10). For

Ceftazidim, Ciprofloxacin, and Trimethoprim, the threshold was set to 0.45, whereas for Piperacillin/Tazobactam and Tobramycin, the threshold was set to 0.60.

These thresholds were incorporated into the modelling pipeline to account for differences in class distribution and improve sensitivity to minority classes. The optimisation was performed exclusively within the training data and applied after feature selection and before training the RF Classifier.

3.7 Reproducibility and Reliability

All analyses were implemented in Python using the Scikit-learn, Pandas, and NumPy libraries. Random seeds were fixed for all stochastic components, including data splitting, CV, and model training, to ensure reproducibility of results. The full ML pipeline was designed to be deterministic given identical inputs, and all preprocessing, feature selection, and model training steps were executed in a controlled and consistent computational environment.

4

Results

This chapter presents the results of applying ML to urinary proteomics data for AMR prediction. Multiple modelling strategies were evaluated, including multi-label and single-label classification, followed by feature selection and stability-based feature reduction. Antibiotic-specific analyses were also performed to assess variation in predictive performance across antibiotics.

4.1 Dataset Characteristics

This section summarizes the characteristics of the processed proteomics dataset prior to ML analysis. The AMR phenotypes showed varying degrees of class imbalance across antibiotics. Several antibiotics contained substantially fewer resistant samples compared to susceptible samples, creating challenges for model training and performance evaluation. Tobramycin and Piperacillin/Tazobactom both have the resistant as the dominant class. The distribution of resistant (R) and susceptible (S) samples for each antibiotic is shown in Figure 4.1.

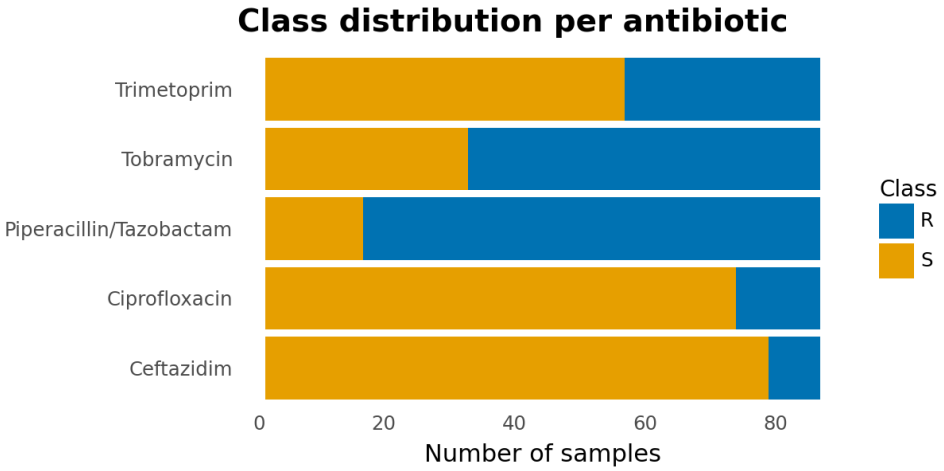

Figure 4.1: Class distribution showing the number of resistant (R) and susceptible (S) samples per antibiotic.

To evaluate the effect of preprocessing and normalization, protein intensity distributions before and after normalization were visualized in Figure 4.2. In addition, heatmap visualizations were generated to assess overall similarity patterns between samples and proteins following preprocessing, which is shown in Figure 4.3.

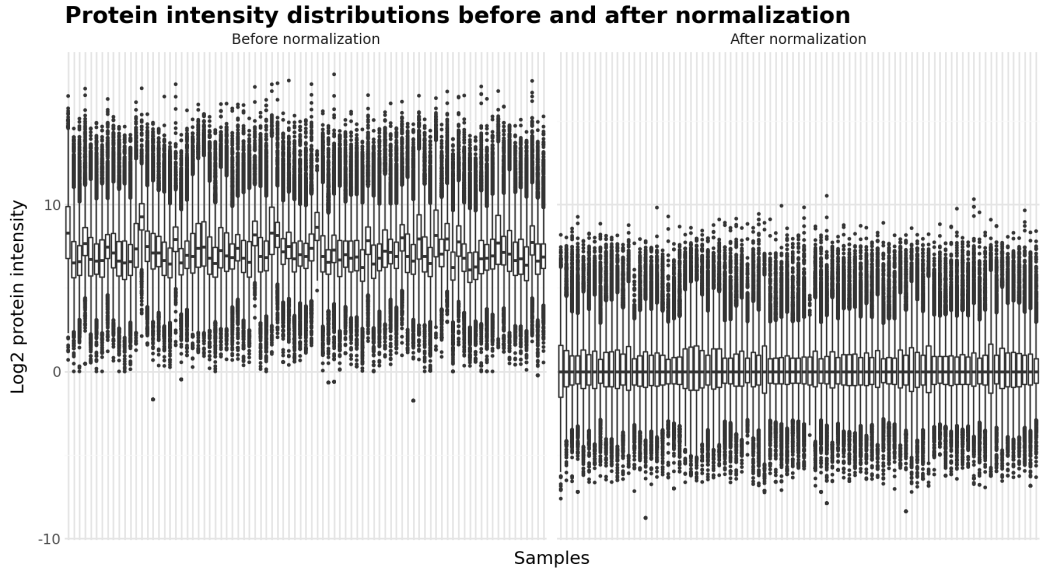

Figure 4.2: Distribution of log-transformed protein intensities across samples before and after median-centering normalization.

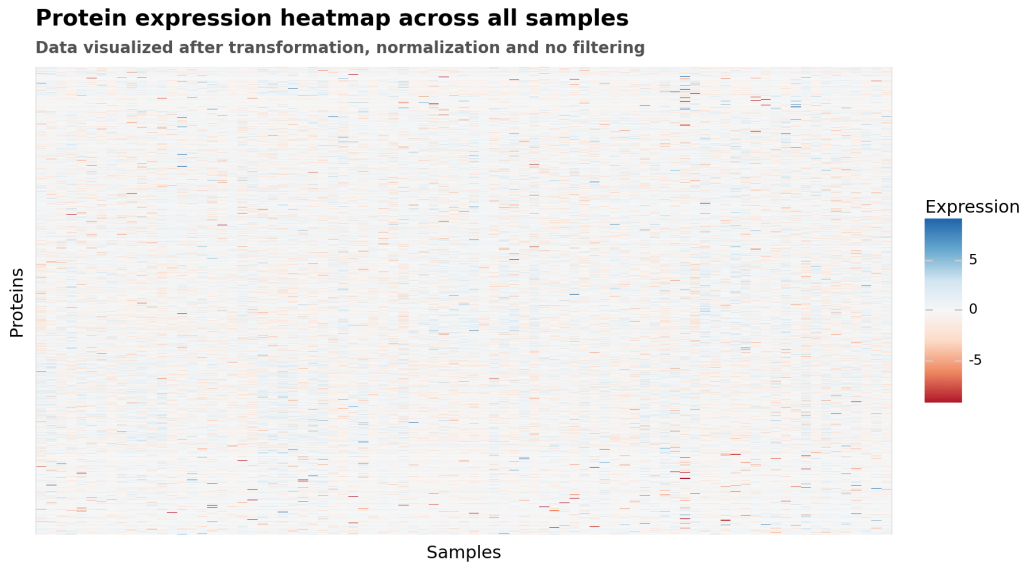

Figure 4.3: Heatmap showing normalized protein expression patterns across all samples included in the study. Protein abundance values were log-transformed and median-centered prior to visualization

4.2 Classification Framework

This section evaluates the predictive performance of different classification frameworks for predicting AMR phenotypes from urinary proteomics data.

