

Development and validation of a Machine Learning-Based Pharmacodynamic Model Of Rocuronium induced neuromuscular block during general anesthesia

Using the Leiden Data logger



10-2023

TM3 MSc Thesis
Anesthesiology
Leiden University
Medical Center

Leiden University
Delft University of
Technology
Erasmus University
Rotterdam

This page was intentionally left blank

DEVELOPMENT AND VALIDATION OF A MACHINE LEARNING-BASED PHARMACODYNAMIC MODEL OF ROCURONIUM INDUCED NEUROMUSCULAR BLOCK DURING GENERAL ANESTHESIA

- Using the Leiden Data logger -

L.C. (Luuk) Bolhuis

Student number : 4392116

October 2023

Thesis in partial fulfilment of the requirements for the joint degree of Master of Science in

Technical Medicine

Leiden University ; Delft University of Technology ; Erasmus University Rotterdam

Master thesis project (TM30004 ; 35 ECTS)

Department of Anesthesiology, LUMC, Leiden

January 2023 – October 2023

Supervisor(s):

Dr. M. (Martijn) Boon MD, LUMC

Dr. J.H.G. (Justin) Dauwels, TU Delft

Drs. M. (Michel) Abdel Malek MD, LUMC

Thesis committee members:

Prof. dr. A. (Albert) Dahan MD, LUMC

Dr. M. (Martijn) Boon MD, LUMC

Dr. J.H.G. (Justin) Dauwels, LUMC

Drs. M. (Michel) Abdel Malek MD, LUMC

An electronic version of this thesis is available at <http://repository.tudelft.nl/>.

Assessment committee

Independent Senior Staff Member & Chair	Prof. dr. A. (Albert) Dahan MD Professor of Anesthesiology, anesthesiologist Department of Anesthesiology Leiden University Medical Center, Leiden, The Netherlands
Medical Supervisor	Dr. M. (Martijn) Boon MD Anesthesiologist Department of Anesthesiology Leiden University Medical Center, Leiden, The Netherlands
Technical Supervisor	Dr. J.H.G. (Justin) Dauwels Assistant professor Signal Processing Systems, Department of Microelectronics, Faculty of Electrical Engineering, Mathematics and Computer Science Delft University of Technology, Delft, The Netherlands
Daily Supervisor	Drs. M. (Michel) Abdel Malek MD Anesthesiology resident, PhD candidate Department of Anesthesiology Leiden University Medical Center, Leiden, The Netherlands

Abstract

Neuromuscular blocking agents (NMBAs) are commonly employed in anesthesia to facilitate intubation and improve surgical working conditions. Understanding their pharmacokinetics (PK) and pharmacodynamics (PD) is crucial for optimizing their administration. PK describes drug absorption, distribution, metabolism, and elimination, while PD links drug concentration to its pharmacological effects. Integrating PK and PD information through PKPD modeling offers insights into the relationship between drug concentration and effect. Recent advancements in machine learning (ML) have shown promise in modeling the PD of anesthetics, offering potential benefits over traditional PKPD models. This study aimed to develop an automated data logger for recording neuromuscular transmission (NMT) measurements and rocuronium infusion data during surgery. A secondary goal was to predict TOF ratios using both traditional PKPD models and newer ML techniques.

The experimental setup involved development of data logger software. Data preprocessing consolidated data, removed outliers, and applied interpolation for missing values. Machine learning models, including linear regression, decision trees, and extreme gradient boosting, were trained and evaluated using double Leave-One-Group-Out cross-validation. Additionally, a traditional PKPD model estimated pharmacokinetic parameters based on patient characteristics and rocuronium administration data. Model performance was assessed using metrics such as Root Mean Squared Error (RMSE), Normalized RMSE (NRMSE), R-squared (R^2), and the Pearson correlation coefficient.

From March 23 to June 20, 2023, a prospective observational study at Leiden University Medical Center (LUMC) included 42 patients in the operating room. Data were collected from three distinct monitors using data logger software. The collected data was divided into TOF-Cuff and GE NMT monitor subsets, excluding continuous rocuronium infusion records.

In terms of model performance, machine learning models displayed suboptimal results when applied to GE NMT monitor data, indicated by high RMSE and low R^2 values. In contrast, basic and optimized PKPD models exhibited better predictive capabilities. Similar trends were observed in the performance evaluation of TOF-Cuff data, with machine learning models less effective compared to PKPD models. An in-depth analysis of NRMSE revealed outliers, mainly in the optimized PKPD model. Cumulative distribution plots highlighted performance variations across subjects, particularly in the TOF-Cuff results.

The dataset comprised 42 subjects undergoing surgical procedures, but the effective sample size for analysis was limited due to deep neuromuscular blockade, sensor placement issues, and data loss. Traditional pharmacokinetic-pharmacodynamic (PKPD) models, based on data from 423 patients, outperformed machine learning models in predicting Train of Four (TOF) ratios, as the latter faced overfitting challenges with a smaller dataset. Future directions suggest collecting more extensive data (ideally closer to 100 subjects) to improve machine learning model performance and possibly include features like time until full neuromuscular blockade recovery. Additionally, identifying the most critical features behind machine learning predictions can help streamline computational methods. Overall, refining the software, increasing data, and feature analysis can enhance machine learning-based neuromuscular blockade prediction.

In conclusion, this study introduced a novel data logger software for recording neuromuscular blockade data during surgery at LUMC. While machine learning approaches fell short in approximating TOF ratios, future research with expanded datasets and more comprehensive feature analysis holds promise for the development of more robust machine learning models.

Table of Contents

Assessment committee.....	iii
Abstract.....	iv
Nomenclature.....	2
1. Introduction.....	3
2. Materials and Methods.....	5
2.1 Experimental Setup.....	5
2.1.1 Physical Connections between Devices.....	5
2.1.2 Development of the Data logger Software.....	6
2.2 Study Population.....	7
2.3 Study Design.....	7
2.4 Data Extraction.....	8
2.5 Data Pre-processing.....	8
2.6 Model Development.....	9
2.6.1 Machine Learning Techniques.....	9
2.6.2 Traditional PKPD Modelling.....	10
2.7 Model Evaluation.....	13
3. Results.....	14
3.1 Study population.....	14
3.2 Dataset.....	14
3.3 Data Exploration.....	15
3.4 Data Pre-processing.....	16
3.5 Model Evaluation.....	16
4. Discussion.....	22
References.....	25
Supplement A. Pinout configurations of converters.....	26
Supplement B. Data logger OR – user manual.....	27
Supplement C. TOF Ratios per subject.....	29

Nomenclature

Abbreviation	Definition
ACh	Acetylcholine
AMG	Acceleromyography
AUC	Area Under the Curve
BIS	Bispectral Index
BMI	Body Mass Index
CRC	Cyclic Redundancy Check
DBS	Double Burst Stimulation
eGFR	Estimated Glomerular Filtration Rate
EHR	Electronic Health Record
EMG	Electromyography
LOGO	Leave One Group Out
LUMC	Leiden University Medical Center
ML	Machine Learning
nAChRs	Nicotinic Acetylcholine Receptors
NaN	Not a Number
NMB	Neuromuscular Block
NMBA	Neuromuscular Blocking Agent
NMJ	Neuromuscular Junction
NMT	Neuromuscular Transmission
NRMSE	Normalized Root Mean Squared Error
OR	Operating Room
PACU	Post Anaesthetic Care Unit
PDMS	Patient Data Management System
PD	Pharmacodynamic
PK	Pharmacokinetic
PKPD	Pharmacokinetic-Pharmacodynamic
PTC	Post Tetanic Count
R ²	Coefficient of Determination
rNMB	Residual Neuromuscular Block
RMSE	Root Mean Squared Error
ROC	Receiver Operating Characteristic
RS-232	Recommended Standard 232
SD	Standard Deviation
SOL	Serial over LAN
SOP	Standard Operating Procedure
TOF	Train of Four
TOFR	Train of Four Ratio
UI	User Interface

1. Introduction

Pharmacokinetics and pharmacodynamics in neuromuscular blockade

Neuromuscular blocking agents (NMBAs) are routinely applied during general anesthesia to facilitate endotracheal intubation and improve surgical working conditions. One common used NMBA is rocuronium due to its rapid onset of action [2]. However, because of their long half-lives, residual neuromuscular block (rNMB) persists in the post-anesthetic care unit (PACU), which is associated with postoperative adverse events [3]. Understanding the mechanism of action of rocuronium and other drugs, as well as assessing their clinical effects, plays a crucial role in the safe administration of rocuronium. Pharmacokinetics (PK) describe how a drug is absorbed, distributed, metabolized, and eliminated by the body, while pharmacodynamics (PD) focuses on the relationship between drug concentration and its pharmacological effects. In the case of rocuronium, its PK properties determine the speed of onset, duration, and offset of action. After administration, rocuronium spreads throughout the body and eventually reaches its site of action, the neuromuscular junction (NMJ), where it binds to nicotinic acetylcholine receptors (nAChRs). This binding prevents the endplate potential from reaching the threshold required to initiate a propagating action potential, resulting in muscle paralysis [4].

The concentration of rocuronium in plasma over time is influenced by factors such as absorption, distribution to various tissues, metabolism, and elimination. These PK parameters and PD interactions can be used to create pharmacokinetic-pharmacodynamic (PKPD) models, enabling us to understand and predict the drug's effects. These models often incorporate PK data (concentration-time profiles) and PD data (concentration-effect profiles) to quantify the relationship between drug concentration and time (see Figure 1-1). For rocuronium, the current PKPD model is developed by Kleijn, et al. This model was developed using neuromuscular transmission (NMT) monitoring evaluated with acceleromyography at the thumb [5].

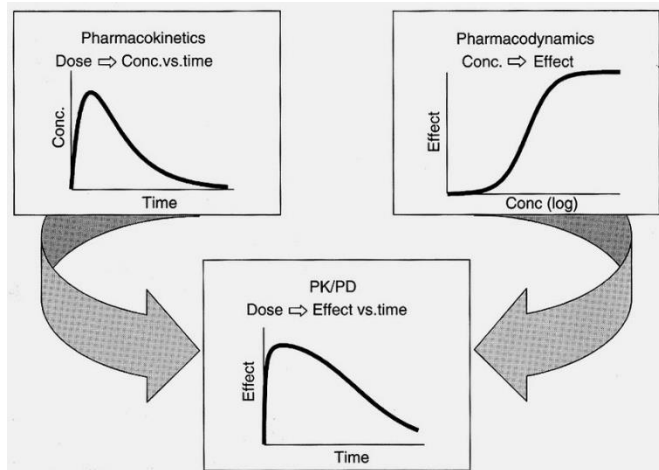


Figure 1-1 Basic concept of PKPD modelling. Time versus concentration measurements (PK) and effects versus concentration relationship (PD) results in an effect versus time model (PKPD) [1].

Monitoring depth of neuromuscular block

NMT monitoring can be performed in multiple ways. The assessment of neuromuscular transmission is crucial to ensure adequate blockade, monitor the depth of the NMB, and facilitate safe recovery. Various monitoring techniques and devices have been developed to assess neuromuscular function. The most common method of NMB monitoring is the train of four (TOF) method [6]. In the Leiden University Medical Center (LUMC) three commonly TOF measuring techniques are available. TOF-Cuff, GE Healthcare NMT EMG (Neuromuscular Transmission; electromyography) module and the Philips IntelliVue NMT AMG (acceleromyography) module are widely used in clinical practice.

