2023

Classification Models for Predicting Mouse Spinal Motoneuron Physiological Type based on their Electrical Properties

Reuben Mawuena Kwadzo Ahorklo
University of Rhode Island, rmahork@uri.edu

CLASSIFICATION MODELS FOR PREDICTING MOUSE SPINAL MOTONEURON PHYSIOLOGICAL TYPE BASED ON THEIR ELECTRICAL PROPERTIES

BY

REUBEN AHORKLO

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

STATISTICS

UNIVERSITY OF RHODE ISLAND

2023

MASTER OF SCIENCE THESIS

OF

REUBEN AHORKLO

APPROVED:

Thesis Committee:

Major Professor

Natallia Katenka

Manuel Marin

Yichi Zhang

Jonathan Chavez-Casillas

Jing Wu

Brenton DeBoef
DEAN OF THE GRADUATE SCHOOL

UNIVERSITY OF RHODE ISLAND

2023

ABSTRACT

Accurate prediction of mouse spinal motoneuron physiological types based on electrical properties faces challenges due to missing data and imbalanced class distributions. Technical difficulties, physiological variations, and experimental issues contribute to data gaps in electrophysiological recordings. Imbalanced class distributions arise from the rarity of certain motoneuron types. The resulting risk of biased or unreliable classification models hampers their utility in motor control studies. Thus, we claim that the electrical properties of mouse spinal motoneurons can be accurately predicted and classified based on specific, measurable features.

This study focuses on two classification models, a multinomial logistic regression (MLM) and a Random Forests (RF) model, to predict motoneuron physiological types based on electrical properties since motoneurons' vital role in signal transmission relies on diverse electrical properties. Both model types are applicable to more-than-two-class problems, and MLM excels in subtle pattern identification, while RF handles complex relationships within the data. We investigate the impact of the threshold choices on the class distribution, class-specific, and overall prediction accuracy. This analysis showed that the model's performance in terms of accuracy depends on the threshold set. Next, we incorporated the over-sampling technique and hotdeck imputation to compensate for class imbalances and missing data. While contingent on selecting an appropriate threshold, the results illustrate that imputation and oversampling offer notable benefits by preserving data size, thereby enhancing the accuracy and stability of classification models. Specifically, when we set the contraction time to 20 ms and twitch amplitude to 8m mN, we demonstrated that incorporating imputation techniques for handling missing data and utilizing resampling methods to address class imbalances significantly enhances the overall accuracy of multinomial logistic model to 0.78. Class-specific

accuracy ranges between 0.64 and 0.88, contributing to the robustness of motor unit classification based on electrical properties. The emphasis on managing missing data, addressing imbalanced class distributions, and understanding the predictor-response relationship (Y) guided the preference for MLM. The decision to exclude other models was based on data characteristics, small sample size, and specific project goals. This research advances our understanding of motor control and suggests potential clinical applications in diagnosing and treating motor neuron diseases, such as amyotrophic lateral sclerosis (ALS). In ALS, determining physiological types relies on contractile properties rather than traditional measures like computed twitch contraction or amplitude time.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my major supervisor, Dr. Natallia Katenka, and my co-major supervisor, Dr. Marin Manuel, for their invaluable guidance, support, and expertise throughout the completion of this thesis on Classification Models for Predicting Mouse Spinal Motoneuron Physiological Type based on their Electrical Properties, despite the challenges of dealing with missing data and imbalanced class distribution. Their constructive feedback and insightful suggestions have been crucial in shaping the direction of this research and improving the quality of the final product.

I would also like to extend my appreciation to my internal committee members, Dr. Jing Wu, Dr. Yichi Zhang and Dr. Johnathan Chavez-Casillas, for their valuable input and feedback that helped to strengthen this work. Their critical reviews, thoughtful comments, and constructive criticisms have been instrumental in shaping the direction of my research and improving the quality of my work.

Finally, I am deeply grateful to all my colleagues, friends especially Amari, Dom and Delvin, and family members who have supported and encouraged me throughout my research journey. Their unwavering support and motivation have been a source of inspiration for me, and I am thankful for their unwavering support throughout this journey.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Figure **Page**

Figure **Page**

LIST OF TABLES

CHAPTER 1

Introduction

Accurate prediction of mouse spinal motoneuron physiological types based on electrical properties faces challenges due to missing data and imbalanced class distributions. Although motor units are classified based on their contractile properties, seminal experiments in cats have demonstrated that motoneurons also exhibit different properties based on their motor unit types [1],[2]. S-type motoneurons are smaller, require less current for firing, and tend to fire at low frequencies; FF-type motoneurons are the largest, need the most amount of current, and fire at high frequencies; FR-type motoneurons have an intermediate profile. More recently, simultaneous recordings of motoneuron properties and the force developed by their motor units have been performed in mice [3],[4], confirming this general trend in this species, albeit with quantitative differences [1]. However, there are large overlaps in these properties between motoneurons of different types. Currently, the only way to reliably identify motor unit type remains to perform experiments during which the electrical properties of motoneurons and the forces developed by their motor unit are recorded simultaneously. These experiments are very challenging. The number of motoneurons that can be recorded in a single experiment is small. Furthermore, the time during which each motoneuron can be recorded is highly variable and unpredictable, which means that, often, only a subset of the electrical properties of the recorded motoneuron can be obtained before the motoneuron is lost. Our knowledge of the relationships between the electrical properties of the motoneurons and the contractile properties of their motor unit is further limited by the fact that the three types of motor units are present with variable proportions in different muscles [5]. In the muscles of the hind limbs that are commonly

studied, the number of motoneurons belonging to the FF or S types is very small compared to the FR motoneurons.