4.2.1 Multi-label classification

A multi-label classification approach was initially applied to simultaneously predict AMR phenotypes across the five selected antibiotics using a multi-output RF model. Preliminary comparisons between RF, Logistic Regression, and SVM classifiers showed that RF models provided the most stable and consistently highest predictive performance across antibiotics. Consequently, subsequent analyses focused on the RF framework, while comparative results for alternative classifiers are provided in Appendix A.

In the multi-label model, each antibiotic was represented as a separate output variable, while a shared feature matrix was used across all targets. Model performance was evaluated using stratified 5-fold CV. The model showed limited predictive performance across antibiotics. CV results were more stable than bootstrap resampling, which exhibited higher variability across iterations. Due to this increased stability, stratified CV was used as the primary evaluation strategy. The mean performance across all antibiotics under CV was:

$$\text{Accuracy} = 0.271 \pm 0.061$$

$$F1_{\text{macro}} = 0.368 \pm 0.031$$

$$F1_{\text{weighted}} = 0.627 \pm 0.046$$

Prediction behaviour indicated a strong influence of class distribution. In particular, minority classes were rarely or never predicted, as reflected in the confusion matrices in Figure 4.4, which show a pronounced bias toward the majority class.

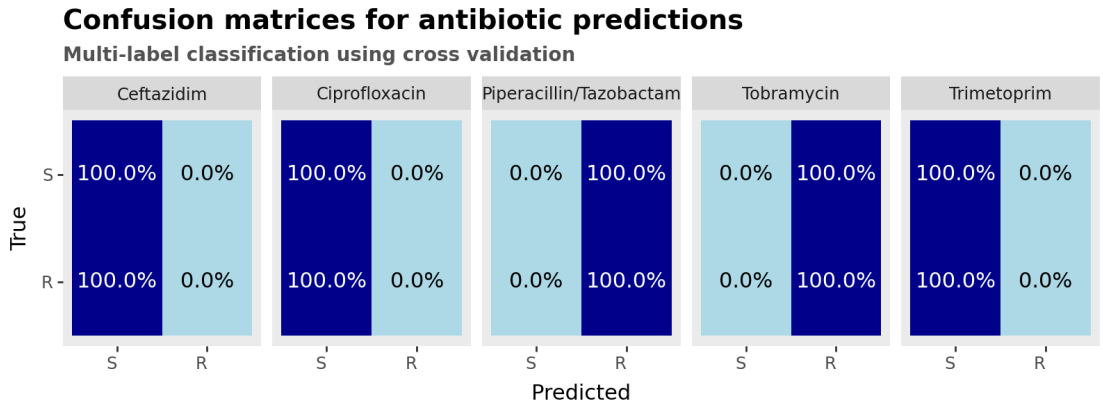

Figure 4.4: Confusion matrices for the multi-label classification model, showing a strong majority-class bias.

These results indicated that the multi-label formulation was not optimal for this dataset and motivated the subsequent evaluation of single-label classification models.

4.2.2 Single-Label Classification

In the single-label classification framework, separate binary classifiers were trained independently for each antibiotic using the same underlying feature representation as in the multi-label setting.

Overall, performance was comparable between the multi-label and single-label approaches, with slight improvements for certain antibiotics, as summarised in Table 4.1. Differences were most evident in minority class detection, as reflected in the confusion matrices in Figure 4.5. The largely similar performance metrics between the multi-label and single-label frameworks suggest that shared feature representations and class imbalance had a stronger influence on predictive behaviour than modelling formulation alone. Based on these results, the single-label framework was selected for subsequent feature selection and model refinement.

Table 4.1: Comparison of multi-label and single-label classification performance.

Antibiotic	Multi-label	Single-label
	Mean Accuracy	
Ceftazidim	0.905882	0.905882
Ciprofloxacin	0.847059	0.847059
Piperacillin/Tazobactam	0.823529	0.823529
Tobramycin	0.611765	0.611765
Trimetoprim	0.635294	0.588235
	Mean Macro F1-score	
Ceftazidim	0.475189	0.475189
Ciprofloxacin	0.458468	0.458468
Piperacillin/Tazobactam	0.451613	0.451613
Tobramycin	0.379365	0.379019
Trimetoprim	0.470550	0.453257
	Mean Weighted F1-score	
Ceftazidim	0.861386	0.861386
Ciprofloxacin	0.777182	0.777182
Piperacillin/Tazobactam	0.743833	0.743833
Tobramycin	0.482228	0.482351
Trimetoprim	0.556045	0.525293

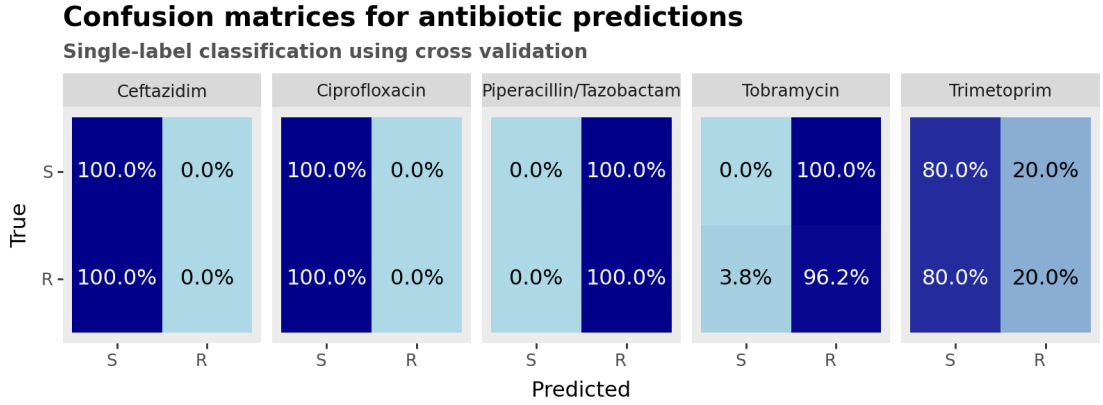

Figure 4.5: Confusion matrices for the single-label classification model across antibiotics.

4.3 Feature Selection Analysis

Feature selection was performed to reduce the high dimensionality of the proteomics dataset and to improve model generalisation and interpretability. Multiple feature selection strategies were evaluated within the single-label classification framework, where LASSO-based feature selection was selected for further analysis based on comparative results presented in Appendix B. For the LASSO-based models, classification performance and corresponding confusion matrices are shown in Table 4.2 and Figure 4.6.

While LASSO demonstrated the most consistent performance across antibiotics, improvements compared to alternative approaches were overall modest, with broadly similar classification behaviour observed across methods. However, a large number of proteins were still retained per antibiotic, motivating a subsequent stability-based feature selection step to further reduce dimensionality and identify reproducible protein signals across CV folds.

Table 4.2: Classification performance using LASSO-based feature selection, showing mean accuracy, macro F1-score, and weighted F1-score for each antibiotic.

Antibiotic	Accuracy	Macro-F1	Weighted-F1
Ceftazidim	0.898485	0.473104	0.850762
Ciprofloxacin	0.848485	0.458874	0.779221
Piperacillin/Tazobactam	0.813636	0.448442	0.744675
Tobramycin	0.578788	0.499563	0.547391
Trimetoprim	0.527273	0.380145	0.457485

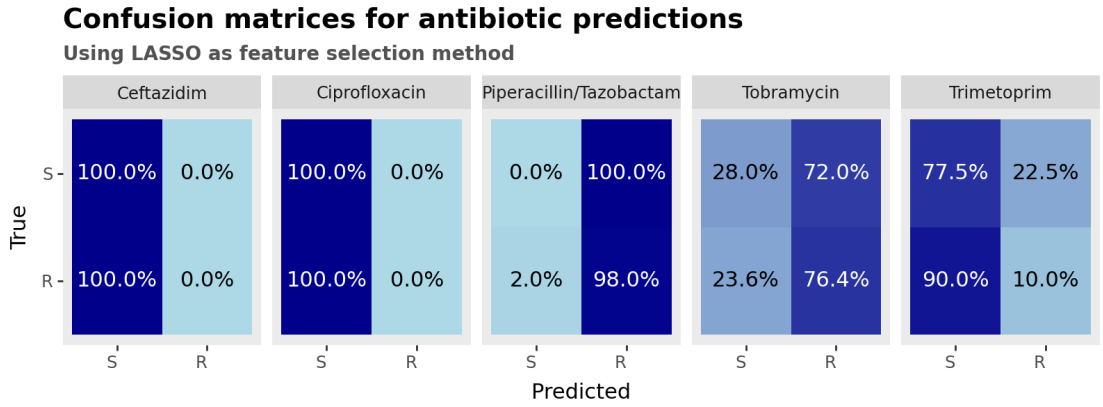

Figure 4.6: Confusion matrices for the single-label classification model using LASSO-selected features.

