Feature Selection based Multivariate Time Series Forecasting: An Application to Antibiotic Resistance Outbreaks Prediction

Fernando Jiménez[a], José Palma[a], Gracia Sánchez[a], David Marín[a], Francisco Palacios, M.D[a,b], Lucía López, M.D[b]

[a]*Artificial Intelligence and Knowledge Engineering Group. Faculty of Computer Science. University of Murcia, Spain*
[b]*Universitary Hospital of Getafe, Madrid, Spain*

Abstract

Antimicrobial resistance has become one of the most important health problems and global action plans have been proposed globally. Prevention plays a key role in these actions plan and, in this context, we propose the use of *Artificial Intelligence*, specifically *Time Series Forecasting* techniques, for predicting future outbreaks of *Methicilin-resistant Staphylococcus aereus* (MRSA). Infection incidence forecasting is approached as a *Feature Selection based Time Series Forecasting* problem using multivariate time series composed of incidence of *Staphylococcus Aereus Methicilin-sensible* and MRSA infections, influenza incidence and total days of therapy of both of *Levoflaxin* and *Oseltamivir* antimicrobials. Data were collected from the University Hospital of Getafe (Spain) from January 2009 to January 2018, using months as time granularity. The main contributions of the work are the following: the applications of wrapper feature selection methods where the search strategy is based on multi-objective evolutionary algorithms (MOEA) along with evaluators based on the most powerful state-of-the-art regression algorithms. The performance of the feature selection methods has been measured using the *root mean square error* (*RMSE*) and *mean absolute error* (*MAE*) performance metrics. A novel multi-criteria decision-making process is proposed in order to select the most satisfactory forecasting model, using the metrics previously mentioned, as well as the slopes of model prediction lines in the 1, 2 and 3 steps-ahead predictions. The multi-criteria decision-making process is applied to the best models resulting from a ranking of databases and regression algorithms obtained through multiple statistical tests. Finally, to the best of our knowledge, this is the first time that a feature selection based multivariate time series methodology is proposed for antibiotic resistance forecasting. Final results show that the best model according to the proposed multi-criteria decision making process provides a $RMSE = (0.1349, 0.1304, 0.1325)$ and a $MAE = (0.1003, 0.096, 0.0987)$ for 1, 2, and 3 steps-ahead predictions.

Keywords: Feature Selection, Multi-objective Evolutionary Algorithms, Multivariate Time Series, Antibiotic Resistance Forecasting, Multiple Criteria Decision Making.

1. Introduction

The discovery of penicillin in 1928 constituted a great stride in human health, providing us with a simple cure for infections that, otherwise, could cause death. Nonetheless, the massive use of antimicrobials and, more importantly, their misuse are threatening with an increasing spread of multi-resistant bacteria which can cause infections with fatal consequences. According to recent studies, antimicrobial resistance (AMR) is estimated to be responsible for 25,000 death per year in the EU [1] and 700,000 deaths per year globally, and it is estimated that, by 2050, deaths caused by AMR surpass death caused by cancer. As a result, AMR has become one of the most important health problems and global action plans have been proposed globally both at European Union [2] and worldwide levels [3]. Prevention plays a key role in these actions plan and, in this context, we proposed the use of *Artificial Intelligence*, specifically *Time Series Forecasting* techniques, for predicting outbreaks of multi-resistant bacteria from hospital-level data.

To this end, we have focused on infections caused by *Methicilin-sensible Staphylococcus Aureus* (SA) and *Methicilin-resistant Staphylococcus Aereus* (MRSA). SA is an important human pathogen associated with a wide range of infections, ranging from infective endocarditis to skin and device-related infections [4]. Furthermore, SA is the leading cause of nosocomial pneumonia, surgical site infections [5] and community-acquired pneumonia, which is a well known as a potentially catastrophic complication of influenza with a high mortality rate [6, 7]. MRSA is

a methicilin-resistant SA strain which can persist not only in hospital (where the use of an antimicrobial agent is high) but also at the community level. MRSA, isolated in 1961, is still an important life-threatening multi-drug resistant pathogen, being the major cause of nosocomial infections although, from 1997, has also been detected at community level infections [8]. One of the major concerns related to MRSA is that its resistant pattern change to adapt to each new antimicrobial agents. Pneumonia infections (both nosocomial and community-acquired), caused by either SA or MRSA can be a consequence of influenza complications. Consequently, being able to predict the incidence of them could have positive implications in the definitions of protocols for empiric antimicrobial regimens as well as preventive measures such as seasonal influenza vaccination campaigns [9]. In this work, infection incidence forecasting is approached as a time series forecasting problem using five time series involved with the influenza protocol collected from the Hospital Universitario de Getafe (Spain).

Time series forecasting is the process of using a model to generate forecasts for future events based on known past events. Time series data have a natural temporal ordering. This differs from typical machine learning applications where each data point is an independent example of the concept to be learned, and the ordering of data points within a data set does not matter. For this reason, standard machine learning methods should not be used directly to analyse time series data. In this paper, we propose a methodology to, firstly, transform the time series into a form that standard machine learning algorithms can process, and then, systematically apply a set of feature selection methods for regression [10]. Time series data is transformed by removing the temporal ordering of individual input examples and adding a set of delays to the input which are called *lagged variables* and provide the temporal information. This approach to time series forecasting is more powerful and more flexible than classical statistics techniques such as *ARMA* and *ARIMA* [11]. Feature selection methods are applied for the selection of lagged variables. *Random Forest*, *Instance-Based learning*, *Linear Regression*, *Support Vector Machines*, *Gaussian Processes* and *Deep Learning* algorithms are used in this paper for regression with the different reduced databases. We also consider an autoregressive model in the set of experiments. A multi-criteria decision making process is applied to the best forecasting models resulting from statistical tests in order to choose the most satisfactory model for the *h-steps-ahead* predictions, where *RMSE* and *MAE* are used as performance metrics. The experiments have been carried out using the *Waikato Environment for Knowledge Analysis* (*Weka*) [12] and the packages *caret* [13] and *marima* of R [14]. In summary, the main contributions of the work are the following:

1. We have applied wrapper feature selection methods where the search strategy is based on multi-objective evolutionary computation, and evaluators based on the most powerful regression algorithms of the state-of-the-art have been used, among which Support Vector Machines, Gaussian Processes and Deep Learning are included. These multivariate wrapper feature selection methods have been compared to other popular wrapper methods such as *Recursive Feature Elimination* or filter methods such as *Minimum Redundancy - Maximum Relevance* and *Correlation-based Feature Selection*. Autoregressive models built with *MARIMA* and *VAR* have also been included in the comparison set.
2. Both *RMSE* and *MAE* metrics have been used in the configuration of wrapper feature selection methods to measure the merit of candidate attribute subsets.
3. We have proposed a novel *multi-criteria decision-making process* to choose the most satisfactory forecast model, which uses *RMSE* and *MAE* metrics, as well as slopes of the prediction lines of the models in *h-steps-ahead* predictions. The multi-criteria decision-making process is applied to the 10 best models resulting from a ranking of databases and regression algorithms obtained through multiple statistical tests.
4. Finally, to the best of our knowledge this is the first time that a multivariate time series feature selection methodology is proposed for predicting antibiotic resistance.

With this background the paper has been organized as follows: section 2 defines the concept of feature selection and their categorization, shows the related works and describes the data set used for experiments; section 3 proposes a methodology for multivariate time series forecasting of antibiotic resistance based on feature selection; section 4 analyses and discusses the results, and finally section 5 concludes the paper.

2. Background

2.1. Feature Selection

Feature Selection (FS) is defined in [15] as the process of eliminating features from the database that are irrelevant

to the task to be performed. FS facilitates data understanding, reduces the measurement and storage requirements, the computational process time, and the size of a data set, so that model learning becomes an easier process. An FS method is basically a *search strategy* where the performance of candidate subsets is measured with a given *evaluator*. The search space for candidate subsets has cardinality $O(2^w)$, where w is the number of features. A *stopping criterion* establishes when the FS process must finish. It can be defined as a control procedure that ensures that no further addition or deletion of features produces a better subset, or it can be as simple as a counter of iterations. FS methods are typically categorized into *wrapper*, *filter* and *embedded*, *univariate* and *multivariate* methods. *Wrapper methods* [16] use a predetermined learning algorithm to determine the quality of selected features according to an evaluation metric [17]. *Filter methods* apply statistical measures to evaluate the set of attributes [18, 19, 20]. *Embedded methods* achieve model fitting and FS simultaneously [21]. *Multivariate methods* evaluate features in batches. *Univariate methods* evaluate each feature independently.

Some advantages and disadvantages are as follows: filter methods are computationally faster than wrappers; wrapper methods are more accurate than filters; wrapper methods perform feature selection depending on the learning algorithm used, while the feature selection made by the filters methods is conceived for a more general-purpose (statistical); embedded methods perform the feature selection and learning processes simultaneously, which is, at the same time, an advantage and a disadvantage (both processes are integrated, but the feature selection may not be optimal); univariate methods do not take into account interactions between factors, while multivariate do.

2.2. Multi-objective Evolutionary Feature Selection

The first evolutionary approach involving multi-objective optimization for FS was proposed by Ishibuchi [22] in 2000. Since then, many multi-objective evolutionary approaches for FS have appeared in the literature, both filter and wrapper methods, and in supervised and unsupervised environments. Below are some of the most relevant works published during the last five years.

Kimovski et. al. [23] propose a parallel multi-objective optimization approach to cope with high-dimensional FS problems. Several parallel multi-objective evolutionary alternatives are proposed and experimentally evaluated by using some synthetic and *BCI* (Brain-Computer Interface) benchmarks. Paul and Das [24] propose a filter FS method for simultaneous FS and weighting. They use inter-class and intra-class distance measures which are maximized and minimized simultaneously by using a MOEA based on *Decomposition* (*MOEA/D* [25]). Jiménez et al. [26] propose a MOEA for FS, called *ENORA*, applied in on-line sales forecasting. *ENORA* is implemented as a wrapper FS method for regression tasks, where the *RMSE* obtained with *Random Forest* for the selected attributes is minimized along with the number of selected attributes. In [27], Jiménez et al. investigate whether the use of MOEAs is more appropriate for FS than the single-objective evolutionary algorithms, as well as the accuracy metric versus the area under the ROC curve, in the context of virtual screening for drug discovery in classification tasks. Jiménez et al. [28] propose a wrapper FS method for fuzzy rule-based classification systems where both the search strategy and the evaluator consist of a MOEA. Due to the high complexity of the method, the authors investigate the optimal configuration for the population size and the number of generations in both MOEAs, obtaining a compromise between the performance of the classifier and the run time.

Multi-objective Differential Evolution (MODE) has also been applied successfully to FS in recent years. Sikdar et al. [29] apply a wrapper FS method for entity extraction in biomedical texts using classifier ensemble evaluated with F-measure and number of selected attributes as objectives, which is compared with the existing biomedical entity extraction systems that were developed using the same datasets. Nayak et al. [30] propose a filter approach (*FAEMODE*) using elitism based MODE algorithm for FS. The dependency of feature subset with the target class is maximized, and feature redundancy is minimized. Both linear and non-linear dependency among features was considered to handle the redundant and unwanted features of a dataset. Results were compared with filter and wrapper methods. Another multi-objective approach based on differential evolution (*DEMO*) has been proposed by Mlakar et al. [31] as wrapper FS method for facial expression recognition. The number of used features was minimized, while the emotion recognition accuracy of the support vector machine classifiers was maximized simultaneously. The results have been compared state-of-the-art methods, where *NSGA-II* [32] is included. Das et al. [33] propose a filter FS method using a MODE with two objectives: set approximation accuracy of rough set theory and relational algebra based derived score. Hancer et al. [34] propose a filter FS approach consist of a three-objective differential evolution algorithm to optimize mutual relevance, *ReliefF* ranking and *Fisher Score* ranking. Finally, Hancer also proposes MODE based filter FS methods which combine standard mutual information and fuzzy mutual information in [35], and

fuzzy and kernel-based information measures in [36]. Bidgoli et al. [37] propose a MODE algorithm for multi-label FS with considering number of features and classification accuracy as objectives. A binary operator is proposed based on opposition-based learning concept and partially voting between two candidate solutions to decide about the absence or presence of a feature in the third randomly selected solution. The proposed operator is used in third evolution step of generalized differential evolution (*GDE3* [38]). Zhang et al. [39] propose a binary differential evolution algorithm with self-learning strategy, called *MOFS-BDE*, to solve multi-objective feature selection problems. *MOFS-BDE* uses a new binary mutation operator based on probability difference to guide the individuals to locate potentially optimal areas fast, a new one-bit purifying search operator (OPS) for improving the self-learning capability of elite individuals, and a non-dominated sorting operator with crowding distance to reduce the time consumption of the selection operator in differential evolution. Zhang et al. also proposes other metaheuristics for multi-objective feature selection, such as *particle swarm optimization* and *artificial bee colony algorithm*, where cost-based feature selection [40], cost-sensitive feature selection [41], and feature selection of unreliable data [42] are focused.