Statistical classification models are essential tools in various domains, from medical diagnosis to credit risk assessment and natural language processing. These models encounter significant challenges when dealing with unbalanced class distribution and missing data. Unbalanced design, being a problem in our work, refers to a situation where the number of observations in different classes is unequal, thus leading to biased estimates and poor model performance [6]. Various oversampling and undersampling methods, such as Synthetic Minority Over-sampling Technique (SMOTE) and Adaptive Synthetic (ADASYN), with 154 instances classified as FR, 16 as S, and 54 as FF, we opted for oversampling to uphold the dataset's overall size, thereby averting potential information loss associated with downsampling. This approach is versatile across different machine learning models, enhancing compatibility with various algorithms and helping mitigate bias by ensuring the model allocates appropriate attention to minority class instances.

Similarly, missing data is a problem where some observations lack values for specific variables, which can also introduce bias and reduce the statistical power of the models. Imputation methods like single and multiple imputations help fill in missing values[7]. The rejection of the null hypothesis in the Little's Missing Completely at Random (MCAR) test, indicated by a p-value of 0.0002 (≈ 0) being less than the significance level α of 0.05, prompted the acknowledgment that the missing data is not MCAR. Consequently, we adopted the assumption of Missing at Random (MAR) status, recognizing that the missing data depends on observed variables. In response, we employed hot deck imputation to tackle the missing values, drawing on schemes that select units based on similarity, as outlined by [8]. This strategic choice was motivated by recognizing that the electrical properties,

which are pivotal predictors for motoneuron classification, exhibit inherent similarities and interrelationships. These electrical properties, encompassing membrane potential dynamics, firing patterns, and responses to stimuli, jointly influence the classification outcomes, revealing their interconnected nature.

While research on statistical classification models for unbalanced class distribution and missing data shows promise, there is still room for improvement and a need for more effective methods since there are several reasons to believe that this field is not yet fully optimized or that current methods may have limitations. Ensemble methods are machine learning techniques combining the predictions of multiple base models to produce a more robust and accurate final prediction instead of relying on a single model. Techniques such as random forests, decision trees, and gradient boosting have been explored to enhance the performance of classification models by combining multiple base models[9].

Our analysis initially considered several classification methods, including the Multinomial Logistic Regression models, Decision trees, and Boosting techniques. However, after careful consideration, we settled on using Multinomial Logistic Regression and Random Forest for specific reasons. Even though many classification methods focus on multi-class problems, Random Forest features very high accuracy, is notably efficient with large data sets, and provides an estimate of important variables in classification. Random forest generated can be saved and reused and unlike other models, it does not overfit with more features. The random forest technique can also handle big data with numerous variables running into thousands. It can automatically balance data sets when a class is more infrequent than other classes in the data. Hence, this probably is not as sensitive to class imbalance or missingness. On the other hand, multinomial logistic regression is easier to implement and interpret and very efficient to train. Moreover, it is very fast at clas-

sifying unknown records and performs well when the dataset is linearly separable. MLM can interpret model coefficients as indicators of feature importance(unlike random forest or any other classifiers).

With this insight, our primary goal is to leverage on these statistical models to address classification challenges related to the motoneuron physiological types in mice. In neuroscience, motor units serve as the fundamental building blocks of the skeletal muscle system, responsible for generating force and facilitating movement. Classically, motor units have been categorized into three main types(Figure2): Slow-twitch (S), Fast-twitch (FF), and Fast-Fatigue Resistant (FR) [10], [1], [5]. Each motor unit type exhibits distinct properties and functions. The Slow-twitch (S) motor units comprise type I muscle fibers, characterized by their high oxidative and low glycolytic capacity. These muscle fibers are well-suited for sustaining low-intensity activities, such as postural control, as they experience slower fatigue rates.

In contrast, the Fast-twitch (FF) motor units are larger and more powerful but also more prone to fatigue. These motor units generate rapid, forceful contractions, making them prevalent in muscles used for explosive movements and high-force activities. They rely less on oxidative capacity and have a higher concentration of glycolytic enzymes, enabling them to break down glycogen for quick energy production. Consequently, they are efficient in short bursts of intense exertion, such as sprinting or weightlifting, but exhibit limited endurance. The Fast-Fatigue Resistant (FR) motor units occupy an intermediate position. These motor units contract quickly but rely on oxidative respiration to produce energy, making them relatively fatigue-resistant. Despite their quick response, they tend to develop less force than FF motor units.