The TOF-Cuff is a novel monitor specifically designed for NMB monitoring. In contrast to acceleromyography and electromyography, it utilizes a pressure cuff placed around a patient's upper arm or upper leg and applies a series of electrical stimuli, called twitches, to the nerves running in the upper arm biceps groove. These nerve stimuli cause a response in the upper arm muscles, which the cuff detects. Based on the change in the cuff's internal pressure, the TOF ratio (TOFR) is calculated by comparing the amplitude of the fourth twitch (T4) to the first twitch (T1) [7]. This ratio indicates the

level of neuromuscular blockade, with a higher ratio indicating a reduced degree of NMB and a lower ratio signifying a deeper level of blockade.

The second monitor used in NMB depth monitoring is the NMT module from GE Healthcare. This quantitative device uses electromyography (EMG) measurements. This is done by placing five electrodes on the distal arm; two of the electrodes stimulate the ulnar nerve, one electrode functions as ground and two electrodes record the compound muscle-contraction action potentials. Depending on the level of NMB, different parameters can be used to quantify the NMB depth. As rocuronium is a non-depolarizing blocking agent, fade can be measured using the TOF ratio. When the number of twitches is below 1, TOF measurements are not helpful anymore. In that case, post-tetanic count (PTC) can be used to measure deep NMB. After five seconds of stimulation at 50Hz, the ulnar nerve is stimulated and post tetanic twitches are counted.

Another way of NMB monitoring is via acceleromyography (AMG), which is done with the Philips IntelliVue NMT module. This technique involves the measurement of mechanical responses of muscles to electrical stimulation. Again, two sensors are placed at the wrist above the ulnar nerve and an acceleration sensor placed at the end of the thumb detects the movement (acceleration) of the thumb after nerve stimulation.

With the use of data recorded by these different devices, differences in sensitivity between muscle groups can be measured. Central muscles have a different sensitivity for NMBAs than peripheral muscles. This means that the onset and offset times of NMB differ between these muscle groups.

Machine Learning approaches

In recent years, machine learning (ML) techniques have been used in PD modelling of other anaesthetics. The results produced by the ML model were more accurate than the results generated by the older mathematic PD model [8, 9]. Lee's study utilized propofol and remifentanyl infusion data along with BIS values to model and predict the bispectral index (BIS). In contrast to classical pharmacokinetic models that independently model the pharmacokinetics of the drugs and combine them in a response surface model, the neural network consisted of a Long-Term Short Term (LSTM) network dealing with infusion history, followed by a feed-forward neural network. The deep learning model demonstrated superiority over traditional PK/PD models when it came to predicting BIS, indicating the potential superiority of AI-based approaches over traditional pharmacokinetic models. Ingrande's study used an artificial neural network (ANN) to describe propofol pharmacokinetics and compared its performance with a 4-compartment model and a recirculatory model. An LSTM and gated recurrent unit (GRU) were used in the ANN. The ANN model outperformed the 4-compartment and showed similar results with the recirculatory model in predicting the pharmacokinetics of propofol.

While these studies focussed on propofol administration, Wang's study used multiple deep-learning methods to predict NMB duration and the recovery profile of cisatracurium [10]. A recurrent neural network (RNN) gated recurrent unit (GRU) and long short-term memory (LSTM) model were built. All of these models are suitable for temporal sequence predictions because they all possess a recurrent structure to memorize a previous part of the sequence. The first half of a TOFR sequence was used to train the deep learning models and the predicted output, which was the second half of the sequence, was evaluated using the RSME to assess every model's performance. The GRU model outperformed both LSTM and RNN. With the use of transfer learning to identify similar patients, the RMSE was even lower.

All of these studies made use of large datasets with high-frequency numeric data to train their machine- and dee- learning models. This thesis hypothesizes that with the use of the numeric TOFR values measured with NMT monitoring devices during general anesthesia, new ML based PD models can be trained that might outperform existing traditional PKPD models in predicting TOFR for rocuronium. Therefore, the primary aim of this thesis is to develop an automated data logger that captures TOFR measurements from the TOF-Cuff and GE NMT monitor and the rocuronium infusion data during surgical procedures. The secondary aim is to predict the TOF ratio based on the recorded data using both the traditional PKPD model and newer machine learning techniques.

2. Materials and Methods

This chapter is divided into three parts. The first part discusses the experimental setup for the study, providing a detailed description of the physical connections between devices and an overview of the data logger software. The second part of this chapter outlines the study design and the target population. In the third part, we delve into the model development and data pre-processing steps.

2.1 Experimental Setup

The experimental setup in the operating room consisted of four different devices. For NMT monitoring, two NMT monitors were employed; the TOF-Cuff NMT Monitor, RGB Medical Devices, SA, Madrid, Spain (hereafter referred to as TOF-Cuff) and the CARESCAPE B450 Patient Monitor, GE Healthcare, Chicago, IL, USA (hereafter referred to as GE NMT monitor). To capture data from the syringe pump, an UniQueDOC multi-channel infusion station, Arcomed AG Medical Systems, Zurich, Kloten, Switzerland (hereafter referred to as Arcomed docking station) was utilized. These three monitors were connected to a Lantronix EDS-MD 8-Port Server, Lantronix, Irvin, CA, USA (hereafter referred to as Lantronix). Figure 2-1 shows the experimental setup in the operating room.



Figure 2-1 Experimental setup in the operating room, the Arcomed docking station with an inserted syringe pump is positioned on the lower left. On top of it, a TOF-Cuff is mounted to the pole. In the top middle, two Lantronix devices are placed, and on the right, a GE NMT monitor is positioned on top of a ventilator machine.

2.1.1 Physical Connections between Devices

To establish communication between the monitoring devices and the Lantronix, a physical connection was required. In the medical domain, serial communication is a commonly used communication interface. This method involves using two transmission lines to send and receive data one bit at a time and typically utilizes a serial port (RS-232). The communication protocols provided by the manufacturers were consulted to determine the type of RS-232 serial port used by each monitor. It's important to note that not all monitors use the same type of RS-232 port. Specifically, the TOF-Cuff has a DB9 male port, while the GE NMT monitor and the Arcomed docking station both use a USB

port. An overview of these different ports is presented in Figure 2-2. Figure 2-2: RS-232 ports on the GE NMT monitor (left), the Arcomed docking station (middle) and the TOF-Cuff (right). To connect each monitoring device to the Lantronix serial RJ45 ports, converters were employed. Subsequently, the configuration of the RS232 output pins for data transfer was identified within the communication protocols. This involved determining which pins were used for specific types of data transfer. A schematic representation of the pinouts is included as a Supplementary Figure in Supplement A.

To facilitate the transformation of the serial connection through the RS232 protocol into a network signal, serial-to-ethernet converters were utilized. Within the Leiden University Medical Center (LUMC), a Lantronix was employed to connect these types of devices to the LUMC intranet network. The Lantronix employed the Serial over LAN (SOL) technique, allowing for the remote redirection of input and output from the serial port over IP. This capability enabled remote access to the RS232 communication devices. The Ethernet serial server played a crucial role in assigning TCP ports and IP addresses, thereby enabling seamless communication between users and the interconnected devices.



Figure 2-2: RS-232 ports on the GE NMT monitor (left), the Arcomed docking station (middle) and the TOF-Cuff (right).

2.1.2 Development of the Data logger Software

Once the physical connections between all devices were established, data exchange became possible. While some devices can passively transmit data over the transmit pin (Tx), most devices require an initial handshake with another device to initiate data transmission through their RS232 port. Typically, a data receive request arrives on the receive pin (Rx) of the device. To manage this communication traffic effectively, an algorithm was developed using Python [11]. Three scripts were created for each device: a socket script (for communication with the serial devices), a connection protocol (acting as the data handling and control center of the algorithm), and a save protocol (for saving data into a data file). With the use of the Python module Threading, threads were incorporated so that each device had its separate execution path in case one device encountered an error.

Socket connection

This script comprises three classes, one for each device. Within these classes, connections with the corresponding monitoring devices can be established and terminated. Once the specified port is opened, and a handshake is successfully performed, data can be sent and received via the socket. As serial communication requires a Cyclic Redundancy Check (CRC) to detect errors or corruption in transmitted data, a CRC checksum is computed on both incoming and outgoing messages. Each message consists of a start and a stop byte, allowing for message detection. A failback mechanism for termination to stop communication and close the communication port is included in case the script crashes or is interrupted.

Connection protocol

In this second script, data handling is managed. With the assistance of the socket script, a Send-Receive request is sent to the monitor, and after receiving a confirmation byte string, the connection with the monitoring devices is established. Subsequently, a continuous data stream is processed in this script. Each byte string message is decrypted, and the connection protocol determines the appropriate action for the incoming data based on its message type. For each data type, a separate procedure can be defined. Once the data is decrypted and interpreted as numeric data, it is forwarded and stored in the save protocol.

Save protocol

In this final script, an HDF5 data file is constructed, complete with a file name containing a sequence number and a timestamp. The data file also contains a large data frame filled with NaN values. The timestamp is set as soon as the program is started and data collection commences. This timestamp serves as the reference throughout the entire data collection period for all monitors. Within the save protocol, a data frame is generated that includes all desired output variables from different monitors. When a new measurement is received from a monitor, a new receipt timestamp is set and compared to the reference timestamp. The time difference in seconds between the reference timestamp and the receipt timestamp is calculated and used as an index to store the variable values for that specific monitor.

User interface

To manage and activate these algorithms, a user interface (UI) was implemented. Django was chosen as the Python web framework, as it allows for managing both a front-end and a back-end to run software and perform data analysis. A detailed description of the user interface has been previously documented [BRON]. Additionally, a user manual has been created to guide users on how to activate the data logger and initiate data collection (see Supplement B).

2.2 Study Population

All patients aged between 18 and 80 years old who were undergoing elective general non-cardiac surgery requiring general anesthesia with the use of rocuronium as a muscle relaxant were eligible for the study. Eligible patients had an estimated glomerular filtration rate (eGFR) >60 ml/min/1.73m² and were classified under the ASA physical status classification system as grade I to III. Patients did not qualify for participation if they met any of the following specific criteria: patients with neuromuscular diseases, patients taking medications known to significantly affect rocuronium's behavior (such as magnesium or lithium, aminoglycosides, tetracyclines, polymyxins, or clindamycin), or patients with documented muscle relaxant allergies. These measures were put in place to ensure both the integrity of the study and the safety of potential participants.