2.3. Related Work

Tyralis and Papacharalampous [43] conducted a study on the optimal number of lag variables that should be used for time series forecasting with *Random Forest*. Crone and Kourentzes [44] performed feature selection for time series prediction using a neural network. By combining contemporaneous and lagged realisations of the independent variables and lagged dependent variables more general models of dynamic regression, autoregressive (*AR*) transfer functions and intervention models are constructed. It has also been done by Sun et al. [45] using the Granger causality discovery to identify important features with effective sliding window sizes, considering the influence of lagged observations of features on the target time series. Hido and Morimura [46] have searched for the optimal time-windows and time lags for each variable based on feature pre-processing and sparse learning in order to configure the input dataset.

The first works of the application of time series analysis to antibiotic resistance were performed by López-Lozano et al. [47, 48]. The authors demonstrate a temporal relationship between antimicrobial use and resistance, to quantify the effect of their use on resistance and to estimate the delay between variations of use and subsequent variations in resistance. Willmann et al. [49] investigate the association between antimicrobial use and resistance rates in Pseudomonas Aeruginosa by using time series analysis. Erdeljić et al. [50] compare two different commonly used statistical methods in their ability to investigate the relationship between antipseudomonal antimicrobial consumption and resistance rates of P. Aeruginosa isolate in a single Intensive Care Unit (ICU) of a tertiary hospital, namely simple linear correlation (Pearson's r) and distributed lags time series analysis. Faust et al. [51] apply time series analysis on multiple longitudinal datasets to illustrate their potential for microbiome research. Dalum Hansen et al. [52] study how well antimicrobial drug consumption can be predicted based on web search queries, compared to historical purchase data of antimicrobial drugs. First, they select web search queries that are likely to indicate antimicrobial drug consumption; then, for each query frequency time series they generate several lagged versions and decide which lags should be used for the prediction; and finally they use appropriate prediction models (*Linear Elastic Net* and *Gaussian Processes*) to infer antimicrobial drug consumption.

2.4. Antibiotic resistance dataset

For this experiment, we have used a multivariate time series dataset with five time series. Each time series is composed of 108 events collected from a hospital between January 2009 to January 2018, using months as time granularity. Figure 1 shows the five time series used in this work and the unit in which each series is measured is shown in Table 1. We have selected these time series since they are involved in the influenza protocol. Influenza first symptoms are treated with Oseltamivir© antiviral drug to improve disease symptoms. In order to prevent bacterial infections as a complication of influenza, Levofalxin© antibiotic is also administrated. The most common, and more risky, bacterial infections are those provoked by SA and MRSA which, as said before, can lead to fatal consequences.

SA and MRSA time series are measured in monthly incidences and Levoflaxin and Oseltamivir in total days of therapy (DOT) by months. Incidence is calculated as a proportion between the number of inpatients that, before the moment that it is calculated, are affected by the event (that is, new events) and the total number of inpatients. That is to say, the incidence of influenza in march, 31st (the measures are taking at the end of the month) indicates the number of inpatients affected by influenza divided by the total number of inpatients. DOT represents the number of

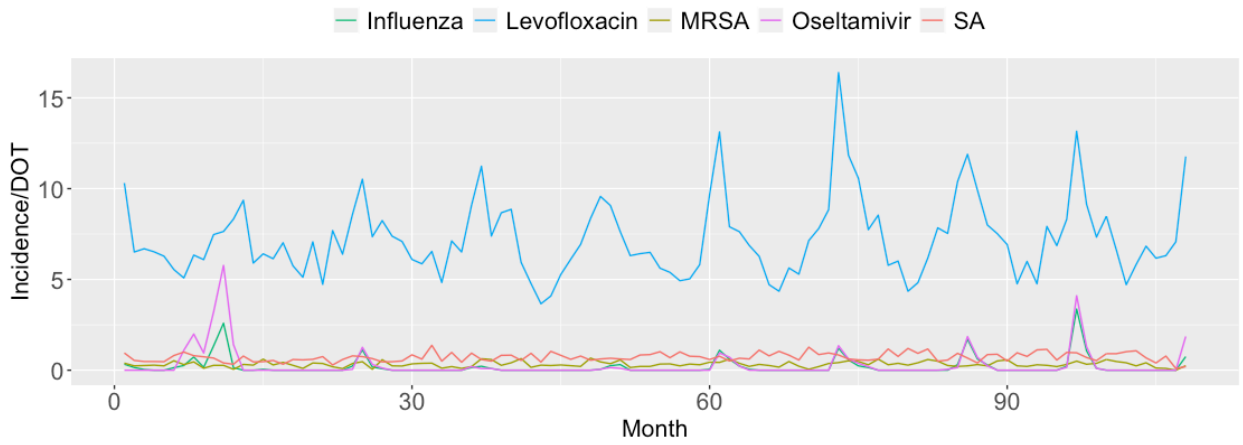

Figure 1: Time series used in this work.

Series	*Unit*
Staphylococcus aureus meticilin sensible (SA)	Incidence
Staphylococcus aureus meticilin resistant (MRSA)	Incidence
Influenza	Incidence
Levoflaxin	Days of Therapy (DOT)
Oseltamivir	Days of Therapy (DOT)

Table 1: Time series considered in this work and their units.

days in which a patient is treated with the corresponding antimicrobial. Therefore, in the corresponding time series is represent the sum of DOT over a month in the hospital. The choice of time series is justified by the bibliographical evidence that interrelates them:

- **MRSA**: The association between exposure to antibiotics and the isolation of MRSA has been demonstrated in [53]. Explicitly, the Society for Healthcare Epidemiology of America (SHEA) guidelines for preventing nosocomial transmission of MRSA and Vancomycin-Resistant Enterococci (VRE), highlights the importance of reducing quinolones (levofloxacin) for the prevention of MRSA infection in hospitals [54].
- **Influenza**: Influenza increases the chance of getting a secondary bacterial infection. However, the proportion of patients who develop a bacterial infection related to a flu episode is relatively small. Influenza per se is not an indication of antibiotic treatment [55, 56].
- **SA**: Unlike MRSA, it is usually sensitive to Levoflaxin©. This is why the use of Levoflaxin© creates the conditions in which MRSA acquires a competitive advantage by eradicating susceptible microorganisms with SA [57].
- **Oseltamivir©**: It is indicated as a symptomatic treatment of influenza, restricted in our environment to the seasonal period of influenza. It is an antiviral treatment without any antibacterial activity [58].
- **Levofloxacin©**: There is evidence of seasonal prescription of levofloxacin© for respiratory pathologies and that this seasonal prescription coincides with that of influenza. It is estimated that a 20% reduction in the incidence of influenza leads to an 8% reduction in levofloxacin© [59].

3. A methodology for multivariate time series forecasting of antibiotic resistance based on feature selection

We have followed the methodology shown in the Figure 2 to perform the viral incidence time series forecasting. The following five steps have been systematically applied: database transformation, feature selection, forecasting, statistical tests and decision making. Next, each step of the methodology is described separately in a section. Table 13 summarizes the *Weka* and *R* packages, classes and functions used in this work to implement the proposed methodology.

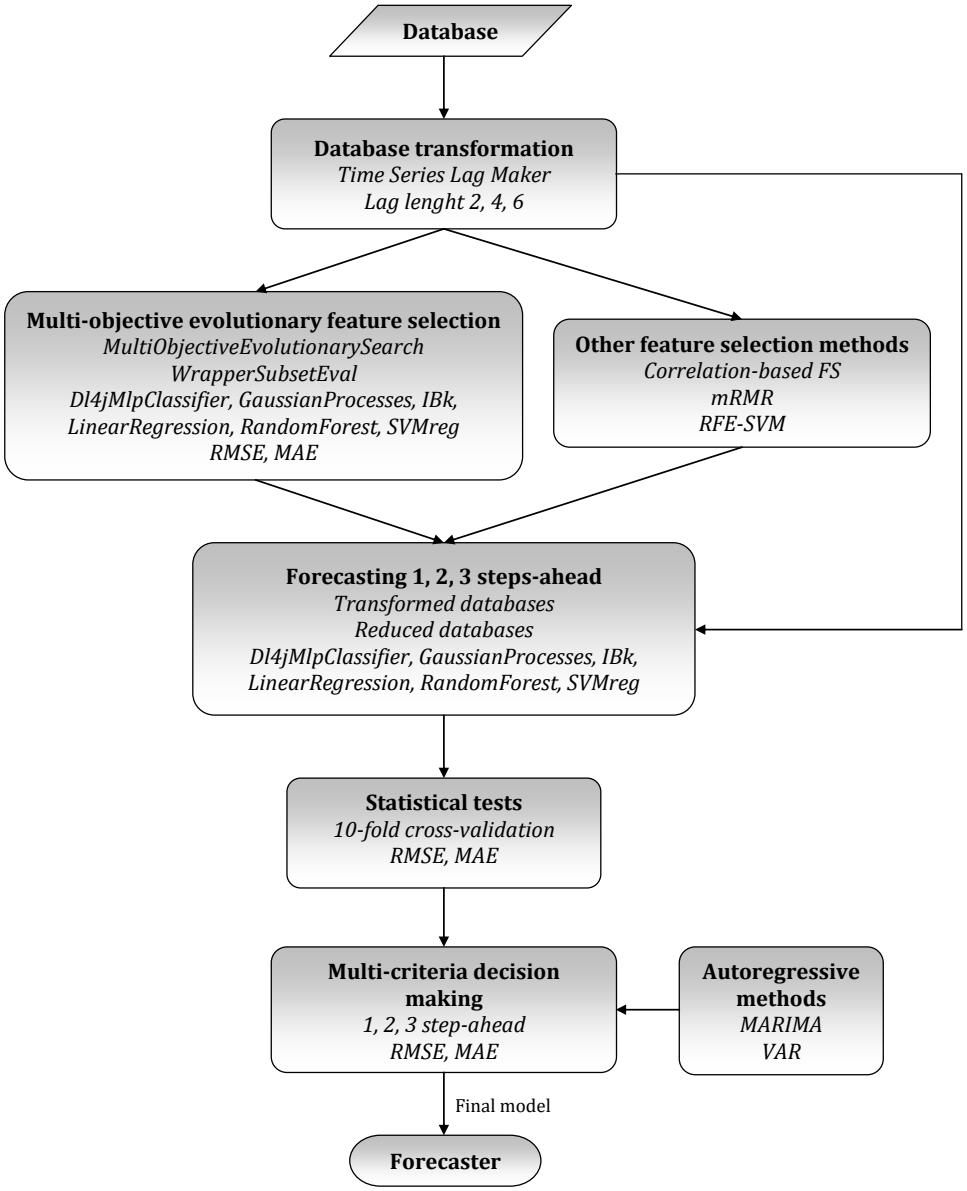

Figure 2: Methodology for feature selection for antibiotic resistance multivariate time series forecasting.

3.1. Database transformation

The first step of our methodology is to transform the database by creating lagged versions of variables for use in the time series problem. We use the class *weka.classifiers.timeseries.core.TSLagMaker* for this task. Data transformation can be done from the plugin tab in *Weka*'s graphical "Explorer" user interface, or and using the *API* through a *Java* program. The following considerations have been taken into account for the database transformation:

1. Since the database does not contain a date field, we use an artificial time index.
2. The system can jointly model multiple attributes to lag simultaneously to capture dependencies between them. Because of this, modelling several series simultaneously can give different results for each series than modelling them individually. We set all the attributes 1 to 5 as lagged attributes.
3. We have experienced setting the *maximum lag length* to 2, 4 and 6. A value of n means that a lagged variable will be created that holds target values at time n. All time periods between the minimum and maximum lag will be turned into lagged variables. In this way, for example with *minimum lag length* equal to 0 and *maximum lag length* equal to 2 for the variable *Staphylococcus_aureus meticilin_resistant*, this variable will be transformed into 3 lagged variables *Lag_Staphylococcus_aureus meticilin_resistant+0* (equivalent to the variable *Staphylococcus_aureus_meticilin_resistant*) *Lag_Staphylococcus_aureus_meticilin_resistant-1* and *Lag_Staphylococcus-_aureus_meticilin_resistant-2*. Lagged variables are the main mechanism by which the relationship between past and current values of a series can be captured by propositional learning algorithms. Lagged variables create a "window" over a time period. So, the number of lagged variables created determines the size of the window. For example, if we have monthly data then including lags up to 12 time steps into the past would make sense. We have tested with 6, 4 and 2 lag lengths, which represent window-size of half-year, four-months and two-months respectively.
4. Three transformed databases have been created (one for each lag length 2, 4 and 6) containing respectively 16, 26 and 36 attributes (($lag\ length + 1) \cdot 5 + 1$ attributes in total, where 5 is the number of lagged attributes).
5. We save the transformed databases with the names *LL2*, *LL4* and *LL6*. These transformed databases will be used later in the forecasting phase.