To address these challenges, our project aims to develop a statistical classifica-

tion model trained on data on motoneuron electrical properties and corresponding labels for the motoneuron type. We seek to explore the potential of statistical classification models in resolving issues of unbalanced class distribution and missing data in neuroscience research. Ultimately, we aspire to accurately predict the physiological type of motoneurons solely based on their electrical properties, even when access to force output is restricted. This research has broader implications for other fields facing similar challenges.

List of References

[1] M. Manuel and D. Zytnicki, "Molecular and electrophysiological properties of mouse motoneuron and motor unit subtypes," *Current opinion in physiology*, vol. 8, pp. 23–29, 2019.

[2] J. E. Zengel, S. A. Reid, G. W. Sypert, and J. B. Munson, "Membrane electrical properties and prediction of motor-unit type of medial gastrocnemius motoneurons in the cat," *Journal of neurophysiology*, vol. 53, no. 5, pp. 1323–1344, 1985.

[3] M. Manuel and D. Zytnicki, "Alpha, beta and gamma motoneurons: functional diversity in the motor system's final pathway," *Journal of integrative neuroscience*, vol. 10, no. 03, pp. 243–276, 2011.

[4] M. d. L. Martinez-Silva, R. D. Imhoff-Manuel, A. Sharma, C. Heckman, N. A. Shneider, F. Roselli, D. Zytnicki, and M. Manuel, "Hypoexcitability precedes denervation in the large fast-contracting motor units in two unrelated mouse models of als," *Elife*, vol. 7, p. e30955, 2018.

[5] C. Heckman and R. M. Enoka, "Motor unit," *Comprehensive physiology*, vol. 2, no. 4, pp. 2629–2682, 2012.

[6] X.-W. Chen and M. Wasikowski, "Fast: a roc-based feature selection metric for small samples and imbalanced data classification problems," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008, pp. 124–132.

[7] J. L. Schafer and J. W. Graham, "Missing data: our view of the state of the art." *Psychological methods*, vol. 7, no. 2, p. 147, 2002.

[8] R. J. Little and D. B. Rubin, *Statistical analysis with missing data*. John Wiley & Sons, 2019, vol. 793.

[9] V. A. Dev and M. R. Eden, "Formation lithology classification using scalable gradient boosted decision trees," *Computers & chemical engineering*, vol. 128, pp. 392–404, 2019.

[10] M. Bkaczyk, M. Manuel, F. Roselli, and D. Zytnicki, "Diversity of mammalian motoneurons and motor units," in *Vertebrate Motoneurons*. Springer, 2022, pp. 131–150.

CHAPTER 2

Review of Literature

2.1 Classification Methods

Classification methods play a pivotal role in various fields, including machine learning, data mining, and pattern recognition. In our study, our objective for using classification is to maximize the predictive power of classifying classes based on their predictors(Electrical Properties). This literature review delves into the comparative analysis of two prominent classification methods[1][2]: Multinomial Logistic Regression (MLR) and Random Forests (RFs). MLR, an extension of binary logistic regression designed to handle scenarios with multiple classes, boasts strengths in interpretability, statistical rigor, and computational efficiency. However, it grapples with the assumption of linearity, demanding meticulous feature engineering, and susceptibility to overfitting in high-dimensional data. On the other hand, RFs, as ensemble learning methods, harness the collective power of decision trees to enhance classification accuracy. Their strengths lie in handling non-linear relationships, robustness to noise, and the provision of feature importance scores. Yet, RFs pose challenges in terms of interpretability, computational cost, and potential overfitting[3]. Performance comparisons across various studies reveal that, in general, RFs tend to surpass MLR in accuracy, particularly when faced with datasets featuring complex non-linear relationships. However, the choice between MLR and RFs hinges on the specific characteristics of the dataset and the interpretability requirements of the classification task, highlighting the need for a nuanced selection based on the task's intricacies and objectives.

2.2 Imbalance Class Distribution

Imbalanced class distribution poses a significant challenge in many classification tasks, resulting in unequal representation of data samples across different classes. This imbalance can lead to biased models that favor the majority class, causing poor performance for the minority class and leading to misclassification problems. The issue of class imbalance has become prevalent in various real-world applications, including fraud detection, medical diagnosis, anomaly detection, face recognition, email filtering, and environmental monitoring, among others [4].

Researchers have proposed various methods to address the imbalanced class distribution problem, and these methods can be categorized into three groups: algorithm-level approaches, data-level techniques, and cost-sensitive methods [5]. Algorithm-level approaches involve modifying existing algorithms to consider the significance of positive examples and include techniques such as cost-sensitive learning, threshold-moving, and ensemble learning. Data-level techniques focus on preprocessing steps to rebalance the data distribution, including oversampling, undersampling, and hybrid sampling. Cost-sensitive methods combine both algorithm and data-level approaches by incorporating different misclassification costs for each class during the learning phase [4].