2.3 Study Design

In this prospective observational study, patients were screened for eligibility between March 21, 2023, and June 1, 2023, at the LUMC. The study received approval from the non-WMO committee. Since few studies have been published on PKPD modelling using artificial intelligence, a sample size has been deduced from comparative studies. Lee's study had about 2 million data points extracted from approximately 230 cases [8]. Wang's study, which modelled cis-atracurium to TOFR, included 83 patients [10]. With a maximum of 228 data points per patient, the total amount of data points would be 18,924. Ingrande's study extracted 1128 data points from 24 different subjects [9]. Although in this study the model overestimated the outcome considerably during induction. It may be hypothesized that this overestimation is explained due to the lack of data points. Because this study will be similar to

Wang's study in terms of the study subject, but considerably less complex than Lee's study, the assumption is made that a total of approximately 10.000 data points would be sufficient to train the models. It is estimated that each patient will generate approximately 360 data points. Based on the ratio between training set, validation set and test set (70%-15%-15%), the amount totals out to 44 patients. After the screening, patients provided verbal consent for their medical data to be recorded and stored in the LUMC database. Once consent was obtained, both monitors were connected to the patient. Depending on the placement of the blood pressure cuff, the TOF-Cuff was positioned on the opposite arm, while the GE NMT monitor electrodes were attached to the distal forearm of the arm used for blood pressure measurement. After ensuring that all devices were correctly connected to the research Lantronix, data collection commenced. Standard anesthetic procedures were followed during induction. After administering propofol and sufentanil, and ensuring that the patient was securely sedated with a target bispectral index (BIS) between 40 and 60, both NMB monitors were calibrated. Following calibration, 0.6 mg/kg of rocuronium was administered via the syringe pump. As a backup measure, the time of rocuronium bolus administration was recorded in a study file and as per standard procedure in the patient's file system. The syringe pump was positioned in the lower slot of the docking station, following the standard operating procedure (SOP) of the Data logger (see Supplement B). After the administration of rocuronium and achieving a deep NMB, an endotracheal tube was safely inserted into the patient, and the surgical procedure commenced. Throughout the surgery, it was the responsibility of the anesthesiologist to administer additional rocuronium, as long as it was administered via the syringe pump to collect and store the data. Depending on the TOF ratio at the end of the surgery, a decision was made to either wait for spontaneous recovery or to administer sugammadex to reverse the NMB. If sugammadex was administered, the time of administration and the amount were recorded in the study file. After both NMT monitoring devices displayed a TOF Ratio value of 100, they were turned off and removed from the patient. Finally, the data logger software was stopped.

2.4 Data Extraction

Patient data was retrospectively extracted from the electronic health record (EHR) system. Patient characteristics, including age, weight, height, and sex at the time of the surgical procedure, were collected. Preoperative ASA scores and the most recent preoperative eGFR measurements were also obtained. In cases where multiple measurements were available, the last measurement taken before the surgery was selected. Body Mass Index (BMI) was calculated using the following formula:

$$BMI = \frac{weight (kg)}{height (m)^2} \quad Eq. 1$$

2.5 Data Pre-processing

The data pre-processing was carried out using Python version 3.9 [BRON]. Initially, all individual subject files were consolidated into one comprehensive data frame. Patient characteristics were appended to the data frame for each respective subject. Bolus infusion times were extracted, and the dataset was trimmed to begin 20 seconds before the first rocuronium bolus was administered. This ensured that the calibration steps were included in the data.

Since both TOF-Cuff measurements and GE NMT monitor measurements were stored in the same data frame, two separate data frames were created for each of the monitors. This separation prevented any cross-contamination of information related to the TOF ratio trend in the model. Outliers in the TOF ratio were identified and removed. If applicable, information about sugammadex administration was added for each subject. To smoothen the TOF ratio curves, a smoothing filter was applied, but the window size was limited to 20 to avoid eliminating critical information during the induction phase of the TOF ratio curve.

Given the dataset's structure, numerous Not-a-Number (NaN) values existed between received measurements. To address this, linear interpolation was performed for both the TOF ratios and syringe pump measurements. Interpolation was confined to occur between consecutive measurements. In cases where TOF measurements were inadequately captured by one of the two monitoring devices throughout the surgery, the subject was excluded from the analysis.

2.6 Model Development

To approximate the TOF Ratio curve during surgery based on patient characteristics and rocuronium administration data, both traditional and multiple machine learning regression models were employed. These models were all trained using the same train-test split that was created during the pre-processing phase. Three distinct machine learning techniques were utilized to construct the models:

- Traditional Linear Regression
- Decision Tree Regressor
- Gradient Boosting Regressor

Two Python packages were employed for this purpose: scikit-learn for the first two machine learning techniques and XGBoost for the gradient boosting regressor. All models were trained and fitted using the same training dataset. For hyperparameter tuning of the machine learning models, a grid search technique was employed to identify the optimal hyperparameters, employing an eight-fold cross-validation approach. The combination of hyperparameters that exhibited the best performance was selected for fitting the model to the training dataset. An overview of the hyperparameters tested is presented in Table 2-1.

Table 2-1 Hyperparameters for machine learning models, including Python package/module

Technique	Python package	Python module	Hyperparameter	Range
Decision Tree Regressor	sklearn.tree	DecisionTreeRegressor	"min_samples_split"	[2, 5, 10]
			"min_samples_leaf"	[1, 2, 4, 7]
			"max_leaf_nodes"	[10, 20, 40, 80]
			"max_depth"	[5, 10, 15]
			"max_features"	'log2', None
			"splitter"	'best', 'random'
XGBoost Regressor	xgboost	XGBRegressor	"n_estimators"	[50, 100, 200]
			"learning_rate"	[0.01, 0.1, 0.2]
			"max_depth"	[3, 5, 7]
			"subsample"	[0.8, 0.9, 1.0]
			"colsample_bytree"	[0.8, 0.9, 1.0]
			"min_child_weight"	[1, 5, 10]

2.6.1 Machine Learning Techniques

When developing machine learning models, it is important to develop an accurate model. But only an accurate model is not enough. To ensure its robustness and generalizability across diverse subjects, a comprehensive methodology known as Double Leave-One-Group-Out (LOGO) cross-validation was employed. This double LOGO approach is unique in that it combines two levels of cross-validation to thoroughly assess the performance of machine learning models. It is particularly valuable when the objective is to create models capable of consistent predictions across multiple subjects or groups.

The method used two levels of validation. In the first level of cross-validation, the LOGO method was applied, leaving out one entire subject at a time for testing purposes. The remaining subjects formed the training dataset. Each subject was allowed to serve as the test dataset at least once, ensuring that evaluations were unbiased. For each iteration of the outer loop, a machine learning model was trained on the training dataset within an inner loop. In this inner loop, all of the remaining subjects are held out

once for internal cross-validation using the LOGO technique. The RMSE is calculated for each internal cross-validation fold to assess the model's performance on the validation subject's data. After completing the inner loop, the RMSE values from the internal cross-validation are combined and the best model is selected. This internal cross-validation provides insights into the model's stability and generalizability. Once all subjects had been omitted once, the final model configuration was employed to predict TOF Ratios on unseen test data from the outer loop.

Traditional linear regression

To address this regression problem, a linear regression function was applied to the training data to estimate the TOFR. The scikit-learn package's Linear Regression module was utilized for this purpose. Coefficients were determined by minimizing the residual sum of squares between the observed targets in the dataset and the targets predicted by the linear approximation.

Decision tree regressor

The decision tree regressor is a machine learning algorithm used for regression tasks. It constructs a tree-like structure where each internal node represents a decision based on a feature, and each leaf node provides a prediction for the target variable. If a decision tree has multiple deep layers, it is more susceptible to overfitting. These models work by recursively dividing the data into subsets based on feature values, with each split optimized to minimize prediction errors.

Gradient boosting

The XGBoost regressor is a potent machine learning algorithm that employs gradient boosting with decision trees as base models for regression tasks. It iteratively corrects errors from previous models by adding new trees and incorporates regularization techniques to prevent overfitting.

2.6.2 Traditional PKPD Modelling

To compare newer machine learning techniques with a traditional PKPD (Pharmacokinetic-Pharmacodynamic) model, the collected data was used to implement a classic PKPD model developed by Kleijn et al. [BRON]. Kleijn and colleagues employed a nonlinear mixed-effects modelling approach to construct a compartment model for rocuronium and sugammadex. For the estimation of the TOF (Train-of-Four) ratio, a simplified version of this model, focusing solely on the rocuronium compartment, was utilized and is depicted in Figure 2-3. Using patient weight, age, and eGFR as input characteristics, compartment parameters related to rocuronium PK (Pharmacokinetics) could be determined. These calculated rocuronium PK parameters (ke_0 , E_0 , EC_{50} and γ) were then incorporated into the model, which is elaborated upon in the following section.

PKPD model development

The following equations were incorporated to calculate the diffusion rate constants k_{12} and k_{21} between the two compartments, and the elimination rate constant k_{10} , as illustrated in Figure 2-3.

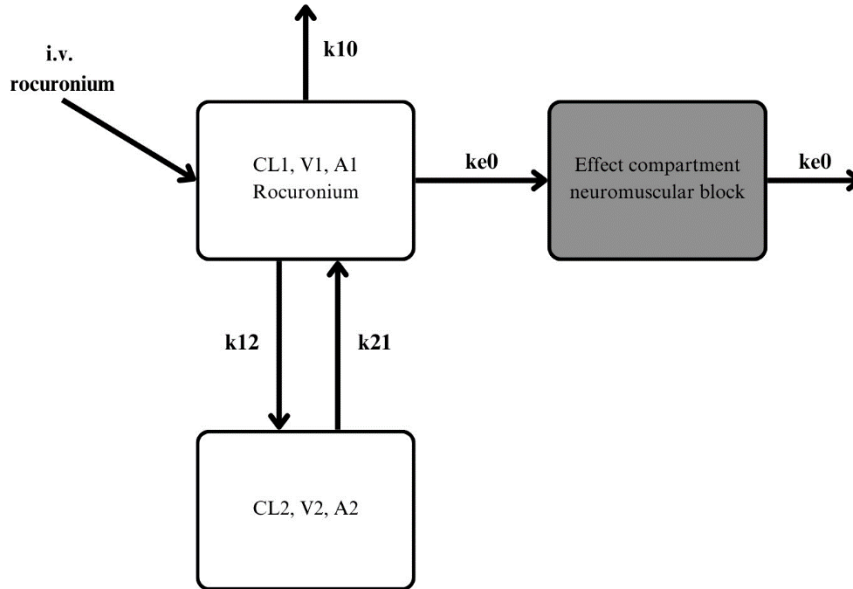


Figure 2-3 Simplified rocuronium compartment model of Kleijn. Abbreviations: A_1 , amount of rocuronium in central compartment; A_2 , amount of rocuronium in the peripheral compartment; CL_1 , rocuronium clearance; i.v., intravenous; k_{10} , rate constant of rocuronium elimination; k_{12} , diffusion rate constant of rocuronium from the central to the peripheral compartment; k_{21} , diffusion rate constant of rocuronium from the peripheral to the central compartment; ke_0 , distribution rate constant between central and effect compartments; V_1 , volume of distribution of rocuronium in the central compartment; V_2 , volume of distribution of rocuronium in the peripheral compartment

$$k_{10} = \frac{CL_1}{V_1} \quad \text{Eq. 2}$$

$$k_{12} = \frac{CL_2}{V_1} \quad \text{Eq. 3}$$

$$k_{21} = \frac{CL_2}{V_2} \quad \text{Eq. 4}$$

where CL_1 and V_1 are the clearance and volume of the central compartment, respectively; and CL_2 and V_2 are the clearance and volume of the peripheral compartment, respectively. CL_1 , CL_2 , V_1 and V_2 were estimated based on the PK parameter calculations provided in the article of Kleijn, et al.

