Improving the forecast demand process for Emergency Medical Services

The case study of INEM

Nika Shahidian

Dissertation to obtain the Master of Science Degree in

Industrial Engineering and Management

Supervisors: Prof. Daniel Rebelo dos Santos
Prof. Ana Paula Ferreira Dias Barbosa Póvoa

Examination Committee

Chairperson: Prof. Tânia Rodrigues Pereira Ramos
Supervisor: Prof. Daniel Rebelo dos Santos
Member of the Committee: Prof. Rui Miguel Carrasqueiro Henriques

December 2021

ABSTRACT

Emergency Medical Services (EMS) are a vital component of pre-hospital medical care. These services are paramount to save lives as they focus on providing quality care and minimizing response times to ensure patient survivability.

The Portuguese EMS system, SIEM, is responsible for providing prompt and adequate medical care for mainland Portugal. This complex system is managed by the National Emergency Medical Institute (INEM), a public entity designated to coordinate SIEM's operations. INEM balances multiple objectives, budget constraints, and uncertainty. Complex planning decisions must be made, which require accurate demand estimates to serve as inputs to make informed decisions.

The method currently adopted by INEM is based on simple averaging techniques, which may lead to inefficient allocation of scarce resources. Recognizing this, this work aims to identify forecasting models exploring spatial-temporal datasets and contextual data to provide reliable information for decision-makers to support their decisions about resource allocation, and to meet the volume of calls answered at dispatch centres and the need for multiple emergency vehicles.

The time-varying Gaussian mixture model is distinguished as one of the methods with good performance for obtaining accurate demand estimates on the operational level. Together with other Machine Learning models, unexplored in the literature for EMS demand forecasting, these predictive models are developed, optimized, and validated. Validation with INEM's data shows that the Gradient Boosting model achieves, on average, 1.14% higher accuracy than the Gaussian mixture model. Furthermore, having surpassed all other models, boosting algorithms prove to be the most promising for similar applications.

Keywords: Emergency medical services; Operational planning; EMS demand forecasting; Ambulance service demand; Gaussian mixture models; Machine learning.

RESUMO

Os serviços de emergência médica (EMS) são uma componente essencial dos cuidados médicos pré-hospitalares. Estes serviços são fulcrais para salvar vidas e focam-se em minimizar tempos de resposta para garantir a sobrevivência do paciente.

O Sistema Integrado de Emergência Médica (SIEM) é responsável por fornecer cuidados médicos para Portugal continental. Este sistema complexo é gerido pelo Instituto Nacional de Emergência Médica (INEM), uma entidade pública designada a coordenar as operações do SIEM. O INEM tem em conta múltiplos objetivos, restrições orçamentais, e fontes de incerteza, sendo necessário tomar decisões de planeamento complexas, que requerem estimativas precisas de procura.

O método atual adotado pelo INEM é baseado em médias simples, podendo levar a alocações ineficientes dos seus recursos limitados. Tendo isto em conta, o objetivo deste trabalho é identificar modelos de previsão explorando dados espácio-temporais para fornecer informações fiáveis para os decisores apoiarem decisões de alocação de recursos, garantir o atendimento do volume de chamadas nos centros de despacho, e planear a necessidade de veículos de emergência médica.

O modelo de mistura Gaussiano apresenta bom desempenho, obtendo estimativas precisas de procura a nível operacional. Juntamente com outros algoritmos de Aprendizado de Máquina, pouco explorados para previsões de EMS, estes modelos de previsão são desenvolvidos, otimizados, e validados. A validação com dados do INEM mostra que o modelo Gradient Boosting obtém, em média, 1,14% maior precisão do que o modelo de mistura Gaussiano. Além disso, tendo superado os outros modelos, os algoritmos boosting provaram ser os mais promissores para aplicações similares.

Palavras-chave: Serviços de emergência médica; Planeamento operacional; Previsão da procura de EMS; Procura de serviços ambulatórios; Modelo de mistura Gaussiano; Aprendizado de máquina.

ACKNOWLEDGEMENTS

I would like to take the opportunity to express my gratitude to all that have accompanied and supported me throughout this journey.

Foremost, my deepest gratitude goes to Paulo Abreu, for his constant support and guidance, without whom this work would have not been possible. Thank you for sharing your invaluable knowledge and for your constant availability. I would also like to thank Professor Daniel Santos and Professor Ana Póvoa for their indispensable insights and advice. I am deeply grateful for all the guidance and recommendations given throughout the development of this dissertation.

I also wish to express my gratitude to my friends at IST, who made my experience much easier and enjoyable. I would like to thank Márcia for being my constant companion, helping me throughout every stage and keeping me motivated. I am deeply grateful for your friendship and unlimited support. I would also like to thank Jessica and Joana for their companionship and joyful memories.

Also, I would like to thank Cassandro for his unconditional support and endless encouragement that helped me achieve this goal.

Last but surely not least, I wish to express my deepest gratitude to my parents for supporting me throughout my life. I also thank my sisters, Anisa for being my role-model and encouraging me throughout all these years, and Sama for her unshakeable belief in me.

Finally, I thank the Portuguese EMS provider (INEM) for providing the data.

This work was supported by the Portuguese National Funding (FCT) within the scope of project DSAIPA/AI/0044/2018.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

AdaBoost	Adaptive Boosting
AEM	*Ambulância de Emergência Médica* (Medical Emergency Ambulance)
ANN	Artificial Neural Network
ARIMA	Autoregressive Integrated Moving Average
Bagging	Bootstrap Aggregating
CA	Correlation Analysis
CODU	*Centro de Orientação de Doentes Urgentes* (Urgent Patient Dispatching Centre)
EED	European Emergency Data
EMS	Emergency Medical Services
Extra Trees	Extremely Randomized Trees
GA	Genetic Algorithm
GPCG	*Gabinete de Planeamento e Controlo de Gestão* (Planning and Management Control Office)
GMM	Gaussian Mixture model
iCARE	Integrated Clinical Ambulance Record
INEM	*Instituto Nacional de Emergência Médica* (National Emergency Medical Institute)
KNN	K-Nearest-Neighbour
LASSO	Least Absolute Shrinkage and Selection Operator
MEM	*Motociclo de Emergência Médica* (Medical Emergency Motorcycle)
ML	Machine Learning
NINEM	*Ambulância Não-INEM* (Non-INEM Ambulance)
PEM	*Ambulância de Postos de Emergência Médica* (Medical Emergency Station Ambulance)
RS	Random Search
RES	*Ambulância de Postos de Reserva* (Reserve Station Ambulance)
SARIMA	Seasonal Autoregressive Integrated Moving Average
SHEM	*Serviço de Helicópteros de Emergência Médica* (Medical Emergency Helicopter System)

SIADEM	*Sistema Integrado de Atendimento e Despacho de Emergência Médica* (Integrated Answering and Dispatching System of Medical Emergency)
SIEM	*Sistema Integrado de Emergência Médica* (Integrated Medical Emergency System)
SIV	*Ambulância de Suporte Imediato de Vida* (Immediate Life Support Ambulance)
SSA	Singular Spectrum Analysis
SVM	Support Vector Machine
TEPH	*Técnico de Emergência Pré-Hospitalar* (Pre-Hospital Emergency Technician)
TETRICOSY	Telephonic Triage and Counselling System
TIP	*Ambulância de Transporte Inter-hospitalar Pediátrico* (Inter-hospital Paediatric Transport Ambulances)
UMIPE	*Unidade Móvel de Intervenção Psicológica de Emergência* (Mobile Unit of Phycological Emergency Intervention)
VMER	*Viatura Médica de Emergência e Reanimação* (Medical Vehicle of Emergency and Resuscitation)

1. Introduction

This chapter aims to introduce the dissertation, motivating the problem and identifying the goals of this research, as well as presenting the research methodology and the structure of the document. Section 1.1 provides the context regarding the problem under study, Section 1.2 presents the objectives of this dissertation, Section 1.3 characterizes the research methodology, and Section 1.4 outlines the structure of the dissertation.

1.1. Problem Context

Emergency medical services (EMS) are complex systems that are designed to provide medical assistance to patients with serious injuries or illnesses and are a vital component of pre-hospital medical care. Such systems play a key role preserving lives, as patients benefit from these services from the moment they make a call to 112, until the moment they receive pre-hospital medical care and are transported to the hospital where they will receive the appropriate treatment according to their needs (Bélanger et al. 2019).

The Portuguese Integrated Emergency Medical System, *Sistema Integrado de Emergência Médica* (SIEM), is managed by the National Emergency Medical Institute, *Instituto Nacional de Emergência Médica* (INEM). Their primary goal is to provide adequate and prompt medical care to patients in mainland Portugal.

INEM is responsible for allocating scarce resources, which implies planning decisions that affect different planning levels. These decisions are, however, supported by intuition or simple averages at INEM due to the lack of decision support tools based on the state-of-the-art (Santos et al. 2019). Nevertheless, these decisions have an impact on the quality of the care that is provided, justifying the use of models addressed in the state-of-the-art to support decision-making.

These decision support tools usually require the volume of call arrivals and transportation needs as input variables. Based on that, improvement in the forecasting process is a basis to ensure efficient planning, and hence contribute to reducing response times.

1.2. Dissertation Goals

The contextualization presented in the previous section is what motivates this study, whose main goal is to apply forecasting techniques to improve the planning of physical and human resources at INEM. In addition, this work contributes to the production of inputs for EMS planning problems that are explored in a collaborative project between the Centre for Management Studies of *Instituto Superior Técnico* (CEG-IST) and INEM. These problems include Staff Scheduling Problem, Ambulance Location Problem, and Base Location Problem.

The developed forecasting models should provide useful information for INEM to improve the management of its resources, as well as be easily integrated with the optimization models that are under development in the collaborative project. The accuracy of these models contributes to matching demand and supply so that resources can be efficiently allocated both in time and location, and consequently response times can be reduced. Secondary goals include characterizing the system and analysing historical data to provide managerial insights to INEM. Once the importance of accurate and reliable predictive models is recognized, the techniques that have been already applied in the literature are reviewed to identify potential gaps and improvement opportunities. In this context, forecasting models are developed and validated with the Portuguese case study, while comparing the performance of different techniques.

1.3. Research Methodology

A research methodology based on five steps is proposed and shown in Figure 1, in order to achieve the objectives defined in the previous section.

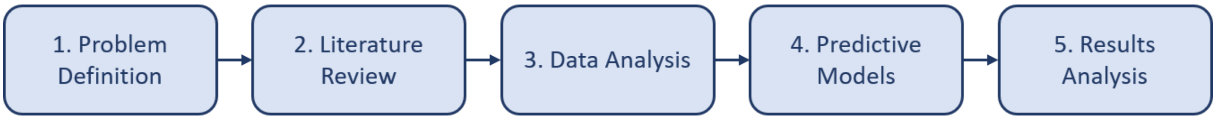


Figure 1 – Steps of the proposed methodology

Step 1 – Problem Definition

The first step consists of characterizing the problem and understanding how it can be solved. Moreover, it includes identifying the objectives of the dissertation and defining the problem that is to be solved. To characterize the problem under analysis, EMS systems are introduced along with their main objectives and features, as well as the related planning problems. The inputs required for decision-making at different planning levels are analysed to identify the need for accurate demand forecasts. These forecasts guarantee efficient planning and quality service, which are monitored by key performance indicators such as response times. This step also includes the introduction of the Portuguese EMS system SIEM, managed by INEM, which shared the data explored in the case study. A description of its features and processes is provided to obtain an overview of the main challenges of demand forecasting and the problem statement.

Step 2 – Literature Review

This step consists of exploring methods to solve the problem defined, and so the state-of-the-art of EMS demand forecasting models is presented. A theoretical context is provided, and the sort of forecasting models already explored in the literature are presented, as well as the attributes that each model considers to explain demand fluctuations. An understanding of how the problem is approached in the literature and what has been done so far allows to recognize improvement possibilities and identify what there is still to do in EMS forecasting. Additionally, methods that have not yet been applied to EMS demand forecasting are presented along with the processes of how to adequately obtain and evaluate the results.

Step 3 – Data Analysis

In this step, the historical data provided by INEM is analysed, and managerial insights are obtained from this analysis. These insights are expected to provide INEM with relevant information to assist in planning and managing operations, as well as obtain further understandings of demand patterns. Additionally, these insights can be eventually useful to the literature in situations where they can be generalized to other EMS systems.

Step 4 – Predictive Models

In this step, different approaches are explored to deal with the problem at hand, and forecasting models are developed and validated using the Portuguese case study. The forecasting process currently in use by INEM uses methods based on simple calculations such as basic averaging techniques. Possible improvement opportunities are identified to reduce response times by developing more accurate forecasting models to assist with EMS planning decisions on the operational level.

Step 5 – Results Analysis

This step consists of analysing the results obtained from the previous step to understand the impact of forecasting errors and the benefit of using the proposed models to support INEM's operations. The proposed models are evaluated in terms of accuracy and efficiency, and a comparative study is conducted to identify the advantages and disadvantages of each method, and produce recommendations.

1.4. Structure of the Dissertation

The remainder of this dissertation is organized into the following chapters:

Chapter 2 – Problem Description

Chapter 2 provides an overview of EMS systems. Decisions on different planning levels are introduced, and the need for accurate EMS demand forecasts is highlighted. The case study addressed in this dissertation is also introduced to understand the organization and identify improvement opportunities. The resources available to INEM are presented, and the call answering and dispatching processes are detailed. Finally, the problem addressed in the dissertation is presented.

Chapter 3 – Literature Review

In Chapter 3, the literature on EMS demand is reviewed, and an overview is given of the prediction methods already applied to the problem at hand. This overview provides sufficient detail to identify the most appropriate method for this application, as well as potential further developments. Other algorithms applied to similar problems are identified and briefly explained to justify their application for predicting the volume of EMS calls and dispatches of emergency vehicles. The procedure that must be followed to apply these algorithms is also characterized to define the computational methodology of this work.

Chapter 4 – Exploratory Data Analysis

Chapter 4 covers an analysis of the data shared by the EMS provider. This data is introduced along with the available variables, which are studied to detect relationships with demand. This is done through the use of data visualization, which is explored to detect patterns and understand demand behaviour. The relationship between predictor variables and defined target variables is further studied through an analysis of correlation.

Chapter 5 – Experimental Results

The data is subjected to attribute selection methods to reduce its dimension by eliminating attributes that provide repeated information to the model, and consequently lead to it becoming biased. The re-dimensioned dataset is used to train and validate predictive models, while their respective hyperparameters are tuned. The performance of the obtained models is evaluated through different performance measures, and their results are compared to identify the best model.

Chapter 6 – Decision Support Framework

Chapter 6 presents a decision support framework based on predictive analytics and applicable to EMS demand forecasting. The knowledge gathered from this work is systematized into concrete recommendations to facilitate future applications for similar problems.

Chapter 7 – Conclusions and Future Work

The final chapter presents the conclusions of the dissertation and recommendations for future work. The contribution of this dissertation is summarized along with its associated limitations and future recommendations, including suggestions to overcome these issues.

2. Problem Description

The aim of this chapter is to define the problem and describe the case study addressed in the dissertation. Section 2.1 introduces the purpose of EMS systems and describes relevant features and issues. Section 2.2 presents the Portuguese EMS system, describing its structure and operations. Section 2.3 describes the research problem, and Section 2.4 presents the chapter's conclusions.

2.1. Problem Setting

EMS systems are complex structures built to provide medical assistance as fast as possible to patients with serious injuries or illnesses. The benefit provided is a unique and vital component of the health care system, as it provides primary care and serves as a bridge to hospital care. These services are crucial in preserving human lives, reducing mortality rate, and improving social welfare (Aringhieri et al. 2017).

Demand for EMS, which includes both pre-hospital medical care and transportation to a medical facility, is increasing largely due to the ageing and growing population (Reuter-Oppermann et al. 2017). Changes in population health, health system practices, public expectations, and accessibility of ambulance services are other factors contributing to rising demand (Lowthian et al. 2011).

2.1.1. EMS Performance Measures

EMS are constrained by tight budgets, yet their goal is to provide maximum benefit to citizens. Performance measures are paramount to evaluate the benefit provided, and ensure that budgets and resources are efficiently allocated (Reuter-Oppermann et al. 2017).

To identify common performance indicators for European EMS systems, the European Community funded the European Emergency Data (EED) Project. This project identifies five indicators to enable monitoring, evaluation, and comparison of pre-hospital emergency care throughout the European Union: availability, reliable access, demand and workload, rate of critical conditions, and level of care. These indicators also serve as benchmarks for EMS across Europe and beyond (Krafft et al. 2003).

Although a set of indicators is defined, other performance measures are recurrent to evaluate the system's performance, such as time-related measures since the data is relatively easy to collect, understand, and enables the measuring of response times. Response times correspond to the time from when the call is received at dispatch centres until medical assistance arrives at the incident's location (Ingolfsson 2013). Thus, response times are directly related to the measure of coverage, as well as the fraction of calls that can be reached within a target response time (Van Den Berg 2016). However, there is no consensus on the proportion of incidents that must be reached within a given response time target. For instance, for urgent urban calls, the UK has a target of 95% of calls to be reached within 19 minutes, while North America has a target of 90% to be reached within 9 minutes (Ingolfsson 2013).

The objectives of minimizing response times and maximizing overall coverage are conflicting with the issue of cost-effectiveness. In order to guarantee a high level of coverage for an area, a large number of costly resources need to be available to offer a proper and fast response to any emergency. Time is a critical factor when it comes to the survivability of the victim, making the balance of cost and coverage challenging, since the value of human life is not estimable and higher costs may be accepted in return for higher protection against losing lives.

Ensuring that all lives are valued equally and providing equal service to all citizens is also a major objective of EMS, and a large concern of the healthcare sector. However, the cost of saving a life in rural areas is much higher than in urban areas. This leads to lives being valued differently depending on the location of the citizen (Aringhieri et al. 2017).

There are two perceptions of equity, horizontal and vertical. Horizontal equity suggests that all demand nodes are to be considered equal and urban areas treated with no discrimination. On the other hand, vertical equity considers different distribution approaches for different areas (Aringhieri et al. 2017). Evaluating satisfaction is also an important aspect when it comes to measuring fairness. Often customers may receive services above a given standard, but feel dissatisfied if they feel that it was worse than that of another customer (Bélanger et al. 2019). These issues are of great concern to the planning and management of EMS systems as they affect how resources are allocated to patients in different areas.

2.1.2. EMS Planning Problems

To minimize response times and ensure fair service and efficient use of resources, EMS require thorough planning. Planning is, however, complex and challenging due to the variability in terms of volume, location, and priority of calls (Ingolfsson 2013).

There are three different planning levels defined, each referring to different planning horizons: strategic, tactical, and operational. On the strategic level decisions are made for several years, on the tactical level decisions refer to periods of one month to one year, and on the operation level decisions are made on a daily basis or in real time (Reuter-Oppermann et al. 2017). At each planning level different planning decisions are made, such as the following (Bélanger et al. 2019):

- Strategic level: location of ambulance stations, fleet dimensioning, staff hiring;
- Tactical level: location of ambulances' standby sites, staff scheduling and crew pairing, fleet management strategies;
- Operational level: ambulance location and relocation, ambulance dispatching, assignment of calls to resources.

Planning decisions on the operational level are particularly significant when aiming to reach given performance levels. Errors at this level, however small, can have severe consequences especially when considering that they may have a direct impact on whether or not a life is saved. These planning decisions require accurate inputs, such as correct demand estimations, to ensure that there are sufficient resources available at the right time and place, so the patient does not need to wait longer than necessary.

2.1.3. Forecasting

Planning problems on all three levels require forecasting as an input. Forecasting can also be divided into the same three levels depending on which decision level it supports (Reuter-Oppermann et al. 2017):

- Strategic forecasting: assists workforce planning and strategic planning of ambulances and stations;
- Tactical forecasts: used for shift scheduling and tactical ambulance planning;
- Operational forecasts: serve as inputs for operational ambulance planning and transport planning.

Operational forecasts must be accurate as they are the input for critical operational planning decisions that if not optimized, may have a direct impact on the survivability of the victim. Simple average-based models may provide satisfactory information for long time horizon decisions, but cannot make sufficiently detailed predictions to support operational planning decisions (Aringhieri et al. 2017). These decisions require accurate estimates of emergency demand and transportation demand, to plan and manage both human and transport resources. Failure in accurately estimating demand can result in inefficient resource allocation, and preventable time inefficiencies. The costs associated with failure in responding to life-threatening incidents in a timely manner are not estimable.

Two areas are distinguished in forecasting: demand forecasting and workload or service time forecasting. While EMS demand prediction has been given more attention, fewer models have been developed that consider how response times and workload are expected to fluctuate over time (Reuter-Oppermann et al. 2017). Service time forecasting is mainly important for strategic planning, as it helps determine the workload on the EMS system. Estimating travel time, which is the largest component of service time, is particularly relevant on the strategic and tactical levels for decisions regarding ambulation location (Reuter-Oppermann et al. 2017).

Accurate predictions of emergency and transportation demand per time period are essential to develop effective EMS strategies. The time period for which the demand is predicted can be hourly or at a certain set granularity. Short-term forecasts provide real time decision support for dynamic ambulance deployment and hourly operational deployment plans, while large time period forecasting is useful for strategic planning and budgeting (Reuter-Oppermann et al. 2017).

2.2. Case Study

The following section presents the case study that is addressed in the dissertation. Section 2.2.1 introduces INEM and its operations, and Section 2.2.2 presents the structure of the organization. Section 2.2.3 presents the four types of resources managed by INEM. Section 2.2.4 describes INEM’s dispatching centres and their integration with other services, and, lastly, the call prioritization system is presented in Section 2.2.5.

2.2.1. SIEM

SIEM serves as an extension of the Emergency Departments of National Health Service hospitals in mainland Portugal. It is responsible for the intervention process from the moment a call is placed until the patient is transferred to an appropriate health unit. Their activities are heavily coordinated and organized in order to maximize the survivability of the victim.

In the literature, these emergency medical activities are distinguished based on two models of EMS systems, the Anglo-American and the Franco-German systems (Reuter-Oppermann et al. 2017). SIEM follows, for the most part, a Franco-German model, where life-threatening emergencies are treated on scene and during transportation by specialized paramedics and physicians. The goal of this model is to provide an initial diagnostic and early treatment to enable the stabilization of the victim. Pre-hospital diagnosis also allows the selection of the most appropriate health unit for the patient. Contrary, in the Anglo-American system, the goal is to respond as fast as possible to calls, and paramedics provide minimal intervention at the scene, mainly transporting victims to a medical facility. While both models share the same general goals of preserving human lives, they have different advantages and drawbacks, resulting in most EMS systems being hybrids that explore both models (Dick 2003).

Driven by the goal of preserving human lives, the operations of SIEM are coordinated by an organism within the Health Ministry, INEM. Founded in 1981, INEM's mission is to define, organize, coordinate, participate in, and evaluate the activities and operations of SIEM, and to provide prompt and error-free health care services in safety and with quality (INEM 2018).

INEM's vision is to be an innovative, sustainable, and motivated organization with a culture of permanent improvement, development of new ideas, processes, competences, and capabilities of professionals, and constant evaluation of results (INEM 2018). Regarding INEM's values, in addition to rigour and responsibility, the following have been set (INEM 2017):

- Quality: assume a compromise with the necessities and expectations of citizens;
- Competence: possess deep and extensive knowledge in the area of medical emergency and its several domains;
- Ethics: act with integrity, patience, and generosity;
- Credibility: receive trust and recognition from society;
- Efficiency: achieve the best possible results with the available resources.

This work aims to contribute in the scope of the values of quality and efficiency, as it aims to assist in improving response times and efficient use of available resources, so that INEM is able to provide quality health care and fulfil its duties suitably. The main responsibilities of INEM include (INEM 2017):

- Ensuring the reception of calls, triage, counselling, and appropriate vehicle dispatching;
- Providing pre-hospital emergency medical care, both medicalized and non-medicalized, and transportation to the adequate health unit;
- Coordinating intervening entities to offer an integrated response;
- Patient transportation, referral, reception, and treatment at the hospital;
- Defining, planning, providing, and certifying emergency medical training of SIEM members;
- Civil planning and awareness-raising actions towards medical emergency;
- Maintaining an emergency medical telecommunication network.

INEM's responsibilities include the coordination and management of SIEM, which is responsible for the intervention process that consists of six stages (INEM 2013):

- Detection: emergency situation detected by civilian and 112 call;
- Alert: screening, triage, priority level assignment, and vehicle dispatch by a Pre-hospital Emergency Technician (TEPH);
- Pre-aid: guidance and assistance to the caller, to perform first-aid basic care if necessary;
- Initial aid in the accident's location: after vehicle arrival, stabilization of the victim and initial treatment;
- Transport and care during transit: transportation to the appropriate health unit and in-transit treatment;
- Transfer and treatment in health unit: transfer of the victim to receiving health unit to finalize treatment.

It is important to note that the last two stages are optional since there may be no need to transport a patient to a health unit. The precedence of the six stages of SIEM is presented in Figure 2, as well as the typical EMS response process. In case no patient transportation is needed, the ambulance becomes idle after departing the scene and traveling to its base.

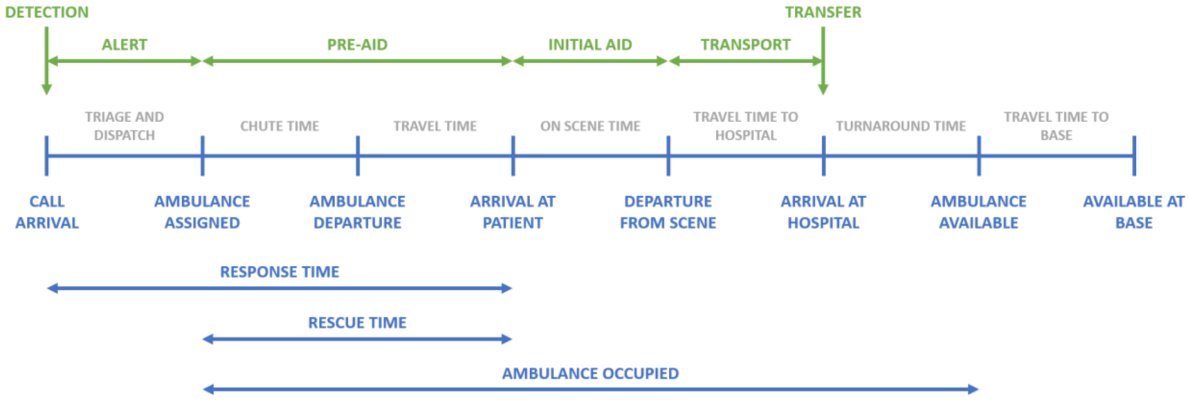


Figure 2 – Response process adapted from Reuter-Oppermann et al. (2017) (blue), and SIEM's six stage intervention (green)

A delay in any of these stages has a direct impact on the overall response time, and consequently on the overall performance of the system. Therefore, these activities must be heavily coordinated as there are different entities involved in delivering emergency medical care. In addition to INEM workers, other entities contribute to the same goal, including firefighters (mostly in rural areas), the Portuguese Red Cross, police officers, doctors and nurses, and hospitals.

2.2.2. Organic Structure

Within INEM's organization, there are three decentralized units, the North, Centre, and South Regional Delegation, as shown in Figure 3. Each of these units is responsible for the operational management of their respective geographical areas, while staying coordinated with the remaining units. These remaining units are again divided into three, the Operational Unit, the Logistics Support Unit, and the Management Support Unit, which all provide services for the entire territory. Each of these units is divided into several departments, each with a different individual focus (INEM 2018).

The Planning and Management Control Office (GPCG), a department inside the Management Support Unit, is responsible for the strategic and operational planning of INEM's activities. This includes studying INEM's activities, and making decisions to optimize processes and improve effectiveness and efficiency (INEM 2018). Demand forecasting is used in numerous planning problems, most of which fall under the GPCG. All the units report to the Executive Board, where the final decisions are made, balancing the multiple objectives of the organization.

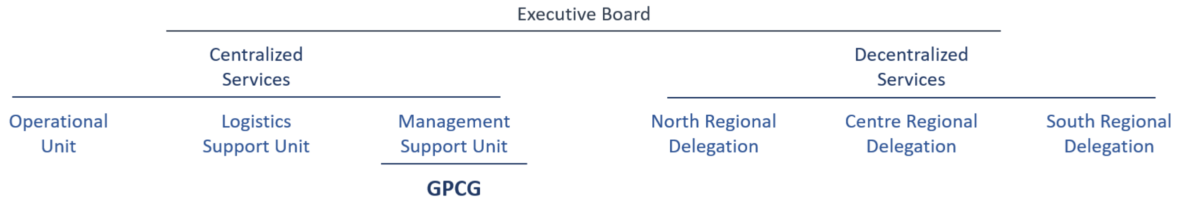


Figure 3 – Organic Structure of INEM

2.2.3. Resources

To achieve the objectives of the organization, INEM has financial, technological, logistic, and human resources (INEM 2018). The largest source of INEM's financial resources is the currently 2.5% applied premiums and contributions relative to health insurance contracts. Other sources include capital provided by European funds, income from financial investments, and training courses (INEM 2017).

Regarding technological resources, four main information systems support INEM's operational activities. The Call Management System registers all the information from received calls, and the Telephonic Triage and Counselling System (TETRICOSY) triage system allows for systematic call evaluation. The electronic registration system iCARE (Integrated Clinical Ambulance Record) improves articulation between the various health care entities, and the Integrated Answering and Dispatching System of Medical Emergency (SIADEM) system classifies and identifies the geographic location of calls (INEM 2017).

Dealing with the unavailability of crew and vehicles, and the management of these resources on a daily basis are some of the main EMS planning problems on the operational planning level (Reuter-Oppermann et al. 2017). Hence, logistics and human resources are the most impacted resources by operational forecasting.

The logistic resources include all emergency medical vehicles at INEM's disposal. In 2018, the INEM fleet was made up of 658 emergency medical vehicles distributed throughout mainland Portugal, and an additional 62 seasonal reinforcement vehicles (INEM 2018).

INEM manages multiple types of emergency medical vehicles, including Medical Emergency Ambulances (AEM), Medical Emergency Motorcycles (MEM), Medical Emergency Station Ambulances (PEM), and Reserve Station Ambulances (RES), which are considered basic life support ambulances that respond to non-critical situations. PEM and RES ambulances are based in and maintained by a civil protection entity that is a partner of SIEM. The PEM ambulances are owned and equipped by INEM, while RES vehicles are owned by the partner and simply financed by INEM (INEM 2019).

Medical Emergency Helicopter Services (SHEM) and Vehicles of Medical Emergency and Reanimation (VMER) are considered advanced life support ambulances as they are staffed by trained doctors and nurses, and have advanced medical equipment (INEM 2019).

Immediate Life Support Vehicles (SIV), Mobile Units of Psychological Emergency Intervention (UMIPE), and Inter-Hospital Paediatric Transport Ambulances (TIP) are essential in special situations that require a differentiated vehicle. SIV ambulances have immediate life support equipment and provide differentiated care like defibrillation. UMIPE intervenes in psychological and psychiatric emergencies, and TIP ambulances provide care to premature and newly born babies in grave condition, and also to underage children (INEM 2019).

Finally, Non-INEM Ambulances (NINEM) are firefighter vehicles or Red Cross ambulances that can be serviced to INEM in case of no other available vehicles, or if these vehicles are significantly closer to the incident's location (INEM 2019).

The human resources of INEM are mainly Pre-hospital Emergency Technicians (TEPHs), nurses, and doctors. In addition to performing duties in their designated emergency medical vehicles, they also have an essential role in the Urgent Patient Dispatching Centres (CODUs) (INEM 2018).

2.2.4. Dispatching Centres

CODUs ensure, for mainland Portugal, daily and continuous emergency medical call reception forwarded through the European Emergency Number 112. The calls are answered by TEPHs, who are supported by a team of medical doctors and psychologists. They evaluate, through a system of triage algorithms and in the shortest possible time, the received aid requests to determine the necessary and adequate resources for each case (INEM 2018).

Although CODU is physically decentralized in four cities, Lisbon, Porto, Coimbra, and Faro, the call handling is centralized. Regardless of the location of the caller, the call is answered by the TEPH that has been idle the longest. With the goal of guaranteeing an integrated response, the CODUs provide the following services (INEM 2018):

- Performance of medical counselling in urgent emergencies and, if necessary, guiding the caller in performing basic emergency manoeuvres;
- Transfer of non-urgent calls to other National Health System entities, namely the call centre of the National Health Service (*Linha Saúde 24*);
- Selection and triggering of the appropriate emergency medical means;
- Counselling of in-field teams when necessary, and validation of performance protocols to non-medic personnel;
- Contact with health units to prepare hospital reception and urgent treatment, based on clinical, geographical, and resource criteria of the destination health unit.

In addition to 112 calls, CODUs also answer forwarded calls from the National Health Service line (*Linha Saúde 24*) and inter-hospital emergency transportation requests.

There has been a significant increase in emergency calls placed throughout the years. In 2018, 1,393,594 emergency calls were answered by CODU, representing an increase of 16% since 2013 and 1.9% from 2017. This demand increase is reflected in CODU's activity levels and on other emergency medical resources, as their activity proportionally increases with the calls. This rise in demand has been justified by the ageing population and increase in chronic diseases, in addition to the increased incidence of flu-like activity (INEM 2018).

2.2.5. TETRICOSY Call Triage System

Once a call is answered by CODU, a TEPH begins a triage process which determines the severity of the incident and the appropriate vehicles required to dispatch. Since 2012, this triage process has been done through the use of the technology TETRICOSY, which allows for standardization of procedures and greater efficiency. The model follows an algorithm that allows for more straightforward call answering, less randomness and treatment errors, and a systematic and thorough evaluation of every situation. This resulted in reduced dispatch time through early prioritization of events, and overall shorter response times (INEM 2017).

The software, developed by INEM, offers a set of questions for the TEPH to ask the caller, records the answers given, and employs different triage algorithms according to the type of emergency, suggesting new questions. Based on the information registered by the TEPH in the software, TETRICOSY assigns a priority level for the incident according to the information provided by the call operator (INEM 2018). Although there is a total of nine priority levels (P1-P9), an emphasis is placed on those with the most frequency:

- Priority 1 (P1): critical life-threatening incidents, originating the dispatch of several advanced life support emergency medical vehicles;
- Priority 3 (P3): urgent situation and dispatching of basic life support emergency medical vehicles;
- Priority 5 (P5): non-urgent situation where the triage results in no vehicle dispatching and the call is transferred to the appropriate health support line;
- Other priorities: other situations that require differentiated assistance.

Throughout the years, the percentage of P3 calls has been increasing significantly, and they hold a large majority of the total number of emergency calls (around 70-75%). The second most common medical emergencies are of P1 priority (around 10-15%), followed by P5 calls (around 7-12%) (Santos et al. 2019). The remaining priorities represent less than 13% of the total volume of calls. Further detail regarding these priorities is unavailable due to the lack of information from public sources, and the inability to meet with INEM representatives on account of the SARS-CoV-2 pandemic.

Depending on the priority level assigned to the incident, different vehicles are selected to respond to the call. Table 1 presents the commonly dispatched emergency medical vehicle combinations for priority levels P1 and P3 (Santos et al. 2019).

Table 1 – Vehicle types dispatched based on call priority

Priority level	Dispatched vehicles – possible combinations
Priority 1 – Emergency situations	AEM / PEM / RES / NINEM + VMER / SIV VMER + SIV SHEM (in exceptional situations)
Priority 3 – Urgent situations	AEM / PEM / RES / NINEM / SIV MEM (to determine if the call is valid)

The most commonly dispatched vehicles are AEM, PEM, RES, NINEM, VMER, and SIV. Even so, of these six, only three (AEM, VMER, and SIV) are owned and managed by INEM, while the rest belong to partners or are managed by them.

2.3. Problem Definition

The goal of INEM is to improve their operational planning, which can be done through the use of an effective forecasting tool to provide accurate volumes of emergency calls and transportation needs for multiple periods. Therefore, this dissertation aims to contribute through the development of a forecasting model to provide input for operational problems and support operational planning decisions.

The forecasting model is expected to provide reliable and accurate estimates to assist the Portuguese EMS provider in the planning of resources on the operational level. The resources entailed by this study are TEPHs, the human resources trained and managed by INEM, and three life support vehicles: VMER, AEM, and SIV. These three vehicle types correspond to the most commonly dispatched vehicles owned and managed by INEM, representing 87% of the vehicles maintained by INEM (INEM 2018).

Regarding the forecasting of transportation demand, the predictive model is to be developed considering just the Lisbon area. It is expected that improvements for this densely populated area represent significant performance gains for INEM.

The Call Management System in place by INEM provides call data which includes information on the priority level of the call, the time, location and type of incident, and the type of vehicle that was dispatched and the associated station. The ratio of dispatching relative to the number of answered emergency medical calls is, on average, 93% (INEM 2017).

For the volume of emergency calls, the forecasting focus is on mainland Portugal, given that call screening is centralized. Moreover, the calls considered for this study are those of priority levels P1 and P3, as they represent around 80-90% of the total volume of calls placed (INEM 2018).

In order to tackle the problem of estimating call volumes and emergency vehicle demand, addressed in the literature as Forecasting of EMS Demand and Ambulance Service Demand, this work aims to present a forecasting model validated with real data shared by INEM. Both types of forecasting are directly related to INEM's operational planning problems, which include those mentioned in Section 2.1.2. Currently, their demand forecasts are based on averages from the historical ratio of calls. An updated decision support tool based on state-of-the-art methodologies will certainly contribute to more effective and efficient planning.

2.4. Chapter conclusions

EMS play a vital role in people’s lives as they contribute to reducing mortality and morbidity, and overall improve the population’s well-being. In order to obtain efficient and effective EMS management, and achieve a given set of performance levels, a good forecasting model is needed since planning problems use it as input to support planning decisions.

Historical data from 2017-2018 shared by the Portuguese EMS provider is to be used to develop and validate an effective forecasting tool. INEM operates a complex system that requires medical intervention on multiple stages, coordination with numerous entities, and efficient management of resources. The goal is to provide a helpful tool that can be relied on for decision-making on an operational planning level.

3. Literature Review

This chapter provides a review of the forecasting models already explored to predict EMS demand and characterizes unexplored predictive models that can be used for this problem. Section 3.1 presents a general overview of EMS planning decisions, and their assortment into planning levels is presented in Section 3.2, as well as the corresponding forecasting levels. Section 3.3 provides a chronological review of the forecasting models developed for predicting EMS demand. Forecasting methods not yet explored for EMS demand forecasting are presented in Section 3.4, as well as the general procedure for their application. Section 3.5 summarizes the various attributes explored in the literature to explain demand fluctuations in forecasting models. Finally, Section 3.6 presents the chapter's conclusions.

3.1. EMS Planning

EMS are an essential part of health care as they provide quick and efficient medical treatment and transportation of patients (Steins et al. 2019). The primary goal of EMS is to minimise response times and deliver an early response to emergency calls, while managing operational costs. Several evaluation metrics serve to determine whether or not an EMS system is delivering high quality health care in terms of efficiency, effectiveness, and fairness. The most commonly used are metrics in terms of coverage and response times (Aringhieri et al. 2017).

Major planning problems are associated with the management and allocation of EMS resources that require planning decisions at every stage of the emergency response (Aringhieri et al. 2017; Steins et al. 2019). These resources consist mainly of emergency vehicles and crew (TEPH, nurses, and medical doctors).