3.2. Feature selection

Once the transformed databases *LL2*, *LL4* and *LL6* are obtained, the next step is to apply FS to each of them. In *Weka*, FS is implemented with the class *weka.attributeSelection.AttributeSelection* through two components: the *search strategy* (*weka.attributeSelection.ASSearch* abstract class) and the *evaluator* (*weka.attributeSelection.ASEvaluation* abstract class). This allows users and programmers to configure a multitude of different methods for FS, both filter and wrapper, univariate and multivariate. We are interested in the wrapper methods due to its greater precision. The search strategy *MultiObjectiveEvolutionarySearch* has been developed by authors of this paper demonstrating a great efficiency in feature selection problems for regression [26]. *MultiObjectiveEvolutionarySearch* class has two multi-objective evolutionary algorithms implemented, *ENORA* and *NSGA-II*. *ENORA* [60] is our *MOEA*, on which we are intensively working over the last decade. *NSGA-II* [32] was designed by K. Deb et al. and has been proved to be a very powerful and fast algorithm in multi-objective optimization contexts. In [26] is statistically tested that *ENORA* performs better than *NSGA-II* in terms of *hypervolume* [61, 62] for regression tasks, for which we have decided to use *ENORA* in this work. *ENORA* is an elitist Pareto-based multi-objective evolutionary algorithm that uses a $(\mu + \lambda)$ survival with a uniform random initialization, binary tournament selection, ranking based on a local non-domination level with crowding distance, self-adaptive uniform crossover and self-adaptive one-bit flip mutation. *MultiObjectiveEvolutionarySearch*, in conjunction with *WrapperSubsetEval*[1] [16], solves the following 2-objective optimization problem:

[1] *Weka* incorporates the class *WrapperSubsetEval* to allow users to configure wrapper FS methods with attribute subset evaluation. *WrapperSubsetEval* evaluates the worth of an attribute subset using a user-specified learning algorithm (classification or regression), p-fold cross-validation (by default 5), and a measure to evaluate the performance of the learning algorithm with the attribute subset. The performance measure could be accuracy (only for nominal class), *RMSE* (over the probabilities of the nominal class), *MAE* (over the probabilities of the nominal class), *F-measure* (only for nominal class), *area under the ROC curve* (only for nominal), *area under the precision-recall curve* (nominal class only), and *correlation coefficient* (numerical class only). Repeating cross-validation (5 times as maximum) is required if the standard deviation of the mean exceeds a threshold (by default 0.01).

$$\begin{array}{ll} Maximize & \mathcal{F}_{CV}^{\Phi}(\mathbf{x}, p) \\ Minimize & C(\mathbf{x}) \end{array} \tag{1}$$

where $\mathbf{x} = \{x_1, x_2, \ldots, x_w\}$ is a boolean set of decision variables, i.e. $x_k \in \{true, false\}$, $k = 1, \ldots, w$, being w the number of attributes of the database. $\mathcal{F}_{CV}^{\Phi}(\mathbf{x}, p)$ function measures the performance of a learning algorithm Φ trained with the attributes $x_k = true$, $k = 1, \ldots, w$, and evaluated with p-fold cross-validation, with $2 \leq p \leq T$, where T is the number of instances of the database. $C(\mathbf{x})$ function measures the number of selected attributes, i.e.

$$C(\mathbf{x}) = \sum_{k=1}^{w} \mathcal{N}(x_k)$$

where $\mathcal{N}$ is a function that transforms a boolean value into numeric ($true = 1$ and $false = 0$). Note that some performance metrics such as *RMSE* or *MAE* (commonly used in regression problems) must be multiplied by -1 so that the function $\mathcal{F}_{CV}^{\Phi}(\mathbf{x}, p)$ remain maximized. The problem (1) is therefore a *multi-objective boolean optimization problem* where $x_k = 1$ represents that attribute x_k is selected, and $x_k = 0$ represents that attribute x_k is not selected, for all $k = 1, \ldots, w$. The non-dominated solution in the last population with the best fitness for the first objective is chosen as output.

The methodology proposed in this paper includes 12 multivariate wrapper FS methods that are the result of combining *MultiObjectiveEvolutionarySearch* with 6 regression algorithms and 2 performance metrics by using the evaluator *WrapperSubsetEval*. We considered for this research the regression algorithms *Deep Learning* [63] (*Dl4jMlpClassifier* in *Weka*), *Gaussian Processes* [64] (*GaussianProcesses* in *Weka*), *k-Nearest Neighbours* [65] (*IBk* in *Weka*), *Linear Regression* [66] (*LinearRegression* in *Weka*), *Random Forest* [67] (*RandomForest* in *Weka*) and *Support Vector Machine* [68] (*SMOreg* in *Weka*), combined with the *RMSE* and *MAE* performance metrics [69]. *Dl4jMlpClassifier* [70] is a deep learning package for the *Weka* workbench (classification and regression with multi-layer perceptrons using *DeepLearning4J*). Following the recommendations of [71] for multivariate time series forecasting, we use an architecture with an internal dense layer of 100 output units, with activation function *ActivationRELU* and loss function *LossMSE*. *GaussianProcesses* implements gaussian processes for regression without hyperparameter-tuning. To make choosing an appropriate noise level easier, this implementation applies normalization/standardization to the target attribute as well as the other attributes. We have used the *polynomial kernel* [72] in the experiments. *IBk* is the *k-nearest neighbours* classifier that is also valid for regression. *k-NN* is a type of *instance-based learning*, or *lazy learning*. *LinearRegression* uses linear regression for prediction, with the *Akaike* criterion for model selection. *RandomForest* is an *ensemble learning* method which constructs a forest of random trees with controlled variance, for classification or regression purposes. *SMOreg* implements the support vector machine for regression. The parameters can be learned using various algorithms. We use support vector machines for regression using *Sequential Minimal Optimization* with Shevade et al. adaptation of the stopping criterion [68] and polynomial kernel.

We have also considered other non-evolutionary FS methods widely used in the literature: *Correlation-based Feature Selection* (*CFS*) [73], *Minimum Redundancy Maximum Relevance* (*mRMR*) [74] and *Recursive Feature Elimination* (*RFE*) [75]. *CFS* method evaluates how well each attribute is able to predict the target as well as the similarity degree between the attributes. In such a way that the feature sets correlated with the target and with features poorly correlated with each other obtain the higher scores. *CFS* has been used in this paper in conjunction with the *best-first* search strategy [76], which searches the space of attribute subsets by greedy hill-climbing augmented with a backtracking facility. We have used the implementation in *Weka* (*BestFirst* and *CfsSubsetEval* classes). *mRMR* method ranks the attributes based on their relevance to the target and, at the same time, penalizes their redundancy. Once the attributes have been ranking, we use sequential forward selection along with random forest to obtain a attribute subset. *RFE* method is basically a backward elimination procedure. This technique is an iterative process that begins by building a model with all the attributes and calculating the importance of each of them. In each iteration the least important attributes are removed, the model is rebuilt and the importance of each attribute is calculated again. The optimal subset is used to train the final model. We use support vector machines for regression with kernel radial. Table 2 summarizes the FS methods used in this paper, indicating the short name, type, search strategy and evaluator of each of the 15 methods. Table 12 shows the parameters used for each FS method.

Short name	*Type*	*Search strategy*	*Evaluator*
BF-CFS	*Filter*, *Multivariate*	*Best First, Floating Selection*	*Correlation*
MOES-DL-MAE	*Wrapper*, *Multivariate*	*Multi-objective Evolutionary*	*Deep Learning – MAE*
MOES-DL-RMSE	*Wrapper*, *Multivariate*	*Multi-objective Evolutionary*	*Deep Learning – RMSE*
MOES-GP-MAE	*Wrapper*, *Multivariate*	*Multi-objective Evolutionary*	*Gaussian Processes – MAE*
MOES-GP-RMSE	*Wrapper*, *Multivariate*	*Multi-objective Evolutionary*	*Gaussian Processes – RMSE*
MOES-IBk-MAE	*Wrapper*, *Multivariate*	*Multi-objective Evolutionary*	*k-Nearest Neighbours – MAE*
MOES-IBk-RMSE	*Wrapper*, *Multivariate*	*Multi-objective Evolutionary*	*k-Nearest Neighbours – RMSE*
MOES-LR-MAE	*Wrapper*, *Multivariate*	*Multi-objective Evolutionary*	*Linear Regression – MAE*
MOES-LR-RMSE	*Wrapper*, *Multivariate*	*Multi-objective Evolutionary*	*Linear Regression – RMSE*
MOES-RF-MAE	*Wrapper*, *Multivariate*	*Multi-objective Evolutionary*	*Random Forest – MAE*
MOES-RF-RMSE	*Wrapper*, *Multivariate*	*Multi-objective Evolutionary*	*Random Forest – RMSE*
MOES-SMOreg-MAE	*Wrapper*, *Multivariate*	*Multi-objective Evolutionary*	*Support Vector Machine – MAE*
MOES-SMOreg-RMSE	*Wrapper*, *Multivariate*	*Multi-objective Evolutionary*	*Support Vector Machine – RMSE*
mRMR-RF	*Filter + Wrapper*, *Multivariate*	*Ranking + Forward Selection*	*Redundancy/Relevance + Random Forest – RMSE*
RFE-SVM	*Wrapper*, *Multivariate*	*Ranking + Backward Selection*	*Support Vector Machine – RMSE*

Table 2: Proposed feature selection methods for antibiotic resistance forecasting.

3.3. Forecasting

We have considered three lag lengths (2, 4 and 6) and fifteen FS methods. Each FS method has been applied to these databases transformed with 2, 4 and 6 lag length, obtaining $3 \cdot 15 = 45$ reduced databases. It is important to indicate that some feature selection methods may not select any lag variable for the output variable. In that case, the corresponding reduced dataset is discarded from the process. In our experiments, 14 reduced datasets have been discarded, resulting in a total of $45 - 14 = 31$ reduced databases. This gives a total of $3 + 31 = 34$ databases (3 transformed databases plus 31 reduced databases). The six regression algorithms *Dl4jMlpClassifier*, *GaussianProcesses*, *IBk*, *LinearRegression*, *RandomForest* and *SMOreg* have been applied to each of the 34 databases, resulting in a total of $34 \cdot 6 = 204$ forecasting models. 1, 2 and 3 steps-ahead predictions have been made to analyze the 204 models, thus making a total of $204 \cdot 3 = 612$ predictions. All forecasting models have been trained on the first 70% of the instances and tested on the last 30% of the instances.

Our methodology also makes predictions with *autoregressive models* using the *marima* and *vars* packages. The *vars* package [77] fits a *VAR* model, and the *marima* package fits *MARIMA* model using the Spliid's algorithm [14]. Since *VAR* model includes only autoregressive terms [78], a *VARMA* model, which includes both autoregressive and moving average terms [79], has also been considered. However, the contribution of the moving average component of the *VARMA* model was negligible, so only *VAR* model was taking into account, together with the *MARIMA* model.

3.4. Statistical tests

The next step in our methodology is to perform statistical tests to detect statistically significant differences between the reduced databases, on the one hand, and between the regression algorithms, on the other hand. For this, each of the 6 regression algorithms was executed with each of the 34 databases, and the models obtained were evaluated with 10-fold cross validation and 1 step-ahead predictions. In order to get statistically meaningful results, the number of iterations was 10. This means 100 calls of each regression algorithm for each database with training data and tested against test data . Tables 3 and 4 show the results for the *RMSE* and *MAE* metrics, respectively. In order to identify each database, the databases with 2, 4 and 6 lag length have been named as *LL2*, *LL4* and *LL6* respectively. The reduced databases have been named with the short name of the FS method followed by *LL2*, *LL4* or *LL6* as appropriate. For example, the *MOES-DL-RMSE* method applied to the *LL2* database results in the reduced database called *MOES-DL-RMSE-LL2*.

We have performed multiple *paired t-tests* with significance 0.05 to compare each database with the others, on the one hand, and each regression algorithm with the others, on the other hand, for both *RMSE* and *MAE* metrics. Afterwards, a ranking test of databases (for *RMSE* and *MAE* separately) and a ranking test of regression algorithms (for *RMSE* and *MAE* separately) have been performed. The ranking test ranks the schemes according to the total number of significant wins and losses against the other schemes. The difference between the number of wins and the number of losses is used to generate the ranking. Tables 5 and 6 show the ranking test for databases and regression algorithms, respectively.