Feature selection was further evaluated within a stratified 5-fold CV framework. Proteins consistently selected across folds were defined as stable features. Figure 4.7 illustrates the overlap of LASSO-selected proteins across antibiotics, where diagonal elements represent within-antibiotic consistency and off-diagonal elements indicate shared feature selection between antibiotic pairs.

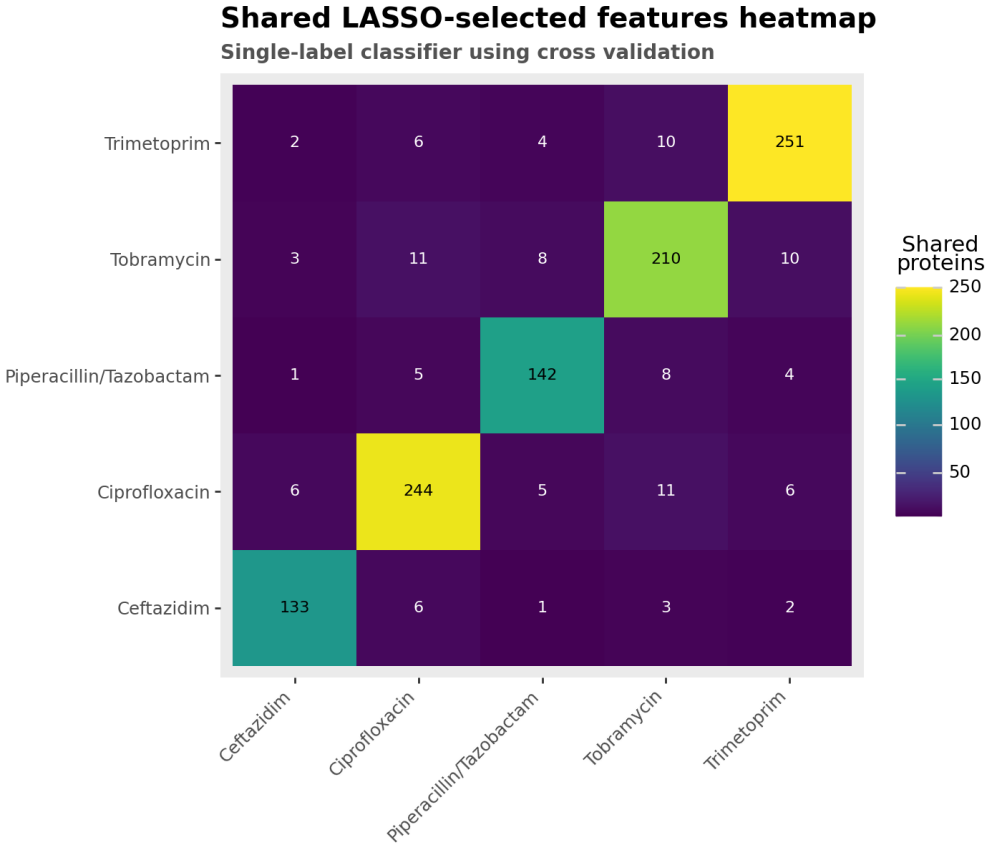

Figure 4.7: Heatmap of shared LASSO-selected features across antibiotics, showing within-antibiotic consistency (diagonal) and pairwise overlap in selected proteins.

To improve interpretability and further reduce dimensionality, the most consistently selected proteins per antibiotic were retained for downstream analysis. The resulting stable feature sets are summarised in Appendix C. Using this reduced feature space maintained comparable predictive performance while improving interpretability. Classification performance using the stable feature set is shown in Table 4.3, with corresponding confusion matrices in Figure 4.8.

Table 4.3: Model performance using LASSO feature selection and training RF classifier on the top 30 stable features

Antibiotic	Accuracy	Macro-F1	Weighted-F1
Ceftazidim	0.923077	0.480000	0.886154
Ciprofloxacin	0.884615	0.668085	0.853682
Piperacillin/Tazobactam	0.769231	0.434783	0.702341
Tobramycin	0.384615	0.277778	0.363248
Trimetoprim	0.769231	0.745098	0.769231

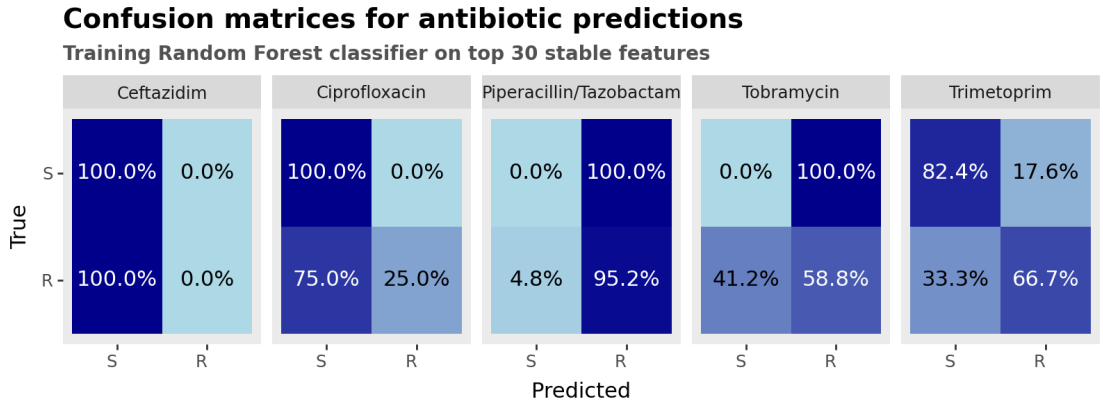

Figure 4.8: Classification performance using the top 30 stable features selected via LASSO, showing mean accuracy, macro F1-score, and weighted F1-score for each antibiotic.

The performance remained comparable to the full LASSO-based feature set, with slight improvements observed for certain antibiotics, while others showed stable or reduced performance. Notably, for Ciprofloxacin and Piperacillin/Tazobactam, the confusion matrices indicate improved detection of the minority class compared to earlier models, where predictions were more strongly biased toward the majority class. This suggests improved class-wise sensitivity, although limitations due to class imbalance remain.

An additional analysis using the top 10 most consistently selected features per antibiotic was performed to investigate whether further dimensionality reduction could improve performance. As shown in Table 4.4 and Figure 4.9, reducing the feature set further did not consistently improve performance, except for Tobramycin, which showed a modest improvement in certain metrics.