To optimize resource management, three main planning problems are defined in the literature: the ambulance location problem, ambulance relocation models, and dispatching and routing policies (Aringhieri et al. 2017). The ambulance location problem is a static strategic problem that requires long-term and mid-term decisions such as the establishment of stations, assignment of EMS vehicles to those stations, and determination of fleet size. Due to the uncertainty of demand and in order to deal with its variations, some ambulance location models also consider the relocation of EMS vehicles (Degel et al. 2015; Nair & Miller-Hooks 2009). In fact, redeployment is a real time decision-making process that consists in repositioning an idle EMS vehicle to support busy vehicles and maximize overall coverage. It captures the dynamic fluctuations in the system, which is also the case of dispatching and routing, two important real time operational problems. Dispatching consists in assigning appropriate and available vehicles to emergencies, while routing concerns with the route that should be followed by the vehicle assigned to reach the patient (Aringhieri et al. 2017).

Nevertheless, decision problems at different levels are also connected to each other, meaning that decisions made on one level affect subsequent levels and their decisions (Aringhieri et al. 2017). These levels are separated according to the time horizon for which decisions are made. Recognizing that, three levels are addressed: strategic, tactical, and operational.

3.2. Planning Levels

The three hierarchical planning decisions depend greatly on one another since the outcome of the decisions and activities of one level are the input for the following one (Guerriero & Guido 2011). The main decision problems for each planning level are summarized in Table 2, along with the time horizon for which the decisions are made.

Table 2 – Planning problems at different levels, adapted from Reuter-Oppermann et al. (2017)

Planning level	Time horizon	Planning problems
Strategic	Yearly or longer	Locating stations Fleet dimensioning Staff hiring
Tactical	Monthly or weekly	Locating ambulances Staff scheduling and crew pairing
Operational	Daily or in real time	Relocating ambulances Assigning resources to calls Ambulance dispatching Patient transport scheduling

These planning decisions can be supported by mathematical models that require data inputs such as estimates of demand, response time, and workload (Ingolfsson 2013). Thus, these inputs must be accurate to ensure efficient planning.

Problems addressed at the strategic level are mainly static long-term resource allocation problems, for several years or decades. Strategic planning is based on historical data and forecasts typically in the planning horizon of one or more years (Guerriero & Guido 2011; Reuter-Oppermann et al. 2017).

The tactical level concerns decisions such as scheduling and dimensioning for periods of one month and up to one year. The ambulance location problem is typically solved simultaneously for the strategic and tactical levels (Reuter-Oppermann et al. 2017). Both historical data and forecasted demand are used as input. Given that these decisions are made for a few months, they must be systematically revised and re-optimized by considering several aspects such as the availability of ambulances and crew, and seasonal events. At this level, some decisions such as hiring and training staff or purchasing more equipment can be occasionally made. These decisions belong to the strategic level, but since they are made for long periods of time, corrections must occasionally be made (Guerriero & Guido 2011).

In addition to responding to emergencies, EMS vehicles also occasionally perform patient transports, typically from one health unit to another. These transports can be scheduled and planned, meaning that the uncertainty inherent to this service is significantly less, and the planning decisions can be made at the tactical level.

On the other hand, operational level concerns with short-term scheduling and detailed planning on a daily, hourly, or real time basis. Decisions on this level include dynamic relocation of ambulances throughout the day, assignment of crew to ambulances, and assignment and deployment of ambulances to calls (Reuter-Oppermann et al. 2017). Recent research has focused on real time decision-making and on the development of dynamic models to address relocation and dispatching decisions (Bélanger et al. 2019).

Accurate demand forecasts are important for EMS planning on all three levels since they serve as inputs for planning models used for decision-making. Forecasting models that explain demand fluctuations for large areas over long periods are useful for planning on the strategic level. On the other hand, real time decision support for dynamic ambulance deployment and operational planning requires short-term forecasts (Reuter-Oppermann et al. 2017).

The accuracy of these forecasts affects all planning decisions since overestimated predictions lead to over-staffing and unnecessarily high costs, while underestimated predictions lead to under-staffing and high response times (Matteson et al. 2011). The consequences associated with miscalculations and non-optimal decisions are not equal for the different levels. Due to the available time, models that develop forecasts to support strategic level decisions can have large computation times and be made to obtain optimal solutions. On the other hand, for operational level problems, non-optimal solutions can be accepted since efficiency is an important factor. The dimension of the problem and the short time interval available to plan and manage resources make the search for efficient solutions the main goal for this level, justifying the recent attention addressed to short-term forecasting (Reuter-Oppermann et al. 2017).

Short-term demand forecasting provides estimates of hourly emergency call volumes for a given area, which represents an important input for operational planning. These estimates are not useful for long-term planning where managers require aggregate forecasts for a long period of time in a given area (Setzler et al. 2009). The territory is typically divided into areas so that coordinates are translated into a given area of a grid, and arrival times are grouped into slots (Nicoletta et al. 2017). The concepts of time and space granularity refer to the time interval and grid area for which the forecasts are produced. Different granularities are useful for different planning horizons, with short-term decisions requiring fine estimates and long-term decisions needing less detailed forecasts for longer periods of time, often using basic averaging estimates at this level. There are, however, areas where data might be too sparse for a given granularity to describe the spatial structures accurately. EMS data can also be sparse for a desired temporal granularity, making it difficult to estimate accurate spatial structures for each time period. Meaningful levels of both temporal and spatial aggregation must be determined to adequately represent forecasts for each planning decision level (Setzler et al. 2009; Zhou et al. 2015).

Accurate forecasts in fine time and space granularity provide information for EMS managers to make supported decisions at the operational level and to assist with critical time-dependent decisions (Chen et al. 2016). In order to obtain these forecasts, data collected from EMS systems are used. These data usually consist of a timestamp, occurrence location, priority level, vehicle(s) dispatched, etc. (Aringhieri et al. 2017). Patterns are identified based on the data, since call volumes vary throughout the months, days of the week, and hours of the day. Even so, several attributes have been explored to explain the demand fluctuations in forecasting models, such as weather conditions, special-events, and celebrations. Detecting demand patterns is an essential aspect of developing accurate forecasts (Ingolfsson 2013). The initial forecasting models developed were very simplistic and had many shortcomings. Basic statistics and simple regression models allowed to estimate daily demand but failed to account for trend data or other causal factors (Vile et al. 2012). A common assumption in planning models, still currently in use, is that call volumes follow a Poisson process, stationary or time-varying (Ingolfsson 2013).

3.3. Explored Forecasting Methods

The literature presents three types of forecasting models to explain EMS demand, regression models, time series, and spatial-temporal models, however, each model is explored with a variety of forecasting techniques (Steins et al. 2019). A review of these models is presented in the following subsections.

3.3.1. Regression Models

Regression models are the first models addressed to forecast EMS demand, however, these models have some drawbacks such as multicollinearity and difficulty in selecting relevant predictors (Steins et al. 2019). Despite this, these models are still explored nowadays due to their simplicity and ease of application. The models presented in this section refer to multiple regression, where a single dependent variable is predicted using multiple independent variables. Nonetheless, multivariate regression presents an interesting approach that pertains to multiple dependent variables. This method is often used with time series, which is the case of Ho Ting Wong & Lin (2020), as presented in the next section.

The first least squares regression forecasting model applied to EMS was developed by Aldrich et al. (1971) using 32 independent variables, 25 of which reflected sociodemographic characteristics. The study indicates that aged people and single men generate more calls than the rest of the population, and the relationship between the dependent and the independent variables is assumed to be linear. Siler (1975) on the other hand, adopts a nonlinear relationship to construct a multiple regression model considering four socioeconomic variables.

Both Kvalseth & Deems (1979) and Kamenetzky et al. (1982) present first-order and second-order regression models to predict ambulance demand. Contrary to Kvalseth & Deems (1979), Kamenetzky et al. (1982) consider employment in the area as an independent variable, since Siler (1975) previously proved it to be an important variable in predicting demand.

The effect of population aging on pre-hospital EMS demand is considered in the regression model developed by McConnel & Wilson (1998). Their research estimates age-associated rates of utilization and examines the differences that underlie age groups (McConnel & Wilson 1998). Svenson (2000) also identifies a dependency of EMS use rates with age, by adopting a Poisson multiple regression analysis. They also identify a correlation between the use of EMS and increasing levels of poverty (Svenson 2000). Wong & Lai (2010) also use a multiple regression model to examine the weather effects on daily demand, by assessing the dependency of selected variables against weather factors. Their model predicts daily ambulance calls regressed on weather factors and by target groups, and concludes that weather factors can reasonably predict demand for older people, more severe patients, hospital admitted cases, and lower-income groups (Wong & Lai 2010). The impact of age on EMS demand rates is also identified by Lowthian et al. (2011). In a study to measure the impact of population growth and ageing in emergency ambulance services, they use log-linear regression to model the effects of gender and age on demand, and compare the performance with a linear regression model. The study identifies increasing and accelerating demand rates in patients over 85 years of age (Lowthian et al. 2011).

Recent regression models have aimed to incorporate the geographical area of EMS calls, in addition to factors such as time. This is the case of Cramer et al. (2012) who use spatial analysis methods to determine areas of high call volume and understand the factors that contribute to these high volumes. They use stepwise regression to determine significant independent variables, and a geographically weighted regression model to develop a spatial analysis (Cramer et al. 2012). Recently, Steins et al. (2019) used a Zero Inflated Poisson regression model to forecast EMS calls. In contrast to previous regression models, time is considered an independent factor, as well as socioeconomic and geographic factors. The model is capable of providing estimates for geographical areas with low call frequency significantly better than previous models (Steins et al. 2019).

3.3.2. Time Series Models

Since the 1980s, time series models such as autoregressive integrated moving average (ARIMA) and Holt-Winters methods have been explored to forecast call volumes, and have been specifically applied to ambulance demand (Vile et al. 2016). Both methods are successful in overcoming many issues in regression techniques such as multicollinearity, autocorrelation, and the difficulty of selecting covariates (Vile et al. 2016).

To predict daily emergency and non-emergency demand, Baker & Fitzpatrick (1986) adopt Winters' exponential smoothing model. They use a multistep approach to determine the optimal parameters of the exponential smoothing model, and goal programming to combine emergency call and routine call forecasts. The model weighs demand by severity and provides a reliable estimate of overall demand. The results show that a multiple-objective approach provides more accurate forecasts than single-objective models (Baker & Fitzpatrick 1986). For short-term planning, however, EMS planners consider it impractical to disaggregate demand by emergency status, meaning that total demand is typically used as a basis for forecasting, and the distinction of forecasts by severity is unnecessary. Therefore, future time series models do not consider this disaggregation.

This is the case of Channouf et al. (2007), who develop time series models to estimate daily and hourly call volumes. They recognize that EMS demand is influenced by when people work, commute, sleep, and celebrate, and attempt to capture these influences in their models. For daily volumes, they model the arrival rates as Gaussian distributions and develop two models, (1) an autoregressive model of data obtained after eliminating trend, seasonality, and special-day effects, and (2) a doubly-season ARIMA model with special-day effects. Their results show that the first model's performance (1) is superior and the doubly-seasonal ARIMA model (2) performs poorly when forecasting more than one week into the future. For hourly call volume rates, another two approaches are considered: (3) a multinomial distribution for calls in each hour, conditional on the total daily call volume, and (4) fitting a time series to the data at the hourly level. Their results show that the conditional distribution approach (3) generally worked better. The results also demonstrate that forecast accuracy can be improved considerably by updating hourly forecasts using call volumes from earlier that day (Channouf et al. 2007).

Similar to the models proposed by Channouf et al. (2007), most models are based on Gaussian linear time series. This approach is often inaccurate when call-arrival rates are low, which is the case of EMS calls at the hourly level. It also conflicts with the standard industry assumption used in operations research methods that call-arrival volumes follow a Poisson distribution (Ingolfsson 2013). Contrary to Channouf et al. (2007), Matteson et al. (2011) assume that the hourly EMS call-arrival volume has a Poisson distribution. Matteson et al. (2011) develop a call-arrival forecasting method by combining integer-valued time series models with a dynamic latent factor structure for the hourly call-arrival rate. The day-of-week and week-of-year effects are included through constraints on the factor loadings, significantly reducing the number of model parameters. In order to capture fine-scale dependencies, an approach at the hourly level rather than at the daily level is used. Their results show a reduced error in hourly call-arrival volume forecasting (Matteson et al. 2011).

There has been a growing interest in applying a non-parametric technique for time series analysis known as the Singular Spectrum Analysis (SSA), due to the promising prediction performance in various fields (Kalantari 2021; Malamiri et al. 2018). A great advantage of SSA is its flexibility since it is not dependent on parametric assumptions like linearity, stationarity, and normality (Al-Azzani et al. 2020). Vile et al. (2012) apply this technique to EMS demand and show that SSA produces superior long-term forecasts and comparable short-term forecasts to well-established methods.

Similar to Channouf et al. (2007), Ho Ting Wong & Lai (2014) also use ARIMA models to forecast daily demand. They successfully develop a 7-day demand forecast system using weather forecast data as a predictor. Comparing the performance of four ARIMA models, they show that by integrating weather factors such as temperature, the accuracy of daily EMS demand forecasts can be improved (Ho Ting Wong & Lai 2014). ARIMA models have been, more recently, modelled to incorporate seasonality, giving rise to SARIMA models. Gijo & Balakrishna (2016) use SARIMA to model the evolution of hourly and daily call volume data to assist manpower and resource planning at dispatch centres.

Recently, Ho Ting Wong & Lin (2020) focus on understanding the effects of weather to help EMS management, like Ho Ting Wong & Lai (2014) and Wong & Lai (2010) had previously done, and examine the relationship between weather and emergency ambulance service demand. They aggregate records in time series data according to patients' characteristics, and then regress on meteorological data through multivariate forward regression. They observe that elderly and critical patients are more sensitive to weather than other patients, and that non-trauma cases are related to weather (Ho Ting Wong & Lin 2020). Also recently, Al-Azzani et al. (2020) compare the performance of four forecasting approaches, ARIMA, Holt-Winters, multiple regression, and SSA, on a selection of planning horizons, weekly, monthly, and 3-monthly. Their results show that ARIMA provides the most accurate forecasts for weekly and monthly predictions, and that long-term demand is best predicted by SSA.

3.3.3. Spatial-temporal Models

Contrary to time series and regression models, spatial-temporal models are capable of exploring both time and location. In addition to staffing and fleet size management, spatial-temporal demand estimates are critical to decisions such as the selection of station locations and for dynamic deployment planning (Zhou et al. 2015). Dynamic deployment decisions, as well as ambulance dispatching and relocating decisions, are made at the operational level and require short-term forecasts and fine estimates. The spatial-temporal models developed in the literature predict demand volumes at fine spatial and temporal granularity using different methods. This is the case of Setzler et al. (2009), who use Artificial Neural Networks (ANN) to develop accurate emergency call forecasts based on both time and location. The model considers four temporal attributes, hour of the day, day of the week, month, and season. Compared to the practice adopted by the local EMS provider (moving average model), the results show that ANN outperform it at low spatial granularity, although with marginal gains, and that both methods produce noisy results for high spatial resolutions. Although ANN is suitable for estimations on very fine scales in time and space, the volume of data necessary to train ANN models can be seen as the main limitation for the emergence of new publications. Additionally, the results showed in Setzler et al. (2009) suggest that the accuracy improvements obtained from ANN models are not sufficient to justify the computational effort and costs associated with the application of this method. Although they have not been applied to EMS demand forecasting, other adaptations of ANN models are worth mentioning as they are considered state-of-the-art forecasting approaches in a wide variety of problems (Kapoor et al. 2020; McDermott & Wikle 2019). Recurrent Neural Networks deal with ordered data and are commonly used in problems involving time sequences of events. The Graph Neural Network proposed by Scarselli et al. (2009) also presents an interesting opportunity for further exploration of ANN models in the EMS domain.

Contrasting Setzler et al. (2009), where the accuracy improvement over industry practice is marginal, the three models presented in Zhou (2016) show significant accuracy improvements over the industry practice. In addition to this, the computational effort associated with these models is significantly lower, allowing them to be used in operational planning. The methods addressed in Zhou (2016) are a time-varying Gaussian Mixture Model (GMM), a spatial-temporal Kernel Density Estimation, and a Kernel Warping method. They each assume that the set of spatial locations in each time period independently follows a non-homogeneous Poisson process, and all three models consider spatial and temporal patterns such as location-specific seasonality and daily and weekly seasonality. For each method 4-8 weeks of historical data is used, and predictions are made for 4 weeks into the future.

For the same set of data, Zhou et al. (2015) and Zhou & Matteson (2015) each present a model to develop spatial and temporal ambulance call forecasts to estimate Toronto’s demand at fine time and location scales. Zhou et al. (2015) apply a time-varying GMM and consider weekly seasonality by constraining time periods with the same week position to have the same mixture weights. Location-specific temporal patterns such as short-terms serial dependence and daily seasonality are considered with varying strengths at different locations. Zhou & Matteson (2015) on the other hand, apply a spatial-temporal Kernel Density Estimation model and use spatial-temporal weight functions to incorporate several temporal and spatial patterns such as location-specific seasonality and short-term serial dependence. In this model, the spatial-temporal weight functions score how helpful each historical demand is to a given predictive task. Both methods show higher statistical predictive accuracy than the current industry practice, with a comparable computational expense. While the Kernel Density Estimation model proves to be fast, accurate, and easy to interpret and use by non-experts, the proposed GMM produces smoother estimates for Toronto data and proves to be a more accurate method for predicting ambulance demand at fine time and location scales.

Although the results of Zhou (2016) showed that the Kernel Density Estimation model was outperformed by GMM, it is still a powerful tool for non-parametric density estimation that has been widely applied to forecast spatial-temporal data (Nakaya & Yano 2010; Z. Zhang et al. 2011). This popular non-parametric procedure is capable of estimating the probability density function of random occurrences from minor assumptions. Harvey & Oryshchenko (2012) generalize this model to estimate time dependent probability density functions or cumulative distribution functions, and propose the Time Depended Kernel Density Estimation. This model assists in capturing the temporal progress of real-life phenomenon, and has been used to estimate probability distributions in a wide variety of applications (Pérez 2012; Wang et al. 2018; Zambom & Dias 2012).

The time-varying GMM presented by Zhou et al. (2015) is compared with a Kernel Warping method applied in Zhou & Matteson (2016) for Melbourne data, which is highly sparse and has complex spatial-temporal patterns. Zhou et al. (2015) fix the mixture component distributions across time to overcome data sparsity within each time period and accurately describe the spatial structure of the data. The complex spatial-temporal dynamics are represented through time-varying mixture weights, which change over time to capture the dynamics in population movements and actions at different locations and times. On the other hand, due to the high sparsity of Melbourne demand, Zhou & Matteson (2016) focus on data sparsity at high resolutions and modelling complex urban spatial domains. The complexity of this model lies in overcoming sparsity through smoothing, while capturing complex spatial-temporal patterns that require fine-resolution modelling. Spatial-temporal characteristics are difficult to detect accurately at high granularities due to data sparsity. The results for Melbourne data show that the time-varying GMM is slightly less adequate than the Kernel Warping approach for modelling highly complex spatial domains and incorporating spatial boundaries. However, the accuracy improvement of the Kernel Warping model is not considered sufficient to justify the increased complexity of this model.

A different approach addressed to spatial-temporal models is the Bayesian approach, which had previously been successfully applied in health care, although Nicoletta et al. (2017) were the first to consider it in predicting future EMS demand. This method allows the combination of available data with prior information, and then have those results be used as prior information once new data is available. This capability is an important feature for health care applications. The model considers the impact of population, area, and type of area on EMS call demand, in addition to the time slot of the day. Nicoletta et al. (2017) show results that validate this model, producing low prediction errors and demonstrating the applicability of the model. The model and the parameters considered are, however, not sufficient to capture the complexity inherent to EMS demand. Large-scale datasets typically exhibit complex spatial-temporal dynamics and sparsity at high resolutions. The methods addressed in Zhou (2016) are significantly more complex and represent a more accurate model of the data. The methods overcome data sparsity and represent complex spatial and temporal patterns through priors and weights. They capture dynamics in the spatial density, and consider weekly and daily seasonality, as well as location-specific serial dependence. In addition to this, the parameters and variables adopted are superior in complexity and relevance.

Important considerations when developing a model are accuracy, computational times, robustness, and accessibility for EMS managers (Zhou 2016). Ambulance demand data in large cities is usually large-scale which can present computation challenges, especially when trying to develop fine-scale predictions. The time-varying GMM proposed by Zhou et al. (2015) provides the best trade-off of accuracy and complexity for modelling EMS demand, and provides fast estimates for operational planning. The model explains ambulance demand continuously on the spatial domain and over discretized 2-hour intervals. The model is flexible, straightforward to implement, and computationally feasible for large-scale datasets. It predicts the operational performance accurately, and is perfectly suitable for estimations on very fine scales in time and space. The results obtained in Zhou et al. (2015) show that the model reduces prediction error in measuring EMS operational performance by two-thirds compared to the industry practice.

3.4. Unexplored Forecasting Methods

In addition to the ANN explored by Setzler et al. (2009) and the GMM explored by Zhou et al. (2015), other ML algorithms can be applied to the problem under analysis. ML algorithms have been used in the literature for diverse applications (Erickson et al. 2017; Yildiz et al. 2017), yet have not been directly applied to the problem of predicting EMS demand. The goal of this section is to present several ML algorithms and identify their strengths. Such algorithms include those that explore ensemble methods due to their capability to achieve good results for a variety of problems, that can be modelled using the concepts of regression or classification (Maxwell et al. 2018; Thanh Noi & Kappas 2017; Voyant et al. 2017; Were et al. 2015).

3.4.1. Machine Learning Algorithms

Depending on the type of data available and the problem to be addressed, ML algorithms are selected according to four learning approaches: 1) supervised learning; 2) unsupervised learning; 3) semi-unsupervised learning; and 4) reinforcement learning, as shown in Figure 4.

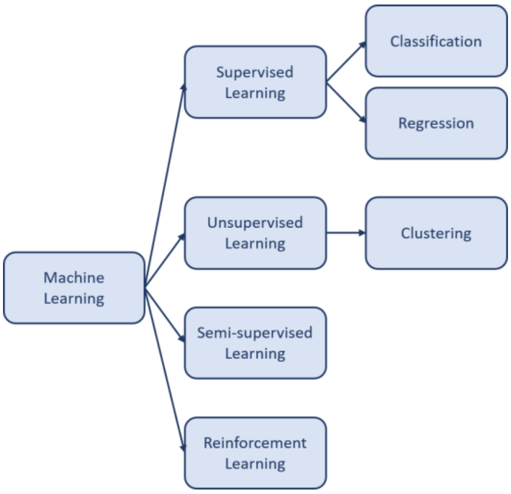


Figure 4 – Taxonomy of ML models adapted from Praveena & Jaiganesh (2017) and Stetco et al. (2019)

The goal of supervised learning is to build a concise model capable of making predictions about future instances that can be either a continuous value (e.g., -1.5, 0.6, 1.2, … , n) or a class (e.g., 0-10 calls, 0-5 dispatches, …, n). To do so, this type of learning requires a training set with both inputs and outputs for each observation of the historical data. While supervised learning aims to capture the relations between the inputs and outputs, unsupervised learning aims to identify patterns in the data and is frequently used for clustering and labelling data. This is particularly useful in situations where the goal is to identify patterns such as consumption profiles (Laspidou et al., 2015; Tureczek et al., 2018). A combined approach is that of semi-supervised learning, which deals with a training set consisting of only some labelled data and mostly unlabelled instances. Often times the data is clustered similarly to unsupervised learning, after which the labels are extended to all observations in the same cluster, completing the dataset and making it suitable for applying supervised algorithms (Ashfaq et al. 2017). On the other hand, reinforcement learning focuses on defining which action should be taken at each given point by identifying which one results in the greatest reward. In order to develop this strategy, the learning system interacts with its environment and receives feedback on the performed actions, meaning that the environment must reward each action either positively or negatively.

Within the supervised learning approach, problems can be modelled as regression or classification. In regression problems the model predicts a continuous value, while in classification problems a label or class is predicted. Typically, regression is used when the output of the problem is in the form of continuous data and classification when the output is in the form of categorical data, however, this is not always the case. Continuous outputs can be easily transformed into classes, though the opposite transformation is not so straightforward (Torgo & Gama 1996). Some ML algorithms are directed either towards regression or classification, though many regression algorithms can be used for classification and vice-versa. This is the case of Logistic Regression, frequently used as a classifier providing a value that indicates the probability of an instance belonging to a given class (Caruana et al. 2015). Additionally, the same algorithm can be explored for both classification and regression in different applications, which is the case of the Random Forest algorithm (Chen et al. 2017; Were et al. 2015).

The following algorithms are mainly used in the context of supervised learning and can be applied for both regression and classification problems. The first two presented algorithms, Naïve Bayes and K-Nearest-Neighbour (KNN), have an explicit underlying probability model that gives a numerical probability as an output. When using these algorithms for classification problems, the model considers the probability of an instance belonging to each class, given the inputs. For example, if a model aims to predict the weather forecast and three classes exist such as rainy, cloudy, and sunny, the model will obtain the respective probabilities of each one occurring and return as output the one with the highest probability.

Naïve Bayes

The Naïve Bayes algorithm is most frequently used as a classifier, where it independently applies the Bayes' theorem for each attribute and a decision rule such as a majority vote is used to attribute a class to an unknown instance. The algorithm assumes that, within a given class, each attribute is independent of any other attribute, i.e., it considers each attribute to contribute independently to the probability of an instance belonging to a class, regardless of any possible correlations between the attributes. For example, given a class winter and attributes temperature, rain, and sunlight, each attribute contributes separately and equally to the probability of it being winter. The algorithm is easy to implement with high efficiency, it is robust to missing values, and requires a short computational time during the training stage. However, the conditional independence assumption is frequently wrong in real-world situations, making Naïve Bayes less accurate than more sophisticated learning algorithms (Bokulich et al. 2018; Clark & Niblett 1989).

K-Nearest-Neighbour

The logic behind the KNN algorithm is that the class or value of an unknown instance is assigned based on its nearest neighbours. As illustrated in Figure 5, the goal is to classify the instance represented by the black point. To do so, the algorithm first calculates the distance between the black point and all the other points, and selects the K shortest distances. For instance, if K=3 the algorithm identifies points p1, p2, and p3 as those with the shortest distances, and assigns to the black point the same class of these three points. If these points do not belong in the same class, the predominant class is selected which in this example is class A (green). Although Figure 5 represents an example for a classification problem, the same logic is applied for regression problems, assigning a continuous value to the black point based on the values of its closest points in the training set. Despite its ease of implementation and good performance in many situations, the algorithm has a large computational cost, and the selection of K has a high influence on the quality of the results (Arunkumar et al. 2017; Cover & Hart 1967).

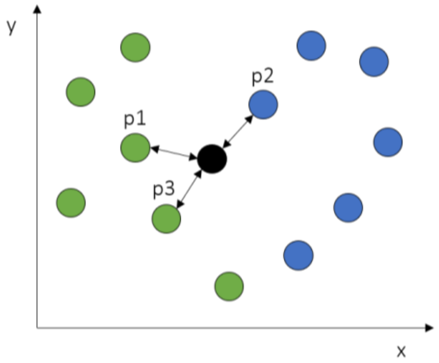


Figure 5 – Visual representation of the KNN algorithm with K=3 and two classes, class A (green) and class B (blue)

Support Vector Machine

The Support Vector Machine (SVM) algorithm focuses on finding the optimal boundary between the training data. The Support Vectors are the coordinates of each training instance, and they are plot in n-dimensional space in order to separate them, where n is the number of features. As a classifier, the algorithm is capable of separating two classes with a single linear boundary, known as a hyperplane, which is placed with the goal of maximizing the distance between the two classes. However, Figure 6 (left) shows a case where a linear hyperplane is not capable of isolating the classes. In these situations, the data is transformed to a higher dimensional space, represented in Figure 6 (centre), aiming to separate the data there. The projection of the feature space to a higher dimensionality is known as the Kernel trick, and the transformed feature space in which the training set is classified is defined by a

Kernel function. The Kernel trick allows separating two classes with a single linear hyperplane when this isn't possible in the original dimensionality of the data. The result is demonstrated in Figure 6 (right).

Although the algorithm can only separate two classes, in multi-class problems this issue is resolved by repeatedly applying the classifier to each combination of classes, making the computational time exponentially greater with the increase in the number of classes. For regression problems the algorithm considers a decision boundary that is set at a given distance from the hyperplane, so that the data points closest to the hyperplane are included in the decision boundary. Despite being computationally expensive, the SVM algorithm usually offers high accuracy (Boser et al. 1992; Thanh Noi & Kappas 2017).

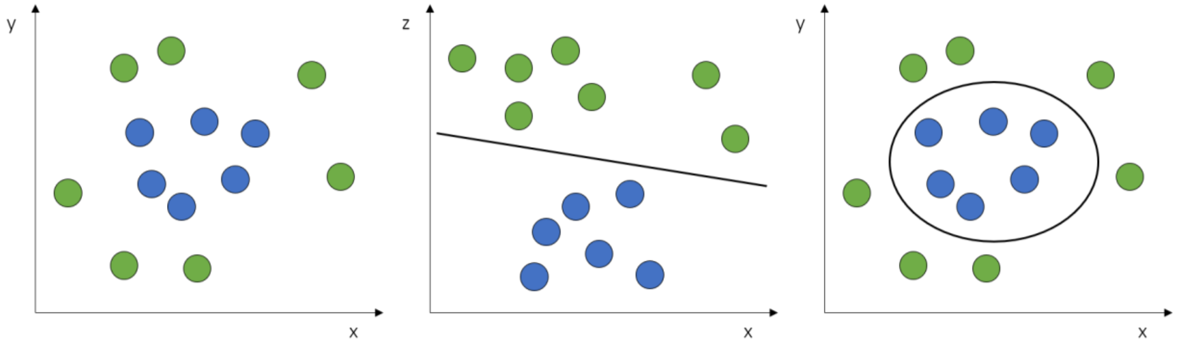


Figure 6 – Visual representation of the SVM Kernel trick

Random Forest

The Random Forest algorithm is an ensemble method that builds multiple decision trees to diversify the use of the training data and build a generalized model. Note that the generalization of a model represents its ability to adapt to unknown data and obtain high predictive accuracy. The logic behind the ensemble method is to build several models and combine those that perform best, resulting in an ensemble that performs better than the best individual model. Two of the most popular ensemble methods are boosting and bagging (bootstrap aggregating). Boosting refers to any ensemble method that combines multiple weak learners into a single strong learner. Unlike strong learners that achieve high accuracy, weak learners perform only slightly better than random guessing. The logic behind boosting is to sequentially train predictors so that each one attempts to correct the previous one. Contrasting boosting where the models are sequentially built, bagging trains the models in parallel on different random subsets of the training dataset. Once the training stage is complete, the prediction for a new instance is made by aggregating the predictions of all the learners in the ensemble. In the case of classification, this aggregation is done by selecting the most frequent prediction, while in the case of regression an average of the predictions is made.

Models based on the Random Forest algorithm are trained via bagging method, meaning that they are repeatedly applied on different random subsets of training data. While the classical Decision Tree algorithm selects the order of the attributes in the tree based on all attributes, Random Forest randomly selects two attributes and decides which is best to continue the tree, and repeats this process until the decision tree is complete. The model then repeats the process on a different subset of data thus creating a different decision tree, as shown in Figure 7. Once all decision trees are completed, the predictions are made by a majority rule in the case of classification and by averaging in the case of regression problems. The algorithm aims to increase generalization accuracy without trading accuracy on training data, and has shown to be computationally faster than other bagging and boosting algorithms. Additionally, it is relatively robust to outliers and noise. The main challenge of this algorithm is selecting the number of decision trees, as insufficient decision trees may result in a low generalization of the model and a large number of decision trees significantly increases the computational time of the algorithm (Breiman 2001; Ho 1995; Ma et al. 2017).

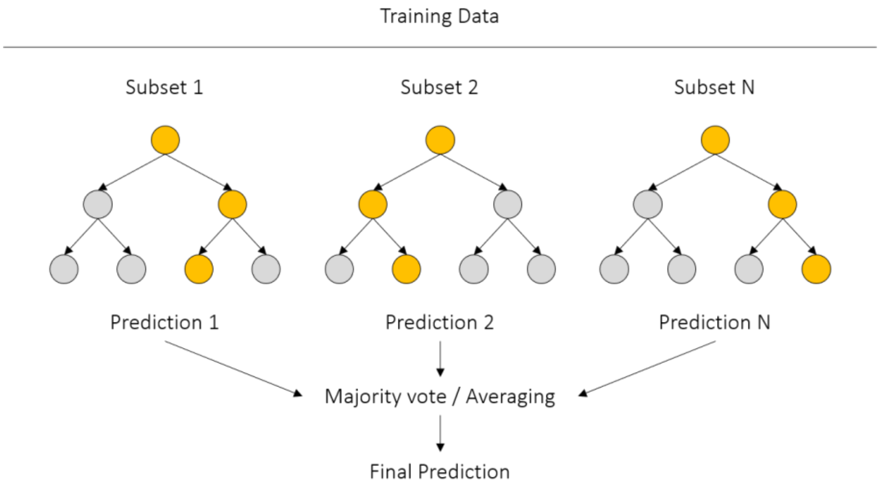


Figure 7 – Visual representation of the Random Forest prediction method with N decision trees

Extremely Randomized Trees

The Extremely Randomized Trees (Extra Trees) algorithm is an ensemble method similar to Random Forest, though it adds additional randomization to each decision tree. There are two main differences compared with the Random Forest algorithm: 1) It does not use a bagging technique to train the model, meaning that the entire training data is used for each decision tree; 2) The selection of the next attribute in the decision tree is done randomly. This allows the algorithm to produce weaker decision trees, making more generalizable models. Other advantages of Extra Trees include its computational efficiency as well as its improved accuracy. However, due to the random selection of attributes, if redundant attributes are present in the dataset they can be included in the model, resulting in a biased model (Geurts et al. 2006; Li et al. 2018).

Gradient Boosting

The Gradient Boosting algorithm is an ensemble method composed of decision trees. It is a boosting method, which means that each predictor is built sequentially based on the previous one, and it uses a gradient descent procedure to minimize loss when adding decision trees. The general purpose of the gradient descent procedure is to minimize a function, so it is used in Gradient Boosting as a way to minimize the loss function. The algorithm starts with prediction 1 (P1) as the average of the target variable for regression problems. It then calculates the loss function for each observation, for example, the mean squared error loss function, and creates a decision tree to predict those errors. At the end of that tree, the second prediction (P2) is the sum of P1 with the error that resulted from the decision tree, although this error is multiplied by a learning rate before being added to P1. The purpose of the learning rate is to reduce the contribution of each decision tree and improve the models' generalization ability. From P2, the algorithm then creates a second decision tree to predict the error of this second prediction and it repeats this process until a stopping criterion is reached. The difference with classification problems is that it uses the probability value of a class occurring as prediction. Gradient Boosting has been shown to provide high predictive accuracy and be robust, making it easily applicable to imperfect data. Additionally, it has the advantage of being flexible as different loss functions can be used. The main limitation of this method is its computational expense since it often requires a large number of trees, making it also memory exhaustive (Breiman 1997; Friedman 1999; Weng et al. 2017).

Adaptive Boosting

Similar to Gradient Boosting, Adaptive Boosting (AdaBoost) is a boosting method that sequentially builds predictors based on decision trees. However, instead of working with full decision trees, it uses decision stumps which only have one node and two leaves, meaning that each decision stump makes predictions based on a single attribute. The algorithm assigns weights to each training observation, representing their importance of being correctly predicted, which is initially equal. A decision stump is created for each attribute, and their predictive accuracy influences the alterations to the observations' weights. High weights are attributed to observations that are incorrectly classified to increase their importance of being correctly classified. The same logic is valid for regression, where a higher prediction error implies a high weight. Additionally, a weight is assigned to each decision stump based on its accuracy, so that once the stopping criterion is reached, the final prediction takes these weights into account. Similar to Gradient Boosting, AdaBoost also uses a learning rate on the weights of the decision stumps to ensure that the model is not overly specific to the training data. In addition to dramatically decreasing generalization error when compared to single tress, AdaBoost is highly accurate when compared with other ensemble methods. Another advantage of this method is that it can be easily used with any base learner. The main disadvantage of AdaBoost is its sensitivity to noisy data and outliers (Freund & Schapire 1997; Hong et al. 2018).

Bootstrap Aggregating

The Bootstrap Aggregating (Bagging) algorithm is an ensemble method that builds decision trees using different subsets from a dataset. Random subsets of data are split to train multiple models, and the multiple versions of the same predictor are used to generate a final prediction. In the case of regression, this procedure is performed by averaging all predictions, while a majority vote is used for classification. The main advantage of Bagging is its ability to build highly generalizable models, however, a critical factor in whether Bagging will improve predictive accuracy when compared to other methods is the instability of the prediction method. If the predictor is sensitive to perturbations in the learning set then Bagging can largely improve accuracy, while in the case of stable procedures it can slightly degrade the performance of the model. Nonetheless, some limitations include its lack of interpretability and relatively high computational expense (Breiman 1996; Rastegari et al. 2019).

3.4.2. Training and Validation in ML

Training and validation are common processes in ML algorithms, in which each one plays an important role in ensuring the construction of a reliable and accurate model. The first process, training, consists of identifying patterns between the inputs and outputs from historical data, where improvements are measured by scoring methods. Although this process is common for all ML algorithms, the way that the model learns depends on the algorithm, since each one follows a different approach or a combination of them, as presented in the previous subsection. The second process, validation, aims to ensure that the developed model is generalizable, i.e., a test set is used to compare the predictions with the real values to check if the model keeps the same efficiency observed during the training process. Additionally, the validation process can be performed using cross-validation, which increases the reliability of the measured efficiency since it is based on resampling procedures (Raschka 2018).

Two problems can arise in the training process, known as overfitting and underfitting, both representing situations where the model is incapable of correctly obtaining the underlying information within the data. Overfitting occurs when the model is not generalizable, meaning that it performs well on the training dataset but not on other sets of data, which is often either because the training set is too noisy or too small. This results in the model detecting patterns that are too specific to the training data or that simply correspond to noise. Contrarily, underfitting occurs when the model is not able to identify the patterns and underlying structure of the data. This may be either due to the model being too simple to learn the structure of the data, or because the data does not provide enough information to capture the patterns related to predictions that are intended (Géron 2019).

Learning curves are used to identify whether the training model is overfitting or underfitting the data. Indeed, the learning curves plot is a useful tool to analyse the model's performance during the training and validation processes. Two curves are plotted, a learning curve representing the training and another representing the validation, indicating how accurate the model is and if this accuracy can be expected with a different dataset. As explained, there are two main reasons for which underfitting occurs, the first being that the model is too simple to grasp the complexity of the dataset, and the second that the training data is insufficient and further training is necessary. Each of these occurrences results in a different learning plot, with the first being depicted in Figure 8 (a), and the second in Figure 8 (b). In the first, both the training and the validation curves reach a point where they remain flat, meaning that increasing more instances to the training set does not reduce or increase the error, and a larger training set or additional attributes are required. In Figure 8 (b), the training and validation curves continue to decrease until the end of the plot, indicating that the model is capable of further learning and additional improvements could be obtained if the training set were larger. Contrastingly, when overfitting occurs, the training curve continues to decrease until the end of the plot, as shown in Figure 8 (c), while the validation curve decreases until a certain point after which it begins to increase. Additionally, there is typically a large gap between the two curves, meaning that the model performs considerably better on the training data than on the validation set. Finally, a good fitting learning curve is presented in Figure 8 (d), where both training and validation curves decrease to a point of stability and the gap between the two curves is insignificant (Géron 2019; Xie et al. 2018).