Database	*Dl4jMlpClassifier*	*GaussianProcesses*	*IBk*	*LinearRegression*	*RandomForest*	*SMOreg*
LL2	0.1497	0.1496	0.2169	0.1473	0.1480	0.1473
LL4	0.1542	0.1487	0.2001	0.1606	0.1446	0.1526
LL6	0.1550	0.1504	0.2068	0.1717	0.1436	0.1700
BF-CFS-LL2	0.1458	0.1473	0.2051	0.1398	0.1506	0.1367
BF-CFS-LL4	0.1478	0.1385	0.2033	0.1447	0.1406	0.1390
BF-CFS-LL6	0.1458	0.1353	0.2177	0.1475	0.1413	0.1426
MOES-DL-MAE-LL2	0.1468	0.1478	0.2011	0.1476	0.1480	0.1450
MOES-DL-MAE-LL4	0.1401	0.1389	0.1950	0.1438	0.1500	0.1400
MOES-DL-MAE-LL6	**0.1371**	0.1401	0.1829	0.1404	0.1375	0.1406
MOES-DL-RMSE-LL2	0.1443	0.1491	0.2017	0.1459	0.1565	0.1423
MOES-DL-RMSE-LL4	0.1395	0.1376	0.2116	0.1469	0.1499	0.1362
MOES-DL-RMSE-LL6	0.1377	0.1452	0.2001	0.1496	0.1505	0.1512
MOES-GP-MAE-LL4	0.1447	0.1357	0.2081	0.1415	0.1378	0.1329
MOES-GP-MAE-LL6	0.1409	0.1319	0.1974	0.1391	0.1388	0.1319
MOES-GP-RMSE-LL2	0.1500	0.1438	0.1950	0.1445	0.1526	0.1438
MOES-GP-RMSE-LL4	0.1443	0.1341	0.1995	0.1370	0.1387	0.1294
MOES-GP-RMSE-LL6	0.1436	**0.1313**	0.1856	0.1423	0.1394	0.1306
MOES-IBk-MAE-LL4	0.1527	0.1472	0.1540	0.1493	0.1407	0.1495
MOES-IBk-MAE-LL6	0.1561	0.1467	**0.1471**	0.1602	0.1371	0.1496
MOES-IBk-RMSE-LL6	0.1439	0.1451	0.1530	0.1468	0.1422	0.1487
MOES-LR-MAE-LL4	0.1503	0.1408	0.1821	0.1369	0.1437	0.1362
MOES-LR-MAE-LL6	0.1439	0.1342	0.1951	0.1355	0.1379	0.1351
MOES-LR-RMSE-LL4	0.1476	0.1411	0.1625	**0.1351**	0.1404	0.1369
MOES-LR-RMSE-LL6	0.1439	0.1342	0.1951	0.1355	0.1379	0.1351
MOES-RF-MAE-LL2	0.1497	0.1486	0.1848	0.1483	0.1444	0.1433
MOES-RF-MAE-LL6	0.1464	0.1368	0.1890	0.1394	**0.1307**	0.1379
MOES-RF-RMSE-LL2	0.1513	0.1496	0.2022	0.1467	0.1421	0.1470
MOES-SMOreg-MAE-LL4	0.1471	0.1368	0.2139	0.1403	0.1412	0.1304
MOES-SMOreg-MAE-LL6	0.1487	0.1348	0.2178	0.1433	0.1405	0.1310
MOES-SMOreg-RMSE-LL4	0.1492	0.1364	0.1974	0.1388	0.1369	0.1304
MOES-SMOreg-RMSE-LL6	0.1473	0.1324	0.2132	0.1412	0.1396	**0.1290**
mRMR-RF-LL4	0.1500	0.1515	0.1953	0.1574	0.1494	0.1565
RFE-SVM-LL4	0.1549	0.1478	0.2010	0.1536	0.1441	0.1521
RFE-SVM-LL6	0.1538	0.1413	0.2123	0.1534	0.1415	0.1543

Table 3: Results of *RMSE* with 10-fold cross-validation, 10 repetitions, 1 step-ahead.

3.5. Multiple criteria decision making

In order to choose the best forecasting model we have considered the 10 best databases identified in the ranking tests for *RMSE* and *MAE* (the union of both sets of databases is a set of 11 databases). As can be seen in Table 5, all the chosen databases come from a wrapper FS method with the *MultiObjectiveEvolutionarySearch* search strategy. Next, we consider the models obtained with the regression algorithm used by the corresponding wrapper FS method to obtain each of the selected reduced databases. To these 11 forecasting models we add the autoregressive models corresponding to the *MARIMA* and *VAR* methods. Therefore we have a set of $n = 13$ forecasting models. The next step in our methodology is to compare the 13 forecasting models to choose the best. For this purpose, we propose the following *multiple criteria decision making* process:

Let $X = \{x_1, \ldots, x_n\}$ a set of n forecasting models. Each forecasting model is a pair *(database, algorithm)*, where *algorithm* is either a regression algorithm, or an autoregressive method. We consider the following *multi-objective optimization problem*:

$$\begin{array}{ll} Min & \mathcal{RMSE}(x,i), \quad i = 1,\ldots,h \\ Min & \mathcal{MAE}(x,i), \quad i = 1,\ldots,h \end{array} \tag{2}$$

In (2), $x \in X$ is a decision variable that represents a forecasting model, and $\mathcal{RMSE}(x,i)$ and $\mathcal{MAE}(x,i)$ is the *RMSE* and *MAE* respectively of the i steps-ahead prediction in test data (30%) for the forecasting model x (a total of $2 \cdot h$ objective functions for minimization). Solution of (2) is a set $S = \{s_1, \ldots, s_m\} \subset X$, $m \leq n$, of *non-dominated*

Database	*Dl4jMlpClassifier*	*GaussianProcesses*	*IBk*	*LinearRegression*	*RandomForest*	*SMOreg*
LL2	0.1234	0.1221	0.1751	0.1203	0.1229	0.1200
LL4	0.1283	0.1181	0.1626	0.1271	0.1190	0.1196
LL6	0.1269	0.1197	0.1662	0.1356	0.1177	0.1349
BF-CFS-LL2	0.1188	0.1219	0.1661	0.1142	0.1231	0.1096
BF-CFS-LL4	0.1205	0.1106	0.1634	0.1137	0.1138	0.1085
BF-CFS-LL6	0.1200	0.1074	0.1788	0.1153	0.1145	0.1126
MOES-DL-MAE-LL2	0.1190	0.1214	0.1654	0.1197	0.1222	0.1159
MOES-DL-MAE-LL4	0.1164	0.1141	0.1565	0.1182	0.1204	0.1169
MOES-DL-MAE-LL6	**0.1106**	0.1127	0.1509	0.1106	0.1115	0.1128
MOES-DL-RMSE-LL2	0.1186	0.1240	0.1608	0.1172	0.1287	0.1139
MOES-DL-RMSE-LL4	0.1150	0.1130	0.1713	0.1210	0.1216	0.1119
MOES-DL-RMSE-LL6	0.1133	0.1169	0.1686	0.1188	0.1233	0.1212
MOES-GP-MAE-LL4	0.1205	0.1073	0.1705	0.1122	0.1110	0.1033
MOES-GP-MAE-LL6	0.1149	**0.1034**	0.1572	0.1100	0.1122	0.1039
MOES-GP-RMSE-LL2	0.1250	0.1191	0.1517	0.1182	0.1270	0.1178
MOES-GP-RMSE-LL4	0.1189	0.1068	0.1603	0.1089	0.1122	**0.0989**
MOES-GP-RMSE-LL6	0.1171	0.1038	0.1511	0.1122	0.1138	0.1023
MOES-IBk-MAE-LL4	0.1265	0.1198	0.1207	0.1205	0.1167	0.1183
MOES-IBk-MAE-LL6	0.1286	0.1191	**0.1181**	0.1321	0.1122	0.1199
MOES-IBk-RMSE-LL6	0.1172	0.1176	0.1278	0.1179	0.1156	0.1181
MOES-LR-MAE-LL4	0.1239	0.1160	0.1444	0.1095	0.1169	0.1076
MOES-LR-MAE-LL6	0.1188	0.1060	0.1624	**0.1054**	0.1110	0.1056
MOES-LR-RMSE-LL4	0.1222	0.1157	0.1293	0.1074	0.1127	0.1074
MOES-LR-RMSE-LL6	0.1188	0.1060	0.1624	**0.1054**	0.1110	0.1056
MOES-RF-MAE-LL2	0.1232	0.1217	0.1496	0.1193	0.1199	0.1118
MOES-RF-MAE-LL6	0.1210	0.1081	0.1499	0.1093	**0.1035**	0.1087
MOES-RF-RMSE-LL2	0.1265	0.1236	0.1606	0.1194	0.1184	0.1174
MOES-SMOreg-MAE-LL4	0.1201	0.1096	0.1718	0.1119	0.1150	0.0991
MOES-SMOreg-MAE-LL6	0.1223	0.1062	0.1728	0.1136	0.1135	0.1007
MOES-SMOreg-RMSE-LL4	0.1233	0.1090	0.1601	0.1104	0.1118	0.0992
MOES-SMOreg-RMSE-LL6	0.1209	0.1036	0.1703	0.1115	0.1140	0.0996
mRMR-RF-LL4	0.1243	0.1242	0.1618	0.1281	0.1248	0.1263
RFE-SVM-LL4	0.1274	0.1174	0.1624	0.1210	0.1188	0.1185
RFE-SVM-LL6	0.1281	0.1128	0.1710	0.1215	0.1157	0.1245

Table 4: Results of *MAE* with 10-fold cross-validation, 10 repetitions, 1 step-ahead.

(or *Pareto*) solutions [80]. In order to choose a solution $s^* \in S$, we take into account the sum of the values of *RMSE* and *MAE* in the 1, 2 and 3 steps-ahead, together with the sum of the slopes (in absolute value) of the prediction lines evaluated with *RMSE* and *MAE* in 1, 2 and 3 steps-ahead. Algorithm 1 describes the full multiple criteria decision making process.

Solving the problem (2) we have obtained a set $S = \{s_1, s_2\}$ composed of 2 non-dominated solutions. The first solution s_1 is the regression model obtained with *GaussianProcesses* and the reduced database obtained with the wrapper FS method *MOES-GP-RMSE* from the transformed dataset with 6 lag length. The second solution s_2 is the autoregressive model obtained with *MARIMA*. Table 7 shows the *RMSE* and *MAE* of each solution in the 1, 2 and 3 steps-ahead. In addition, Table 7 shows the value v_j of each solution s_j, $j = 1, 2$, calculated by the algorithm 1. The solution s_1 is chosen as the best solution. This is because, in addition to having the lowest sum of *RMSE* and *MAE* in the 1, 2 and 3 steps-ahead predictions, the prediction lines of *RMSE* and *MAE* in the 1, 2 and 3 steps-ahead have a minimum sum of its slopes. This can be seen graphically in Figures **??** and **??**. In this way, the multiple criteria decision-making process takes into account the following aspects:

1. Multi-objective optimization problem (2), which identifies the best solutions in each step-ahead for both *RMSE* and *MAE* metrics.
2. The following criteria and aggregation operators are taken into account to distinguish between non-dominated solutions:
   (a) The joint optimality of the solution in all steps ahead. For this, we use the addition operator, in both *RMSE*

RMSE				
Rank	***Database***	***Wins***	***Losses***	***Difference***
1	*MOES-GP-RMSE-LL4*	46	4	42
2	*MOES-GP-MAE-LL6*	46	5	41
3	*MOES-SMOreg-RMSE-LL6*	42	5	37
4	*MOES-GP-RMSE-LL6*	39	3	36
5	*MOES-RF-MAE-LL6*	43	7	36
6	*MOES-SMOreg-RMSE-LL4*	41	6	35
7	*MOES-LR-RMSE-LL4*	39	5	34
8	*MOES-LR-MAE-LL6*	37	4	33
9	*MOES-LR-RMSE-LL6*	37	4	33
10	*MOES-GP-MAE-LL4*	28	4	24
11	*MOES-SMOreg-MAE-LL4*	31	7	24
12	*MOES-DL-MAE-LL6*	28	7	21
13	*MOES-SMOreg-MAE-LL6*	27	11	16
14	*MOES-IBk-RMSE-LL6*	29	16	13
15	*MOES-LR-MAE-LL4*	16	5	11
16	*MOES-DL-RMSE-LL4*	23	15	8
17	*BF-CFS-LL4*	19	12	7
18	*BF-CFS-LL6*	18	11	7
19	*MOES-DL-MAE-LL4*	19	12	7
20	*MOES-IBk-MAE-LL4*	31	32	-1
21	*MOES-IBk-MAE-LL6*	39	41	-2
22	*BF-CFS-LL2*	12	22	-10
23	*MOES-GP-RMSE-LL2*	5	20	-15
24	*MOES-RF-MAE-LL2*	3	23	-20
25	*MOES-DL-RMSE-LL6*	11	35	-24
26	*MOES-RF-RMSE-LL2*	1	26	-25
27	*MOES-DL-MAE-LL2*	1	29	-28
28	*LL2*	1	41	-40
29	*RFE-SVM-LL4*	3	43	-40
30	*RFE-SVM-LL6*	3	44	-41
31	*MOES-DL-RMSE-LL2*	3	47	-44
32	*LL4*	2	48	-46
33	*mRMR-RF-LL4*	0	53	-53
34	*LL6*	1	77	-76

MAE				
Rank	***Database***	***Wins***	***Losses***	***Difference***
1	*MOES-GP-RMSE-LL4*	58	5	53
2	*MOES-RF-MAE-LL6*	58	6	52
3	*MOES-SMOreg-RMSE-LL6*	55	5	50
4	*MOES-SMOreg-RMSE-LL4*	52	5	47
5	*MOES-GP-MAE-LL6*	52	6	46
6	*MOES-GP-RMSE-LL6*	47	3	44
7	*MOES-LR-MAE-LL6*	46	4	42
8	*MOES-LR-RMSE-LL6*	46	4	42
9	*MOES-GP-MAE-LL4*	43	5	38
10	*MOES-SMOreg-MAE-LL6*	45	8	37
11	*MOES-SMOreg-MAE-LL4*	43	8	35
12	*MOES-LR-RMSE-LL4*	40	11	29
13	*MOES-DL-MAE-LL6*	37	16	21
14	*BF-CFS-LL4*	24	14	10
15	*MOES-IBk-RMSE-LL6*	30	20	10
16	*BF-CFS-LL6*	24	17	7
17	*MOES-IBk-MAE-LL4*	29	27	2
18	*MOES-LR-MAE-LL4*	14	13	1
19	*MOES-IBk-MAE-LL6*	37	49	-12
20	*MOES-DL-MAE-LL4*	6	21	-15
21	*MOES-DL-RMSE-LL4*	12	27	-15
22	*BF-CFS-LL2*	9	26	-17
23	*MOES-RF-MAE-LL2*	2	20	-18
24	*MOES-DL-RMSE-LL6*	10	33	-23
25	*MOES-RF-RMSE-LL2*	1	28	-27
26	*MOES-DL-MAE-LL2*	1	39	-38
27	*MOES-GP-RMSE-LL2*	0	38	-38
28	*RFE-SVM-LL4*	3	41	-38
29	*RFE-SVM-LL6*	5	44	-39
30	*LL4*	2	50	-48
31	*MOES-DL-RMSE-LL2*	2	51	-49
32	*LL2*	0	52	-52
33	*LL6*	2	66	-64
34	*mRMR-RF-LL4*	0	73	-73

Table 5: Ranking of databases for *RMSE* and *MAE* metrics with 10-fold cross-validation, 10 repetitions, 1 step-ahead.