The change in rocuronium amount in each compartment ($\frac{dA_1}{dt}$, $\frac{dA_2}{dt}$) can be calculated using the diffusion rate constants (k_{10} , k_{12} , k_{21}) of Eq. 2, Eq. 3, Eq. 4:

$$\frac{dA_1}{dt} = k_{21} * A_2 - (k_{10} + k_{12}) * A_1 \quad \text{Eq. 5}$$

$$\frac{dA_2}{dt} = k_{12} * A_1 - (k_{21}) * A_2 \quad \text{Eq. 6}$$

where A_1 and A_2 are the existing amount of rocuronium in the central and peripheral compartments, respectively. Applying Euler's rule to Eq. 5 and Eq. 6, the total amount of rocuronium in a compartment at a certain point in time can be calculated as follows:

$$A_1 = A_1 + (k_{21} * A_2 - (k_{10} + k_{12}) * A_1) * \delta t \quad \text{Eq. 7}$$

$$A_2 = A_2 + (k_{12} * A_1 - (k_{21}) * A_2) * \delta t \quad \text{Eq. 8}$$

Extracting the amount of rocuronium in the peripheral compartment from the central compartment and multiplying that number with the rocuronium diffusion rate constant ke_0 , given in Eq. 9:

$$\frac{dA_3}{dt} = (A_1 - A_2) * ke_0 \quad \text{Eq. 9}$$

Where $\frac{dA_3}{dt}$ is the change in rocuronium amount in the effect compartment. Applying Euler's rule again on Eq. 9, the total amount of rocuronium in the effect compartment becomes:

$$A_3 = A_3 + ((A_1 - A_2) * ke_0) * \delta t \quad \text{Eq. 10}$$

To calculate the effect site concentration (C_{eff}), A_3 , calculated in Eq. 10, can be divided by V_1 :

$$C_{eff} = \frac{A_3}{V_1} \quad \text{Eq. 11}$$

Eventually, by applying the effect site concentration of rocuronium (C_{eff}) from Eq. 11 into Eq. 12, the TOFR ratio (TOFR) can be calculated:

$$TOFR = \frac{E_0}{1 + \left(\frac{C_{eff}}{EC_{50}}\right)^\gamma} \quad \text{Eq. 12}$$

Where E_0 , EC_{50} and γ are the rocuronium PK parameters derived based on weight, age and eGFR.

Model optimization and model performance

The model was constructed using RStudio, version 4.3.1 [BRON]. Subsequently, the predicted TOFR (Train-of-Four Ratio) of the base model was compared with the actual TOFR values from the test dataset. To generate a predicted TOFR, rocuronium administration records were included in the model, along with weight, age, and eGFR as input variables. The model's predictive performance was evaluated using the root mean squared error (RMSE), normalized root mean square error (NRMSE) and the Pearson correlation coefficient (r).

An attempt was also made to optimize the model. In this optimization process, a unique adjusted set of coefficients was computed for each patient from the training set based on their TOFR data. This was accomplished using Levenberg-Marquardt optimization, which is an iterative method aiming to estimate the parameters that best fit a nonlinear model by minimizing the least squares of the residuals. After optimization, the median coefficient values were extracted and utilized to fit the optimized model to the test dataset.

2.7 Model Evaluation

After the regression models were created and fitted based on the training data, the resulting models were run on the test dataset to determine the model performances.

The performance measures for the regression models were the coefficient of determination (R^2), the root mean squared error (RMSE) and the normalized root mean squared error (NRMSE). The R^2 represents the proportion of the variation in the outcome that is explained or predicted by the model. It quantifies how well the model fits the data, with values closer to 1 indicating a better fit. The RMSE is a measure of the accuracy of a predictive model in regression problems. It quantifies how closely the predicted values from the model align with the actual values in the dataset. Lower RMSE values indicate better predictive accuracy. The NRMSE expresses a percentage of the range of the data, making it a normalized metric that's independent of the scale of the data. Table 2-2 shows the metrics and their calculation, where N is the number of samples, y is the tested variable, \hat{y} is the estimated value, \bar{y} is the mean of the measured variable, m_y is the mean of tested vector y and $m_{\hat{y}}$ is the mean of the estimated vector \hat{y} . At last, the Pearson correlation coefficient is calculated to evaluate the models. This metric measures the strength and direction of a linear relationship between two continuous variables, the estimated variable \hat{y} and the tested variable y . It quantifies how well the two variables are correlated.

Table 2-2 Evaluation metrics

Performance measure	Formula
Coefficient of determination (R^2)	$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$
Root mean squared error (RMSE)	$RMSE = \sqrt{\frac{\sum (\hat{y}_i - y)^2}{N}}$
Normalized root mean squared error (NRMSE)	$NRMSE = \frac{RMSE}{(y_{max} - y_{min})}$
Pearson correlation coefficient (r)	$r = \frac{\sum (y - m_y)(\hat{y} - m_{\hat{y}})}{\sqrt{\sum (y - m_y)^2 \sum (\hat{y} - m_{\hat{y}})^2}}$

3. Results

In this chapter, we will present the results of the implementation. The chapter will commence with an overview of the dataset utilized for model development. Following that, we will delve into data exploration and pre-processing steps, and subsequently, we will discuss the evaluation of all the models.

3.1 Study population

From March 23 to June 20, patients were eligible for inclusion in this prospective observational study conducted in the operating room (OR) of the LUMC. A total of 42 patients were enrolled in this study. The average age of the patients was 59 years (with a standard deviation (SD) of ± 15.1), with an even distribution of male and female patients (50% each). The mean estimated glomerular filtration rate (eGFR) was 84 (with an SD of ± 9.8). An overview of the characteristics of the study population is provided in Table 3-1.

Table 3-1 Study population characteristics of the total data set

Variables	N (%)
Total number of subjects	42
Sex	
Male (%)	21 (50%)
Female (%)	21 (50%)
Body mass index	
18,5-25	18 (42,9%)
25-30	19 (45,2%)
30-35	3 (7,1%)
>35	2 (4,8%)
ASA Score	
1	9 (21,4%)
2	26 (61,9%)
3	7 (16,7%)
	Mean [SD]
Age (years)	59 [15,1]
Weight (kg)	80 [12,4]
Height (cm)	175 [9]
eGRF (mmol/L)	84 [9,8]

ASA, American Society of Anaesthesiologists; eGFR, estimated Glomerular Filtration Rate; N, number of patients; SD, standard deviation

3.2 Dataset

During each surgical procedure, data from 3 different monitors was recorded using the data logger software. Infusion pump data was captured every 10 seconds and parameters included cumulative and infused bolus volume and cumulative and infused total volume measurements. GE NMT monitor measurements included stimulus mode, stimulus current, pulse width, TOF count, TOF ratio, T1-T4 measurement, T1 reference-% and PTC count. TOF-Cuff included stimulus current, impedance, TOF count, TOF ratio, PTC count, blood pressure measurements and the arm position. TOF-Cuff data was

captured every 12 seconds at induction, followed by 30-second interval measurements and even 2-minute intervals when the TOF ratio became zero during deep NMB. GE NMT monitor data was captured at 20-second interval at induction, followed by a 30-second interval after induction during (deep) NMB. Patient characteristics were taken from the HER and added to the dataset as well as sugammadex administration records, as this data could not be captured by the data logger software.

3.3 Data Exploration

TOF ratio curves of both the TOF-Cuff and the GE NMT monitor were visualized including the moments rocuronium was administered. An overview of all the TOF ratio curves is added as Supplement C. An example of two TOF ratio curves is shown in Figure 3-2. After calibration of the NMT monitors, a bolus of rocuronium is administered and the TOF ratio drops within minutes to 0. Depending on the type of procedure, either spontaneous recovery was allowed or extra boluses of rocuronium were administered to maintain (deep) NMB. Although these data records look rather clean, some records show deviations in the measurements recorded. For some procedures a deep NMB had to be maintained during the entire length of the procedure and rocuronium infusion was continuously administered. NMB was at the end of the procedure lifted using sugammadex. Therefore, almost no meaningful data was recorded regarding the TOF ratio curve for recovery. An example of this problem is shown in the left record of Figure 3-2. As can be seen in right record of Figure 3-2 not all TOF ratio curves are captured adequately. Therefore, the datasets were divided into a TOF-Cuff dataset and a GE NMT monitor dataset. Consequently, continuous rocuronium administration records were excluded for further analysis. After the elimination of faulty TOF-Cuff or GE NMT monitor data and after the exclusion of continuous infusion data, a total of 26 TOF-Cuff records and a total of 27 GE NMT monitor records remained.

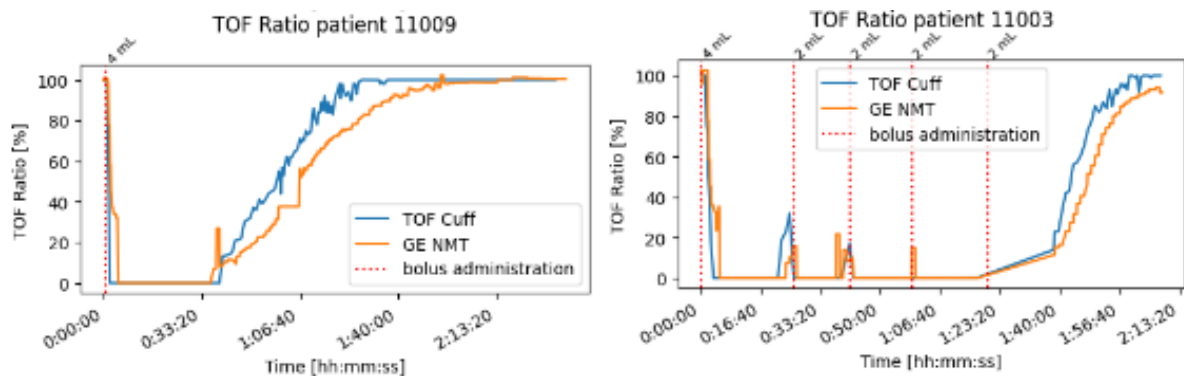


Figure 3-2 TOF ratio curve measurements with spontaneous recovery. On the left: a TOF ratio measurement with spontaneous recovery of both TOF-Cuff and GE NMT monitor after a 4 mL rocuronium administration. On the right: a TOF ratio curve with spontaneous recovery of both monitors with multiple rocuronium bolus administrations to maintain a deep NMB.

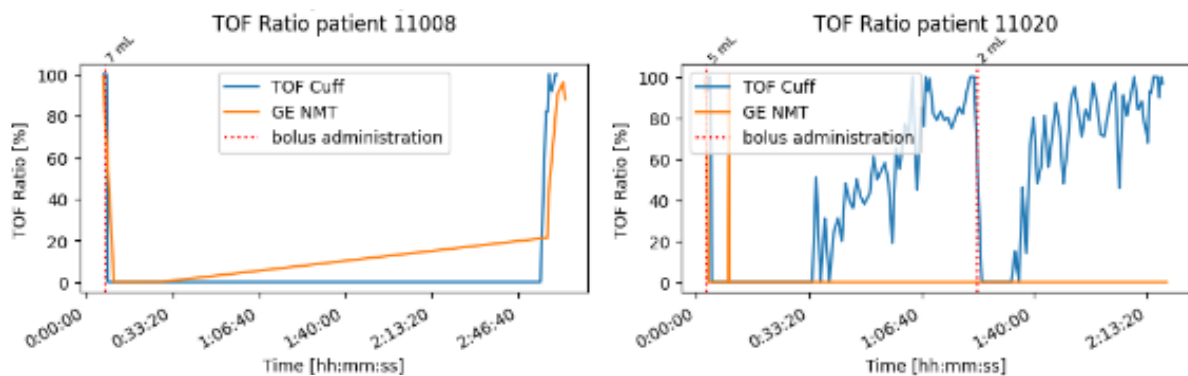


Figure 3-2 Data records with continuous rocuronium administration and sugammadex reversal (left) or partial TOF ratio curve measurement only captured by the TOF-Cuff (right).