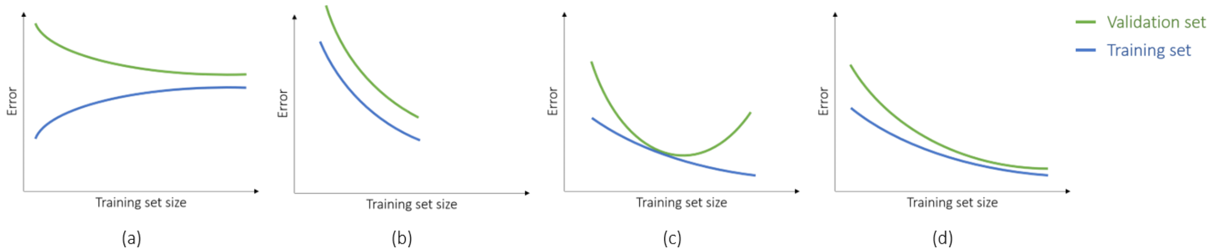


Figure 8 – Reference for performance diagnosis based on learning curves

The validation process measures the generalization error of a model, which refers to the error rate of a model's predictions when it is applied to new data. Two measures commonly used to evaluate this error are bias and variance. Bias is due to incorrect assumptions about the form of the target function and can result in a model underfitting the training data, while variance represents the sensitivity of the model to small variations in the training data. A model with high variance is strongly influenced by the specifics of the training data and is therefore overfitting the training data. There is typically a trade-off between these two errors, known as the bias-variance trade-off: by increasing the complexity of the model the variance increases and the bias decreases, while by reducing the model's complexity the opposite occurs. Additionally, it is important to note that the generalization error of a model is composed of three errors: bias, variance, and irreducible error. This last error represents noise and is difficult to reduce since it cannot be easily identified (Voyant et al. 2017).

To obtain an accurate estimate of the generalization error, the model is repeatedly trained and tested on different sets of data. This process is called cross-validation: the dataset is repeatedly split into different training and validation sets and an accurate performance error is obtained by averaging the evaluations. The main disadvantage of this method is the computational time since it increases with the number of validation sets (Raschka 2018).

3.4.3. Improving the Performance of ML Models

Instead of selecting a ML algorithm and immediately start building a model with the dataset, a series of steps are followed to ensure that a well performing model with the best accuracy is obtained. In this context, the dataset is analysed and prepared by selecting the most relevant attributes. The aim is to remove those which feed redundant information to the model, i.e., highly correlated attributes. This selection also reduces the number of inputs given to the model, which in turn reduces its computation effort. Once a relevant dataset is obtained, models are built exploring different algorithms, and each model is trained and validated with their respective default hyperparameter values. These default values are those that empirically result in a good fit for various datasets. Finally, the model whose algorithm resulted in the best precision is selected and its hyperparameter values are fine tuned to check if there is a possibility of obtaining any improvement in terms of accuracy.

Attribute Selection

The attribute selection process, as mentioned earlier, aims to select the most relevant attributes and remove those that are redundant or irrelevant to the problem. This process can be done through multiple methods. The goal is to increase the model's performance by eliminating noise in the data, leading to improved model interpretation as well as reducing the risk of overfitting. Additionally, with a reduced subset of attributes, the training and prediction speed is increased, and the models become more practical for planners since fewer inputs are required to obtain predictions. Attribute selection is often used to reduce the effects of the curse of dimensionality, which refers to various problems that arise when applying a model to high dimensional data. By reducing dimensionality, computational effort decreases, and models become simpler and more generalizable. Attribute selection methods for supervised learning can be separated into four groups: 1) filter methods; 2) wrapper methods; 3) embedded methods; and 4) hybrid methods (Jain & Singh 2018; Venkatesh & Anuradha 2019).

The filter methods select attributes regardless of the learning algorithm by exploring statistical measures. Some of the most common filter method techniques include Fisher's score, Pearson's correlation coefficient, and Chi-Square test (Guerra-Manzanares et al. 2019; Verma & Kusiak 2012). Filter methods are computationally fast and are particularly recommended when dealing with high dimensional data. On the other hand, wrapper methods use a heuristic approach to consider possible subsets of attributes, evaluating the performance of the predetermined learning algorithm to determine the most relevant attribute subset. The most common are Sequential Forward Feature Selection, Sequential Backward Feature Elimination, Recursive Feature Elimination, and Genetic Algorithms (GA) (Gauthama Raman et al. 2017; Guerra-Manzanares et al. 2019; Yan & Zhang 2015). Wrapper methods have been shown to achieve higher predictive accuracy than filter methods (Zhang et al. 2014). The main reason for this gain in terms of accuracy is because the wrapper methods use the performance of the learning algorithm to evaluate and determine which attributes to select, and therefore find the features that are best suited to be considered in the model. However, wrapper methods are computationally more expensive than filter methods due to the learning process and cross-validation. Another disadvantage of this method is that it must be re-executed if another learning algorithm is to be used. Additionally, due to its complexity, it is susceptible to lead to an overfitting model when applied to small training datasets.

In embedded methods, the attribute selection process is incorporated in the training of the learning algorithm. For each iteration of the training process, the attributes that contribute the most to the training are extracted, meaning that the search is guided by the learning process. The method typically uses ensemble learning and hybrid learning methods to select the attributes. Some of the most popular algorithms are Least Absolute Shrinkage and Selection Operator (LASSO), and Random Forest (Janet & Kulik 2017). Since this method does not require splitting the dataset into training and validation sets, it makes better usage of the available data and also provides a faster solution. Additionally, due to collective decision-making, its performance is frequently better than both filter and wrapper methods. Compared to wrapper methods, embedded methods are computationally less expensive, and less prone to overfitting. Nonetheless, they are computationally more expensive than filter methods. The main drawback of this method is its complexity as well as the fact that it makes decisions depending on the learning algorithm, making it specific to the learning model. Taking advantage of the benefits of each attribute selection method, hybrid methods combine several approaches. They are mainly used to combine filter and wrapper algorithms to achieve great performance with a low computational expense. Commonly, the first stage consists of applying a filter selection algorithm to remove irrelevant and redundant features, reducing the number of features so that a wrapper method can be applied in the second stage with low computational cost (Bins & Draper 2001; Sinayobye et al. 2019). Hybrid methods tend to achieve higher accuracy when compared to wrapper methods, and similar computational efficiency compared to filter methods, and are particularly advantageous when dealing with high dimensional data.

Although each attribute selection method has its benefits and drawbacks, there is no single method nor algorithm within a given method that performs well for any problem. Therefore, the selection of the best method and algorithm is highly dependent upon the available data and problem under analysis.

Hyperparameter Tuning

The hyperparameter tuning is paramount to ensure that the algorithm is running with the best combination of hyperparameter values which, in turn, produces better accuracy. A model's hyperparameters typically remain constant during training, and directly impact the performance and predictive accuracy. The three most popular methods for hyperparameter tuning are Grid Search, Random Search (RS), and Genetic Algorithm (GA) (Liashchynskyi & Liashchynskyi 2019).

The most traditional method is Grid Search, which makes a complete search based on a set of values for each hyperparameter, i.e., it tests every possible combination of hyperparameter values in a pre-defined set. Thus, the number of trials increases exponentially with the number of values defined for each hyperparameter, making this method suffer from the curse of dimensionality. Additionally, Grid Search results in lower model predictive performance when compared to other methods. Nonetheless, Grid Search is a reliable method in low dimensional search spaces and is fairly simple to implement. A more efficient method is RS that tests random combinations of the hyperparameters to find the best solution. It can be applied to discrete, continuous, or mixed search spaces, and typically outperforms Grid Search by finding equal performing models in less computational time. Some researchers have shown that RS has the same practical advantages as Grid Search, such as conceptual simplicity and ease of implementation, and that it trades a small efficiency reduction in low dimensional search spaces for a large efficiency improvement in high dimensional spaces (Bergstra & Bengio 2012). However, the main drawback of this method is its high variance during computing.

Whilst it has many applications, GA is also explored to find an efficient combination of hyperparameter values. GA is an evolutionary search algorithm that sequentially selects, combines, and varies hyperparameters in a manner that simulates the process of natural selection. It begins by generating a random population of individuals, where each individual represents a possible solution in the search space, consisting of values for each hyperparameter. The selection process is based on the performance of each individual, and individuals are repeatedly combined and altered to produce the next generation. This process is iterated until a stopping criterion is reached. Although this method is computationally expensive due to the need to repeat the process for each ML algorithm, the computation time can be reduced by controlling the number of generations and length of the population. Compared to other methods, GA is expected to yield better performing models, and it executes faster in cases where there is a large number of hyperparameters (Liashchynskyi & Liashchynskyi 2019).

Although it is fairly hard to determine which method is most suitable for a given problem, Grid Search is an adequate option for low dimensional search spaces since it tests all the possible combinations, while RS is more suitable for high dimensional spaces due to its efficiency. Despite its computational complexity, GA is also a good option when searching for an efficient combination of a large number of hyperparameters.

Finally, despite each problem having its specificities, the workflow presented in Figure 9 is identified as a pattern in the development of predictive models based on ML algorithms (Al-Janabi et al. 2017; Rastegari et al. 2019; Stetco et al. 2019; Were et al. 2015). This workflow represents the computational methodology that is to be followed in the remainder of this dissertation.

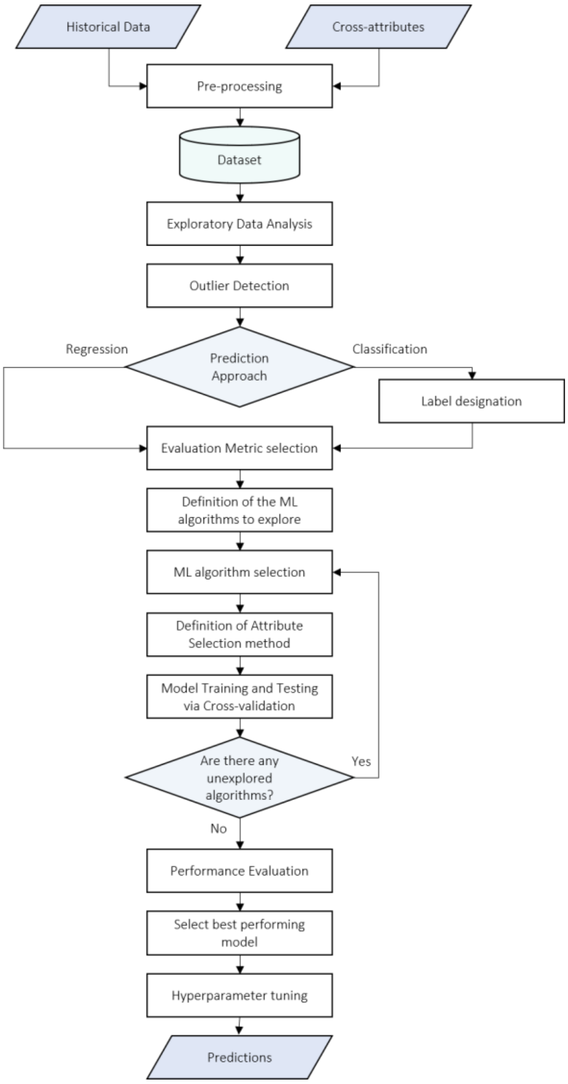

Historical Data
Cross-attributes
Pre-processing
Dataset
Exploratory Data Analysis
Outlier Detection
Regression
Prediction Approach
Classification
Label designation
Evaluation Metric selection
Definition of the ML algorithms to explore
ML algorithm selection
Definition of Attribute Selection method
Model Training and Testing via Cross-validation
Are there any unexplored algorithms?
Yes
No
Performance Evaluation
Select best performing model
Hyperparameter tuning
Predictions

Figure 9 – Flowchart describing workflow of building predictive ML models

3.5. Explored Attributes

The attributes mentioned in the previous subsections are explored in the literature without a consensus since each author explores different types of attributes. These attributes are explored to contextualize occurrences, and to enhance the performance of models by capturing their dependencies, which in turn help explain the fluctuations in demand. Table 3 presents a summary of the most frequently explored attributes that explain demand fluctuations, along with the studies that identify and incorporate those attributes.

Table 3 – Attributes considered to explain demand in forecasting models

Attributes	References
Weather	H. T. Wong & Lai (2010) Ho Ting Wong & Lai (2014) Ho Ting Wong & Lin (2020)
Special-event	Channouf et al. (2007) Ingolfsson (2013)
Resident population	Kamenetzky et al. (1982) Nicoletta et al. (2017)
Age	McConnel & Wilson (1998) Svenson (2000) Lowthian et al. (2011)
Gender	Aldrich et al. (1971) Lowthian et al. (2011)
Employment	Aldrich et al. (1971) Siler (1975) Kamenetzky et al. (1982)
Seasonal patterns	Setzler et al. (2009) Matteson et al. (2011) Gijo & Balakrishna (2016)

The seven attributes presented in Table 3 represent external factors as well as demographic variables that impact EMS demand. Their incorporation in forecasting models results in estimates capable of explaining demand fluctuations throughout time and location, contributing to the production of reliable predictions. It is important, however, not to over-specify the model by including a large number of variables, as it may lead to a computationally complex model. Additionally, some variables may provide redundant information to the model making it biased, or they may not make sense for another system as each system is subject to different conditions. Thus, the selection of these attributes must be explored carefully, which is not observed in the literature since the authors do not justify their choices.

3.6. Chapter conclusions

Research in the area of demand forecasting is vital to develop new methods and decision-making tools that allow improved prediction accuracy and optimized planning decisions. The main planning problems require demand estimates as inputs. Although there are many approaches that produce reasonable predictions, old and inaccurate practices such as simple averaging techniques that produce noisy estimates are still in use by some EMS providers (Setzler et al. 2009; Zhou & Matteson 2015).

Forecasting models are classified into three methods, regression, time series, and spatial-temporal. Few spatial-temporal approaches have been applied thus far, and not many papers have been published that address the spatial component of demand. There is a need, therefore, for future works to focus on these methods and explore further applications of spatial-temporal models on EMS demand forecasting. From the models employed in the literature, GMM is identified as the best trade-off in terms of accuracy, model complexity, and computational speed. Other ML algorithms are presented due to their ease of implementation and high predictive accuracy in similar applications. Their implementation approach is standardized to ensure the obtainment of accurate and generalizable models, including processes such as attribute selection and hyperparameter tuning. Due to their successful implementations in other fields, ML models are expected to achieve high accuracy in predicting EMS demand, and they represent an interesting comparison to models already explored in the literature.

4. Exploratory Data Analysis

This chapter provides an exploratory data analysis of historical data regarding processed call volumes and vehicle dispatches during the temporal interval of 2017-2018. Section 4.1 introduces the datasets that are to be used for the development of predictive models, focusing on the available attributes, while Section 4.2 analyses the target variables with the goal of obtaining insights useful for managerial decisions. Finally, Section 4.3 presents the chapter conclusions.

4.1. Attributes

Historical data shared by INEM referent to years 2017 and 2018, as well as attributes obtained through multiple sources, are presented in two datasets. The data is aggregated in discrete time and spatial intervals. The first dataset contains hourly volumes of calls of priorities P1 and P3 answered per each of the eighteen districts in mainland Portugal. The second dataset has the number of dispatches of vehicles SIV, VMER, and AEM per 8-hour shift from each of the twenty bases in the municipality of Lisbon. Other than the differences in time intervals and spatial origin of demand, the remaining attributes are equal for both datasets. The data has been pre-processed and treated for outliers before this dissertation, so these procedures are not performed.

Since the attributes in the vehicle dataset only have values for the municipality of Lisbon, and the call dataset contains data regarding all districts in mainland Portugal, this second dataset is used for the initial analysis of the attributes. Table 4 presents the attributes available in the datasets grouped by the categories found in the literature and defined in Chapter 3.5, as well as two additional categories: one containing other provided attributes that do not fit into any other category, and another containing historical information obtained from INEM. This information concerns the thirty-seven types of medical occurrences registered by INEM in each given time interval and location. Table 4 also includes the information source of each attribute and the form in which the data is presented. It is important to note that values for attribute *average salary* are only available for 2017, so this attribute is excluded from both datasets on account of half of the information being missing.

Table 4 – Attributes available in datasets

Category	Data type	Source	Attributes
Weather	Continuous	World Weather Online	Maximum temperature (ºC)
			Minimum temperature (ºC)
			Average temperature (ºC)
			Humidity
			Wind speed (Km/h)
Special-event	Binary	-	National holiday
		Footystats	*Primeira Liga* (NOS)
			Taça de Portugal
			UEFA European Championship
			UEFA Champions League

Table 4 – continuation

Category	Data type	Source	Attributes
Resident population	Continuous	National Institute of Statistics	Total deaths
			Total live births
			Total resident population
			Medical doctors per 1000 inhabitants
			Migratory balance
		National Tourism Online	Total tourism guests
Age	Continuous	National Institute of Statistics	Ageing index
Gender	Continuous	National Institute of Statistics	Total female resident population
			Total male resident population
			Total female unemployed
			Total male unemployed
Employment	Continuous	National Institute of Statistics	Average salary
			Average pension value
			Total employed
			Total unemployed
Seasonal patterns	Binary	-	Time interval
			Spatial interval
			Day of the week
			Month
			Season
Accident/crime	Continuous	National Road Safety Authority	Road traffic accident victims
		National Institute of Statistics	Total reported crimes
			Total crimes against people
			Total homicide crimes
Occurrence type	Continuous	Portuguese EMS provider (INEM)	Aviation accident
			Drowning or Diving accident
			Aggression
			Allergies
			Altered state of consciousness
			Psychological support
			Headaches
			Fake call
			Aboard vessels
			Convulsions
			Sick child
			Non-emergency medical transportation
			Sensorimotor deficit
			Diabetes
			Dyspnoea
			Abdominal pain or Bladder weakness
			Back pain
			Chest pain
			General

Table 4 – continuation

Category	Data type	Source	Attributes
Occurrence type	Continuous	Portuguese EMS provider (INEM)	Gynaecology or Pregnancy
			Helicopter transportation
			Bleeding
			Intoxication
			Non-occurrence
			Negligence or Domestic violence or Ill-treatment
			Airway obstruction
			Exceptional occurrences or Nuclear biological chemical
			Eyes or Ears or Nose or Throat
			Other problems
			Cardiac arrest
			Childbirth
			Differentiated support
			Psychiatric problems or Suicide
			Burn injuries or Electrocution
			New-born or Advanced paediatric life support
			Secondary transport
			Trauma

Unlike the majority of the presented attributes where a numerical value is provided, categorical data returns a label, which is the case of attributes such as *day of the week* or *season*. For these situations, instead of a single column representing the attribute and returning a label, each possible label is converted into a column in the dataset. This results in the creation of dummy variables for each label, where the value of 1 is attributed to the column for which the label is true. Dummy variables are used for attributes *time interval* and *district* in the call dataset, as well as *shift* and *nearest base* in the case of the vehicle dataset. Additionally, dummy variables are created for attributes *day of the week*, *month*, and *season*. It is also important to note that attributes of the category *special-event* have binary values depending on whether or not that type of special-event occurred.

An initial analysis of the statistical dispersion of each attribute is used to measure the spread of the data distribution. The goal is to remove features with zero or near-zero deviation, which represents those that are practically constant and do not add information to the model nor improve its performance. For this purpose, the standard deviation of the numerical valued attributes is measured, and attribute *new-born or advanced paediatric life support* is identified as the one having the lowest standard deviation, with a value of 0.0123. However, this value is not considered sufficiently low to justify its removal, and this analysis does not result in the exclusion of any attributes.

The relationship between attributes, whether causal or not, is measured through a coefficient of correlation, which indicates the strength of the statistical association between two attributes. Despite the existence of several correlation coefficients, Pearson's correlation coefficient is selected to evaluate linear relationships due to its wide application in the literature (Liu et al. 2020; Rastegari et al. 2019). A positive correlation indicates that both attributes move in the same direction, where the increase of one is matched by the increase of the other and vice-versa. Contrarily, when negatively correlated, the attributes move in opposite directions. A null correlation represents a lack of relationship between the two attributes. A correlation matrix is used to easily visualize the degree of correlation between any two attributes, and it is shown in Figure 10. As mentioned, the call dataset is used for this analysis because it has the complete information for mainland Portugal, and is a more reliable source for gaining insights due to the higher volume of observations.

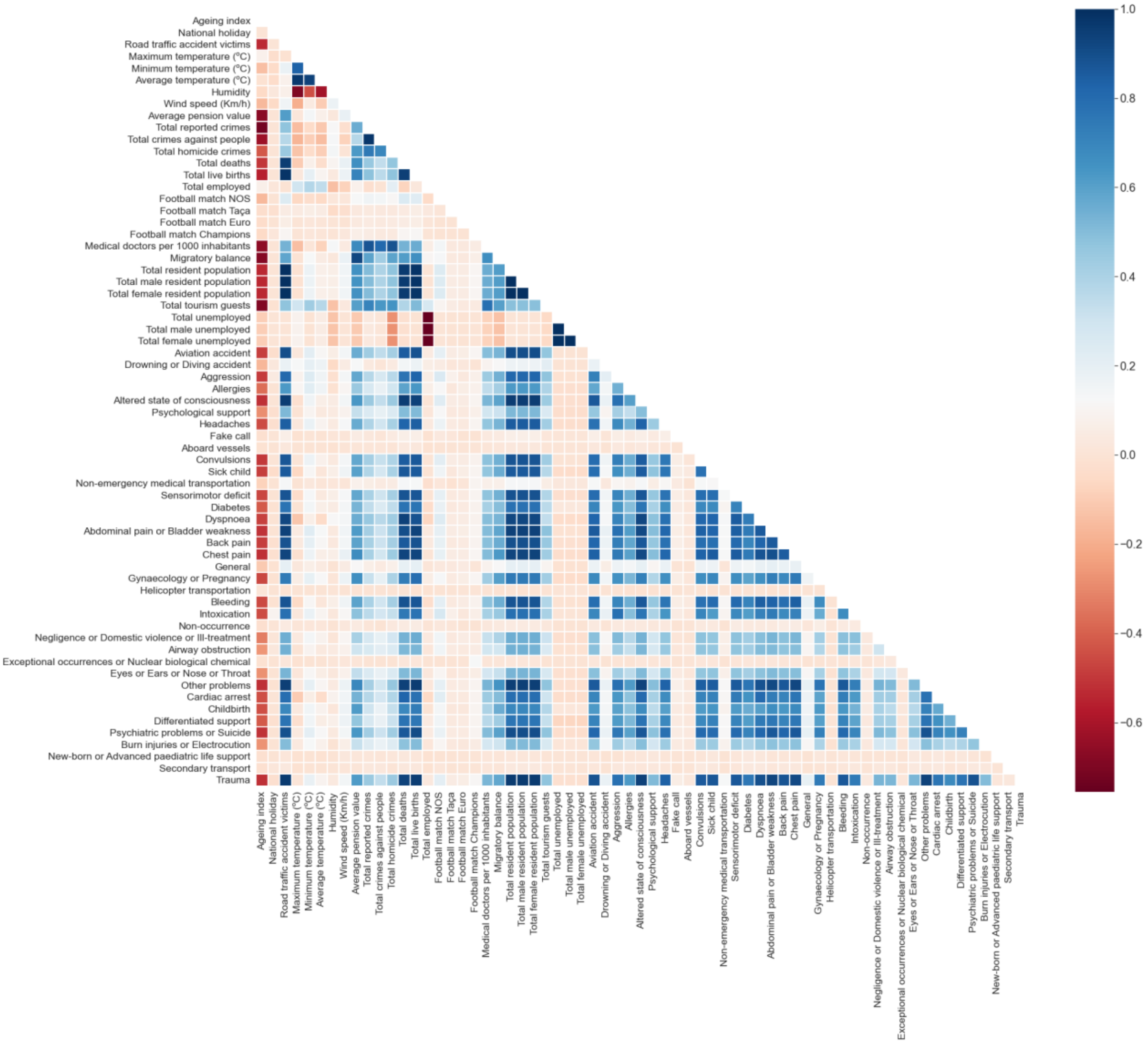


Figure 10 – Correlation matrix of numerical attributes from call dataset

Correlation between attributes within the categories defined in Table 4 is expected to be high as they refer to similar measures. However, this is not always the case, and their relationships are the following:

- *Weather*: temperature attributes are positively correlated with each other, *humidity* is negatively correlated with temperature, and *wind speed* has little to no correlation with the others;
- *Special-event*: little to no correlation is identified between the attributes in this category;

- *Resident population*: positive correlation is identified between the attributes in this category, although attribute *total tourism guests* has lower positive correlation with the others, comparatively;
- *Gender*: the male and female attributes are positively correlated with each other, although unemployment and resident population do not present correlation;
- *Employment*: *total employment* and *total unemployment* are positively correlated, although no correlation is found between this set and attribute *average pension value*;
- *Accident/crime*: *road traffic accident* has low positive correlation with the three crime attributes, which are positively correlated with each other.

Regarding the correlation between categories, categories *accidents/crime*, *age*, *resident population* and attribute *pension value* from category *employment* are highly positively correlated with each other. It is important to note that attributes in category *gender* have a high positive correlation with their respective attribute that is not distinguished by gender. For example, attributes *female resident population* and *male resident population* are highly correlated with *total resident population*, meaning that any attribute with a high correlation with *total resident population* consequently has a similar correlation with the gender equivalents.

The previous correlation depiction does not include category *occurrence type*. This category contains data shared by the Portuguese EMS provider regarding the occurrences that led to call or vehicle needs, justifying the need for a separate and more in-depth analysis. The call dataset, specifically, informs on the number and type of occurrences per hour and per district. Out of the thirty-seven attributes of category *occurrence type*, presented in Table 5, six of them can be considered as rare since they have a daily average rate of lower than 1 occurrence in the entire country: *onboard vessels*, *helicopter transportation*, *non-occurrence*, *exceptional occurrences or nuclear biological chemical*, *new-born or advanced paediatric life support*, and *secondary transport*. In addition to these, four other occurrence types have low incidence rates: *fake call*, *non-emergency medical transportation*, *general*, and *drowning or diving accident*. These ten occurrence types have significantly lower daily rates than the remaining ones, which results in a lack of correlation with other attributes due to their sporadic behaviour. The remaining occurrence types all show similar correlation patterns with the other attributes in the dataset: negative correlation with *age*, no correlation with *special-event*, *weather*, nor *employment*, and positive correlation with *accident/crime* and *resident population*. Additionally, the occurrence types demonstrate positive correlation with each other. Motivated by the goal of understanding the growth or decline rate of each type of occurrence without considering their patterns throughout the year, the relative change value is calculated and presented in Table 5. The relative change refers to the percentage difference from 2017 to 2018 of the mean daily values in the entire country. The majority of the occurrence types suffered a small growth in 2018. The greatest increases were those of *psychological support*, *sick child*, *general*, *gynaecology or pregnancy*, *non-occurrence*, *negligence or domestic violence or ill-treatment*, and *differentiated support*, which had increase rates between 10% and 16%. The highest growth, however, was that of *exceptional occurrences or nuclear biological chemical*, which doubled its occurrences in 2018, although its extremely low incidence rate is noteworthy. Only eight out of the thirty-seven types of occurrences suffered a decline in 2018, the highest being *fake call*, *aboard vessels*, *non-emergency medical transportation*, and *secondary transport*.

Table 5 – Occurrence type yearly change analysis in mean daily values

Occurrence type	**2017**	**2018**	**Absolute change**	**Relative change**
Aviation accident	2647.04	2675.31	28.27	+1.07%
Drowning or Diving accident	32.09	32.55	0.46	+1.43%
Aggression	972.56	1010.37	37.81	+3.89%
Allergies	320.35	343.17	22.82	+7.12%
Altered state of consciousness	10828.67	11436.36	607.69	+5.61%
Psychological support	115.53	131.05	15.52	+13.43%
Headaches	1077.50	1165.41	87.91	+8.16%
Fake call	2.56	2.10	-0.46	**-17.95%**
Aboard vessels	0.13	0.07	-0.07	**-50.00%**
Convulsions	1325.13	1339.40	14.27	+1.08%
Sick child	1605.90	1803.22	197.33	**+12.29%**
Non-emergency medical transportation	8.02	5.52	-2.50	**-31.15%**
Sensorimotor deficit	2035.33	2148.36	113.03	+5.55%
Diabetes	891.09	848.81	-42.28	-4.74%
Dyspnoea	8505.27	9130.59	625.32	+7.35%
Abdominal pain or Bladder weakness	4662.90	4786.78	123.88	+2.66%
Back pain	1812.43	1901.85	89.42	+4.93%
Chest pain	3930.15	4216.96	286.82	+7.30%
General	15.52	17.69	2.17	**+13.98%**
Gynaecology or Pregnancy	411.22	467.90	56.68	**+13.78%**
Helicopter transportation	0.13	0.13	0.00	0.00%
Bleeding	1972.27	2021.39	49.12	+2.49%
Intoxication	2023.30	2017.84	-5.46	-0.27%
Non-occurrence	0.85	0.99	0.13	**+15.38%**
Negligence or Domestic violence or Ill-treatment	116.78	131.44	14.66	**+12.56%**
Airway obstruction	217.97	226.32	8.35	+3.83%
Exceptional occurrences or Nuclear biological chemical	0.07	0.13	0.07	**+100.00%**
Eyes or Ears or Nose or Throat	146.83	158.01	11.18	+7.61%
Other problems	8920.04	9325.28	405.24	+4.54%
Cardiac arrest	1187.38	1279.56	92.19	+7.76%
Childbirth	324.76	301.41	-23.34	-7.19%
Differentiated support	1563.81	1787.18	223.36	**+14.28%**

Table 5 – continuation

Occurrence type	2017	2018	Absolute change	Relative change
Psychiatric problems or Suicide	1583.61	1731.29	147.68	+9.33%
Burn injuries or Electrocution	133.61	128.94	-4.67	-3.49%
New-born or Advanced paediatric life support	0.07	0.07	0.00	0.00%
Secondary transport	0.20	0.13	-0.07	**-33.33%**
Trauma	11785.18	12200.22	415.04	+3.52%

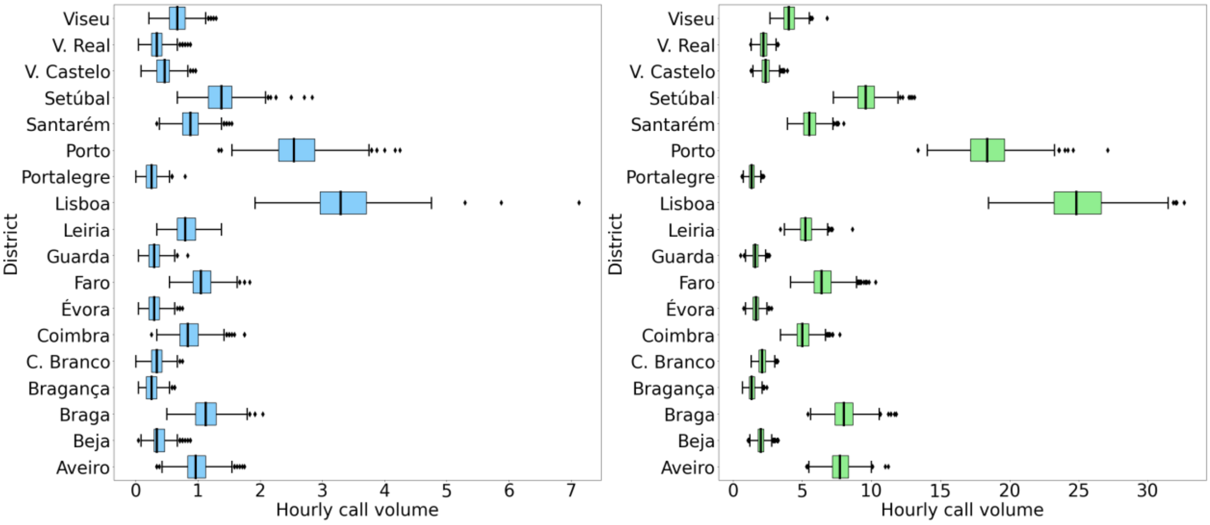
Regarding the concentration of occurrences in different districts, Lisbon and Porto have the highest number of occurrences for the majority of the cases, typically followed by Setúbal. However, the values of the occurrences are deeply related to the *resident population size*, as is visible by the high correlation between them in the correlation matrix of Figure 10. In order to identify behavioural differences in the various districts, values per resident population are used. The hourly mean values of each district show that for most occurrence types, Faro, Beja, and Portalegre are the districts with the highest number of occurrences per resident population. This general conclusion has several exceptions including the previously mentioned rare types of occurrences which do not follow any trend pattern.

4.2. Target Variables

Once the individual attributes have been studied, it is then important to identify the target variables and analyse their relationship with the presented predictive variables. For the problem at hand, there are a total of five target variables: P1 and P3 call volumes which are available in the call dataset; and SIV, VMER, and AEM vehicle dispatches from the vehicle dataset.

Calls of priority P1 are those of emergency life-threatening incidents, while priority P3 is assigned for urgent situations. Different combinations of advanced life support emergency medical vehicles are dispatched for P1 calls depending on the incident, although they typically result in the dispatch of two out of the three vehicles under analysis. On the other hand, P3 calls usually only result in the dispatch of one life support emergency medical vehicle, for example, a AEM or a SIV vehicle. Despite the existence of other priorities, only P1 and P3 calls result in the dispatch of the vehicles SIV, VMER, and AEM. Calls of priority P3 are significantly more frequent than P1, and this gap increased in 2018. Nonetheless, both priority calls had a significant demand growth from 2017 to 2018, with 3.9% and 5.43% increases for P1 and P3 calls, respectively.

Figure 11 shows the number of calls per hour for each district in mainland Portugal, highlighting the differences in demand volumes for each district. The comparison of the graphs also shows the contrast of P3 calls comparatively to P1, with P3 having on average 5.24x higher demand.

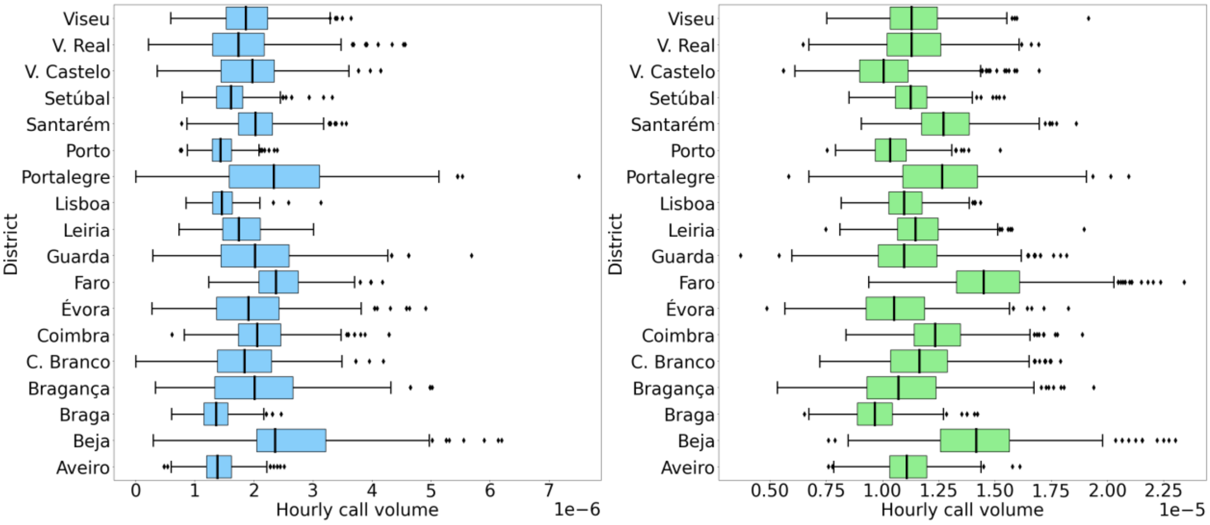


(a) P1 calls

(b) P3 calls

Figure 11 – Hourly priority P1 and P3 call volumes per district

Similar to attributes of *occurrence type*, the values of the call target variables are deeply related to *resident population size*, with the most populated districts (Lisboa, Porto, and Setúbal) having the highest demand volumes. On the other hand, Faro, Beja, and Portalegre present the greatest volumes when analysing demand per resident population, as shown in Figure 12. Nonetheless, extreme variations between districts are not observed in mean values, although some districts present significantly larger standard deviation, as is the case of Portalegre.



(a) P1 calls

(b) P3 calls

Figure 12 – Hourly priority P1 and P3 call volumes per district and per resident population

The three types of vehicles that are target variables in this study represent the three most commonly dispatched vehicles owned and managed by INEM. Each vehicle serves a different purpose: AEM vehicles are basic life support emergency vehicles that require a crew of two TEPH; VMER vehicles are advanced life support vehicles with advanced medical equipment, that require a doctor and a nurse; and SIV vehicles are differentiated ambulances with immediate life support equipment, that require a nurse and a TEPH. AEM vehicles are significantly more issued. In 2017, the proportion of dispatches was the following: 4.77% SIV, 16.54% VMER, and 78.69% AEM. The next year, SIV and VMER vehicles represented a slightly bigger portion of dispatches, leaving AEM vehicles with 77.29% of dispatches. Although it is not significant, this could indicate a trend towards a more balanced distribution.

Dispatches of vehicles SIV, VMER, and AEM in the municipality of Lisbon are spatially aggregated according to the base from which the vehicle left. The location of the twenty origin bases is shown in Figure 13. Additional information including the name of these bases and the ratio of vehicles per base is available in Appendix A.

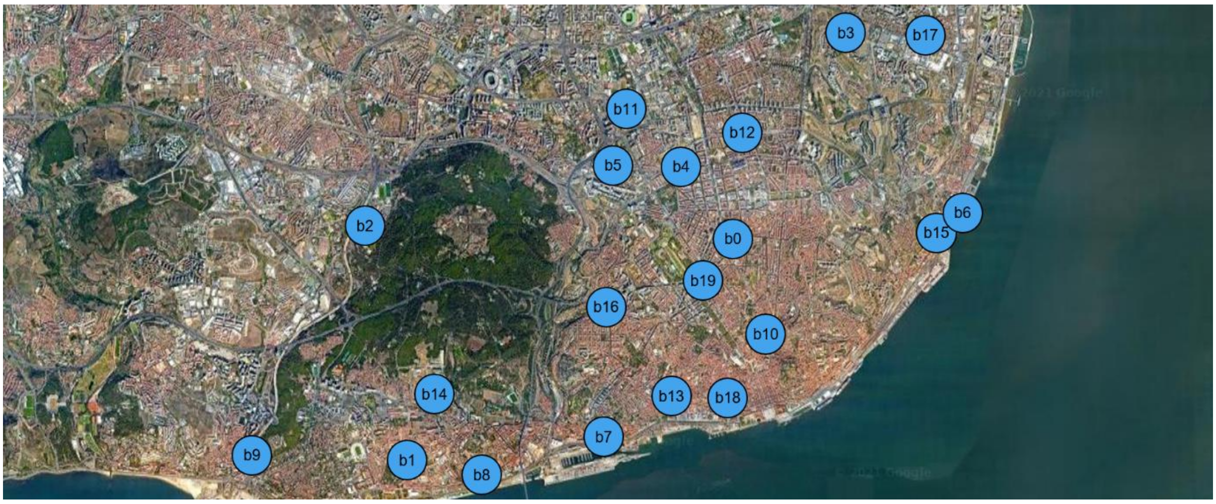


Figure 13 – Location of the twenty vehicle dispatch bases in the municipality of Lisbon

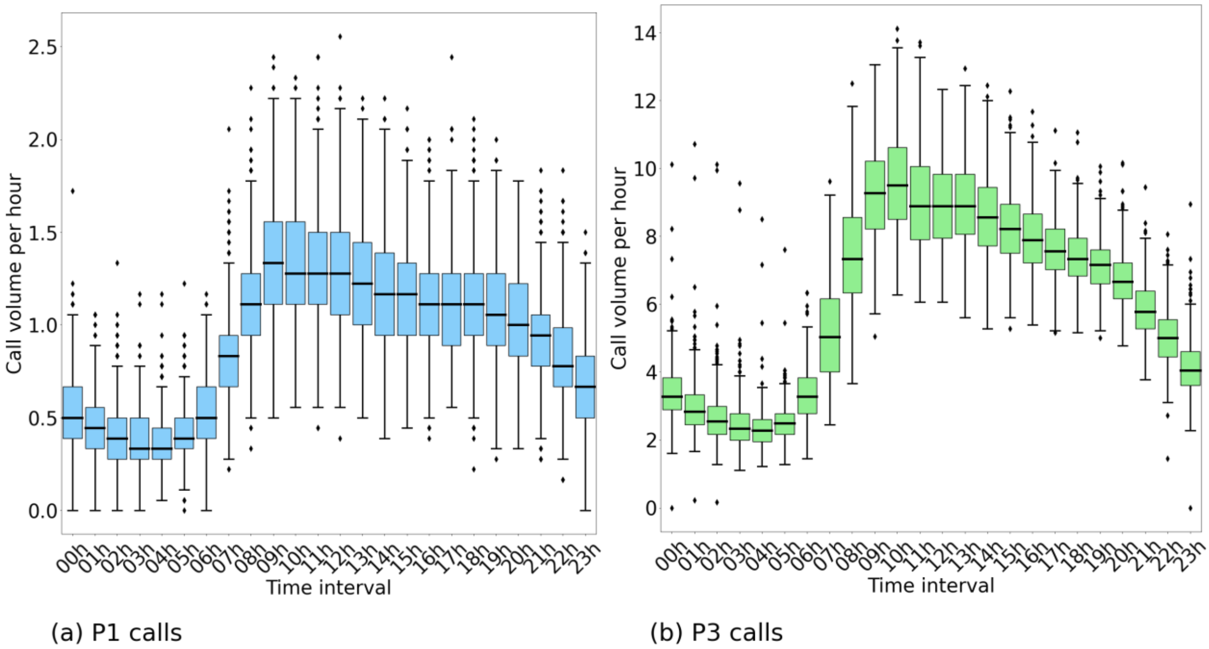
Although call volumes increased from 2017 to 2018, the same is not true for emergency vehicles. Out of the three types of vehicles in analysis, demand only increased for SIV vehicles at 5.72%, while VMER and AEM demand decreased by 0.59% and 7.07%, respectively. Since only three types of vehicles are being analysed, it is possible that other emergency vehicles are being used to answer the increasing number of calls, limiting the ability to make conclusions about these values.