RMSE				
Rank	***Regression algorithm***	***Wins***	***Losses***	***Difference***
1	*GaussianProcesses*	46	0	46
2	*SMOreg*	42	3	39
3	*RandomForest*	37	3	34
4	*LinearRegression*	33	12	21
5	*Dl4jMlpClassifier*	31	19	12
6	*IBk*	0	152	-152

MAE				
Rank	***Regression algorithm***	***Wins***	***Losses***	***Difference***
1	*SMOreg*	61	3	58
2	*GaussianProcesses*	52	6	46
3	*RandomForest*	36	9	27
4	*LinearRegression*	37	13	24
5	*Dl4jMlpClassifier*	30	35	-5
6	*IBk*	0	150	-150

Table 6: Ranking of regression algorithms for *RMSE* and *MAE* metrics with 10-fold cross-validation, 10 repetitions, 1 step-ahead.

and *MAE* metrics separately.

(b) The robustness of the forecasting model along all steps ahead. For this, we use the sum of the slopes of the prediction lines (in absolute value) between every two steps ahead, again in both *RMSE* and *MAE* metrics separately.

(c) The optimality and robustness of the forecasting model are aggregated into a single function by means of the the multiplication operator (step 8 of Algorithm 1).

4. Analysis of results and discussion

The best solution with our methodology comes from applying the *MOES-GP-RMSE* method on the transformed database *LL6*. Figure 4 shows the evolution of the average *hypervolume ratio* [62] over 10 runs of the *MOES-GP-*

Algorithm 1 Multiple criteria decision making.

Require: $X = \{x_1, \ldots, x_n\}$ {Set of n forecasting models}
Require: h {Number of steps ahead}
1: $S = \{s_1, \ldots, s_m\} \leftarrow$ Solution of the multi-objective optimization problem (2)
2: $\mathcal{RMSE}'(s_j, i) \leftarrow$ Normalized $\mathcal{RMSE}(s_j, i)$, $j = 1, \ldots, m$, $i = 1, \ldots, h$
3: $\mathcal{MAE}'(s_j, i) \leftarrow$ Normalized $\mathcal{MAE}(s_j, i)$, $j = 1, \ldots, m$, $i = 1, \ldots, h$
4: $sRMSE_j \leftarrow \sum_{i=1}^{h} \mathcal{RMSE}'(s_j, i)$, $j = 1, \ldots, m$
5: $sMAE_j \leftarrow \sum_{i=1}^{h} \mathcal{MAE}'(s_j, i)$, $j = 1, \ldots, m$
6: $mRMSE_j \leftarrow \sum_{i=1}^{h-1} \left|\mathcal{RMSE}'(s_j, i+1) - \mathcal{RMSE}'(s_j, i)\right|$, $j = 1, \ldots, m$
7: $mMAE_j \leftarrow \sum_{i=1}^{h-1} \left|\mathcal{MAE}'(s_j, i+1) - \mathcal{MAE}'(s_j, i)\right|$, $j = 1, \ldots, m$
8: $v_j \leftarrow sRMSE_j \cdot mRMSE_j + sMAE_j \cdot mMAE_j$, $j = 1, \ldots, m$
9: $s^* \leftarrow s_{min} \mid v_{min} = \min_{j=1}^{m}\{v_j\}$
10: **return** s^*

	RMSE			***MAE***				
Database / Algorithm	***1 step-ahead***	***2 steps-ahead***	***3 steps-ahead***	***1 step-ahead***	***2 steps-ahead***	***3 steps-ahead***	***Dominated***	v_j
MOES-GP-MAE-LL4 / GaussianProcesses	0.1434	0.1382	0.1405	0.1107	0.1059	0.1091	Yes	–
MOES-GP-MAE-LL6 / GaussianProcesses	0.1394	0.1346	0.1367	0.1046	0.1000	0.1027	Yes	–
MOES-GP-RMSE-LL4 / GaussianProcesses	0.1430	0.1387	0.1410	0.1094	0.1051	0.1085	Yes	–
MOES-GP-RMSE-LL6 / GaussianProcesses (s_1)	**0.1349**	**0.1304**	**0.1325**	**0.1003**	**0.0960**	**0.0987**	**No**	$v_1 = 0.0030$ *
MOES-LR-MAE-LL6 / LinearRegression	0.1413	0.1367	0.1377	0.1056	0.1012	0.1011	Yes	–
MOES-LR-RMSE-LL4 / LinearRegression	0.1413	0.1367	0.1377	0.1056	0.1012	0.1011	Yes	–
MOES-LR-RMSE-LL6 / LinearRegression	0.1413	0.1367	0.1377	0.1056	0.1012	0.1011	Yes	–
MOES-RF-MAE-LL6 / RandomForest	0.1486	0.1395	0.1417	0.1157	0.1090	0.1120	Yes	–
MOES-SMOreg-MAE-LL6 / SMOreg	0.1580	0.1525	0.1546	0.1187	0.1135	0.1154	Yes	–
MOES-SMOreg-RMSE-LL4 / SMOreg	0.1560	0.1497	0.1518	0.1173	0.1118	0.1135	Yes	–
MOES-SMOreg-RMSE-LL6 / SMOreg	0.1514	0.1473	0.1497	0.1138	0.1096	0.1122	Yes	–
Original / MARIMA (s_2)	**0.1463**	**0.1352**	**0.1305**	**0.1204**	**0.1078**	**0.1074**	**No**	$v_2 = 0.0232$
Original / VAR	0.1628	0.1715	0.1590	0.1342	0.1352	0.1293	No	–

Table 7: Results of the multi-criteria decision making applied to the best databases obtained from statistical tests together with the *MARIMA* and *VAR* autoregressive models.

RMSE method with *LL6*. The hypervolume is defined as the volume of the search space dominated by a population P, and is formulated as:

$$HV(P) = \bigcup_{i=1}^{|Q|} v_i \tag{3}$$

where $Q \subseteq P$ is the set of non-dominated individuals of P, and v_i is the volume of the individual i. Subsequently, the *hypervolume ratio* (*HVR*) is defined as the ratio of the volume of the non-dominated search space over the volume of the entire search space, and is formulated as follows:

$$HVR(P) = 1 - \frac{HV(P)}{VS} \tag{4}$$

where VS is the volume of the search space. Computing *HVR* requires reference points that identify the maximum and minimum values for each objective. For optimization problem (2), the following reference points ($\mathcal{F}_{min}, C_{min}$) and ($\mathcal{F}_{max}, C_{mas}$) are set:

$$\mathcal{F}_{min} = 0, \quad \mathcal{F}_{max} = \max_k \mathcal{F}_k^{\Phi}, \; k = 1, \ldots, w, \quad C_{min} = 1, \quad C_{max} = w$$

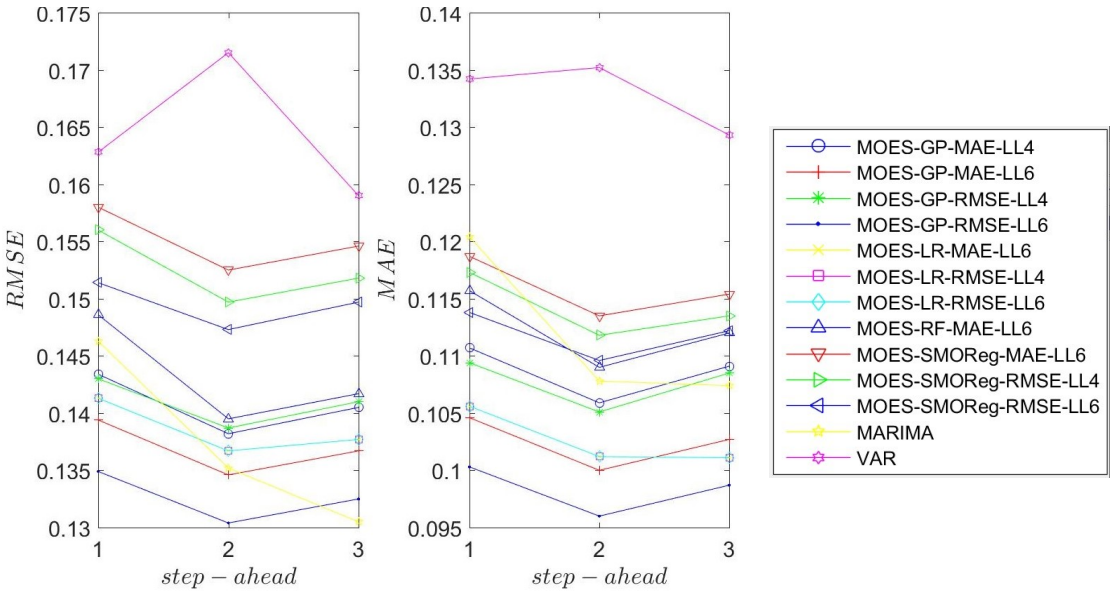

Figure 3: *RMSE* (a) and *MAE* (b) in 1, 2 and 3 steps-ahead of the non-dominated solutions.

where $\mathcal{F}_k^{\Phi}$ is the error (*RMSE* or *MAE* as appropriate) of the regression algorithm Φ over the database composed by only one attribute k. Note that if any individual of the population has a worst value than $\mathcal{F}_{max}$, then that individual is not taken into account in the calculation of the hypervolume because it is dominated by the point with objective values ($\mathcal{F}_{max}$, C_{min}). The table 8 shows statistics for the hypervolume obtained with 10 runs of algorithm (minimum, maximum, mean, standard deviation of mean, confidence interval for the mean). Figure 5 shows the Pareto front of the run with seed 1 corresponding to the solution considered in this paper. The selected solution is the one with the lowest $\mathcal{F}_{CV}^{\Phi}$ (solution with 11 attributes in Figure 5).

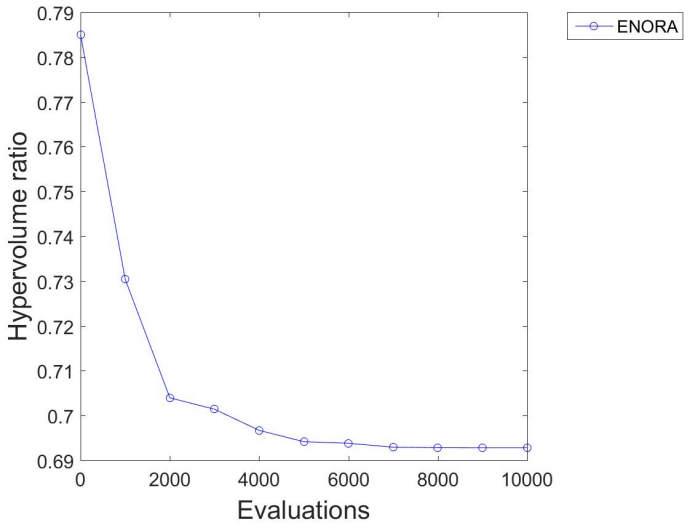

Figure 4: Evolution of the average hypervolume ratio with 10 runs of *MOES-GP-RMSE* with the transformed database *LL6*.

In order to further analyze the effectiveness of the FS process, we compared the best forecasting model ob-

	Minimum	*Maximum*	*Mean*	*S.D.*	*C.I. Low*	*C.I. High*
ENORA	0.6833	0.6992	0.6930	0.0049	0.6895	0.6965
S.D. = Standard deviation of mean						
C.I. = Confidence interval for the mean (95%)						

Table 8: Statistics for the hypervolume ratio obtained with 10 runs of *MOES-GP-RMSE* with the transformed database *LL6*.