Table 4.4: Model performance using LASSO feature selection and training RF classifier on the top 10 stable features

Antibiotic	Accuracy	Macro-F1	Weighted-F1
Ceftazidim	0.846154	0.458333	0.846154
Ciprofloxacin	0.884615	0.668085	0.853682
Piperacillin/Tazobactam	0.615385	0.380952	0.615385
Tobramycin	0.384615	0.369697	0.399534
Trimetoprim	0.500000	0.333333	0.435897

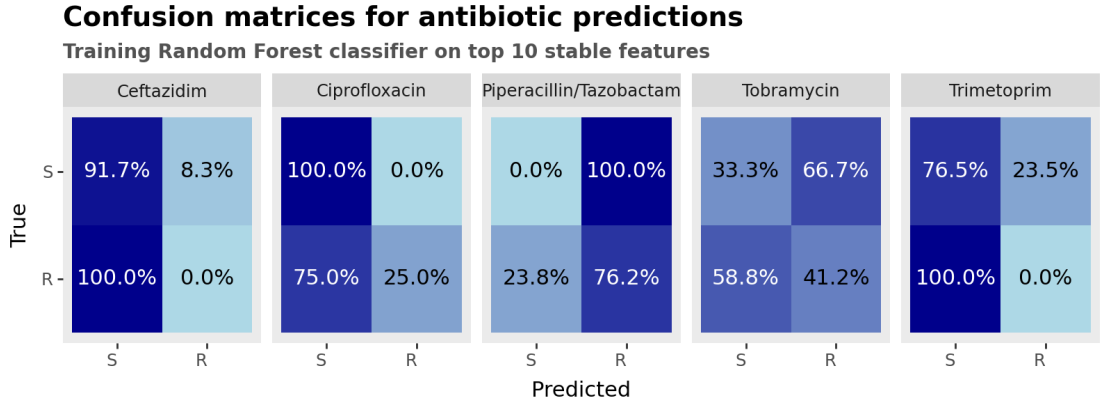

Figure 4.9: Classification performance using the top 10 stable features selected via LASSO, showing mean accuracy, macro F1-score, and weighted F1-score for each antibiotic.

4.4 Antibiotic-Specific Optimisation

To further investigate resistance prediction, classification performance was evaluated separately for each antibiotic using the model trained on the 30 stable features. Performance varied substantially across antibiotics, reflecting differences in resistance prevalence, class imbalance, and underlying biological variability. The resulting performance is summarised in Table 4.5, with corresponding confusion matrices shown in Figure 4.10.

Table 4.5: Model performance using antibiotic-specific model.

Antibiotic	Accuracy	Macro-F1	Weighted-F1
Ceftazidim	0.480000	0.923077	0.886154
Ciprofloxacin	0.623188	0.846154	0.823857
Piperacillin/Tazobactam	0.556818	0.769231	0.745629
Tobramycin	0.499259	0.500000	0.493333
Trimetoprim	0.640553	0.653846	0.661822

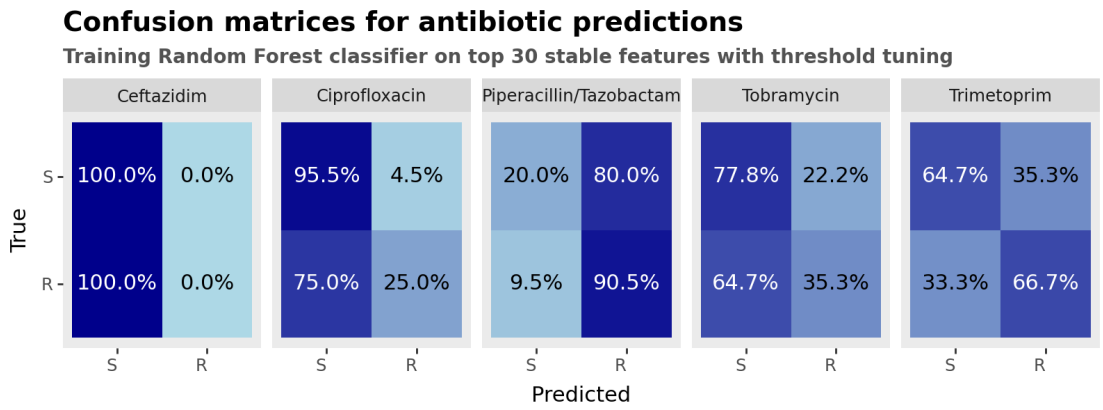

Figure 4.10: Confusion matrices summarizing prediction outcomes when using antibiotic-specific model.

Ciprofloxacin and Trimethoprim showed decreased performance under this setting, whereas Tobramycin and Piperacillin/Tazobactam exhibited improvements compared to earlier models. Overall, the results indicate that urinary proteomics contains predictive information relevant to AMR phenotypes.

5

Discussion

The results demonstrate that proteomics-derived protein abundance profiles contain predictive information associated with AMR phenotypes. However, several characteristics of the dataset influenced both model performance and interpretability. A major challenge was the high dimensionality of the proteomics data, where the number of protein features substantially exceeded the number of patient samples. In addition, pronounced class imbalance across several antibiotics limited the ability of the models to reliably predict minority resistance phenotypes.

A multi-label classification framework was initially evaluated under the assumption that resistance phenotypes may share biological mechanisms and therefore could be jointly predicted within a single model. While this approach enabled simultaneous prediction across antibiotics, predictive performance remained limited and the models demonstrated a strong bias toward the majority class. In particular, minority resistance classes were rarely, or never, predicted, despite relatively high accuracy values for certain antibiotics. This discrepancy highlights the limitations of accuracy as a performance metric under imbalanced class distributions and suggests that weighted F1-score provides a more informative measure of class-wise predictive performance in this study.

Compared to the multi-label framework, single-label classification models generally showed more stable and interpretable predictive behaviour. The improved performance observed for certain antibiotics suggests that AMR phenotype prediction benefits from antibiotic-specific modelling strategies, likely due to differences in resistance prevalence, class distribution, and underlying biological variability between antibiotics.

5.1 Impact of Targeted Methods

Several targeted methodological strategies were explored throughout the study to improve classification robustness, interpretability, and generalisation performance.

5.1.1 Feature Selection

Feature selection analyses demonstrated that dimensionality reduction could improve interpretability while maintaining similar predictive performance. The im-

proved performance observed for certain feature selection strategies may indicate that removal of noisy or non-informative proteins reduced overfitting and improved model generalisation. Because proteomics datasets contain large numbers of correlated and potentially redundant protein measurements, reducing the feature space may simplify the learning problem and improve robustness of the classifier.

The number of selected features also appeared important. Extremely large feature sets may increase the risk of fitting technical noise and dataset-specific variation, whereas overly restrictive filtering may remove biologically relevant proteins and reduce predictive information. Consequently, balancing dimensionality reduction against information preservation represents an important consideration in proteomics-based ML workflows.

Although the identified protein signatures demonstrated predictive utility within the current dataset, the biological relevance of these proteins remains uncertain. Further biological validation would therefore be necessary to determine whether the identified proteins are mechanistically associated with resistance phenotypes or instead reflect indirect host or pathogen-related responses.

5.1.2 Antibiotic-Specific Modeling

A unified modelling strategy initially appeared attractive because resistance phenotypes may share biological mechanisms and a unified model give better reproducibility. However, the results suggested that antibiotic-specific behaviour limited the effectiveness of more generalized prediction approaches.

These differences likely reflect both methodological and biological factors. Variability in resistance prevalence and class imbalance directly influenced the ability of the models to identify minority resistance classes. At the same time, the results may also indicate that resistance-associated proteomic signatures differ between antibiotics, potentially reflecting differences in bacterial resistance mechanisms, host responses, or proteomic detectability.

Antibiotic-specific modelling additionally improved interpretability by enabling direct investigation of proteins associated with each resistance phenotype separately. The improved performance observed for Trimetoprim following antibiotic-specific optimisation further supports the importance of adapting modelling strategies to antibiotic-dependent dataset characteristics rather than relying on a single generalised framework.

5.2 Limitations and Sources of Error

Several limitations should be considered when interpreting the results of this study. One major limitation was the relatively small cohort size. ML models trained on small datasets are more sensitive to random variation and may exhibit limited generalisability to independent patient populations. Despite the use of CV and an

independent hold-out test set, the risk of overfitting cannot be fully excluded due to the high-dimensional nature of the dataset.