Among the data-level techniques, Synthetic Minority Over-sampling Technique (SMOTE) has emerged as a prominent method [6]. SMOTE works by generating synthetic samples of the minority class by interpolating between existing instances. The algorithm selects a minority class instance and creates a new synthetic instance by randomly selecting one or more of its k-nearest neighbors and interpolating between them. Hence, with 154 instances classified as FR, 16 as S, and 54 as FF, we opted for over-sampling to uphold the dataset's overall size, thereby averting potential information loss associated with downsampling.This approach

effectively increases the number of minority class instances, helping to balance the class distribution and improve classification performance [6].

Overfitting, where a model learns the training data too well, capturing noise or random fluctuations in the data as if they were genuine patterns becomes another concern when dealing with imbalanced class distributions. To address this issue, [7] proposed a two-stage training approach for convolutional neural networks (CNNs) [7]. The first stage involves training the CNN with equal samples from each class to prevent overfitting to the majority class. In the second stage, the model is fine-tuned with the original imbalanced data. This approach helps improve the model's performance on the minority class and prevents overfitting to the majority class [7]. Additionally, Random Forest (RF) has been shown to be an efficient approach for handling the overfitting in the presence of imbalanced class distribution [8].

Random forest is one of the ensemble methods, which involve combining decisions from multiple classifiers, and has also gained popularity in tackling the imbalanced class distribution problem [5]. By training several classifiers and aggregating their outputs, ensemble methods can enhance classification accuracy and robustness.

Addressing the imbalanced class distribution problem is crucial for achieving accurate and reliable classification results in various applications. The development and evaluation of these methods have shown that they outperform traditional approaches, highlighting the importance of effectively handling imbalanced data for successful classification tasks[5].

2.3 Missing Data

Missing data is a common challenge statisticians and researchers face when analyzing datasets [9]. It arises due to various reasons, such as non-response, data collection errors, and attrition, and can significantly impact the validity and reli-

ability of statistical analyses. Ignoring missing data may lead to biased estimates and reduced statistical power. Handling missing data is critical to data analysis to ensure accurate and robust conclusions.

The problem of missing data is prevalent in many research fields, including health, social sciences, economics, and engineering [10]. Addressing missing data involves identifying the type of missingness, which can be categorized into Missing Completely at Random (MCAR), Missing at Random (MAR), and Not Missing at Random (NMAR) [11][12][13]. Distinguishing between these types is essential, as it influences the underlying assumptions of statistical modeling techniques.

Traditional approaches to handling missing data involved editing, deletion, and mean imputation, provided as options in statistical software packages [14, 15]. However, more advanced approaches, such as regression imputation, imputation using the EM(Expectation-Maximization) Algorithm, and Multiple Imputation (MI), have been developed to handle missing data more effectively[16]. Among these methods, multiple imputation has gained prominence. It replaces each missing value with M possible values to create M complete datasets, leading to more robust and accurate analyses [17][12].

Recently, there have been evolving approaches to handling missing values in data, including methods that ignore missing observations, single imputation methods, other imputation methods, likelihood-based methods, hot deck imputation, and indicator methods. Each approach has advantages and disadvantages, and researchers must carefully consider the characteristics of their data and the goals of their analysis to select the most appropriate method. Thus, in our analysis, we used hot deck imputation to handle missing data. Hot deck imputation is one of the several available approaches to address missing values in datasets, and our choice was based on several key considerations. Unlike other imputation methods

that may introduce significant bias, hot deck imputation is designed to minimize data distortion[16]. However, it also has some drawbacks, such as being subjective and arbitrary in selecting the donor unit, ignoring the temporal or spatial correlation of the data, and possibly introducing bias or error in the control charts or other statistical tests. Regardless these drawbacks, it aims to provide a plausible substitute for missing values while preserving the underlying characteristics of the data.

In conclusion, addressing missing data is essential for conducting reliable and informative statistical analyses. Researchers have a range of methods at their disposal to handle missing data, each with its strengths and limitations. By appropriately addressing missing data, statisticians and non-statisticians can ensure the validity and accuracy of their data analysis, as we then showed in Hot Deck Imputation.

2.4 Motoneurons and Motor Units

Motor units are essential components of the neuromuscular system, responsible for generating force and coordinating muscle movements. Understanding their properties and classification is vital for unraveling motor control mechanisms and neuromuscular disorders. Motoneurons within the spinal cord and brainstem act as the "final common pathway" of the nervous system [18]. These specialized nerve cells receive input from various brain regions and integrate these signals to generate action potentials. Each action potential then travels along the motoneuron's axon, ultimately triggering the contraction of muscle fibers.

Motor units, the basic functional units of muscle control, comprise a single motoneuron and all the muscle fibers it innervates. The number of muscle fibers within a motor unit varies significantly, impacting the strength and precision of the contractile force generated. Smaller motor units, with fewer fibers, excel in fine

and delicate movements, like those required for precise finger control. Conversely, larger motor units, housing numerous muscle fibers, unleash powerful contractions for forceful actions, such as lifting heavy objects.