3.4 Data Pre-processing

After removal of TOF ratio outliers, TOF ratio values were smoothed and linear interpolation was performed between consecutive measurements. Still, part of the dataset contained NaN-values. Consequently, the start of the dataset was cut back to start right before the first rocuronium bolus administration. The remaining values from the TOF ratio were filled with the value 100, assuming that right before and after the surgical procedure full NMB reversal had occurred. The remaining infusion parameters were filled with the value zero, assuming no infusion data was administered outside the measurement period.

3.5 Model Evaluation

After data pre-processing, the models were trained, validated and tested using the double LOGO cross-validation method. Each subject was held out once as a test subject and each ML model was trained on the other subjects and validated using an inner loop LOGO cross-validation. The best model was selected based on the lowest RMSE cross-validation result in the inner loop. As a final performance evaluation, the NRMSE, Pearson r correlation coefficient and the R² score were calculated. The average performance of the regression models on the test subjects is shown in Table 3-2 and Table 3-3. The best-performing models are highlighted.

Table 3-2 Performance evaluation of machine learning models and classic PKPD model on GE NMT monitor data.

	GE NMT monitor TOFR performance			
	RMSE [SD]	NRMSE [SD]	Pearson r [SD]	R ² [SD]
Linear Regression	38,99 [9,92]	0,37 [0,09]	0,55 [0,32]	-1,03 [2,85]
DecisionTreeRegressor	36,29 [13,15]	0,34 [0,13]	0,45 [0,42]	-0,66 [1,96]
XGBRegressor	31,92 [9,23]	0,30 [0,08]	0,63 [0,29]	-0,23 [1,02]
Basic PKPD model	22,97 [10,79]	0,27 [0,09]	0,82 [0,17]	0,69 [0,25]
Optimized PKPD model	23,96 [15,96]	0,09 [0,07]	0,76 [0,23]	0,63 [0,27]

PKPD; Pharmacokinetic-Pharmacodynamic, NRMSE; Normalized Root Mean Squared Error, Pearson r; Pearson correlation coefficient, RMSE; Root Mean Squared Error, R²; coefficient of determination, SD; Standard Deviation

Table 3-3 Performance evaluation of machine learning models and classic PKPD model on TOF-Cuff data.

	TOF-Cuff TOFR performance			
	RMSE [SD]	NRMSE [SD]	Pearson r [SD]	R ² [SD]
Linear Regression	45,01 [13,97]	0,45 [0,14]	0,57 (±0,24)	-1,16 (±2,79)
DecisionTreeRegressor	33,95 [10,05]	0,34 [0,10]	0,61 (±0,30)	-0,01 (±0,67)
XGBRegressor	33,09 [10,48]	0,33 [0,10]	0,67 (±0,27)	-0,07 (±1,04)
Basic PKPD model	24,22 [11,99]	0,25 [0,07]	0,81 (±0,18)	0,69 (±0,25)
Optimized PKPD model	18,53 [8,11]	0,10 [0,06]	0,82 (±0,16)	0,70 (±0,23)

PKPD; Pharmacokinetic-Pharmacodynamic, NRMSE; Normalized Root Mean Squared Error, Pearson r; Pearson correlation coefficient, RMSE; Root Mean Squared Error, R²; coefficient of determination, SD; Standard Deviation

To better understand the results shown in the tables above and to see how the models performed on different subjects, an in-depth analysis has been performed on the NRMSE as these results show the normalized performance for all models.

The results in Figure 3-4 show an almost uniform distribution for the NRMSE in all models, but there are some outliers as can be seen on the right side for the optimized PKPD model. To understand these observations even better, a cumulative distribution was plotted for all models. This can be seen in Figure 3-4.

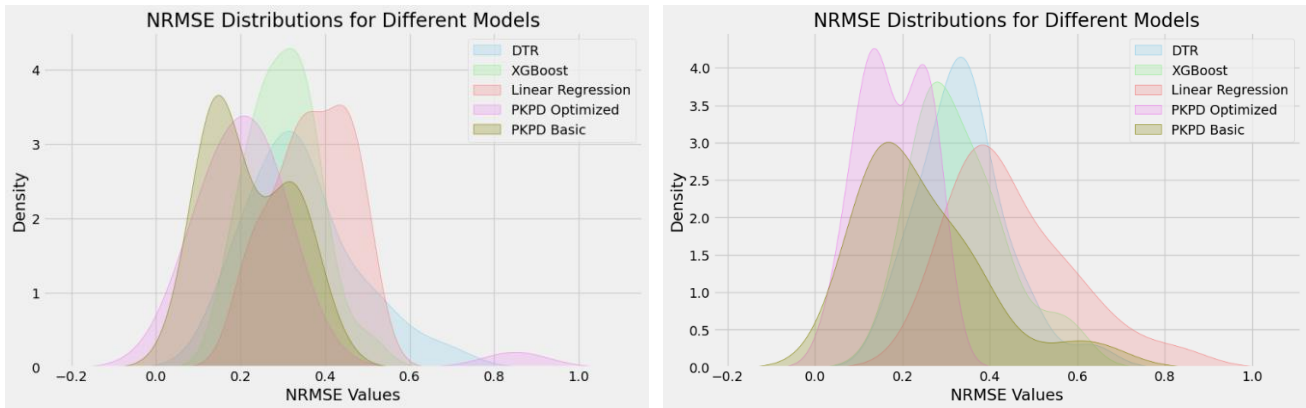


Figure 3-4 Kernel Density Estimation of the NRSME for the GE NMT dataset (left) and the TOF-Cuff dataset (right).

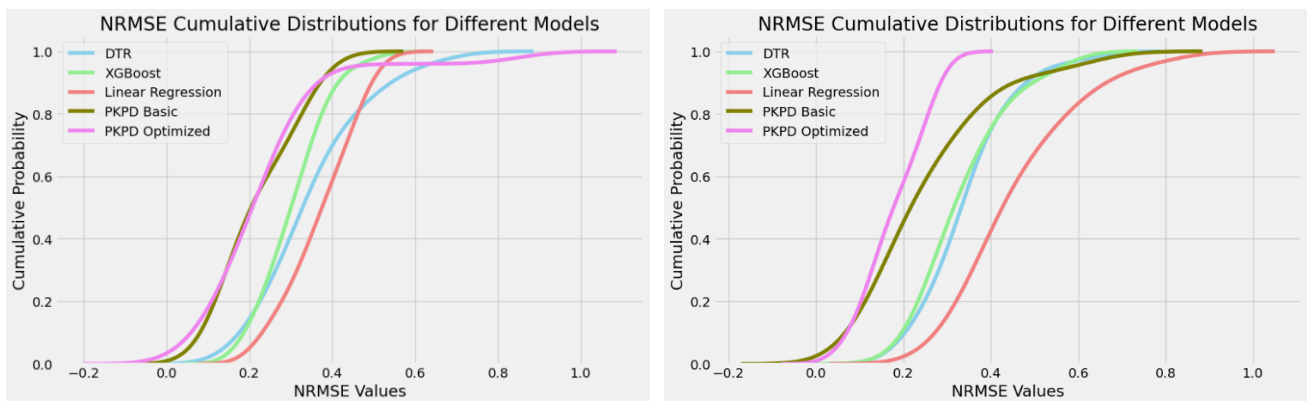


Figure 3-4 Cumulative Distribution of the NRMSE for all models. GE NMT data on the left and TOF-Cuff data on the right.

Based on the results from the cumulative distribution in Figure 3-4, some subjects have rather high NRMSE scores for both models, especially in the case of the TOF-Cuff results. Figure 3-5 and Figure 3-6 show the individual NRMSE scores for each subject per model ordered from small to large. For the three ML models, the best and the worst predictions are shown in Figure 3-7 and Figure 3-8.

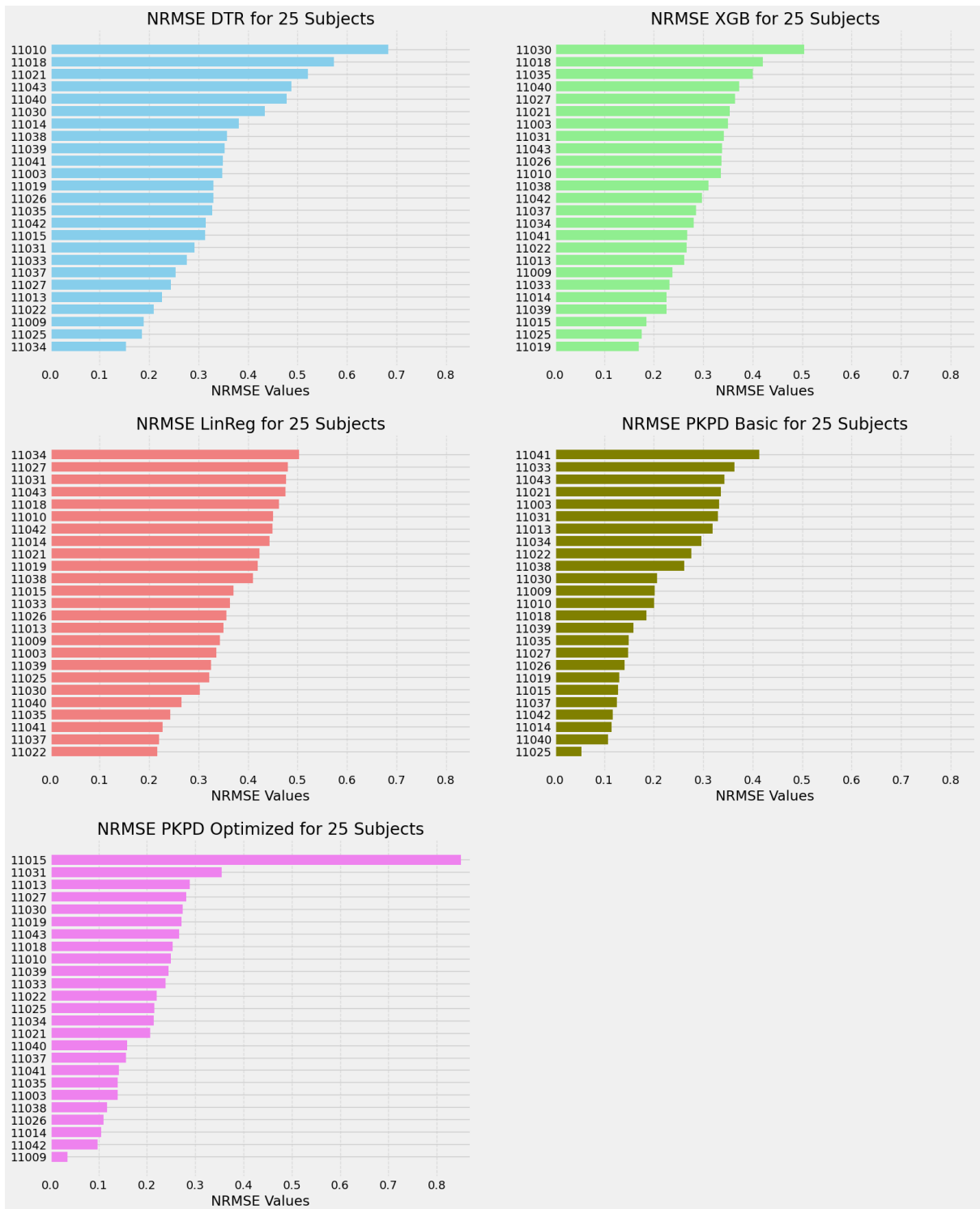


Figure 3-5 Visualization of the NRMSE per subject for each model trained on GE NMT data ordered from small to large.