Class imbalance represented another important challenge. Several resistance phenotypes contained relatively few resistant samples, making robust minority-class classification difficult and increasing uncertainty in performance estimates. This imbalance also contributed to prediction behaviour biased toward the majority class.

Missing values are common in mass spectrometry-based proteomics data due to incomplete peptide detection and stochastic sampling effects. Although preprocessing and imputation strategies were applied to minimise the impact of missingness, missing-value handling may still influence downstream analyses and model behaviour. Additionally, technical variation originating from sample preparation, instrument performance, and potential batch effects may introduce unwanted variability into the dataset.

Another limitation is that the identified protein signatures were not experimentally validated biologically. Consequently, the observed associations should primarily be interpreted as predictive patterns rather than confirmed biological resistance mechanisms. Further experimental studies would therefore be necessary to validate the biological relevance of the identified proteins.

Finally, because the study focused on a relatively homogeneous cohort of *E. coli*-associated infections with varying bacterial loads, the findings may have limited generalisability. In addition, all samples originate from a single clinical setting, which further restricts external validity. As a result, the findings may not directly generalise to other bacterial species, infection types, or patient populations without additional validation. External validation using independent cohorts is therefore necessary to assess the robustness and potential clinical applicability of the proposed modelling framework.

6

Conclusion

This study investigated whether proteomics-derived protein abundance profiles can be used to predict AMR phenotypes using ML approaches. Urinary proteomics data associated with UTIs from *E. coli* were preprocessed and analysed using both multi-label and single-label classification frameworks. Multiple ML models and feature selection strategies were evaluated to assess predictive performance and identify informative protein signatures associated with resistance phenotypes.

The results demonstrate that AMR phenotypes can be predicted from urinary proteomics data, although predictive performance varies substantially between antibiotics. While a unified multi-label framework was initially explored, single-label classification approaches generally provided more stable and interpretable results, reflecting the influence of antibiotic-specific class distributions and underlying biological variation.

Feature selection analyses showed that substantial dimensionality reduction is possible without major loss of predictive performance. In particular, models trained on reduced and stability-selected protein sets achieved comparable performance to full-feature models, suggesting that a limited subset of proteins captures a significant proportion of the predictive signal.

The study further highlights several important methodological considerations for proteomics-based ML, including the impact of class imbalance, feature selection stability, and antibiotic-specific modelling. These factors were found to strongly influence model behaviour and should be carefully considered when developing predictive models in high-dimensional biological datasets.

Although the cohort size was limited and external validation was not performed, the findings support the potential of urinary proteomics combined with ML as a complementary approach for AMR phenotype prediction.

6.1 Future Work

Several directions for future research may further improve the robustness and clinical relevance of the presented framework.

First, validation on larger and more diverse patient cohorts is necessary to evaluate

model generalisability across independent datasets and clinical settings. Expanding the analysis to include additional bacterial species and infection types would further enhance the applicability of the approach.

Second, future studies could explore more advanced ML and deep-learning methods capable of capturing complex nonlinear relationships in high-dimensional proteomics data. Integration of additional omics layers, such as genomics or transcriptomics, may further improve predictive performance and biological resolution.

Finally, future work should focus on the biological interpretation of the identified protein signatures. This could include filtering and prioritising features using relevant protein databases prior to model training in order to improve biological plausibility and reduce noise. Another potential approach is the application of sequence clustering methods, such as CD-HIT, to reduce redundancy when merging protein or sequence databases [37].

Bibliography

[1] Antimicrobial Resistance Collaborators. "Global burden of bacterial antimicrobial resistance in 2019: a systematic analysis". *Lancet*. (2022 Feb) 12;399(10325):629-655. doi: 10.1016/S0140-6736(21)02724-0. Epub (2022 Jan 19). Erratum in: Lancet. (2022 Oct) 1;400(10358):1102. doi: 10.1016/S0140-6736(21)02653-2. PMID: 35065702; PMCID: PMC8841637.

[2] Kaprou, G. D., Bergšpica, I., Alexa, E. A., Alvarez-Ordóñez, A., Prieto, M. (2021). "Rapid Methods for Antimicrobial Resistance Diagnostics". *Antibiotics*, 10(2), 209. https://doi.org/10.3390/antibiotics10020209

[3] Burton, R. J., Albur, M., Eberl, M., Cuff, S. M. (2019). "Using artificial intelligence to reduce diagnostic workload without compromising detection of urinary tract infections". *BMC medical informatics and decision making*, 19(1), 171.

[4] Swan AL, Mobasheri A, Allaway D, Liddell S, Bacardit J. "Application of machine learning to proteomics data: classification and biomarker identification in postgenomics biology". *OMICS.* 2013 Dec;17(12):595-610. doi: 10.1089/omi.2013.0017. Epub (2013 Oct 12). PMID: 24116388; PMCID: PMC3837439.

[5] Shahin-Shamsabadi A, Cappuccitti J, "Proteomics and machine learning: Leveraging domain knowledge for feature selection in a skeletal muscle tissue meta-analysis", *Heliyon*, Volume 10, Issue 24, (2024), e40772, ISSN 2405-8440, https://doi.org/10.1016/j.heliyon.2024.e40772

[6] Salam MA, Al-Amin MY, Salam MT, Pawar JS, Akhter N, Rabaan AA, Alqumber MAA. "Antimicrobial Resistance: A Growing Serious Threat for Global Public Health". *Healthcare* (Basel). (2023 Jul) 5;11(13):1946. doi: 10.3390/healthcare11131946. PMID: 37444780; PMCID: PMC10340576.

[7] David M. Livermore, "Bacterial Resistance: Origins, Epidemiology, and Impact", *Clinical Infectious Diseases*, Volume 36, Issue Supplement_1, (January 2003), Pages S11–S23, https://doi.org/10.1086/344654

[8] Gajic, I., Kabic, J., Kekic, D., Jovicevic, M., Milenkovic, M., Mitic Culafic, D., Trudic, A., Ranin, L., Opavski, N. (2022). "Antimicrobial Susceptibility

Testing: A Comprehensive Review of Currently Used Methods". *Antibiotics*, 11(4), 427. https://doi.org/10.3390/antibiotics11040427

[9] Flores-Mireles AL, Walker JN, Caparon M, Hultgren SJ. "Urinary tract infections: epidemiology, mechanisms of infection and treatment options". *Nat Rev Microbiol.* (2015 May) ;13(5):269-84. doi: 10.1038/nrmicro3432. Epub 2015 Apr 8. PMID: 25853778; PMCID: PMC4457377.

[10] Pothoven R. "Management of urinary tract infections in the era of antimicrobial resistance". *Drug Target Insights.* (2023 Dec) 20;17:126-137. doi: 10.33393/dti.2023.2660. PMID: 38124759; PMCID: PMC10731245.

[11] Kalantari, Shiva, Jafari, Ameneh, Moradpoor, Raheleh, Ghasemi, Elmira, Khalkhal, Ensieh, "Human Urine Proteomics: Analytical Techniques and Clinical Applications in Renal Diseases", *International Journal of Proteomics*, (2015), 782798, 17 pages, 2015. https://doi.org/10.1155/2015/782798

[12] Bilal Aslam, Madiha Basit, Muhammad Atif Nisar, Mohsin Khurshid, Muhammad Hidayat Rasool, "Proteomics: Technologies and Their Applications", *Journal of Chromatographic Science*, Volume 55, Issue 2, 1 February 2017, Pages 182–196, https://doi.org/10.1093/chromsci/bmw167

[13] Aebersold, R., Mann, M. Mass spectrometry-based proteomics. *Nature* 422, 198–207 (2003). https://doi.org/10.1038/nature01511

[14] Vitko D, Chou WF, Nouri Golmaei S, Lee JY, Belthangady C, Blume J, Chan JK, Flores-Campuzano G, Hu Y, Liu M, Marispini MA, Mora MG, Ramaswamy S, Ranjan P, Williams PB, Zawada RJX, Ma P, Wilcox BE. "timsTOF HT Improves Protein Identification and Quantitative Reproducibility for Deep Unbiased Plasma Protein Biomarker Discovery". *J Proteome Res.* (2024 Mar) 1;23(3):929-938. doi: 10.1021/acs.jproteome.3c00646. Epub 2024 Jan 15. PMID: 38225219; PMCID: PMC10913052.