Table 6 shows the average number of dispatches per shift and per type of vehicle for each base. Although it shows that a large number of bases have little to no dispatches, it is important to note that there is a total of ten types of vehicles available to INEM, out of which only three are under analysis. This means that these bases are not purposeless since other vehicles may be stationed and dispatched from them. Additionally, the concentration of certain vehicles in a given set of bases may be related to the management strategy of the EMS provider in the selection of the number and type of vehicles available in each base. In the two years under analysis, the majority of SIV dispatches originated from two bases: b0 (77.79%) and b4 (15.58%). Similarly, the majority of dispatches involving VMER vehicles were concentrated from three bases: b9 (29.06%), b10 (35.68%), and b11 (34.09%). AEM dispatches are the most varied, although ten out of twenty bases issued less than 1.10% of all AEM vehicles. Note the vast number of 0 vehicle dispatches per shift, evidenced by the median measure of Table 6. Out of the 43,800 observations from the two years under analysis, SIV, VMER, and AEM vehicles have no dispatches in 94.07%, 85.63%, and 61.45% of cases, respectively.

Table 6 – Vehicle dispatches per shift for each base in the municipality of Lisbon

	SIV vehicles					**VMER vehicles**					**AEM vehicles**				
Base	**Mean**	**Median**	**Standard deviation**	**Min**	**Max**	**Mean**	**Median**	**Standard deviation**	**Min**	**Max**	**Mean**	**Median**	**Standard deviation**	**Min**	**Max**
b0	**1.87**	**2.00**	**1.30**	**0.00**	**8.00**	0.01	0.00	0.12	0.00	2.00	**3.25**	**4.00**	**2.08**	**0.00**	**8.00**
b1	0.00	0.00	0.04	0.00	1.00	0.00	0.00	0.02	0.00	1.00	**4.16**	**4.00**	**2.71**	**0.00**	**13.00**
b2	0.01	0.00	0.07	0.00	1.00	0.02	0.00	0.14	0.00	2.00	**2.55**	**3.00**	**1.98**	**0.00**	**8.00**
b3	0.01	0.00	0.12	0.00	2.00	0.00	0.00	0.06	0.00	1.00	**6.41**	**7.00**	**2.86**	**0.00**	**18.00**
b4	**0.37**	**0.00**	**0.87**	**0.00**	**5.00**	0.02	0.00	0.15	0.00	2.00	**4.41**	**4.00**	**2.24**	**0.00**	**15.00**
b5	0.01	0.00	0.11	0.00	2.00	0.00	0.00	0.06	0.00	1.00	**1.68**	**0.00**	**2.39**	**0.00**	**17.00**
b6	0.00	0.00	0.05	0.00	1.00	0.00	0.00	0.05	0.00	1.00	**0.98**	**0.00**	**1.58**	**0.00**	**8.00**
b7	0.00	0.00	0.06	0.00	1.00	0.00	0.00	0.02	0.00	1.00	**3.65**	**4.00**	**1.36**	**0.00**	**9.00**
b8	0.00	0.00	0.04	0.00	1.00	0.01	0.00	0.08	0.00	1.00	**0.58**	**0.00**	**1.60**	**0.00**	**11.00**
b9	0.00	0.00	0.00	0.00	0.00	**2.34**	**2.00**	**1.54**	**0.00**	**9.00**	0.00	0.00	0.06	0.00	1.00
b10	0.02	0.00	0.15	0.00	2.00	**2.88**	**3.00**	**1.68**	**0.00**	**9.00**	0.09	0.00	0.31	0.00	2.00
b11	0.02	0.00	0.14	0.00	1.00	**2.75**	**3.00**	**1.51**	**0.00**	**8.00**	0.09	0.00	0.31	0.00	4.00
b12	0.02	0.00	0.14	0.00	2.00	0.01	0.00	0.10	0.00	1.00	**9.01**	**9.00**	**3.88**	**1.00**	**27.00**
b13	0.01	0.00	0.07	0.00	1.00	0.00	0.00	0.02	0.00	1.00	0.03	0.00	0.16	0.00	1.00
b14	0.00	0.00	0.06	0.00	1.00	0.00	0.00	0.03	0.00	1.00	0.02	0.00	0.16	0.00	2.00
b15	0.01	0.00	0.10	0.00	1.00	0.00	0.00	0.06	0.00	1.00	0.03	0.00	0.16	0.00	1.00
b16	0.01	0.00	0.09	0.00	1.00	0.00	0.00	0.06	0.00	1.00	0.04	0.00	0.19	0.00	2.00
b17	0.01	0.00	0.12	0.00	1.00	0.00	0.00	0.05	0.00	1.00	0.05	0.00	0.23	0.00	3.00
b18	0.01	0.00	0.08	0.00	2.00	0.00	0.00	0.03	0.00	1.00	0.04	0.00	0.22	0.00	2.00
b19	0.01	0.00	0.11	0.00	1.00	0.00	0.00	0.06	0.00	1.00	0.02	0.00	0.15	0.00	1.00

The attribute category *seasonal patterns* identified in Table 4 has been of little focus so far since the impact of these attributes is evaluated on the target variables, justifying the need to introduce and analyse them beforehand. Both call and vehicle demand are dependent on *time of day*, *day of week*, *month*, and *season*. Regarding *time of day*, a distinct pattern is identified by observing the hourly demand of call volumes, as shown in Figure 14. For both P1 and P3, the lowest number of calls is during the night, it peaks in the morning, after which it slowly decreases throughout the day. Additionally, the standard deviation is also lower during the night for both priority calls, yet these behaviours are slightly more pronounced for P3 calls.



(a) P1 calls

(b) P3 calls

Figure 14 – Call volumes per time interval

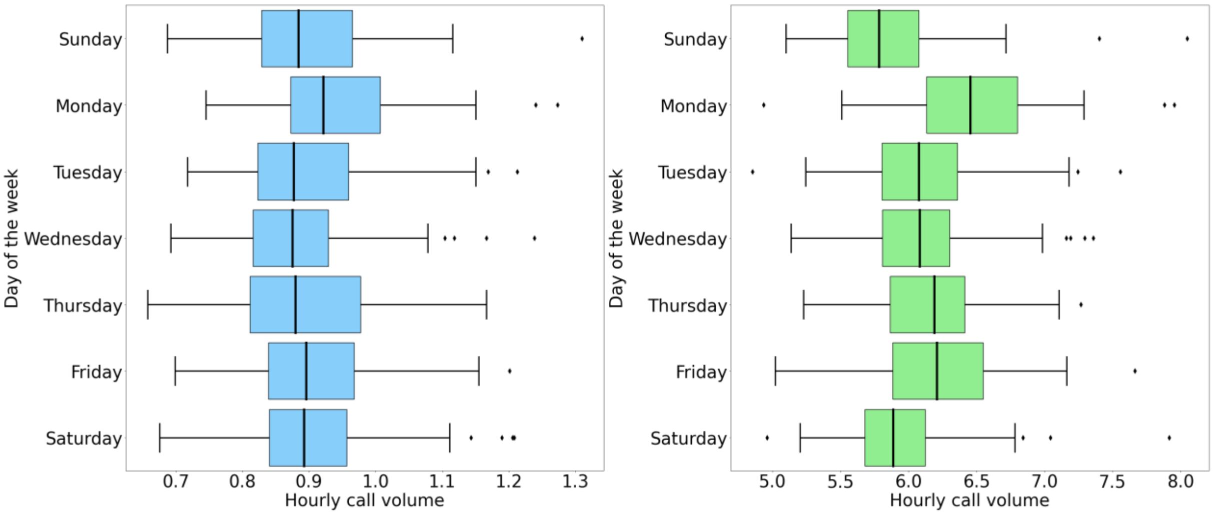
Similarly, the three types of vehicles follow an equivalent pattern, despite having values aggregated in larger time intervals. The values of mean, standard deviation, minimum, and maximum dispatches per shift are available in Table 7. The first shift (00h-08h) has the lowest demand on average, followed by the last shift (16h-00h), and the highest demand is observed in the second shift (08h-16h).

Table 7 – Vehicle dispatches per time interval

	SIV vehicles				VMER vehicles				AEM vehicles			
Shift	**mean**	**std**	**min**	**max**	**mean**	**std**	**min**	**max**	**mean**	**std**	**min**	**max**
00h-08h	0.08	0.38	0.00	8.00	0.23	0.67	0.00	5.00	1.11	1.83	0.00	27.00
08h-16h	0.15	0.66	0.00	6.00	0.53	1.38	0.00	9.00	2.54	3.75	0.00	20.00
16h-00h	0.13	0.56	0.00	6.00	0.45	1.19	0.00	8.00	1.91	2.97	0.00	17.00

As mentioned, different demand volumes are observed depending on the day of the week. Although it not extremely pronounced, weekdays have a slightly higher demand for both calls and vehicles comparatively to weekends. Additionally, Monday is the day of the week with the highest demand and Sunday typically has the lowest. Nonetheless, these variations are relatively low, which can be observed in Figure 15. For example, the difference between the hourly mean of P3 calls between Monday and

Sunday is approximately 0.62 calls, and it is 0.06 for P1 calls. The same pattern is identified for SIV, VMER, and AEM vehicles, although with higher standard deviation during the weekends.

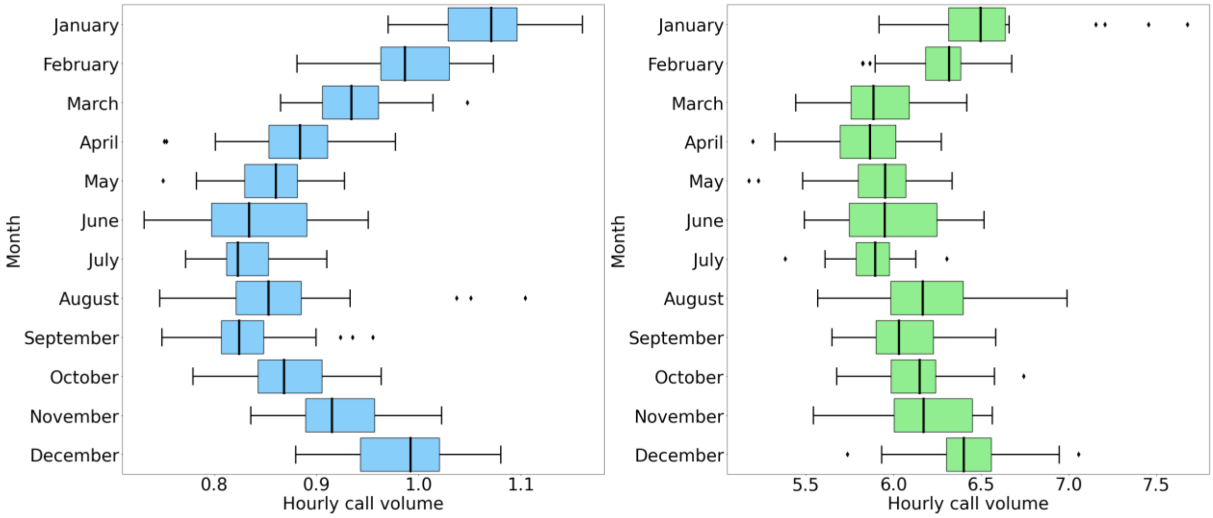


(a) P1 calls

(b) P3 calls

Figure 15 – Hourly call volumes per day of the week

Regarding yearly patterns, the beginning and end of the year are the periods with the highest demand, and demand decreases sequentially throughout the middle of the year, as shown in Figure 16. There is a slight increase in call volumes in August, although it is not replicated in vehicle demand. Additionally, the standard deviation of August is significantly higher than the remaining months. This is a probable consequence of the intense heat that frequently impacts Portugal during this month, which may result in occasional higher EMS demand. These yearly patterns may also be related to the effects of the season. During seasons autumn and winter, higher call and vehicle demand volumes are observed, with summer being the season with the lowest demand and winter with the highest.



(a) P1 calls

(b) P3 calls

Figure 16 – Hourly call volumes per month

Having understood the effects of *seasonal patterns* on the target variables, the correlation with the numerical attributes can be analysed to identify potential relationships. Both priority calls P1 and P3 share similar relationships with the majority of the attributes as is visible by comparing the histograms of Figure 17 and Figure 18. These histograms allow the visualization of pronounced relationships with the target variable of the following: attributes from categories *age*, *resident population*, *accidents/crime*, and attribute *average value pension* from category *employment*. All of these relationships are positive with the exception of *age*. Regarding the relationship with the type of occurrence, low correlation is detected with the ten occurrence types that have significantly low occurrence rates, and different degrees of positive correlation are observed with the remaining. Additionally, the correlation between the two types of calls is highly positive, with a value of 0.66.

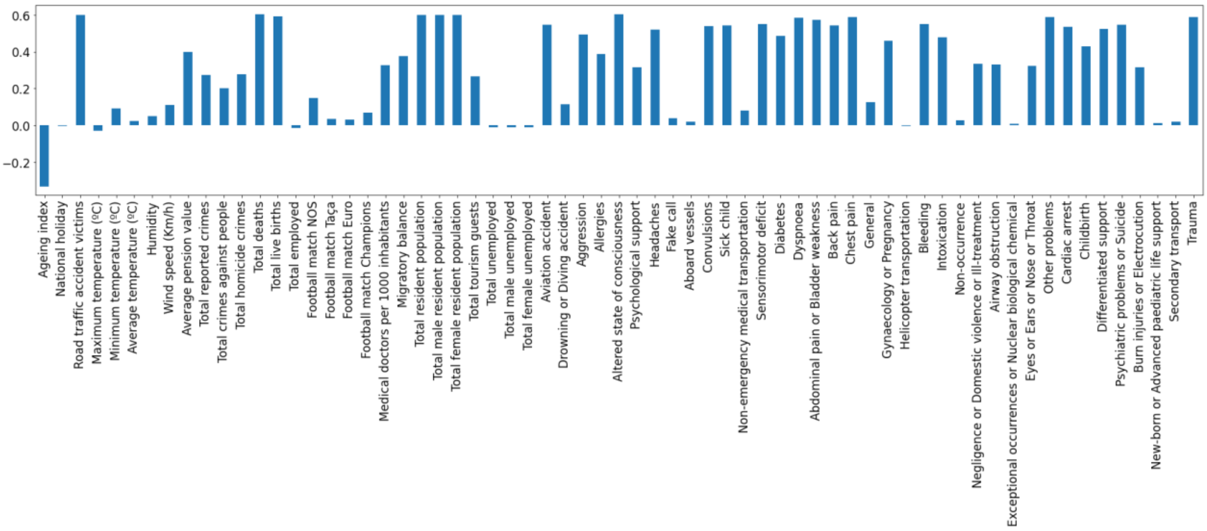


Figure 17 – Correlation between calls of priority P1 and numerical attributes

Although P3 calls generally present higher correlation with the attributes than P1 calls, the demand of both calls is similarly impacted by changes in attribute values.

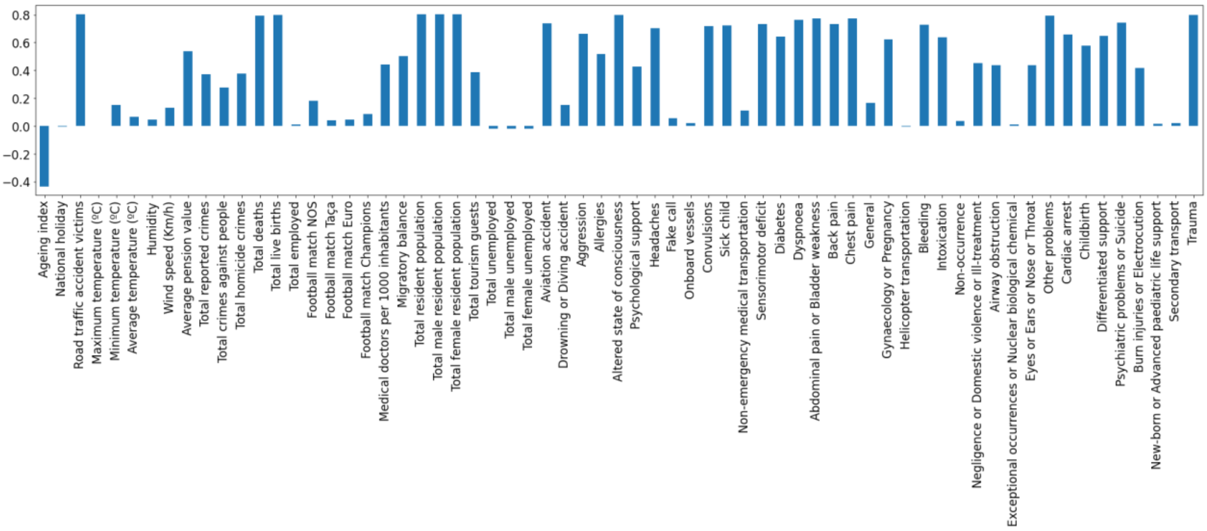


Figure 18 – Correlation between calls of priority P3 and numerical attributes

The relationship between the numerical attributes and dispatches of vehicles SIV, VMER, and AEM is characterized by correlation values lower than 0.05. Additionally, the attributes present different relationships with each one, mimicking random behaviour. No conclusions can be made from this analysis since little to no correlation is observed. The lack of correlation is mainly due to the reduced size of entries in the vehicle dataset and sparse data.

The correlation histograms in Appendix B show that SIV vehicles present the lowest correlation with the available attributes, followed by VMER and AEM. This further confirms the idea that low correlation is due to low demand volumes and insufficient data representation since higher correlation is observed in target variables with greater demand. Regarding relationships between the target variables, the following correlation values are observed: -0.06 between SIV and VMER, 0.17 between SIV and AEM, and -0.21 between VMER and AEM.

4.3. Chapter conclusions

Historical data from 2017-2018 is available in two datasets, one regarding hourly call volumes and the other concerning vehicle dispatches per 8-hour shift. While the first includes information for continental Portugal, vehicle demand is only studied for the municipality of Lisbon. Both datasets include sixty-five attributes collected through multiple sources, that are valuable to explain demand variations.

Prior to using these datasets to develop predictive models, this chapter serves to introduce the available data and showcase its particularities. The attributes are presented, and demand patterns are evaluated with the goal of obtaining insights about the behaviour of EMS demand. Many of the observations gathered through this analysis align with the realizations previously identified in the literature regarding EMS demand.

5. Experimental Results

This chapter provides an in-depth description of the process leading to the obtainment of the results from the case study application. The implementation process of the applied algorithms is characterized, including attribute selection and hyperparameter tuning processes. The data preparation process is demonstrated in Section 5.1, followed by the application of the GMM algorithm in Section 5.2 and the multiple ML algorithms in Section 5.3. Finally, the obtained results are analysed in Section 5.4, and chapter conclusions are presented in Section 5.5.

5.1. Data Preparation

The goal of this dissertation is the elaboration of a reliable forecasting tool applied to the case study of INEM, to predict call volumes and vehicle dispatch demand. This tool is directed toward operational planning and the elaboration of predictions considering short-term forecasting horizons. For this purpose, the GMM and eight other ML algorithms are trained and tested, to identify and select the best performing model. These eight ML algorithms are the following: Naïve Bayes, KNN, SVM, Random Forest, Extra Trees, Gradient Boosting, AdaBoost, and Bagging. The workflow identified for the development of predictive models based on ML algorithms in Section 3.4.3 is adopted as the computational methodology. The developed model is expected to be an improvement on the existing prediction methods currently in use by INEM. However, this improvement is not measurable because the planning decisions are currently guided by intuition. Although the performance of the current method is not available for comparison, classical techniques such as time series models are common in the industry, making them a good baseline option. However, due to the particularities of each method and the difference in evaluation metrics, a direct comparison is unachievable. Therefore, due to the unavailable calculations currently in use and the difficulty in establishing a comparison with popular time series models, the results obtained in this work are simply compared with one another to identify the best model without defining any baseline. The experiments are conducted on a laptop with a 2.90GHz Intel Core i7-7500U processor and 8.00GB of RAM, with two cores, running on Windows 10.

A separate model is developed for each target demand volume that is predicted, i.e., for each type of vehicle (SIV, VMER, and AEM) and each priority level call (P1 and P3). Although Zhou et al. (2015) used a temporal granularity of 2-hour intervals, this application considers the highest granularity level available since the forecasts are directed towards short-term operational planning. This means that call demand is predicted on an hourly level and vehicle demand for 8-hour shifts. These intervals are expected to be appropriate for the planning decisions for which they are intended. While the temporal granularity is fine for call volumes and low for vehicle dispatches, the opposite is true for spatial granularity. The twenty bases in Lisbon represent the spatial location of the vehicle demand, and call demand is grouped by the eighteen districts in mainland Portugal.

The two datasets introduced in Chapter 4 are subjected to a normalization procedure to obtain uniform data due to the extreme differences in the range of values of the attributes. The range of values of the non-binary attributes presented in Section 4.1 is shown in Table 8. The homogenization of the data facilitates the training process by ensuring a lower computational cost and increasing the ability of the model to rapidly converge. The normalization process is done via a min-max scaling method that rescales the dataset so that all attribute values are in the same range. The values of each attribute are transformed into a [0,1] range whilst preserving the form of their original distribution. The min-max scaling method normalizes the attributes by subtracting their minimum value and dividing it by the range.

Table 8 – Range of values of non-binary attributes

Attributes	Minimum	Maximum	Range
Ageing index	135.8	203.1	67.3
Road traffic accident victims	427	9,919	9,492
Maximum temperature (ºC)	3	46	43
Minimum temperature (ºC)	-6	30	36
Average temperature (ºC)	-1	38	39
Humidity	14	99	85
Wind speed (Km/h)	2	48	46
Average pension value	4,654	7,014	2,360
Total reported crimes	20,238	111,225	90,987
Total crimes against people	4,721	26,989	22,268
Total homicide crimes	5	39	34
Total deaths	122	2,571	2,449
Total live births	42	2,174	2,132
Total employed	4,569,000	4,853,300	284,300
Medical doctors per 1000 inhabitants	2.9	6.5	3.6
Migratory balance	-2,371	11,640	14,011
Total resident population	105,479	2,271,772	2,166,293
Total male resident population	50,064	1,063,052	1,012,988
Total female resident population	55,415	1,208,720	1,153,305
Total tourism guests	52,843	777,079	724,236
Total unemployed	339,035	403,771	64,736
Total male unemployed	150,357	184,051	33,694
Total female unemployed	188,678	219,720	31,042
Aviation accident	0	64	64
Drowning or Diving accident	0	5	5
Aggression	0	35	35
Allergies	0	11	11
Altered state of consciousness (ASC)	0	224	224
Psychological support	0	7	7
Headaches	0	22	22
Fake call	0	3	3
Aboard vessels	0	1	1
Convulsions	0	26	26
Sick child	0	48	48
Non-emergency medical transportation	0	3	3
Sensorimotor deficit	0	37	37
Diabetes	0	18	18
Dyspnoea	0	155	155
Abdominal pain or Bladder weakness	0	67	67
Back pain	0	34	34
Chest pain	0	64	64
General	0	4	4
Gynaecology or Pregnancy	0	19	19
Helicopter transportation	0	1	1
Bleeding	0	37	37
Intoxication	0	122	122
Non-occurrence	0	2	2
Negligence or Domestic violence or Ill-treatment	0	7	7
Airway obstruction	0	10	10
Exceptional occurrences or Nuclear biological chemical	0	1	1

Table 8 – continuation

Attributes	**Minimum**	**Maximum**	**Range**
Eyes or Ears or Nose or Throat	0	6	6
Other problems	0	148	148
Cardiac arrest	0	54	54
Childbirth	0	15	15
Differentiated support	0	28	28
Psychiatric problems or Suicide	0	34	34
Burn injuries or Electrocution	0	14	14
New-born or Advanced paediatric life support	0	1	1
Secondary transport	0	1	1
Trauma	0	164	164

As seen in the literature review, ML algorithms can be applied to regression and classification problems. For the problem at hand, a classification prediction approach is adopted since it allows the use of evaluation metrics such as ROC (Receiver Operating Characteristics) curves and confusion matrices, which are relevant measures in the nature of the demand that is predicted. These metrics not only evaluate the severity of a model's error, but also evaluate a model's capability to recognize each class.

The target variable in a classification problem has labels or classes instead of continuous numerical values. Therefore, the continuous outputs of the five target variables are transformed into classes through a clustering procedure. The K-means clustering algorithm is selected due to its vast application in similar problems, computational speed, and ease of convergence. The main limitation of this algorithm is that it requires a specification for the value of the K number of clusters to create, which has a great impact on the results. For this reason, the elbow method is used to determine the optimal number of clusters. This method runs K-means clustering for a range of values of K, and then computes a distortion score for each value of K, which corresponds to the sum of the square distances from each point to its assigned centre. K values from 1 to 5 are tested to identify the elbow point, which indicates the optimal number of clusters. Figure 19 shows the graphs where a clear elbow point can be observed visually for each target variable. Nonetheless, this point is identified using an elbow locator function.

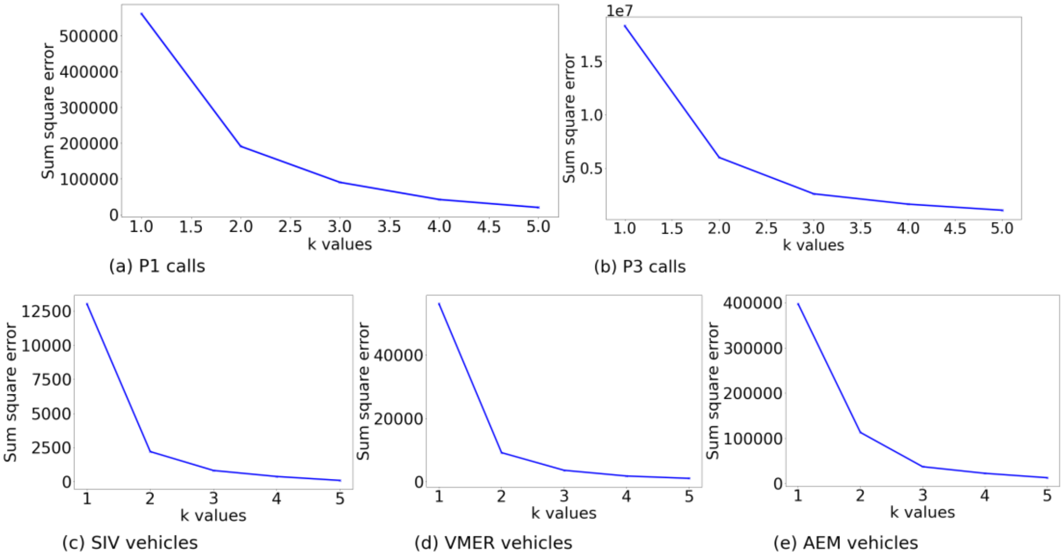


Figure 19 – Elbow method for target variables

For all target variables, an optimal number of two clusters is obtained, and the ranges of each are presented in Table 9. It is important to note that the obtained classes demonstrate imbalance in terms of the number of observations in each class. For all the target variables, the number of instances in the cluster with the lowest values is significantly higher (>75%) than the one with the highest values.

Table 9 – K-means clustering of target variables

Target variable	Range of values	Optimal K	Clusters	Range of clusters	Observations per cluster	Ratio per cluster
P1 calls	0 - 15	2	Cluster 0	[0,1]	247,224	78.39%
			Cluster 1	[2,15]	68,136	21.61%
P3 calls	0 - 68	2	Cluster 0	[0,13]	280,271	88.87%
			Cluster 1	[14,68]	35,089	11.13%
SIV vehicles	0 - 8	2	Cluster 0	[0,1]	42,268	96.50%
			Cluster 1	[2,8]	1,532	3.50%
VMER vehicles	0 - 9	2	Cluster 0	[0,1]	38,907	88.83%
			Cluster 1	[2,9]	4,893	11.17%
AEM vehicles	0 - 27	2	Cluster 0	[0,3]	33,081	75.53%
			Cluster 1	[4,27]	10,719	24.47%

The most common evaluation metric of predictive models is accuracy, which measures the percentage of correct predictions. However, this metric alone is often not sufficient to obtain a clear view on the execution of the model. The case of SIV vehicles is an example of highly imbalanced data where accuracy can wrongly measure the performance of a model, since an accuracy of 96.50% is achieved if the algorithm predicts cluster 0 for all of the samples.

Four metrics are selected for the evaluation of the classification models developed in this work in order to obtain a broad and in-depth understanding of each model's performance. Furthermore, it helps to overcome the difficulty in identifying a single measure capable of measuring the overall conduct of a model. These metrics include the ROC Area Under the Curve (AUC), accuracy, Precision, and Recall. ROC curves are commonly used to measure the performance of classification models since they measure a model's ability to distinguish between two classes. The ROC curve plots two parameters: true positive rate and false positive rate, presented in Equation (1) and Equation (2), respectively. These equations use the concepts of true positive and true negative which represent the number of samples that were correctly classified as 1 and 0, respectively, as well as the concepts of false positive and false negative which are errors where a 0 is predicted as a 1, and a 1 predicted as a 0, respectively.

$$True\ positive\ rate\ or\ Recall\ = \frac{True\ positives}{True\ positives\ +\ False\ negatives} \tag{1}$$

$$False\ positive\ rate\ = \frac{False\ Positives}{True\ negatives\ +\ False\ positives} \tag{2}$$

The area under a ROC curve is known as the AUC, and it provides an overall summary of the ROC curve's performance. An AUC that is close to 1 indicates that the model has good class separation capabilities, while an AUC under 0.5 performs worse than random. The AUC metric is frequently used as an overall measure of the performance of classification models. Since the classes are treated as equal, ROC curves are mainly useful when dealing with roughly equal numbers of observations in each class, while Precision-Recall curves are suitable when handling class imbalance. The reason for this is that the calculations of both Precision and Recall do not use the true negatives and are only concerned with correctly predicting class 1. Therefore, when the number of observations in class 1 is significantly lower than class 0, the Precision-Recall curve is useful to recognize how well the minority class is being predicted. Precision describes a model's ability to correctly predict the positive class since it measures the proportion of correct positive identifications. On the other hand, Recall measures the proportion of positives that is correctly predicted. The equation for Precision is presented in Equation (3), and Recall is the same ratio as the true positive rate shown in Equation (1).

$$Precision = \frac{True\ Positives}{True\ Positives + False\ Positives} \tag{3}$$

Since the use of multiple metrics allows a thorough evaluation of a classification model's performance, metrics accuracy, AUC, Precision, and Recall are selected to evaluate the models developed in the following sections.

5.2. GMM Application

The choice for the exploration of the GMM algorithm is motivated by the results found by Zhou et al. (2015), briefly summarized in Section 3.3.3, and their recommendations to further explore attributes to obtain accuracy improvements. Despite it being mostly used for clustering purposes, GMM is a learning algorithm that can be used to model the data and obtain predictions about future instances. The model assumes that all datapoints are generated from a mixture of Gaussian distributions with unknown parameters. Note that Zhou et al. (2015) model the spatial distribution of demand by the K components of the Gaussian mixture, which differs from this application. Here, the spatial and temporal aspects of demand are considered as attributes and the GMM is modelled over the entire set of attributes.

Two different estimation strategies can be used to fit the Gaussian distributions, where the first uses an expectation-maximization algorithm to fit the mixture of normal distributions, and the other uses variational inference algorithms. The variational inference algorithm is an extension of expectation-maximization that uses information from prior distributions to add regularization and avoid singularities. The variational algorithm needs more hyperparameters due to its Bayesian nature, and the inference is significantly slower because of the need for extra parametrization. However, due to the fact that Zhou et al. (2015) use a Bayesian estimation approach in their work, this same approach is selected for this application.

The selection of the most relevant subset of attributes to train and test the GMM algorithm is an important step before the application. Two methods are explored for this purpose, a filter method and a wrapper method, and the results obtained from each are the subject of comparison. The selected filter method, Pearson's correlation coefficient, is used for a correlation analysis (CA). This analysis involves first selecting the attributes that have the highest correlation with the target variable, and then removing the attributes that are highly correlated with each other and therefore provide redundant information to the model. Note that out of a group of attributes that are highly correlated with each other, the one with the highest correlation with the target variable is maintained. Attributes with a correlation higher than 0.4 or lower than -0.4 with the target variable are selected, representing a moderate level of correlation. After this, attributes with very high correlation, of values higher than 0.8 or lower than -0.8 with each other are removed, and only the most relevant one remains (Guerra-Manzanares et al. 2019; Pallonetto et al. 2019; Rastegari et al. 2019).

GA is selected as the wrapper to identify the best subset of attributes. The algorithm is applied over 5 generations with a crossover probability of 0.5 and mutation of 0.2, and with a population size of 15. The high search space means that a low crossover probability should be avoided to ensure that the search space is sufficiently explored, yet it should not be too high to ensure the convergence of the algorithm. The low population size suggests that convergence is likely, so a high mutation probability is used to balance this and avoid being trapped in a local optimum. The measure of accuracy is used as a fitness score to identify the group of attributes that maximizes this metric. The algorithm is shown to converge within the 5 generations, since all the individuals in the final population present the same solution. It is important to note that the attributes subjected to these selection processes do not include those from category *seasonal patterns* due to their impact on the target variables as presented in Section 4.2. These attributes are considered indispensable and are used as a set of base attributes that are always present in all models.

The selected Bayesian GMM algorithm has a large number of hyperparameters, which are subjected to a tuning process in the following stage. However, the prior on the Gaussian distribution, known as the mean prior, and the prior on the Wishart distribution, known as the empirical covariance prior, are not tuned. Both parameters represent priors that are altered throughout the learning process. Therefore, the ideal prior values are the mean of the attributes in the dataset for the mean prior and the covariance of the dataset for the covariance prior. These optimal values are the default for both parameters, eliminating the need for their tuning. The hyperparameters subjected to tuning are shown in Table 10, along with the empirical values that are tested for each. The selection of these values is based on the general values that typically result in well-performing models. The random seed given to the method to initialize the parameters is set to 0 for all applications to control randomness and obtain reproducible outputs. Additionally, the maximum number of iterations is set to 200 to guarantee the convergence of the algorithm, and 10 initializations are performed to ensure its convergence to an optimal configuration.

Table 10 – Empirical values selected for hyperparameter tuning of GMM

Hyperparameter	**Tuning values**
Number of mixture components	[15, 20, 25, 30, 35, 40, 45, 50]
Type of covariance	['full', 'tied', 'diag', 'spherical']
Convergence threshold	[1e-5, 1e-4, 1e-3]
Regularization of covariance	[1e-6, 1e-5, 1e-4]
Initialization method	['kmeans', 'random']

Table 10 – continuation

Hyperparameter	Tuning values
Type of weight concentration prior	['dirichlet_process', 'dirichlet_distribution']
Dirichlet concentration prior	[0.07, 0.05, 0.03, 0.02, 0.01]
Precision prior on the mean distribution	[1, 5, 10, 15]
Prior on the number of degrees of freedom	Different for each dataset: 6 values with intervals of 2 degrees between each, starting at the lowest acceptable value, the number of attributes, which corresponds to the default value

The range of the number of mixture components is based on Zhou et al. (2015), where 15 components were found capable of capturing the wide complexity of the data, yet small enough for computation ease. Additionally, they identified that a large number of mixture components can lead to overfitting, and therefore set the a priori maximum number of components to 50. The four covariance type options of Table 10 define constraints on the general covariance matrices: ‘full’ allows each component to have its covariance matrix, ‘tied’ restricts all components to share the same covariance matrix, ‘diag’ allows different diagonal covariance matrices, and ‘spherical’ allows each component to have its single variance. Although ‘full’ covariance is expected to perform best in general, it is prone to overfitting.

Two other adjustments are made on covariance, the first being of hyperparameter covariance threshold which serves as a stopping criterion since the iterations stop when the average gain on the likelihood of the training data is below this threshold. The second is the regularization of the covariance which is performed by adding a non-negative value to the diagonal of the covariance to ensure that the matrices are positive. This regularization has an impact on the model and should, therefore, be attributed a low value. However, this is not always possible and models with a large number of mixture components occasionally require higher regularization.

Options ‘kmeans’ and ‘random’ refer to the initialization processes of the weights, the means, and the covariances of the model. Furthermore, the type of prior on the distribution of the weights given to each component relates to the way that the number of components is defined, leading to two distinct models: a finite mixture model with Dirichlet distribution and an infinite mixture model with Dirichlet Process. The first uses the value given as a prior for a finite number of mixture components, while the second assumes an infinite number of components and uses the prior value as a maximum, assigning a weight of 0 to components it does not use. The advantage of the second is that there is no need to accurately calculate the number of components since the algorithm is capable of computing it. In practice, the Dirichlet Process is approximated and a distribution with a high fixed maximum number of components is used, called the Stick-breaking representation.

Additional regularization of the weights is done through the prior weight concentrations, known as γ in the literature, which defines the concentration of each component on the Dirichlet weight distribution. A low concentration prior results in the model putting most of the weight on few components and giving weights close to 0 to the remaining, while a high concentration prior allows a larger number of components to be active in the mixture. Another prior is that of the precision prior on the mean Gaussian distribution which controls the range of the mean values, where larger priors concentrate the means around the values of the aforementioned mean priors. Finally, a prior is given for the number of degrees of freedom on the Wishart covariance distributions.