The study of motoneurons and motor units is a crucial step for our understanding of neurological disorders. Many debilitating conditions, including amyotrophic lateral sclerosis (ALS), Parkinson's disease, and stroke, disrupt the intricate function of motoneurons and motor units, leading to muscle weakness, paralysis, and other movement impairments. Deciphering the workings of these cells offers valuable insights into the pathogenesis of such disorders, paving the way for novel therapeutic interventions.

2.4.1 Motor Unit Contractile Properties

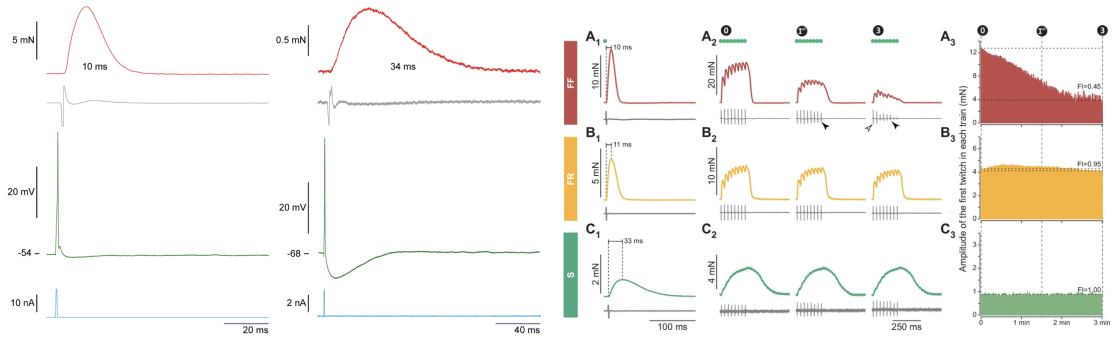
Muscle contraction exhibits remarkable diversity in its properties, allowing us to perform tasks ranging from delicate finger movements to powerful sprints. This diversity arises from the intricate interplay between the nervous system and muscle fibers, primarily through the unique composition of motor units. The unique expression of myosin isoforms, metabolic enzymes, and calcium handling proteins in the muscle fibers lead to the distinction of three major motor unit types: S (slow-twitch), FR (fast-twitch, fatigue-resistant), and FF (fast-twitch, fatigable).

Slow (s) motor units, as their name implies, exhibit slow contraction speeds [19], due to the fact that they primarily express myosin isoform MyHC type I, characterized by low ATPase activity. On the other hand, these muscle fibers have highly developed sarcoplasmic reticulum with a large capacity to store and release calcium and are rich in oxidative enzymes and mitochondria, enabling efficient ATP generation for sustained activity[20]. Consequently, these motor units are mainly used for maintaining posture and performing low-intensity tasks requiring

endurance [18].

On the other hand, fast-Fatigable (FF) motor units exhibit the fastest contraction speed and low fatigue resistance[19]. Their muscle fibers express the myosin isoform MyHC type IIa, characterized by the highest ATPase activity and fastest cross-bridge cycling, leading to rapid contraction. However, they primarily rely on glycolysis for rapid ATP generation, leading to quick fatigue [20]. They are, therefore, mainly used to generate high force for explosive movements (e.g., sprinting, jumping)[18].

Fast, fatigue-resistant (FR) motor units have an intermediate profile with fast contraction speeds but good fatigue resistance[19]. Their muscle fibers express the myosin isoform MyHC type IIa, exhibiting faster contraction speed than S fibers but slower than FF fibers, and a balanced mix of oxidative and glycolytic enzymes, allowing both sustained and rapid bursts of activity[20]. They are essential for the production of rapid movement with moderate force and moderate endurance (e.g., walking, swimming) [18].



(a) (b)

Figure 1: Twitch force elicited by injecting a series of short pulses of current - The figure[21] illustrates twitch and motor unit action potential (MUAP) of an FF motor unit, unfused tetani during a fatigue test, and the decline in twitch amplitude over time. Similar arrangements are shown for FR and S motor units, with traces representing averages of 5-10 sweeps.

Variable Name	Description
‘condSpeed‘	measure of the conduction velocity of the antidromic spike along the axon of the recorded motoneuron
‘rheobase‘	the minimum amount of current to unject in the motoneuron to elicit an action potential
‘rmp‘	the membrane potential of the cell in absence of any stimulation
‘spikeOvershoot‘	the membrane potential reached at the peak of the spike
‘spikeHeight‘	the height of the spike, from ‘rmp‘ to ‘spikeOvershoot‘
‘spikeWidth‘	the width of the spike at half it total height
‘ahpAmp‘	the amplitude of the afterhyperpolarization (AHP) following each spike
‘ahpT2P‘	the time between the foot of the spike and the trough of the AHP
‘ahpHalfRelax‘	the time it takes for the AHP to relax to half its amplitude
‘RinPeak‘	the value of the input resistance of the cell measured at the peak of the response
‘RinPlat‘	the value of the input resistance of the cell measured at the plateau of the response
‘SR‘	the sag ratio, calculated as the ratio between ‘RinPlat‘ and ‘RinPeak‘
‘GinPeak‘	the input conductance of the cell (inverse of ‘RinPeak‘)
‘tauM‘	the membrane time constant, a measure of how fast the membrane potential relaxes
‘L‘	the electrotonic length of the neuron
‘Ion‘	the intensity of the current required to elicit firing on the ascending phase of a triangular ramp of current
‘Ioff‘	the intensity of current at the time of the last action potential on the descending phase of a triangular ramp of current
‘$\triangle I$‘	the difference between ‘Ioff‘ and ‘Ion‘
‘$\triangle F$‘	the difference between the instantaneous frequency of the last inter spike interval and the first
‘ascSlope‘	the primary range slope measured on the response to the ascending phase of a triangular ramp of current
‘descSlope‘	the primary range slope measured on the response to the descending phase of a triangular ramp of current
‘Vth‘	the voltage threshold for spiking, measured on the first action potential of the triangular ramp of current