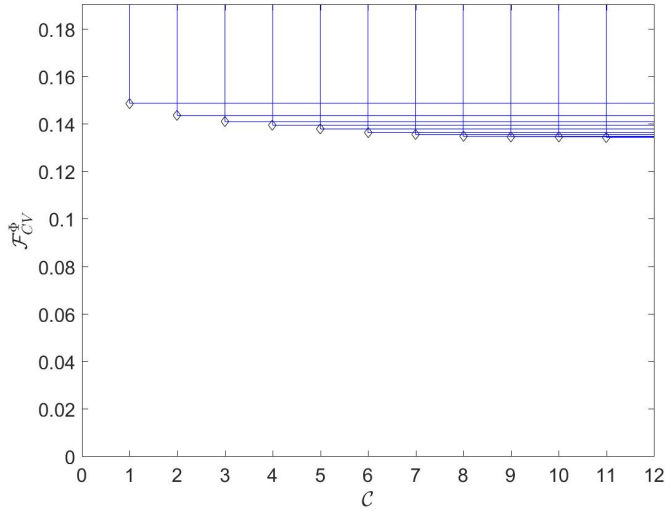

Figure 5: Pareto front found by *MOES-GP-RMSE* with the transformed database *LL6*, run with seed 1.

tained with our proposal (solution s_1) with the regression model obtained without applying feature selection from *LL6* database (6 lag length transformed database) and the *GaussianProcesses* algorithm. Figures 6 and 7 graphically show the predictions in 1 to 6 steps-ahead in test data for both models (Figure 6 without feature selection, and Figure 7 with feature selection). Tables 9 and 10 show numerical results for predictions in 1 to 6 steps-ahead in test data for both models respectively, including number of evaluated instances, *RMSE*, *MAE* and *MAPE* (*mean absolute percentage error*) for each step-ahead. The following statements can be made:

1. The solution with feature selection dominates the solution without feature selection for the multi-objective optimization problem (2) with $p = 6$.
2. Solution without feature selection obtained *RMSE*= $(0.1733, 0.1752, 0.1740)$ and *MAE*= $(0.1349, 0.1426, 0.1461)$. This means that the feature selection process has reduced the *RMSE* by 23.17%, and the *MAE* has been reduced by 30.36%.
3. In addition to being more accurate, the regression model obtained with feature selection is more robust, since the predictions in 1, 2, 3 and 4 steps-ahead are the same. Note that, in the graph of Figure 7, the prediction lines in 2, 3 and 4 steps-ahead are covered by the prediction line in the 1 step-ahead. However, Table 7 shows different *RMSE*, *MAE* and *MAPE* values for the predictions in 1, 2, 3 and 4 steps-ahead, since, in 2 steps-ahead, one instance is evaluated less than in 1 step-ahead, in 3 steps-ahead, two less instances are evaluated, and in 4 steps-ahead, three less instances are evaluated.

Table 11 shows the selected attributes with *MOES-GP-RMSE* from *LL6* and their ranks and importances. The rank and importance of the attributes has been obtained through a univariate wrapper feature ranking method *GaussianProcesses* regression algorithm and *RMSE* metric have been used to evaluate separately each attribute. For this we have used the *Weka* class *ClassifierAttributeEval* which evaluates attributes using a learning scheme. With the parameter *leaveOneAttributeOut* = *false*, an attribute is evaluated considering its worth in isolation, i.e., a model is built through *GaussianProcesses* with only the attribute to evaluate and the output attribute, and its *RMSE* is calculated. When all

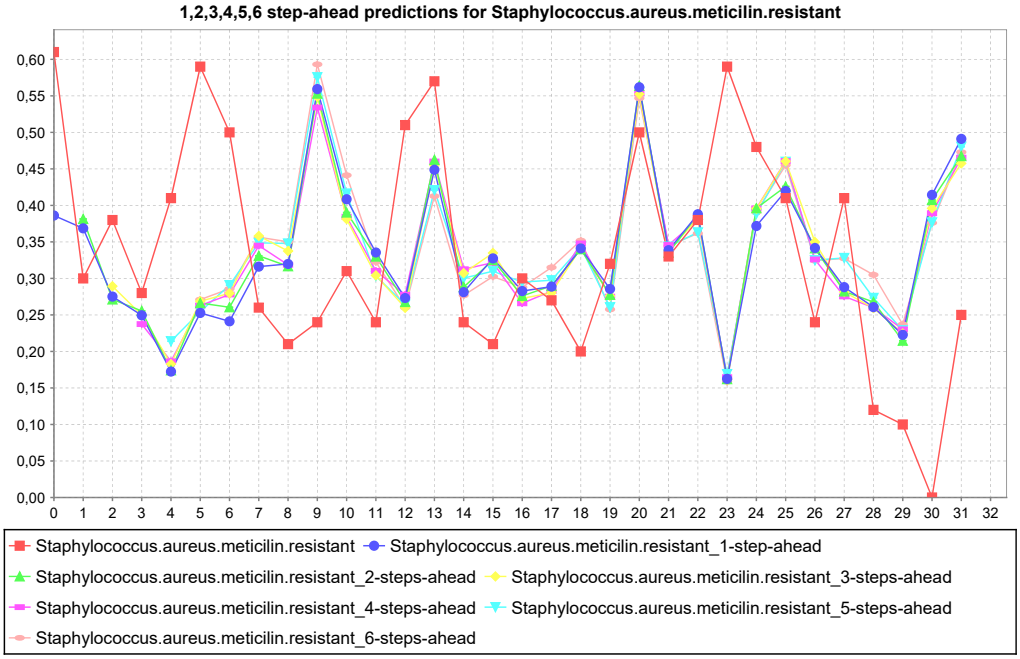

Figure 6: Graphical results of 1 to 6 steps-ahead predictions with *gaussian processes* for Staphylococcus aureus meticilin resistant evaluated on test data with 6 lag length transformed dataset.

Evaluation	***1-step-ahead***	***2-steps-ahead***	***3-steps-ahead***	***4-steps-ahead***	***5-steps-ahead***	***6-steps-ahead***
Test data						
Number of instances	32	31	30	29	28	27
RMSE	0.18	0.1744	0.1753	0.1749	0.1804	0.1834
MAE	0.1386	0.1334	0.1357	0.1358	0.1417	0.1436
MAPE	41.9744	41.4884	42.6367	42.9067	45.2733	47.4543

Table 9: 1 to 6 steps-ahead predictions with *gaussian processes* for Staphylococcus aureus meticilin resistant evaluated on test data with *LL6* transformed dataset.

Evaluation	***1-step-ahead***	***2-steps-ahead***	***3-steps-ahead***	***4-steps-ahead***	***5-steps-ahead***	***6-steps-ahead***
Test data						
Number of instances	32	31	30	29	28	27
RMSE	0.1349	0.1304	0.1325	0.1347	0.1395	0.1421
MAE	0.1003	0.096	0.0987	0.101	0.1094	0.1129
MAPE	28.5268	28.1986	28.9901	29.721	33.3479	36.1422

Table 10: 1 to 6 steps-ahead predictions with *gaussian processes* for Staphylococcus aureus meticilin resistant evaluated on test data with *MOEA-GP-RMSE-LL6* reduced dataset.

the attributes are evaluated, a change of scale is carried out so that the attributes whose resulting *RMSE* is better than the resulting *RMSE* of the model with only the output attribute, are assigned a positive merit (importance), and those attributes with worse resulting *RMSE* that of the model with only the output attribute, are assigned a negative merit.

The reflection on the selected attributes and their ranks shows their suitability to explain the epidemiological relationships between the five time series. We mentioned these relationships when justifying the choice for the study of these five series. The epidemic period of each influenza season is considered to be the period corresponding to the

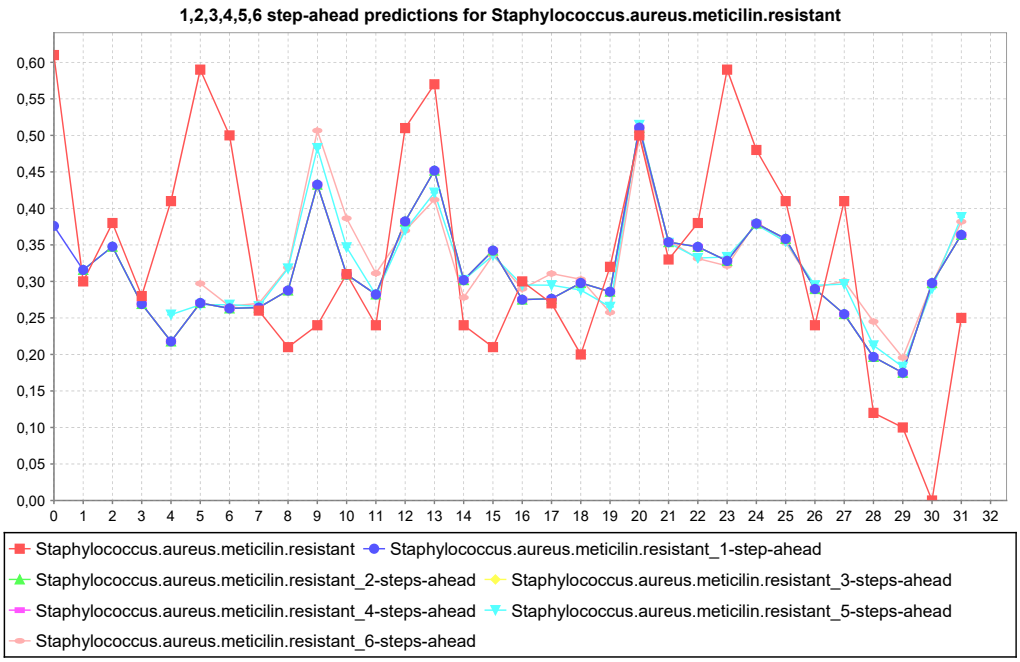

Figure 7: Graphical results of 1 to 6 steps-ahead predictions with *gaussian processes* for Staphylococcus aureus meticilin resistant evaluated on test data with the reduced dataset *MOES-GP-RMSE-LL6* from 6 lag length transformed dataset.

Rank	***Attribute name***	***Importance***
1	Lag_Influenza-4	0.0020182
2	Levofloxacin	0.0014092
3	Lag_Levofloxacin-2	0.0011283
4	Influenza	0.0007227
5	Lag_Staphylococcus.aureus.meticilin.resistant-5	0.0004896
6	Lag_Staphylococcus.aureus.meticilin.resistant-4	0.0002825
7	Lag_Staphylococcus.aureus.meticilin.resistant-6	0.0002069
8	Lag_Staphylococcus.aureus.meticilin.sensible-1	-0.0000752
9	Lag_Levofloxacin-5	-0.0003103
10	Lag_Staphylococcus.aureus.meticilin.sensible-3	-0.0006595
11	Lag_Oseltamivir-1	-0.0011502

Table 11: Selected attributes with *MOES-GP-RMSE* from *LL6*, and their ranks.

epidemiological weeks in which the incidence of influenza exceeds the baseline threshold of activity. The baseline is estimated nationally using the mobile epidemic model [81] and is used to determine the beginning and end of the seasonal influenza wave. Let us think of a theoretical epidemic period in which we place the different attributes provided by the mathematical model in a temporary order. With this temporal arrangement, the Lag Influenza-4 would represent the beginning of the epidemic period. Regarding this beginning, the mathematical model proposes that:

- The incidence of MRSA before the onset of the influenza period (Lag_Staphylococcus.aureus.meticilin.resistant-6 and Lag_Staphylococcus.aureus.meticilin.resistant-5) and at the onset of influenza (Lag_Staphylococcus.-aureus.meticilin.resistant-4) are important attributes for predicting the incidence of MRSA.

- The use of Levofloxacin© is important for predicting the incidence of MRSA prior to and during the influenza period (Lag_Levofloxacin-5). During the whole influenza period its importance in predicting the incidence of MRSA increases (Lag_Levofloxacin-2) and maintains (Levofloxacin).
- SA, coinciding with the increased prescription of Levofloxacin© during the influenza episode, increases its predictive importance in the model (Lag_Staphylococcus.aureus.meticilin.sensible-3 and Lag_Staphylococcus.-aureus.meticilin.sensible-1).
- The Osaltamivir© is of the attributes considered, the least important (Lag_Oseltamivir-1). This proposed mathematical model reading is easily translatable into clinical terminology: when a stable context of Levofloxacin© consumption and MRSA incidence are broken at the expense of an increase in the prescription of Levofloxacin©, an imbalance occurs in the MRSA – SA relationship resulting in an increase in the incidence of MRSA. Seasonal episodes of influenza recreate this situation

From a clinical point of view, our work has evident limitations and shortcomings; for example, the failure to consider the use of other antibiotics or the failure to consider surveillance samples for MRSA. Certainly, the objective was not clinical but the demonstration of the improvement in the prediction capacity provided by the model and together with it, to show the adequacy of the mathematical model proposed to the epidemiological and clinical reality. In particular, the latter has benefited from the choice of *ad hoc* time series.

This last peculiarity of our work, contrary to what one might think at first glance, is one of its contributions. Since it is a mathematical model of clinical observation, on the one hand, it can be easily understood and, on the other hand, it allows the conclusions drawn from the model to be translated directly into concrete clinical actions. Different studies have reported that interventions aimed to reduce the use of antibiotics are related to a reduction in MRSA infection rates [82]. The effectiveness of prevention and hygiene measures are also well-established [83]. The concrete conclusion of our work is the recommendation to reduce the use of levofloxacin and, by extension, all fluoroquinolone antibiotics, particularly in seasonal peaks of use. Our data support the observation that the temporary increase in the use of a fluoroquinolone drug (in our case, Levofloxacin) precedes a temporary increase in MRSA and this association is reproduced by suggesting a causal relationship. The impact of this observation is greater in the context of outpatient setting, as the incidence of community-acquired MRSA is progressively increasing over the last century [84, 85], coexisting with a high intake of fluoroquinolone antibiotics, despite warnings about the adverse effects of these antibiotics and consequent recommendation to restrict their prescription [86]. Our study adds, to the argument based on the adverse effects of fluoroquinolone antibiotics, their direct implication in the temporary increase in the incidence of MRSA due to seasonal overuse of fluoroquinolone antibiotics. The significance of this contribution can be deduced from the figures that delimit the problem. In the late 1960s, MRSA was considered endemic in hospitals, but it appeared quickly and unexpectedly in communities in the 1990s and now prevails throughout the world [87, 88]. The mortality rate associated with invasive MRSA for hospital-onset cases is approximately 29% and 19% for community-onset cases [84].