[15] Rahman, A. (2019). "Statistics-based data preprocessing methods and machine learning algorithms for big data analysis". *International Journal of Artificial Intelligence*, 17(2), 44-65.

[16] Feng C, Wang H, Lu N, Chen T, He H, Lu Y, Tu XM. "Log-transformation and its implications for data analysis". *Shanghai Arch Psychiatry.* (2014) Apr;26(2):105-9. doi: 10.3969/j.issn.1002-0829.2014.02.009. Erratum in: Gen Psychiatr. 2019 Sep 6;32(5):e100146corr1. doi: 10.1136/gpsych-2019-100146corr1. PMID: 25092958; PMCID: PMC4120293.

[17] Dubois, E., Galindo, A. N., Dayon, L., Cominetti, O. (2022). "Assessing normalization methods in mass spectrometry-based proteome profiling of clinical samples". *Biosystems,* 215, 104661.

[18] Cerulli, G. (2023). "Fundamentals of supervised machine learning". *Cham: Springer.*

[19] The Scikit-learn community. “Multiclass and multioutput algorithms”, Accessed: May, 2026. [Online]. Available: https://scikit-learn.org/stable/modules/multiclass.html#multilabel-classification

[20] Mahesh R N, U., Nelleri, A. (2023). "Multi-Class Classification and Multi-Output Regression of Three-Dimensional Objects Using Artificial Intelligence Applied to Digital Holographic Information". *Sensors*, 23(3), 1095. https://doi.org/10.3390/s23031095

[21] Breiman, L. "Random Forests". *Machine Learning* 45, 5–32 (2001). https://doi.org/10.1023/A:1010933404324

[22] Widodo, S., Brawijaya, H., Samudi, S. (2022). "Stratified K-fold cross validation optimization on machine learning for prediction". *Sinkron: jurnal dan penelitian teknik informatika*, 6(4), 2407-2414.

[23] Suthaharan, S. (2016). "Support Vector Machine. In: Machine Learning Models and Algorithms for Big Data Classification". *Integrated Series in Information Systems*, vol 36. Springer, Boston, MA. https://doi.org/10.1007/978-1-4899-7641-3_9

[24] Weerts HJP, Mueller AC, Vanschoren J, "Importance of Tuning Hyperparameters of Machine Learning Algorithms", *arXiv*, (2022), eprint 2007.07588. https://arxiv.org/abs/2007.07588

[25] The Scikit-learn community. “RandomForestClassifier,” Accessed: Oct, 2026. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

[26] The Scikit-learn community. “LogisticRegression,” Accessed: Dec, 2026. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

[27] Martinez, K.M., Wilding, K., Llewellyn, T.R. et al. "Evaluating the factors influencing accuracy, interpretability, and reproducibility in the use of machine learning classifiers in biology to enable standardization". *Sci Rep 15*, 16651 (2025). https://doi.org/10.1038/s41598-025-00245-6

[28] S. Khalid, T. Khalil and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," (2014) Science and Information Conference, London, UK, 2014, pp. 372-378, doi: 10.1109/SAI.2014.6918213.

[29] Zollanvari, A. (2023). Feature Selection. In: "Machine Learning with Python". *Springer*, Cham. https://doi.org/10.1007/978-3-031-33342-2_10

[30] Fonti, V., Belitser, E. (2017). "Feature selection using lasso". *VU Amsterdam research paper in business analytics*, 30, 1-25.

[31] Zhou, J., Gandomi, A. H., Chen, F., Holzinger, A. (2021). "Evaluating the quality of machine learning explanations: A survey on methods and metrics". *Electronics*, 10(5), 593.

[32] Handelman, G. S., Kok, H. K., Chandra, R. V., Razavi, A. H., Huang, S., Brooks, M., Asadi, H. (2019). "Peering into the black box of artificial intelligence: evaluation metrics of machine learning methods". *American Journal of Roentgenology*, 212(1), 38-43.

[33] Rainio, O., Teuho, J. Klén, R. "Evaluation metrics and statistical tests for machine learning". *Sci Rep 14*, 6086 (2024). https://doi.org/10.1038/s41598-024-56706-x

[34] Hinojosa Lee, M. C., Braet, J., Springael, J. (2024). "Performance Metrics for Multilabel Emotion Classification: Comparing Micro, Macro, and Weighted F1-Scores". *Applied Sciences*, 14(21), 9863. https://doi.org/10.3390/app14219863

[35] Alturayeif N, Hassine J. "Data leakage detection in machine learning code: transfer learning, active learning, or low-shot prompting?" *PeerJ Comput Sci.* (2025 Mar) 5;11:e2730. doi: 10.7717/peerj-cs.2730. PMID: 40134878; PMCID: PMC11935776.

[36] Bernett, J., Blumenthal, D.B., Grimm, D.G. et al. "Guiding questions to avoid data leakage in biological machine learning applications". *Nat Methods 21*, 1444–1453 (2024). https://doi.org/10.1038/s41592-024-02362-y

[37] Weizhong L. (2019), CD-HIT (Version 4.8.1). Accessed: May, 2026. [Online] Available: https://github.com/weizhongli/cdhit

A

Comparison between different classifiers

A.1 Random Forest

Table A.1: Classification performance using RF, showing mean accuracy, macro F1-score, and weighted F1-score for each antibiotic.

Antibiotic	Accuracy	Macro-F1	Weighted-F1
Ceftazidim	0.898485	0.473104	0.850762
Ciprofloxacin	0.848485	0.458874	0.779221
Piperacillin/Tazobactam	0.813636	0.448442	0.744675
Tobramycin	0.578788	0.499563	0.547391
Trimetoprim	0.510606	0.334795	0.429399

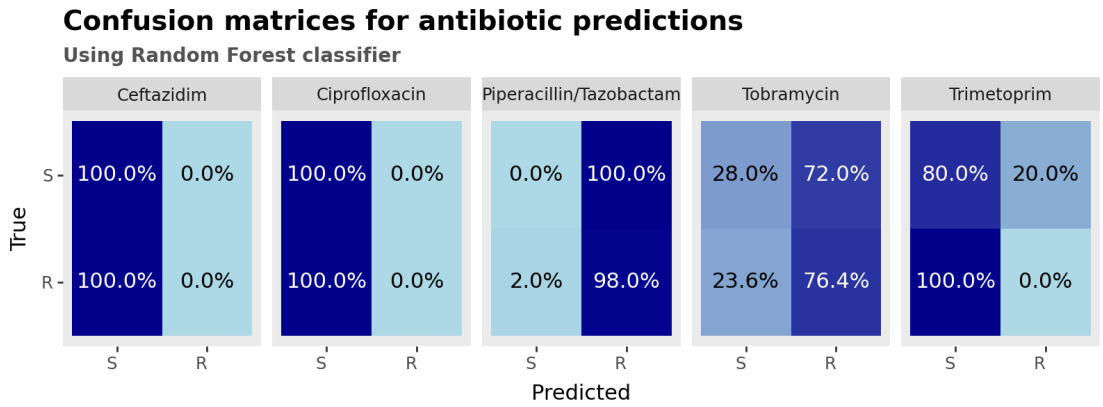

Figure A.1: Confusion matrices for the RF classifier.