Table 1: Description of the electrical properties measured in the experiments from the Manuel lab. Full details can be found in [24]. Due to the unpredictable duration of the recordings, not all of the measures can be obtained in every recorded motoneurons

2.4.2 Motoneuron Electrical Properties

Motoneurons exhibit a number of electrical properties that have been extensively studied over the past fifty years in human and animal models[18],[22], including recently in adult mice[23][24][21]. These properties will form the basis of our classification and are described in detail in [24].

2.4.3 Motor Unit classification:

Motor units were classified based on the profile of the twitch force elicited by injecting a series of short pulses of current in the soma of the recorded motoneurons, thereby triggering single action potentials. The contracting time (‘twCT‘) is the time between the onset of the contraction and the time of the peak of the twitch. The twitch amplitude (‘twAmp‘) is the difference between the baseline and the peak of the twitch. As previously described [21], motor units with a contraction time $\geq$ 20ms were classified as S-type. Motor units with a contraction time $<$ 20ms were classified as fast-twitch. Among those, motor units with a twitch amplitude

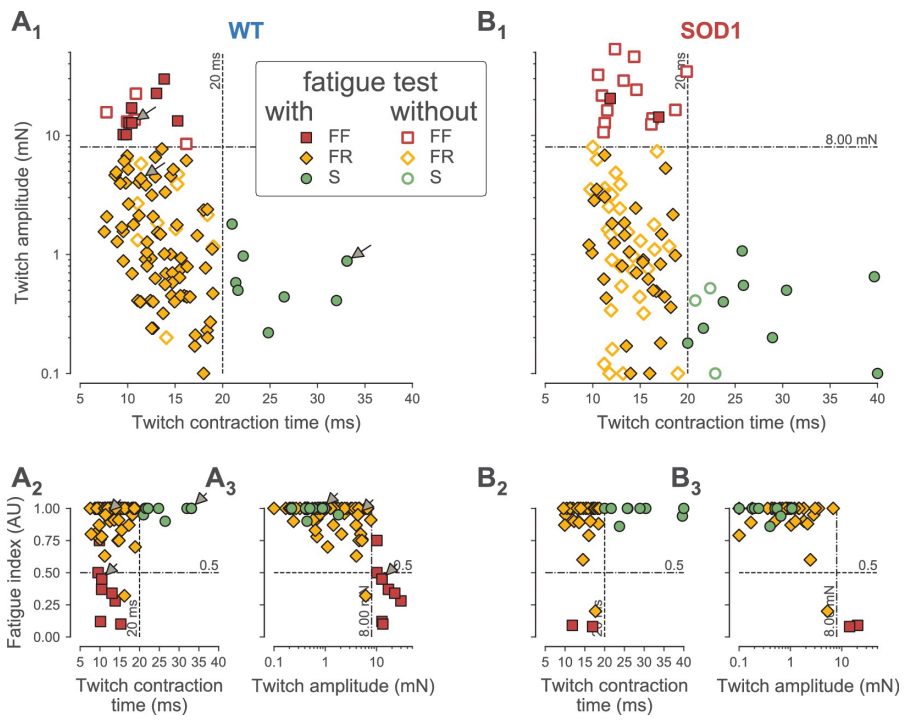


Figure 2: Contractile properties of motor units in (A) wild-type (WT) and (B) SOD1G93A mice. (A1) Distribution of twitch amplitude (logarithmic scale) vs. twitch contraction time, with arrows indicating specific motor units. Vertical dashed line at 20 ms separates fast and slow-contracting units. Horizontal dash-dotted line at 8 mN distinguishes fatigue-resistant (FR) and fatigable (FF) units. Filled markers denote units with fatigue index measurement; empty markers lack fatigue measurement. (A2) Fatigue Index vs. twitch contraction time, with a limit at 0.5 for categorization. (A3) Fatigue Index vs. twitch amplitude. Similar organization in (B) for SOD1G93A motor units. [21]

$\geq$ 8mN were considered to be FF, and those with a twitch amplitude $<$ 8mN were classified as FR. Finally, the work of [21] identified two subpopulations in the FR group. We, therefore also considered the possibility of splitting the FR motor units into two groups, the small FR and large FR motor units, based on a cutoff value of 1.5mN.