The described hyperparameters are tuned via two methods, and their results are used to draw comparisons: RS and GA. The RS tuning process is applied with a scoring strategy of maximizing accuracy, and the parameters of the estimator are optimized by a stratified cross-validated search with 5 folds to preserve the percentage of samples in each class. The parameters are sampled uniformly although not all parameter values are tried out, but rather 10 parameter settings are sampled from the specified listed values. Since the values of the hyperparameters are presented as lists, sampling without replacement is performed, meaning that each sample unit of the population is only selected once. Similar to the previous application, the GA is run through 5 generations with a population size of 15, and the fitness score is set to maximize accuracy. A higher crossover probability of 0.9 is used for this application of the GA because the search space is smaller, reducing the risk of not converging. This also reduces the need for a high mutation value as a sufficient exploration of the search space is conducted. Nonetheless, to ensure the diversity of the population, a mutation probability of 0.03 is selected. The convergence plots of the GA tuning procedures are available in Appendix C.

Four GMM models are obtained at this stage: two are built on the CA dataset, having the hyperparameters of one been tuned with a RS and the other with GA; the other two models are constructed on the GA dataset, with the same distinction in tuning processes. The four GMM are built for each target variable, resulting in a total of twenty models. Results obtained from cross-validation are presented in Table 11. The cross-validation is performed using a stratified K-fold method to have a good representation of each class in each fold. The cross-validator uses 5 splits of the dataset to train and validate, and returns the mean and standard deviation of the results for each evaluation metric.

Table 11 – Cross-validation results from GMM application

		AUC		**Accuracy**	
		RS	GA	RS	GA
SIV	CA	**0.9835**	0.9160	0.9733	**0.9768**
	GA	0.9315	0.9699	0.9733	0.9745
VMER	CA	**0.9856**	0.9842	0.9615	0.9613
	GA	0.9269	0.9241	0.9622	**0.9657**
AEM	CA	**0.9463**	0.9330	0.8867	**0.8888**
	GA	0.7572	0.7339	0.8870	0.8877
P1	CA	**0.5589**	0.5129	0.8370	**0.8384**
	GA	0.2931	0.3260	0.8376	0.8375
P3	CA	0.7663	**0.8070**	0.9502	**0.9536**
	GA	0.6912	0.1269	0.9412	0.9378

Regarding attribute selection, Table 11 shows that, with one exception, the CA dataset provides better results than GA. Additionally, the CA method is significantly faster computationally, further supporting the choice to use this practice. Table 11 also shows different preferences in regard to hyperparameter tuning methods depending on the evaluation metric that is assessed, where AUC recommends RS and accuracy shows GA to be the best method. Therefore, the choice of tuning process is dependent on which evaluation metric is prioritized. The full results for all evaluation metrics including standard deviation values of the cross-validation are included in Appendix D.

5.3. ML Application

Similar to the GMM application, both CA and GA are used as attribute selection methods for the ML algorithms. The CA results in exactly the same selection of attributes since it is independent of the learning algorithm. This is not the case for the GA method which must be repeated for each of the eight algorithms, which is a significantly lengthier procedure. The GA is applied in the same way for each ML algorithm as for GMM, with accuracy as the fitness score.

Following the flowchart of Section 3.4.3 (Figure 9), the hyperparameter tuning process is only conducted after all performances are evaluated. This is because only the best performing ML algorithms follow through to the tuning procedure. The eight ML algorithms are applied with default hyperparameters to identify the ones that best model the data. Preliminary results of accuracy and AUC obtained from a stratified 5-fold cross-validation identify algorithms Gradient Boosting and AdaBoost as the ones that result in the best performing models, as shown in Table 12. These algorithms are similar, as described in Chapter 3.4.1, and both follow a boosting approach. The third best performing algorithm is SVM, however, the lengthy computational time of this algorithm does not justify its further exploration for operation planning. The complete results are available in Appendix E.

Table 12 – Best performing ML algorithms for each dataset

		CA dataset	**GA dataset**
SIV	AUC	Gradient Boosting	Extra Trees
	Accuracy	Gradient Boosting	Gradient Boosting
VMER	AUC	AdaBoost	AdaBoost
	Accuracy	AdaBoost	AdaBoost
AEM	AUC	Gradient Boosting	Random Forest
	Accuracy	SVM	SVM
P1	AUC	AdaBoost	Gradient Boosting
	Accuracy	AdaBoost	Gradient Boosting
P3	AUC	AdaBoost	AdaBoost
	Accuracy	SVM	SVM

Table 13 compares the results obtained through the CA dataset and GA dataset for both Gradient Boosting and AdaBoost. Default hyperparameters are employed and metrics AUC and accuracy are used to draw comparisons. For all cases, Gradient Boosting performs best with the attributes selected through GA, and with a few exceptions, AdaBoost performs best with the CA dataset. The main difference between these two datasets is the number of attributes that are selected for each. The CA method selects significantly fewer attributes, 44.8 on average, while the GA method selects on average 83.3 attributes. This is mainly due to the low correlation between the available attributes and the target variables. Therefore, for this data, Gradient Boosting performs best with additional attributes while AdaBoost tends to perform better with fewer attributes. Due to this, hyperparameter tuning is performed for the Gradient Boosting model trained on the GA dataset and the AdaBoost model with the CA dataset.

Table 13 – Comparison between CA and GA datasets for Gradient Boosting and AdaBoost

		Gradient Boosting		AdaBoost	
		CA dataset	GA dataset	CA dataset	GA dataset
SIV	AUC	0.9873	**0.9908**	**0.9845**	0.9845
	Accuracy	0.9775	**0.9826**	**0.9752**	0.9745
VMER	AUC	0.9905	**0.9906**	**0.9912**	0.9911
	Accuracy	0.9656	**0.9661**	**0.9667**	0.9667
AEM	AUC	0.9596	**0.9658**	0.9567	**0.9575**
	Accuracy	0.8964	**0.9106**	0.8933	**0.8940**
P1	AUC	0.8567	**0.8577**	**0.8588**	0.8566
	Accuracy	0.8415	**0.8428**	**0.8421**	0.8415
P3	AUC	0.9788	**0.9797**	**0.9819**	0.9819
	Accuracy	0.9587	**0.9596**	**0.9594**	0.9592

To identify the best method, hyperparameter tuning is performed both via RS and GA, identically to the GMM application. The tuning plots for the GA method are available in Appendix C for both Gradient Boosting and AdaBoost. The hyperparameters tuned for the Gradient Boosting models are shown in Table 14. These values are chosen based on the best performing ones in other applications, and adjustments were made considering the selected values, i.e., further values were added if the highest or lowest value was always picked.

Table 14 – Empirical values selected for hyperparameter tuning of Gradient Boosting

Hyperparameter	**Tuning values**
Learning rate	[0.05, 0.1, 0.15, 0.2, 0.25, 0.3]
Number of estimators	[50, 150, 250, 350, 450]
Subsample	[0.6, 0.7, 0.8, 0.9, 1]
Minimum samples split	[2, 5, 50, 100, 500, 1000]
Minimum samples leaf	[1, 5, 10, 50, 100, 500]
Maximum depth	[1, 2, 3, 4, 5]
Maximum features	[$\sqrt{\text{number of features}}$, 15, 25, 35, 45, 55]

The logistic regression loss function is selected to be minimized in each split and mean squared error with improvement score by Friedman is chosen as the criterion function to measure the quality of each split. Additionally, no pruning is performed in this initial stage since a learning curve analysis is needed to give input on whether a model is underfitting or overfitting the data. Pruning not only increases the generalizing ability of a model and reduces overfitting, but also increases inference speed. However, it often affects accuracy, meaning that it should be carefully added in situations where either computational time is excessive or in cases of overfitting.

Hyperparameters learning rate, number of estimators, and subsample manage the boosting operation of the model. Learning rate determines the impact of each tree in the final outcome. Lower values shrink the contribution of each tree and result in more generalizable models. However, there is a trade-off between the learning rate and the number of estimators since low learning rates require more trees, which is computationally more expensive. The number of trees that are modelled represent the number of boosting stages that are performed. Although too many stages lead to overfitting, the Gradient Boosting algorithm is sufficiently robust, and a large number of estimators generally results in better performance. The fraction of observations selected to fit each tree is given by subsample. Robust models are obtained when this value is less than 1 due to variance reduction.

The remaining hyperparameters are used to define the individual trees of a model. Minimum samples split expresses the minimum number of observations required in a node for it to be considered for splitting. On the other hand, minimum samples leaf defines the minimum observations required in a leaf node. A split point is only considered if it leaves at least the minimum training observations in each of the two branches. Small values are preferred in imbalanced class problems for both hyperparameters. Additionally, both are used to control overfitting, where high values prevent models from learning overly specific relations of a particular observation, although values too high lead to underfitting. Maximum depth and maximum features are also used to control overfitting. The maximum depth of individual estimators limits the number of nodes in a tree, and the maximum number of features restricts the number of randomly selected attributes that are considered when searching for the best split. Overly high values for either of these two hyperparameters lead to overfitting.

The AUC and accuracy performance comparisons obtained from stratified 5-fold cross-validation for both tuning processes are presented in Table 15. Improvements are most significant for RS, meaning that the hyperparameter values selected through this method are used for the final Gradient Boosting model.

Table 15 – Cross-validation results of Gradient Boosting with GA dataset before and after hyperparameter tuning

		SIV	**VMER**	**AEM**	**P1**	**P3**
AUC	Default	0.9908	0.9906	0.9658	0.8577	0.9797
	RS	**0.9923**	**0.9910**	**0.9737**	**0.8607**	0.9837
	GA	0.9918	0.9909	0.9724	0.8602	**0.9838**
Accuracy	Default	0.9826	0.9661	0.9106	0.8428	0.9596
	RS	**0.9834**	**0.9673**	**0.9151**	**0.8436**	**0.9616**
	GA	0.9833	0.9667	0.9129	0.8433	0.9613

AdaBoost has a significantly lower number of hyperparameters that require tuning. Parameters learning rate and number of estimators are identical to Gradient Boosting, and the trade-off between them is equivalent. These are the only two hyperparameters subjected to tuning for this algorithm, as the others are hand-picked. The choice of two options is given for the hyperparameter named algorithm. The algorithms are adaptations of the main AdaBoost idea with extended multiclass capabilities. One of them uses class probability estimates while the other is discrete-valued, using classification outputs. The first one is selected since it converges faster and achieves a lower test error with fewer boosting iterations. The performance improvements over the default values are presented in Table 16.

Table 16 – Cross-validation results of AdaBoost with CA dataset before and after hyperparameter tuning

		SIV	**VMER**	**AEM**	**P1**	**P3**
AUC	Default	0.9845	0.9912	0.9567	0.8588	0.9819
	RS	**0.9846**	**0.9913**	**0.9569**	**0.8595**	**0.9827**
	GA	0.9844	**0.9913**	**0.9569**	0.8594	0.9826
Accuracy	Default	0.9752	0.9667	**0.8933**	0.8421	0.9594
	RS	**0.9774**	**0.9672**	**0.8933**	**0.8425**	**0.9599**
	GA	**0.9774**	0.9667	**0.8933**	0.8422	0.9598

The improvements obtained for the AdaBoost algorithm are significantly inferior to those obtained for Gradient Boosting. This is mainly due to the lower number of hyperparameters that are tuned for AdaBoost, so the improvements that can be achieved are limited. It is important to note that in some cases evaluation metrics such as Precision and Recall, available in Appendix F, prove to be better with default hyperparameters, which is due to the maximization goal of the tuning processes being set to accuracy. Although only by a small margin, Table 16 shows that the best results for both accuracy and AUC are obtained with the hyperparameter values selected by RS, justifying the use of these values for the final AdaBoost model.

5.4. Results Analysis

The application processes presented in the previous sections allow the initial identification of the best performing models from each application through metrics of accuracy and AUC. These models are further analysed in this section to recognize the best one and provide a recommendation. The results of the final models obtained from stratified 5-fold cross-validation are available in Appendix G. These models are the following:

- GMM application: models trained with the CA dataset surpass those trained with attributes selected via GA, and tuning results are similar for both RS and GA methods, meaning that both models are examined;
- Gradient Boosting application: the GA dataset outperforms the CA dataset, and RS proves to be a better tuning method than GA;
- AdaBoost application: the CA dataset provides better results than GA, and RS tuning produces better models than GA.

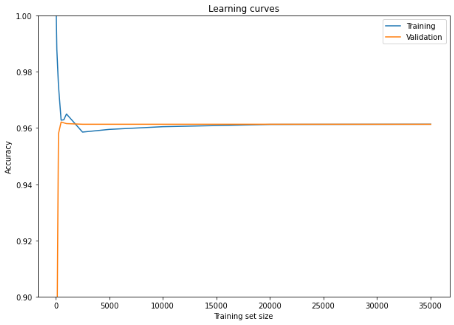
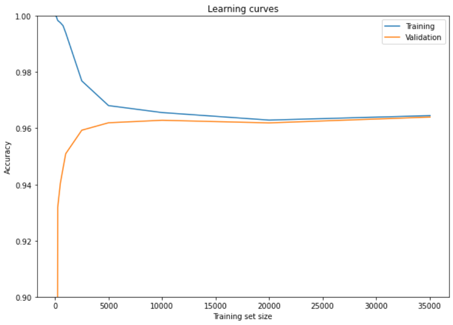
These models are studied through analyses of learning curves, ROC curves, Precision-Recall curves, and confusion matrices. The purpose of these measures is to obtain a deeper understanding of the performance of each model. Although accuracy is the most commonly used metric, it does not provide a full understanding of how models perform with different sets of data and the kind of errors that are made, along with the reasons for these errors. No model is capable of 100% accuracy due to the always existing irreducible error, yet this analysis is expected to give the decision-maker sufficient tools to select the most appropriate model to assist in EMS planning.

5.4.1. Performance Evaluation

The analysis of individual evaluation metrics provides concrete and valuable information to easily compare the performance of the developed models. So far, an emphasis has been placed on AUC and accuracy metrics to select the best models and identify the best methods for their development. This section focuses on other metrics to obtain an overall view of each model's behaviour and predictive capabilities.

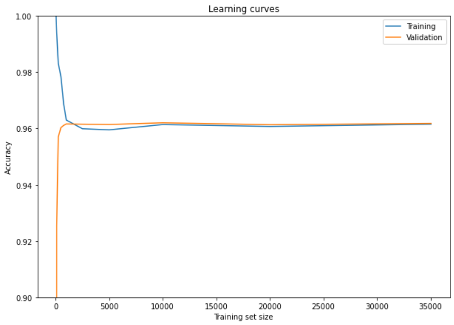
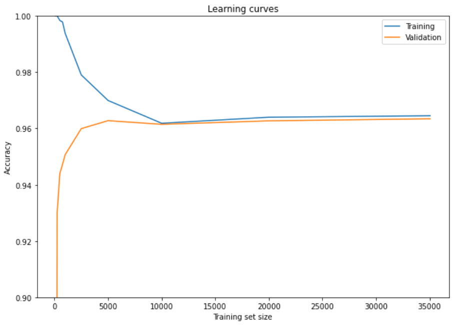
The learning curves for the selected models are presented in Appendix H. These curves measure the quality of the model's fit to the data and provide a view of its generalization ability. In general, the majority of the models have an adequate behaviour, reaching a point of stability with a small gap between the two curves. However, there are two situations where the learning curves demonstrate ill-fitting models. The first is of the GMM VMER models, where the validation curves surpass the training curves in the initial stages, demonstrating slight underfitting as shown in Table 17 (left). As stated before, the use of covariance type 'full' tends to result in overfitting models. After initial experiments, this option was removed due to the frequent errors that occurred when attempting to train the model with it. Problems emerged because the components presented ill-defined empirical covariance, resulting in an inability to fit the model. Nonetheless, due to the underfitting identified for models VMER, the covariance type is changed to 'full' while maintaining the remaining hyperparameters. By allowing each component to have its covariance matrix, the variance of the model increases and the bias decreases, overall reducing underfitting as represented by the learning curves of Table 17 (right). However, careful observation of the GMM GA with covariance type 'full' shows slight overfitting, with the training curve separating itself from the validation curve after a given point. Nonetheless, fluctuations in evaluation metrics show slight improvements in accuracy and Precision, yet a considerable decrease in Recall. The choice of whether or not these changes are acceptable is determined by the decision-maker's preferences.

Table 17 – Results of adjustments to GMM VMER model to fix underfitting

		Other covariance type	**Covariance type 'full'**
VMER	GMM RS		

		AUC = **0.9856**
		Accuracy = 0.9615
		Precision = 0.7452
		Recall = **0.9965**
			AUC = 0.9835
			Accuracy = **0.9632**
			Precision = **0.7715**
			Recall = 0.9597

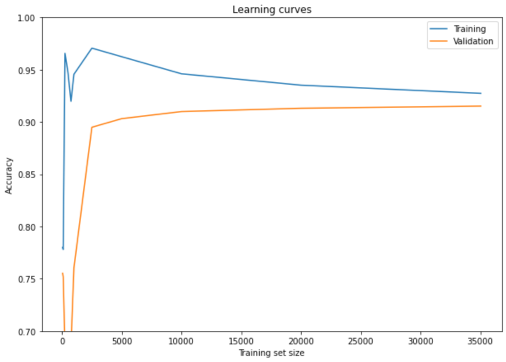
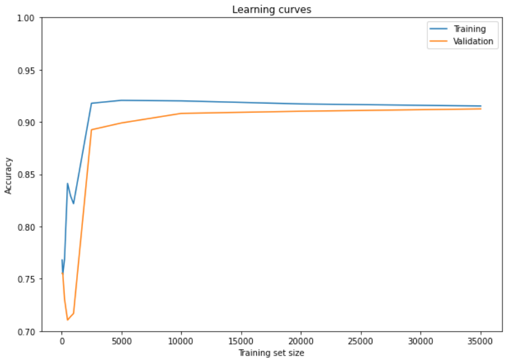
Table 17 – continuation

		Other covariance type	Covariance type 'full'
VMER	GMM GA		

		AUC = 0.9842
		Accuracy = 0.9613
		Precision = 0.7451
		Recall = **0.9939**
			AUC = **0.9884**
			Accuracy = **0.9619**
			Precision = **0.7558**
			Recall = 0.9765

The second situation identifies overfitting in model Gradient Boosting AEM, where a small gap between the two curves is not obtained and the accuracy of the training set is superior to the validation set. As stated in Section 5.3, large values of hyperparameters maximum depth and maximum features lead to overfitting, as well as low values of minimum samples leaf and minimum samples split. Therefore, to decrease the overfitting of the model, the following changes are made to these hyperparameters: maximum depth decrease from 3 to 2; maximum features decrease from 55 to 35; minimum samples leaf increase from 50 to 100; and minimum samples split increase from 5 to 50. These changes result in a learning curve plot with a significantly smaller gap between the curves. However, as shown in Table 18, the evaluation metrics obtained through 5-fold stratified cross-validation represent a decrease in performance with these changes. For this reason, these adjustments are not included in the final version of this model.

Table 18 – Results of adjustments to Gradient Boosting AEM model to fix overfitting

	Without adjustments	With adjustments
AEM		

	AUC = **0.9737**
	Accuracy = **0.9151**
	Precision = **0.8181**
	Recall = **0.8399**
		AUC = 0.9710
		Accuracy = 0.9124
		Precision = 0.8156
		Recall = 0.8299

ROC curves give a visual understanding of each model's capability to distinguish class 1 from class 0. The curves presented in Appendix I represent the empirical ROC curves obtained from plotting the true positive rate (Recall) versus the false positive rate for all possible cut-off values. Each point represents a different cut-off value and is connected to form the final curve. The same logic is applied for the Precision-Recall curves of Appendix J, where Precision is plot versus Recall and each cut-off value is connected to form the empirical curve. Although they share a parameter with ROC curves, Precision-Recall curves provide a different view, directed towards understanding each model's ability to predict the minority class, which corresponds to class 1.

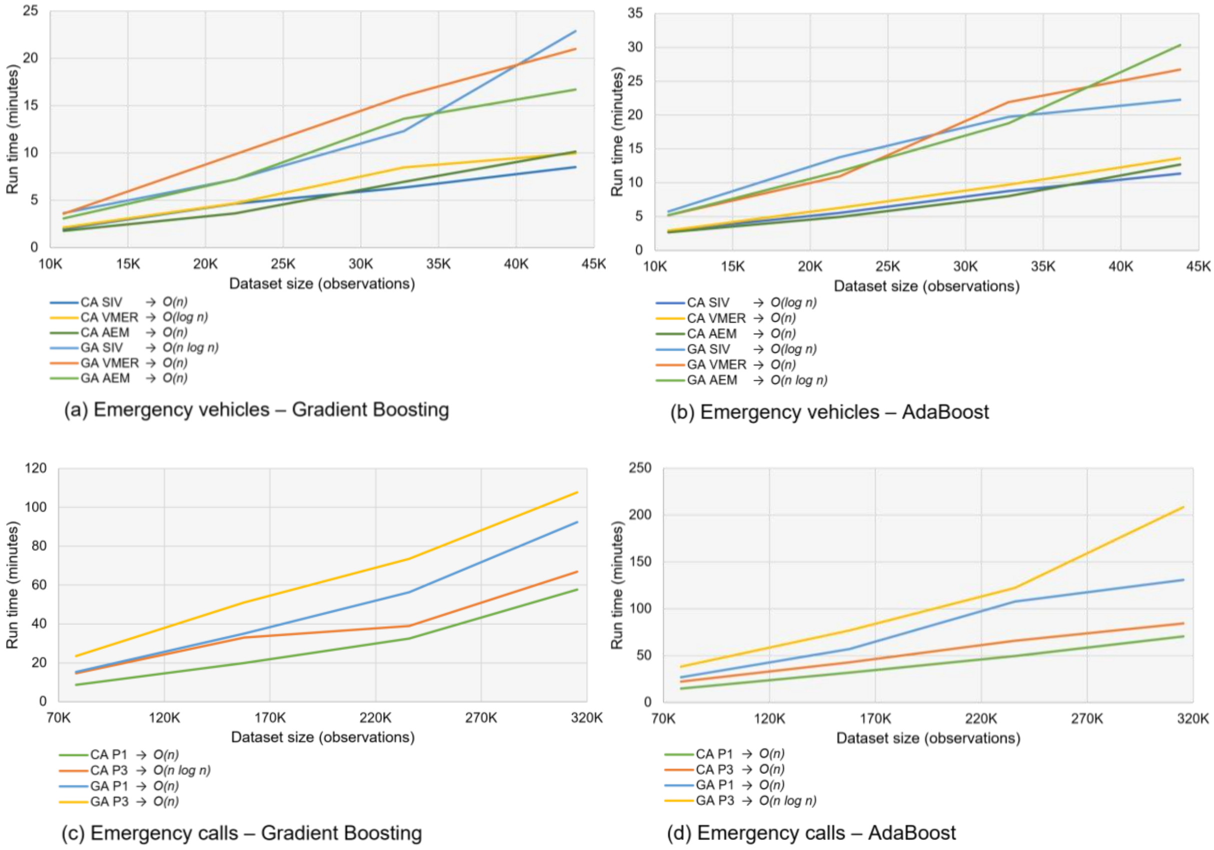
Random chance represents a straight diagonal line across the graph from point (0,0) to (1,1) in the case of ROC curves. Any curve that is drawn above this diagonal represents a model that is capable of distinguishing between the two classes better than random. The ideal curve is one that encompasses the entirety of the graph with an AUC of 1. The same is valid for Precision-Recall curves, although they are mirrored. The random chance diagonal travels from point (0,1) to (1,0) and the ideal curve is one that passes through point (1,1).

Both ROC curves and Precision-Recall curves are significantly worse for the two GMM models when compared to the two ML models, which is especially true for P1 and P3 models. Additionally, the results of the individual evaluation metrics for the four models, available in Appendix G, show that GMM models obtain lower results for the majority of the evaluation metrics. The standard deviation obtained from the stratified 5-fold cross-validation is lower than 0.1 for all cases, further solidifying the results. Overall, both Gradient Boosting and AdaBoost perform better than the GMM models for this data. However, the results from the GMM application are comparable to those of common ML models. GMM performs better than some ML algorithms currently in use in other applications and is a reasonable contender alongside them. Nonetheless, the selected evaluation measures show that it is outperformed by the boosting algorithms explored in this work.

5.4.2. Computational Complexity Evaluation

The big O notation is commonly used to study the performance of algorithms based on the size of the input data they are trained on, i.e., it is used to determine how an algorithm scales in complexity with increased data. For this purpose, the time behaviour of the algorithm is studied by analysing different training times depending on the input dataset size. This behaviour is represented by plotting the execution values of the algorithm by the number of observations in the dataset. The shape of this curve determines the computational complexity of the algorithm and whether it is easily adaptable to large volumes of data. From previous studies of the literature for other applications, ML algorithms typically follow a logarithmic curve, which in big O notation is represented by *O(log n)* (Hafeez et al. 2021). Although complexity analyses of ML algorithms are available in other works, this evaluation contributes by comparing the performance of the algorithms in combination with attribute selection methods. Further analyses of the GMM models are excluded since the previous performance analysis proved that they are surpassed by the boosting models.

The analysis is performed by evaluating the time required to train models Gradient Boosting and AdaBoost with RS hyperparameter tuning performed for 10 iterations, and with a 5-fold stratified cross-validation. A performance comparison is shown between using attribute selection methods CA and GA. Figure 20 shows the computational complexity for both vehicle and call models, which show similar results despite the differences in data sizes. The models with GA attributes have longer run times than CA due to the additional cost that this selection process represents. The big O notation is attributed to each model based on the shape of the curve. Although not all curves represent the expected *O(log n)*, the results do not show high model complexity. Compared to CA, the curves of GA models are further from the expected, which suggests that the algorithm follows a different complexity behaviour. Overall run time is superior for AdaBoost, which represents a significant advantage for Gradient Boosting models in comparison. Other than the average run time, significant differences between the models concerning computational complexity are not observed.



(a) Emergency vehicles – Gradient Boosting

(b) Emergency vehicles – AdaBoost

(c) Emergency calls – Gradient Boosting

(d) Emergency calls – AdaBoost

Figure 20 – Big O computational complexity

It is important to note that only four experiments of different dataset sizes were made, limiting the variations of the curves. Additionally, more complex models capable of proving a point of comparison are not incorporated. Overall, all models perform well and most of them prove to behave adequately with high volumes of data.

The values of the execution times obtained for each model are presented in Table 19. The CA results do not include the attribute selection process since it is almost immediate and does not increase the run time. These results show that although the AdaBoost models have longer run times for RS hyperparameter tuning, the GA attribute selection process is faster than for Gradient Boosting. The reason for this is unclear since both procedures require repeated training of the model. A noteworthy situation is the larger RS tuning times with the GA dataset compared to CA, which is mainly due to the larger number of attributes that are selected. CA selects 42 and 70 columns for vehicles and calls respectively, contrasting the average 76.58 and 96.75 columns selected by GA. The unexpected complexity behaviour of the GA is also observed in this table, where significant time increases are observed as well as inconsistencies such as occasional time decreases with additional data.

Table 19 – Run times in minutes with increasing dataset size

			Emergency vehicles			
			10,860 observations	21,900 observations	32,760 observations	43,800 observations
Gradient Boosting	CA	SIV	1.98**	4.63**	6.33**	8.50**
		VMER	2.13**	4.68**	8.47**	9.95**
		AEM	1.78**	3.63**	6.95**	10.13**
	GA	SIV	0.85* + 2.80**	1.50* + 5.70**	2.97* + 9.33**	8.28* + 14.58**
		VMER	0.48* + 3.10**	3.48* + 6.37**	5.20* + 10.82**	6.78* + 14.22**
		AEM	0.45* + 2.63**	1.65* + 5.57**	5.20* + 8.40**	4.48* + 12.20**
AdaBoost	CA	SIV	2.92**	5.52**	8.73**	11.32**
		VMER	2.92**	6.30**	9.72**	13.63**
		AEM	2.65**	4.92**	8.00**	12.70**
	GA	SIV	0.42* + 5.32**	1.12* + 12.67**	0.98* + 18.73**	1.90* + 20.37**
		VMER	0.35* + 4.85**	0.82* + 10.12**	1.12* + 20.80**	1.55* + 25.15**
		AEM	0.30* + 4.87**	0.90* + 10.87**	1.13* + 17.67**	2.15* + 28.17**
			Emergency calls			
			78,192 observations	157,680 observations	235,872 observations	315,360 observations
Gradient Boosting	CA	P1	8.65**	20.00**	32.58**	57.75**
		P3	14.73**	32.95**	38.93**	66.90**
	GA	P1	4.95* + 10.32**	11.22* + 23.82**	15.87* + 40.53**	34.88* + 57.53**
		P3	7.35* + 16.20**	17.07* + 33.95**	29.03* + 44.43**	44.45* + 63.28**
AdaBoost	CA	P1	14.83**	32.02**	49.73**	70.48**
		P3	22.40**	43.08**	65.87**	84.38**
	GA	P1	1.15* + 25.85**	4.73* + 52.35**	6.15* + 101.58**	3.88* + 126.87**
		P3	1.42* + 36.90**	5.43* + 71.58**	9.42* + 112.92**	9.67* + 198.77**

Time in minutes
* GA attribute selection
** RS tuning process

Despite the faster GA attribute selection process of the AdaBoost models, the overall running time of the Gradient Boosting models is inferior, deeming them the least computationally complex models out of the two. Regarding attribute selection methods, the GA presents significantly higher complexity when directly compared with CA, since the CA process is practically immediate.

5.4.3. Model Limitations

The limitations of the developed models are discussed in this section. These limitations are presented to the decision-maker, who takes them into consideration when determining whether or not to adopt the proposed models.

The GA used for the selection of the best subset of attributes converged within 5 generations since the individuals in the final population were identical. This same number of generations was used for the hyperparameter tuning processes, although not with the same results. The graphs available in Appendix C show the converging process of the algorithm for each generation. Although convergence is obtained for the majority of the models, this is not the case for all.

Despite the high crossover probability of this application which reduces the likelihood of convergence, the best course of action to avoid local optimal solutions is to increase the considerably low number of generations. The population size used is also low, which should facilitate convergence, further suggesting that decreasing crossover probability could result in a local optimum. However, increasing the number of generations is limited by the computational time of the process, as well as the need to run multiple models for each algorithm explored in this work. An added factor is that the computational time of an additional generation does not increase linearly, further complicating the identification of an appropriate number of generations. RS proved to be a strong alternative capable of achieving better results in less time. Having run the models on a personal computer, the GA with 5 generation took on average 2.3x longer than the RS with 10 iterations. This method has the advantage of not requiring convergence since it simply tests 10 combinations and returns the best one. Adding this to the fact that superior results were obtained for both Gradient Boosting and AdaBoost models, the RS is recommended as an overall better hyperparameter tuning process.

Despite the overall high accuracy of the models, issues related to data imbalance present a great challenge. The evaluation metrics of Appendix G show that Recall is especially poor for P1 models. Recall is lower than 0.5 regardless of the applied algorithm, meaning that the majority of positives (class 1) are predicted incorrectly. This is visible in the confusion matrices of Table 20, where in around 12.5% of cases class 0 is being predicted incorrectly for class 1 situations. The confusion matrices allow to quantify the disparity between the actual class of each observation and the class attributed by the predictor. Many classification metrics, such as Precision, Recall, and AUC derive from the confusion matrix and are calculated from the information that it provides. Positions [Class 0, Class 0] and [Class 1, Class 1] indicate correctly predicted occurrences, representing true negative and true positive situations, respectively. On the other hand, [Class 0, Class 1] and [Class 1, Class 0] represent false positive and false negative predictions, respectively. The final column shows the real total number of observations for each class, while the final line does the same for predictions. The accuracy of the model is observed in [Sum, Sum]. The remaining confusion matrices are available in Appendix K, and although the case of P1 is significantly problematic, high percentages of false negatives are also observed in other models. Low Recall is especially bad considering the nature of the service, where incorrect low demand predictions can lead to unpreparedness and failure of EMS resources. Since Precision-Recall is a trade-off, the nature of the service may favour sacrificing Precision to obtain higher Recall values, mainly because it would be better to incorrectly predict higher demand volumes than lower ones.

Table 20 – Confusion matrices of P1 models

	Class 0: [0,1] dispatches **Class 1: [2,15] dispatches**									
	Gradient Boosting					**AdaBoost**				
			Predicted Class					Predicted Class		
			Class 0	Class 1	Sum			Class 0	Class 1	Sum
P1	Actual Class	Class 0	71155 75.21%	3057 3.23%	74212 78.44%	Actual Class	Class 0	70945 74.99%	3267 3.45%	74212 78.44%
		Class 1	11778 12.45%	8618 9.11%	20396 21.56%		Class 1	11676 12.34%	8720 9.22%	20396 21.56%
		Sum	82933 87.66%	11675 12.34%	94608 84.32%		Sum	82621 87.33%	11987 12.67%	94608 84.21%

An approach to fixing this problem is to run attribute selection and hyperparameter tuning processes with the goal strategy of maximizing Recall. Although accuracy is likely to decrease, it may be beneficial since the likelihood of obtaining a class 0 prediction and having the true occurrence be class 1 will decrease. Another option is the selection of the ML algorithm based on this metric alone, which results in the exploration of the Naïve Bayes algorithm. However, it should be considered that accuracy and AUC results from this algorithm are significantly worse.

Note that Recall is bad regardless of the algorithm that is used, which suggests that this problem is related to the data used to train the models. Specifically, the low Recall values are likely to be related with the imbalance of the data. As mentioned previously, the percentage of class 0 occurrences is significantly higher than class 1. This leads to difficulties in the models' ability to correctly recognize and categorize the minority class. It is important to point out that the appropriate precautions were made when developing the models, such as the use of stratified cross-validation procedures to ensure that each class is represented. Nonetheless, the fundamental structure of the data implies a lack of representation due to a large number of observations in the original data with 0 demand. This deficit is related to the set granularity levels which, if reduced, would help overcome data sparsity. The exploration of larger time and/or spatial intervals is expected to increase the demand values of each observation and result in clusters with higher representation. This would overall contribute to reduce data imbalance and increase the model's capacity to recognize each class.

The final limitation of this work is the inability to quantify the improvements that INEM would obtain by changing their current predictive method to the proposed one. Although the advantages of the proposed models are objectively quantifiable, direct improvements are not estimable since no comparison is made. Nonetheless, since it is known that simple techniques are currently in use, it is presumed that these models will greatly improve the operations of the EMS system.

5.4.4. Final Remarks

The final results of the two boosting models, summarized in Table 21, show that Gradient Boosting presents higher values for the majority of the evaluation metrics than AdaBoost. Furthermore, based on all of the analyses that were performed, Gradient Boosting has the overall best performance out of the evaluated models, including lower computational complexity. For this reason, the recommended model is Gradient Boosting trained with attributes selected via GA and with hyperparameters tuned through RS. The selected attributes for this final model, as well as the hyperparameter values, are available in Appendix L.

Table 21 – Cross-validation results of the boosting models

		Gradient Boosting	**AdaBoost**
SIV	**AUC**	**0.9923**	0.9846
	Accuracy	**0.9834**	0.9774
	Precision	**0.7721**	0.6860
	Recall	**0.7454**	0.6534
VMER	**AUC**	0.9910	**0.9913**
	Accuracy	**0.9673**	0.9672
	Precision	0.8259	**0.8264**
	Recall	**0.8960**	0.8943
AEM	**AUC**	**0.9737**	0.9569
	Accuracy	**0.9151**	0.8933
	Precision	**0.8181**	0.7750
	Recall	**0.8399**	0.7949
P1	**AUC**	**0.8607**	0.8595
	Accuracy	**0.8436**	0.8425
	Precision	**0.7468**	0.7353
	Recall	0.4180	**0.4232**
P3	**AUC**	**0.9837**	0.9827
	Accuracy	**0.9616**	0.9599
	Precision	**0.8972**	0.8881
	Recall	**0.7396**	0.7314

Despite the low computational time of the recommended model, models trained with the CA dataset are capable of achieving higher efficiency, as shown in Section 5.4.2. Although the recommended GA Gradient Boosting model is appropriate for operational planning, CA models may be preferred if the main priority goal is computational cost minimization. Additionally, the CA selected less attributes in this application, which is an advantage to planners due to the practicality of needing fewer inputs to obtain predictions. Whether the small increase in model performance obtained from GA is worth the trade-off in terms of computational time is dependent on the decision-maker.

RS is identified as a better option for hyperparameter tuning when compared to GA in regard to both model performance and computational efficiency. Also, fewer obstacles are encountered by using RS since there is no need to ensure convergence.

Overall, the boosting models explored in this work outperformed other ML models for this set of highly sparse data. This includes the GMM, which performed similarly to other ML models and was ultimately surpassed.

The models developed in this work are directed towards operational planning on account of the fine granularity levels. Nonetheless, these models can be used for tactical and even strategic planning by grouping time buckets and/or spatial areas. This utilization requires, however, the availability of the attributes selected for the model. For example, the Gradient Boosting model requires a value for the expected average temperature as an input to obtain forecasts of P3 calls. If this model is used to predict demand concerning a period of time far in the future, the information regarding the anticipated average temperature is likely to be inaccurate. For this reason, the models directed towards tactical and strategic planning levels should only incorporate attributes with correct future information. This includes the main set of base attributes previously mentioned. These attributes concern time and spatial variables, and are indispensable to all models. Therefore, the models developed in this work are easily adaptable to other planning levels through the removal of attributes with unattainable or inaccurate projections.

5.5. Chapter conclusions

After data preparation and establishment of evaluation metrics, the GMM as well as eight ML algorithms are implemented. A filter and a wrapper method are used to identify the most relevant subset from the numerical attributes, and two tuning processes are used to designate values to the hyperparameters in order to maximize accuracy.

Through the study of evaluation metrics and other performance measures, existing issues in the obtained models are identified and solution proposals are provided. The computational complexity of the models is also evaluated, and their limitations are delineated. Finally, the best model to predict the call volumes and vehicle dispatch demand is pinpointed as the Gradient Boosting model trained with attributes selected through GA and with tuned hyperparameters via RS.

6. Decision Support Framework

This chapter aims to present a decision support framework based on prescriptive analytics that can be applied to EMS planning and be used at any planning level. Section 6.1 introduces data analytics and its components, Section 6.2 presents the proposed decision support framework, and Section 6.3 concludes the chapter.

6.1. Analytics

Data analytics, summarized in Figure 21, is commonly split into four sections, descriptive, diagnostic, predictive, and prescriptive analytics, although diagnostic is sometimes considered as the second part of descriptive analytics. The input required is historical knowledge in the form of data, and a sequence of proposed actions is generated as an output (Brandt et al. 2021).

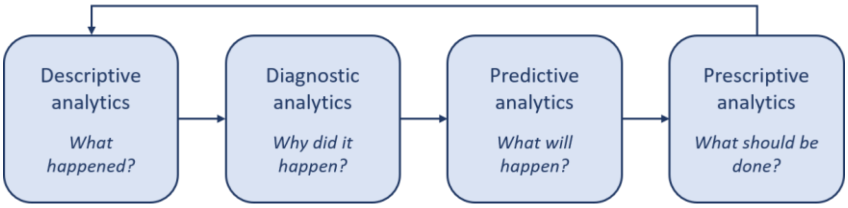


Figure 21 – Data analytics components

The aim of descriptive analytics is to identify and summarize what happened. Techniques such as data visualization and the use of key performance indicators allow decision-makers to assess performance and compare it against targets. The aim is to convert raw data into meaningful information by presenting data in summarized ways. This analysis mainly provides different views on collected data, and finds patterns and trends in this data. The output of descriptive analytics is commonly in the form of data visualization to summarize and report trends (Mustafee et al. 2018). Sequentially, diagnostic analytics follows the information obtained from data and intends to identify why something happened. The aim is to identify causes leading to the obtained performance, which includes understanding the impact of input factors and policies on performance measures. While diagnostic analytics transforms the gathered information from descriptive analytics into insights to aid managerial decisions, further insights are obtained in the form of foresight with predictive analytics. This level focuses on identifying what is likely to happen by developing estimates of outcomes based on planned inputs. Methods capable of supporting predictions such as data mining, forecasting, and mathematical approaches are used. The most common, forecasting, refers to methods used to predict events based on prior knowledge from historical data and other sources of information (Mustafee et al. 2018).

While descriptive, diagnostic, and predictive analytics are information focused, prescriptive analytics is directed towards decision-making. The focus of prescriptive analytics is obtaining prescriptions of specific actions that lead to the desired outcome, with an emphasis on the concrete decision problem. A solution path is suggested by prescribing one or more courses of action and informing on the likely outcome of each one. This is done by building what-if scenarios to determine cause-effect relationships leading to the best possible performance. Each of the generated alternatives is evaluated based on goal attainment in the search for a suitable course of action.

Prescriptive analytics has three core units that represent the methods by which data is translated into actions: simulation, optimization, and evaluation. The simulation unit is used to anticipate the consequences of unforeseen actions and determine the best course of action. The predicted scenarios obtained from predictive analytics are fed to the simulator that generates future conceivable scenarios according to the data, and creates a list of possible actions. Based on predefined goals, the evaluation unit then measures the simulated results, which are kept in storage. Lastly, the optimization unit improves the simulated scenarios based on well-defined objectives and existing possibilities. The final validation by the evaluation unit compares the first retrieved scenario with the optimized one and returns a sequence of actions as the final result. These actionable recommendations can be of various forms such as simple decision suggestions, proposed values for certain variables, or even complete constructed plans (Soltanpoor & Sellis 2016).

The success of prescriptive analytics is mainly dependent on the assessment of the alternatives generated based on the results from the prediction phase, and the impact they have on performance. Nonetheless, the entire process is a cycle, meaning that after the proactive decision and action implementation, events occur and descriptive analysis once again analyses what happened as well as why. New forecasts are then required to incorporate recent developments and utilize new data to improve estimates.

Although each level of analytics can be used individually, a full understanding of demand behaviour and informative insights to support decision-making are best obtained when the four levels of analytics are explored together. As previously stated, the purpose of this work is the development of reliable forecasting tools capable of assisting the EMS provider in resource planning. This goal in itself simply fulfils the predictive part of the analytics. This element, although a fundamental input for planning problems, is most useful when combined with the remaining analytics, giving rise to a definitive decision oriented support tool. Considering prescriptive analytics in the context of EMS planning, each decision problem requires a concrete plan to determine the most suitable actions.

6.2. Framework Proposal

The decision support framework presented in this section is applied to the development of predictive models oriented to EMS demand, corresponding to the predictive analytics module. The studies available in the literature, presented in Chapter 3.3, propose various predictive models for different situations. However, no objective framework has been presented capable of directing and identifying which models are best suited for which problems. The approach presented in this framework is based on the fact that the success and utility of a model depends on the planning level for which it is used. Therefore, the first step is identifying the required level of forecasting for what is intended.

A summary of the decisions made at each planning level has been previously presented in Table 2, and this topic has been discussed in Chapter 3.2. This table demonstrates the different needs of each level in terms of forecasts, which are also dependent on the time horizon of decision-making.

Aspects such as data volume, data aggregation level, and model running time are defining to a predictive model's success in any given planning level. The quantity of available data limits the level of planning for which forecasts can be obtained. This means that an important step when establishing the forecasting level is identifying the available data volume. If data is only available for a few months, then strategic and tactical level forecasting cannot be achieved. This is because sufficient representation of demand behaviour is not available for the corresponding planning horizon. The level of data aggregation is also an important factor regarding forecasting levels. If the available data is grouped in large time intervals, then it does not provide sufficient detail for forecasting at low planning horizons. Weekly forecasts, although valuable for tactical planning, do not provide value for planning on the operational level. The same is true for yearly forecasts that are only suitable for strategic planning. Finally, the computational efficiency of predictive models is of paramount importance. This is related to the frequency of data updates and the periodicity that new models are trained. If recent data is repeatedly incorporated and new models are consistently trained, computational efficiency is of great value. On the other hand, if a single model is trained for a long period of time, it is feasible to consider a more accurate yet lengthier training process. This is typically the case for higher planning levels where, for example, if monthly forecasts are employed, model updates are only required once a month, while daily updates prioritize computational speed.

The proposed framework is intended to generate EMS demand predictions capable of assisting managerial planning decisions and be used in descriptive analytics. It is important to note that although this framework is directed towards demand forecasting, it is also applicable to service time forecasting. These types of forecasts are mainly pertinent for strategic and tactical planning, to estimate travel times for ambulance location decisions. Additionally, the literature on predictive demand models mostly focuses on predicting calls, however, there is an important need to develop predictions for vehicle forecasting. The model building procedure, however, is similar, and this framework is valid for both types of estimates.

6.2.1. Predictive Analytics

The purpose of predictive analytics is determining the number of calls or vehicles that will be received by the EMS provider in a given time interval and for a given location. There are multiple ways these predictions can be obtained, and a review of the available works in the literature has been presented in Chapter 3.3. The purpose of this section is to structure and systematise the options and recognize the applicability of each one.

Regardless of the prediction task, which can be classification or regression, predictors such as time interval, date, and location are common throughout any model. Other attributes can be added, although they should be carefully selected as they could add useless information and increase computational time. There is no consensus on which attributes should be included in EMS predictive models, as each study uses different ones (Lowthian et al. 2011; Ho Ting Wong & Lin 2020). Factors such as weather, age, gender, etc., have been proven to impact EMS demand, and this issue has been discussed in Chapter 3. Nonetheless, available attributes should be submitted to an attribute selection process to determine the most relevant ones. The relationship between target variables and predictive attributes can be studied in multiple ways, however, this study explores the contrasts between CA and GA. CA is significantly faster regarding computational time, although it can result in inferior performance. On the other hand, if different algorithms are applied, the GA must be repeatedly trained for each one. If computational time is a valuable aspect in the training of the model, as is the case in operational planning, CA is recommended. On the other hand, if the aim is to maximize performance with little regard to computational efficiency, GA is a valid option. These recommendations consider only the methods that were explored in this work. However, this same logic applies to other methods: wrapper methods typically provide better results, while the computational burden of filter methods is insignificant with small accuracy deterioration (Venkatesh & Anuradha 2019). Also, although it is not the case for this application, filter methods may tend to select larger subsets of attributes (Y. Zhang et al. 2014). The use of hybrid methods is advantageous in these cases since the execution time of the wrapper applied on the reduced dataset is faster. Unlike filter methods, hybrid methods also suffer from lack of generality, same as wrapper and embedded methods (Guerra-Manzanares et al. 2019). On the other hand, embedded methods typically select a subset of attributes intrinsically while training the model, resulting in faster execution times that wrapper and hybrid methods (Yan & Zhang 2015). It represents a good option if available and feasible for the chosen algorithm since it is computationally inexpensive and less prone to overfitting than wrapper methods (Jain & Singh 2018).

The second aspect explored in this work is hyperparameter tuning. Regardless of the selected algorithm, different parameter values should be explored to achieve the highest-performing model. RS and GA are compared in this work, and RS proved to be a superior method due to it resulting in better performing models and having inferior computational time. The selection of the number of iterations for a RS is dependent on the number of hyperparameters and values to test, as well as the available time to search for an efficient combination. If run time is not an important factor, for instance in strategic planning, RS with a large number of iterations should be run. However, while RS is most appropriate for large search spaces due to the trade-off between run time and quality of the solution, Grid Search should be used for small search spaces since it explores all possible combinations. Similar conclusions are reached by Liashchynskyi & Liashchynskyi (2019), who compare the three classic hyperparameter tuning methods (Grid Search, RS, and GA) for convolutional neural networks. They do not recommend Grid Search for large search spaces, and observe that RS achieves good results faster. Their results show similar accuracy for the GA compared to RS, though with longer run times. Additionally, Bergstra & Bengio (2012) show that RS has the same practical advantages of Grid Search, trading small efficiency reduction in low dimensional search spaces for large improvements in high dimensional spaces. On the other hand, Wainer & Fonseca (2021) compare eighteen search algorithms for SVM and do not identify significant differences among the procedures to select the best set of hyperparameters. They conclude that it is not worth using very precise and computationally expensive searching algorithms since it likely does not result in significant performance improvements.

Regarding potential forecasting methods, the multiple regression models explored in the literature are fast, accurate, and easily applicable. Recent regression models incorporate both time and spatial factors (Cramer et al. 2012; Steins et al. 2019), and can easily include factors such as age, gender, and weather (Lowthian et al. 2011; Wong & Lai 2010). The contribution of these models is substantial in identifying variables that influence demand and that should be incorporated in forecasting models. However, the precision of regression models is often low, and errors are common. Although these models have been addressed in the literature recently, they are frequently not adopted in practice due to the availability of superior alternatives. Despite their inability to capture complex data structures, they may be appropriate for estimates at large scales where detailed forecasts are not required. In practice, time series models have mostly surpassed regression models due to their superior performance and ease of use. The literature has explored ARIMA and Holt-Winters methods to predict EMS demand, as well as a non-parametric technique, SSA. The results of Al-Azzani et al. (2020) show that ARIMA models provide the most accurate forecasts for tactical level planning horizons, while SSA is best for long-term strategic planning horizons. Additionally, they show that both these methods perform better than Holt-Winters. The success of SSA in producing accurate long-term forecasts is also recognized by Vile et al. (2012). Seasonality has been recently incorporated in ARIMA models, making SARIMA models an improved option for tactical forecasting (Gijo & Balakrishna 2016).

The modelling of both time and location is possible with spatial-temporal models. Setzler et al. (2009) obtain accuracy improvements at fine time scale and low spatial granularity by applying ANN, however, this method is most appropriate for tactical and strategic level forecasts due to the associated computational cost. Additionally, large volumes of data are required, making this method unsuitable for operational planning. Zhou (2016) explores GMM, Kernel Density Estimation, and Kernel Warping for operational forecasting. Only 4-8 weeks of historical data are used, and the estimates are obtained at fine time and location scales. Out of the three methods, Kernel Warping proves to be the most accurate method, although closely followed by GMM. The GMM is easily applicable with only a slight decrease in accuracy and is identified as the most promising spatial-temporal model for operational forecasting. For this reason, this dissertation implements the GMM with the goal of obtaining reliable and accurate operational forecasts. Nonetheless, a large data volume of two years is used, making these forecasts suitable for other planning levels. Additionally, eight ML algorithms are implemented, and their results compared with the aim of identifying other potential algorithms capable of providing accurate estimates. The computational efficiency of the tested models, with the exception of SVM, is sufficient to justify their application for operational planning. The explored boosting models (Gradient Boosting and AdaBoost) prove to be the most accurate and provide superior results to GMM. Additionally, a selection of evaluation methods identifies the Gradient Boosting model as the overall superior model out of the ones tested in this work, making it the recommended model for spatial-temporal operational forecasting. Nonetheless, AdaBoost demonstrates great performance and should be considered if multiple models are explored. Overall, boosting algorithms show great potential and are recommended for similar applications.

6.2.2. Prescriptive Analytics

After relevant descriptive and diagnostic analyses through the use of visualization tools, predictive analytics should be applied following the guidelines presented in the previous section, which are dependent on the planning level of the decision problem. The predictions are fed to prescriptive analytics tools such as simulation, resulting in a clear solution path. This process represents the framework that should be followed to solve a decision problem, and is represented in Figure 22. The input data is used at all stages of this framework and the ultimate results, actionable recommendations, are used to assist the original decision problem.

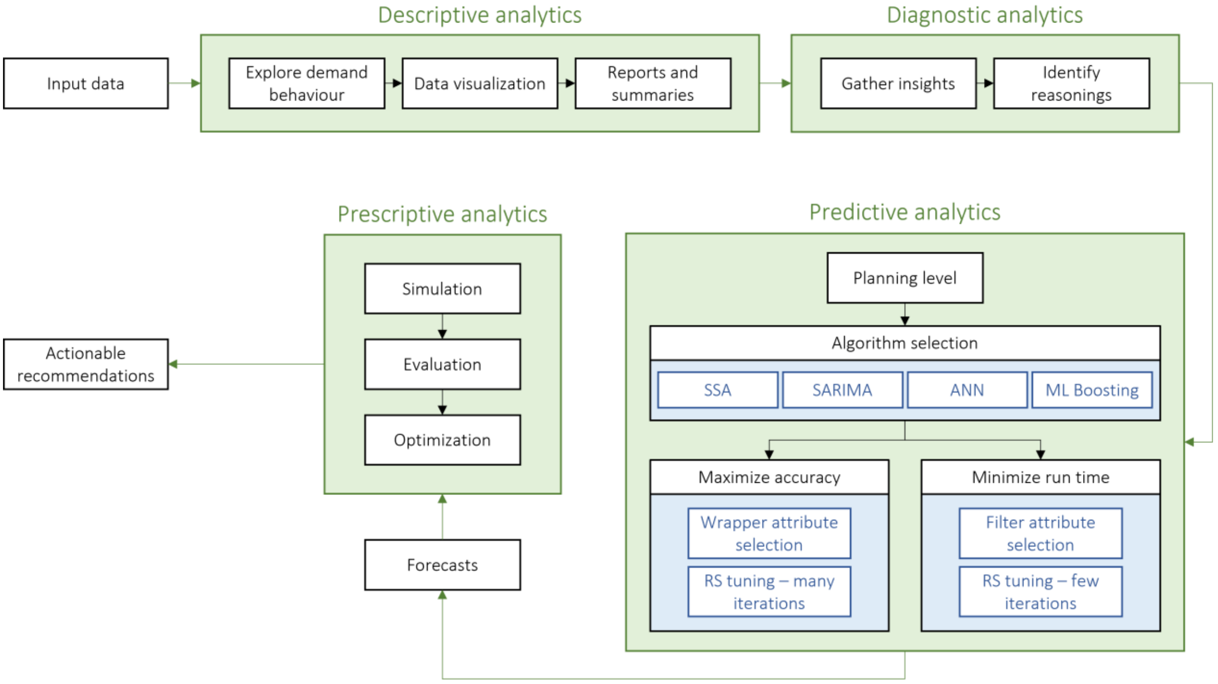


Figure 22 – A guideline for descriptive, diagnostic, predictive, and prescriptive analytics

A prescriptive analytics framework directed to support decision problems includes a full data analytics study. The decisions towards which the analyses are directed have an impact on how the analyses are conducted. While descriptive analytics aims to understand demand behaviour, data collected by EMS providers is vast and can be examined from multiple perspectives. The content of a descriptive analysis is dependent on its purpose, which is also the case for diagnostic analytics. This second analysis is crucial to fully understand and find explanations as to why demand is behaving a certain way. Correlations between variables must be interpreted to discover ways of improving operations and optimizing resources. This interpretation involves recognizing relationships and dependencies between occurrences, and identifying their relevance and impact on the subject under analysis.

Predictive analytics is used to produce predictions of future demand. The value of this type of information derives from its utility in decision-making. Depending on the decision problem, forecasts referent to different time horizons are constructed, and obtained in different run times. There must be a clear understanding of the purpose of the estimates to maximize their benefit. The previous section provides guidelines to produce accurate forecasting models. Nonetheless, it is not possible to identify and recommend a single method that is superior to all others and successful for all cases. Each forecasting model has unique characteristics making different models appropriate for different situations. While complex models tend to achieve higher results, simple models typically have the advantage of speed, ease of application, and fewer inputs, making them useful for situations where information is scarce. Nonetheless, if a forecasting model is surpassed in all measures by another, it is replaced and ultimately falls into disuse. From reviewing the methods applied in the literature, the most promising for EMS forecasting are identified as SSA, SARIMA, and ANN (Al-Azzani et al. 2020; Gijo & Balakrishna 2016;

Setzler et al. 2009). While ANN is appropriate for both tactical and strategic forecasts, Al-Azzani et al. (2020) recommend ARIMA and SSA models for tactical and strategic planning levels, respectively. For operational planning, the results of this work suggest ML boosting algorithms over GMM for highly imbalanced data.

Through prescriptive analytics, the insights obtained from the various stages of data analytics are translated into concrete and actionable recommendations to be used for management purposes. The guidelines represented in Figure 22 apply to any EMS decision problem to obtain proposals for decision problems.

6.3. Chapter conclusions

Data analytics receives collected data and produces recommendations of actions to follow based on the decision problem. The first step, descriptive analytics, conducts a thorough analysis of the available data and gains insights on demand behaviour through data visualization tools. The second step, diagnostic analytics, aims to find reasoning behind the identified data behaviour. The third, predictive analytics, produces estimates of future demand, which are fed to the fourth and final step, prescriptive analytics. Here, potential scenarios are considered, and the best ones are returned as suggestions to the decision-maker.

This chapter presents a decision support framework for predictive analytics, consisting of directions leading to accurate models appropriate for different situations. Additionally, a prescriptive analytics guideline is proposed, incorporating and summarizing the predictive recommendations.

7. Conclusions and Future Work

EMS systems are complex structures vital to preserving human lives and delivering fast and effective health care to the population. Demand for EMS has been increasing largely due to population ageing and growth, resulting in a need to evaluate and study these systems. Reliable forecasting tools capable of supporting planning decisions are required to optimize resource allocation and improve effectiveness.

Short-term demand forecasts are a vital input for operational planning, and accurate estimates obtained in low computational times are needed for detailed planning on daily, hourly, and real time basis. Currently, INEM uses simple averaging techniques to obtain future demand provisions, which does not meet with the forecasting techniques addressed in the state-of-the-art. This dissertation focuses not only on the development of a highly accurate forecasting model, but on the comparison of different models to find the highest performing one for the data. For the problem of operational planning, an extensive search of models applied in the literature for EMS demand forecasting allowed the identification of the time-varying GMM as the most promising option. This model's accuracy and computational efficiency are appropriate for short-term planning, and it presents low complexity considering its high performance. Further research showed that common ML algorithms, frequently applied for predictive modelling in a variety of other problems, have not yet been directly applied to predicting EMS demand. Recognizing this, a wide selection of ML models is chosen to train and validate on real data shared by INEM, and allow a comparison with the selected GMM. The models are explored to incorporate additional attributes in order to explain demand fluctuations. The standardized implementation process of ML algorithms is adopted as the computational methodology of this work. In addition to training and testing predictive models on INEM's data, optimization methods such as attribute selection and hyperparameter tuning processes are explored. This research aims at identifying the best improvement techniques and achieving high performance within reasonable computational time. Unlike the previous application of GMM in the literature, this work considers the problem as classification, allowing extensive evaluation of the obtained models. Additionally, while the literature mainly focuses on the prediction of call volumes, this work tackles the issue of estimating both call volumes and emergency vehicle demand.

The dataset referent to call volumes is grouped by hourly time intervals and districts in mainland Portugal. On the other hand, the dataset referent to vehicle dispatches is separated by 8-hour shifts and origin bases of the vehicles in the municipality of Lisbon. These are the defined granularity levels, which are the highest available levels since operational forecasts on fine time and spatial scales are required. An in-depth analysis of the historical data from 2017-2018 identifies demand patterns similar to those already recognized in the literature. These include relationships between demand and times when people sleep and go to work, as well as the month of the year likely related to vacation periods. The provided datasets are used to train and validate the models and experiment multiple improvement opportunities. The attribute selection processes identified both CA and GA as appropriate methods, the former having significantly lower computational time. Alternative hyperparameter tuning procedures were investigated and the highest improvements were obtained with the RS method. This method surpasses the explored alternatives both in results and computational burden, as well as ease of utilization. The results obtained from a stratified 5-fold cross-validation recognized Gradient Boosting as the best model out of those that were explored in this dissertation, closely followed by AdaBoost. The GMM model achieved results similar to those of other ML models, although it was surpassed by both of the explored boosting models. A noteworthy remark is that while Gradient Boosting performed best with a large subset of attributes, the opposite is true for AdaBoost, having achieved greater results with fewer attributes.

This dissertation set out to improve the forecast demand process for EMS using the case study of INEM to analyse EMS demand behaviour and develop a forecasting tool capable of accurately modelling this behaviour. Although no baseline was considered in this work due to the unavailable averaging techniques currently in use by INEM, the dynamic behaviour of EMS call and vehicle demand is highlighted. This behaviour suggests poor performance of simple averaging measures and inaccurate estimates obtained through the use of these techniques. Although improvements cannot be explicitly proven, models capable of capturing both spatial and temporal aspects of demand are necessary to adequately contribute to the system's effective operations. This work focused not only on the comparison of potential forecasting models, but also on the delimitation of the best optimization procedures to obtain accurate and fast forecasts. This was achieved by identifying the Gradient Boosting model as the one with highest performance, with a subset of attributes selected via GA and having tuned its hyperparameters through a RS procedure. Nonetheless, several limitations are associated with this model. The target variables were clustered via K-means into two classes, giving rise to a binary classification problem. Two limitations arise from this clustering process, concerning the small number of classes and the range of each class. Although the selection of the number of classes was performed according to a legitimate method, it resulted in a binary problem which limits the amount of information provided by the model. Similarly, the range of values in each class was defined through a renowned clustering algorithm. Nonetheless, these classes may be of little use to the decision-maker, who may prefer to delimit these intervals autonomously. Although the models developed in this dissertation were directed towards operational planning at the highest granularity level, data sparsity presented a great challenge that led to the obtainment of imbalanced data and class representation issues. These problems resulted in some models with low Recall due to difficulties in distinguishing the class with higher demand. This issue can greatly impact the quality of the service since it results in the prediction of frequently incorrect low demand volumes. Finally, the proposed and recommended model is directed towards assisting decision-makers with planning operations. This means that the ultimate judgement of whether the model is appropriate for this task is determined by the decision-maker. Although the model was the subject of multiple evaluation procedures, having clearly outlined its perks and limitations, the choice of whether or not to use this model is up to the decision-maker. Nonetheless, the suggestions and conclusions presented in this work are based on the studies that were here conducted, as well as previous literature findings, with clear delineation of the steps and reasoning that were considered.

To overcome the limitations identified in this work, future classification problems should explore the delimitation of the number of classes and their ranges according to the choice of the decision-maker rather than using unsupervised learning techniques. Although the chosen method demonstrates good results, the exploration of other methods for class definition is recommended to produce better performing models, whilst validated by the decision-maker. Nonetheless, future work should attempt to further explore boosting ML algorithms considering the problem as regression instead of classification. The values used for the application of the various methods applied in this work as well as the values selected for the tuning of the hyperparameters of the models were chosen in an empirical manner. On the other hand, a more methodical approach could prove to be a superior alternative capable of easily identifying the best values. Future work should also focus on identifying improved ways to delineate granularity levels. The degree of data sparsity should be closely studied to help define the most appropriate temporal and spatial granularity for the desired planning level. Lower granularity levels may be preferred if sufficient demand representation is not present, though their utility should be approved by the planner. To further combat data sparsity, the distinction of calls by severity can be removed by aggregating total call demand as is commonly done in the literature. Once again, this option must be approved by the decision-maker to understand whether this distinction is required for planning. Furthermore, location-specific and temporal seasonality were not incorporated in this application and represent an interesting addition for future applications of ML. Once a model is approved by the decision-maker for utilization in the context of operational forecasting, the creation of an interface to facilitate the insertion of data and the obtainment of predictions may be of great value to the EMS provider. Finally, the elaboration of a simulation model as an extension of this work could help evaluate the impact of the model's errors and identify improvement opportunities.

In summary, the main recommendation from this work is the exploration of ML boosting models incorporating location-specific and temporal seasonality. Overall, closer collaboration with planners regarding the definition and validation of relevant aspects, such as the range of call volumes in each cluster, is surely advantageous to ensure that the obtained predictions are appropriate for the planning decisions to which they are directed. Nonetheless, the contributions of this work include the application of state-of-the-art methods to develop accurate predictions of EMS demand. This includes the application of ML algorithms not yet considered in the literature for the problem of EMS demand forecasting. Furthermore, this work contributes by comparing various optimization methods and their performance exploring real data, and presenting the associated computational complexity of the applied models.

References

Al-Azzani, M. A. K., Davari, S., & England, T. J. (2020). An empirical investigation of forecasting methods for ambulance calls - a case study. *Health Systems*, *00*(00), 1–18. https://doi.org/10.1080/20476965.2020.1783190

Al-Janabi, M., De Quincey, E., & Andras, P. (2017). Using supervised machine learning algorithms to detect suspicious URLs in online social networks. *Proceedings of the 2017 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2017*, 1104–1111. https://doi.org/10.1145/3110025.3116201

Aldrich, C. A., Hisserich, J. C., & Lave, L. B. (1971). An analysis of the demand for emergency ambulance service in an urban area. *American Journal of Public Health*, *61*(11), 2158–2161. https://doi.org/10.2105/AJPH.61.11.2158

Aringhieri, R., Bruni, M. E., Khodaparasti, S., & van Essen, J. T. (2017). Emergency medical services and beyond: Addressing new challenges through a wide literature review. *Computers and Operations Research*, *78*(July 2015), 349–368. https://doi.org/10.1016/j.cor.2016.09.016

Arunkumar, A., Ramkumar, R. K., Venkatraman, V. V., Abdulhay, E., Lawrence Fernandes, S., Kadry, S., & Segal, S. (2017). Classification of focal and non focal EEG using entropies. *Pattern Recognition Letters*, *94*, 112–117. https://doi.org/10.1016/j.patrec.2017.05.007

Ashfaq, R. A. R., Wang, X. Z., Huang, J. Z., Abbas, H., & He, Y. L. (2017). Fuzziness based semi-supervised learning approach for intrusion detection system. *Information Sciences*, *378*, 484–497. https://doi.org/10.1016/j.ins.2016.04.019

Baker, J. R., & Fitzpatrick, K. E. (1986). Determination of an optimal forecast model for ambulance demand using goal programming. *Journal of the Operational Research Society*, *37*(11), 1047–1059. https://doi.org/10.1057/jors.1986.182

Bélanger, V., Ruiz, A., & Soriano, P. (2019). Recent optimization models and trends in location, relocation, and dispatching of emergency medical vehicles. *European Journal of Operational Research*, *272*(1), 1–23. https://doi.org/10.1016/j.ejor.2018.02.055

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, *13*, 281–305.

Bins, J., & Draper, B. A. (2001). Feature selection from huge feature sets. *Proceedings of the IEEE International Conference on Computer Vision*, *2*, 159–165. https://doi.org/10.1109/ICCV.2001.937619

Bokulich, N. A., Kaehler, B. D., Rideout, J. R., Dillon, M., Bolyen, E., Knight, R., Huttley, G. A., & Gregory Caporaso, J. (2018). Optimizing taxonomic classification of marker-gene amplicon sequences with QIIME 2's q2-feature-classifier plugin. *Microbiome*, *6*(1), 1–17. https://doi.org/10.1186/s40168-018-0470-z

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A Training Algorithm for Optimal Margin Classifiers. *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*.

Brandt, T., Wagner, S., & Neumann, D. (2021). Prescriptive analytics in public-sector decision-making: A framework and insights from charging infrastructure planning. *European Journal of Operational Research*, *291*(1), 379–393. https://doi.org/10.1016/j.ejor.2020.09.034

Breiman, L. (1996). Bagging predictors. *Machine Learning*, *24*, 123–140. https://doi.org/10.3390/risks8030083

Breiman, L. (1997). *ARCING THE EDGE Leo Breiman Technical Report 486 , Statistics Department University of California, Berkeley CA. 94720*. *4*, 1–14. https://pdfs.semanticscholar.org/65b7/b1a0d61fd012f10cfce642d4aa4dec9a5829.pdf

Breiman, L. (2001). Random forests. *Machine Learning*, *45*, 5–32. https://doi.org/10.1201/9780429469275-8

Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., & Elhadad, N. (2015). *Intelligible Models for HealthCare*. 1721–1730. https://doi.org/10.1145/2783258.2788613

Channouf, N., L'Ecuyer, P., Ingolfsson, A., & Avramidis, A. N. (2007). The application of forecasting techniques to modeling emergency medical system calls in Calgary, Alberta. *Health Care Management Science*, *10*(1), 25–45. https://doi.org/10.1007/s10729-006-9006-3

Chen, A. Y., Lu, T. Y., Ma, M. H. M., & Sun, W. Z. (2016). Demand Forecast Using Data Analytics for the Preallocation of Ambulances. *IEEE Journal of Biomedical and Health Informatics*, *20*(4), 1178–1187. https://doi.org/10.1109/JBHI.2015.2443799

Chen, W., Xie, X., Wang, J., Pradhan, B., Hong, H., Bui, D. T., Duan, Z., & Ma, J. (2017). A comparative study of logistic model tree, random forest, and classification and regression tree models for spatial prediction of landslide susceptibility. *Catena*, *151*, 147–160. https://doi.org/10.1016/j.catena.2016.11.032

Clark, P., & Niblett, T. (1989). The CN2 induction algorithm. *Machine Learning*, *3*(4), 261–283. https://doi.org/10.1007/bf00116835

Cover, T. M., & Hart, P. E. (1967). Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory*, *13*(1), 21–27. https://doi.org/10.1109/TIT.1967.1053964

Cramer, D., Brown, A. A., & Hu, G. (2012). Predicting 911 calls using spatial analysis. *Studies in Computational Intelligence*, *377*(January 2011), 15–26. https://doi.org/10.1007/978-3-642-23202-2-2

Degel, D., Wiesche, L., Rachuba, S., & Werners, B. (2015). Time-dependent ambulance allocation considering data-driven empirically required coverage. *Health Care Management Science*, *18*(4), 444–458. https://doi.org/10.1007/s10729-014-9271-5

Dick, W. F. (2003). Anglo-American vs. Franco-German Emergency Medical Services System. *Prehospital and Disaster Medicine*, *18 (01)*, 29–37. https://doi.org/https://doi.org/10.1017/S1049023X00000650

Erickson, B. J., Korfiatis, P., Akkus, Z., & Kline, T. L. (2017). Machine learning for medical imaging. *Radiographics*, *37*(2), 505–515. https://doi.org/10.1148/rg.2017160130

Freund, Y., & Schapire, R. E. (1997). A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, *55*(1), 119–139. https://doi.org/10.1006/jcss.1997.1504

Friedman, J. H. (1999). *Greedy Function Approximation: A Gradient Boosting Machine*.

Gauthama Raman, M. R., Somu, N., Kirthivasan, K., Liscano, R., & Shankar Sriram, V. S. (2017). An efficient intrusion detection system based on hypergraph - Genetic algorithm for parameter optimization and feature selection in support vector machine. *Knowledge-Based Systems*, *134*, 1–12. https://doi.org/10.1016/j.knosys.2017.07.005

Géron, A. (2019). *Hands-on Machine Learning with Scikit-Learn , Keras & TensorFlow*.

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, *63*(1), 3–42. https://doi.org/10.1007/s10994-006-6226-1

Gijo, E. V., & Balakrishna, N. (2016). SARIMA models for forecasting call volume in emergency services. *International Journal of Business Excellence*, *10*(4), 545–561. https://doi.org/10.1504/IJBEX.2016.079252

Guerra-Manzanares, A., Bahsi, H., & Nomm, S. (2019). Hybrid feature selection models for machine learning based botnet detection in IoT networks. *Proceedings - 2019 International Conference on Cyberworlds, CW 2019*, 324–327. https://doi.org/10.1109/CW.2019.00059

Guerriero, F., & Guido, R. (2011). Operational research in the management of the operating theatre: A survey. *Health Care Management Science*, *14*(1), 89–114. https://doi.org/10.1007/s10729-010-9143-6

Hafeez, M. A., Rashid, M., Tariq, H., Abideen, Z. U., Alotaibi, S. S., & Sinky, M. H. (2021). Performance improvement of decision tree: A robust classifier using tabu search algorithm. *Applied Sciences (Switzerland)*, *11*(15). https://doi.org/10.3390/app11156728

Harvey, A., & Oryshchenko, V. (2012). Kernel density estimation for time series data. *International Journal of Forecasting*, *28*(1), 3–14. https://doi.org/10.1016/j.ijforecast.2011.02.016

Ho, T. K. (1995). Random Decision Forests Tin Kam Ho Perceptron training. *Proceedings of 3rd International Conference on Document Analysis and Recognition*, 278–282.

Hong, H., Liu, J., Bui, D. T., Pradhan, B., Acharya, T. D., Pham, B. T., Zhu, A. X., Chen, W., & Ahmad, B. Bin. (2018). Landslide susceptibility mapping using J48 Decision Tree with AdaBoost, Bagging and Rotation Forest ensembles in the Guangchang area (China). *Catena*, *163*(July 2017), 399–413. https://doi.org/10.1016/j.catena.2018.01.005

INEM. (2013). *Sistema Integrado de Emergência Médica*. https://www.inem.pt/wp-content/uploads/2017/06/Sistema-Integrado-de-Emergência-Médica.pdf

INEM. (2017). *Plano Estratégico 2017-2019: Aplicação da metodologia Balanced Scorecard*. https://www.inem.pt/category/transparencia/instrumentos-de-gestao/

INEM. (2018). Relatório Anual. In *Relatório anual de atividades e contas*. https://www.inem.pt/wp-content/uploads/2019/07/Relatório-Anual-Atividades-e-Contas-de-2018-2.pdf

INEM. (2019). Relatório de Atividade. In *Relatório de Atividade dos meios de emergência médica*. https://www.inem.pt/wp-content/uploads/2020/05/Relatório-Meios-de-Emergência-Médica-2019.pdf

Ingolfsson, A. (2013). EMS planning and management. *International Series in Operations Research and Management Science*, *190*, 105–128. https://doi.org/10.1007/978-1-4614-6507-2_6

Jain, D., & Singh, V. (2018). Feature selection and classification systems for chronic disease prediction: A review. *Egyptian Informatics Journal*, *19*(3), 179–189. https://doi.org/10.1016/j.eij.2018.03.002

Janet, J. P., & Kulik, H. J. (2017). Resolving Transition Metal Chemical Space: Feature Selection for Machine Learning and Structure-Property Relationships. *Journal of Physical Chemistry A*, *121*(46), 8939–8954. https://doi.org/10.1021/acs.jpca.7b08750

Kalantari, M. (2021). Forecasting COVID-19 pandemic using optimal singular spectrum analysis. *Chaos, Solitons and Fractals*, *142*, 110547. https://doi.org/10.1016/j.chaos.2020.110547

Kamenetzky, R. D., Shuman, L. J., & Wolfe, H. (1982). Estimating Need and Demand for Prehospital Care. *Operations Research*, *30*(6), 1148–1167. https://doi.org/10.1287/opre.30.6.1148

Kapoor, A., Ben, X., Liu, L., Perozzi, B., Barnes, M., Blais, M., & O'Banion, S. (2020). *Examining COVID-19 Forecasting using Spatio-Temporal Graph Neural Networks*. http://arxiv.org/abs/2007.03113

Krafft, T., García Castrillo-Riesgo, L., Edwards, S., Fischer, M., Overton, J., Robertson-Steel, I., & König, A. (2003). European Emergency Data Project (EED Project): EMS data-based health surveillance system. *European Journal of Public Health*, *13*(3 SUPPL.), 85–90. https://doi.org/10.1093/eurpub/13.suppl_3.85

Kvalseth, T. O., & Deems, J. M. (1979). Statistical models of the demand for emergency medical services in an urban area. *American Journal of Public Health*, *69*(3), 250–255. https://doi.org/10.2105/AJPH.69.3.250

Laspidou, C., Papageorgiou, E., Kokkinos, K., Sahu, S., Gupta, A., & Tassiulas, L. (2015). Exploring patterns in water consumption by clustering. *Procedia Engineering*, *119*(1), 1439–1446. https://doi.org/10.1016/j.proeng.2015.08.1004

Li, W., Jacobs, R., & Morgan, D. (2018). Predicting the thermodynamic stability of perovskite oxides using machine learning models. *Computational Materials Science*, *150*(April), 454–463. https://doi.org/10.1016/j.commatsci.2018.04.033

Liashchynskyi, P., & Liashchynskyi, P. (2019). Grid Search, Random Search, Genetic Algorithm: A Big Comparison for NAS. *ArXiv*, *2017*, 1–11.

Liu, Y., Mu, Y., Chen, K., Li, Y., & Guo, J. (2020). Daily Activity Feature Selection in Smart Homes Based on Pearson Correlation Coefficient. *Neural Processing Letters*, *51*(2), 1771–1787. https://doi.org/10.1007/s11063-019-10185-8

Lowthian, J. A., Jolley, D. J., Curtis, A. J., Currell, A., Cameron, P. A., Stoelwinder, J. U., & McNeil, J. J. (2011). The challenges of population ageing: Accelerating demand for emergency ambulance services by older patients, 1995-2015. *Medical Journal of Australia*, *194*(11), 574–578. https://doi.org/10.5694/j.1326-5377.2011.tb03107.x

Ma, L., Li, M., Ma, X., Cheng, L., Du, P., & Liu, Y. (2017). A review of supervised object-based land-cover image classification. *ISPRS Journal of Photogrammetry and Remote Sensing*, *130*, 277–293. https://doi.org/10.1016/j.isprsjprs.2017.06.001

Malamiri, H. R. G., Rousta, I., Olafsson, H., Zare, H., & Zhang, H. (2018). Gap-filling of MODIS time series land surface temperature (LST) products using singular spectrum analysis (SSA). *Atmosphere*, *9*(9). https://doi.org/10.3390/atmos9090334

Matteson, D. S., McLean, M. W., Woodard, D. B., & Henderson, S. G. (2011). Forecasting emergency medical service call arrival rates. *Annals of Applied Statistics*, *5*(2 B), 1379–1406. https://doi.org/10.1214/10-AOAS442

Maxwell, A. E., Warner, T. A., & Fang, F. (2018). Implementation of machine-learning classification in remote sensing: An applied review. *International Journal of Remote Sensing*, *39*(9), 2784–2817.

https://doi.org/10.1080/01431161.2018.1433343

McConnel, C. E., & Wilson, R. W. (1998). The demand for prehospital emergency services in an aging society. *Social Science and Medicine*, *46*(8), 1027–1031. https://doi.org/10.1016/S0277-9536(97)10029-6

McDermott, P. L., & Wikle, C. K. (2019). Bayesian recurrent neural network models for forecasting and quantifying uncertainty in spatial-temporal data. *Entropy*, *21*(2). https://doi.org/10.3390/e21020184

Mustafee, N., Powell, J. H., & Harper, A. (2018). RH-RT: A data analytics framework for reducing wait time at emergency departments and centres for urgent care. *Angewandte Chemie International Edition, 6(11), 951–952.*, *2017*(July 2017).

Nair, R., & Miller-Hooks, E. (2009). Evaluation of relocation strategies for emergency medical service vehicles. *Transportation Research Record*, *2137*, 63–73. https://doi.org/10.3141/2137-08

Nakaya, T., & Yano, K. (2010). Visualising crime clusters in a space-time cube: An exploratory data-analysis approach using space-time kernel density estimation and scan statistics. *Transactions in GIS*, *14*(3), 223–239. https://doi.org/10.1111/j.1467-9671.2010.01194.x

Nicoletta, V., Lanzarone, E., Guglielmi, A., Bélanger, V., & Ruiz, A. (2017). Nicoletta, V., Lanzarone, E., Guglielmi, A., Belanger, V. (2017). A Bayesian Model for Describing and Predicting the Stochastic Demand of Emergency Calls.pdf. *Springer Proceedings in Mathematics & Statistics*, *194*(Bayesian Statistics in Action), 203–212. https://doi.org/https://doi.org/10.1007/978-3-319-54084-9_19

Pallonetto, F., De Rosa, M., Milano, F., & Finn, D. P. (2019). Demand response algorithms for smart-grid ready residential buildings using machine learning models. *Applied Energy*, *239*(October 2018), 1265–1282. https://doi.org/10.1016/j.apenergy.2019.02.020

Pérez, A. (2012). Comments on “Kernel density estimation for time series data.” *International Journal of Forecasting*, *28*(1), 15–19. https://doi.org/10.1016/j.ijforecast.2011.02.001

Praveena, M., & Jaiganesh, V. (2017). A Literature Review on Supervised Machine Learning Algorithms and Boosting Process. *International Journal of Computer Applications*, *169*(8), 32–35. https://doi.org/10.5120/ijca2017914816

Raschka, S. (2018). Model evaluation, model selection, and algorithm selection in machine learning. *ArXiv*.

Rastegari, E., Azizian, S., & Ali, H. (2019). Machine Learning and Similarity Network Approaches to Support Automatic Classification of Parkinson’s Diseases Using Accelerometer-based Gait Analysis. *Proceedings of the 52nd Hawaii International Conference on System Sciences*, *6*, 4231–4242. https://doi.org/10.24251/hicss.2019.511

Reuter-Oppermann, M., van den Berg, P. L., & Vile, J. L. (2017). Logistics for Emergency Medical Service systems. *Health Systems*, *6*(3), 187–208. https://doi.org/10.1057/s41306-017-0023-x

Santos, G., Marques, I., & Barbosa-Póvoa, A. (2019). *Improving Emergency Medical Services Through Vehicle Location Optimization*. *December*.

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., & Monfardini, G. (2009). The graph neural network model. *IEEE Transactions on Neural Networks*, *20*(1), 61–80. https://doi.org/10.1109/TNN.2008.2005605

Setzler, H., Saydam, C., & Park, S. (2009). EMS call volume predictions: A comparative study.

Computers and Operations Research, *36*(6), 1843–1851. https://doi.org/10.1016/j.cor.2008.05.010

Siler, K. F. (1975). Predicting demand for publicly dispatched ambulances in a metropolitan area. *BMC Health Services Research*, *10*(3), 254–263.

Sinayobye, J. O., Kaawaase Kyanda, S., Kiwanuka, N. F., & Musabe, R. (2019). Hybrid Model of Correlation Based Filter Feature Selection and Machine Learning Classifiers Applied on Smart Meter Data Set. *Proceedings - 2019 IEEE/ACM Symposium on Software Engineering in Africa, SEiA 2019*, 1–10. https://doi.org/10.1109/SEiA.2019.00009

Soltanpoor, R., & Sellis, T. (2016). Prescriptive analytics for big data. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *9877 LNCS*, 245–256. https://doi.org/10.1007/978-3-319-46922-5_19

Steins, K., Matinrad, N., & Grandberg, T. A. (2019). *Forecasting the Demand for Emergency Medical Services*. https://doi.org/10.24251/HICSS.2019.225

Stetco, A., Dinmohammadi, F., Zhao, X., Robu, V., Flynn, D., Barnes, M., Keane, J., & Nenadic, G. (2019). Machine learning methods for wind turbine condition monitoring: A review. *Renewable Energy*, *133*, 620–635. https://doi.org/10.1016/j.renene.2018.10.047

Svenson, J. E. (2000). Patterns of use of emergency medical transport: A population-based study. *American Journal of Emergency Medicine*, *18*(2), 130–134. https://doi.org/10.1016/S0735-6757(00)90002-0

Thanh Noi, P., & Kappas, M. (2017). Comparison of Random Forest, k-Nearest Neighbor, and Support Vector Machine Classifiers for Land Cover Classification Using Sentinel-2 Imagery. *Sensors (Basel, Switzerland)*, *18*(1). https://doi.org/10.3390/s18010018

Torgo, L., & Gama, J. (1996). Regression by classification. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *1159*, 51–60. https://doi.org/10.1007/3-540-61859-7_6

Tureczek, A., Nielsen, P. S., & Madsen, H. (2018). Electricity consumption clustering using smart meter data. *Energies*, *11*(4), 1–18. https://doi.org/10.3390/en11040859

Van Den Berg, P. L.-J. (2016). *Logistics of emergency response vehicles*.

Venkatesh, B., & Anuradha, J. (2019). A review of Feature Selection and its methods. *Cybernetics and Information Technologies*, *19*(1), 3–26. https://doi.org/10.2478/CAIT-2019-0001

Verma, A., & Kusiak, A. (2012). Fault monitoring of wind turbine generator brushes: A data-mining approach. *Journal of Solar Energy Engineering, Transactions of the ASME*, *134*(2). https://doi.org/10.1115/1.4005624

Vile, J. L., Gillard, J. W., Harper, P. R., & Knight, V. A. (2012). Predicting ambulance demand using singular spectrum analysis. *Journal of the Operational Research Society*, *63*(11), 1556–1565. https://doi.org/10.1057/jors.2011.160

Vile, J. L., Gillard, J. W., Harper, P. R., & Knight, V. A. (2016). Time-dependent stochastic methods for managing and scheduling Emergency Medical Services. *Operations Research for Health Care*, *8*, 42–52. https://doi.org/10.1016/j.orhc.2015.07.002

Voyant, C., Notton, G., Kalogirou, S., Nivet, M. L., Paoli, C., Motte, F., & Fouilloy, A. (2017). Machine learning methods for solar radiation forecasting: A review. *Renewable Energy*, *105*, 569–582. https://doi.org/10.1016/j.renene.2016.12.095

Wainer, J., & Fonseca, P. (2021). How to tune the RBF SVM hyperparameters? An empirical evaluation of 18 search algorithms. *Artificial Intelligence Review*, *54*(6), 4771–4797. https://doi.org/10.1007/s10462-021-10011-5

Wang, X., Tsokos, C. P., & Saghafi, A. (2018). Improved parameter estimation of Time Dependent Kernel Density by using Artificial Neural Networks. *Journal of Finance and Data Science*, *4*(3), 172–182. https://doi.org/10.1016/j.jfds.2018.04.002

Weng, S. F., Reps, J., Kai, J., Garibaldi, J. M., & Qureshi, N. (2017). Can machine-learning improve cardiovascular risk prediction using routine clinical data? *PLoS ONE*, *12*(4), 1–14.

Were, K., Bui, D. T., Dick, Ø. B., & Singh, B. R. (2015). A comparative assessment of support vector regression, artificial neural networks, and random forests for predicting and mapping soil organic carbon stocks across an Afromontane landscape. *Ecological Indicators*, *52*, 394–403. https://doi.org/10.1016/j.ecolind.2014.12.028

Wong, H. T., & Lai, P. C. (2010). Weather inference and daily demand for emergency ambulance services. *Emergency Medicine Journal*, *29*(1), 60–64. https://doi.org/10.1136/emj.2010.096701

Wong, Ho Ting, & Lai, P. C. (2014). Weather factors in the short-term forecasting of daily ambulance calls. *International Journal of Biometeorology*, *58*(5), 669–678. https://doi.org/10.1007/s00484-013-0647-x

Wong, Ho Ting, & Lin, J. J. (2020). The effects of weather on daily emergency ambulance service demand in Taipei: a comparison with Hong Kong. *Theoretical and Applied Climatology*, *141*(1–2), 321–330. https://doi.org/10.1007/s00704-020-03213-4

Xie, Y., Zhu, C., Zhou, W., Li, Z., Liu, X., & Tu, M. (2018). Evaluation of machine learning methods for formation lithology identification: A comparison of tuning processes and model performances. *Journal of Petroleum Science and Engineering*, *160*, 182–193. https://doi.org/10.1016/j.petrol.2017.10.028

Yan, K., & Zhang, D. (2015). Feature selection and analysis on correlated gas sensor data with recursive feature elimination. *Sensors and Actuators, B: Chemical*, *212*, 353–363. https://doi.org/10.1016/j.snb.2015.02.025

Yildiz, B., Bilbao, J. I., & Sproul, A. B. (2017). A review and analysis of regression and machine learning models on commercial building electricity load forecasting. *Renewable and Sustainable Energy Reviews*, *73*(March 2016), 1104–1122. https://doi.org/10.1016/j.rser.2017.02.023

Zambom, A. Z., & Dias, R. (2012). *A Review of Kernel Density Estimation with Applications to Econometrics*. 20–42. http://arxiv.org/abs/1212.2812

Zhang, Y., Wang, S., Phillips, P., & Ji, G. (2014). Binary PSO with mutation operator for feature selection using decision tree applied to spam detection. *Knowledge-Based Systems*, *64*, 22–31. https://doi.org/10.1016/j.knosys.2014.03.015

Zhang, Z., Chen, D., Liu, W., Racine, J. S., Ong, S. H., Chen, Y., Zhao, G., & Jiang, Q. (2011). Nonparametric evaluation of dynamic disease risk: A spatio-temporal kernel approach. *PLoS ONE*, *6*(3). https://doi.org/10.1371/journal.pone.0017381

Zhou, Z. (2016). *Predicting Ambulance Demand: Challenges and Methods*. 11–15. http://arxiv.org/abs/1606.05363

Zhou, Z., & Matteson, D. S. (2015). Predicting ambulance demand: A spatio-temporal kernel approach. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, *2015-Augus*, 2297–2303. https://doi.org/10.1145/2783258.2788570

Zhou, Z., & Matteson, D. S. (2016). Predicting melbourne ambulance demand using kernel warping. *Annals of Applied Statistics*, *10*(4), 1977–1996. https://doi.org/10.1214/16-AOAS961

Zhou, Z., Matteson, D. S., Woodard, D. B., Henderson, S. G., & Micheas, A. C. (2015). A Spatio-Temporal Point Process Model for Ambulance Demand. *Journal of the American Statistical Association*, *110*(509), 6–15. https://doi.org/10.1080/01621459.2014.941466

Appendix A – Additional information on Lisbon bases

This appendix presents additional information to complement the analysis of the vehicle dataset of Chapter 4.2. Table 22 shows the distribution of each type of vehicle dispatches for each base, as well as the location of each base.

Table 22 – Location of Lisbon bases and vehicle type ratio per base

Base	SIV ratio	VMER ratio	AEM ratio	Base location
b0	36.39%	0.25%	63.36%	*INEM Sede*
b1	0.03%	0.01%	99.96%	*GNR Reg. Cavalaria - Ajuda*
b2	0.19%	0.71%	99.10%	*Esquadra PSP - Bº Boavista - Benfica*
b3	0.18%	0.05%	99.77%	*Centro Saúde Lóios - Olivais*
b4	7.78%	0.43%	91.79%	*Hosp. Curry Cabral*
b5	0.70%	0.24%	99.06%	*Escola S.D. Benfica*
b6	0.23%	0.23%	99.53%	*GNR Brigada Fiscal - Beato*
b7	0.11%	0.01%	99.88%	*GNR Brigada Trânsito - Alcântara*
b8	0.31%	0.01%	98.68%	*Hosp. Egas Moniz*
b9	0.00%	99.84%	0.16%	*Hosp. São Francisco Xavier*
b10	0.72%	96.23%	3.06%	*Hosp. São José*
b11	0.66%	96.27%	3.07%	*Hosp. Santa Maria*
b12	0.21%	0.12%	99.67%	*INEM - R. Infante D. Pedro*
b13	16.44%	1.37%	82.19%	*Reg. Sapadores Bomb. - Av D. Carlos I*
b14	12.70%	3.17%	84.13%	*BV Ajuda*
b15	25.00%	9.52%	65.48%	*BV Beato*
b16	17.31%	7.69%	75.00%	*BV Campo de Ourique*
b17	22.70%	3.55%	73.76%	*BV Cabo Ruívo*
b18	11.01%	1.83%	87.16%	*BV Lisboa*
b19	30.23%	9.30%	60.47%	*BV Lisbonenses*

Appendix B – Correlation with vehicle target variables

This appendix presents the remaining histograms representing the correlation between the numerical attributes and the target variables from the vehicle dataset: SIV, VMER, and AEM. These graphs are parallel to those presented in Chapter 4.2.

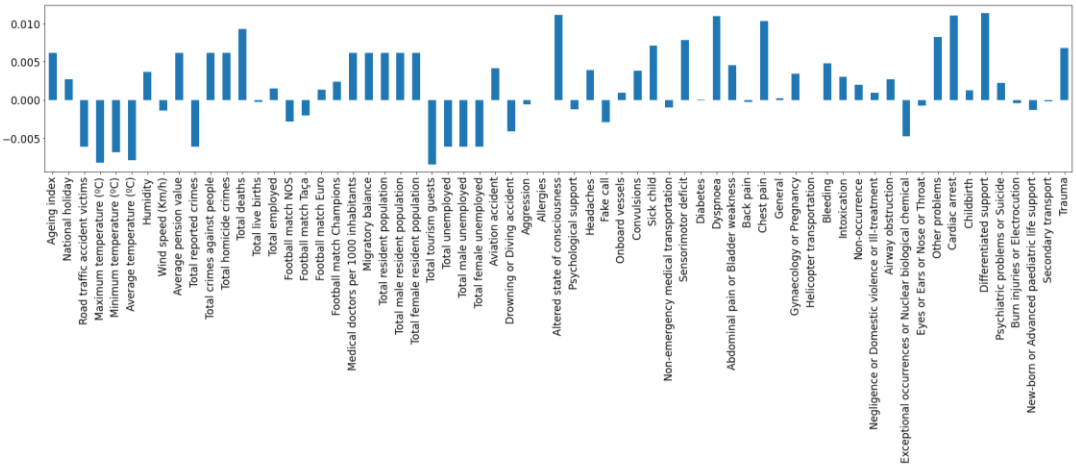


Figure 23 – Correlation between vehicles of type SIV and numerical attributes

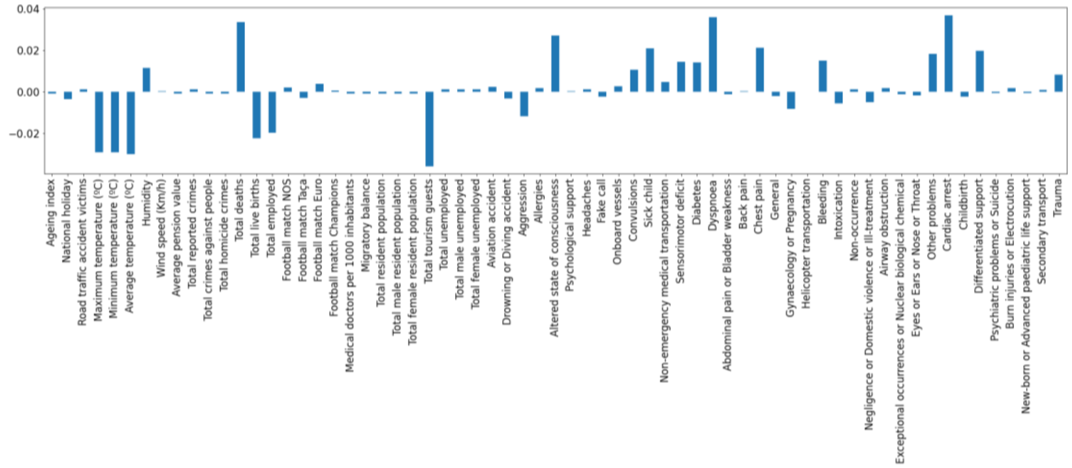


Figure 24 – Correlation between vehicles of type VMER and numerical attributes

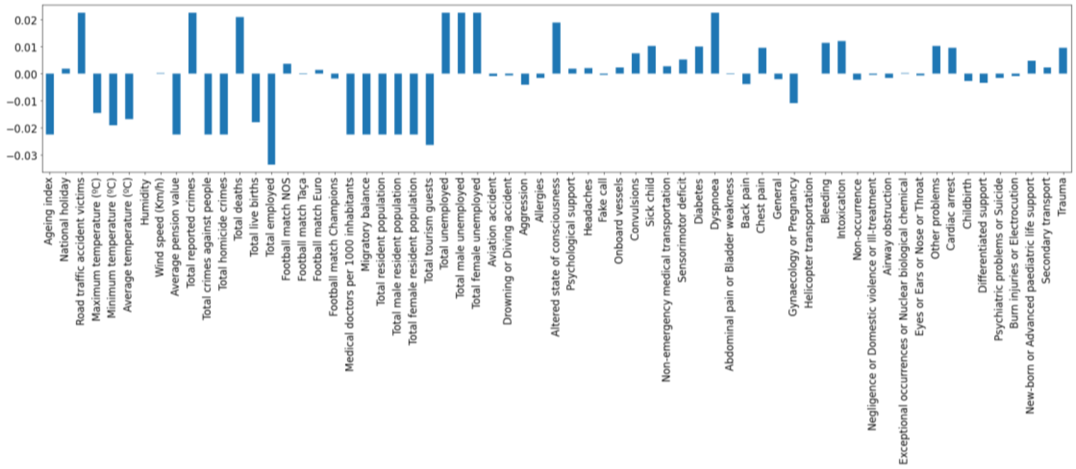


Figure 25 – Correlation between vehicles of type AEM and numerical attributes

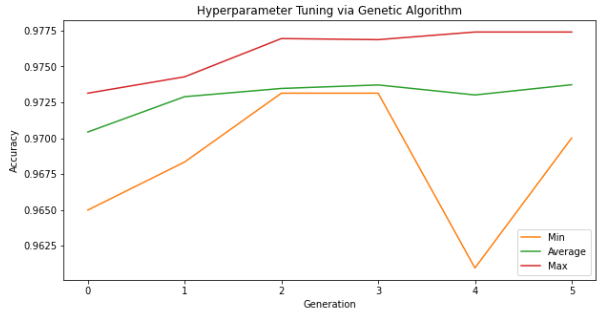
Appendix C – GA algorithm convergence

This appendix includes all the graphs obtained from applying GA hyperparameter tuning with 5 generations to GMM, Gradient Boosting, and AdaBoost models.

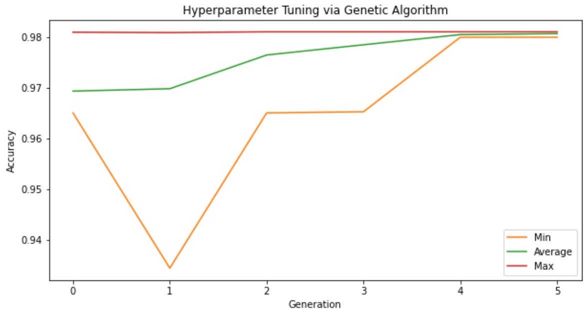
Table 23 – Hyperparameter tuning via GA plots for GMM models

SIV

CA database

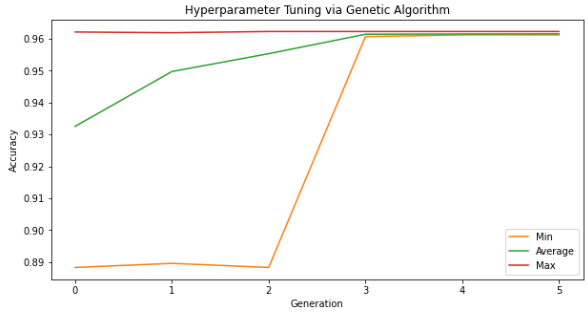


GA database

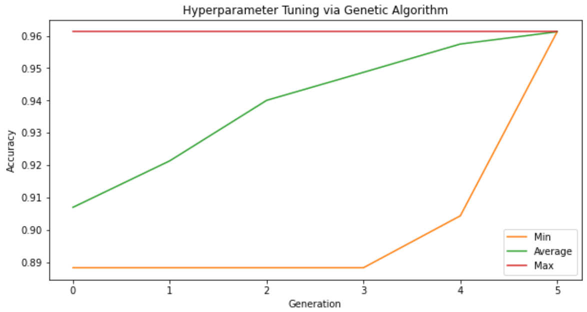


VMER

CA database

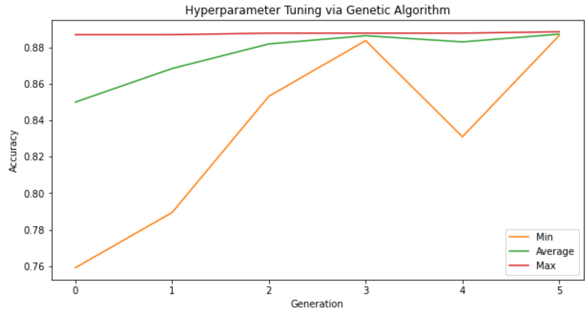


GA database

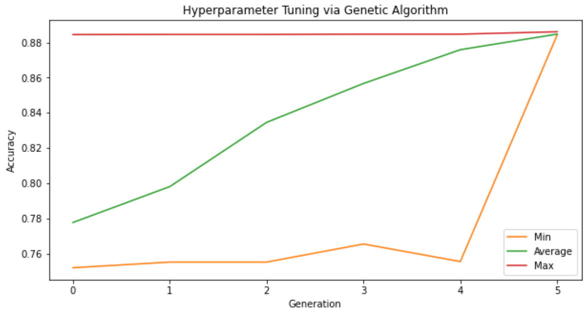


AEM

CA database

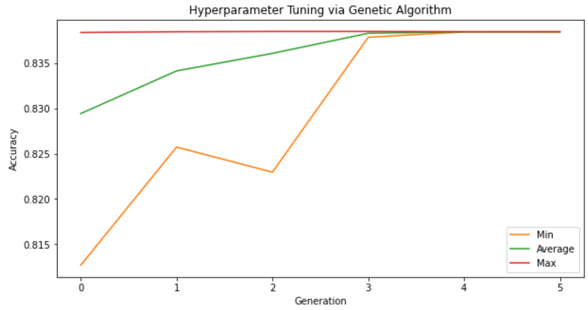


GA database



P1

CA database



GA database

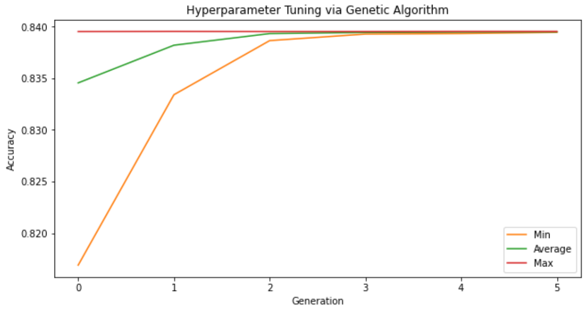


Table 23 – continuation

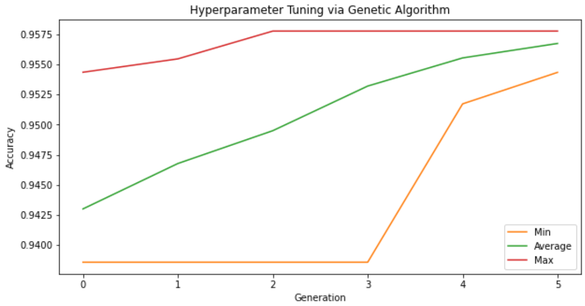
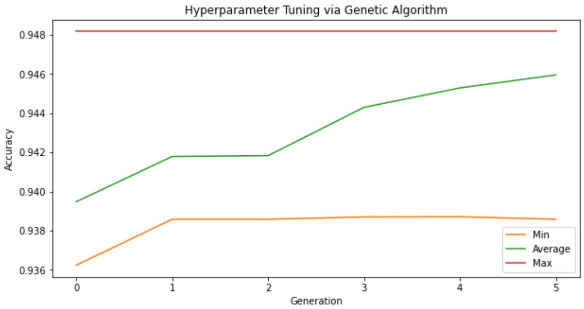
P3	
CA database	GA database
	

Table 24 – Hyperparameter tuning via GA plots for Gradient Boosting model with GA dataset

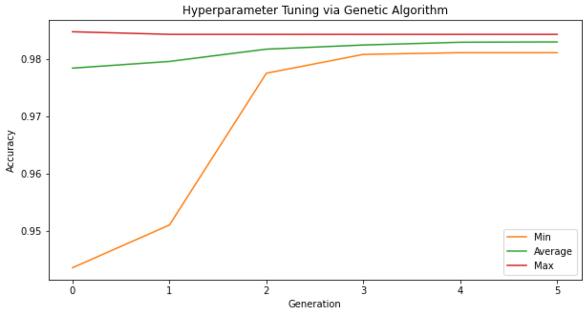
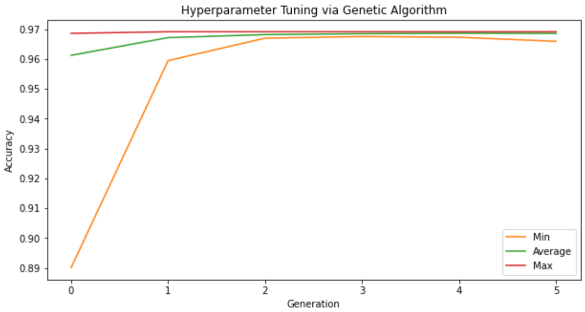
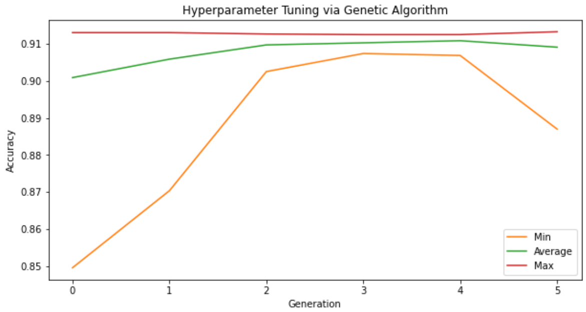
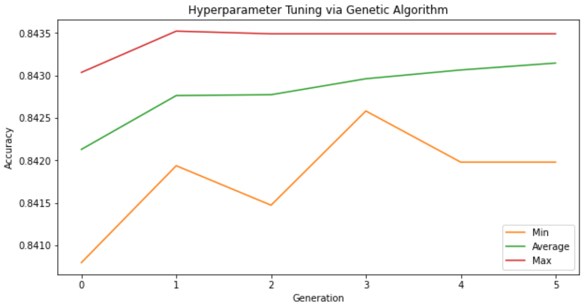
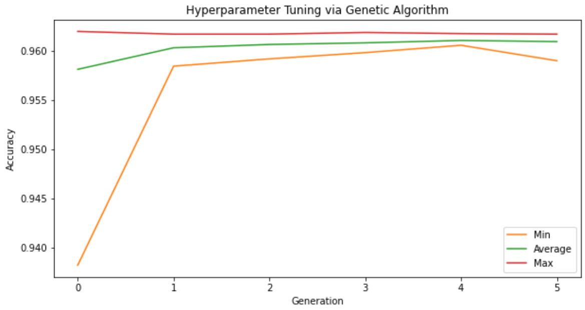
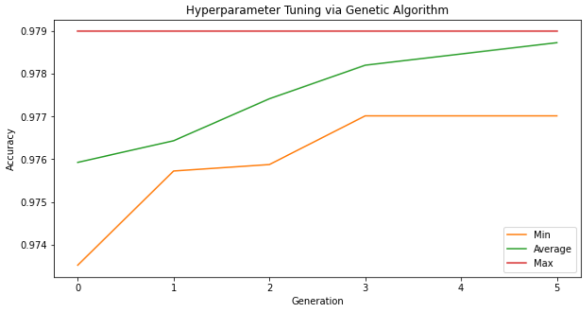
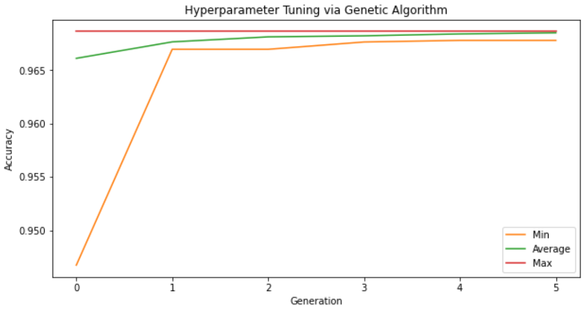
SIV	VMER
	
AEM	**P1**
	
P3	
	

Table 25 – Hyperparameter tuning via GA plots for AdaBoost model with CA dataset

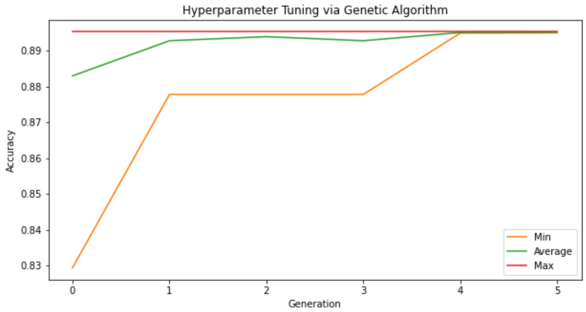
SIV



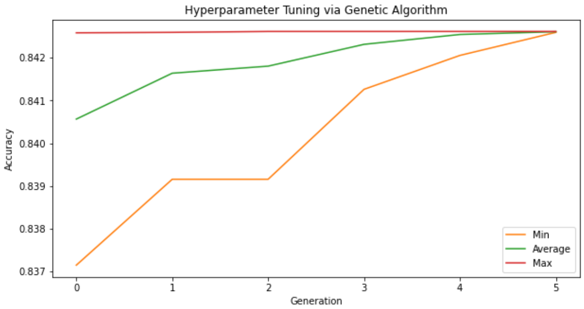
VMER



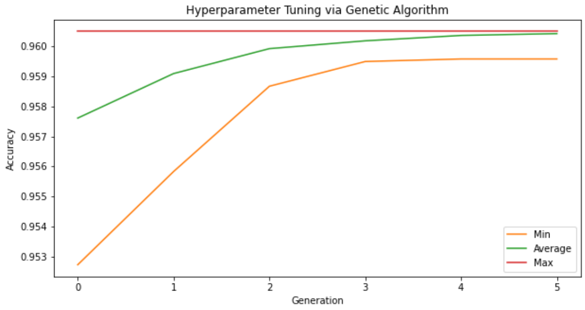
AEM



P1



P3



Appendix D – GMM Results

This appendix presents the results obtained from stratified 5-fold cross-validation from the multiple GMM models.

Table 26 – Cross-validation results of GMM models

			AUC		**Accuracy**		**Precision**		**Recall**	
			RS	GA	RS	GA	RS	GA	RS	GA
SIV	CA	μ	**0.9835**	0.9160	0.9733	**0.9768**	0.5827	0.6700	**0.8329**	0.6919
		σ	0.0016	0.0144	0.0008	0.0020	0.0084	0.0475	0.0258	0.0860
	GA	μ	0.9315	0.9699	0.9733	0.9745	0.5827	**0.7464**	**0.8329**	0.4107
		σ	0.0132	0.0062	0.0008	0.0075	0.0084	0.1583	0.0258	0.3265
VMER	CA	μ	**0.9856**	0.9842	0.9615	0.9613	**0.7452**	0.7451	**0.9965**	0.9939
		σ	0.0014	0.0013	0.0015	0.0011	0.0090	0.0074	0.0040	0.0055
	GA	μ	0.9269	0.9241	**0.9623**	0.9592	0.7515	0.7513	0.9906	0.9542
		σ	0.0251	0.0271	0.0018	0.0033	0.0147	0.0206	0.0147	0.0874
AEM	CA	μ	**0.9463**	0.9330	0.8867	**0.8888**	0.7557	**0.7955**	0.7935	0.7358
		σ	0.0022	0.0138	0.0020	0.0046	0.0061	0.0199	0.0044	0.0141
	GA	μ	0.7572	0.7339	0.8870	0.8877	0.7561	0.7600	**0.7947**	0.7908
		σ	0.0051	0.0126	0.0017	0.0049	0.0050	0.0062	0.0057	0.0280
P1	CA	μ	**0.5589**	0.5129	0.8370	**0.8384**	0.7334	**0.7579**	0.3856	0.3718
		σ	0.0045	0.0571	0.0007	0.0016	0.0030	0.0219	0.0030	0.0112
	GA	μ	0.2931	0.3260	0.8376	0.8375	0.7396	0.7365	0.3837	**0.3861**
		σ	0.0084	0.0064	0.0008	0.0008	0.0080	0.0025	0.0053	0.0033
P3	CA	μ	0.7663	**0.8070**	0.9502	**0.9536**	0.8177	**0.8590**	0.7118	0.6983
		σ	0.0430	0.0654	0.0041	0.0014	0.0273	0.0211	0.0221	0.0196
	GA	μ	0.6912	0.1269	0.9412	0.9378	0.7572	0.7217	0.7033	**0.7170**
		σ	0.0439	0.0062	0.0053	0.0011	0.0523	0.0055	0.0197	0.0063

Appendix E – ML Results

This appendix presents the results from stratified 5-fold cross-validation of the eight ML algorithms with default hyperparameters.

Table 27 – Cross-validation results of ML algorithms on CA dataset with default hyperparameters

		AUC	Accuracy	Precision	Recall
SIV	Naive Bayes	0.9647	0.9347	0.3483	**0.9954**
	KNN	0.9565	0.9743	0.6401	0.5875
	SVM	0.9456	0.9774	**0.7116**	0.5979
	Extra Trees	0.9664	0.9746	0.6671	0.5542
	Random Forest	0.9669	0.9745	0.6440	0.5953
	Gradient Boosting	**0.9873**	**0.9775**	0.6973	0.6312
	AdaBoost	0.9845	0.9752	0.6436	0.6547
	Bagging	0.9678	0.9742	0.6406	0.5986
VMER	Naive Bayes	0.9775	0.9613	0.7437	**0.9984**
	KNN	0.9815	0.9629	0.8138	0.8659
	SVM	0.9811	0.9658	0.8276	0.8698
	Extra Trees	0.9858	0.9629	**0.8280**	0.8510
	Random Forest	0.9872	0.9634	0.8132	0.8655
	Gradient Boosting	0.9905	0.9656	0.8246	0.8923
	AdaBoost	**0.9912**	**0.9667**	0.8154	0.9117
	Bagging	0.9861	0.9633	0.8164	0.8694
AEM	Naive Bayes	0.8309	0.7447	0.4894	**0.9999**
	KNN	0.9444	0.8899	0.7758	0.7824
	SVM	0.9533	**0.9042**	0.8013	0.8075
	Extra Trees	0.9507	0.8914	0.7925	0.7530
	Random Forest	0.9533	0.8901	0.7739	0.7843
	Gradient Boosting	**0.9596**	0.8964	**0.8165**	0.7391
	AdaBoost	0.9567	0.8933	0.7748	0.7953
	Bagging	0.9505	0.8901	0.7731	0.7848
P1	Naive Bayes	0.8515	0.7587	0.4667	**0.8142**
	KNN	0.7924	0.8223	0.6188	0.4635
	SVM	0.8023	0.8418	0.7647	0.3864
	Extra Trees	0.8325	0.8294	0.6562	0.4428
	Random Forest	0.8432	0.8374	0.7080	0.4206
	Gradient Boosting	0.8567	0.8415	**0.7800**	0.3715
	AdaBoost	**0.8588**	**0.8421**	0.7370	0.4175
	Bagging	0.8078	0.8246	0.6424	0.4251
P3	Naive Bayes	0.8781	0.7098	0.2768	**0.9954**
	KNN	0.9408	0.9563	0.8491	0.7420
	SVM	0.9731	**0.9607**	**0.9149**	0.7127
	Extra Trees	0.9783	0.9591	0.8867	0.7278
	Random Forest	0.9793	0.9599	0.9029	0.7181
	Gradient Boosting	0.9788	0.9587	0.8977	0.7095
	AdaBoost	**0.9819**	0.9594	0.8894	0.7193
	Bagging	0.9590	0.9563	0.8634	0.7222

Table 28 – Cross-validation results of ML algorithms on GA dataset with default hyperparameters

		AUC	**Accuracy**	**Precision**	**Recall**
SIV	Naive Bayes	0.9646	0.9346	0.3482	**0.9954**
	KNN	0.9098	0.9721	0.7828	0.2813
	SVM	0.9744	0.9822	**0.7992**	0.6579
	Extra Trees	**0.9913**	0.9817	0.7412	0.7317
	Random Forest	0.9905	0.9807	0.7557	0.6691
	Gradient Boosting	0.9908	**0.9826**	0.7709	0.7173
	AdaBoost	0.9845	0.9745	0.6335	0.6397
	Bagging	0.9778	0.9809	0.7436	0.7154
VMER	Naive Bayes	0.9775	0.9612	0.7433	**0.9967**
	KNN	0.9304	0.9319	**0.8553**	0.4701
	SVM	0.9860	0.9666	0.8439	0.8604
	Extra Trees	0.9898	0.9650	0.8266	0.8753
	Random Forest	0.9900	0.9655	0.8261	0.8766
	Gradient Boosting	0.9906	0.9661	0.8218	0.8903
	AdaBoost	**0.9911**	**0.9667**	0.8176	0.9037
	Bagging	0.9874	0.9608	0.8313	0.8138
AEM	Naive Bayes	0.8309	0.7447	0.4894	**0.9995**
	KNN	0.8928	0.8652	0.7977	0.5983
	SVM	0.9652	**0.9143**	0.8153	0.8419
	Extra Trees	0.9721	0.9131	0.8216	0.8240
	Random Forest	**0.9722**	0.9123	0.8256	0.8101
	Gradient Boosting	0.9658	0.9106	**0.8413**	0.7824
	AdaBoost	0.9575	0.8940	0.7805	0.7942
	Bagging	0.9640	0.9034	0.8123	0.7875
P1	Naive Bayes	0.8452	0.8086	0.5501	**0.6296**
	KNN	0.7882	0.8197	0.6097	0.4604
	SVM	0.7979	0.8422	**0.7806**	0.3753
	Extra Trees	0.8451	0.8386	0.7177	0.4162
	Random Forest	0.8427	0.8386	0.7257	0.4070
	Gradient Boosting	**0.8577**	**0.8428**	0.7637	0.3945
	AdaBoost	0.8566	0.8415	0.7433	0.4073
	Bagging	0.8094	0.8283	0.6618	0.4200
P3	Naive Bayes	0.9148	0.8044	0.3602	**0.9762**
	KNN	0.9311	0.9451	0.7781	0.7092
	SVM	0.9743	**0.9609**	**0.9164**	0.7134
	Extra Trees	0.9793	0.9606	0.9014	0.7242
	Random Forest	0.9776	0.9603	0.9139	0.7105
	Gradient Boosting	0.9797	0.9596	0.8971	0.7197
	AdaBoost	**0.9819**	0.9592	0.8846	0.7280
	Bagging	0.9602	0.9575	0.8723	0.7228

Appendix F – Results of ML hyperparameter tuning

This appendix shows the results of stratified 5-fold cross-validation for Gradient Boosting and AdaBoost with default hyperparameters, after tuning with RS and after tuning with GA.

Table 29 – Cross-validation results of hyperparameter tuning of Gradient Boosting with GA dataset

		SIV	**VMER**	**AEM**	**P1**	**P3**
AUC	Default	0.9908	0.9906	0.9658	0.8577	0.9797
	RS	**0.9923**	**0.9910**	**0.9737**	**0.8607**	0.9837
	GA	0.9918	0.9909	0.9724	0.8602	**0.9838**
Accuracy	Default	0.9826	0.9661	0.9106	0.8428	0.9596
	RS	**0.9834**	**0.9673**	**0.9151**	**0.8436**	**0.9616**
	GA	0.9833	0.9667	0.9129	0.8433	0.9613
Precision	Default	0.7709	0.8218	**0.8413**	**0.7637**	0.8971
	RS	**0.7721**	**0.8259**	0.8181	0.7468	**0.8972**
	GA	0.7648	0.8102	0.8152	0.7448	0.8923
Recall	Default	0.7173	0.8903	0.7824	0.3945	0.7197
	RS	0.7454	0.8960	**0.8399**	0.4108	0.7396
	GA	**0.7572**	**0.9172**	0.8326	**0.4178**	**0.7412**

Table 30 – Cross-validation results of hyperparameter tuning of AdaBoost with CA dataset

		SIV	**VMER**	**AEM**	**P1**	**P3**
AUC	Default	0.9845	0.9912	0.9567	0.8588	0.9819
	RS	**0.9846**	**0.9913**	**0.9569**	**0.8595**	**0.9827**
	GA	0.9844	**0.9913**	**0.9569**	0.8594	0.9826
Accuracy	Default	0.9752	0.9667	**0.8933**	0.8421	0.9594
	RS	**0.9774**	**0.9672**	**0.8933**	**0.8425**	**0.9599**
	GA	**0.9774**	0.9667	**0.8933**	0.8422	0.9598
Precision	Default	0.6436	0.8154	0.7748	**0.7370**	0.8894
	RS	**0.6860**	**0.8264**	0.7750	0.7353	0.8881
	GA	**0.6860**	0.8161	**0.7751**	**0.7370**	**0.8906**
Recall	Default	**0.6547**	0.9117	**0.7953**	0.4175	0.7193
	RS	0.6534	**0.8943**	0.7949	**0.4232**	**0.7314**
	GA	0.6534	0.9060	0.7947	0.4195	0.7277

Appendix G – Results of final models

This appendix presents the results from stratified 5-fold cross-validation of the four final models.

Table 31 – Cross-validation results of final models

			GMM RS	GMM GA	Gradient Boosting	AdaBoost
SIV	**AUC**	μ	0.9835	0.9160	**0.9923**	0.9846
		σ	0.0016	0.0144	0.0009	0.0014
	Accuracy	μ	0.9733	0.9768	**0.9834**	0.9774
		σ	0.0008	0.0020	0.0009	0.0011
	Precision	μ	0.5827	0.6700	**0.7721**	0.6860
		σ	0.0084	0.0475	0.0093	0.0183
	Recall	μ	**0.8329**	0.6919	0.7454	0.6534
		σ	0.0258	0.0860	0.0270	0.0206
VMER	**AUC**	μ	0.9856	0.9842	0.9910	**0.9913**
		σ	0.0014	0.0013	0.0009	0.0006
	Accuracy	μ	0.9615	0.9613	**0.9673**	0.9672
		σ	0.0015	0.0011	0.0012	0.0021
	Precision	μ	0.7452	0.7451	0.8259	**0.8264**
		σ	0.0090	0.0074	0.0075	0.0122
	Recall	μ	**0.9965**	0.9939	0.8960	0.8943
		σ	0.0040	0.0055	0.0106	0.0096
AEM	**AUC**	μ	0.9463	0.9330	**0.9737**	0.9569
		σ	0.0022	0.0138	0.0016	0.0016
	Accuracy	μ	0.8867	0.8888	**0.9151**	0.8933
		σ	0.0020	0.0046	0.0041	0.0009
	Precision	μ	0.7557	0.7955	**0.8181**	0.7750
		σ	0.0061	0.0200	0.0082	0.0017
	Recall	μ	0.7935	0.7358	**0.8399**	0.7949
		σ	0.0044	0.0141	0.0102	0.0036
P1	**AUC**	μ	0.5589	0.5129	**0.8607**	0.8595
		σ	0.0045	0.0571	0.0012	0.0012
	Accuracy	μ	0.8370	0.8384	**0.8436**	0.8425
		σ	0.0007	0.0016	0.0006	0.0008
	Precision	μ	0.7334	**0.7579**	0.7468	0.7353
		σ	0.0030	0.0219	0.0028	0.0037
	Recall	μ	0.3856	0.3718	0.4180	**0.4232**
		σ	0.0030	0.0112	0.0044	0.0041
P3	**AUC**	μ	0.7663	0.8070	**0.9837**	0.9827
		σ	0.0429	0.0654	0.0007	0.0007
	Accuracy	μ	0.9502	0.9536	**0.9616**	0.9599
		σ	0.0041	0.0014	0.0006	0.0007
	Precision	μ	0.8177	0.8590	**0.8972**	0.8881
		σ	0.0273	0.0211	0.0046	0.0032
	Recall	μ	0.7118	0.6983	**0.7396**	0.7314
		σ	0.0221	0.0196	0.0084	0.0075

Appendix H – Learning curves

This appendix shows the obtained learning curves for the final models.

Table 32 – Learning curves of final models

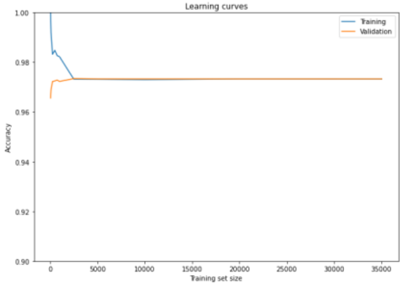
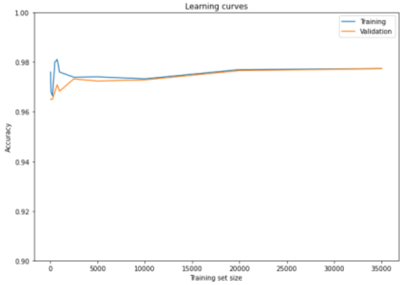
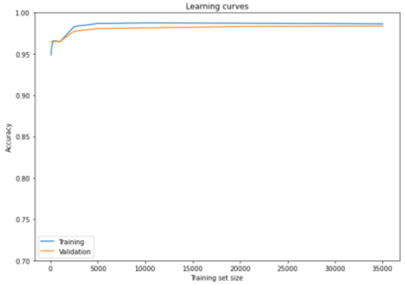
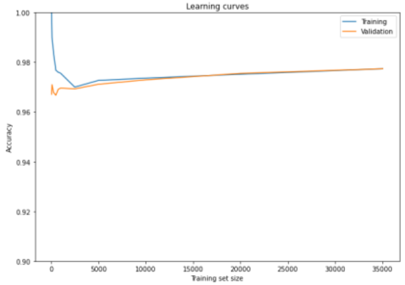
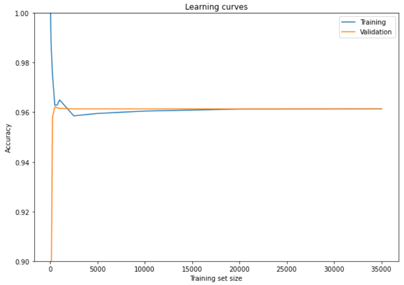
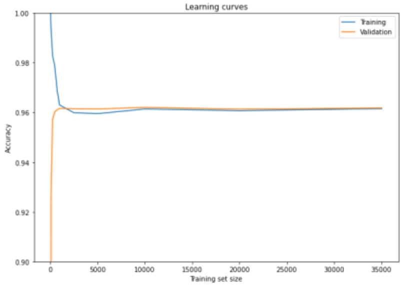
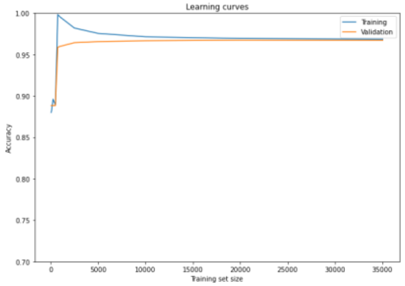
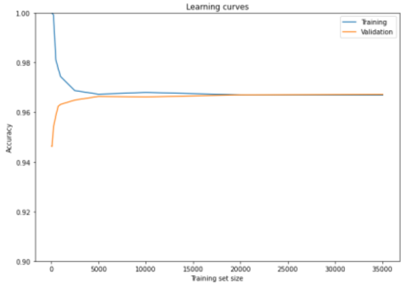
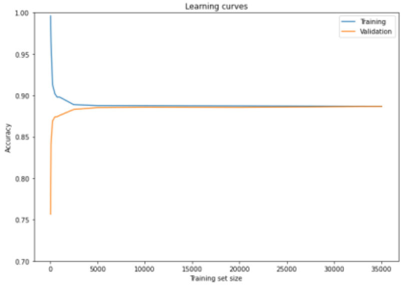
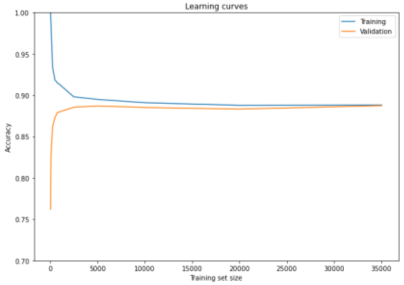
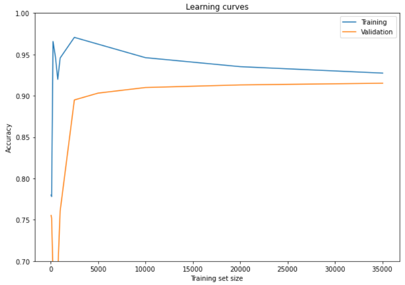
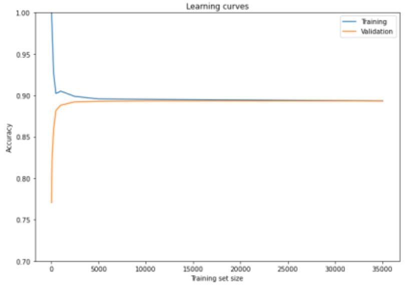
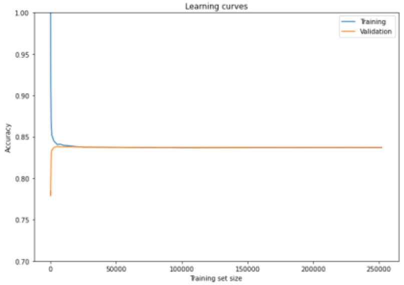
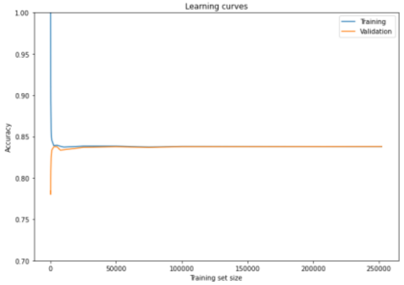
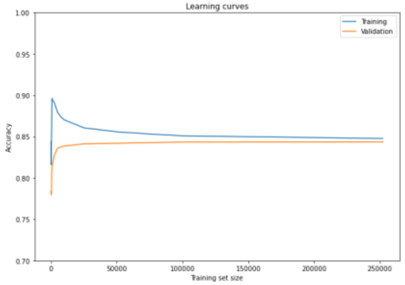
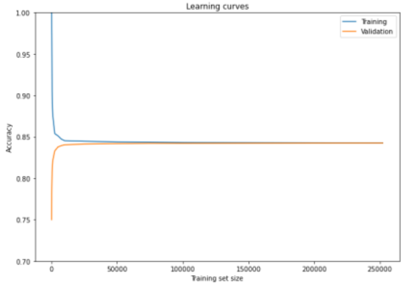
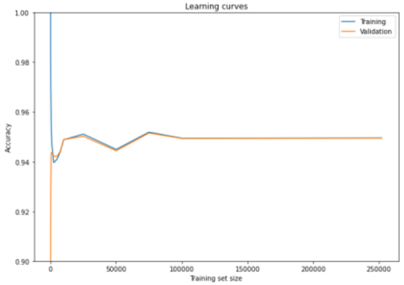
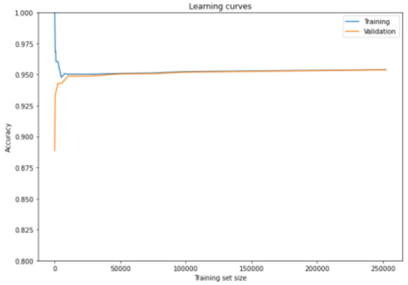
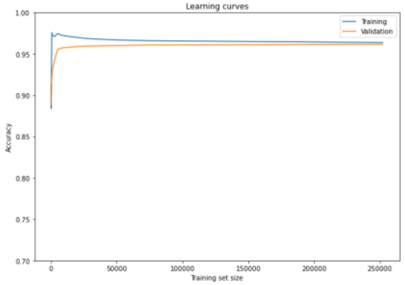
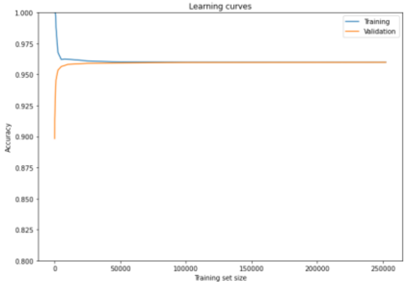
	GMM RS	GMM GA	Gradient Boosting	AdaBoost
SIV				
	GMM RS	**GMM GA**	**Gradient Boosting**	**AdaBoost**
VMER				
	GMM RS	**GMM GA**	**Gradient Boosting**	**AdaBoost**
AEM				

Table 32 – continuation

	GMM RS	GMM GA	Gradient Boosting	AdaBoost
P1				
	GMM RS	**GMM GA**	**Gradient Boosting**	**AdaBoost**
P3				

Appendix I – ROC curves

This appendix shows the obtained ROC curves for the final models.

Table 33 – ROC curves of final models

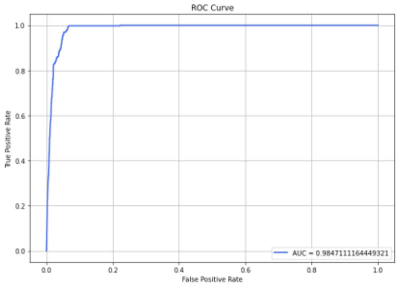
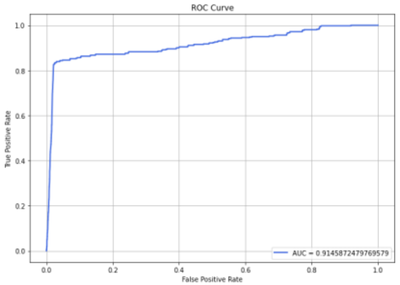
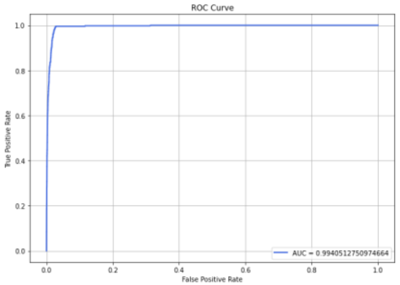
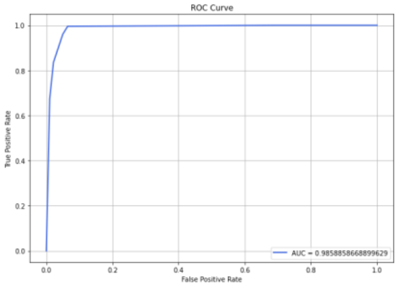
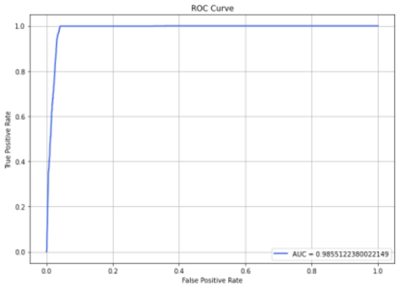
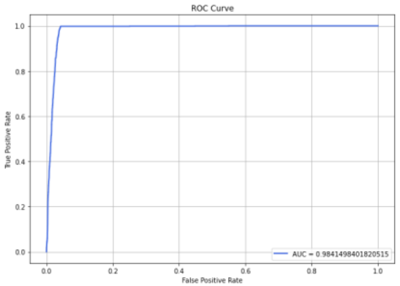
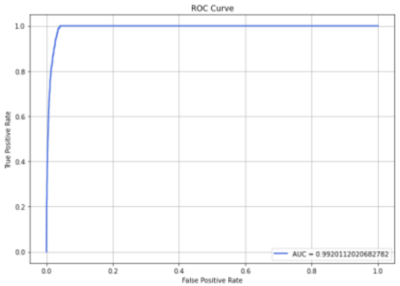
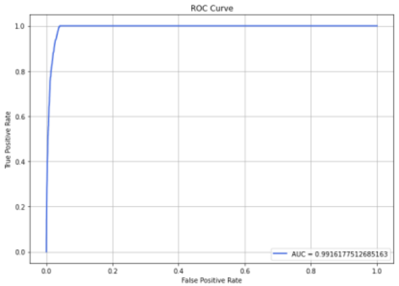
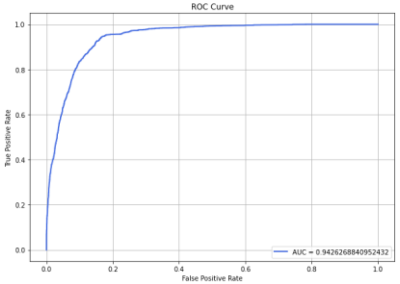
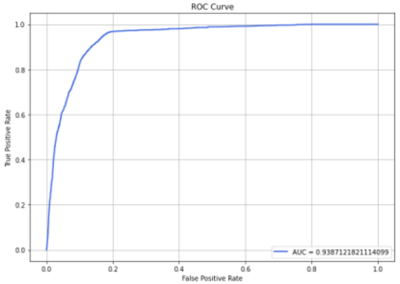
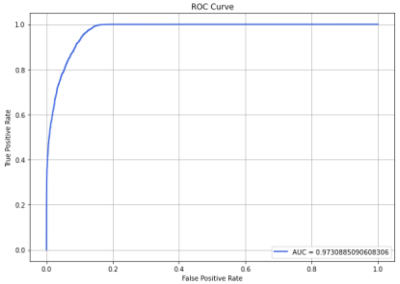
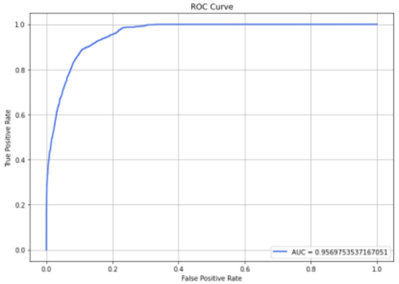
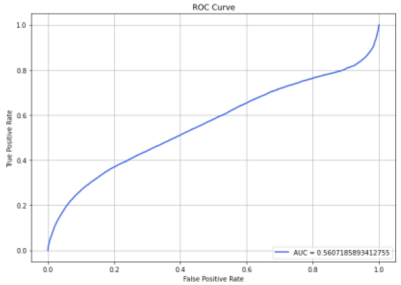
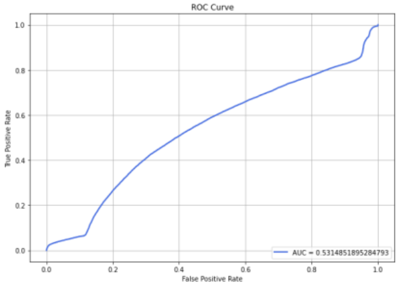
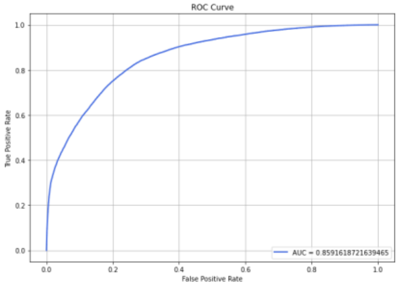
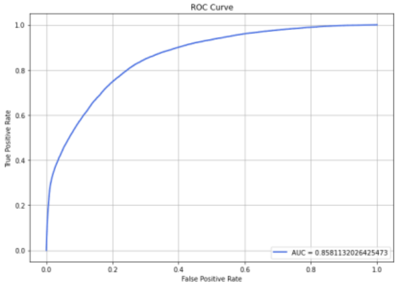
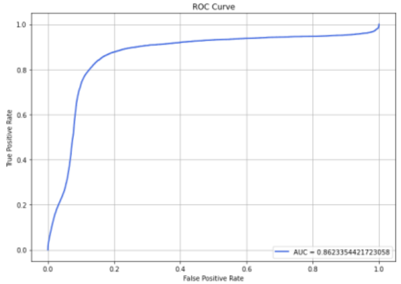
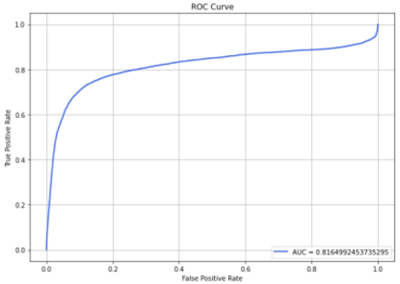
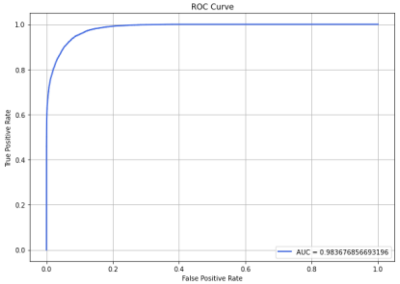
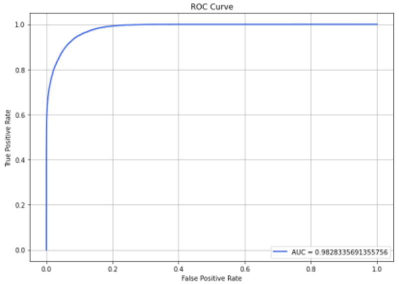
	GMM RS	GMM GA	Gradient Boosting	AdaBoost
SIV				
	GMM RS	**GMM GA**	**Gradient Boosting**	**AdaBoost**
VMER				
	GMM RS	**GMM GA**	**Gradient Boosting**	**AdaBoost**
AEM				

Table 33 – continuation

	GMM RS	**GMM GA**	**Gradient Boosting**	**AdaBoost**
P1				
	GMM RS	**GMM GA**	**Gradient Boosting**	**AdaBoost**
P3				

Appendix J – Precision-Recall curves

This appendix shows the obtained Precision-Recall curves for the final models.

Table 34 – Precision-Recall curves of final models

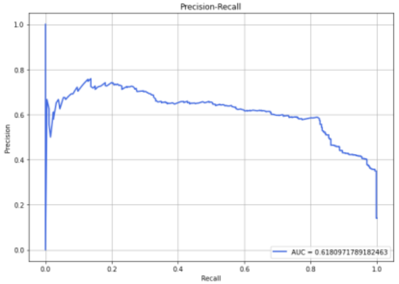
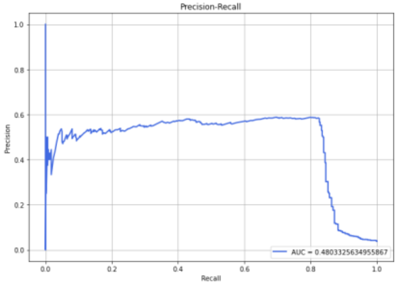
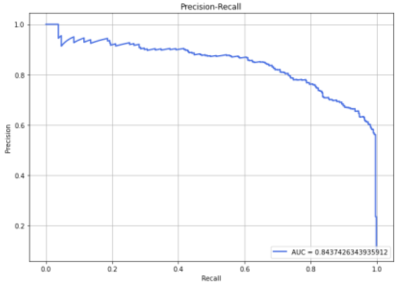
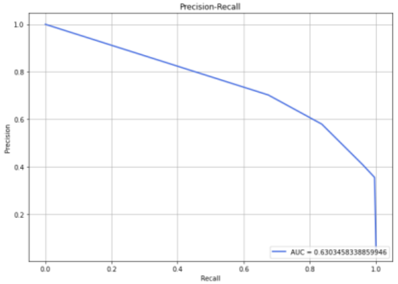
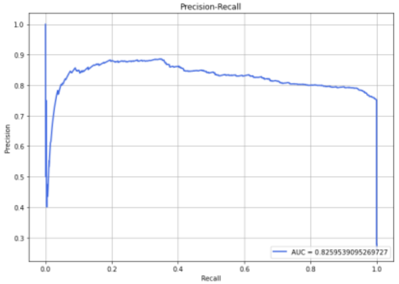
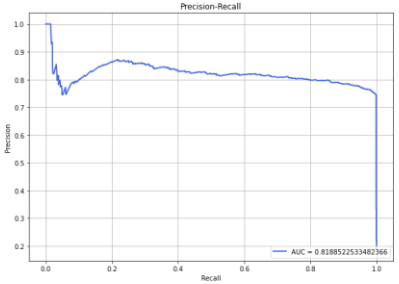
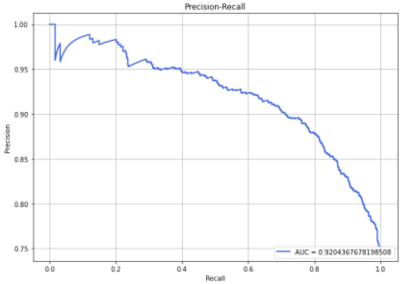
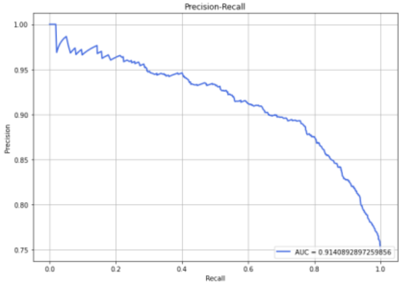
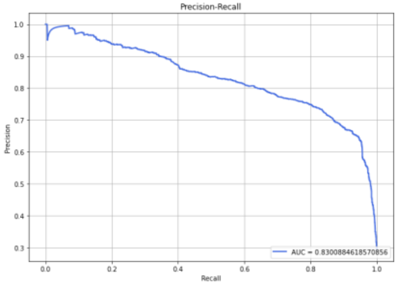
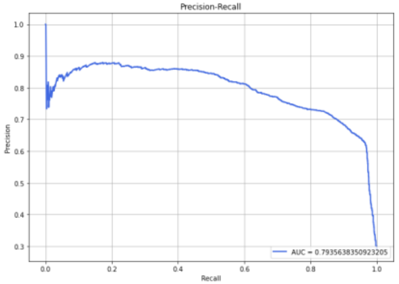
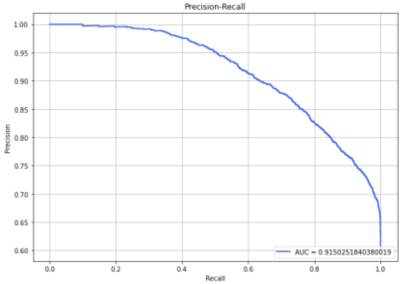
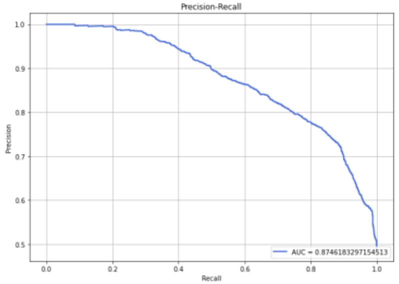
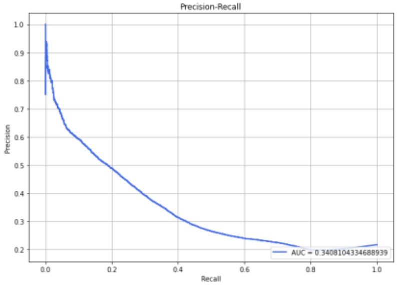
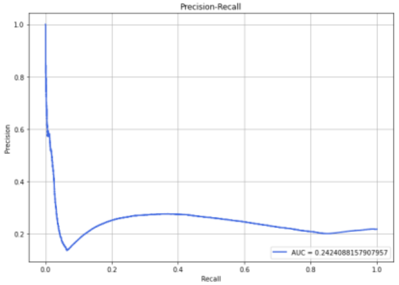
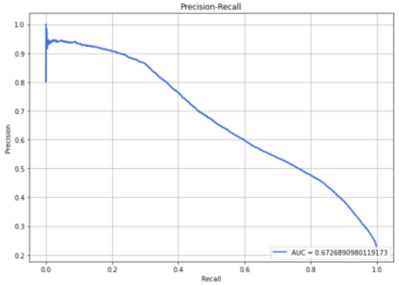
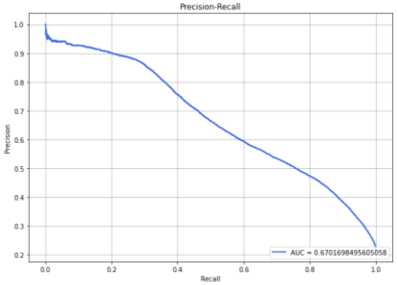
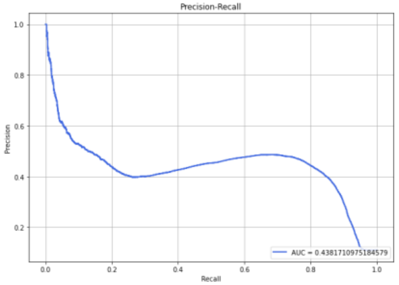
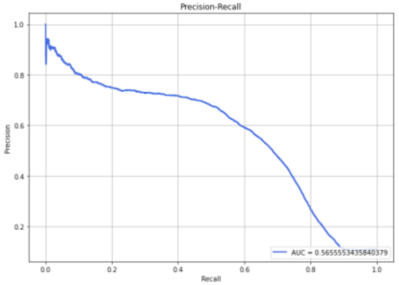
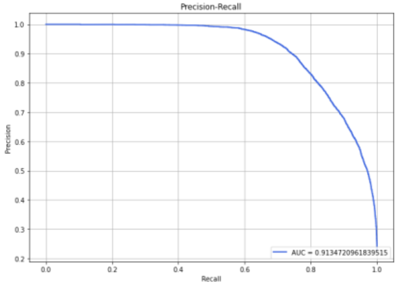
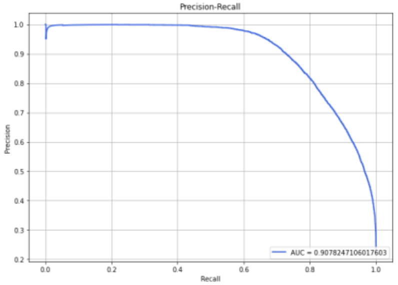
	GMM RS	GMM GA	Gradient Boosting	AdaBoost
SIV				
	GMM RS	**GMM GA**	**Gradient Boosting**	**AdaBoost**
VMER				
	GMM RS	**GMM GA**	**Gradient Boosting**	**AdaBoost**
AEM				

Table 34 – continuation

	GMM RS	GMM GA	Gradient Boosting	AdaBoost
P1				
	GMM RS	**GMM GA**	**Gradient Boosting**	**AdaBoost**
P3				

Appendix K – Confusion matrices

This appendix contains the confusion matrices of the final models.

Table 35 – Confusion matrices of final models

SIV

Class 0: [0,1] dispatches
Class 1: [2,8] dispatches

GMM RS

		Predicted Class		
		Class 0	Class 1	Sum
Actual Class	Class 0	12407 94.42%	273 2.08%	12680 96.50%
	Class 1	80 0.61%	380 2.89%	460 3.50%
	Sum	12487 95.03%	653 4.97%	13140 97.31%

GMM GA

		Predicted Class		
		Class 0	Class 1	Sum
Actual Class	Class 0	12533 95.38%	147 1.12%	12680 96.50%
	Class 1	150 1.14%	310 2.36%	460 3.50%
	Sum	12683 96.52%	457 3.48%	13140 97.74%

Gradient Boosting

		Predicted Class		
		Class 0	Class 1	Sum
Actual Class	Class 0	12593 95.84%	96 0.73%	12689 96.57%
	Class 1	114 0.87%	337 2.56%	451 3.43%
	Sum	12707 96.70%	433 3.30%	13140 98.40%

AdaBoost

		Predicted Class		
		Class 0	Class 1	Sum
Actual Class	Class 0	12560 95.59%	129 0.98%	12689 96.57%
	Class 1	147 1.12%	304 2.31%	451 3.43%
	Sum	12707 96.70%	433 3.30%	13140 97.90%

VMER

Class 0: [0,1] dispatches
Class 1: [2,9] dispatches

GMM RS

		Predicted Class		
		Class 0	Class 1	Sum
Actual Class	Class 0	11141 84.79%	531 4.04%	11672 88.83%
	Class 1	3 0.02%	1465 11.15%	1468 11.17%
	Sum	11144 84.81%	1996 15.19%	13140 95.94%

GMM GA

		Predicted Class		
		Class 0	Class 1	Sum
Actual Class	Class 0	11209 85.30%	463 3.52%	11672 88.83%
	Class 1	32 0.24%	1436 10.93%	1468 11.17%
	Sum	11241 85.55%	1899 14.45%	13140 96.23%

Gradient Boosting

		Predicted Class		
		Class 0	Class 1	Sum
Actual Class	Class 0	11429 86.98%	267 2.03%	11696 89.01%
	Class 1	146 1.11%	1298 9.88%	1444 10.99%
	Sum	11575 88.09%	1565 11.91%	13140 96.86%

AdaBoost

		Predicted Class		
		Class 0	Class 1	Sum
Actual Class	Class 0	11430 86.99%	266 2.02%	11696 89.01%
	Class 1	161 1.23%	1283 9.76%	1444 10.99%
	Sum	11591 88.21%	1549 11.79%	13140 96.75%

AEM

Class 0: [0,3] dispatches
Class 1: [4,27] dispatches

GMM RS

		Predicted Class		
		Class 0	Class 1	Sum
Actual Class	Class 0	9106 69.30%	818 6.23%	9924 75.53%
	Class 1	675 5.14%	2541 19.34%	3216 24.47%
	Sum	9781 74.44%	3359 25.56%	13140 88.64%

GMM GA

		Predicted Class		
		Class 0	Class 1	Sum
Actual Class	Class 0	9322 70.94%	602 4.58%	9924 75.53%
	Class 1	863 6.57%	2353 17.91%	3216 24.47%
	Sum	10185 77.51%	2955 22.49%	13140 88.85%

Gradient Boosting

		Predicted Class		
		Class 0	Class 1	Sum
Actual Class	Class 0	9336 71.05%	628 4.78%	9964 75.83%
	Class 1	526 4.00%	2650 20.17%	3176 24.17%
	Sum	9862 75.05%	3278 24.95%	13140 91.22%

AdaBoost

		Predicted Class		
		Class 0	Class 1	Sum
Actual Class	Class 0	9252 70.41%	712 5.42%	9964 75.83%
	Class 1	664 5.05%	2512 19.12%	3176 24.17%
	Sum	9916 75.46%	3224 24.54%	13140 89.53%

Table 35 – continuation

	Class 0: [0,1] dispatches Class 1: [2,15] dispatches									
P1	**GMM RS**					**GMM GA**				
			Predicted Class					Predicted Class		
			Class 0	Class 1	Sum			Class 0	Class 1	Sum
	Actual Class	Class 0	71465 75.54%	2702 2.86%	74167 78.39%	Actual Class	Class 0	71351 75.42%	2816 2.98%	74167 78.39%
		Class 1	12502 13.21%	7939 8.39%	20441 21.61%		Class 1	12466 13.18%	7975 8.43%	20441 21.61%
		Sum	83967 88.75%	10641 11.25%	94608 83.93%		Sum	83817 88.59%	10791 11.41%	94608 83.85%
	Gradient Boosting					**AdaBoost**				
			Predicted Class					Predicted Class		
			Class 0	Class 1	Sum			Class 0	Class 1	Sum
	Actual Class	Class 0	71155 75.21%	3057 3.23%	74212 78.44%	Actual Class	Class 0	70945 74.99%	3267 3.45%	74212 78.44%
		Class 1	11778 12.45%	8618 9.11%	20396 21.56%		Class 1	11676 12.34%	8720 9.22%	20396 21.56%
		Sum	82933 87.66%	11675 12.34%	94608 84.32%		Sum	82621 87.33%	11987 12.67%	94608 84.21%
	Class 0: [0,13] dispatches Class 1: [14,68] dispatches									
P3	**GMM RS**					**GMM GA**				
			Predicted Class					Predicted Class		
			Class 0	Class 1	Sum			Class 0	Class 1	Sum
	Actual Class	Class 0	83099 87.84%	982 1.04%	84081 88.87%	Actual Class	Class 0	83474 88.23%	607 0.64%	84081 88.87%
		Class 1	3324 3.51%	7203 7.61%	10527 11.13%		Class 1	3390 3.58%	7137 7.54%	10527 11.13%
		Sum	86423 91.35%	8185 8.65%	94608 95.45%		Sum	86864 91.81%	7744 8.19%	94608 95.78%
	Gradient Boosting					**AdaBoost**				
			Predicted Class					Predicted Class		
			Class 0	Class 1	Sum			Class 0	Class 1	Sum
	Actual Class	Class 0	83217 87.96%	824 0.87%	84041 88.83%	Actual Class	Class 0	83114 87.85%	927 0.98%	84041 88.83%
		Class 1	2770 2.93%	7797 8.24%	10567 11.17%		Class 1	2802 2.96%	7765 8.21%	10567 11.17%
		Sum	85987 90.89%	8621 9.11%	94608 96.20%		Sum	85916 90.81%	8692 9.19%	94608 96.06%

Appendix L – Gradient Boosting final model

This appendix presents the final attributes selected for the Gradient Boosting model, in addition to the base set of attributes, as well as the values of the hyperparameters.

Table 36 – Attributes selected for Gradient Boosting model

Attributes	SIV	VMER	AEM	P1	P3
Ageing index					X
National holiday	X	X	X	X	X
Road traffic accident victims	X	X	X		
Maximum temperature (ºC)				X	
Minimum temperature (ºC)		X	X		
Average temperature (ºC)					X
Humidity	X				X
Wind speed (Km/h)		X		X	X
Average pension value					X
Total reported crimes	X				
Total crimes against people	X	X	X	X	
Total homicide crimes	X			X	X
Total deaths		X	X	X	X
Total live births	X	X	X	X	
Total employed	X		X	X	X
Football match NOS		X			
Football match Taça	X				
Football match Euro		X	X	X	X
Football match Champions	X			X	X
Medical doctors per 1000 inhabitants		X	X	X	X
Migratory balance			X		
Total resident population	X			X	X
Total male resident population		X	X	X	
Total female resident population				X	
Total tourism guests	X	X	X	X	X
Total unemployed	X			X	X
Total male unemployed	X	X			X
Total female unemployed	X	X		X	
Aviation accident	X		X		
Drowning or Diving accident				X	
Aggression	X		X	X	
Allergies					
Altered state of consciousness (ASC)					
Psychological support			X	X	
Headaches	X				
Fake call		X			X
Onboard vessels	X	X	X		X
Convulsions	X		X		
Sick child	X	X	X	X	X
Non-emergency medical transportation					X
Sensorimotor deficit	X	X			
Diabetes		X			X
Dyspnoea		X		X	
Abdominal pain or Bladder weakness			X	X	X
Back pain			X	X	X
Chest pain	X			X	
General	X	X			X
Gynaecology or Pregnancy		X	X	X	X
Helicopter transportation	X	X	X	X	X

Table 36 – continuation

Attributes	SIV	VMER	AEM	P1	P3
Bleeding			X	X	
Intoxication		X			
Non-occurrence	X				
Negligence or Domestic violence or Ill-treatment	X				X
Airway obstruction			X	X	X
Exceptional occurrences or Nuclear biological chemical			X	X	
Eyes or Ears or Nose or Throat	X			X	X
Other problems	X			X	X
Cardiac arrest	X	X	X	X	X
Childbirth	X			X	
Differentiated support	X	X			X
Psychiatric problems or Suicide		X	X		
Burn injuries or Electrocution	X	X	X	X	X
New-born or Advanced paediatric life support		X			X
Secondary transport	X	X		X	X
Trauma	X	X	X		X

Table 37 – Hyperparameter values used for Gradient Boosting model

Hyperparameter	SIV	VMER	AEM	P1	P3
Learning rate	0.05	0.2	0.15	0.05	0.05
Number of estimators	50	450	350	350	350
Subsample	0.8	0.9	1	0.9	0.9
Minimum samples split	100	500	5	50	50
Minimum samples leaf	50	5	50	100	100
Maximum depth	5	1	3	5	5
Maximum features	55	35	55	45	45
Loss function	‘deviance’	‘deviance’	‘deviance’	‘deviance’	‘deviance’
Criterion function	‘friedman_mse’	‘friedman_mse’	‘friedman_mse’	‘friedman_mse’	‘friedman_mse’