5. Conclusions

In this paper, we proposed a methodology for antibiotic-resistant forecasting based on feature selection with lagged variables via database transformation. Different wrapper feature selection methods with multi-objective evolutionary search strategy have been used to identify the most relevant lagged variables. Gaussian processes together with the *RMSE* metric on a database with 6 lagged variables have produced the best solution on a total of 206 different forecasting models. In order to choose the best forecasting model, in this paper we have proposed a multiple criteria decision-making process, under which a multi-objective problem is optimized where the *RMSE* and *MAE* metrics in different steps-ahead predictions are defined as problem objectives. The robustness of the forecasting models along the steps-ahead predictions is also taken into account. The results show that the forecasting model obtained by feature selection improves by 23.17% and by 30.36% the *RMSE* and *MAE* respectively of the forecasting model without applying feature selection, as well as its robustness in the 1, 2 and 3 steps-ahead predictions.

In our opinion, the methodology developed has two fields of application in clinical epidemiology. One, to make predictions that optimise surveillance systems and, therefore, prevention. Two, as a knowledge acquisition tool for the

interpretation of the complex relationships between time series, particularly those related to the monitoring of antibiotic consumption and the incidence of micro-organisms and their resistances. The proposed mathematical model can provide more objectivity and quantification capabilities to the visual analysis of the temporal series carried out by epidemiologist experts. Furthermore, the models and the selected variables, make possible to extract knowledge from the temporal series. Predictions of future infections outbreaks allow the reallocation of resources (scarce and insufficient) to control de infection and avoid its propagation. Finally, in a context with a high probability of an outbreak according to predictions, epidemiological active surveillance techniques could adjust its sensitivity and specificity improving the outbreak early diagnosis.

Among future works, the use of information regarding doses is to be approached. Other open lines are related to the automation of the methodology proposed, including an automatic selection of the time series relevant for forecasting. We are currently working on a multi-objective evolutionary algorithm to simultaneously optimize the lag length, select attributes and find the best regression algorithm (ensemble learning) at prediction intervals [89]. Finally, to make possible the integration of the process in clinical practice, providing results in terms of probability and confidence intervals are going to be tackled.

Acknowledgments

This work was partially funded by the SITSUS project (Ref: RTI2018-094832-B-I00), given by the Spanish Ministry of Science, Innovation and Universities (MCIU), the Spanish Agency for Research (AEI) and by the European Fund for Regional Development (FEDER). This work was also partially supported by the Fundación Séneca del Centro de Coordinación de la Investigación de la Región de Murcia under Project 20988/PI/18.

References

[1] ECDC/EMEA Joint Technical Report, The bacterial challenge: time to react, Technical Report EMEA/576176/2009, European Centre for Disease Prevention and Control (2009).
[2] European Commission, A european one health action plan against antimicrobial resistance (amr), Technical report, European Commission (2017).
[3] World Health Organization, Global action plan on antimicrobial resistance, Technical Report WHA68/2015/REC/1, Annex 3, World Health Organization (2015).
[4] S. Y. Tong, J. S. Davis, E. Eichenberger, T. L. Holland, V. G. Fowler, Staphylococcus aureus infections: epidemiology, pathophysiology, clinical manifestations, and management, Clinical microbiology reviews 28 (3) (2015) 603–661.
[5] R. J. Rubin, C. A. Harrington, A. Ponn, K. Dietrich, J. A. Greene, A. Moiduddin, The economic impact of staphylococcus aureus infection in new york city hospitals, Energing Infectious Diseases 5 (1) (1999) 9–17.
[6] A. J. Kallen, J. Brunkard, Z. Moore, P. Budge, K. E. Arnold, G. Fosheim, L. Finelli, S. E. Beekmann, P. M. Polgreen, R. Gorwitz, et al., Staphylococcus aureus community-acquired pneumonia during the 2006 to 2007 influenza season, Annals of emergency medicine 53 (3) (2009) 358–365.
[7] C. for Disease Control, P. (CDC, et al., Severe methicillin-resistant staphylococcus aureus community-acquired pneumonia associated with influenza–louisiana and georgia, december 2006-january 2007., MMWR. Morbidity and mortality weekly report 56 (14) (2007) 325.
[8] R. Köck, K. Becker, B. Cookson, J. van Gemert-Pijnen, S. Harbarth, J. Kluytmans, M. Mielke, G. Peters, R. Skov, M. J. Struelens, et al., Systematic literature analysis and review of targeted preventive measures to limit healthcare-associated infections by meticillin-resistant staphylococcus aureus, Eurosurveillance 19 (29).
[9] T. Yamamoto, A. Nishiyama, T. Takano, S. Yabe, W. Higuchi, O. Razvina, D. Shi, Community-acquired methicillin-resistant staphylococcus aureus: community transmission, pathogenesis, and drug resistance, Journal of Infection and Chemotherapy 16 (4) (2010) 225–254.
[10] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (2003) 1157–1182.
[11] R. Adhikari, R. K. Agrawal, An introductory study on time series modeling and forecasting, CoRR abs/1302.6613. arXiv:1302.6613.
[12] I. H. Witten, E. Frank, M. A. Hall, Data Mining: Practical Machine Learning Tools and Techniques (Third Edition), The Morgan Kaufmann Series in Data Management Systems, Morgan Kaufmann, Boston, 2011.
[13] M. Kuhn, The caret Package, CRAN, http://topepo.github.io/caret/index.html (2019).
[14] H. Spliid, Multivariate ARIMA and ARIMA-X Analysis, CRAN, license GPL-2, Version 2.2, RoxygenNote 5.0.1 (2017).
[15] H. Liu, H. Motoda, Feature Selection for Knowledge Discovery and Data Mining, Kluwer Academic Publishers, Norwell, MA, USA, 1998.
[16] R. Kohavi, G. H. John, Wrappers for feature subset selection, Artificial intelligence 97 (1-2) (1997) 273–324.
[17] N. Japkowicz, M. Shah, Evaluating Learning Algorithms: A Classification Perspective, Cambridge University Press, New York, NY, USA, 2011.
[18] A. G. Karegowda, A. S. Manjunath, M. A. Jayaram, Comparative study of attribute selection using gain ratio and correlation based feature selection, International Journal of Information Technology and Knowledge Management 2 (2) (2010) 271–277.
[19] M. A. Hall, Correlation-based feature selection for machine learning, Tech. rep., University of Waikato (1999).
[20] A. Ahmad, L. Dey, A feature selection technique for classificatory analysis, Pattern Recognition Letters 26 (1) (2005) 43–56.

[21] S. Salzberg, C4.5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993, Machine Learning 16 (3) (1994) 235–240.
[22] H. Ishibuchi, Multi-objective pattern and feature selection by a genetic algorithm, in: Proc. of Genetic and Evolutionary Computation Conference (GECCO'2000, Morgan Kaufmann, 2000, pp. 1069–1076.
[23] D. Kimovski, J. Ortega, A. Ortiz, R. Banos, Parallel alternatives for evolutionary multi-objective optimization in unsupervised feature selection, Expert Systems with Applications 42 (9) (2015) 4239 – 4252.
[24] S. Paul, S. Das, Simultaneous feature selection and weighting – an evolutionary multi-objective optimization approach, Pattern Recognition Letters 65 (2015) 51 – 59.
[25] Q. Zhang, H. Li, Moea/d: A multiobjective evolutionary algorithm based on decomposition, IEEE Transactions on evolutionary computation 11 (6) (2007) 712–731.
[26] F. Jiménez, G. Sánchez, J. García, G. Sciavicco, L. Miralles, Multi-objective evolutionary feature selection for online sales forecasting, Neurocomputing 234 (2017) 75–92.
[27] F. Jiménez, H. Pérez-Sánchez, J. Palma, G. Sánchez, C. Martínez, A methodology for evaluating multi-objective evolutionary feature selection for classification in the context of virtual screening, Soft Computing 23 (18) (2019) 8775–8800.
[28] F. Jiménez, C. M. Martínez, E. Marzano, J. T. Palma, G. Sánchez, G. Sciavicco, Multiobjective evolutionary feature selection for fuzzy classification, IEEE Transactions on Fuzzy Systems 27 (2019) 1085–1099.
[29] U. K. Sikdar, A. Ekbal, S. Saha, Mode: multiobjective differential evolution for feature selection and classifier ensemble, Soft Computing 19 (12) (2015) 3529–3549.
[30] S. K. Nayak, P. K. Rout, A. K. Jagadev, T. Swarnkar, Elitism based multi-objective differential evolution for feature selection: A filter approach with an efficient redundancy measure, Journal of King Saud University - Computer and Information Sciences.
[31] U. Mlakar, I. Fister, J. Brest, B. Potocnik, Multi-objective differential evolution for feature selection in facial expression recognition systems, Expert Systems with Applications 89 (2017) 129 – 137.
[32] K. Deb, A. Pratab, S. Agarwal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: Nsga-ii., IEEE Transactions on Evolutionary Computation 6 (2) (2002) 182 – 197.
[33] S. Das, C.-C. Chang, A. K. Das, A. Ghosh, Feature selection based on bi-objective differential evolution, Journal of Computing Science and Engineering 11 (4) (2017) 130–141.
[34] E. Hancer, B. Xue, M. Zhang, Differential evolution for filter feature selection based on information theory and feature ranking, Know.-Based Syst. 140 (C) (2018) 103–119.
[35] E. Hancer, A multi-objective differential evolution feature selection approach with a combined filter criterion, 2018 2nd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT) (2018) 1–8.
[36] E. Hancer, Fuzzy kernel feature selection with multi-objective differential evolution algorithm, Connection Science (2019) 1–19.
[37] A. Asilian Bidgoli, H. Ebrahimpour-Komleh, S. Rahnamayan, A novel multi-objective binary differential evolution algorithm for multi-label feature selection, in: IEEE Congress on Evolutionary Computation (CEC), 2019, pp. 1588–1595.
[38] S. Kukkonen, J. Lampinen, Gde3: The third evolution step of generalized differential evolution, in: IEEE Congress on Evolutionary Computation (CEC), Vol. 1, 2005, pp. 443 – 450 Vol.1.
[39] Y. Zhang, D. wei Gong, X. zhi Gao, T. Tian, X. yan Sun, Binary differential evolution with self-learning for multi-objective feature selection, Information Sciences 507 (2020) 67 – 85.
[40] Y. Zhang, D.-W. Gong, J. Cheng, Multi-objective particle swarm optimization approach for cost-based feature selection in classification, IEEE/ACM transactions on computational biology and bioinformatics / IEEE, ACM 14.
[41] Y. Zhang, S. Cheng, Y. Shi, D. wei Gong, X. Zhao, Cost-sensitive feature selection using two-archive multi-objective artificial bee colony algorithm, Expert Systems with Applications 137 (2019) 46 – 58.
[42] Z. Yong, G. Dun-wei, Z. Wan-qiu, Feature selection of unreliable data using an improved multi-objective pso algorithm, Neurocomputing 171 (2016) 1281 – 1290.
[43] H. Tyralis, G. Papacharalampous, Variable selection in time series forecasting using random forests, Algorithms 10 (4).
[44] S. F. Crone, N. Kourentzes, Feature selection for time series prediction–a combined filter and wrapper approach for neural networks, Neurocomputing 73 (10-12) (2010) 1923–1936.
[45] Y. Sun, J. Li, J. Liu, C. Chow, B. Sun, R. Wang, Using causal discovery for feature selection in multivariate numerical time series, Machine Learning 101 (1-3) (2015) 377–395.
[46] S. Hido, T. Morimura, Temporal feature selection for time-series prediction, in: 2012 21st International Conference on Pattern Recognition (ICPR 2012), IEEE, 2012, pp. 3557–3560.
[47] J.-M. López-Lozano, D. L. Monnet, P. C. Alonso, A. C. Quintero, N. G. Jiménez, A. Y. Muñoz, C. Thomas, A. Beyaert, M. Stevenson, T. V. Riley, Applications of time-series analysis to antibiotic resistance and consumption data, in: Antibiotic Policies, Springer, 2005, pp. 447–463.
[48] J.-M. López-Lozano, D. L. Monnet, A. Yagüe, A. Burgos, N. Gonzalo, P. Campillos, M. Saez, Modelling and forecasting antimicrobial resistance and its dynamic relationship to antimicrobial use: a time series analysis, International Journal of Antimicrobial Agents 14 (1) (2000) 21 – 31.
[49] M. Willmann, M. Marschal, F. Hölzl, K. Schröppel, I. B. Autenrieth, S. Peter, Time series analysis as a tool to predict the impact of antimicrobial restriction in antibiotic stewardship programs using the example of multidrug-resistant pseudomonas aeruginosa, Antimicrobial agents and chemotherapy 57 (4) (2013) 1797–1803.
[50] V. Erdeljić, I. Francetić, Z. bošnjak, A. Budimir, S. Kalenic, L. Bielen, K. makar ausperger, R. Likic, Distributed lags time series analysis versus linear correlation analysis (pearson's r) in identifying the relationship between antipseudomonal antibiotic consumption and the susceptibility of pseudomonas aeruginosa isolates in a single intensive care unit of a tertiary hospital, International journal of antimicrobial agents 37 (2011) 467–71.
[51] K. Faust, L. Lahti, D. Gonze, W. M. de Vos, J. Raes, Metagenomics meets time series analysis: unraveling microbial community dynamics, Current Opinion in Microbiology 25 (2015) 56 – 66, environmental microbiology * Extremophiles.
[52] N. Dalum Hansen, K. Mølbak, I. J. Cox, C. Lioma, Predicting antimicrobial drug consumption using web search data, in: Proceedings of the