List of References

[1] G. James, D. Witten, T. Hastie, R. Tibshirani, *et al.*, *An introduction to statistical learning*. Springer, 2013, vol. 112.

[2] D. Witten and G. James, *An introduction to statistical learning with applica-*

tions in R. springer publication, 2013.

[3] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.

[4] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 463–484, 2011.

[5] H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

[6] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[7] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural networks*, vol. 106, pp. 249–259, 2018.

[8] C. Drummond, R. C. Holte, *et al.*, "C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling," in *Workshop on learning from imbalanced datasets II*, vol. 11, 2003, pp. 1–8.

[9] P. D. Allison, *Missing data*. Sage publications, 2001.

[10] D. A. Bennett, "How can i deal with missing data in my study?" *Australian and New Zealand journal of public health*, vol. 25, no. 5, pp. 464–469, 2001.

[11] D. B. Rubin, "Inference and missing data," *Biometrika*, vol. 63, no. 3, pp. 581–592, 1976.

[12] J. L. Schafer and J. W. Graham, "Missing data: our view of the state of the art." *Psychological methods*, vol. 7, no. 2, p. 147, 2002.

[13] J. Scheffer, "Dealing with missing data," 2002.

[14] S. Van Buuren and K. Groothuis-Oudshoorn, "mice: Multivariate imputation by chained equations in r," *Journal of statistical software*, vol. 45, pp. 1–67, 2011.

[15] C. Enders, "Applied missing data analysis: Guilford press," *New York*, 2010.

[16] R. J. Little and D. B. Rubin, *Statistical analysis with missing data*. John Wiley & Sons, 2019, vol. 793.

[17] R. J. Little, "A test of missing completely at random for multivariate data with missing values," *Journal of the American statistical Association*, vol. 83, no. 404, pp. 1198–1202, 1988.

[18] C. Heckman and R. M. Enoka, "Motor unit," *Comprehensive physiology*, vol. 2, no. 4, pp. 2629–2682, 2012.

[19] R. Burke, D. Levine, P. Tsairis, and F. Zajac Iii, "Physiological types and histochemical profiles in motor units of the cat gastrocnemius," *The Journal of physiology*, vol. 234, no. 3, pp. 723–748, 1973.

[20] S. Schiaffino and C. Reggiani, "Fiber types in mammalian skeletal muscles," *Physiological reviews*, vol. 91, no. 4, pp. 1447–1531, 2011.

[21] M. d. L. Martinez-Silva, R. D. Imhoff-Manuel, A. Sharma, C. Heckman, N. A. Shneider, F. Roselli, D. Zytnicki, and M. Manuel, "Hypoexcitability precedes denervation in the large fast-contracting motor units in two unrelated mouse models of als," *Elife*, vol. 7, p. e30955, 2018.

[22] M. Manuel and D. Zytnicki, "Molecular and electrophysiological properties of mouse motoneuron and motor unit subtypes," *Current opinion in physiology*, vol. 8, pp. 23–29, 2019.

[23] M. Manuel, M. Chardon, V. Tysseling, and C. Heckman, "Scaling of motor output, from mouse to humans," *Physiology*, vol. 34, no. 1, pp. 5–13, 2019.

[24] M. Manuel, C. Iglesias, M. Donnet, F. Leroy, C. Heckman, and D. Zytnicki, "Fast kinetics, high-frequency oscillations, and subprimary firing range in adult mouse spinal motoneurons," *Journal of Neuroscience*, vol. 29, no. 36, pp. 11 246–11 256, 2009.

CHAPTER 3

Methodology

3.1 Classification Models

The various classification methods that we will use to solve the to predict the physiological types based on the electrical properties are as follows:

3.1.1 Multinomial Logistic Model

Multinomial Logistic Regression (MLM) is implemented as an extension of binary logistic regression, accommodating outcomes with multiple levels. The outcomes in this study are in a multilevel scale. The outcome variable Y encompasses a categories, where category S coded as 0, FF as 1 and FR as 3 with p predictor variables. Unlike binary cases, the multinomial model establishes a baseline selecting $Y = 0$ and forms $a1$ logit functions expressing the natural logarithm of the odds as:

$$\begin{aligned} g_i(x) &= \ln\left[\frac{Pr(Y = i|x)}{Pr(Y = i|x)}\right] \\ &= \beta_{i,0} + \beta_{i,1}x_1 + \ldots + \beta_{i,p}x_p \end{aligned}$$

In the equation above, x represents a vector of p covariates, denoted as $x_1, x_2, ..., x_p$, also the parameters included, $\beta_{i,0}$ as the intercept and $\beta_{i,1}, ..., \beta_{i,p}$ as the coefficients corresponding to the covariates. Here, i takes values from 2 to a. Therefore, the conditional probabilities of each category are determined based on the observed values of the covariates, thus given as:

$$\begin{aligned} \pi_i(x) &= Pr(Y = i|x) \\ &= \frac{e^{g_i(x)}}{\sum_{k=1}^{a} e^{g_k(x)}} \end{aligned}$$