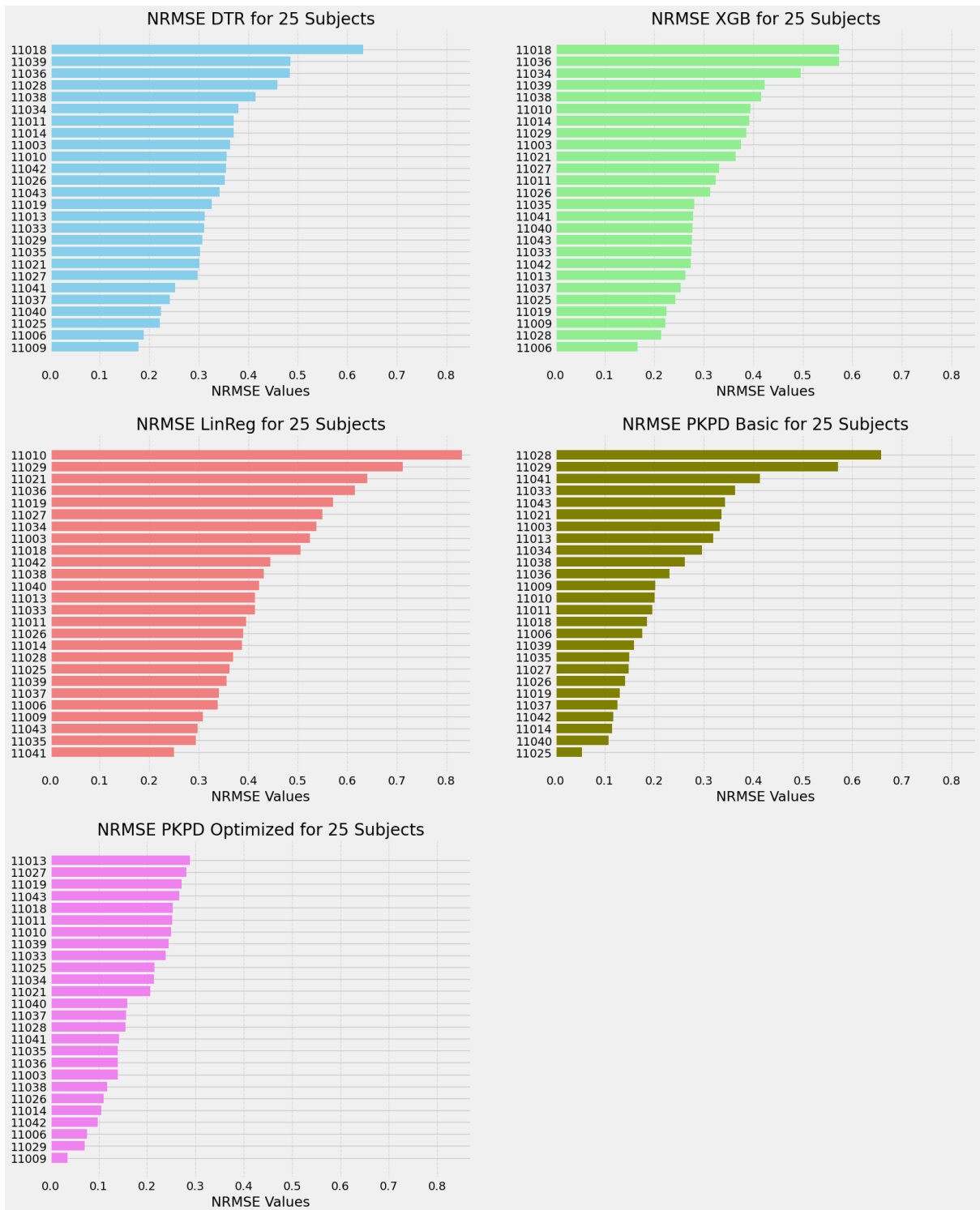


Figure 3-6 Visualization of the NRMSE per subject for each model trained on TOF-Cuff data ordered from small to large.

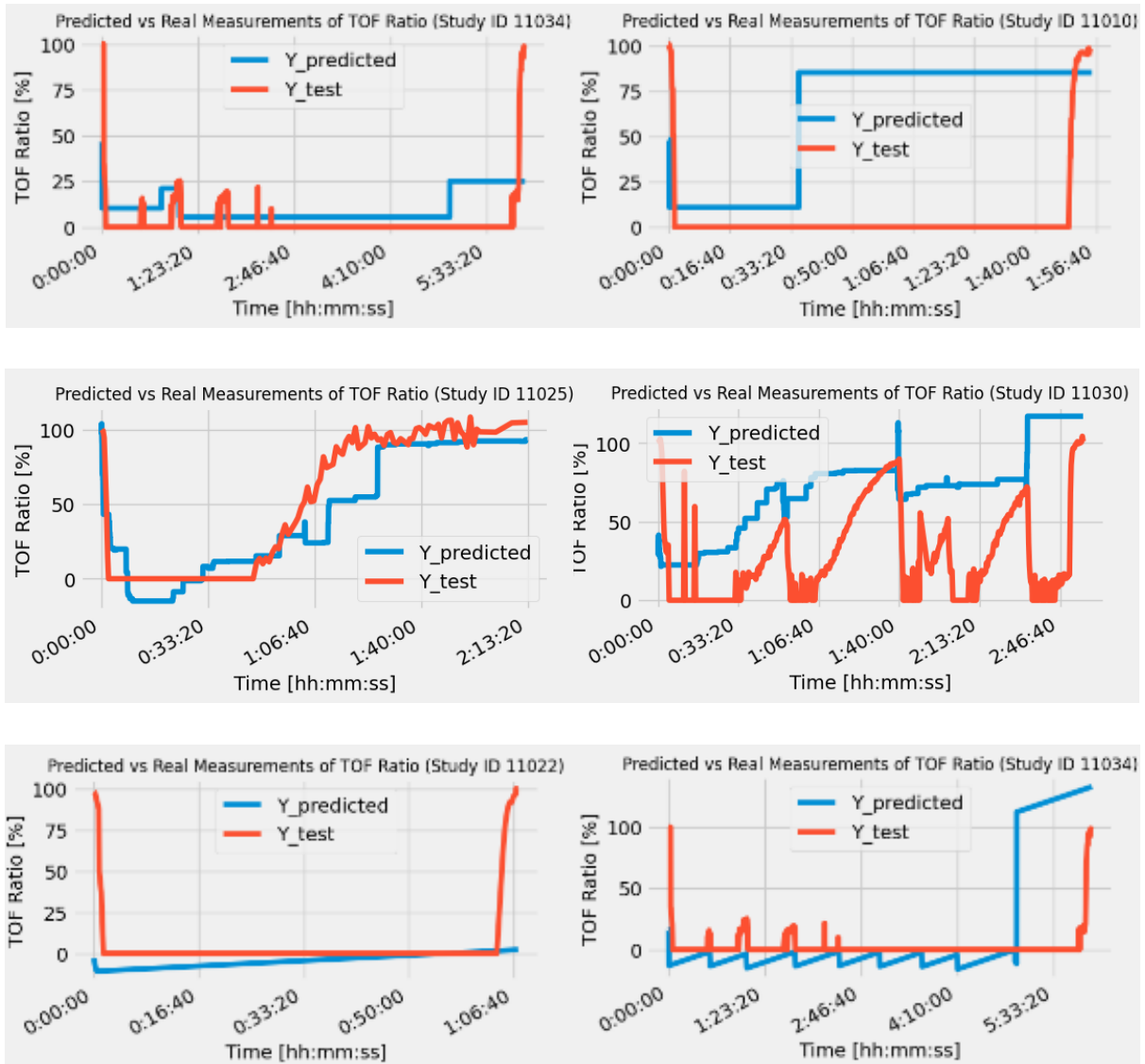


Figure 3-7 Highest (left) and lowest (right) scoring machine learning models based on NRMSE score for the GENMT dataset. Decision Tree Regressor (top), XGBRegressor (middle) and Linear Regression (bottom).

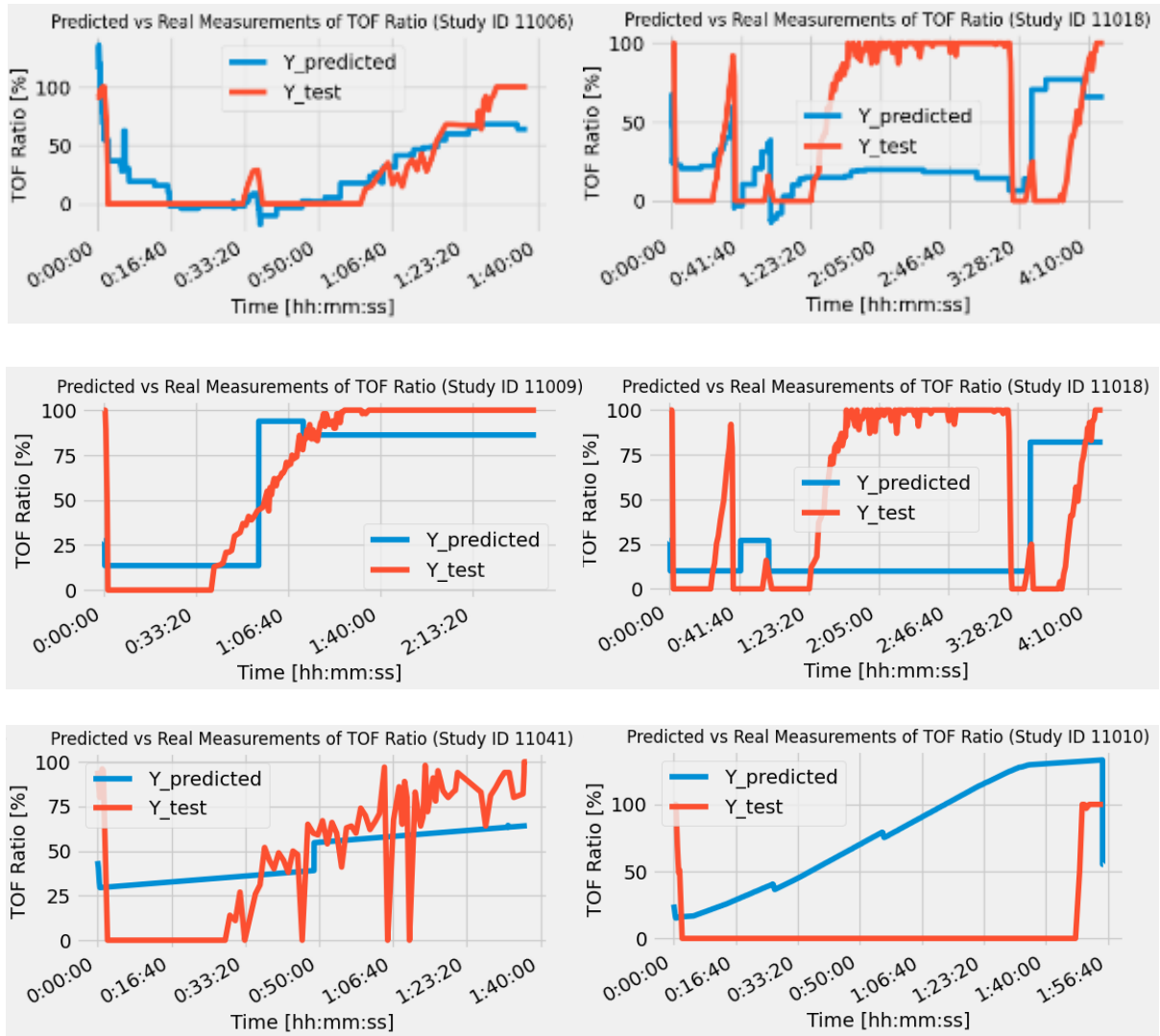


Figure 3-8 Highest (left) and lowest (right) scoring machine learning models based on NRMSE score for the TOF-Cuff data set. Decision Tree Regressor (top), XGBRegressor (middle) and Linear Regression (bottom).