A.2 Logistic Regression

Table A.2: Classification performance using Logistic regression classifier, showing mean accuracy, macro F1-score, and weighted F1-score for each antibiotic.

Antibiotic	Accuracy	Macro-F1	Weighted-F1
Ceftazidim	0.675758	0.488483	0.732202
Ciprofloxacin	0.598485	0.498921	0.631748
Piperacillin/Tazobactam	0.615152	0.516333	0.652402
Tobramycin	0.557576	0.520469	0.541153
Trimetoprim	0.442424	0.410137	0.430311

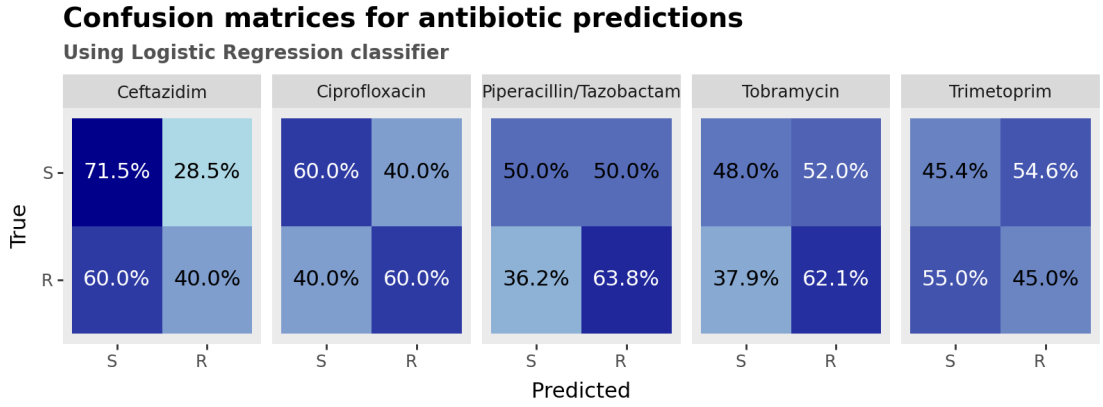

Figure A.2: Confusion matrices for the Logistic regression classifier.

A.3 Support Vector Machines

Table A.3: Classification performance using the SVM classifier, showing mean accuracy, macro F1-score, and weighted F1-score for each antibiotic.

Antibiotic	Accuracy	Macro-F1	Weighted-F1
Ceftazidim	0.898485	0.473104	0.850762
Ciprofloxacin	0.848485	0.458874	0.779221
Piperacillin/Tazobactam	0.830303	0.453636	0.753333
Tobramycin	0.627273	0.385146	0.484253
Trimetoprim	0.627273	0.385146	0.496534

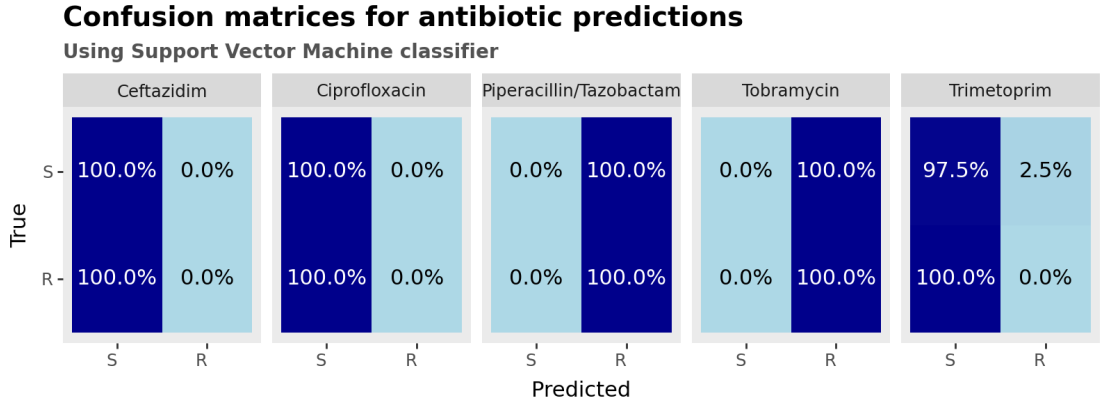

Figure A.3: Confusion matrices for the SVM classifier.

B

Comparison between different feature selection methods

B.1 Filtering missing values

Table B.1: Classification performance when filtering missing values as feature selection, showing mean accuracy, macro F1-score, and weighted F1-score for each antibiotic.

Antibiotic	Accuracy	Macro-F1	Weighted-F1
Ceftazidim	0.898485	0.473104	0.850762
Ciprofloxacin	0.848485	0.458874	0.779221
Piperacillin/Tazobactam	0.796970	0.443247	0.736017
Tobramycin	0.578788	0.463940	0.521946
Trimetoprim	0.577273	0.469726	0.529620

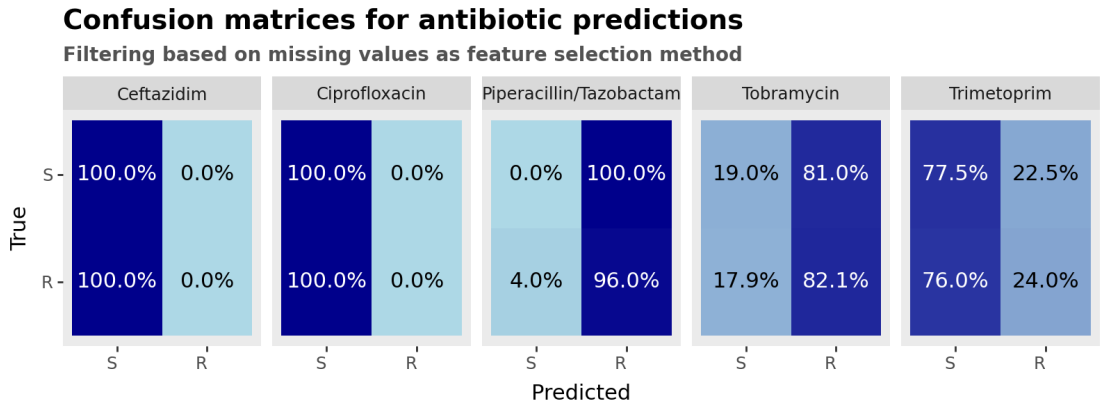

Figure B.1: Confusion matrices for the single-label classification model when filtering missing values is used for feature selection.

B.2 Statistical t-test

Table B.2: Classification performance using t-test as feature selection, showing mean accuracy, macro F1-score, and weighted F1-score for each antibiotic.

Antibiotic	Accuracy	Macro-F1	Weighted-F1
Ceftazidim	0.898485	0.473104	0.850762
Ciprofloxacin	0.763636	0.429394	0.729293
Piperacillin/Tazobactam	0.796970	0.492338	0.751169
Tobramycin	0.456061	0.351733	0.416546
Trimetoprim	0.543939	0.432435	0.501832

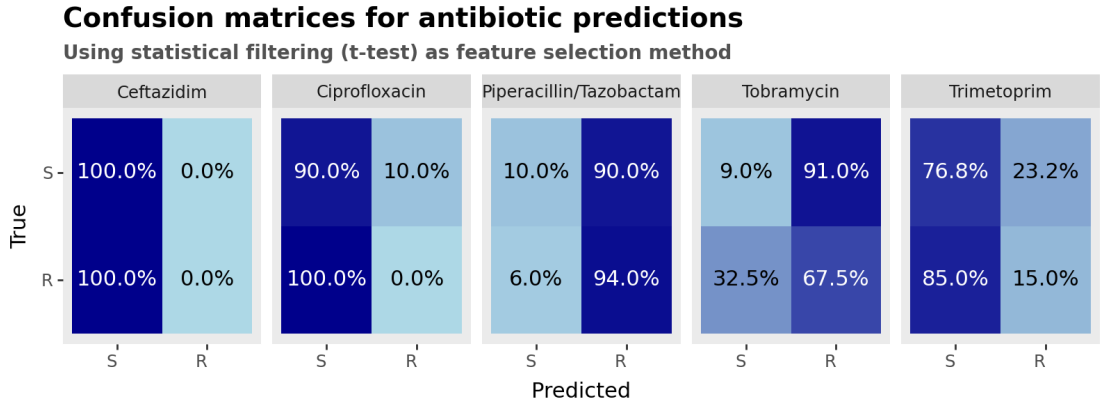

Figure B.2: Confusion matrices for the single-label classification model using t-test as feature selection.