2018 International Conference on Digital Health, DH '18, ACM, New York, NY, USA, 2018, pp. 133–142.
[53] E. Tacconelli, G. De Angelis, M. A. Cataldo, E. Pozzi, R. Cauda, Does antibiotic exposure increase the risk of methicillin-resistant staphylococcus aureus (mrsa) isolation? a systematic review and meta-analysis, Journal of antimicrobial chemotherapy 61 (1) (2007) 26–38.
[54] C. A. Muto, J. A. Jernigan, B. E. Ostrowsky, H. M. Richet, W. R. Jarvis, J. M. Boyce, B. M. Farr, Shea guideline for preventing nosocomial transmission of multidrug-resistant strains of staphylococcus aureus and enterococcus, Infection Control & Hospital Epidemiology 24 (5) (2003) 362–386.
[55] D. Low, Reducing antibiotic use in influenza: challenges and rewards, Clinical Microbiology and Infection 14 (4) (2008) 298–306.
[56] C. Meier, P. Napalkov, Y. Wegmüller, T. Jefferson, H. Jick, Population-based study on incidence, risk factors, clinical complications and drug utilisation associated with influenza in the united kingdom, European Journal of Clinical Microbiology and Infectious Diseases 19 (11) (2000) 834–842.
[57] D. Diekema, M. Pfaller, F. Schmitz, J. Smayevsky, J. Bell, R. Jones, M. Beach, S. P. Group, Survey of infections due to staphylococcus species: frequency of occurrence and antimicrobial susceptibility of isolates collected in the united states, canada, latin america, europe, and the western pacific region for the sentry antimicrobial surveillance program, 1997–1999, Clinical Infectious Diseases 32 (Supplement_2) (2001) S114–S132.
[58] E. M. Agency, Tamiflu (oseltamivir), Tech. Rep. EMA/880602/2018, EMEA/H/C/000402, European Medicines Agency (2019 (last update)).
[59] P. M. Polgreen, M. Yang, R. Laxminarayan, J. E. Cavanaugh, Respiratory fluoroquinolone use and influenza, Infection Control & Hospital Epidemiology 32 (7) (2011) 706–709.
[60] F. Jiménez, A. Gómez-Skarmeta, G. Sánchez, K. Deb, An evolutionary algorithm for constrained multi-objective optimization, in: Proceedings of the Evolutionary Computation on 2002. CEC '02. Proceedings of the 2002 Congress, Vol. 2 of CEC '02, IEEE Computer Society, Washington, DC, USA, 2002, pp. 1133–1138.
[61] E. Zitzler, K. Deb, L. Thiele, Comparison of multiobjective evolutionary algorithms: empirical results, Evolutionary Computation 8 (2) (2000) 173 – 195.
[62] E. Zitzler, L. Thiele, M. Laumanns, C. Fonseca, V. Grunert da Fonseca, Performance assessment of multiobjective optimizers: An analysis and review, IEEE Transactions on Evolutionary Computation 7 (2002) 117–132.
[63] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016, http://www.deeplearningbook.org.
[64] C. E. Rasmussen, C. K. I. Williams, Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning), The MIT Press, 2005.
[65] T. Cover, P. Hart, Nearest neighbor pattern classification, IEEE Trans. Inf. Theor. 13 (1) (2006) 21–27.
[66] X. Yan, Linear Regression Analysis: Theory and Computing, World Scientific Publishing Company Pte Limited, 2009.
[67] L. Breiman, Random forests, Machine learning 45 (1) (2001) 5–32.
[68] S. K. Shevade, S. S. Keerthi, C. Bhattacharyya, K. R. Murthy, Improvements to the smo algorithm for svm regression, Trans. Neur. Netw. 11 (5) (2000) 1188–1193.
[69] C. J. Willmott, K. Matsuura, Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance, Climate Research 30 (1) (2005) 79–82, cited By (since 1996)149.
[70] S. Lang, F. Bravo-Marquez, C. Beckham, M. Hall, E. Frank, Wekadeeplearning4j: A deep learning package for weka based on deeplearning4j, Knowledge-Based Systems 178 (2019) 48 – 50.
[71] J. Brownlee, Deep learning for time series forecasting (2018).
URL https://machinelearningmastery.com/deep-learning-for-time-series-forecasting/
[72] G. Rubio, H. Pomares, L. J. Herrera, I. Rojas, Kernel methods applied to time series forecasting, in: F. Sandoval, A. Prieto, J. Cabestany, M. Graña (Eds.), Computational and Ambient Intelligence, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007, pp. 782–789.
[73] M. A. Hall, Correlation-based feature selection for machine learning, Ph.D. thesis, The University of Waikato (1999).
[74] H. Peng, F. Long, C. Ding, Feature selection based on mutual information: Criteria of max-dependency,max-relevance, and min-redundancy, IEEE transactions on pattern analysis and machine intelligence 27 (2005) 1226–38.
[75] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, Gene selection for cancer classification using support vector machines, Machine Learning 46 (2002) 389–422. doi:10.1023/A:1012487302797.
[76] J. Pearl, Heuristics: Intelligent Search Strategies for Computer Problem Solving, Addison-Wesley Longman Publishing Co., Inc., USA, 1984.
[77] B. Pfaff, Var, svar and svec models: Implementation within r package vars, Journal of Statistical Software 27 (4).
[78] H. Lutkepohl, New Introduction to Multiple Time Series Analysis, Springer Publishing Company, Incorporated, 2007.
[79] H. Spliid, A fast estimation method for the vector autoregressive moving average model with exogenous variables, Journal of The American Statistical Association 78 (1983) 843–849.
[80] Y. Collette, P. Siarry, Multiobjective Optimization: Principles and Case Studies, Springer Berlin Heidelberg, 2004.
[81] T. Vega, J. E. Lozano, T. Meerhoff, R. Snacken, J. Mott, R. Ortiz de Lejarazu, B. Nunes, Influenza surveillance in europe: establishing epidemic thresholds by the moving epidemic method, Influenza and other respiratory viruses 7 (4) (2013) 546–558.
[82] T. Lawes, J.-M. Lopez-Lozano, C. A. Nebot, G. Macartney, R. Subbarao-Sharma, C. R. Dare, K. D. Wares, I. M. Gould, Effects of national antibiotic stewardship and infection control strategies on hospital-associated and community-associated meticillin-resistant staphylococcus aureus infections across a region of scotland: a non-linear time-series study, The Lancet Infectious Diseases 15 (12) (2015) 1438–1449.
[83] R. Jain, S. M. Kralovic, M. E. Evans, M. Ambrose, L. A. Simbartl, D. S. Obrosky, M. L. Render, R. W. Freyberg, J. A. Jernigan, R. R. Muder, et al., Veterans affairs initiative to prevent methicillin-resistant staphylococcus aureus infections, New England Journal of Medicine 364 (15) (2011) 1419–1430.
[84] A. P. Kourtis, K. Hatfield, J. Baggs, Y. Mu, I. See, E. Epson, J. Nadle, M. A. Kainer, G. Dumyati, S. Petit, et al., Vital signs: epidemiology and recent trends in methicillin-resistant and in methicillin-susceptible staphylococcus aureus bloodstream infections—united states, Morbidity and Mortality Weekly Report 68 (9) (2019) 214.
[85] S. Pichereau, W. E. Rose, Invasive community-associated mrsa infections: epidemiology and antimicrobial management, Expert opinion on pharmacotherapy 11 (18) (2010) 3009–3025.
[86] U. Food, D. Administration, et al., Fda updates warnings for fluoroquinolone antibiotics. july 2016, Tech. rep., US Food and Drug Adminis-

tration and others (2018).
[87] F. R. DeLeo, H. F. Chambers, Reemergence of antibiotic-resistant staphylococcus aureus in the genomics era, The Journal of clinical investigation 119 (9) (2009) 2464–2474.
[88] L. D. Liebowitz, Mrsa burden and interventions, International journal of antimicrobial agents 34 (2009) S11–S13.
[89] P. Cortez, P. Pereira, R. Mendes, Multi-step time series prediction intervals using neuroevolution, Neural Computing and Applicationsdoi:10.1007/s00521-019-04387-3.

Appendix

Name	*Parameters*
	Weka
MultiObjectiveEvolutionarySearch	-generations 100 -population-size 100 -seed 1 -algorithm 0 -report-frequency 100
Dl4jMlpClassifier	-S 1 -cache-mode MEMORY -early-stopping "weka.dl4j.earlystopping.EarlyStopping -maxEpochsNoImprovement 0 -valPercentage 0.0" -normalization "Standardize training data" -iterator "weka.dl4j.iterators.instance.DefaultInstanceIterator -bs 1" -iteration-listener "weka.dl4j.listener.EpochListener -eval true -n 5" -layer "weka.dl4j.layers.DenseLayer -nOut 100 -activation "weka.dl4j.activations.ActivationReLU" -name "Dense layer"" -layer "weka.dl4j.layers.OutputLayer -lossFn "weka.dl4j.lossLossMSE" -nOut 1 -activation "weka.dl4j.activations.ActivationReLU" -name "Output layer"" -config "weka.dl4j.NeuralNetConfiguration -biasInit 0.0 -biasUpdater "weka.dl4j.updater.Sgd -lr 0.001 -lrSchedule "weka.dl4j.schedules.ConstantSchedule -scheduleType EPOCH"" -dist "weka.dl4j.distribution.Disabled" -dropout "weka.dl4j.dropout.Disabled" -gradientNormalization None -gradNormThreshold 1.0 -l1 NaN -l2 NaN -minimize -algorithm STOCHASTIC_GRADIENT_DESCENT -updater "weka.dl4j.updater.Adam -beta1MeanDecay 0.9 -beta2VarDecay 0.999 -epsilon 1.0E-8 -lr 0.001 -lrSchedule "weka.dl4j.schedules.ConstantSchedule -scheduleType EPOCH"" -weightInit XAVIER -weightNoise "weka.dl4j.weightnoise.Disabled"" -numEpochs 10 -queueSize 0 -zooModel "weka.dl4j.zoo.CustomNet"
GaussianProcesses	-L 1.0 -N 0 -K "weka.classifiers.functions.supportVector.PolyKernel -E 1.0 -C 250007" -S 1
IBk	-K 1 -W 0 -A "weka.core.neighboursearch.LinearNNSearch -A "weka.core.EuclideanDistance -R first-last""
LinearRegression	-S 0 -R 1.0E-8 -num-decimal-places 4
RandomForest	-P 100 -I 100 -num-slots 1 -K 0 -M 1.0 -V 0.001 -S 1
SMOreg	-C 1.0 -N 0 -I "weka.classifiers.functions.supportVector.RegSMOImproved -T 0.001 -V -P 1.0E-12 -L 0.001 -W 1" -K "weka.classifiers.functions.supportVector.PolyKernel -E 1.0 -C 250007"
	R
rfe	functions=caretFuncs, method="cv", number = 5
rfeControl	method="svmRadial", subsets = c(2:length(dataset)), preProcess=c("center","scale")
mRMR.classic	defaults
trainControl (*Random forest*)	method = "cv", number=10, tuneLength=10

Table 12: Parameters of the feature selection methods and related functions.

Name	***Description***
	Weka
weka.classifiers.timeseries.core.TSLagMaker	Class for creating lagged versions of target variable(s) for use in time series forecasting
weka.attributeSelection	Package for feature selection
weka.attributeSelection.AttributeSelection	Class for feature selection
weka.attributeSelection.ASSearch	Abstract class for search strategy
weka.attributeSelection.ASEvaluation	Abstract class for evaluation
weka.attributeSelection.MultiObjectiveEvolutionarySearch	Class for multi-objective evolutionary search strategy, extends *ASSearch*
weka.attributeSelection.WrapperSubsetEval	Class for multivariate wrapper feature selection methods, extends *ASEvaluation*
weka.classifiers.AbstractClassifier	Abstract classifier
weka.classifiers.functions.Dl4jMlpClassifier	A wrapper for *DeepLearning4j* that can be used to train a multi-layer perceptron using that library, extends *weka.classifiers.RandomizableClassifier*
weka.classifiers.functions.GaussianProcesses	Class for using Gaussian processes for regression, extends *weka.classifiers.RandomizableClassifier*
weka.classifiers.lazy.IBk	Class that implements an instance-based learning algorithm, extends *weka.classifiers.Classifier*
weka.classifiers.functions.LinearRegression	Class for using linear regression for prediction, extends *weka.classifiers.AbstractClassifier*
weka.classifiers.trees.RandomForest	Class for constructing a forest of random trees, extends *weka.classifiers.meta.Bagging*
weka.classifiers.functions.SMOreg	Class for using support vector machines for regression, extends *weka.classifiers.AbstractClassifier*
weka.classifiers.timeseries.WekaForecaster	Class that implements time series forecasting using a *Weka* regression scheme
	R
caret	Package (short for Classification And REgression Training) is a set of functions for creating predictive models
rfe	Function of the *caret* package that performs an Recursive Feature Elimination method for feature selection
mRMRe	Package for Parallelized Minimum Redundancy, Maximum Relevance Ensemble Feature Selection
mRMR.classic	Function of the *mRMRe* package that performs an *mRMR* feature selection

Table 13: *Weka* and *R* packages, classes and functions used in this paper.