4. Discussion

In this prospective, observational study the main goal was to apply machine learning techniques to approximate the TOF ratio curve in comparison with classical PKPD modelling. NMT data was recorded using an in-house developed automated data logger.

Data logger development

A frequently used tool to collect and store real-time data is the VitalDB recorder [12]. Vital DB offers an open-source program that can be used mainly for research purposes. Through this program, you can connect several patient monitors and through a user-friendly user interface you can select the monitors and read out selected data streams. Unfortunately, this program is unsuitable for this study because not all monitoring devices were compatible with the VitalDB software. In addition, vitalDB uses physical connections between the monitors and the PC on which the program runs. This causes extra space occupation in the OR and it is only possible to monitor successful data collection in the OR. To overcome these problems, much of this study has focused on developing proprietary software that is suitable for LUMC's NMT devices and can be controlled remotely. By adding a communication protocol that transfers serial communication to the IP, data collection can be viewed anywhere in LUMC network. This is a major advantage over VitalDB's software because there is a limit to the length (<15m) of the serial cables at a high baud rate (19,200 kbps). Since the distances at the OR do not exceed 15m, and no cables have to be routed outside the OR, the limits of serial communication can be overcome. In addition, no additional devices are needed in the OR except for some ethernet cables connected to a medical-graded device, the Lantronix. This device is licensed to be placed in sterile areas.

Unfortunately, some problems arose while developing the algorithm and conducting the study. First of all, at certain periods during a surgical procedure, data dropped from the Arcomed syringe pump. Reconnecting the docking station, changing USB ports or restarting the data logger software made no difference. This issue needs to be addressed to the manufacturer to debug this problem even further. Data capturing of the GE NMT monitor also encountered some problems. When the twitch current was set to 200, no data was detected by the data logger software. By simply changing the current to a different value, the data capturing could be continued. When debugging this problem, the incoming byte string did not meet the specified requirements as described by the communication manual. Therefore, the data was disregarded as faulty data and not stored. A possible explanation for this problem at hand is the way the CRC checksum is calculated. The communication manual did not specify what CRC checksum calculation was needed and a simple CRC calculator was implemented. This CRC calculator always returned a CRC value between 0 and 255.

These imperfections need to be addressed before the algorithm is stable enough to record more data in the OR, especially when clinicians need to use the software. The majority of clinicians lack the technical knowledge to debug the software. They can only work with the user interface.

Dataset

With the data logger software, 42 subjects were recorded during different surgical procedures. Although the initial sample size calculation was estimated at 44 patients, only 27 and 26 for the GE NMT monitor and the TOF-Cuff, respectively, were useful for full data analysis. One of the reasons for this low amount of useful datasets would be that some surgical procedures require deep NMB. Therefore, anesthesiologist chose to administer rocuronium continuously. As a result, the TOF ratio had a value of 0 for a long period of time. Despite the fact that you could measure the PTC during deep NMB, there is limited information available in this phase of NMB. Although spontaneous recovery could still provide useful information, these types of surgery often require deep NMB throughout the entire duration of the procedure. Due to limited time for spontaneous recovery sugammadex is often administered to reverse the NMB, resulting in very rapid restoration of the TOF ratio.

In other cases, data from one (or both) of the monitoring devices were missing. Due to inaccurate placement of the sensors or after repositioning a patient, data could not be captured. There was a limited time window between checking sensor placement, calibrating the monitors, and administering the first

bolus of rocuronium. As the patient is already under anesthesia by the administration of propofol, the anesthesiologist's primary objective is to administer rocuronium for safe intubation. After successful intubation, the patient is often repositioned and covered with sterile cloths, which makes it hard to reach and adjust the position of the sensors.

Model development

The development of the traditional model of Kleijn, et al. is based on PK and PD parameters carefully constructed by performing multiple blood sample measurements on 423 patients. This ensures that reasonably reliable relationships can be drawn from patient characteristics incorporated into the PKPD model. When presented with new patient characteristics, a close representation of the TOFR can be predicted. In contrast, only 27 patients could be included in the development of the machine learning models, resulting in a poor fit to predict TOF ratios. The small sample dataset increased the possibility of overfitting. This means that the predictions on the validation set may be accurate, but the performance of the model on a new unseen sample from the test set may be poor. For this reason, 8-fold cross-validation was chosen during hyperparameter tuning. By applying multiple folds, the model becomes more generic and thus reduces the chance of overfitting. After hyperparameter tuning, the models were presented with a double LOGO cross-validation to include as many subjects as possible. This ensured that the majority of the subjects were included for training while preserving subjects for independent test results. Another benefit of the LOGO method is that all subjects are tested. This way, the performance of the model is tested on each subject once, which allows for more accurate performance evaluation.

For this analysis, no extra features were added to the dataset or derived from the dataset. At this point, only the time since rocuronium bolus administration gave information regarding the TOFR curve. Because there is no feedback from the system, the models can be seen as an *in t* system. A more realistic approach would be to include previous TOFR measurements in the prediction of the next TOFR value. This means that at the start of a measurement, you don't have information about the previous values, but as time progresses, you get more TOFR measurements and therefore more information. Using this previous information can highly boost the performance of the model as the model becomes a closed loop system.

Model Evaluation

When evaluating the different models, the performance of the machine learning models was poor in comparison with the PKPD models. The linear regression, decision tree regressor and the XGBRegressor scored lower on the NRMSE scores and the R^2 score fell below zero. Normally, an R^2 score between 0 and 0.2 can be seen as a poor model. When looking at the correlation coefficient, which gives the correlation between the actual TOFR and the predicted TOFR, we see that again the correlation for the ML models score lower than the PKPD models. A correlation coefficient of 1 would be an exact match between the predicted and the real TOFR values. Despite the ML models scoring lower, there is still a moderate positive correlation between the predicted TOFR curves and the real TOFR values, with the XGBRegressor showing the highest correlation. Although the metrics show poor results for the XGBRegressor, the TOFR curve predicted by the XGBRegressor is most flexible when visually inspecting the predictions of the ML models. Therefore, the XGBRegressor shows the most potential. This is most likely due to the ensemble learning and gradient boosting. Ensemble learning combines decision trees to make predictions. A decision tree regressor is limited to only one decision tree. Gradient boosting iteratively corrects the errors made by the previous trees.

The performance results also show a difference in performance between the models on the GE NMT dataset and the TOF-Cuff dataset. All models score higher on the GE NMT dataset. One possible explanation is that the data captured by the GE NMT monitor could be more accurate than the data captured by the TOF-Cuff. The data via the EMG is captured using EMG rather than determining a difference in cuff pressure. In addition, there is a difference in sensitivity between central muscles (such

as the diaphragm and upper arm) and peripheral muscles (such as the adductor pollicis muscle near the thumb). Specifically, the muscle receptors need not be completely dissolved from rocuronium to regain their function [13]. The TOFR can only be measured when there is a receptor occupancy of 70%.

Future directions

Regardless of the unfavorable results of the machine learning models, collecting more data might resolve the performance issues encountered in this study. At least 44 patients should be included in that analysis, preferably closer to 100 subjects for a definitive comparison with the traditional PKPD models. With the use of the data logger software and with more time, collecting more data should be feasible.

No feature analysis has been performed in this study, nor extra features have been added. Maybe a more informative machine learning approach would include an estimation of the time until full NMB recovery. This way, a clinician can wait for spontaneous recovery or decide to administer sugammadex for a quick and full NMB reversal.

In future development of the machine learning models, the first recommended step would be to determine the most important features on which XGB based its predictions. This would render relevant information concerning the additional value of the development of this type of machine learning to a computationally less expensive logistic regression method.

Conclusion

In this thesis, as a proof of concept, data logger software has been developed and implemented at the OR complex of the LUMC. Neuromuscular blockade data during general anesthesia was successfully captured and analyzed. Despite the underwhelming performance of ML models in this study, collecting a larger dataset and conducting feature analysis may improve their predictive capabilities. Future research should focus on obtaining more data, determining essential features for ML predictions, and considering additional factors, such as the time until full neuromuscular blockade recovery, to enhance model accuracy and usability. These steps could pave the way for a more efficient and informative ML-based approach to neuromuscular blockade prediction, potentially complementing traditional PKPD models.

References

1. Kuznetsov, D.A., et al., *A tumor selective chemotherapy. Can this be managed by an algorithm based on the non- Markovian population dynamics?* 2023.
2. Khuenl-Brady, K.S. and H. Sparr, *Clinical pharmacokinetics of rocuronium bromide*. Clin Pharmacokinet, 1996. **31**(3): p. 174-83.
3. Duțu, M., et al., *Neuromuscular monitoring: an update*. Rom J Anaesth Intensive Care, 2018. **25**(1): p. 55-60.
4. Martyn, J.A.J., M.J. Fagerlund, and L.I. Eriksson, *Basic principles of neuromuscular transmission*. Anaesthesia, 2009. **64**(s1): p. 1-9.
5. Kleijn, H.J., et al., *Population pharmacokinetic-pharmacodynamic analysis for sugammadex-mediated reversal of rocuronium-induced neuromuscular blockade*. Br J Clin Pharmacol, 2011. **72**(3): p. 415-33.
6. Ali, H.H., et al., *Quantitative assessment of residual curarization in humans*. Br J Anaesth, 1970. **42**(9): p. 802-3.
7. Kazuma, S., et al., *Comparative Study of TOF-Cuff, a New Neuromuscular Blockade Monitor, and TOF-Watch, an Acceleromyography*. Anesth Analg, 2019. **129**(1): p. e16-e19.
8. Lee, H.C., et al., *Prediction of Bispectral Index during Target-controlled Infusion of Propofol and Remifentanyl: A Deep Learning Approach*. Anesthesiology, 2018. **128**(3): p. 492-501.
9. Ingrande, J., et al., *The Performance of an Artificial Neural Network Model in Predicting the Early Distribution Kinetics of Propofol in Morbidly Obese and Lean Subjects*. Anesth Analg, 2020. **131**(5): p. 1500-1509.
10. Wang, K., et al., *The Real-Time and Patient-Specific Prediction for Duration and Recovery Profile of Cisatracurium Based on Deep Learning Models*. Front Pharmacol, 2021. **12**: p. 831149.
11. *Python: A high-level programming language*. 2020, Python Software Foundation.
12. Lee, H.-C. and C.-W. Jung, *Vital Recorder—a free research tool for automatic recording of high-resolution time-synchronised physiological data from multiple anaesthesia devices*. Scientific Reports, 2018. **8**(1): p. 1527.
13. Connor, T. *A History of Neuromuscular Block and Its Antagonism*. 2017 [cited 2023 Oct 10]; Available from: <https://aneskey.com/a-history-of-neuromuscular-block-and-its-antagonism/>.