B.3 LASSO

Table B.3: Classification performance using LASSO as feature selection, showing mean accuracy, macro F1-score, and weighted F1-score for each antibiotic.

Antibiotic	Accuracy	Macro-F1	Weighted-F1
Ceftazidim	0.898485	0.473104	0.850762
Ciprofloxacin	0.848485	0.458874	0.779221
Piperacillin/Tazobactam	0.813636	0.448442	0.744675
Tobramycin	0.578788	0.499563	0.547391
Trimetoprim	0.527273	0.380145	0.457485

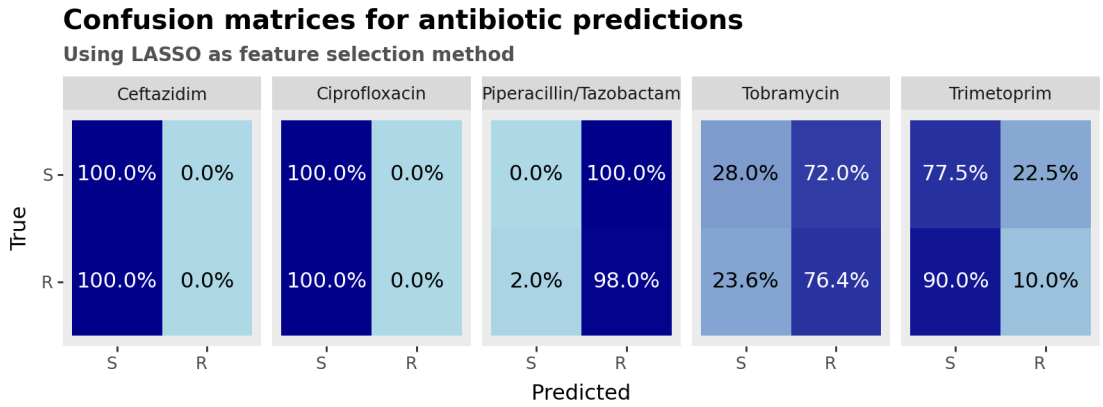

Figure B.3: Confusion matrices for the single-label classification model using LASSO as feature selection.

B.4 Elastic net

Table B.4: Classification performance using Elastic net as feature selection, showing mean accuracy, macro F1-score, and weighted F1-score for each antibiotic.

Antibiotic	Accuracy	Macro-F1	Weighted-F1
Ceftazidim	0.898485	0.473104	0.850762
Ciprofloxacin	0.848485	0.458874	0.779221
Piperacillin/Tazobactam	0.796970	0.443247	0.736017
Tobramycin	0.578788	0.499563	0.547391
Trimetoprim	0.527273	0.385079	0.465762

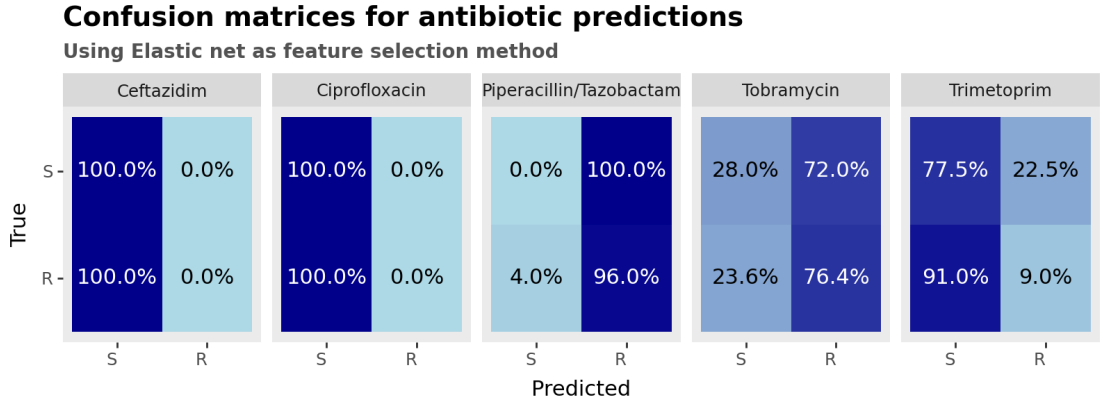

Figure B.4: Confusion matrices for the single-label classification model using Elastic net as feature selection.

C

Top 30 stable features identified for each antibiotic

Ceftazidim	Ciprofloxacin	Piperacillin/Tazobactam	Tobramycin	Trimetoprim
WP_002354936.1	NP_418718.4	P53985	Q9BSQ5	Q17RN3
HBW4268457.1	P67870	HBW4267851.1	WP_013095032.1	P08034
NP_414897.1	NP_416069.2	P01772	Q9C0I1	NP_415340.1
Q9UBU6	CAH5607516.1	O75173	Q8N511	HBW4270523.1
Q8WUF5	NP_418555.1	O95450	P02671	Q96HP8
O60749	Q99470	NP_417359.1	NP_417259.1	HBW4266481.1
Q96CP6	HBW4267777.1	HBW4267143.1	Q9BQ61	O75071
Q8NBX0	Q86T13	P34897	O43306	Q8N556
P51153	O95352	WP_230134272.1	A0A024R1R8	Q3ZCV2
Q9UBK8	Q9HB19	HBW4271608.1	HBW4271603.1	WP_230134272.1
P31512	Q96NZ9	Q96H55	P30679	NP_416876.1
Q96GE9	Q13564	WP_041080602.1	P83881	WP_002357765.1
Q96SL1	HBW4267668.1	Q9Y6M7	Q8IXK0	Q6AI08
Q6UWF7	Q9P035	Q96FF7	Q8NHL6	Q8TDW0
Q7Z794	Q9H0Q0	O75880	NP_416367.1	Q99626
P36959	P07306	O00115	Q9Y259	WP_219248991.1
NP_416861.1	NP_414847.3	P78329	Q8TE02	NP_416693.1
Q9UL52	Q9Y265	NP_416142.1	WP_230134272.1	CAH5712960.1
NP_414828.1	P01258	Q93062	Q96LB4	P07359
A0A0A0MT89	NP_415172.1	NP_416186.4	Q93052	Q9Y530
Q9NWH9	HBW4266729.1	Q6PKC3	HBW4271608.1	HBW4269559.1
Q9NWB6	YP_026224.1	Q9Y4G6	Q9UHY7	NP_418493.1
HBW4266944.1	NP_416138.1	Q9UNW1	HBW4266515.1	O14684
Q9BRT2	Q9NNX6	Q99727	Q9BXW7	Q9NXV2
Q9Y3L3	NP_415774.1	WP_002483004.1	HBW4268500.1	HBW4268838.1
P01871	Q92990	HBW4271044.1	Q15021	Q99996
O75506	Q99962	HBW4266176.1	Q9NR28	WP_001144069.1
Q9UMZ2	NP_418239.1	P01782	NP_415276.1	WP_069819505.1
Q8TBZ3	Q9ULV4	HBW4269346.1	P48449	Q99719
NP_418771.1	Q99983	Q6IN84	Q99727	Q5MNZ9

DEPARTMENT MATHEMATICAL SCIENCES
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY