UNIVERSITAT DE BARCELONA

Final Degree Project

Biomedical Engineering Degree

MACHINE LEARNING PREDICTION OF BURST SUPPRESSION UNDER GENERAL ANESTHESIA

Barcelona, 21st January 2022

Author - Joana Collet i Fàbregas

Tutor - Dr. Pedro Luís Gambús Cerrillo

ABSTRACT

During propofol-remifentanil induced general anesthesia, burst suppression (BS) EEG patterns commonly occur in around 50% of the patients, with an increasing incidence with age. However, this phenomenon has been reported to be an indicator of too high anesthetic doses and produce adverse outcomes such as postoperative delirium, cognitive deficits, and it has even reported to be a postoperative mortality predictor.

In light of the above, the present study aims to address the lack of predictive techniques for BS occurrence anticipation by developing Machine Learning predictive models such as SVM, KNN, RF, and XGB. Therefore, a large dataset including different monitored parameters during propofol-remifentanil induced general anesthesia from many patients has been used for both training and testing the models, as well as for final validation of the selected model.

Obtained results present an acceptable overall performance of the SVM model with a ROC-AUC score of 0.829, and a feature importance analysis shows a high influence of age and BIS value for the final prediction. Nonetheless, 25% of the predictions have been reported to have accuracies under 0.6, questioning the reliability of the model and making it useful as an orientative aiding tool for anesthesiologists, but not the ultimate decisive factor. Hence, further studies involving more variability on the data, validation techniques and confidence intervals for each process, and an exhaustive feature selection analysis, along with the repetition of the study with different ML algorithms should be performed to improve the predictive ability of the current model and achieve better performances.

Keywords: *General anesthesia, Anesthesia monitoring, Burst Suppression, Predictive model, Machine Learning*

ACKNOWLEDGEMENTS

First of all, I would like to express my sincere appreciation to the project supervisor Dr. Pedro Luis Gambús for offering me the opportunity to carry out this project with the SPEC-M research group as well as for his supervision and advice throughout the project and for being always available for any problem or doubt. Special thanks also to all the people working in operating room number 4 of the CMA in Hospital Clínic, for making my stay there a very pleasant experience with a friendly environment while allowing me to know how surgical surroundings work.

Secondly, I would like to show my deepest gratitude to Joan Altés for his continuous guidance and help from the beginning of the project and throughout the development of the entire script and final results evaluation. I am immensely thankful for his constant availability and for being so generous with his time.

My greatest appreciation also to all my friends for accompanying me both in the study hours and in my leisure time, always showing interest in anything related to this study, and for trusting in me and the success of the project even when I doubted.

Last but not least, I am deeply thankful to my parents, Xevi and Susi, for encouraging me through all my student years. For their patience, constant help, and unconditional support, and for teaching me the importance of a job well done while enjoying it. Special thanks also to my mother for his revision and counseling during the writing of the project and for always showing a smile when it comes to giving a hand.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
AMG	Acceleromyography
BIS	Bispectral index
BP	Blood pressure
BS	Burst suppression
BSR or SR	Burst suppression rate
CF	Confusion matrix
CMA	Major ambulatory surgery area
CNS	Central nervous system
COPD	Chronic obstructive pulmonary disease
EEG	Electroencephalogram
EMG	Electromyography
FFT	Fast Fourier Transform
FN	False Negative
FP	False Positive
KNN	K-Nearest Neighbors
ML	Machine Learning

MMG	Mechanomyography
NIBP	Non-invasive arterial blood pressure
RF	Random Forest
ROC	Reciever operating characteristic
SVC	Support Vector Classifier
SVM	Support Vector Machine
SWOT	Strengths, Weaknesses, Opportunities and Threats
TCI	Target controlled infusion system
TIVA	Total intravenous anesthesia
TN	True Negative
TNR	True negative rate
TP	True Positive
TPR	True positive rate
XGB	XGBoost

TABLE OF CONTENTS

1. INTRODUCTION

The number of annual surgical procedures performed under general anesthesia in Europe is estimated at approximately 29 million interventions [1]. In addition, the current aging of the population due to the increase in life expectancy worldwide, predicts an increase in the number of surgeries under general anesthesia in the coming decades [2].

The main purpose of general anesthesia is to relieve the sensation of pain during a surgical procedure; hence, through the administration of anesthetic drugs, a reversible state of unconsciousness, amnesia, analgesia and akinesia is induced [3]. However, this condition also involves significant changes in the physiologic balance of the body including cardiovascular, respiratory, renal, hepatic and endocrine systems, caused by anesthetic drugs [4]. Therefore, it is of vital importance to perform a complete monitoring of the patient in order to artificially maintain the optimal physiological conditions during general anesthesia.

One of the effects related to the administration of anesthetic drugs is the phenomenon of EEG burst suppression (BS), which is mainly caused by a too high dose of drugs [5], and according to some authors could lead to adverse outcomes [6][7]. These possible unfavorable effects prove the need of developing predictive models that can allow anesthesiologists to anticipate the phenomenon of BS, thus reducing the potential risks they may involve.

1.1. DESCRIPTION OF THE PROJECT

The clinical need of a system that could predict the occurrence of BS to avoid its appearance and the potential side effects associated was the starting point to develop the present project, which aims to generate an individualized predictive model capable of continuously indicating the likelihood of BS occurrence in the following two minutes. This way, during an intervention under general anesthesia, each second the model will display an index showing the probability of BS two minutes ahead of time, according to currently monitored physiological signs as well as continuous drug input.

The project has been carried out with the SPEC-M (*Systems Pharmacology Effect Control and Modeling*) research group in the Department of Anesthesiology at Hospital Clínic de Barcelona. The data used for the study belongs to the SPEC-M group and it has been collected since 2013 in the operating room number 4 of the CMA Unit of the Hospital Clínic. This operating room carries out gynecological procedures, and data is assembled from patients undergoing a general anesthesia surgery.

At present, the database consists of more than 1500 patients, with a large majority of women. However, in order to further increase its volume of data, at the beginning of this project a two-month stay was carried out in the aforementioned operating room to continue with the data collection.

1.2. OBJECTIVES

As already stated, the present study aims to conceive a prediction system for BS during surgeries undergoing general anesthesia. Hence, the main goal consists on building a model able to predict,

each second during an intervention, the probability of BS occurrence within two minutes. In order to accomplish this general goal, different tasks and more specific objectives are required, which are described below.

- Regarding data acquisition

  → Data acquisition from patients during general anesthesia surgeries

  A short stay in the CMA operating room will be carried out in order to increase the amount of available data of the SPEC-M database. Another objective is to get familiar with patient management as well as with the meaning of the kind of data collected that will be used in the development of the project. The duration of the stay allowed to increase the database up to 50 more patients, approximately.

  → Use of the intraoperative Data Acquisition Set

  The mentioned data acquisition implies the use of the intraoperative Data Acquisition Set, which allows to collect the monitored physiological parameters in real time during procedures under general anesthesia. During the automatic collection of data, a manual register of relevant events is performed.

- Regarding research in Biomedical Engineering field

  → Bibliographic research

  Understand the physiological causes and effects of BS, as well as the current studies and knowledge on this topic, including studies on BS prediction.

  In addition, research on the current State of the Art in patient monitoring during procedures under general anesthesia will also be carried out.

  → Review and improve current skills on data analysis, statistics and Machine Learning (ML) predictive models

  Gather information on the basis of ML models, along with the statistics, data analysis and information extraction behind them. This does not only imply knowledge acquisition on ML, but also on programming and its use in the present study.

- Regarding data management and the development of a BS predictive model

  → Elaborate a program for data analysis of the current database

  From an already existing database belonging to the SPEC-M group, all the data must be analysed as a means to determine which parameters must be taken into account for BS prediction.

Moreover, data must be processed in order to be properly prepared for further analysis, implying a checking of the available data as well as to perform en errors detection.

→ Elaborate a program for BS prediction

Develop different ML models so as to find the most accurate one, understanding as such the one that has a better performance.

Finally, this model must be applied to different patients with the aim of assessing which would be its functioning and behavior if tested in real patients.

1.3. METHODOLOGY AND OUTLINE

The duration of the present study is five months, from September 2021 to January 2022. This time period includes the execution of the entire project, from the development of the first general idea of the project, the exhaustive research on the topic and the two-month stay at the operating room for data acquisition, through the execution of the program for BS prediction with its validation, and ending with the drafting of the final project report.

Once the need for a BS prediction system was known, the present study was structured and different tasks were detailed in order to accomplish the previously mentioned objectives. These tasks were grouped and structured in six stages, shown in *Figure 1* and described below.

Figure 1. Methogology flowchart

First of all, the study started with the two-month stay at the CMA operating room for data acquisition. During this first stage, an initial bibliographical research was done, not only into anesthesia and BS, but also in ML basics and algorithms and a market analysis. This review has provided some basic knowledge on the current status on the subject, essential before continuing with the study.

Secondly, the processing of the existing database was performed by developing a *Python* script using the *Spyder* environment for programming. This program has been designed so it can collect all the data from the database and select relevant parameters and information and structure them in a suitable way for further usages, while eliminating unrelated data.

Next, another *Python* script was elaborated in order to build a BS predictive model. This program allows to develop different ML models, so an exhaustive comparation and analysis has been done with the aim of selecting the model presenting a better performance and accuracy. The chosen model has been later tested on a large number of patients as to simulate its behavior when applied in real-time.

Finally, the obtained results have been evaluated and discussed, and the final report of the study has been carefully written and detailed.

1.4. SCOPE AND SPAN

According to the previously mentioned objectives, and the tasks detailed in the previous section, the present study will include an extensive research on the State of the Art of studies regarding BS prediction on patients undergoing general anesthesia. If any, the current BS predictive models will be evaluated in order to know their usage conditions, characteristics and limitations.

After the market analysis, the study will contain different project management methods as to ensure the maximum efficiency and optimal conditions during the execution of the entire project. Concretely, an execution schedule will allow to define the different tasks to be carried out, specifying the time required for each of them and the order in which they must be performed, as well as their cost. Moreover, a GANTT diagram will be built in order to display the temporary planning of the project in a visual way, showing the different activities to be performed, simultaneously or not, throughout the project, along with the optimal deadlines for each task.

Once the background research is done and the current status on the topic is analyzed, the tasks involving the final goal of this project will be able to start. This crucial part of the project will include both the data processing and the building of several ML models for BS prediction, together with full-scale evaluation of each of the obtained models, allowing to make an evidence-based selection of the most favorable one. To end with, the resulting BS predictive model will be tested and valideted with an extense amount of patients, and the development and operation of the project itself will be evaluated. This last assessment will enable to compare the resources and deadlines initially established with those finally used; thus, it will allow to estimate the success of the project and whether it has been carried out in accordance with the expected.

1.5. LIMITATIONS

Regarding the limitations of the project, it should be noted that since the data collection has been done in the gynecology operating room of the Hospital Clínic of Barcelona, the vast majority of patients in the database are women and, also in a high proportion, of white people. Hence, the final results, although representing well the population in our environment, will not be representative for all global population nor extrapolable to any person, since no masculine patients are taken into consideration, and other ethnic groups are likely to be underrepresented in the used data. Nonetheless, since gender and ethnic group do not greatly affect brain function, data can be considered to accurately represent general patients behavior and results could be cautiously applied both to male patients and patients of different ethnicities.

In addition, given that the project is carried out in the framework of a Final Degree project, the final testing and validation of the obtained predictive model will not be carried out in a real-life surgery in an operating room with a physical patient, but will be implemented on a computer using patients from the current database. Nevertheless, the obtained result can be compared to what would be expected in a real-life application, so the validation can be considered reliable and meaningful.

Moreover, since all the data for the model developement is obtained from propofol-remifentanil induced general anesthesia surgeries, the obtained predictive model will only be useful for BS prediction during anesthesia of the same type.

2. BACKGROUND

2.1. GENERAL ANESTHESIA

General anesthesia is a drug-induced, reversible condition which involves certain behavioral and physiological characteristics, such as hypnosis, analgesia, amnesia and akinesia. These effects, resulting from the administration of anesthetic drugs, entail severe side effects such as cardiovascular inestability, respiratory depression, and an altered function of the thermoregulatory systems. Consequently, anesthesiologists must adopt measures in order to maintain the physiological conditions inside the normality range and homeostatic equilibrium [3][8].

In addition to the mentioned effects, general anesthesia also produces alterations on the EEG patterns, the most common of which is a gradual increase of low-frequency and high-amplitude activity as the level of general anesthesia intensifies. Due to the similarity of the EEG patterns under general anesthesia and the known ones from a sleep or coma state (*Figure 2*), general anesthesia is often considered a drug-induced reversible coma [9].

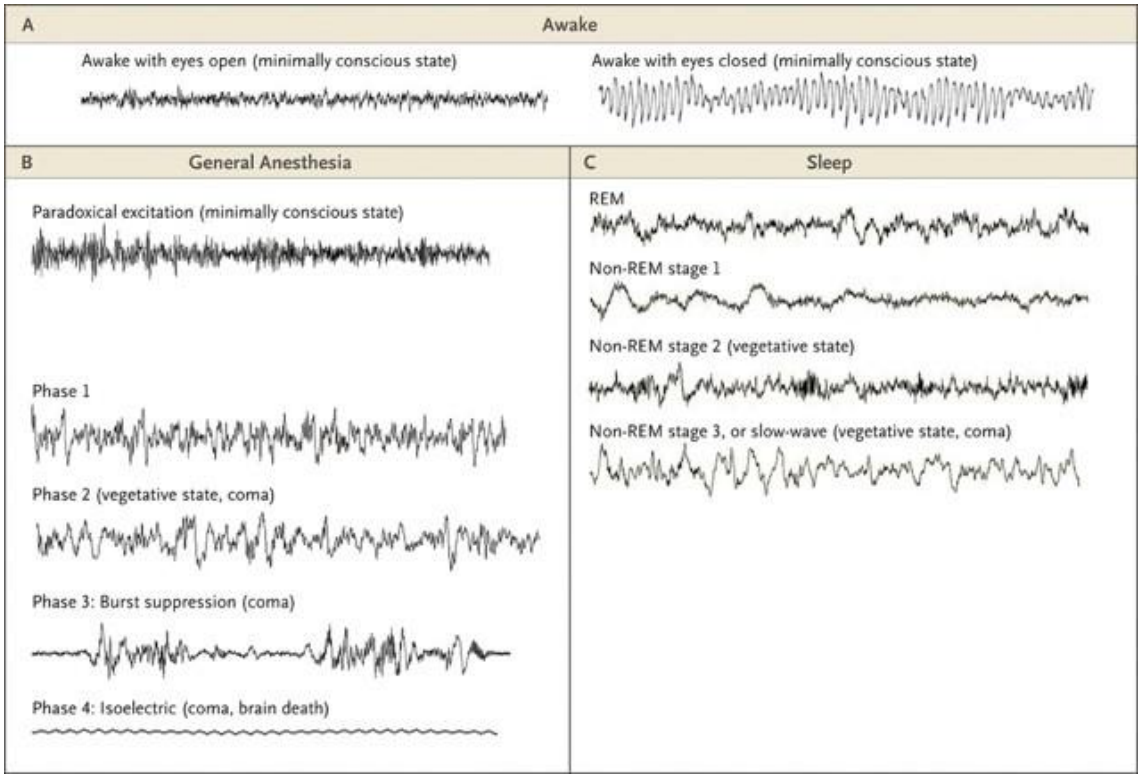

Figure 2. Electroencephalographic (EEG) Patterns during the Awake State, General Anesthesia, and Sleep [3]

The following sections will provide a brief review on the different effects desired to reach with anesthesia, along with a concise description of the drugs used to achieve these effects.

2.1.1. TARGET EFFECTS OF GENERAL ANESTHESIA

During a surgical procedure, anesthesiologists play a key role on protecting the patient from the aggressiveness of the procedure itself. In order to achieve the aforementioned reversible drug-induced coma state, different pharmacological effects must be combined.

2.1.1.1. HYPNOSIS

Hypnosis or unconsciousness is described as a drug-induce impairment of perceptive awareness, which involves the loss of cognitive functions required for responding to verbal, tactile or painful stimulation, comprising attention, perception and even spoken commands [10]. Hence, in clinical situations, hypnosis is assumed when patients fail to respond verbal stimuli or mild shaking [11].

Induction of hypnosis leads the patient to disregarding the external, generating a well-being and deep relaxation feeling followed by eyelid heaviness and regular breathing. Clinically, hypnosis state involves a reduction of respiratory rate and can be examinated by checking eyelid flickering, loose of muscular activity and relieve of facial tension, which usually causes dropping of the jaw and a slight opening of the mouth, sometimes including a deep yawning reflex [12].

Among the drugs able to produce hypnosis, $GABA_A$ receptor agonists, such as propofol, can be highlighted because its use is pretty much present in 90% of anesthesia induction. The hypnotic state can be monitored and analysed through EEG and EEG derived parameters [13][14], and it is crucial to continuously evaluate its depth in order to avoid an excessive or insufficient effect.

2.1.1.2. ANALGESIA

Analgesia is known as the absence or modulation of pain perception when recieving a noxious stimuli. This state is an essential effect to target due to the intensity of painful stimuli during a surgical intervention, which otherwise would be unbearable.

This analgesic effect is achieved by blocking the pathways responsible of transmitting the noxious stimuli recieved by the nociceptive receptors to the cortex [15]. A way of blocking this pathway is by the use of opioids such as remifentanil. Analgesia induces collateral effects such as respiratory depression and sometimes muscle rigidity; thus, it is essential to monitor this state [16]. Although it can not be directly measured, it is possible to indirectly evaluate it by monitoring hemodynamic and EEG changes.

2.1.1.3. AMNESIA

Amnesia is defined as a profund loss of memory and impossibility to retain information, which can be induced during surgery in order to improve the stress suffered by the patient in a surgical procedure situation [17]. Each anesthetic produces amnesia by affecting on distinct pathways and at different doses [18], but given the fact that patients who are truly unconscious are also on amnesia, this effect is maintained by ensuring unconsciousness [9].

2.1.1.4. AKINESIA

Akinesia or immobility is the loss of movement capacity, which is induced during the procedure to facilitate the surgeon's job, resulting in an improved exposure and precision [9], and also allowing the endothracheal intubation during the induction and maintenance of the anesthetic state [19]. This absence of movement is usually achieved by the administration of the so-called neuromuscular blocking agents, being the most commonly used succinylcholine and rocuronium.

There are different types of neuromuscular blocking agents, each with its own target and mechanism of action [20].

Immobility can be monitored through an electromyography (EMG), measuring the electrical activity of a specific nerve. Moreover, mechanomyography (MMG) and acceleromyography (AMG) can also be useful by measuring muscles actual movement.

2.1.2. ANESTHESIC DRUGS

The amount of agents cabaple of achieving the priorly described effects is large. Nevertheless, these drugs must be carefully selected and combined so they produce the most benefitial effect [16]; thus, it is of high clinical interest to find synergistic interactions between agents, since these would enable the use of smaller doses of each drug, consequently reducing potential side effects [20]. Depending on the effect they produce, anesthesics can be classified as shown in *Table 1*, where some of the most common agents can be observed.

Table 1. Common intravenous anesthetics classified according their effect [21]

Hypnotics	Opioids	Muscle relaxants
Propofol	Fentanyl	Cisatracurium
Thiopental	Sufentanil	Vecuronium
Etomidate	Remifentanil	Pancuronium
Ketamine	Morphine	Rocuronium
		Succinylcholine

Anesthetics can be classified according to its administration pathway; hence, they can be either inhalational or intravenous. The so-called total intravenous anesthesia (TIVA) refers to the exclusive use of the intravenous route for anesthetic administration.

Both techiniques are widely used and many indicators must be taken into account when selecting the most appropriate administration type for each patient and procedure. However, one factor to consider is that TIVA may present some advantages over inhalational anesthetics in terms of inflammatory and immunomodulatory effects, as well as better outcome and recovery [22][23][24].

Although intravenous drugs can be administered manually, target controlled infusion systems (TCI) are often used [25]. These use pharmacokinetic and pharmacodynamic models in order to compute the amount of anesthetic to deliver over time; this way, the desired level of anesthetic effect can be rapidly changed if required [26]. Compared to manual drug administration, TCI allows a more accurate control of the desired pharmacodynamic effect, along with shorter recovery time [27].

In the operation room from the CMA in Hospital Clínic, a TCI-TIVA induced anesthesia based on a synergic combination of propofol, remifentanil and rocuronium, when needed, is used for those surgeries requiring general anesthesia in order to achieve all the aforementioned effects. This combination of agents is known to produce a synergic effect, thus positively complementing each other in pharmacodynamic terms and even enabling a reduction of the required dosage for both drugs [20]. Due to their usage in the CMA in Hospital Clínic, these anesthesics are reviewed in the following sections.

2.1.2.1. PROPOFOL

Propofol is an intravenous agent which quickly produces an hypnotic effect, due to its lipophilic nature that allows it to rapidly cross the blood-brain barrier and reach the CNS. Hence, induction of hypnosis is fast, and its effect can be maintained either by continuous infusion or intermittent injection. Furthermore, propofol produces more immobility and is associated with rapid recovery and a low incidence of nausea and vomiting after the anesthesia.

However, in some cases it can produce pain or a stinging feeling during injection, and it produces a drop in blood pressure as well as a reduction in heart rate. However, these last consequences can be monitored and controlled by keeping a continuous infusion at low doses [28][29].

In order to achieve the whole anesthetic state, according to the abovementioned effects, it is usually used alongwith opioids, which provide analgesia.

2.1.2.2. REMIFENTANIL

Remifentanil is a fast opioid analgesic agent which, due to its rapid mechanism of action, is suitable for TIVA, in which effective agents are required [30]. It is associated with deeper analgesia and anesthesia, resulting in fewer responses to noxious stimuli and with a fast recovery time, among others [31].

Nonetheless, remifentanil causes respiratory depression as well as bradichardia and hypotension. These effects, though, can be controlled by controlling ventilation, which is already required in general anesthesia, and by keeping administration by infusion [32].

2.1.2.3. ROCURONIUM

Rocuronium is a non-depolarizing neuromuscular blocker used to achieve immobility and muscle relaxation during surgery. Despite its longer duration of action compared to other agents, its main advantage is its rapid effect and reversibility [33].

In procedures under propofol-remifentanil anesthesia, tracheal intubation without using muscular relaxant agents could cause hypotension and bradycardia; hence, the use of muscular blockers such as rocuronium remarkably reduce these effects [34].

Regarding its use in the CMA in Hospital Clínic, rocuronium is only used before tracheal intubation, while during laryngeal mask airway introduction it is not required due to its low complication incidence [35].

2.2. ANESTHESIA MONITORING

As stated previously, autonomous homeostatic control is lost under general anesthesia, so the administration of the correct anesthetics in the precise dose are fundamental for homeostatic equilibrium maintainance. Hence, monitoring of both for vital constants and drug doses is crucial to maintain the patient in the physiological normality range as well as to provide an individualized anesthetic management.

The following sections will be focused on briefly describing the different monitoring systems used in the CMA in Hospital Clínic during a surgical procedure under general anesthesia, classified as basic and advanced monitoring systems.

2.2.1. BASIC MONITORING SYSTEMS

2.2.1.1. ELECTROCARDIOGRAM

An electrocardiogram (ECG) measures the electrical activity of the heart through different electrodes placed on the skin of the patient. It provides direct and derived information on different parameters, such as heart rate, heart rythm and ST signals, among others.

Variations on these parameters can be due to changes on the homeostatic equilibrium, as well as on the hypnotic and analgesic effect, highlighting the need to monitor cardiac activity [36].

2.2.1.2. ARTERIAL BLOOD PRESSURE

Arterial blood pressure (BP) is also used for monitoring the cardiovascular function, as it continuously measures the blood flow pressure exerted on the arterial walls. It can either be measured in an invasive way by using intra arterial catheters, or non invasively by using cuffs [37].

As well as ECG, out-of-range values of BP might produce important physiological changes as well as effect on the overall anesthetic effect.

2.2.1.3. PULSE OXIMETRY

Pulse oximetry is a noninvasive technique for oxygenation monitoring. It quantifies oxygen blood saturation through a sensor usually placed on the index fingertip of the patient, by applying spectrophotometric methods able to measure hemoglobin levels and pulse rate, among others [38].

2.2.1.4. CAPNOGRAPHY

Capnography consists on monitoring the ventilatory function of the patient by measuring the concentration or partial pressure of CO_2 in respiratory gases. During anesthesia, capnography is used to ensure proper CO_2 elimination from the lungs, along with correct ventilation and pulmonary perfusion [39].

2.2.2. ADVANCED MONITORING SYSTEMS

2.2.2.1. ELECTROENCEPHALOGRAM

Electroencephalogram (EEG) is one of the main monitoring systems and plays a key role on anesthesia monitoring. It measures the electrical activity of the cortical area of the brain through several electrodes placed on the forehead of the patient; thus, it is a powerful continuous indicator of the anesthetic effect produced by the administered drugs, since these cause changes in the EEG waves.

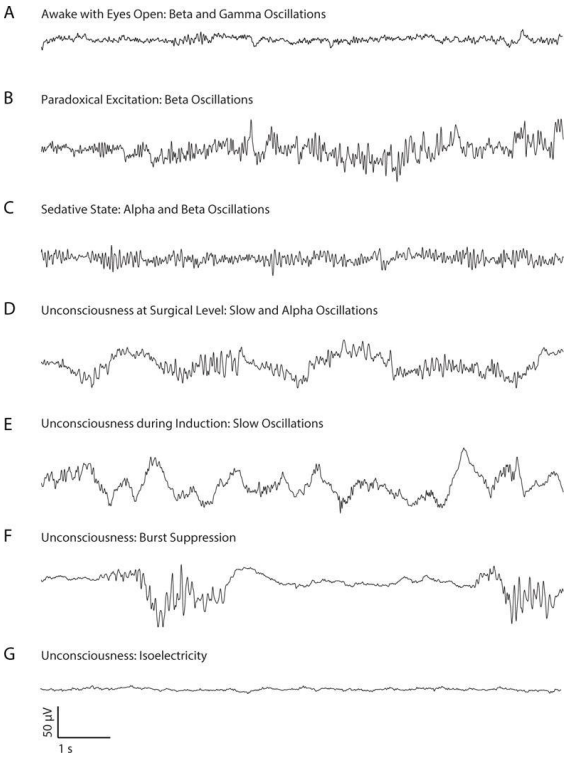

Figure 3. Raw electroencephalogram waveforms of propofol-induced anesthesia [41]

Under general anesthesia, each EEG waveform are related with different anesthetic states, as described in *Figure 3*. As can be observed, as a general rule, anesthesia is responsible of a gradual reduction on frequency along with an increase of high amplitude waves as the level of unconsciousness deepens [3].

As could be observed in *Figure* 3, at stages of deep anesthesia, a phenomenon called *"burst suppression"* can occur, which involves an alternating pattern of bursts (high frequency and high amplitude waves) and periods of isoelectric EEG with absence of electric activity [40]. This pattern, except in cases of brain ischaemia or other factors, is an indicator of a too deep anesthesia; thus, ideally it should be avoided.

Given the complicated interpretation of raw EEG waveforms in the time domain, data extraction from these signals is challenging. Therefore, using the principles of the Fast Fourier Transform (FFT), a frequency decomposition of the signals can be performed, obtaining a plot with the different frequencies on the x-axis and their power on the y-axis [41].

This enables to classify different waveforms according to their characteristic frequencies and, thus, identify anesthetic stages (*Table 2*).

Table 2. Spectral frequency bands derived from EEG and their clinical meaning [41][42]

Waveform	Frequency range (Hz)	Clinical interpretation
Beta (β)	13 – 25	Wide awake, conscioussness
Alpha (α)	9 – 12	Awake, relaxed, conscioussness
Theta (θ)	5 – 8	Light sleep, relaxed
Delta (δ)	1 – 4	Deep sleep

Hence, EEG monitoring provides information on which waveform is predominant over time; therefore, it is a powerful indicator of the anesthetic state of the patient.

2.2.2.2. BISPECTRAL INDEX (BIS)

As previously stated, raw EEG signals are difficult to evaluate, requiring high expertise from the anesthesiologist. To facilitate this analysis, advanced signal processing algorithms have been developed in order to extract parameters that could be easily evaluated. A popular EEG derived parameter is the well-known Bispectral Index (BIS), which uses a confidential algorithm able to measure the pharmacodynamic anesthetic effect on the CNS by displaying a single index value ranged from 0 to 100, corresponding values to "no brain activity" and "fully awake", respectively. In

propofol induced aesthesia, in order to maintain an optimal hypnosis effect, BIS values should range from 40 to 60 [43].

This value is obtained from four electrodes placed on the forehead of the patient and, besides the BIS index, BIS monitor also provide a predicted trend graph of the BIS values over time, the raw EEG signals in real-time, EMG activity, and different signal quality indicators, such as SQI, which the highest its value, the more reliable the BIS index [44].

Despite the powerful information provided by the BIS index, it is important to highlight the need to monitor all the previously mentioned parameters, since the overall anesthetic state can not only be monitored through EEG. Hence, hemodynamic variables and autonomic and somatic responses, among others, must also be considered before defining the effect of anesthesia on the patient.

2.3. BURST SUPPRESSION

As already stated, an often observed EEG pattern phenomenon is the so-called "*burst suppression*" (BS), which comprises high-voltage activity (bursts) and isoelectrical EEG (suppression of activity).

Outside from anesthesia, this pattern can be found in patients suffering from brain pathologies, such as coma, severe brain trauma, epilepsia, stroke, Ohtahara syndrome and hypothermia. As previously mentioned, it can be a consequence of a deep anesthetic effect. In fact, the absence of BS during sleep is a differential feature between sleep and general anesthesia [5]. Besides too intense anesthetic effect, known risk factors for BS include older age and previous comorbidities, such as COPD [45].

Although it can even be considered desirable in some specific conditions, such as for patients treated for severe seizures, when referring to surgeries under general anesthesia with too intense anesthetic effects BS has been associated to adverse outcomes [5][46].

According to different studies, it is believed that sustained states of BS during anesthesia might harm the brain, and it can also be related to postoperative delirium or even cognitive deficits [47][48]. However, other studies have found that EEG-guided anesthesia resulted in a decrease of BS time during surgery, but the incidence of postoperative delirium was not reduced [49]. Moreover, despite BS was associated with high anesthetic doses and comorbidities, it has been seen that EEG suppression can be a predictor of postoperative mortality only when accompanied with low mean arterial pressure [45].

Regardless of the controversy, it can be assumed that BS can be related to adverse outcomes, specially when combined with other factors, and that it does not have benefitial effects when referring to surgeries under general anesthesia. Hence, a predictive model for BS would be a powerful technique as to avoid EEG suppression patterns.

2.4. PREDICTIVE MODELS

Nowadays, technology and data processing systems have revolutionized the operating way in all fields, even making possible what seemed impossible. Regarding the medical field, these improvements are the result of combining medicine with computer sciences and engineering.

Diagnosis and health status forecasts of patients, alongwith monitoring and classification of biomedical signals are essential tasks of medical care with a common main goal: predicting the medical contitions of the patient in order to anticipate and avoid possible adverse outcomes. Hence, predictive models have been developed as to properly reproduce the patient's condition with the aim to make predictions [50].

Predictive models use known data, statitics and mathematical algorithms to predict outcomes. In general, this predictive models are known as Artificial Intelligence (AI), which can be defined as a field concering all computational techniques aiming to mimic and reproduce human intelligence for predicting results [51]. This offers a great number of benefits compared to human-performed predictions, such as flexibility and adaptability, followed by a more accurate pattern recognition through large amounts of data and variables. All these advantages are also accompanied with a fast computing capability, way faster than humans [52]. Inside AI, different disciplines can be distinguished. Among these, Machine Learning (ML) is considered to be an major subfield and it is widely used for predictive models in healthcare.

2.4.1. MACHINE LEARNING

By definition, ML refers to "*computational methods for improving performance by mechanizing the acquisition of knowledge from experience*"; hence ML models use already collected data in order to develop algorithms able to predict an outcome when new unobserved data is introduced [51]. When developing a ML model, the data used for the development of the algorithm is known as *Train data,* since it is used for training the model so it is later able to predict results. In order to test obtained models, a set of data known as *Test data* is used to test the performance and accuracy of the developed model. ML models can be classified in four large categories, which are supervised, unsupervised, semisupervised and reinforcement learning.

The main distinction between supervised and unsupervised learning models is the use of labeled training data. While supervised algorithms use data in which there is prior knowledge on the final output values that should be obtained, unsupervised models use not labeled data. A combination of these two types results in semisupervised algorithms. As for reinforcement learning models, they are based on algorithms which use a punishment-reward approach for making predictions on a dynamic environment [51].

Depending on the available data and on the expected performance and application of the desired model, one class or other will be used, or even a combination of them. Once the type of learning is known, however, there is a wide range of algorithms avaiable for each class, and the challenge relies on selecting the most optimal one. Since there is no rule on which algorithm to choose depending on each situation, many of them must be tried and compared in order to pick the best performing one.

Given the fact that the present study aims to build a BS predictive model, and basing the decision on the structure of the collected and available data, four different models have been developed using supervised learning. Hence, only the algorithms supporting these models will be described.

2.4.2. MACHINE LEARNING SUPERVISED ALGORITHMS

As mentioned, there is no way to know which algorithm will have a better performance prior to testing them. For this reason, several models must be built in order to compare their results and select the one with higher accuracy. In the present study, four different supervised learning algorithms have been chosen under the recommendations from the supervisors, which will be described below.

2.4.2.1. SUPPORT VECTOR CLASSIFIER

Support Vector Classifiers (SVC) are a class of Support Vector Machines (SVM), which are supervised ML approaches which analyse datasets in order to predict outcomes. Usually, this model is used for binary classification and prediction; thus, a SVC is able to classify data in two different outputs by examinating an input dataset. However, as seen in *Figure 4*, SVM can also classify data into multiple classes [51].

In order to understand the principle behind SVC algorithm, a large dataset with many different inputs and each input belonging to one of two classes must be assumed. With this dataset, the SVC model would develop an algorithm able to represent each of these inputs as a point in a space in which data for both classes would be divided by a gap. This way, by mapping new data into this space, the model is able to predict the class of this input according to the position of new data with respect to the gap. The space in which data is mapped will be *N-dimensional*, *N* being the number of features inside the input data [53]. The output of the model can either be the predicted class in which the input belongs to, or the probability for the data to belong to the different classes.

For now, the classification method described works by drawing a line between classes for a 2-dimensional space, a plane when referring to 3-dimensional spaces, or a hyperplane for higher dimensionalities; these are known as linear classifications. However, SVC models include a parameter known as *kernel*, which allows to modify the so-called linear classification in order to obtain a decision boundary with different shapes. Four types of *kernels* along with their classification behavior can be observed in *Figure 4*.

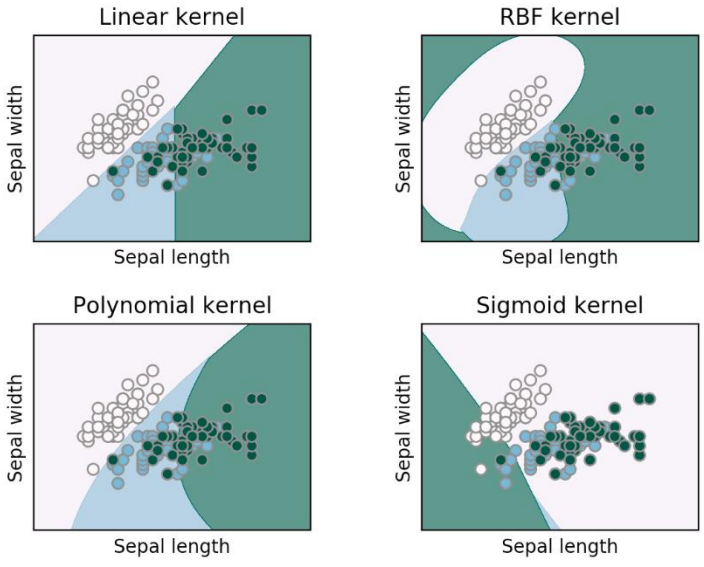

Figure 4. Decision boundaries approaches of different types of kernels for SVC models [54]

2.4.2.2. K-NEAREST NEIGHBORS

The K-Nearest Neighbors (KNN) supervised ML algorithm is a classification method based on the hypothesis that observations of the same class will have similar feature values, resulting in close points if each sample was mapped in a space.

Basing classification on this premise, new samples can be categorized by using the information and knowledge of the *k* nearest neighbors, being *k* a variable parameter. This way, an observation is mapped in the same space of other samples, and the distance from this point to the *k* nearest observations is used to weigh the influence of each neighbor so that a close neighbor influences more than a distant one [51][55].

There is no rule find the optimal *k* value; therefore, different values must be tested and compared in order to select the most favorable number of neighbors for the most accurate performance.

2.4.2.3. RANDOM FOREST

Random Forest (RF) is an ensemble supervised learning method, which are methods built from combining several other algorithms and classifiers in order to improve the performance and accuracy of the resulting model [56]. In the case of RF, the algorithm is based on decision trees classifiers.

Decision trees are predictive classification algorithms which split data into subsets and apply conditions on features to predict the output class. These conditions are structured in the form of a tree, and the final output for each combination of conditions, which is known as *leaf*, will classify observations according to the fulfilled conditions [57].

Regarding RF classifiers, these are built by creating several decision trees from the training data and computing an average of all their outputs to obtain the final RF predictive model. This way, the final RF model takes into consideration different condition combinations in order to provide the final classification; thus, accuracy gets increased in comparison with using one simple decision tree.

Besides classification, RF models provide a deep analysis on the weight of each feature for the final classification decision, which is useful in terms of knowing which variables are more important and decisive for classification.

2.4.2.4. XGBOOST

Extreme Gradient Boosting (XGBoost, XGB) is also an ensemble supervised learning method based on decision trees classifiers. Instead of building a bunch of treen in parallel and combining them at the end, XGBoost creates one tree at the time, and the created tree is added to the previous ones if there is any missing data [58].

This is done by differently weighting correctly classified and missclassified samples. This method usually results in an important reduction of computing time and, like RF classifiers, it performs a feature importance analysis in order to recognize those variables with higher relevance for the final classification [59].

2.4.3. MACHINE LEARNING MODELS PERFORMANCE EVALUATION

Once ML models are built and implemented, their performance needs to be properly evaluated in order to ensure a correct predictive behavior and to know the fiability and reliability of the predictions. In order to test a model, data different from the one used for the training must be used. Otherwise, the model would be overfitted, meaning that the predictions perfectly match with the expected results because the data for the test would have also been used for training the model; this must be considered mistaken since predictive models can not be perfect.

In order to evaluate classification models, different methods and metrics can be used. The ones employed in the present study are described below.

2.4.3.1. ACCURACY SCORE

Accuracy score is a value representing the number of correctly classified predictions with respect with the overall predictions; thus, it is computed by dividing the number of correctly predicted samples over the total number of predicted samples.

According to this calculation, the higher the accuracy score, the better the performance of the model and the more reliable their predictions.

2.4.3.2. CONFUSION MATRIX

Although accuracy gives an easy and fast interpretation on the performance of the model, it does not represent some prediction differences that, when talking about ML applied to medicine, could involve important results. This way, it is often important to differentiate, among the correctly predicted observations, the rate of true positive (TP) and true negative (TN) observations. Moreover, it is also useful to distinguish, among the misclassified samples, the rate of observations classified as positive but should be negative, known as false positive (FP), and the ratio of samples classified as negative but should be positive, the so-called false negative (FN).

These four scores can be displayed in a visual and simple way using a confusion matrix (CF), which is structures as showed in *Table 3*.

Table 3. Confusion matrix layout comparing the real values with the predicted ones

	Positive class	Negative class
Positive prediction	TP	FP
Negative prediction	FN	TN

From CF, different rates can be obtained in order to evaluate the above mentioned concepts. The most simple one is the previously described accuracy which, using the newly described concepts, can be obtained as shown in *Equation 1*.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad \text{Eq. 1}$$

Another useful parameter is the one indicating the ratio of correctly predicted positive outcomes over the entire number of actual positive outcomes. This, which is known as True Positive Rate (TPR) or sensitivity, can be computed according to *Equation 2*.

$$Sensitivity = \frac{TP}{TP + FN} \qquad \text{Eq. 2}$$

Along with the TPR, it can also be obtained a True Negative Rate (TNR), also referred to as specificity. Analogously, this ratio can be obtained as in *Equation 3*.

$$Specificity = \frac{TN}{TN + FP} \qquad \text{Eq. 3}$$

2.4.3.3. ROC CURVE

Given the fact that for the desired goal of the models built in the present study the required output of the models should not be a class, but a probability of belonging to one class or the other, another evaluation method has to be considered.

When obtaining a probability, this score has to be used to predict, with more or less certainty, if the observation will belong to one class or another. This classification is done by setting a threshold at which the values above it will be classified to one class, while the scores below will belong to the other class. As can be expected, though, the performance of the model, as well as the accuracy, sensitivity and specificity concepts previously described, will depend on the stablished threshold. In order to evaluate the performance of the model over different thresholds, a reciever operating characteristic (ROC) curve can be obtained.

This ROC curve graphically plots relationship between sensitivity and 1 – specificity for all the values the threshold can take. As a way to measure and assess the overall performance of the model, the area under the ROC curve (AUC). The values of AUC will range from 0 to 1, being 1 an ideal and perfect model and 0 a perfectly inaccurate model. Between these limits, a value of 0.5 reflects that half of the predictions are correct, thus indicating that the model randomly classifies the observations, resulting in a useless but not adverse model. Hence, a predictive model is expected to have values between 0.5 and 1. Inside this range, scores between 0.7 -

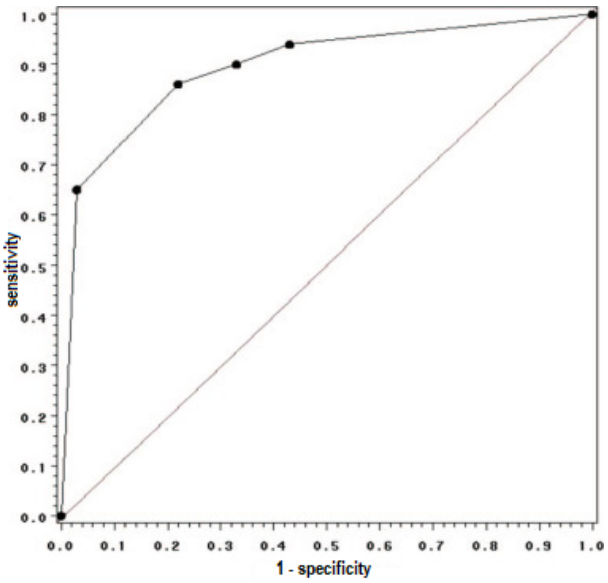

Figure 5. Example of ROC curve [60]

0.8 can be considered acceptable; values ranging 0.8 - 0.9 are considered excellent; and values above 0.9 can be assumed as outstanding. Nonetheless, it must be pointed that these ranges and limits are general approximations; in each particular scenario these assumptions could be considered differently [60].

In *Figure 5* an example ROC curve is represented, and its corresponding AUC would be the entire area comprised below the curve. As can be observed, there is a straight line from the origin to the top right corner which represents the shape of the ROC curve if the AUC was 0.5. This is always displayed as a fast and simple way of assessing the actual ROC curve for the evaluated model, as if its curve is above the line, the model will have an AUC over 0.5. The model represented within *Figure 5* has an area over 0.5, since the ROC curve is above the random-classification line.

2.5. STATE OF THE ART

General anesthesia is the use of anesthetic drugs in order to induce a reversible status of unconsciousness, amnesia, analgesia and akinesia [3]. Due to changes and side effects on the homeostatic equilibrium, it is crucial to monitor the behavior of the hypnotic agents as to maintain the patient in a safe normality range. Therefore, during a surgery under general anesthesia, different systems are used to control the physiological state of the patient by measuring and monitoring parameters, such as ECG, arterial blood pressure, pulse oximetry, capnography, EEG, and the EEG-derived BIS index.

Among the possible adverse outcomes resulting from the administration of drug and the induced anesthetic state, some studies suggest that the phenomenon of BS, often observed during anesthesia, might be involved in postoperative delirium, cognitive deficits, or even considered a predictor of postoperative mortality if accompanied by some other factors [5][45][46][47][48]. Nonetheless, other studies do not show any direct relation between BS during anesthesia and postoperative delirium [49].

All in all, there is no evidence that BS suppression may imply some advantageous or benefitial effect on the patient when occurred under general anesthesia, and all the studies on the topic point to adverse results under the presence of BS, or no effects at all. Hence, according to the present knowledge on the subject, it would be desirable to avoid the occurrence of BS during surgeries under general anesthesia.

In this direction, the final goal of the present study is to build and implement a BS predictive model as a way to provide an individualized anesthesia as well as more protection both during and after the surgery under general anesthesia.

2.5.1. BURST SUPPRESSION

BS is known to appear in patients suffering from comorbidities, and in situations in which too high doses of anesthetics have been administered. Due to the impossibility to change the comorbidities of the patient, the target for avoiding BS must be focused on the administered dose of drugs.

Existing EEG monitors include a measure of the burst suppression rate (BSR or SR), which is an EEG-based parameter which measures the ratio of suppressions in a period of time, considering a

suppression those voltages lower than 5μV within the EEG signal [61]. This parameter is a powerful indicator during anesthesia, since it reflects the amount of BS at the present time and the anesthesiologist can adapt the drug administration as to reduce the BSR. However, it would not be useful in order to completely avoid and anticipate to BS as the value is in real-time; thus, when it indicates that there is BS it is because it has already appeared.

Here lies the need of constructing predictive models able to provide enough information to the anesthesiologists so they can decrease and adapt the administered drug doses in order to anticipate to BS and, thus, reduce possible adverse outcomes.

2.5.2. PREDICTIVE MODELS

As for now, after an exhaustive reaserch on the topic, no external studies regarding predictive models for BS under general anesthesia have been found. In this aspect, the importance of developing a BS predictive system in order to cover this need is clear.

Within the SPEC-M research group, two similar studies have been carried out in the frame of Final Degree Projects. The first one was named *"Burst Suppression Occurrence under General Anesthesia",* done in 2016 by Carlos Gil, and the last one was the project written by Marcel Pons in 2019, *"Occurrence of the EEG Burst Suppression Pattern during General Anesthesia and its Hemodynamic Effects".*

Both works develop a predictive model for BS occurrence by using a Logistic Regression model. However, no studies using other ML algorithms have been carried out; hence, and due to the aforementioned fact that there is no ultimate method of knowing the best-performing ML algorithm, but many have to be tested and compared, the present work aims to build and implement a BS predictive model based on different ML methods for surgeries under general anesthesia.

3. MARKET ANALYSIS

As previously mentioned, this study aims to cover the need to obtain a system capable of predicting BS with the purpose of being able to anticipate its occurrence during a surgery under general anesthesia and, consequently, avoid possible adverse outcomes related to it both during and after the intervention. According to these objectives, the present section focuses on the applications and uses of the BS predictive model, as well as to future prospects.

3.1. POTENTIAL USERS

The BS predictive model is a system able to indicate the probability of BS occurrence within a period of two minutes. Due to the known possible adverse outcomes when BS appears on a surgery under general anesthesia, this model would be a powerful tool in these procedures. Therefore, its role in the operating room would give relevant information to the anesthesiologist on the future anesthetic effect of the currently-administered drugs; this way, the physician would be able to control and adjust the anesthetics infusion in order to anticipate and prevent a possible occurrence of BS.

Given that all surgical procedures used for the development of the BS predictive model were surgeries under propofol-remifentanil induced general anesthesia, the final obtained model will only be applicable in same-conditions situations. Similarly, since all the patients from the available dataset are women, the resulting model from this study would fit optimally to new data coming from patients with similar characteristics although this does not prevent crom cautiously using it under different conditions.

Finally, one of the parameters that the model requires to optimize its predictions is the age of the patient and the monitoring of different physiological parameters, since the model has been trained with data including these. These parameters are the mean non-invasive arterial blood pressure (NIBP), the effect-site concentrations of propofol and remifentanil, the BIS index and the ECG heart rate; thus, ECG, EEG, and arterial blood pressure monitoring will be indispensable, although they would already be a must for simply correctly monitoring anesthesia during the intervention.

3.2. MARKET EVOLUTION

The applications of ML predictive models are being widely spread into many different fields, among which medicine is not an exception. Their main usages, however, include diagnostic analysis of chronic diseases [62] and adverse outcomes prediction, but few is done regarding intraoperative prediction of adverse events, such as BS. In fact, there is no commercially available solution to predict BS in the clinical setting where only clinical eye of the attending physician is the only control method.

As for anesthesia monitoring, EEG-guided procedures enable to monitor the effects of drugs and control the anesthetic status of the patient. Therefore, all general anesthesia surgeries include an EEG device and, increasingly, a BIS monitor, so all the surgeries which meet the above-stated conditions already have the infrastructure to use the BS predictive model presented with this study.

3.3. FUTURE MARKET PROSPECTIVES

Given the fact that BS does not only occur in women nor in propofol-remifentanil induced general anesthesia, more data could be collected in order to enable te use of the developed model in other situations, such as in surgeries in men and for procedures under general anesthesia induced for different anesthetic agents.

In order to expand the uses of the model, therefore, a large amount of data should be acquired from male patients, and from patients under general anesthesia induced by other anesthetics.

Once the data has been collected, the current database could either be expanded to include data from men or surgeries induced by other anesthetics, or different models could be created using the same methodology as the current one.

4. CONCEPTION ENGINEERING

As previously stated, due to the lack of predictive power of the current intraoperative anesthesia monitors, the development of a BS predictive model would be a great advance in order to anticipate and avoid BS with its respective possible side effects both during and after a surgery under general anesthesia. The options to carry out the model building are wide and diverse; thus, the present section aims to describe different available options, as well as to explain the reasons behind the final options selection.

4.1. OPTIONS DESCRIPTION

Prior to the model building, different execution options and aspects of the development itself must be considered and evaluated. Therefore, the different programming languages available, as well as the different existing ML algorithms must be assessed.

4.1.1. PROGRAMMING LANGUAGE

According to the programming skills and knowledge on different programming languages, the three considered languages for the script used both for the data processing and the predictive model building and testing are the following:

- MATLAB

  MATLAB is a programming language developed by MathWorks consisting in a free numeric computing environment widely used for data manipulation, numeric computing, algorithms implementation and plotting of data, among many others. Besides the language itself, it includes a wide range of complementary toolboxes which enable different scientific and engineering applications [63].

- R

  R is free software generally used for statistical computing. Its computing abilities can be expanded by using open-source user-created packages available online, which offer a wide range of tools for statistical and data analysis, graphical visualization, modeling and predicting, among others [64].

- Python

  Python is a free dynamic programming language used for multiple programming situations. Alongwith a large amount of libraries and modules, the capabilities of Python can be expanded for different applications such as data analytics and processing, graphical interfaces and image processing, along with others [65].

Regarding predictive models building, testing and implementing, the three programming languages offer similar characteristics and usages; thus, all three could be considered valid for the purpose of this study.

4.1.2. MACHINE LEARNING ALGORITHMS

As previously mentioned, and according to the structure of the available data, supervised ML algorithms must be used. These algorithms are classified in two classes: classification and regression. The first one referres to the algorithms able to predict class labels for a given data. As for regression algorithms, these are those which predict a continuous result from a given set of variables. Therefore, the main difference between both classes is that classification algorithms predict class labels while regression algorithms predict a continuous quantity [66].

The desired model to construct in the present study aims to predict whether BS will appear or not within a period of two minutes, and its occurrence probability. At first, this may seem a regression problem, since the desired output is a continuous value. However, what the model should predict is if there will or there will not be BS and the certainty of the prediction, not the BSR value, which is what the output would indicate if a regression algorithm were used. This being so, the algorithms used for building different models in order to at the end compare them and select the one with best performance will be based on a classification analysis.

There are several different classification algorithms, all of which would be suitable for the present case. However, given the fact that they are all expected to give similar results, and due to time and computational limitations with the available equipment, only four of these algorithms will be used, and the obtained results will be considered representative. The most common existing classification algorithms are listed below.

- Naive Bayes (NB)
- Linear Discriminant Analysis (LDA)
- Logistic Regression (LR)
- K-Nearest Neighbors (KNN)
- Support Vector Classifier (SVC)
- Decision Tree (DT)
- Random Forest (RF)
- Adaptive Boosting (AdaBoost)
- Extreme Gradient Boosting (XGBoost)
- Stochastic Gradient Descent (SGD)

All these are able to predict if BS will occur in a 2-minute period when a set of data is introduced, as well as to obtain the probability of the input data to be labeled as BS or not, thus indicating the reliability of the predicted class.

4.2. OPTIONS SELECTION

Once the different options for carrying out this study have been presented and analyzed, the most optimal ones, the most favorable ones in terms of accuracy and optimization of time and resources must be carefully selected.

Regarding the programming language, given that all three options would be suitable for the study and none of them presents advantages over the others in terms of data processing and predictive model building and performance, the selection criteria was based on previous experience with each

of the languages. This way, Python has been preferred to be the programming language used for the code to be developed.

As for the different ML classification algorithms, the selection has been made according to the software-developing supervisor criteria and having into consideration which are the most widely used algorithms. Besides, since the other studies on the topic both built a BS predictive model by using LR, it has been considered more appropriate to use different algorithms. Therefore, the selected algorithms used to build four different BS predictive models in order to compare their performances have been KNN, SVC, RF and XGBoost.

On the whole, a Python script will be developed in order to properly process the available dataset and four BS predictive models will be built by using the KNN, SVC, RF and XGBoost supervised classification algorithms. These models will later be tested, compared and assessed in order to finally select the most accurate and best performing one.

5. DETAILED ENGINEERING

The present section describes in detail the followed steps for the project development. Therefore, the data acquisition and processing methods, the construction and comparison of the different predictive models, and the implementation of the model with better results will be thoroughly detailed below.

5.1. DATA ACQUISITION

The dataset used for the present study has been provided by the SPEC-M research group from Hospital Clínic de Barcelona. Data is recorded anonymously from patients undergoing propofol-remifentanil induced general anesthesia surgeries in the operating room number 4 from the CMA in Hospital Clínic, which is mainly assigned for gynecological procedures, under the authorization of the Ethics and Clinical Research Committee (CEIC) of the same hospital (Ref. nº 2013/8356); therefore, the vast majority of data belongs to women.

The data collection began in 2013 and by the date it contains about 1500 patients. However, everyone involved in the SPEC-M research group, as well as the students who collaborate with it, are committed on keeping it on growing. With this goal, a two-month stay at the formerly mentioned operating room was done from September to Novemeber 2021, coinciding with the beginning of this project.

Data recording is performed using the monitors already employed to monitor the patient and the anesthetic effect, which collect different patient parameters every second throughout the entire intervention. At the end of the procedure, the information of all the different variables over time is synchronized and put together, resulting in a single record per patient which is stored in a computer as a CSV file.

The different monitoring and recording equipments are arranged in the anesthesiology control tower of the SPEC-M research group. This structure, which can be observed in *Figure 6*, includes three monitoring devices and the TCI-TIVA system. *Table 4* includes a description of each of these monitors along with the adquisition techique used and the commercial company of the deviced used in the mentioned operating room.

Table 4. Monitoring devices description and their commercial company

DEVICE	ACQUISITION METHOD	COMMERCIAL COMPANY
BIS VISTA® Bilateral Monitoring System	Electrodes placed on the forehead	Aspect Medical Systems, Inc., Norwood (MA), USA
Conox®	Electrodes placed on the forehead	Quantium Medical, Mataró, Spain
Dräger Infinity® Gamma	Three electrodes placed on the chest, a cuff on the arm and a pulsioximeter on the index finger	Dräger, Lübeck, Germany
TCI-TIVA Orchestra® Base Primea and two Module DPS	Intravenous infusion	Fresenius Kabi, Homburg, Germany

As priorly stated, each monitoring device allows to measure and control different parameters, which are recorded in the aforementioned CSV file for each patient and stored in the SPEC-M database. The monitored parameters are listed and described in *Table 5*, alongside with the device that allows its register. Demographic parameters such as age, gender, weight and height are previously known variables which, since are required and manually introduced to the TCI-TIVA system, in *Table 5* are considered among the variables obtained through this device, although they are not actually monitored. As for the patient ID and the temporal variable, since their registration is automated and they are not monitored nor used by any device, no monitor is linked with them.

Besides these automatically monitored parameters, several events that occur during the anesthesia could be relevant when performing studies on the obtained data, such as the insertion moment of the LMA, the presence of movement and the administration of other drugs, among others. Therefore, they are manually registered using an adapted keyboard and synchronized with the other variables through the Rugloop® software.

Table 5. Monitored parameters and their description classified according to their acquisition device

DEVICE	PARAMETER	DESCRIPTION
-	Patient ID	Identification number of the patient
-	Time	Time register throughout the procedure (s)
BIS VISTA® Bilateral Monitoring System	BIS	Bispectral index
	EMGBIS	Electromiogram intensity index
	BSBIS	Burst Suppression index
	SQI09	Signal Quality Index of the BIS
Conox®	qCON	Hypnotic effect index
	qCONEMG	Electromiogram intensity index
	qCONBS	Burst Suppression Index
	qCONSQI	Signal Quality Index of the qCON
	qCONqNOX	Analgesic effect index
	nHz	EEG spectrum (n = 0-127 Hz)
Dräger Infinity® Gamma	HR	Heart Rate
	NIBPsys	Systolic blood pressure (mmHg)
	NIBPdia	Diastolic blood pressure (mmHg)
	NIBPmean	Mean blood pressure (mmHg)
	RR	Respiratory Rate
	SPO2	Oxygen saturation
TCI-TIVA Orchestra® Base Primea and two Module DPS	CpRemi	Remifentanil plasmatic concentration
	CeRemi	Remifentanil effect-site concentration
	RateRemi	Remifentanil infusion rate
	VolRemi	Remifentanil infusion volume
	CpPropo	Propofol plasmatic concentration
	CePropo	Propofol effect-site concentration
	RatePropo	Propofol infusion rate
	VolPropo	Propofol infusion volume
	Age	Current age of the patient
	Weight	Current weight of the patient
	Height	Current height of the patient
	Sex	Gender of the patient
Rugloop® software	Events	Surgery-related events

Figure 6. Anesthesia control tower with the monitoring and anesthesia-infusion devices (left) and adapted keyboard for surgery-related events registration (right)

This exhaustive data acquisition entails the obtention of a high-resulution and semi-automatically created dabasase including a CSV file per patient with information on the previously mentioned parameters over time. From these, a subset of data including 457 patients has been used for the BS predictive model building.

5.2. DATA PROCESSING

Before building a ML predictive model, all the data both for the training and the testing sets must be accuretely processed and modified in order to have it structured according to the models requirements. This section includes a complete description of all the used methods and decisions taken with the final goal of achieving a data frame structured in conformity with the ML algorithms demands. In parallel to the study, and for purely informative purposes, the incidence of BS in the studied population has been estimated and analysed.

5.2.1. DATA FRAME CONSTRUCTION

ML predictive models require a well-structured data frame containing all the different measured features, which are arranged as columns of a matrix, and a large amount of observations of these features, which are organized in rows of the same matrix. Moreover, in the case of supervised algorithms, an extra column corresponding to the known label for each observation is needed for the training dataset. This is the case for the present study, in which the ML predictive models will be based on supervised algorithms and, thus, the training of the model requires a data frame with a large amount of observations and the known output for them.

Besides, each second of every patient is considered an independent observation, since the prediction of BS will depend only on the parameters measured at the time; hence, the probability of BS occurrence within two minutes will be displayed each second during the intervention and will depend on the monitored variables at that very moment.

Therefore, each patient provides as many observations as the duration of its intervention in seconds. This assumption makes it irrelevant to know to which patient correspond each observation, since the desired model aims to predict the occurrence of BS by only considering the

monitored variables at a certain moment; thus, all the observations from all the available patients can be put together in the same data frame.

As a result, an initial large data frame can be obtained by constructing a matrix with all the features as columns, and the data from all the available patients concatenated as different rows.

5.2.1.1. LABEL OBTENTION

In light of the above, the training data frame of the model needs a large matrix with the known output for them, the so-called label. According to this, the input features to the model will be all the measured parameters, and the expected output will be the BS occurrence probability. Hence, as supervised models, the required data frame will include all the recorded variables as features, and a new variable indicating the occurrence or not of BS after two minutes, which will be used as label. This way, the model will be trained by analysing all the features recorded at the same time and their resulting label.

This new variable, referred to as *Future_BIS*, which is represented as a new column in the data frame, will be obtained by taking the SR parameter and applying a two-minute shifting. By doing this, it is important to highlight that the 2 last minutes of the intervention will not have this parameter, so these observations should be removed, as well as the 120 first values of the SR parameter.

Given the classification nature of the selected predictive models, the label must be binarized in order to represent two classes: BS occurrence and no BS. Therefore, all values greater than 0 will be considered as BS occurrence and replaced with a 1, while scores equal to 0 will not be modified and will represent those observations with no BS within two minutes.

5.2.1.2. SIGNAL QUALITY ASSESSMENT

Due to limitations in the acquisition and measuring devices, some obtained values might be noisy or not much accurate; thus, it is important to drop them out of the data frame since could lead to erroneous results. This reliability of the recorded data is measured by the signal quality index (SQI); according to its score, and considering a signal greater than 60 to be sufficiently valid, those observations below this threshold will be disregarded.

5.2.1.3. GENDER SELECTION

As mentioned earlier, the data has been collected and recorded from patients undergoing propofol-remifentanil induced general anesthesia in the gynecology operating room of the CMA in Hospital Clínic; for this reason, the vast majority of the data correspond to women. However, some procedures belonging to men have also been monitored. Since the gender proportion is not representative, it has been considered more convenient to neglect those patient recording corresponding to men and focus the whole study in data from women.

With this purpose, the variable indicating the gender of the patients has been analysed in order to detect and exclude those coming from procedures performed on men.

5.2.1.4. *FEATURE SELECTION*

The recorded and collected data from the patients includes a large number of variables regarding the overall anesthetic and physiological status of the patient during the intervention. However, when aiming to predict BS, some of them become meaningless or dispensable, either due to their dependence on other variables, thus providing redundant information, or due to its irrelevance in the appearance or not of BS according to the criteria of the project supervisor.

Thus, a feature selection has been carried out, reducing the amount of variables from 34 to 6; therefore, the resulting data frame consists of 7 columns, 6 of them corresponding to features, and the last one to the previously obtained label, as can be observed in *Table 6*.

Table 6. Feature and label structure of the data frame

FEATURES						LABEL
Age	Remi_Ce	Propo_Ce	BIS	ECG_HR	NIBP_Mean	Future_SR

5.2.1.5. *OBSERVATIONS SELECTION*

Although it would be ideally expected to have information on all of the aforementioned selected parameters at each second throughout the intervention, this assumption would be far from reality.

The propofol and remifentanil effect-site concentrations, as well as the heart rate measured with the ECG (ECG_HR) are intermittently-monitored variables which, with no detectable pattern, at some seconds are not recorded, so some observations have features with null values. Therefore, taking into account that the number of observations in which these are not monitored is much smaller than the amount of observations in which data from all the variables is available, and considering more suitable to use less amount of data instead than *inventing* the missing values by interpolation or other methods in order to fill the blanks, it has been concluded that these observations can be ignored.

Moreover, given the fact that each monitoring device starts its recording at different times, depending on the moment of placement of the different electrodes, and taking into consideration that drug infusion starts later than the monitoring, some observations belonging to the beginning of the surgery will also be neglected, since they have null values for some of the features. Similarly, observations from the end of the intervention will also be disregarded, since the anesthetic infusion finishes before the monitoring devices, and the electrodes are not all removed at the same time.

This removal of certain observations will be performed by deleting those rows of the data frame having some null value for any of the features.

5.2.1.6. *COMPENSATION OF THE PROPORTION OF DIFFERENT LABELS*

Ideally, the training set for ML predictive models should have a similar proportion of observations for each label in order to equally train the model for predictions on all the classes. According to the nature of BS periods, their length and abundance during surgeries usually represent a very small proportion of the total intervention time, making it difficult to obtain similar proportions for both labels.

An option would be to reduce the number of observations with no BS to the same number of observations with a positive BS label. However, this would drastically reduce the size of the data frame and, given the importance of having a large amount of different data for a better training of the models, this solution has been rejected.

In order to compensate as much as possible for the difference between the number of observations for each class, those patients without any BS period have been removed from the study; this way, the number of observations from the no-BS label was considerably reduced.

5.2.1.7. SCALING

As required by the ML predictive models, all data must be numerical and inside a same range, usually between 0 and 1; this way, all features will be equally weighted and predictions will not be interfered for the measuring scale of each variable.

Hence, the scaling of the data frame was performed per rows and each value was normalized according to *Equation 4*.

$$z_i = \frac{x_i - \min(x)}{\max(x) - \min(x)} \qquad \text{Eq. 4}$$

Where x is the entire set of values for a particular feature, x_i represents a specific value from x, and z_i corresponds to the new scaled value for x_i.

5.2.1.8. FINAL OVERVIEW OF THE INCLUDED PATIENTS AND DATA

According to the previous considerations, the used data for the study has been considerably reduced from the initial amount of observations. However, given that each second of every patient is a different observation and, hence, the starting number of observations was immense, the final amount of observations was still large enough. A schematic overview of the followed criteria and considerations for the final data frame obtention can be observed in *Figure 7*.

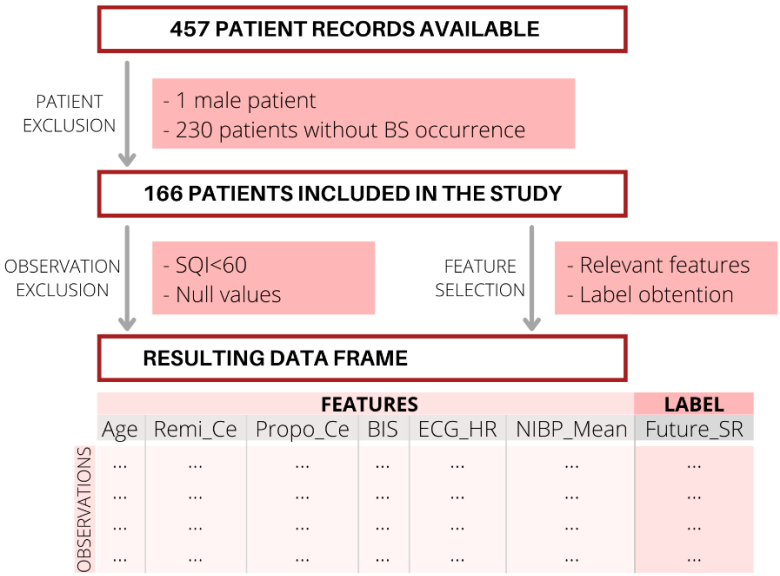

Figure 7. Overview of the exclusion and selection process of data

After the processing of the initial data, the resulting data frame included the information of the different features for each observation that would later be used for the predictive model building. The structure and arrangement of this matrix are shown in *Figure 8*, in which the different features and the label conform to the columns of the data frame, and the rows include all the remaining observations. Due to the large number of observations available, only a small part of these is shown in the figure as an example.

Index	AGE	REMI_CE	PROPO_CE	BIS	ECG_HR	NIBP_MEAN	FUTURE_SR
322	0.238095	0.621762	0.376028	0.269527	0.265823	0.50289	0
323	0.238095	0.621762	0.376028	0.250825	0.265823	0.50289	0
324	0.238095	0.621762	0.376028	0.227723	0.261603	0.50289	0
325	0.238095	0.621762	0.376028	0.213421	0.261603	0.50289	0
326	0.238095	0.621762	0.376028	0.212321	0.261603	0.50289	0
327	0.238095	0.621762	0.376028	0.223322	0.261603	0.50289	0
328	0.238095	0.621762	0.376028	0.254125	0.261603	0.450867	1
329	0.238095	0.621762	0.376028	0.257426	0.261603	0.450867	1
330	0.238095	0.621762	0.376028	0.250825	0.261603	0.450867	1
331	0.238095	0.621762	0.376028	0.238724	0.261603	0.450867	1
332	0.238095	0.621762	0.376028	0.240924	0.257384	0.450867	1
333	0.238095	0.621762	0.376028	0.245325	0.257384	0.450867	1

Figure 8. Structure of the input matrix for the ML models

5.2.2. BURST SUPPRESSION INCIDENCE ASSESSMENT

Given the fact that the presence of BS at some time during propofol-remifentanil induced general anesthesia has been related to age, all the patients in the database were classified into five groups of age and the occurrence of BS has been examined by using the SR parameter as well as the age of the patients.

The criteria for determining BS appearance was limited to looking at whether the SR parameter acquired values greater than 0 for periods greater than 15 seconds. Simplifying, the length of periods with BS, the number of periods, and its intensity have not been quantified in any way.

5.3. OUTLIERS ANALYSIS

Once the matrix was created and processed, an outliers analysis was performed to identify possible anomalous observations in the data frame. This process was carried out to reassure that there are no erroneous data caused by artifacts or noise in the measurement of the variables.

Outliers can be removed by not considering the tails of a normal distribution of data; thus, data has to follow this type of distribution. Otherwise, outliers can not be detected nor rejected since they might not be inside the first or last percentiles of the data.

This being so, a full-scale analysis was performed in order to determine which statistical distribution was followed by each variable, if there was any. Hence, each variable was plotted using a qq-plot against four different already known distributions (normal, uniform, exponential, and Laplacian), since this graphical method allows to compare two distributions, as seen in *Figure 9*. According to the similarity between the data and the reference line from the known distribution, it can be assessed whether the data follows this distribution or not.

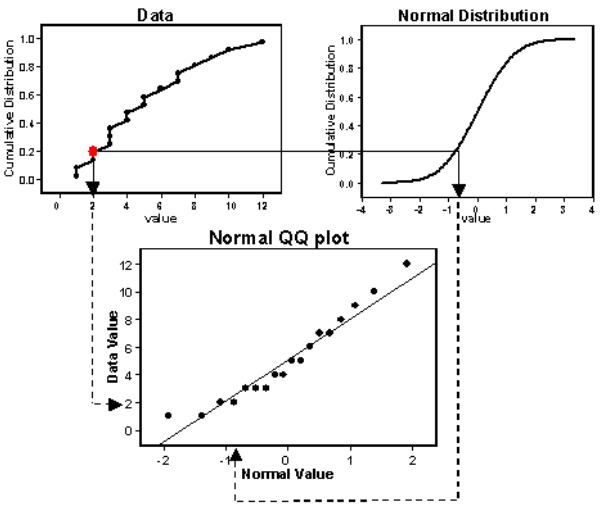

Figure 9. QQ-plot description

In addition, since qq-plots provide only qualitative information, a Shapiro-Wilks test has been also performed on each variable as a way of quantifying their similarity with the normal distribution. This method allows considering whether if the data follows a normal distribution or not from the resulting p-value; therefore, in the case of this being less than 0.05, it can not be considered that the data is distributed in a normal way; otherwise, normality cannot be rejected.

In the case of identifying any outliers in some of the features, those observations containing these outliers would be removed from the data frame since they could include erroneous information, which would directly affect posterior analysis and the performance of the predictive models.

5.4. DATA SPLIT

Once the Python script for all the processes above has been developed, the patients were manually and arbitrary distributed according to a 80/20% split for the training and test datasets, respectively. After this split, the Python script was applied separatedly to each set with the aim of obtaining two data frames that followed all the mentioned conditions, one for the training of the models, and a second one to test them. The number of patients before and after the split, as well as the final number of patients remaining after the patient exclusion can be observed in *Figure 10*.

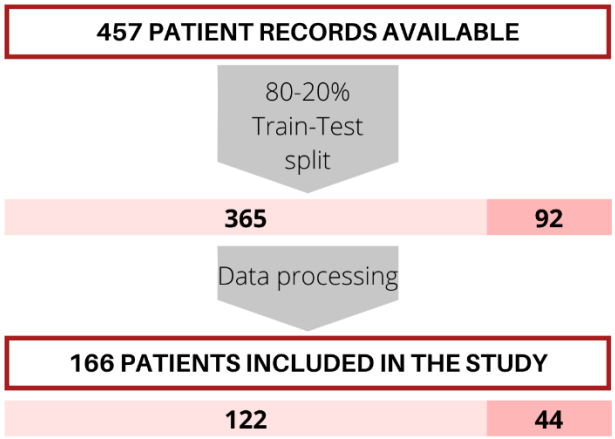

Figure 10. Schematics of the patients before and after the train-test split and the data processing

Since the testing of the models consists on applying the already trained model to new data so it can perform its predictions, the column with the labels had to be separated from the features before the models testing.

With all these processes, the matrixes required for the training and testing of the models were ready to use. Regarding the matrix for the training, it included all the features and labels. As for the testing, two matrixes were obtained, one with the features, which was the one later used to test the models, and another one with the labels, used to evaluate the performance of the model by comparing this labels to the predicted ones.

5.5. MACHINE LEARNING MODEL TRAINING AND TEST

After all the processing of the data, the outliers analysis, and the split of the data, the remaining steps for the study were the building, training and testing of all the previously mentioned models.

Each model has some parameters that can be adjusted in order to improve the performance of the models. However, there is no way to select these parameters other than by testing them and comparing how the models work; hence, for each model different options were tried and assessed by comparing its AUC score.

5.5.1. MODELS BUILDING

5.5.1.1. SVC

For the SVC model, two parameters had to be selected with the aim of achieving the best performance possible. One of these was the *kernel* of the model, which adjusted the decision boundaries, and the gamma (γ) constant, which adjusts the classification. Therefore, 16 different models were built in order to finally select the best-performing one. These models were all the possible combinations for four different *kernels* and four different γ, as can be seen in *Table 7*.

Table 7. Different kernels and γ used for the SVC models

Kernel	Linear	Polynomial	RBF	Sigmoid
Gamma (γ)	Automatic	0.025	0.05	Scale

5.5.1.2. KNN

Regarding the KNN model, the number of neighbors to take into consideration for making the prediction is also an adjustable parameter. Therefore, the model was tested with 10, 50, 100 and 500 number of neighbors.

5.5.1.3. RF AND XGB

As for the RF and XGB models, the same variable related to the number of estimators to use is also an adjustable parameter. Similarly to the previous models, thus, four different values were tried in order to later assess their performances and select the most optimal one. Therefore, these two models were tested with 50, 100, 200 and 500 estimators.

5.5.2. TRAINING, TESTING AND EVALUATION

Once all the models were built, the already prepared data for the training was introduced to each model so they could get trained from all the observations and their respective labels. After this process, the data from the testing group of patients was also introduced to the model so it could perform its predictions on the label of each observation.

The predictive ability of each model was assessed by comparing the predicted labels to the known ones. This way, the accuracy score and the AUC value could be obtained for every model, thus enabling to evaluate the performance of all the models individually, as well as to know which ones offer better predictions. The criteria followed for the selection of the final model was the AUC; thus, the model with the highest value would be the most optimal one to implement on patients. In the case where more than one model matched with the highest AUC value, the final decision would be based on the required time for the training, so the model which performs it faster would be selected.

5.6. MODEL IMPLEMENTATION

Finally, the selected model was individually applied to all the patients from the testing set, thus simulating a real use of the model. In order to evaluate the predictions, and due to the large number of patients, only the best and worst predictions in terms of accuracy were selected.

Given the fact that the models require scaled input data, and that when aiming to introduce observations from an individual patient the *Age* feature can not be scaled since its value is the same for all the observations, a solution had to be found. Hence, the age of the patients was divided by 100 in order to obtain a value ranging from 0 to 1. Despite not being exactly the numbers for which the models had been trained to, errors can be considered negligible and the results reliable.

5.7. PYTHON MODULES

In order to perform all the previous analysis as well as to obtain graphical results, several Python modules have been used, which are briefly described below.

As for the initial data processing, the modules `numpy`, `os`, `pandas`, `statsmodels`, `pyplot` and `seaborn` have been used. The `pandas` and `numpy` modules both provide powerful and high-performance tools and methematical functions to store and manipulate data structures and matrixes. Regarding the `os` module, it allows the script to interact with the operating system; thus, it is a useful package for file manipulation. As for `seaborn`, it is a simple and intuitive data visualization module used to obtain and display the obtained results.

The outliers analysis, in turn, the `statsmodels` module enabled the obtention of qq-plots in order to compare two distributions of data, while the `scipy.stats` package provided the required functions for the Shapiro-Wilks test.

Concerning the development and evaluation of the ML predictive models, the `sklearn` module provided the needed functions for the SVM, KNN and RF models as well as for the obtention of the accuracy score and ROC curves. The XGB model was obtained from the package `xgboost`.

6. RESULTS AND DISCUSSION

All the previously mentioned steps allowed to evaluate BS incidence by age groups, as well as to develop and test different ML predictive models and implement the most optimal one to a set of patients, which was the main goal for this project. Hence, the present section includes all the obtained results, validations and assessments for the entire study, as well as a concise discussion about all of them and the overall implications and limitation of the project.

6.1. BS OCCURRENCE ANALYSIS

As already stated in *Section 5.2.2*, BS occurrence is known to have a relation with age; hence the presence of BS periods was assessed for five different groups of age, using those patients whose age is specified in their recorded data.

As it can be observed in *Figure 11a*, there are only two patients under the age of 20; for this reason, the 50% incidence of BS obtained is not representative of this age group and cannot be considered reliable, a fact that is visually explained by the error bar of this age group in *Figure 11b*. As for the group of ages over 80, although the number of patients is quite low, it can be considered enough to assess the BS ocurrence with reliability, as the error bar of this group of age shows.

Therefore, regarding all groups of age except for the first one, it can be observed that the incidence of BS occurrence is always over 50%, with a constant increase with age, leading up to 90.9% of patients with some BS period in patients over 80.

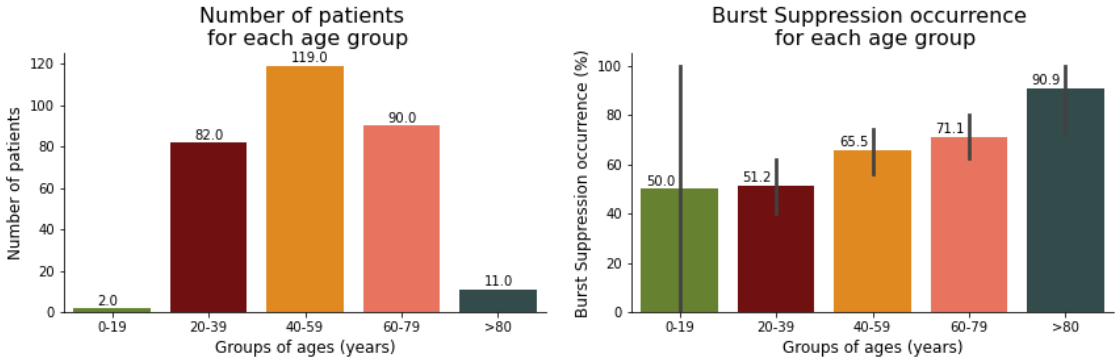

Figure 11. (a) Number of patients by groups of age (b) barplot of BS occurrence by groups of age and its respective error bars

These results highlight the importance of developing a predictive technique for BS anticipation and prevention during propofol-remifentanil induced general anesthesia.

6.2. OUTLIERS ANALYSIS

After processing the data from all the initial patients, the outliers analysis was performed to study the distribution of each feature to identify outliers and reject them afterwards. Therefore, qq-plots comparing the distribution of each parameter with four already known distributions were obtained, as shown in *Figure 12*.

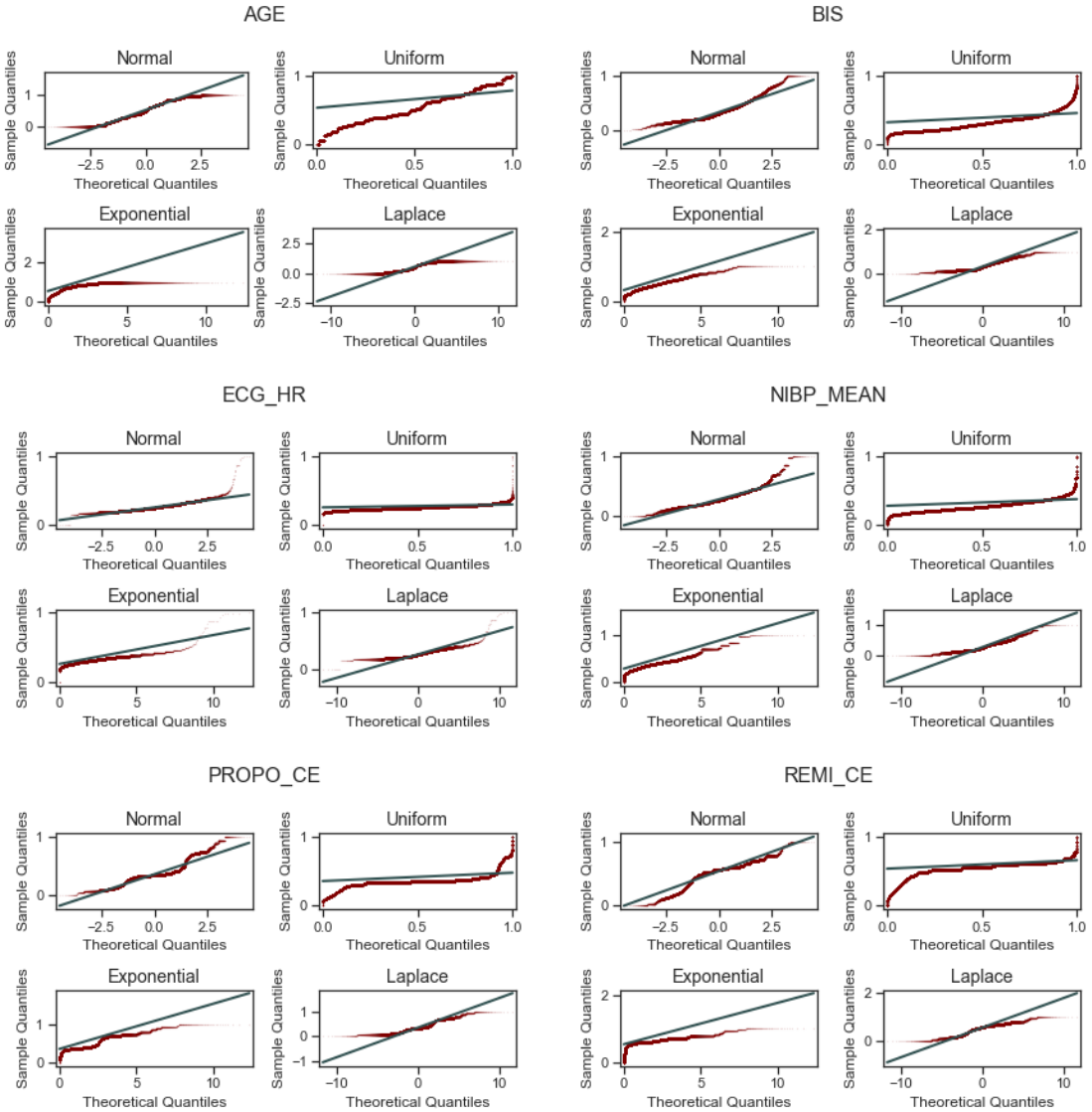

Figure 12. QQ-plots of each feature over four different known distributions

As it can be observed, no variable correctly follows any distribution. However, to quantitatively ensure whether the data follows a normal distribution, a Shapiro-Wilks test was performed on each feature. The resulting p-values from the test were all below the standard threshold of 0.05, therefore implying that any feature can be considered to follow a normal distribution. Having obtained these results, there is no way of identifying outliers for any variable; hence, no more data was discarded except for the already neglected in the data processing process.

6.3. MODELS VALIDATION AND SELECTION

After the data processing and the distribution analysis, the different models were built, trained and tested in order to assess their performances. With the 80/20% split of the initial patients in the training and testing sets respectively, and the subsequent data processing, the distribution groups of ages of the remaining patients can be observed in *Figure 13.* Despite the lack of patients with ages below 20 and the reduced amount of patients over 80 years of age, the aging effect will not change much if younger patients were added since according to different studies the effect of ageing on BS is significant for ages over 65 years old. Therefore, although further improvements

could include a database with a higher representation of patients within these age ranges, the results are not expected to significantly change.

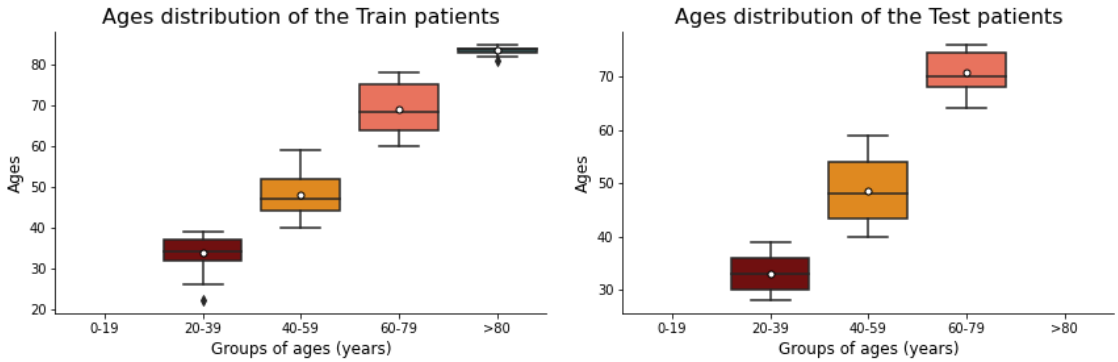

Figure 13. Boxplot of the distribution of the patients by groups of age for the training (left) and testing (right) sets

This evaluation was performed by obtaining the AUC score for each model with different adjustable variables; hence, for the KNN, RF and XGB models, four different AUCs were obtained. However, since for the SVC 16 different models were built, only the four ones with higher AUC values were plotted. The results for all the models are displayed in *Figure 14*.

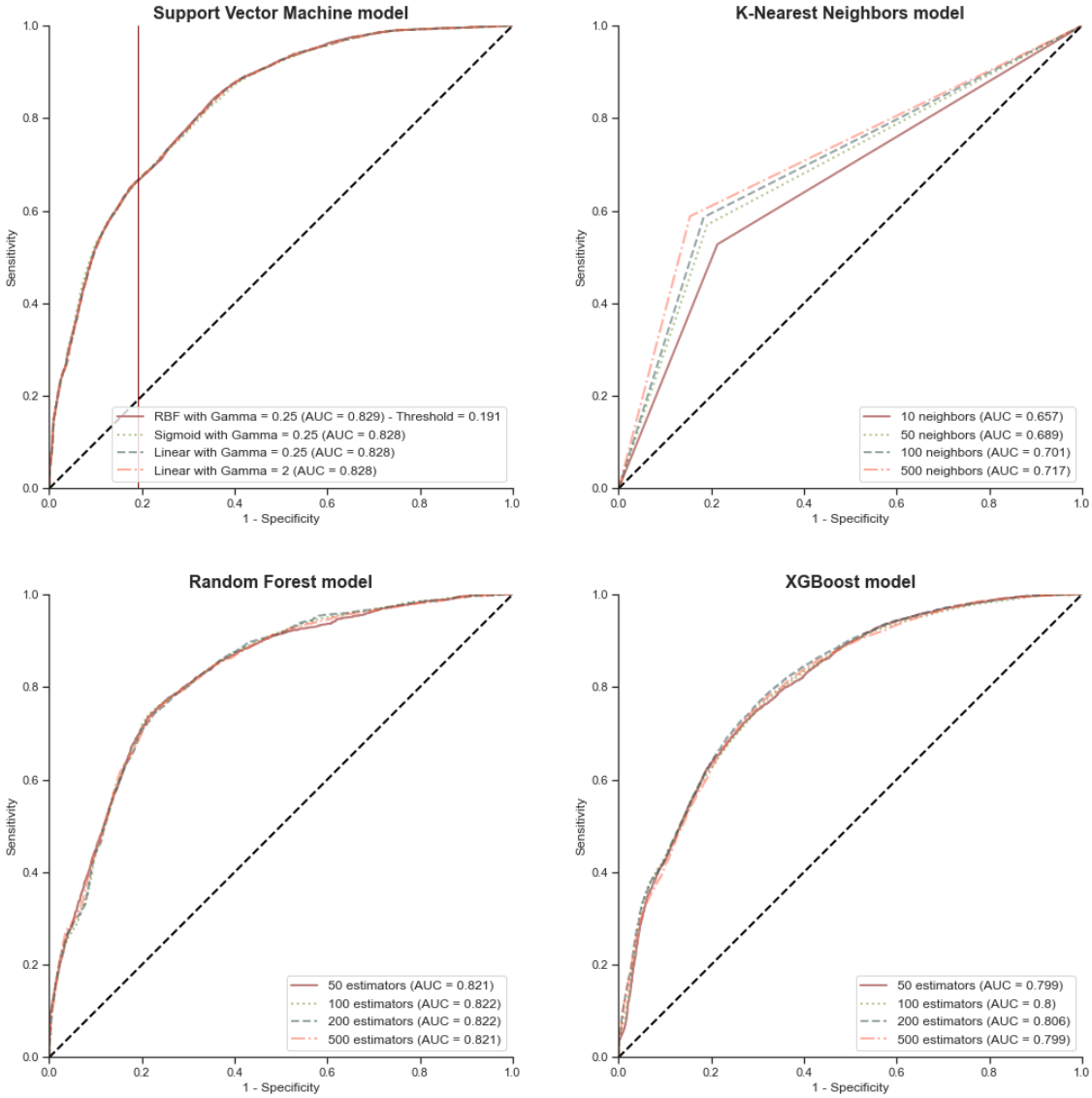

Figure 14. ROC curves for each model with its AUC score, and optimal threshold for the model with highest AUC

First of all, it is important to highlight the relative significance of the obtained threshold. This value is computed by assessing which values of sensitivity and specificity lead to the highest accuracy. However, this threshold cannot be extrapolated to any application of the model, as it will depend on the false positives (FP) and false negatives (FN) that are considered to be allowed. Hypothetically, if a binary classification was desired, thus indicating if BS will occur within two minutes instead of the probability of its occurrence, the obtained threshold could be used to maximize accuracy. Still, the medical implications of both FP and FN should be assessed in order to ensure the reliability of the obtained threshold or the need to adjust it to the application needs.

As for the AUC scores, the first conclusion that can be drawn from these resuls is that predictions from the KNN models are way worse than the other models, having its highest AUC value in the lower limit of the 0.7-0.8 range, which is considered acceptable. Regarding the models from SVC, RF and XGB, all their AUCs are in the higher limit of the 0.7-0.8 range or in the 0.8-0.9 range, which allows to consider predictions as excellent. As shown, SVC with RBF kernel and $\gamma = 0.25$ achieves the highest AUC; thus, it was considered the most optimal model for implementing on patients.

Despite the good results of the models, there may still be room for improvement in order to obtain the most possible accurate predictions. One option would be to perform multiple-class classifications, thus classifying the SR variable on different range labels representing the degree of BS, since this would lead to more variability in the data. In this case, instead of replacing all values greater than 0 with a 1, different classes could represent the level of BS. However, the development of these models requires previous studies in order to assess which thresholds should be used when classifying degrees of BS and to know the medical significance of these classes, as well as a comparative study with the actual models in order to assess whether permormance has been improved or not.

Finally, it is important to take into account that the obtained AUCs should be validated with a 95% confindence interval in order to ensure similar performances when applying the predictive model [67][68]; hence, further improvements of the study must include confidence intervals.

6.4. FEATURE IMPORTANCE

As a way of assessing the weight of each parameter on predicting the occurrence of BS so that the anesthesiologists can understand what factors influence the most when making a prediction, a feature importance analysis has been used, for simplicity, only in the RF and XGB models; hence, barplots indicating the relevance of each feature for both models are displayed in *Figure 15*.

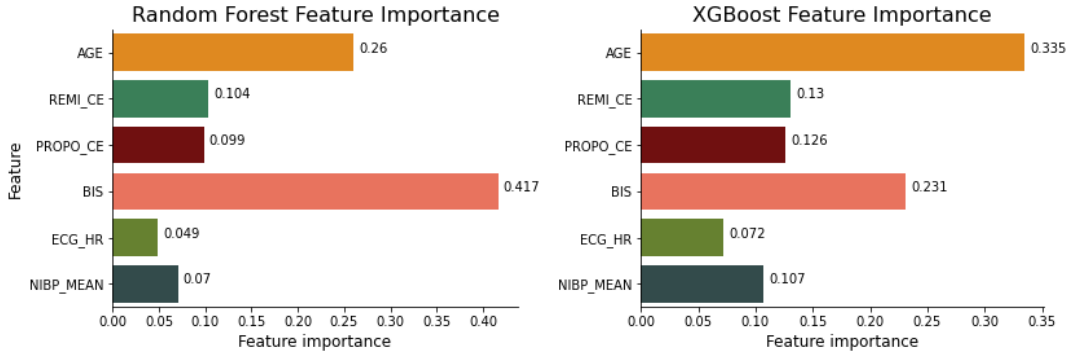

Figure 15. Feature importance for RF and XGB models

It is important to highlight that these feature importance analysis offer a qualitative and orientative measure on the influence of each factor over the prediction of the model; thus, it can only be used to provide assistance in interpretation on the factors influence. However, there is no consensus on the way feature importance is quantified, so the results can not be considered much reliable and should be treated carefully.

Therefore, as can be seen at first sight, the features influencing the most are age and BIS value for RF and XGB models. This high influence of these two features matches with the expected results as age is directly related to BS occurrence and BIS accounts for the anesthetic effect on patient and, thus, the level of brain activity.

Another conclusion that can be drawn from these results is that the propofol effect-site concentration does not have as much influence as would be expected given its hypnotic effect and, hence, its direct relation with BS. In fact, at first it would be expected that the concentration of propofol should have more influence than age for BS prediction. A possible explanation could be the small variability on the values of effect-site concentration of propofol, since in the collected data this parameter always has values inside a relatively reduced range while features such as age and BIS oscillate inside a wider range. According to this, it is possible that when keeping the propofol concentration in a range very close to that of the training data, age becomes a more relevant factor. Another possible justification could be that similar propofol concentrations lead to very different responses. However, this results can only be explained by assumptions due to the small variability for the propofol concentration of the available data and the limitations of the feature importance analysis itself.

In light of the above, it can be concluded that age and BIS value can be considered features with a strong influence on the predictive ability of the models and it can be useful in terms of providing orientative assistance to the anesthesiologist, although this cannot be blindly believed due to the low reliability of the metric. As a way of expanding the feature importance analysis, a correlation matrix was obtained as shown in *Figure 16*, in which colors close to 1 imply a strong positive correlation, colors near -1 represent strong negative correlation, and colors around zero suggest a weak or null correlation.

According to the correlation matrix, age and BIS can also be assumed to be the most important factors in determining BS occurrence, which matches the previous results. However, due to the fact that any statistical metric by itself is capable of providing reliable results, as each is computed and considers features differently, a full-scale analysis considering a large set of metrics and combining multiple feature

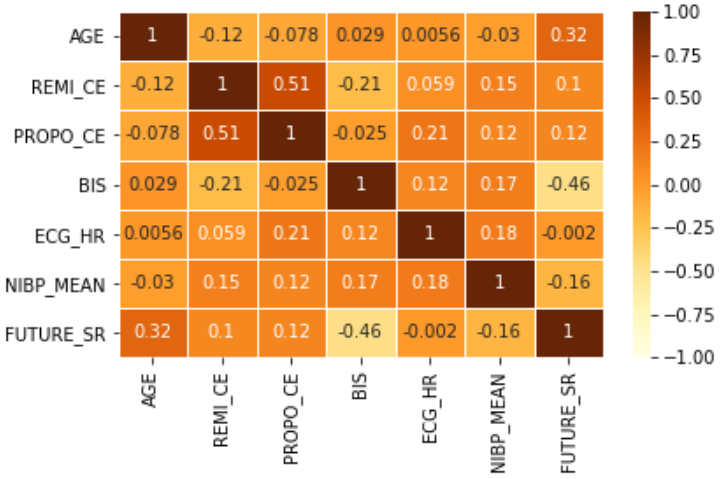

Figure 16. Confusion matrix for the features and the label

importance quantifiers should be performed. This way, the lack of agreement when drawing conclusions from individual feature importance quantifiers could be reduced, since this method would improve the results by decreasing the variance of the influence of each feature [69].

6.5. MODEL IMPLEMENTATION ON PATIENTS

Despite the limitations when evaluating the performance of the models, the results obtained could be considered reliable enough and, therefore, the selected model was tested with individual patient data. The AUC score is a way of assessing the overall performance of the model, but when implementing it individually the accuracy can oscillate within a wide range; therefore, as a way of assessing the entire predictive ability of the selected model, only the best and the worst performances in terms of accuracy are displayed in *Figure 17*. In the figure, the top plots show the probability of BS occurrence within two minutes over time, and the horizontal line drawn in the top plots indicates the *optimal* threshold obtained with the ROC-AUC curves, which it is used to binarize the BS occurrence prediction, as is shown in the second plots. As for the bottom plots, they correspond to the known occurrence of BS within two minuts at each second of the intervention; hence, there is a two-minute shift of this plot in order to easily compare it with the previous ones, which indicate the future occurrence of BS.

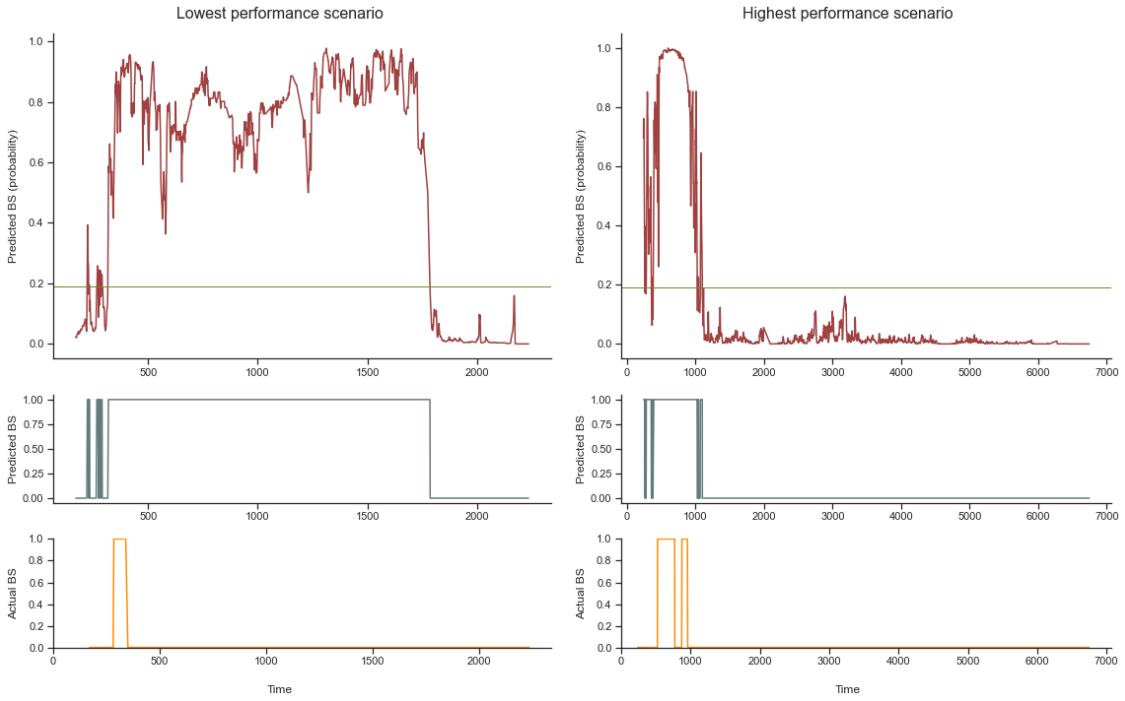

Figure 17. Highest and lowest accuracy performances of the BS occurrence predictive model

As it can be seen, the best performance is actually very adjusted to the real values on BS occurrence and, taking only into account this result, it could be considered reliable enough to implement the model as a tool to assist the anesthesiologist in adjusting the doses of anesthetics in order to prevent episodes of BS. However, the results obtained from the performance with less reliability show a very low predictive ability of BS occurrence, which could lead to decisions that are inconsistent with the actual condition of the patient.

Moreover, in order to evaluate the distribution of each predicted probability over the known BS occurrence, the plots in *Figure 18* were obtained. In it, it can be assessed the probability distribution of those observations with a known positive label, as well as to those with known negative labels.

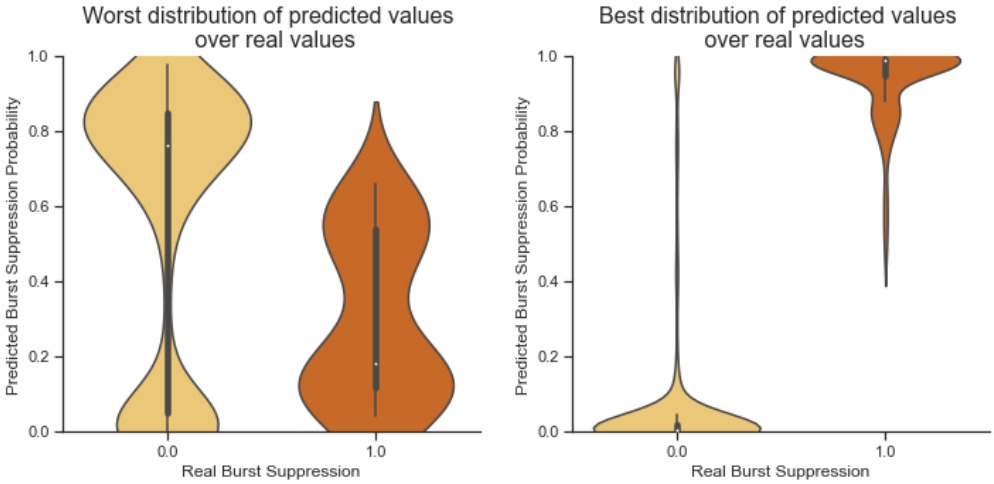

Figure 18. Distribution of the predicted probabilities against the label they should predict for the highest and the lowest accuracy performances

As can be observed, in the implementation showing the worst performance, a large proportion of observations that should lead to a negative value actually shows scores above 0.6 and with a mean value almost at 0.8, and observations with a known positive value are mainly distributed below a 0.5 probability, with a mean value close to 0.2. On the other hand, in the best-performance scenario, it can be clearly seen that observations that should be labeled as negative have probabilities mainly below 0.2, with a mean value close to 0, and the observations known to correspond to a positive label show probabilities over 0.7, with a mean value of almost 1.

Due to this huge difference in accuracy between the best and the worst performances, the distribution of accuracies was obtained as a way of assessing the proportion of patients with sufficiently good and reliable predictive performances, as can be seen in *Figure 18*.

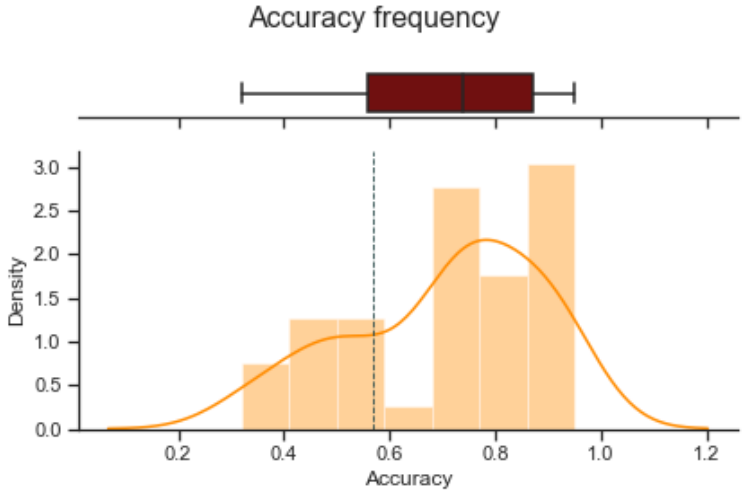

Figure 19. Accuracy frequency barplot and accuracy distribution boxplot

As it can be noted, the first percentile of accuracies are below 0.6, thus implying that the remaining 75% of the obtained accuracies have values greater than 0.6, with a mean score ranging between 0.7 and 0.8. In other words, the 75% of the patients in which the model was implemented the

accuracy scored values over 0.6; hence, it can be considered that the obtained model has an acceptable predictive ability and could be useful in advising the anesthesiologist when adjusting the doses of anesthetics.

However, since the model is not reliable enough and has some limitations, the BS occurrence probability obtained cannot be taken as absolute truth and the index must be understood as an orientative indicator.

6.6. IMPLICATIONS

Despite the high quality of the collected data in terms of the number of parameters continuously monitored, and the large amount of patients recorded, the data processing considerably decreased the number of used patients, thus reducing variability among some parameters, such as age. This variability is also limited by the type of patients undergoing general anesthesia procedures in the CMA in Hospital Clínic, since almost all the patients are white women with ages above 40, thus underrepresenting young population, male gender and ethnical diversity. Nonetheless, since gender and ethnic group are not considered much significant factors for brain function, and age becomes an important factor when it goes beyond 65, the used data accurately reflects the general behavior of patients.

The high ROC-AUC scores obtained in the developed models evidence the predictability of BS occurrence, hinting that there is room for anticipating this phenomenon during propofol-remifentanil induced general anesthesia. However, due to its poor performance in a 25% of the implementations on patients, the final selected model is far from being reliable, and further studies regarding the lack of confidence intervals and a more exhaustive feature selection should be carried out in order to improve the present model and obtain more reliable predictions.

Therefore, the proposed solution needs to be considered as an aiding tool for the anesthesiologist in terms of providing orientative knowledge on the current status of the patient and a probability predictive index of BS occurrence, which could be taken into account when adjusting the anesthetics doses. Besides, although it has been trained only with white women, the model can be considered a robust initial approach which could also be cautiously implemented on patients with these different conditions due to the previously mentioned low significance of these conditions.

6.7. LIMITATIONS

Bearing in mind that the proposed solution is mainly based on data analysis and processing, it can be expected that an important part of the limitations of the final predictive model will be due to limitations on the initial data as well as on its processing.

It is important to take into consideration that the initial feature selection was performed by previous knowledge on the influence of each parameter in BS occurrence, but it lacked statistical background. Therefore, in order to ensure the influence between variables, as well as to assess their relationship with BS occurrence, different feature importance analysis should be performed at the beginning of the study. In addition, after excluding the features with less influence for BS occurrence predictions, several ML predictive models could be built, trained and tested with

different combinations of the remaining features in order to find the best-performing one, and also as a way of assessing once again the importance of each feature when predicting BS.

Regarding the data processing, the lack of validation techniques such as cross validation and confidence intervals in each performed process might lead to less reliable results. Accordingly, other statistical metrics should be adopted in future studies in order to improve the accuracy and credibility of the obtained predictions, as well as a full-scale feature extraction.

Lastly, it should be noted that the implementation of these models is limited to those interventions under propofol-remifentanil induced general anesthesia which include all the mentioned monitoring systems. Otherwise, the model would not have all the continuously required parameters and indexes to obtain the desired BS occurrence predictions.

In order to increase validity of the model, information regarding the disease state and medication intake of the patients should be included in the database, as well as procedures with general anesthesia induced by inhalation agents, which can easily be incorporated as new covariate factors in the model.

7. EXECUTION SCHEDULE

The required tasks and phases for acomplishing the initial objectives of the study, as well as the time devoted to each task and the overall schedule are covered in this section as a way of organizing, prioritizing and optimizing all the steps during the execution.

7.1. TASKS DEFINITION

First of all, the entire project was decomposed in small tasks to fulfill in order to respect the duration of the project and its established deadline. Hence, this section includes a brief definition of all the phases of the study, along with the estimated time for each task, which are shown in *Table 8*.

Table 8. Overview of the tasks of the project, their description and their estimated duration time (in days)

PROJECT PREPARATION	**5**
Visit to CMA in Hospital Clínic	1
Visit to CMA in order to see the workspace environment, the monitoring devices and the anesthesia control tower	
Bibliographic background research	3
Introduction on the subject, reading of previous studies from the research group, and personal interests search	
Topic selection	1
Analysis of interests of the reasearch group, different topic proposals and final selection according with personal interests	
DATA ACQUISITION	**17**
Existing database examination	1
Inspection of the already collected data, the recorded parameters and its structure	
New data acquisition	16
Stay at the gynecological operating room in the CMA in Hospital Clínic for data collection from patients undergoing propofol-remifentanil induced general anesthesia	
DATA PROCESSING	**63**
Data frame construction	49
Data manipulation in order to obtain a final matrix with all the available data after processing it to fulfilling the requirements of ML predictive models. This phase includes the label obtention, the signal quality assessment, a gender, feature and observations selections, a compensation of the proportion of different labels and a feature scaling	
Burst Suppression incidence assessment	4
Evaluation of the incidence of BS occurrence by groups of age on the initial patients available for the study	
Outliers analysis	9
Analysis of the statistical distribution of each feature as a way of identifying possible outliers and rejecting them, if found	
Data split into training and test datasets	1
Division of the available patients into a training and a test datasets following a 80/20% split, respectively	
ML PREDICTIVE MODELS BULDING	**23**
Model training	5
Training of each model with the training dataset of patients after preprocessing its data	

Model testing	2
Testing of each model by applying it to the test dataset	
Model evaluation and selection	4
Obtention of the ROC-AUC curves for each model and assessment of the results to select the most optimal model	
Model implementation	12
Implementation of the model with all the patients from the test dataset, evaluation of its best and worse predictions of the model in terms of accuracy, and assessment of the overall performance	
PROJECT WRITING	**44**
Written report	37
Writing of the final report of the project	
Presentation	7
Realization of a brief final presentation of the project	

7.2. TIMING AND PHASES – GANTT DIAGRAM

After defining all the phases and the required tasks to accomplish all the initial goals, a schedule must be generated in order to obtain and assess the deadline for each task in order to carry out the whole project within the overall deadline. With this objective, a GANTT diagram is provided in *Figure 20* within the present section, along with the GANTT legend from which the chart is obtained, which can be observed in *Table 9*. This diagram allows to visualize in a simple way the tasks to be carrying out at each moment, as well as to know which tasks must be performed simultaneously.

Table 9. GANTT legend

DESCRIPTION	START DATE	DURATION	END DATE
Visit to CMA in Hospital Clínic	13 Sep.	1	13 Oct.
Bibliographic background research	14 Sep.	3	17 Sep.
Topic selection	28 Sep.	1	28 Sep.
Existing database examination	19 Sep.	1	19 Sep.
New data acquisition	13 Sep.	45*	27 Oct.
Data frame construction	4 Oct.	49	21 Nov.
Burst Suppression incidence assessment	18 Nov.	4	21 Nov.
Outliers analysis	22 Nov.	9	30 Nov.
Data split into training and test datasets	1 Dec.	1	1 Dec.
Model training	2 Dec.	5	6 Dec.
Model testing	7 Dec.	2	8 Dec.
Model evaluation and selection	9 Dec.	4	12 Dec.
Model implementation	13 Dec.	12	24 Dec.
Written report	16 Dec.	37	21 Jan.
Presentation	22 Jan.	7	28 Jan.

** Data was collected for 16 days spread over 7 weeks, with an overall duration of 45 days*

TASK	START	END	SEPTEMBER	OCTOBER	NOVEMBER	DECEMBER	JANUARY
PROJECT PREPARATION							
Visit to CMA in Hospital Clínic	9/13/21	9/13/21					
Bibliographic background research	9/14/21	9/17/21					
Topic selection	9/28/21	9/28/21					
DATA ACQUISITION							
Existing database examination	9/19/21	9/19/21					
New data acquisition	9/13/21	10/27/21					
DATA PROCESSING							
Data frame construction	10/4/21	11/21/21					
Burst Suppression incidence assessement	11/18/21	11/21/21					
Outliers analysis	11/22/21	11/30/21					
Data split into training and test datasets	12/1/21	12/1/21					
ML PREDICTIVE MODELS BUILDING							
Model training	12/2/21	12/6/21					
Model testing	12/7/21	12/8/21					
Model evaluation and selection	12/9/21	12/12/21					
Model implementation	12/13/21	12/24/21					
PROJECT WRITING							
Written report	12/16/21	1/21/22					
Presentation	1/22/22	1/28/22					

Figure 20. GANTT diagram for the project execution

8. TECHNICAL VIABILITY – SWOT ANALYSIS

The project development directly depends on several aspects both external and internal which might affect the previous work, the course of the project, and its final success once completed. Hence, an exhaustive and full-scale analysis determines the course of the project and allows to anticipate and constantly review those aspects that can be improved. Bearing this in mind, a SWOT analysis was performed due to its utility in identifying the positive and negative factors involving the project, both from an external (opportunities and threats) and internal (strengths and weaknesses) point of view, as displayed in *Table*.

Table 10. SWOT analysis on the project

STRENGTHS
Already existing database including up to 1500 patients
Help and guidance from professionals and master students
Previous Python experience and knowledge
Resulting model based on real data
WEAKNESSES
Availability of only 457 patients from the database
Limited prior knowledge on anesthesia monitoring
No previous experience with ML predictive models
OPPORTUNITIES
Interest in individualized anesthesia is on the rise
Hospital Clínic and the SPEC-M research group support the study
ML models are still not implemented in anesthesiology
No similar studies have been carried outside the SPEC-M resarch group
Several studies report possible adverse effects with the occurrence of BS
THREATS
Lack of consensus regarding the adverse outcomes of BS occurrence
No similar studies on the topic to compare and validate results
Monitoring devices required must be installed and incorporated in the operating room
Database expansion depends on the number of surgeries carried out
Legislation and regulation

9. ECONOMIC VIABILITY

With the aim of having a record and knowledge on the total cost of the development of a BS occurrence predictive model, an estimated study of costs and budgets has been carried out by taking into account all the expenses, both direct and indirect, involving the overall project. These costs, classified into material and human resources, are detailed in the present section and displayed in *Table 11*.

9.1. MATERIAL RESOURCES

9.1.1. HARDWARE REQUIREMENTS

The data collected during the stay at the CMA in Hospital Clínic was not subsequently used for the data processing and predictive model, since the new data acquisition was carried out with the aim of expanding the existing database, but the patients used for the study were already collected. In addition, some of the used equipment for the monitoring of the patients consisted of disposable material, thus being difficult to estimate since the number of patients from the database is constantly increasing. Bearing these two facts in mind, the study of costs for the hardware requirements of the project was estimated for the number of patients included in the study; thus, taking into consideration the acquisition of data from 457 patients.

Moreover, it has be taken into account that not all the used devices for general anesthesia monitoring are exclusive for this project, since some of these are already required for a proper control of the anesthetic state of the patients. Therefore, only the BIS VISTA® Bilateral Monitoring System has been included, considering that this monitor is not always used in operating rooms but it is explicitly required for the present study, as well as the computer required for recording and storing the collected data.

Finally, given that the data processing included computationally expensive techniques, a powerful computer is required which, since the project is not carried out in any office or fixed location, has been considered more preferable for it to be a laptop.

9.1.2. SOFTWARE REQUIREMENTS

The data analysis and processing, as well as the ML models building, training, testing and final implementation on patients was performed using open source software such as Python. In addition, the recorded data is stored in .xlsx files, so Microsoft Excel® was used to examine the patients data, and the final report and presentation used Microsoft® 365 services. Lastly, the storing of the recorded data uses the Rugloop® free software, able to access the monitoring devices and save them in a file.

9.2. HUMAN RESOURCES

As a way of assessing the human resources required for successfully carrying out the project, and according to the previously detailed tasks and the overlapping of some phases, it has been considered that the project had an overall duration of 138 days, from September 13, 2021 to

January 28, 2022. During these days, all the mentioned tasks were performed by an undergraduate student, and support and supervision tasks were performed both by a Master student and a professional anesthesiologist. Hence, taking into consideration the amount of hours devoted to the project, the expertise, and according to the current standards and cost of living, the the salary of each person per hour worked has been decided to be of 10€ for the undergraduate, 15€ for the Master student, and 30€ for the professional supervisor.

9.3. COSTS AND BUDGET

All the previously detailed costs are concisely detailed in *Table 11*, estimating the cost of each item for the project as well as its overall budget.

Table 11. Costs and budget for the entire project

	UNIT/HOUR PRICE (€)	UNITS/HOURS	TOTAL (€)
MATERIAL RESOURCES - HARDWARE			
BIS VISTA® Bilateral Monitoring System	5740	1	5740
BIS VISTA® electrodes	13.25	457	6055.25
HP ProPne 400 G6	895	1	895
SUBTOTAL			***12690.25***
MATERIAL RESOURCES – SOFTWARE			
Python software	-	1	-
Microsoft Office® 365	69	1	69
Rugloop® software	-	1	-
SUBTOTAL			***69***
HUMAN RESOURCES			
Undergraduate student	10	400	4000
Master student	15	10	150
Professional supervisor	30	20	600
SUBTOTAL			***4750***
TOTAL			***17509.25***

10. REGULATORY AND LEGAL ASPECTS

The developement of the present project, both regarding the equipment used and the data collection and manipulation, has to follow and agree with the current regulations and legislations. Therefore, some regulatory issues need to be considered.

Regarding the data acquisition, the patients whose data is recorded have previously consent the data collection and usage in studies. However, given the fact that when a patient entries to the pre-operative room its medical history is automatically available for all the professionals assisting the surgery, the *Ley Orgánica 3/2018*, also known as *Ley Orgánica de protección de datos personales y garantía de los derechos digitales* has to be carefully respected in order to ensure data confidentiality during the intervention and of all the information regarding personal aspects of the patient [70].

As for the developed predictive model based on ML algorithms, it can be classified in the category of Software as Medical Device (SaMD), as well as some other ML models applicable to healthcare. Hence, it must follow the directives set by the US Food and Drug Administration (FDA), responsible for the regulation of drugs and medical devices [71], or by the corresponding and appropriate government agencies in each country.

According to the European Union, however, there are no regulations regarding the use of ML in healthcare. If the developed BS occurrence predictive model was improved in terms of accuracy, since it displays an index which could influence the anesthesiologist on adjusting and manipulating the anesthesic administration and, thus, immediately affect physiological parameters, it could be classified as IIb device.

Finally, in order to introduce the predictive model to the market, a detailed manual specifying the operation of the device, as well as the type of data it requires and recommendations on how to interpret the results must be submitted to FDA or similar agencies, depending on the comercialization country, for evaluation and, if permitted, the BS predictive model could be distributed.

11. CONCLUSIONS AND FUTURE LINES

The reported possible adverse outcomes regarding BS occurrence, as well as its high incidence under propofol-remifentanil induced general anesthesia evidence the need of developing predictive methods in order to anticipate and avoid this phenomenon. This can be accomplished using ML techniques, which currently are gaining popularity on many fields including healthcare; however, they are still not much implemented in operating rooms and patient monitoring. Therefore, this project aims to prove the potential benefits of introducing ML into surgical environments while proposing a solution to the lack of BS occurrence predictive power in current monitoring devices.

According to the obtained results, it has been proved that BS occurrence is susceptible to prediction through ML predictive models trained with a high-quality large database. The model achieving the highest ROC-AUC score has been shown to be the SVC with RBF kernel and $\gamma = 0.25$ with a value of 0.829; thus, it has been the selected model for BS occurrence predicting. Nonetheless, after implementing the model to several patients, it was seen that a 25% of the predictions had a predictive accuracy below 0.6, which can be considered to be a significantly large proportion of unaccurate predictions. Therefore, the current model is a robust initial approach that has to be understood as an orientative guidance tool for the anesthesiologist when assessing the anesthetic state of the patient and adjusting drug administration, but it should not be taken as absolute truth due to its low reliability.

Regarding the analysis on the influence of each used feature in the BS occurrence predictions, both feature importance metrics included in RF and XGB predictive models and correlation matrix statistics from the SVC results show a high influence of age and BIS values on their predictions, followed by the propofol effect-site concentration. Since at first it might be expected a higher influence of the hypnotic agent, further studies should be performed with more variability in the data and using different metrics in order to ensure the reliability of these results.

In light of the above, the predictive ability of the obtained model achieves an acceptable performance but shows that there is still room for improvement. Hence, model refinement should include a larger dataset involving patients under general anesthesia induced by inhalation agents and information regarding disease state and medication of the patients, which can be easily incorporated as covariate factors. As for the data processing, different validation techniques such as confidence intervals should be applied to each process, and more statistical metrics could lead to a more reliable feature importance assessment. In addition, an exhaustive feature selection testing the models with different combinations of the recorded parameters could be carried out, along with a full-scale study on the performance of more ML models in order to test whether an enhancement in the predictive power can be achieved.

To conclude, the proposed solution must be interpreted as a helpful tool for future BS occurrence assessment, but its current low reliability should always be kept in mind. If further studies achieved an improved performance of the model, the ultimate goal would be to integrate it within the monitoring devices already used in surgical environments and patient monitoring. Eventually, if outstanding results were accomplished, the model could be integrated with the TCI system as a way of automatically adjust drug administration according to the BS occurrence prediction.

REFERENCES

[1] PAION (2021). Paion - General Anesthesia (Europe). https://www.paion.com/allgemeinanaesthesie/

[2] Weiser, T. G., Regenbogen, S. E., Thompson, K. D., Haynes, A. B., Lipsitz, S. R., Berry, W. R., & Gawande, A. A. (2008). An estimation of the global volume of surgery: a modelling strategy based on available data. *Lancet (London, England)*, *372*(9633), 139–144. https://doi.org/10.1016/S0140-6736(08)60878-8

[3] Brown, E. N., Lydic, R., & Schiff, N. D. (2010). General anesthesia, sleep, and coma. *The New England journal of medicine*, *363*(27), 2638–2650. https://doi.org/10.1056/NEJMra0808281

[4] Milam S. B. (1984). General anesthetics: a comparative review of pharmacodynamics. *Anesthesia progress*, *31*(3), 116–123.

[5] Shanker, A., Abel, J. H., Schamberg, G., & Brown, E. N. (2021). Etiology of Burst Suppression EEG Patterns. *Frontiers in psychology*, *12*, 673529. https://doi.org/10.3389/fpsyg.2021.673529

[6] Soehle, M., Dittmann, A., Ellerkmann, R. K., Baumgarten, G., Putensen, C., & Guenther, U. (2015). Intraoperative burst suppression is associated with postoperative delirium following cardiac surgery: a prospective, observational study. *BMC anesthesiology*, *15*, 61. https://doi.org/10.1186/s12871-015-0051-7

[7] Andresen, J. M., Girard, T. D., Pandharipande, P. P., Davidson, M. A., Ely, E. W., & Watson, P. L. (2014). Burst suppression on processed electroencephalography as a predictor of postcoma delirium in mechanically ventilated ICU patients. *Critical care medicine*, *42*(10), 2244–2251. https://doi.org/10.1097/CCM.0000000000000522

[8] Forman, S. A., & Chin, V. A. (2008). General anesthetics and molecular mechanisms of unconsciousness. *International anesthesiology clinics*, 46(3), 43–53. https://doi.org/10.1097/AIA.0b013e3181755da5

[9] Brown, E. N., Pavone, K. J., & Naranjo, M. (2018). Multimodal General Anesthesia: Theory and Practice. *Anesthesia and analgesia*, 127(5), 1246–1258. https://doi.org/10.1213/ANE.0000000000003668

[10] Campagna, J. A., Miller, K. W., & Forman, S. A. (2003). Mechanisms of actions of inhaled anesthetics. The New England journal of medicine, 348(21), 2110–2124. https://doi.org/10.1056/NEJMra021261

[11] Rudolph, U., & Antkowiak, B. (2004). Molecular and neuronal substrates for general anaesthetics. *Nature reviews. Neuroscience*, *5*(9), 709–720. https://doi.org/10.1038/nrn1496

[12] Facco E. (2016). Hypnosis and anesthesia: back to the future. *Minerva anestesiologica*, *82*(12), 1343–1356.

[13] Jensen, E. W., Valencia, J. F., López, A., Anglada, T., Agustí, M., Ramos, Y., Serra, R., Jospin, M. Pineda, P., & Gambus, P. (2014). Monitoring hypnotic effect and nociception with two EEG-derived indices, qCON and qNOX, during general anaesthesia. *Acta anaesthesiologica Scandinavica*, 58(8), 933–941. https://doi-org.sire.ub.edu/10.1111/aas.12359

[14] Rani, D. D., & Harsoor, S. (2012). Depth of general anaesthesia monitors. *Indian journal of anaesthesia*, 56(5), 437–441. https://doi.org/10.4103/0019-5049.103956

[15] Fields H. (2004). State-dependent opioid control of pain. *Nature reviews*. Neuroscience, 5(7), 565–575. https://doi.org/10.1038/nrn1431

[16] Alwardt, C. M., Redford, D., & Larson, D. F. (2005). General anesthesia in cardiac surgery: a review of drugs and practices. *The journal of extra-corporeal technology*, 37(2), 227–235.

[17] Allen R. J. (2018). Classic and recent advances in understanding amnesia. F1000Research, 7, 331. https://doi.org/10.12688/f1000research.13737.1

[18] Satwik, A., & Naveed, N. (2015). Anesthesia - a review. *Journal of Pharmaceutical Sciences and Research*, 7(4), 182.

[19] Raghavendra T. (2002). Neuromuscular blocking drugs: discovery and development. *Journal of the Royal Society of Medicine*, 95(7), 363–367. https://doi.org/10.1258/jrsm.95.7.363

[20] Hendrickx, J. F., Eger, E. I., 2nd, Sonner, J. M., & Shafer, S. L. (2008). Is synergy the rule? A review of anesthetic interactions producing hypnosis and immobility. *Anesthesia and analgesia*, 107(2), 494–506. https://doi.org/10.1213/ane.0b013e31817b859e

[21] Introduction of anesthesia (2016). *Anesthesia key*. [Available online] https://aneskey.com/induction-of-anesthesia/

[22] Ke, J. J., Zhan, J., Feng, X. B., Wu, Y., Rao, Y., & Wang, Y. L. (2008). A Comparison of the Effect of Total Intravenous Anaesthesia with Propofol and Remifentanil and Inhalational Anaesthesia with Isoflurane on the Release of Pro- and Anti-Inflammatory Cytokines in Patients Undergoing Open Cholecystectomy. *Anaesthesia and Intensive Care*, 36(1), 74–78. https://doi.org/10.1177/0310057X0803600113

[23] Lim, A., Braat, S., Hiller, J., & Riedel, B. (2018). Inhalational versus Propofol-Based Total Intravenous Anaesthesia: Practice Patterns and Perspectives among Australasian Anaesthetists. *Anaesthesia and Intensive Care*, 46(5), 480–487. https://doi.org/10.1177/0310057X1804600509

[24] Weninger, B., Czerner, S., Steude, U., & Weninger, E. (2004). Vergleich zwischen TCI-TIVA, manueller TIVA und balanzierter Anästhesie während stereotaktischer Gewebsentnahme in der Neurochirurgie [Comparison between TCI-TIVA, manual TIVA and balanced anaesthesia for stereotactic biopsy of the brain]. *Anasthesiologie, Intensivmedizin, Notfallmedizin, Schmerztherapie*: AINS, 39(4), 212–219. https://doi.org/10.1055/s-2004-814363

[25] Masui, K., Upton, R. N., Doufas, A. G., Coetzee, J. F., Kazama, T., Mortier, E. P., & Struys, M. M. (2010). The performance of compartmental and physiologically based recirculatory

pharmacokinetic models for propofol: a comparison using bolus, continuous, and target-controlled infusion data. *Anesthesia and analgesia*, 111(2), 368–379. https://doi.org/10.1213/ANE.0b013e3181bdcf5b

[26] Laso, L.F., López-Picado, A., de La Fuente, E.O., Murua, A., Sánchez-Castro, C., Ruilope, L.P., & Valero-Martínez, C. (2016). Manual vs. target-controlled infusion induction with propofol: An observational study. *Colombian Journal of Anesthesiology*, 44, 272-277.

[27] Mu, J., Jiang, T., Xu, X. B., Yuen, V. M., & Irwin, M. G. (2018). Comparison of target-controlled infusion and manual infusion for propofol anaesthesia in children. *British journal of anaesthesia*, 120(5), 1049–1055. https://doi.org/10.1016/j.bja.2017.11.102

[28] Langley, M. S., & Heel, R. C. (1988). Propofol. A review of its pharmacodynamic and pharmacokinetic properties and use as an intravenous anaesthetic. *Drugs*, 35(4), 334–372. https://doi.org/10.2165/00003495-198835040-00002

[29] McKeage, K., & Perry, C. M. (2003). Propofol: a review of its use in intensive care sedation of adults. *CNS drugs*, 17(4), 235–272. https://doi.org/10.2165/00023210-200317040-00003

[30] Scott, L. J., & Perry, C. M. (2005). Remifentanil: a review of its use during the induction and maintenance of general anaesthesia. *Drugs*, 65(13), 1793–1823. https://doi.org/10.2165/00003495-200565130-00007

[31] Komatsu, R., Turan, A. M., Orhan-Sungur, M., McGuire, J., Radke, O. C., & Apfel, C. C. (2007). Remifentanil for general anaesthesia: a systematic review. *Anaesthesia*, 62(12), 1266–1280. https://doi.org/10.1111/j.1365-2044.2007.05221.x

[32] Glass, P., Gan, T. J., & Howell, S. (1999). A review of the pharmacokinetics and pharmacodynamics of remifentanil. *Anesthesia and analgesia*, 89(4 Suppl), 7. https://doi.org/10.1097/00000539-199910001-00003

[33] Tran, D. T., Newton, E. K., Mount, V. A., Lee, J. S., Wells, G. A., & Perry, J. J. (2015). Rocuronium versus succinylcholine for rapid sequence induction intubation. *The Cochrane database of systematic reviews*, 2015(10), CD002788. https://doi.org/10.1002/14651858.CD002788.pub3

[34] Mencke, T., Jacobs, R. M., Machmueller, S., Sauer, M., Heidecke, C., Kallert, A., Pau, H. W., Noeldge-Schomburg, G., & Ovari, A. (2014). Intubating conditions and side effects of propofol, remifentanil and sevoflurane compared with propofol, remifentanil and rocuronium: a randomised, prospective, clinical trial. *BMC anesthesiology*, 14, 39. https://doi.org/10.1186/1471-2253-14-39

[35] Yu, S. H., & Beirne, O. R. (2010). Laryngeal mask airways have a lower risk of airway complications compared with endotracheal intubation: a systematic review. *Journal of oral and maxillofacial surgery: official journal of the American Association of Oral and Maxillofacial Surgeons*, 68(10), 2359–2376. https://doi.org/10.1016/j.joms.2010.04.017

[36] Jeanne, M., Logier, R., De Jonckheere, J., & Tavernier, B. (2009). Heart rate variability during total intravenous anesthesia: effects of nociception and analgesia. *Autonomic neuroscience : basic & clinical*, 147(1-2), 91–96. https://doi.org/10.1016/j.autneu.2009.01.005

[37] Bartels, K., Esper, S. A., & Thiele, R. H. (2016). Blood Pressure Monitoring for the Anesthesiologist: A Practical Review. *Anesthesia and analgesia*, 122(6), 1866–1879. https://doi.org/10.1213/ANE.0000000000001340

[38] Jubran A. (2015). Pulse oximetry. Critical care (London, England), 19(1), 272. https://doi.org/10.1186/s13054-015-0984-8

[39] Long, B., Koyfman, A., & Vivirito, M. A. (2017). Capnography in the Emergency Department: A Review of Uses, Waveforms, and Limitations. *The Journal of emergency medicine*, 53(6), 829–842. https://doi.org/10.1016/j.jemermed.2017.08.026

[40] Hagihira S. (2015). Changes in the electroencephalogram during anaesthesia and their physiological basis. *British journal of anaesthesia*, 115 Suppl 1, i27–i31. https://doi.org/10.1093/bja/aev212

[41] Purdon, P. L., Sampson, A., Pavone, K. J., & Brown, E. N. (2015). Clinical Electroencephalography for Anesthesiologists: Part I: Background and Basic Signatures. *Anesthesiology*, 123(4), 937–960. https://doi.org/10.1097/ALN.0000000000000841

[42] Abhang, P. A., & Gawali, B. W. (2015). Correlation of EEG images and speech signals for emotion analysis. *British Journal of Applied Science & Technology*, 10(5), 1-13.

[43] Singh, H. (1999). Bispectral index (BIS) monitoring during propofol-induced sedation and anaesthesia. *European Journal of Anaesthesiology*, 16(1), 31-36. doi:10.1046/j.1365-2346.1999.00420.x

[44] Mathur, S., Patel, J., Goldstein, S., & Jain, A. (2021). Bispectral Index. In StatPearls. *StatPearls Publishing*.

[45] Willingham, M., Ben Abdallah, A., Gradwohl, S., Helsten, D., Lin, N., Villafranca, A., Jacobsohn, E., Avidan, M., & Kaiser, H. (2014). Association between intraoperative electroencephalographic suppression and postoperative mortality. *British journal of anaesthesia*, 113(6), 1001–1008. https://doi.org/10.1093/bja/aeu105

[46] Short, T. G., & Leslie, K. (2014). 'Known unknowns and unknown unknowns': electroencephalographic burst suppression and mortality. *British journal of anaesthesia*, 113(6), 897–899. https://doi.org/10.1093/bja/aeu147

[47] Soehle, M., Dittmann, A., Ellerkmann, R.K. et al. (2015). Intraoperative burst suppression is associated with postoperative delirium following cardiac surgery: a prospective, observational study. *BMC Anesthesiol* 15, 61. https://doi.org/10.1186/s12871-015-0051-7

[48] Fritz, B. A., Kalarickal, P. L., Maybrier, H. R., Muench, M. R., Dearth, D., Chen, Y., Escallier, K. E., Ben Abdallah, A., Lin, N., & Avidan, M. S. (2016). Intraoperative Electroencephalogram

Suppression Predicts Postoperative Delirium. *Anesthesia and analgesia*, 122(1), 234–242. https://doi.org/10.1213/ANE.0000000000000989

[49] Wildes, T. S., Mickle, A. M., Ben Abdallah, A., Maybrier, H. R., Oberhaus, J., Budelier, T. P., Kronzer, A., McKinnon, S. L., Park, D., Torres, B. A., Graetz, T. J., Emmert, D. A., Palanca, B. J., Goswami, S., Jordan, K., Lin, N., Fritz, B. A., Stevens, T. W., Jacobsohn, E., Schmitt, E. M., ENGAGES Research Group (2019). Effect of Electroencephalography-Guided Anesthetic Administration on Postoperative Delirium Among Older Adults Undergoing Major Surgery: The ENGAGES Randomized Clinical Trial. *JAMA*, 321(5), 473–483. https://doi.org/10.1001/jama.2018.22005

[50] Vinterbo S. (1999). Predictive models in medicine: some methods for construction and adaptation. Norwegian University of Science and Technology. *Ph.D. Thesis*.

[51] Alanazi, H.O., Abdullah, A.H. & Qureshi, K.N. (2017). A Critical Review for Developing Accurate and Dynamic Predictive Models Using Machine Learning Methods in Medicine and Health Care. *J Med Syst* 41, 69. https://doi.org/10.1007/s10916-017-0715-6

[52] Kumar, G., Kumar, K., & Sachdeva, M. (2010). The use of artificial intelligence based techniques for intrusion detection: a review. *Artificial Intelligence Review*, 34(4), 369-387.

[53] Mountrakis, G., Im, J., & Ogole, C. (2011). Support vector machines in remote sensing: A review. ISPRS Journal of Photogrammetry and Remote Sensing, 66(3), 247-259.

[54] Marius, H. (2020). Multiclass Classification with Support Vector Machines (SVM), Dual Problem and Kernel Functions. *Towards Data Science*. [Available online] https://towardsdatascience.com/

[55] Zhang Z. (2016). Introduction to machine learning: k-nearest neighbors. *Annals of translational medicine*, 4(11), 218. https://doi.org/10.21037/atm.2016.03.37

[56] Alzubi, J., Nayyar, A., & Kumar, A. (2018). Machine learning from theory to algorithms: an overview. *In Journal of physics: conference series* (Vol. 1142, No. 1, p. 012012). IOP Publishing.

[57] Shaik A.B., Srinivasan S. (2019) A Brief Survey on Random Forest Ensembles in Classification Model. In: Bhattacharyya S., Hassanien A., Gupta D., Khanna A., Pan I. (eds) International Conference on Innovative Computing and Communications. *Lecture Notes in Networks and Systems*, vol 56. Springer, Singapore. https://doi-org.sire.ub.edu/10.1007/978-981-13-2354-6_27

[58] Azmi, S. S., & Baliga, S. (2020). An Overview of Boosting Decision Tree Algorithms utilizing AdaBoost and XGBoost *Boosting strategies. International Research Journal of Engineering and Technology (IRJET),* 7(5).

[59] Dhaliwal, S. S., Nahid, A. A., & Abbas, R. (2018). Effective intrusion detection system using XGBoost. *Information*, 9(7), 149.

[60] Mandrekar J. N. (2010). Receiver operating characteristic curve in diagnostic test assessment. *Journal of thoracic oncology : official publication of the International Association for the Study of Lung Cancer*, 5(9), 1315–1316. https://doi.org/10.1097/JTO.0b013e3181ec173d

[61] Chemali, J. J., Wong, K. F., Solt, K., & Brown, E. N. (2011). A state-space model of the burst suppression ratio. Annual International Conference of the IEEE Engineering in Medicine and Biology Society. *IEEE Engineering in Medicine and Biology Society*. Annual International Conference, 2011, 1431–1434. https://doi.org/10.1109/IEMBS.2011.6090354

[62] Battineni, G., Sagaro, G. G., Chinatalapudi, N., & Amenta, F. (2020). Applications of Machine Learning Predictive Models in the Chronic Disease Diagnosis. *Journal of personalized medicine*, 10(2), 21. https://doi.org/10.3390/jpm10020021

[63] MATLAB Product Description - MATLAB & Simulink - MathWorks. (2021). MathWorks.Com. https://es.mathworks.com/help/matlab/learn_matlab/product-description.html?lang=en

[64] RStudio | Open source & professional software for data science teams. (2021). RStudio. https://www.rstudio.com/

[65] What is Python? Executive Summary. (2021). Python.Org. https://www.python.org/doc/essays/blurb/

[66] Sarker, I. H. (2021). Machine learning: Algorithms, real-world applications and research directions. SN Computer Science, 2(3), 1-21.

[67] Cortes, C., & Mohri, M. (2005). Confidence intervals for the area under the ROC Curve. *In Advances in Neural Information Processing Systems 17* - Proceedings of the 2004 Conference, NIPS 2004 (Advances in Neural Information Processing Systems). Neural information processing systems foundation.

[68] Kottas, M., Kuss, O., & Zapf, A. (2014). A modified Wald interval for the area under the ROC curve (AUC) in diagnostic case-control studies. *BMC medical research methodology*, 14, 26. https://doi.org/10.1186/1471-2288-14-26

[69] Rengasamy, D., Rothwell, B. C., & Figueredo, G. P. (2021). Towards a More Reliable Interpretation of Machine Learning Outputs for Safety-Critical Systems Using Feature Importance Fusion. *Applied Sciences*, 11(24), 11854.

[70] BOE-A-2018-16673-C, Ley Orgánica 3/2018, de 5 de desembre de 2018, de Protecció de Dades Personals i Garantia dels Drets Digitals. *Boletín Oficial del Estado, BOE*. [Available online] https://www.boe.es/buscar/doc.php?id=BOE-A-2018-16673

[71] US Food and Drug Administration (2019). Proposed Regulatory Framework for Modifications to Artificial Intelligence/Machine Learning (AI/ML)-Based Software as Medical Device (SaMD) Action Plan. [Available online] https://www.fda.gov/media/145022/download

APPENDICES

```
# -*- coding: utf-8 -*-
"""
Created on Thu Oct 28 17:37:49 2021

@author: Joana
"""

import pandas as pd
import os
import numpy as np
import statsmodels.api as sm
from matplotlib import pyplot as plt
import scipy.stats
from sklearn import svm
from sklearn.metrics import roc_auc_score, roc_curve, accuracy_score
from sklearn.neighbors import KNeighborsClassifier
from sklearn.ensemble import RandomForestClassifier
from xgboost import XGBClassifier
import seaborn as sns

directory = os.getcwd()

# ============================================================================
# FUNCTION TO CHECK IF A LIST HAS ALL EQUAL VALUES
# ============================================================================

def is_unique(s):
    a = s.to_numpy()
    return (a[0] == a).all()

# ============================================================================
# FUNCTION TO CREATE A MATRIX WITH DATA FROM ALL PATIENTS. -FILE PATH-
# MUST SPECIFY WHETHER DATA COMES FROM TRAIN OR TEST DATASET
# ============================================================================

def create_matrix(file_path):

    db_in = pd.read_csv(file_path)

    # DELETE IRRELEVANT FEATURES

    del [db_in['WEIGHT'], db_in['HEIGHT'], db_in['SYSTIME'],
    db_in['EVENT'], db_in['REMI_RATE'], db_in['REMI_VOL'],
db_in['REMI_CT'],
    db_in['PROPO_RATE'], db_in['PROPO_VOL'], db_in['PROPO_CT'],
db_in['SEF'],
    db_in['EMG'], db_in['TOTPOW'], db_in['TV'], db_in['RR'],
    db_in['RR_CO2'], db_in['PLETH_SAT_O2'], db_in['PLETH_HR'],
db_in['NIBP_DIA'],
    db_in['NIBP_SYS'], db_in['ID'], db_in['BMI'], db_in['REMI_CP'],
db_in['PROPO_CP']]
```

```
    # DELETE ROWS WITHOUT SR VALUES

    for i in range(len(db_in)):
        if pd.isna(db_in.loc[i, 'SR']) == True:
            db_in = db_in.drop(i)
    db_in = db_in.reset_index(drop=True)

    # CREATE A NEW COLUMN WITH SR VALUES WITHIN 120 SECONDS

    future_sr = []
    future_sr_arr = np.empty((120,1))
    future_sr_arr[:] = np.nan
    future_sr_arr = future_sr_arr.flatten()
    future_sr_nan = future_sr_arr.tolist()
    db_in = db_in.reset_index(drop=True)

    for i in range(len(db_in)-120):
        future_sr.append(db_in['SR'][i+120])
    future_sr = future_sr + future_sr_nan

    db_in['FUTURE_SR'] = future_sr
    db_in = db_in.reset_index(drop=True)

    # DELETE ROWS WITH SOME NaN VALUE OR SQI<60

    for column in list(db_in):
        for i in range(len(db_in)):
            if pd.isna(db_in.loc[i, column]) == True:
                db_in = db_in.drop(i)
        db_in = db_in.reset_index(drop=True)

    for i in range(len(db_in)):
        if db_in.at[i,'SQI'] <= 60:
            db_in = db_in.drop(i)

    db_in = db_in.reset_index(drop=True)

    # REPLACE VALUES IN FUTURE_SR>0 FOR 1, IT MEANS THERE IS BS

    for i in range(len(db_in)):
        if db_in.at[i, 'FUTURE_SR'] > 0:
            db_in.at[i, 'FUTURE_SR'] = 1
    db_in = db_in.reset_index(drop=True)

    # DELETE SQI AND SR COLUMNS

    del [db_in['SQI'], db_in['SR']]

    # IF THE PATIENT IS A MAN OR IF BS DOES NOT OCCUR, IT IS NOT USED

    try:
        if is_unique(db_in['FUTURE_SR']) == True:
            del db_in
            return

        elif db_in['SEX'][0] != 'F':
            del db_in
```

```
            return

        else:
            del db_in['SEX']
            return db_in

    except:
        return

# ============================================================================
# FUNCTION TO REMOVE OUTLIERS FROM BELOW PERCENTILE 15 AND OVER #
# PERCENTIL 85 IF DATA DISTRIBUTION IS NORMAL
# ============================================================================

def outlier_removal_normal(df, df_normal_data):
    df_numeric = df_normal_data.select_dtypes(include=np.number)
    Q1 = df_numeric.quantile(0.15)
    Q3 = df_numeric.quantile(0.85)
    IQR = Q3 - Q1
    outliers = (df_numeric<(Q1-1.5*IQR))|(df_numeric>(Q3 + 1.5*IQR))
    indexes = []
    for idx, row in outliers.iterrows():
        if True in row.tolist():
            indexes.append(idx)
    df = df.drop(indexes)
    df = df.reset_index(drop=True)

    return df

# ============================================================================
# FUNCTION TO SCALE AND NORMALIZE EACH FEATURE TO VALUES RANGING 0 - 1
# ============================================================================

def normalize(df):
    df = (df-df.min())/(df.max()-df.min())

    return df

# ============================================================================
# FUNCTION TO OBTAIN DISTRIBUTION QQ-PLOTS
# ============================================================================

def plot_distributions(df, distributions, distributions_names, t):
    for col in df.columns:
        fig, axes = plt.subplots(ncols=2, nrows=2, sharex=False)
        x = 0
        for k, ax in zip(df.columns, np.ravel(axes)):
            sm.qqplot(df[col], distributions[x], line='s', ax=ax,
markerfacecolor='Maroon', markeredgecolor='Maroon', alpha=0.9,
marker='.', markersize = 0.1)

ax.set_title(str(distributions_names[distributions.index(distributions
[x])]), fontsize=14)
            ax.xaxis.get_label().set_fontsize(12)
            ax.yaxis.get_label().set_fontsize(12)
            ax.get_lines()[1].set_color("Darkslategrey")
            ax.get_lines()[1].set_linewidth("2")
```

```
            x = x+1
        fig.suptitle(f'{col}', size = 16)
        plt.tight_layout()

        fig.savefig(os.path.join(directory, 'QQ-Plots', str(col) + '_'
+ t + '.png'))
        plt.close(fig)

# ================================================================
# FUNCTION TO FIND THE OPTIMAL THRESHOLD IN TERMS OF ACCURACY
# ================================================================

def Find_Optimal_Cutoff(target, predicted):

    fpr, tpr, threshold = roc_curve(target, predicted)
    i = np.arange(len(tpr))
    roc = pd.DataFrame({'tf' : pd.Series(tpr-(1-fpr), index=i),
'threshold' : pd.Series(threshold, index=i)})
    roc_t = roc.iloc[(roc.tf-0).abs().argsort()[:1]]

    return list(roc_t['threshold'])

# ================================================================
# FUNCTION TO APPLY A MODEL TO ALL TEST PATIENTS AND OBTAINS RESULTS
# FROM THE BEST AND WORST PERFORMANCES
# ================================================================

def model_example_performance(ml_model, threshold):
    highest_accuracy = 0
    lowest_accuracy = 1
    highest_accuracy_predicted_data = []
    highest_accuracy_real_data = []
    lowest_accuracy_predicted_data = []
    lowest_accuracy_real_data = []
    highest_accuracy_patient_time = []
    lowest_accuracy_patient_time = []
    accuracies = []

    for file in os.listdir(os.path.join(directory, 'Database',
'Test')):
        try:
            df_pat = create_matrix(os.path.join(directory, 'Database',
'Test', file))
            df_pat_time = df_pat['TIME']
            df_pat = df_pat.drop(['TIME'], axis = 1)

            df_pat_age = df_pat.drop(['REMI_CE', 'PROPO_CE', 'BIS',
'ECG_HR', 'NIBP_MEAN', 'FUTURE_SR'], axis = 1)
            df_pat_other = df_pat.drop(['AGE'], axis = 1)
            df_pat_other = normalize(df_pat_other)
            df_pat_age = df_pat_age/100

            df_pat = pd.concat([df_pat_age, df_pat_other], axis=1)

            df_pat_np = df_pat.to_numpy()

            x_test, y_test = df_pat_np[:, :-1], df_pat_np[:, -1]
```

```
            y_predicted_proba = ml_model.predict_proba(x_test)[:, -1]

            y_predicted = ml_model.predict(x_test)
            accuracy = accuracy_score(y_test, y_predicted)
            accuracies.append(round(accuracy, 2))

            if accuracy >= highest_accuracy:
                highest_accuracy = accuracy
                highest_accuracy_predicted_data = y_predicted_proba
                highest_accuracy_real_data = y_test
                highest_accuracy_patient_time = df_pat_time.to_list()

            if accuracy <= lowest_accuracy:
                lowest_accuracy = accuracy
                lowest_accuracy_predicted_data = y_predicted_proba
                lowest_accuracy_real_data = y_test
                lowest_accuracy_patient_time = df_pat_time.to_list()

        except:
            continue

    highest_accuracy_threshold =
np.where(highest_accuracy_predicted_data>=threshold, 1, 0)
    lowest_accuracy_threshold =
np.where(lowest_accuracy_predicted_data>=threshold, 1, 0)

    return (highest_accuracy, lowest_accuracy,
            highest_accuracy_predicted_data,
highest_accuracy_real_data,
            lowest_accuracy_predicted_data, lowest_accuracy_real_data,
            highest_accuracy_patient_time,
lowest_accuracy_patient_time,
            accuracies,
            highest_accuracy_threshold, lowest_accuracy_threshold)

# ======================================================================
# FUNCTION TO OBTAIN FEATURE IMPORTANCE BARPLOT
# ======================================================================

def show_values_on_bars(axs, h_v, space):
    def _show_on_single_plot(ax):
        if h_v == "v":
            for p in ax.patches:
                _x = p.get_x() + p.get_width() / 2
                _y = p.get_y() + p.get_height() +
(p.get_height()*0.01)
                value = round(float(p.get_height()), 3)
                ax.text(_x, _y, value, ha = "center")
        elif h_v == "h":
            for p in ax.patches:
                _x = p.get_x() + p.get_width() + float(space)
                _y = p.get_y() + p.get_height() - (p.get_height()*0.5)
                value = round(float(p.get_width()), 3)
                ax.text(_x, _y, value, ha="left")

    if isinstance(axs, np.ndarray):
        for idx, ax in np.ndenumerate(axs):
```

```
            _show_on_single_plot(ax)
    else:
        _show_on_single_plot(axs)

# ====================================================================
# FUNCTION THAT CHECKS IF A PATIENT HAS BS OCCURRENCE PERIODS
# ====================================================================

def has_burst_suppression(df):
    df['SR'] = df['SR'].fillna(0)
    for i in range (len(df)):
        if (df['SR'] == 0).all() == False:
            return 1
        else:
            return 0

# ====================================================================
# FUNCTION THAT CLASSIFIES PATIENTS IN GROUPS OF AGE
# ====================================================================

def classify_ages(ages):
    age_groups = []
    for age in ages:
        if 0<=age<20:
            age_groups.append('0-19')
        if 20<=age<40:
            age_groups.append('20-39')
        if 40<=age<60:
            age_groups.append('40-59')
        if 60<=age<80:
            age_groups.append('60-79')
        if 80<=age:
            age_groups.append('>80')
    return age_groups

# ====================================================================
# CHECK BURST SUPPRESSION INCIDENCE BY GROUPS OF AGE
# ====================================================================

####### BS PERCENTAGE BARPLOT BY AGE GROUP AND NUMBER OF PATIENTS
DISTRIBUTED BY AGE GROUP

is_bs = []
number_patients_train = 0
number_patients_test = 0
ages = []

for file in os.listdir(os.path.join(directory, 'Database', 'Train')):
    df_patient = pd.read_csv(os.path.join(directory, 'Database',
'Train', file))
    if pd.isna(df_patient['AGE'][0]) == False:
        ages.append(df_patient['AGE'][0])
        is_bs.append(has_burst_suppression(df_patient))
        number_patients_train = number_patients_train + 1
    else:
        continue
```

```
for file in os.listdir(os.path.join(directory, 'Database', 'Test')):
    df_patient = pd.read_csv(os.path.join(directory, 'Database',
'Test', file))
    if pd.isna(df_patient['AGE'][0]) == False:
        ages.append(df_patient['AGE'][0])
        is_bs.append(has_burst_suppression(df_patient))
        number_patients_test = number_patients_test + 1
    else:
        continue

my_palette = {'0-19': 'Olivedrab', '20-39': 'Maroon', '40-59':
'DarkOrange', '60-79': 'Tomato', '>80': 'Darkslategrey'}

bs_ages = {'Burst Suppression': is_bs, 'Age group':
classify_ages(ages)}

plt.figure()
ax = sns.barplot(x='Age group', y='Burst Suppression', data=bs_ages,
            estimator=lambda x: sum(x==1)*100.0/len(x), palette =
my_palette,
            order = ['0-19', '20-39', '40-59', '60-79', '>80'])
for p in ax.patches:
   ax.annotate('{:.1f}'.format(p.get_height()), (p.get_x()+0.05,
p.get_height()+1.5))

plt.title('Burst Suppression occurrence \n for each age group',
fontsize = 16)
plt.xlabel('Groups of ages (years)', fontsize = 12)
plt.ylabel('Burst Suppression occurrence (%)', fontsize = 12)
plt.gca().spines['top'].set_visible(False)
plt.gca().spines['right'].set_visible(False)
plt.tight_layout()

plt.savefig(os.path.join(directory, 'Other_plots',
'BS_percentage_per_age_group.png'))
plt.close()

plt.figure()
ax = sns.countplot(data=bs_ages, x="Age group", order = ['0-19', '20-
39', '40-59', '60-79', '>80'],
             palette = my_palette)
for p in ax.patches:
   ax.annotate('{:.1f}'.format(p.get_height()), (p.get_x()+0.25,
p.get_height()+1.5))

plt.title('Number of patients \n for each age group', fontsize = 16)
plt.xlabel('Groups of ages (years)', fontsize = 12)
plt.ylabel('Number of patients', fontsize = 12)
plt.gca().spines['top'].set_visible(False)
plt.gca().spines['right'].set_visible(False)
plt.tight_layout()

plt.savefig(os.path.join(directory, 'Other_plots',
'Number_of_patients_per_age_group.png'))
plt.close()

del number_patients_train, number_patients_test
```

```
# ========================================================================
# MATRIX OBTENTION
# ========================================================================

###### TRAINING

frames = []
number_patients_train = 0
ages = []

for file in os.listdir(os.path.join(directory, 'Database', 'Train')):
    df_patient = create_matrix(os.path.join(directory, 'Database',
'Train', file))
    try:
        ages.append(df_patient['AGE'][0])
        frames.append(df_patient)
        number_patients_train = number_patients_train + 1
    except:
        continue

print('Number of patients for the Training: ', number_patients_train)

df_train = normalize(pd.concat(frames))
df_train = df_train.reset_index(drop=True)
df_train = df_train.drop(['TIME'], axis = 1)

d_ages = {'Group': classify_ages(ages), 'Age': ages}

my_palette = {'0-19': 'Olivedrab', '20-39': 'Maroon', '40-59':
'DarkOrange', '60-79': 'Tomato', '>80': 'Darkslategrey'}

# AGES BOXPLOT TRAIN

plt.figure()
sns.boxplot(x = "Group", y = "Age", data = d_ages,
            order = ['0-19', '20-39', '40-59', '60-79', '>80'],
            palette = my_palette, showmeans = True,
            meanprops = {"marker":"o",
                       "markerfacecolor":"white",
                       "markeredgecolor":"black",
                      "markersize":"5"})
plt.title('Ages distribution of the Train patients', fontsize = 16)
plt.xlabel('Groups of ages (years)', fontsize = 12)
plt.ylabel('Ages', fontsize = 12)
plt.gca().spines['top'].set_visible(False)
plt.gca().spines['right'].set_visible(False)
plt.tight_layout()

plt.savefig(os.path.join(directory, 'Other_plots', 'ages_train.png'))
plt.close()

del frames, ages
```

```
###### TEST

frames = []
number_patients_test = 0
ages = []

for file in os.listdir(os.path.join(directory, 'Database', 'Test')):
    df_patient = create_matrix(os.path.join(directory, 'Database',
'Test', file))
    try:
        ages.append(df_patient['AGE'][0])
        frames.append(df_patient)
        number_patients_test = number_patients_test + 1
    except:
        continue

print('Number of patients for the Test: ', number_patients_test)

df_test = normalize(pd.concat(frames))
df_test = df_test.reset_index(drop=True)
df_test = df_test.drop(['TIME'], axis = 1)

d_ages = {'Group': classify_ages(ages), 'Age': ages}

# AGES BOXPLOT TEST

plt.figure()
sns.boxplot(x="Group", y="Age", data=d_ages,
            order=['0-19', '20-39', '40-59', '60-79', '>80'],
            palette=my_palette, showmeans=True,
            meanprops={"marker":"o",
                       "markerfacecolor":"white",
                       "markeredgecolor":"black",
                      "markersize":"5"})
plt.title('Ages distribution of the Test patients', fontsize = 16)
plt.xlabel('Groups of ages (years)', fontsize = 12)
plt.ylabel('Ages', fontsize = 12)
plt.gca().spines['top'].set_visible(False)
plt.gca().spines['right'].set_visible(False)
plt.tight_layout()

plt.savefig(os.path.join(directory, 'Other_plots', 'ages_test.png'))
plt.close()

del frames, ages

###### GET DATA FOR OUTLIERS ANALYSIS

continuous_distributions = [scipy.stats.norm, scipy.stats.uniform,
scipy.stats.expon, scipy.stats.laplace]
continuous_distributions_names = ['Normal', 'Uniform', 'Exponential',
'Laplace']

df_continuous = df_train.drop(['AGE', 'FUTURE_SR', 'NIBP_MEAN',
'ECG_HR'], axis = 1)
```

```
# ========================================================================
# QQ-PLOTS OBTENTION
# ========================================================================

t = 'BOR'

plot_distributions(df_train, continuous_distributions,
continuous_distributions_names, t)

p_values_bor = {}
for col in df_train.columns:
    p_values_bor[col] = scipy.stats.normaltest(df_train[col])[1]

print('P-values before Outliers removal: ', p_values_bor)

t = 'AOR'

df_outlier = outlier_removal_normal(df_train, df_continuous)
plot_distributions(df_outlier, continuous_distributions,
continuous_distributions_names, t)

p_values_aor = {}
for col in df_continuous.columns:
    p_values_aor[col] = scipy.stats.normaltest(df_outlier[col])[1]

print('P-values after Outliers removal: ', p_values_aor)

# ========================================================================
# CORRELATION MATRIX WITH THE TRAIN PATIENTS
# ========================================================================

plt.figure()
corr_train = df_train.corr()
sns.heatmap(corr_train, annot=True, cmap = 'YlOrBr', linewidths=.5,
vmin=-1, vmax=1)
plt.tight_layout()
plt.savefig(os.path.join(directory, 'Other_plots',
'Correlation_matrix.png'))
plt.close()

# ========================================================================
# ========================================================================
#                         MACHINE LEARNING
# ========================================================================
# ========================================================================

# ========================================================================
# SEPARATE DATA IN X_train, Y_train, X_test, Y_test
# ========================================================================

df_train_np = df_train.to_numpy()
df_test_np = df_test.to_numpy()
features_names = df_test.columns[:-1]

x_train, x_test, y_train, y_test = df_train_np[:, :-1], df_test_np[:,
:-1], df_train_np[:, -1], df_test_np[:, -1]
```

```
# ======================================================================
# SVM KERNELS AND GAMMAS TOGETHER
# ======================================================================

kernels = ['linear', 'poly', 'rbf', 'sigmoid']
kernels_names = ['Linear', 'Polynomial', 'RBF', 'Sigmoid']
gammas = [0.25, 2, 7, 10]
colors = ['Maroon', 'Olivedrab', 'Darkslategrey', 'Tomato']
line_styles = ['solid', 'dotted', 'dashed', 'dashdot']
aucs = []
fpr = []
tpr = []
k = []
g = []
aucs_final = []
kernels_final = []
gammas_final = []
th = []
th_final = []

number_of_rows = x_train.shape[0]
random_indices = np.random.choice(number_of_rows,
size=int(number_of_rows*0.01), replace=False)
random_rows_x = x_train[random_indices, :]
random_rows_y = y_train[random_indices]

plt.figure(figsize=(7,7))

for i in range(len(kernels)):
    for j in range(len(gammas)):
        SVM_classifier = svm.SVC(kernel = kernels[i], C = 7,
probability = True,
                                 gamma = gammas[j], random_state =
95)
        SVM_classifier.fit(random_rows_x, random_rows_y)
        y_pred_svm = SVM_classifier.predict_proba(x_test)[:, -1]

        print('Training with gamma ', gammas[j], ' and kernel kernel
', kernels[i])

        false_positive_rate, true_positive_rate, thresholds =
roc_curve(y_test, y_pred_svm)
        th.append(Find_Optimal_Cutoff(y_test, y_pred_svm)[0])
        aucs.append(roc_auc_score(y_test, y_pred_svm))
        fpr.append(false_positive_rate)
        tpr.append(true_positive_rate)
        k.append(kernels_names[i])
        g.append(gammas[j])

for i in range(4):
    max_index = aucs.index(max(aucs))
    kernels_final.append(k[max_index])
    gammas_final.append(g[max_index])
    aucs_final.append(aucs[max_index])
    th_final.append(th[max_index])

    plt.axis('scaled')
```

```
    plt.xlim([0, 1])
    plt.ylim([0, 1])
    plt.title('Support Vector Machine model', weight='bold', fontsize
= 16)
    plt.plot(fpr[max_index], tpr[max_index], color = colors[i], alpha
= 0.5, lw = 2, linestyle = line_styles[i])
    plt.xlabel("1 - Specificity", fontsize = 12)
    plt.ylabel("Sensitivity", fontsize = 12)
    plt.gca().spines['top'].set_visible(False)
    plt.gca().spines['right'].set_visible(False)

    del aucs[max_index], fpr[max_index], tpr[max_index], k[max_index],
g[max_index], th[max_index]

plt.legend(['%s with Gamma = %s (AUC = %s) - Threshold = %s' %
(kernels_final[0], gammas_final[0], round(aucs_final[0], 3),
round(th_final[0], 3)),
            '%s with Gamma = %s (AUC = %s)' % (kernels_final[1],
gammas_final[1], round(aucs_final[1], 3)),
            '%s with Gamma = %s (AUC = %s)' % (kernels_final[2],
gammas_final[2], round(aucs_final[2], 3)),
            '%s with Gamma = %s (AUC = %s)' % (kernels_final[3],
gammas_final[3], round(aucs_final[3], 3))],
            fontsize = 12, loc = 4)

plt.axvline(th_final[0], color = 'maroon', linewidth = 1)
plt.tight_layout()
plt.plot([0, 1], [0, 1], linestyle='--', lw=2, color='Black',
label='Random guess')
plt.savefig(os.path.join(directory, 'ROC_curves',
'ROC_curves_SVC.png'))
plt.close()

del false_positive_rate, true_positive_rate, thresholds, aucs,
SVM_classifier, th

# ========================================================================
# KNN
# ========================================================================

number_neighbors = [10, 50, 100, 500]
aucs = []
th = []
plt.figure(figsize=(7,7))

for i in range(len(number_neighbors)):
    KNN_classifier = KNeighborsClassifier(n_neighbors =
number_neighbors[i],
                                          weights = 'uniform',
algorithm = 'auto',
                                          leaf_size = 5, n_jobs = -1)

    KNN_classifier.fit(x_train, y_train)
    y_pred_knn = KNN_classifier.predict(x_test)

    print('Training with ', number_neighbors[i], ' neighbors')
```

```
    false_positive_rate, true_positive_rate, thresholds =
roc_curve(y_test, y_pred_knn)
    th.append(Find_Optimal_Cutoff(y_test, y_pred_knn)[0])
    aucs.append(roc_auc_score(y_test, y_pred_knn))

    plt.axis('scaled')
    plt.xlim([0, 1])
    plt.ylim([0, 1])
    plt.title('K-Nearest Neighbors model', weight='bold', fontsize =
16)
    plt.plot(false_positive_rate, true_positive_rate, 'g', color =
colors[i], alpha = 0.5, lw = 2, linestyle = line_styles[i])
    plt.xlabel("1 - Specificity", fontsize = 12)
    plt.ylabel("Sensitivity", fontsize = 12)
    plt.gca().spines['top'].set_visible(False)
    plt.gca().spines['right'].set_visible(False)

plt.legend(['%s neighbors (AUC = %s)' % (number_neighbors[0],
round(aucs[0], 3)),
            '%s neighbors (AUC = %s)' % (number_neighbors[1],
round(aucs[1], 3)),
            '%s neighbors (AUC = %s)' % (number_neighbors[2],
round(aucs[2], 3)),
            '%s neighbors (AUC = %s)' % (number_neighbors[3],
round(aucs[3], 3))],
            fontsize = 12, loc = 4)

plt.tight_layout()
plt.plot([0, 1], [0, 1], linestyle='--', lw=2, color='Black',
label='Random guess')
plt.savefig(os.path.join(directory, 'ROC_curves',
'ROC_curves_KNN.png'))
plt.close()

del false_positive_rate, true_positive_rate, thresholds, aucs, th

# ============================================================================
# RANDOM FOREST
# ============================================================================

number_estimators = [50, 100, 200, 500]
plt.figure(figsize=(7,7))

aucs = []
th = []

for i in range(len(number_estimators)):

    RF_classifier = RandomForestClassifier(n_estimators =
number_estimators[i],
                                           min_samples_split = 5,
min_samples_leaf = 2,
                                           max_depth = 10, n_jobs = -
1, random_state = 95)

    RF_classifier.fit(x_train, y_train)
    y_pred_rf = RF_classifier.predict_proba(x_test)[:,-1]
```

```
    print('Training with ', number_estimators[i], ' estimators')

    false_positive_rate, true_positive_rate, thresholds =
roc_curve(y_test, y_pred_rf)
    th.append(Find_Optimal_Cutoff(y_test, y_pred_rf)[0])
    aucs.append(roc_auc_score(y_test, y_pred_rf))

    plt.axis('scaled')
    plt.xlim([0, 1])
    plt.ylim([0, 1])
    plt.title('Random Forest model', weight='bold', fontsize = 16)
    plt.plot(false_positive_rate, true_positive_rate, 'g', color =
colors[i], alpha = 0.5, lw = 2, linestyle = line_styles[i])
    plt.xlabel("1 - Specificity", fontsize = 12)
    plt.ylabel("Sensitivity", fontsize = 12)
    plt.gca().spines['top'].set_visible(False)
    plt.gca().spines['right'].set_visible(False)

plt.legend(['%s estimators (AUC = %s)' % (number_estimators[0],
round(aucs[0], 3)),
            '%s estimators (AUC = %s)' % (number_estimators[1],
round(aucs[1], 3)),
            '%s estimators (AUC = %s)' % (number_estimators[2],
round(aucs[2], 3)),
            '%s estimators (AUC = %s)' % (number_estimators[3],
round(aucs[3], 3))],
           fontsize = 12, loc = 4)

plt.tight_layout()
plt.plot([0, 1], [0, 1], linestyle='--', lw=2, color='Black',
label='Random guess')
plt.savefig(os.path.join(directory, 'ROC_curves',
'ROC_curves_RF.png'))
plt.close()

del false_positive_rate, true_positive_rate, thresholds, aucs, th

# =============================================================================
# RANDOM FOREST FEATURE IMPORTANCE - 100 NEIGHBORS
# =============================================================================

RF_classifier = RandomForestClassifier(n_estimators = 100,
                                       min_samples_split = 5,
min_samples_leaf = 2,
                                       max_depth = 10, n_jobs = -
1, random_state = 95)

RF_classifier.fit(x_train, y_train)
y_pred_rf = RF_classifier.predict_proba(x_test)[:,-1]

importance = RF_classifier.feature_importances_

data={'feature_names':features_names,'feature_importance':importance}
fi_df = pd.DataFrame(data)
```

```
my_palette_2 = {'AGE': 'DarkOrange', 'REMI_CE': 'SeaGreen',
'PROPO_CE': 'Maroon', 'BIS': 'Tomato', 'ECG_HR': 'Olivedrab',
'NIBP_MEAN': 'Darkslategrey'}

# plot feature importance
plt.figure()
ax = sns.barplot(x = fi_df['feature_importance'], y =
fi_df['feature_names'], palette = my_palette_2)
show_values_on_bars(ax, "h", 0.005)
plt.title('Random Forest Feature Importance', fontsize = 16)
plt.xlabel('Feature importance', fontsize = 12)
plt.ylabel('Feature', fontsize = 12)
axs = plt.gca()
right_side = axs.spines["right"]
right_side.set_visible(False)
upper_side = axs.spines["top"]
upper_side.set_visible(False)
plt.tight_layout()
plt.savefig(os.path.join(directory, 'Feature_importance',
'Feature_importance_RF_nosorted.png'))
plt.close()

del importance, fi_df, ax, axs, right_side, upper_side

# ======================================================================
# XGBOOST
# ======================================================================

number_estimators = [50, 100, 200, 500]
plt.figure(figsize=(7,7))

aucs = []
th = []

for i in range(len(number_estimators)):

    XGboost_classifier = XGBClassifier(n_estimators =
number_estimators[i], eval_metric='mlogloss',
                                       max_depth = 10, n_jobs = -
1, random_state = 95,
                                       use_label_encoder =False)

    XGboost_classifier.fit(x_train, y_train)
    y_pred_xgb = XGboost_classifier.predict_proba(x_test)[:,-1]

    print('Testing with ', number_estimators[i], ' estimators')

    false_positive_rate, true_positive_rate, thresholds =
roc_curve(y_test, y_pred_xgb)
    th.append(Find_Optimal_Cutoff(y_test, y_pred_xgb)[0])
    aucs.append(roc_auc_score(y_test, y_pred_xgb))

    plt.axis('scaled')
    plt.xlim([0, 1])
    plt.ylim([0, 1])
    plt.title('XGBoost model', weight='bold', fontsize = 16)
```

```
    plt.plot(false_positive_rate, true_positive_rate, 'g', color =
colors[i], alpha = 0.5, lw = 2, linestyle = line_styles[i])
    plt.xlabel("1 - Specificity", fontsize = 12)
    plt.ylabel("Sensitivity", fontsize = 12)
    plt.gca().spines['top'].set_visible(False)
    plt.gca().spines['right'].set_visible(False)

plt.legend(['%s estimators (AUC = %s)' % (number_estimators[0],
round(aucs[0], 3)),
            '%s estimators (AUC = %s)' % (number_estimators[1],
round(aucs[1], 3)),
            '%s estimators (AUC = %s)' % (number_estimators[2],
round(aucs[2], 3)),
            '%s estimators (AUC = %s)' % (number_estimators[3],
round(aucs[3], 3))],
           fontsize = 12, loc = 4)

plt.tight_layout()
plt.plot([0, 1], [0, 1], linestyle='--', lw=2, color='Black',
label='Random guess')
plt.savefig(os.path.join(directory, 'ROC_curves',
'ROC_curves_XGB.png'))
plt.close()

del false_positive_rate, true_positive_rate, thresholds, aucs

# ============================================================================
# XGBOOST FEATURE IMPORTANCE - 100 NEIGHBORS
# ============================================================================

XGB_classifier = XGBClassifier(n_estimators = 100, use_label_encoder
=False,
                               eval_metric = 'mlogloss',
                               max_depth = 10, n_jobs = -1,
random_state = 95)

XGB_classifier.fit(x_train, y_train)
y_pred_xgb = XGB_classifier.predict_proba(x_test)[:,-1]

importance = XGB_classifier.feature_importances_

data={'feature_names':features_names,'feature_importance': importance}
fi_df = pd.DataFrame(data)

# plot feature importance
plt.figure()
ax = sns.barplot(x=fi_df['feature_importance'],
y=fi_df['feature_names'], palette = my_palette_2)
show_values_on_bars(ax, "h", 0.005)
plt.title('XGBoost Feature Importance', fontsize = 16)
plt.xlabel('Feature importance', fontsize = 12)
plt.ylabel('Feature', fontsize = 12)
axs = plt.gca()
right_side = axs.spines["right"]
right_side.set_visible(False)
upper_side = axs.spines["top"]
upper_side.set_visible(False)
```

```
plt.tight_layout()
plt.savefig(os.path.join(directory, 'Feature_importance',
'Feature_importance_XGB_nosorted.png'))
plt.close()

del importance, fi_df, ax, axs, right_side, upper_side

# ============================================================================
# TEST WITH PATIENTS
# ============================================================================

SVM_classifier = svm.SVC(kernel = 'linear', C = 7, probability = True,
gamma = 'auto', random_state = 95)
SVM_classifier.fit(random_rows_x, random_rows_y)

threshold = 0.191

highest_accuracy, lowest_accuracy, highest_accuracy_predicted_data,
highest_accuracy_real_data, lowest_accuracy_predicted_data,
lowest_accuracy_real_data, highest_accuracy_patient_time,
lowest_accuracy_patient_time, accuracies, highest_accuracy_threshold,
lowest_accuracy_threshold = model_example_performance(SVM_classifier,
threshold)

print('Highest accuracy = %s' % highest_accuracy, 'Lowest accuracy =
%s' % lowest_accuracy)

# ============================================================================
# BEST AND WORST PERFORMANCES PLOTS - BS VS. TIME
# ============================================================================

fig, axs = plt.subplots(3, gridspec_kw={'height_ratios': [3, 1, 1]},
figsize=(8, 10))
fig.suptitle('Lowest performance scenario', fontsize = 16)
axs[0].plot(lowest_accuracy_patient_time,
lowest_accuracy_predicted_data, color = 'Maroon', alpha = 0.75)
axs[1].plot(lowest_accuracy_patient_time, lowest_accuracy_threshold,
color = 'Darkslategrey', alpha = 0.75)
axs[2].plot(lowest_accuracy_patient_time, lowest_accuracy_real_data,
color = 'DarkOrange')
axs[0].axhline(threshold, linewidth=1, color = 'Olivedrab')
fig.supxlabel("Time", fontsize = 12)
axs[0].set(ylabel='Predicted BS (probability)')
axs[1].set(ylabel='Predicted BS')
axs[2].set(ylabel='Actual BS')
plt.ylim([0, 1])
plt.xlim(0)
sns.despine(left=False, bottom=False, right=True, top=True)
fig.tight_layout()
fig.align_labels()

fig.savefig(os.path.join(directory, 'Other_Plots',
'Lowest_performance.png'))
plt.close()

fig, axs = plt.subplots(3, gridspec_kw={'height_ratios': [3, 1, 1]},
figsize=(8, 10))
```

```
fig.suptitle('Highest performance scenario', fontsize = 16)
axs[0].plot(highest_accuracy_patient_time,
highest_accuracy_predicted_data, color = 'Maroon', alpha = 0.75)
axs[1].plot(highest_accuracy_patient_time, highest_accuracy_threshold,
color = 'Darkslategrey', alpha = 0.75)
axs[2].plot(highest_accuracy_patient_time, highest_accuracy_real_data,
color = 'DarkOrange')
axs[0].axhline(threshold, linewidth=1, color = 'Olivedrab')
fig.supxlabel("Time", fontsize = 12)
axs[0].set(ylabel='Predicted BS (probability)')
axs[1].set(ylabel='Predicted BS')
axs[2].set(ylabel='Actual BS')
plt.ylim([0, 1])
plt.xlim(0)
sns.despine(left=False, bottom=False, right=True, top=True)
fig.tight_layout()
fig.align_labels()

plt.savefig(os.path.join(directory, 'Other_Plots',
'Highest_performance.png'))
plt.close()

# ======================================================================
# BEST AND WORST PERFORMANCES PROBABILITY DISTRIBUTION
# ======================================================================

zeros_low = []
ones_low = []
zeros_high = []
ones_high = []

for i in (range(len(lowest_accuracy_real_data))):
    if int(lowest_accuracy_real_data[i])==0:
        zeros_low.append(i)
    else:
        ones_low.append(i)

for i in (range(len(highest_accuracy_real_data))):
    if int(highest_accuracy_real_data[i])==0:
        zeros_high.append(i)
    else:
        ones_high.append(i)

d_low = {'0': [lowest_accuracy_predicted_data[i] for i in zeros_low],
'1': [lowest_accuracy_predicted_data[i] for i in ones_low]}
d_high = {'0': [highest_accuracy_predicted_data[i] for i in
zeros_high], '1': [highest_accuracy_predicted_data[i] for i in
ones_high]}

df_low = pd.DataFrame(list(zip(lowest_accuracy_real_data,
lowest_accuracy_predicted_data)),
                columns =['Real Burst Suppression', 'Predicted Burst
Suppression Probability'])

df_high = pd.DataFrame(list(zip(highest_accuracy_real_data,
highest_accuracy_predicted_data)),
```

```
                columns =['Real Burst Suppression', 'Predicted Burst
Suppression Probability'])

entire_df_h = pd.DataFrame(list(zip(highest_accuracy_real_data,
highest_accuracy_predicted_data, ['Best
performance']*len(highest_accuracy_predicted_data))),
                           columns = ['Real Burst Suppression',
'Predicted Burst Suppression Probability', 'Performance'])
entire_df_l = pd.DataFrame(list(zip(lowest_accuracy_real_data,
lowest_accuracy_predicted_data, ['Worst
performance']*len(lowest_accuracy_predicted_data))),
                           columns = ['Real Burst Suppression',
'Predicted Burst Suppression Probability', 'Performance'])


entire_df = pd.concat([entire_df_h, entire_df_l])

##### LOWEST PERFORMANCE

plt.figure()
sns.violinplot(x='Real Burst Suppression', y='Predicted Burst
Suppression Probability', data=df_low,
                    inner=None, palette = 'YlOrBr')
sns.catplot(x='Real Burst Suppression', y='Predicted Burst Suppression
Probability',
            kind="violin", data=df_low, palette = 'YlOrBr')

plt.ylim(0, 1)
plt.title('Worst distribution of predicted values \n over real
values', fontsize = 16)
plt.tight_layout()
plt.savefig(os.path.join(directory, 'Other_Plots',
'SwarmPlots_lower.png'))
plt.close()

##### HIGHEST PERFORMANCE

plt.figure()
sns.violinplot(x='Real Burst Suppression', y='Predicted Burst
Suppression Probability', data=df_high,
                    inner=None, palette = 'YlOrBr')
sns.catplot(x='Real Burst Suppression', y='Predicted Burst Suppression
Probability',
            kind="violin", data=df_high, palette = 'YlOrBr')

plt.ylim(0, 1)
plt.title('Best distribution of predicted values \n over real values',
fontsize = 16)
plt.tight_layout()
plt.savefig(os.path.join(directory, 'Other_Plots',
'SwarmPlots_higher.png'))
plt.close()

##### PERFORMANCES TOGETHER

plt.figure()
```

```
sns.violinplot(x="Real Burst Suppression", y="Predicted Burst
Suppression Probability", hue="Performance",
    data=entire_df, split=False, inner="box", palette = 'YlOrBr')
plt.title('Distribution of predicted values over real values', pad =
30, fontsize = 16)
plt.xlabel("Real Burst Suppression", fontsize=12)
plt.ylabel("Predicted Burst Suppression Probability", fontsize=12)
plt.legend(loc='lower left', bbox_to_anchor=(0., 1.02, 1., .102),
          fancybox=True, shadow=True, ncol=2, title=False,
          mode="expand", borderaxespad=0.)
sns.despine()
plt.tight_layout()
plt.savefig(os.path.join(directory, 'Other_Plots',
'SwarmPlots_TOGETHER.png'))
plt.close()

# ========================================================================
# ACCURACIES DISTRIBUTION
# ========================================================================

x = [x for x in range(len(accuracies))]
accuracies.sort()

plt.figure()
plt.plot(x, accuracies, color = 'Maroon', alpha = 0.6)
plt.axhline(y=accuracies[x.index(round(np.percentile(x, 25), -1))],
color='Darkslategrey', linestyle='--', linewidth=0.9)
plt.fill_between(x, y1=0, y2=accuracies, where=(x <= np.percentile(x,
25)), color='Maroon', alpha=0.3)
plt.xlabel('Patients', fontsize = 12)
plt.ylabel('Accuracy', fontsize = 12)
plt.title('Accuracy distribution', fontsize = 16,  pad='10.0')
plt.tight_layout()
plt.gca().spines['top'].set_visible(False)
plt.gca().spines['right'].set_visible(False)
plt.xlim(0)
plt.ylim(0)
plt.savefig(os.path.join(directory, 'Other_Plots',
'Accuracy_distribution_patients.png'))
plt.close()

plt.figure()
sns.histplot(x=accuracies, kde=True, color = 'Maroon', alpha = 0.3,
edgecolor='White', linewidth=2)
plt.xlabel('Accuracy', fontsize = 12)
plt.ylabel('Frequency', fontsize = 12)
plt.title('Accuracy frequency', fontsize = 16, pad='10.0')
plt.axvline(x = accuracies[int(round(np.percentile(x, 25), 0))],
color='Darkslategrey', linestyle='--', linewidth=0.9)
plt.tight_layout()
plt.gca().spines['top'].set_visible(False)
plt.gca().spines['right'].set_visible(False)
plt.savefig(os.path.join(directory, 'Other_Plots',
'Accuracy_frequency_patients.png'))
plt.close()

plt.figure()
```

```
sns.set(style="ticks")
f, (ax_box, ax_hist) = plt.subplots(2, sharex=True,
                                    gridspec_kw={"height_ratios":
(.15, .85)})

sns.boxplot(accuracies, ax=ax_box, color = 'Maroon')
sns.distplot(accuracies, ax=ax_hist, color = 'DarkOrange', bins = 7)

ax_box.set(yticks=[])
sns.despine(ax=ax_hist)
sns.despine(ax=ax_box, left=True)

plt.xlabel('Accuracy', fontsize = 12)
plt.ylabel('Density', fontsize = 12)
plt.suptitle("Accuracy frequency", fontsize=16)
plt.axvline(x = accuracies[int(round(np.percentile(x, 25), 0))],
color='Darkslategrey', linestyle='--', linewidth=0.9)
plt.tight_layout()
plt.gca().spines['top'].set_visible(False)
plt.gca().spines['right'].set_visible(False)
plt.savefig(os.path.join(directory, 'Other_Plots',
'accuracy_histogram_boxplot.png'))
plt.close()
```