Supplement A. Pinout configurations of converters

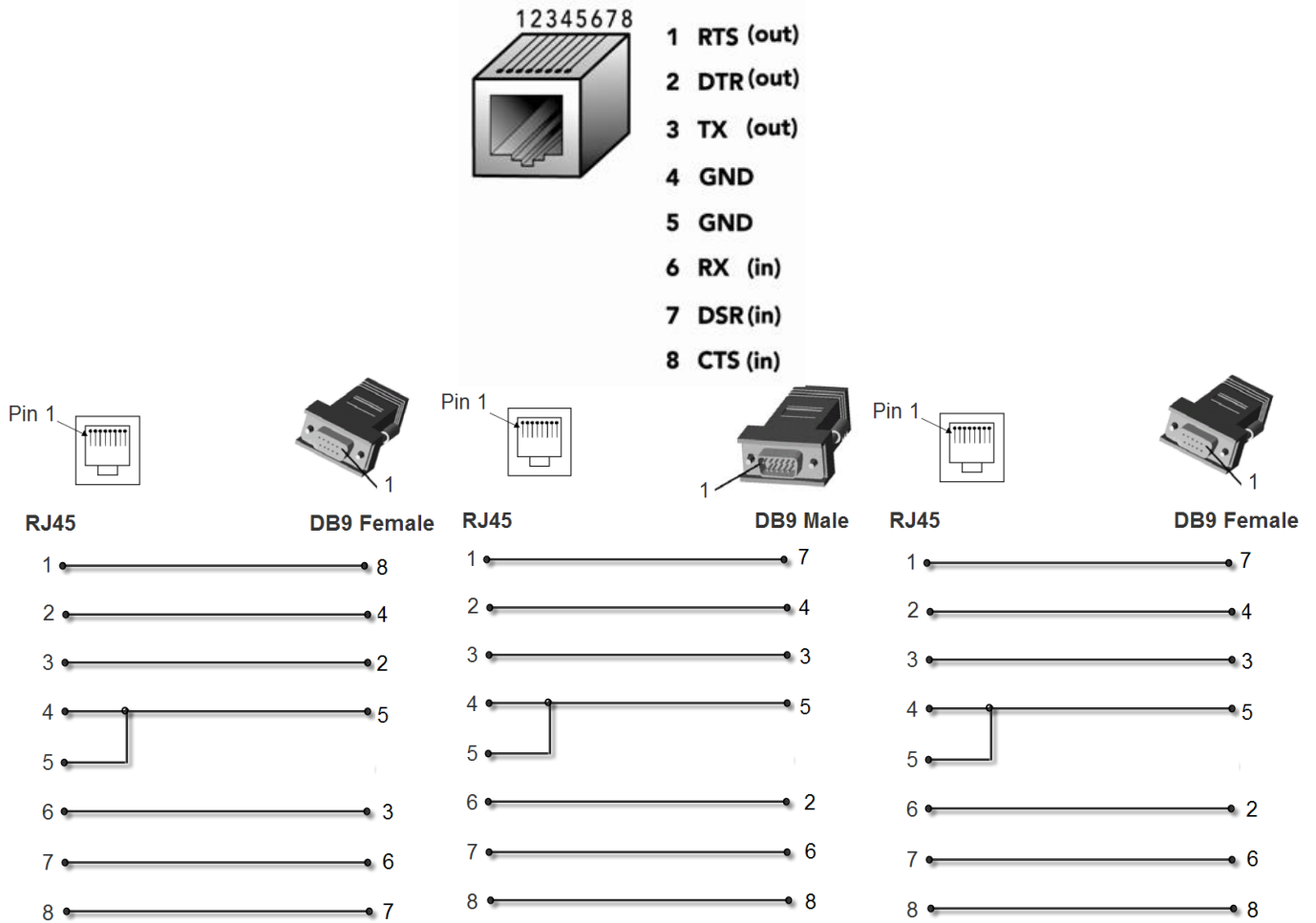


Figure A-1: Schematic representation of the pinout of a RJ-45 to DB9 converter for GE NMT Monitor (left), Arcomed pump (middle) and TOF-Cuff (right). Notice how GE NMT monitor crossed the Rx, Tx, CTS and CTR connector pins. For the RJ45 socket the pinout with corresponding pins is shown. *Copied and adjusted from Lantronix manual.

Supplement B. Data logger OR – user manual

PURPOSE

This user manual describes the standard operation procedure (SOP) for correctly connecting all monitors to the Lantronix in order to retrieve real-time and high-frequency data using the Data Logger running on an internal LUMC server (ldl-app01.lumcnet.prod.intern).

REQUIRED MATERIALS

Table B-1 Connection materials per monitor

TOF-Cuff	Arcomed docking station	GE NMT monitor	IntelliVue
Serial-to-RJ45 adapter UTP cable	Docking station USB A-to-mini-USB cable Mini-USB-to-serial adapter Serial-to-RJ45 adapter UTP cable	USB-to-serial adapter Serial-to-RJ45 adapter UTP cable	RJ45-converter UTP cable



Table B-2 Port registrations on Lantronix devices

LANTRONIX	TOF-CUFF	ARCOMED DOCKING STATION	GE NMT MONITOR	INTELLIVUE
EXTRA	4	1	7	2
OK8	4	1	7	2
OK9	4	1	7	2



Figure B-1 Connection ports on TOF-Cuff (1a), Arcomed docking station (1b) and GE NMT monitor (1c)

CONNECTING MONITORS

Connecting TOF-Cuff

1. Connect the adapter to the RS-232c port on the side of the TOF-Cuff (Figure 1a).
2. Connect the TOF-Cuff to the appropriate Lantronix port using a UTP cable (see Table B-2).

Connecting Arcomed docking station

1. Connect the USB to mini-USB cable to one of the two USB ports on the back of the Arcomed docking station (Figure 1b).
2. Connect the two adapters and connect the adapters using a UTP cable to the Lantronix port (Table B-2).
3. Place the syringe pump in the BOTTOM position of the docking station.

NOTE: At this time, only the bottom syringe pump slot can be read without any other pumps being placed in the docking station. Therefore, position the pump at the bottom of the rack.

Connecting GE NMT monitor

1. Connect the USB to Serial cable to one of the two USB ports on the back of the GE NMT monitor (refer to Figure 1c).
2. Connect the adapter and link it to the correct Lantronix port using a UTP cable (see Table B-2).

Connecting Lantronix (if necessary)

1. Connect a UTP cable from the **MAIN port** of the ‘research’ Lantronix to **ethernet port 2 or 3** on the left side of the existing ‘PDMS’ Lantronix.

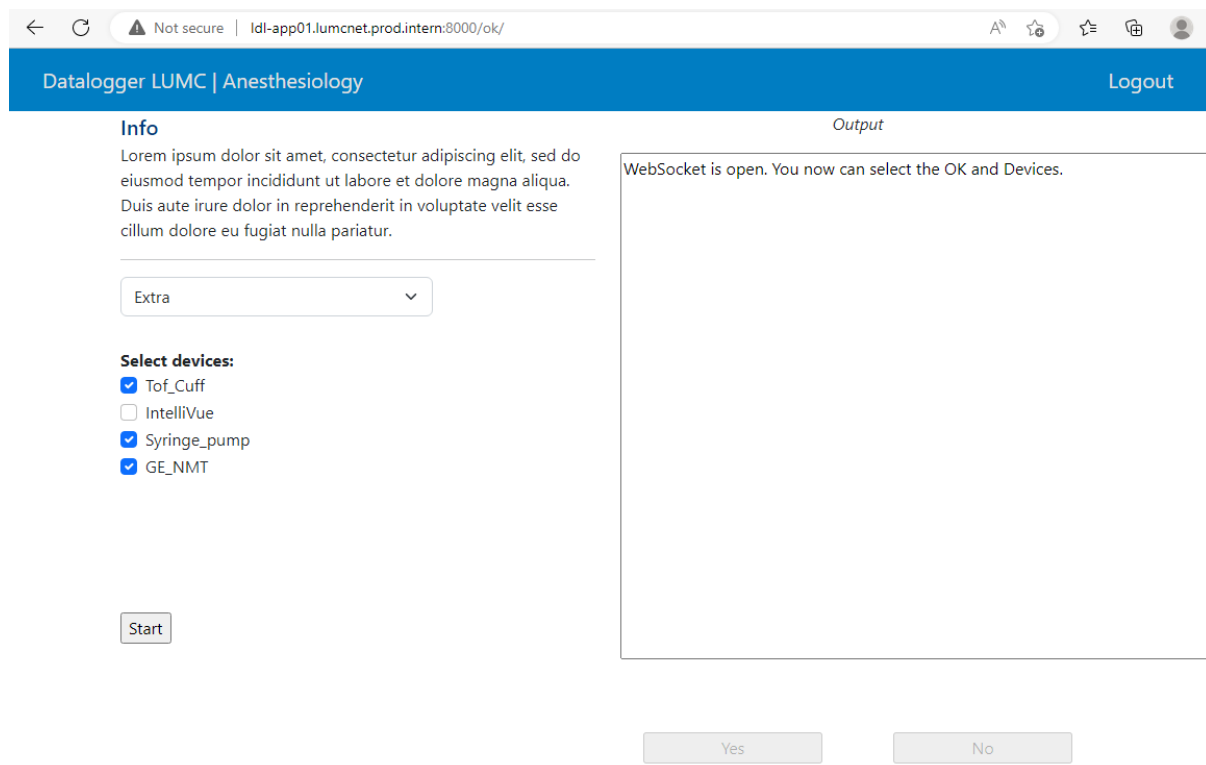
NOTE: the PDMS Lantronix should be connected **at all times** to the internet as the IntelliVue patient monitor and ventilator are connect to the electronic health record (EHR, i.e. HiX)

ACTIVATING DATA LOGGER

1. Turn on all monitors just before use.
2. Navigate to <http://ldl-app01.lumcnet.prod.intern:8000/> on any LUMC PC.
3. Log in using the known login credentials.



4. Select the appropriate OR from the dropout menu and choose the desired devices to collect data.



5. Press Start and verify that the connections are confirmed and data is being received.

CLOSING DATA LOGGER

1. Press Stop and wait at least 10 seconds to stop all processes.

Supplement C. TOF Ratios per subject