For each i where i ranges from 1 to a and $g_i(x) = 0$, we applied the maximum likelihood estimation method is used to fit the model and estimate the coefficients,

using an R package[1]. To formulate the likelihood function, we used the multilevel response variables Y_1 to Y_a. These variables are encoded such that if $Y = i$, then $Y_i = 1$ and $Y_{s\neq i} = 0$, where $i, s = 1, 2, \ldots, a$. Consequently, regardless of the value Y assumes, the sum of these a variables consistently equals 1. Assuming n independent observations, the likelihood function is derived accordingly as:

$$L(\beta) = \prod_{k=1}^{a} [\pi_1(x_k)^{y_{1k}} \pi_2(x_k)^{y_{2k}} \ldots \pi_a(x_k)^{y_{ak}}]$$

Thus,$k = 1, 2, \ldots, n$, and applying the natural logarithm, we consider the condition $\sum_{i=1}^{a} Y_{ik} = 1$ for each k, the logarithm of the likelihood function is derived as:

$$\mathcal{L}(\beta) = X \left[\sum_{i=2}^{a} y_{ik} g_i(x_k) + \ldots + \sum_{i=2}^{a} y_{ik} g_i(x_k) - \ln \left(1 + e^{g_2(x_k)} + \ldots + e^{g_a(x_k)}\right) \right]$$

By computing the partial derivatives of the likelihood function $\mathcal{L}(\beta)$ with respect to each β_{ij}, where i ranges from 1 to n and j ranges from 1 to p,hence the maximum likelihood estimators are derived. Before drawing inferences from the model, it is crucial to evaluate both the overall fit and the contribution of each individual observation to the fit. This assessment becomes more intricate in cases with multiple outcome levels of the motor units based on the 8 explanatory variables. Nonetheless, to estimate $\mathbf{g_i}(\mathbf{x})$, the Multinomial logistic regression assumes that the choice of membership in one category is not related to the choice or membership of another category (i.e., the dependent variable), with some basic assumptions to be met as follows[2]:

1. Observations (X_i, Y_i) are independent.
2. Outcome variable categories Y are mutually exclusive and exhaustive.
3. Independence of errors.
4. Absence of multicollinearity.
5. Lack of strongly influential outliers.

3.1.2 Random Forest

The Random Forest, an ensemble of trees governed by random vectors, enhances classification accuracy by collectively voting for the most prevalent class at input x [3]. This versatile mechanism accommodates both classification and regression tasks. Our focus in this introduction is on classification, presenting a concise algorithm within the non-parametric regression framework to clearly understand the Random Forest's application in regression analysis. Addressing the multilevel classification problem discussed in [4], our random response Y takes values of $\{0, 1, 3\}$. Given X, one must guess the value of Y. A classifier, represented by $\mathbf{m}_n$, is consistent if its probability of error $\mathbf{L}(\mathbf{m_n})$ converges to $\mathbf{L}^*$ as $n \to \infty$ [5].

The Random Forest classifier, obtained via a majority vote among classification trees, classifies a randomized tree based on leaf representation. The equation can be found in [5] [6]. We start with the Ensemble of Trees using the algorithm-based approach, specifically the Classification and Regression Tree (CART) algorithm. The Random Forest comprises M trees $\{T_1(X), \ldots, T_M(X)\}$, where $X = \{x_1, \ldots, x_p\}$ is a p-dimensional vector of electrical properties associated with a system. The ensemble produces M outputs $\{\hat{Y}_1 = T_1(X), \ldots, \hat{Y}_M = T_M(X)\}$, aggregated to produce the final prediction $\hat{Y}$. For classification, $\hat{Y}$ is the majority vote.

Training Procedure

Given a dataset D of n systems for training, $D = \{(X_1, Y_1), \ldots, (X_n, Y_n)\}$, where $X_i, i = 1, \ldots, n$, is a vector of electrical properties, and Y_i is either the class label or activity of interest, the training algorithm proceeds as follows:

1. Draw a bootstrap sample from the training data of n systems (randomly sampled with replacement).

2. For each bootstrap sample, grow a tree with a modification: choose the best split among a randomly selected subset of m_{try} descriptors at each node. Here, m_{try} is a tuning parameter.

3. Repeat the above steps until a sufficiently large number M of trees are grown.

The impact of M is crucial, representing the number of trees in the ensemble. The choice of m_{try} controls randomness, influencing tree diversity. When $m_{\text{try}} = p$, the algorithm is equivalent to Bagging. The tree-growing algorithm is based on CART, with alternative methods also considered.

Though the Random Forest (RF) algorithm is computationally efficient, especially with many descriptors, it tests a subset of descriptors (m_{try}) at each node, resulting in faster searches and eliminating the need for pruning. RFs can be trained faster than a single decision tree in cases with numerous descriptors. Other ensemble methods like Bagging and Boosting further enhance performance. Empirically, Boosting and Random Forest consistently outperform Bagging. Ideally, assessing a prediction algorithm demands a substantial independent test dataset distinct from the training data. Due to data limitations, we frequently resort to cross-validation, a computationally demanding alternative. However, Random Forest offers a solution by conducting a form of cross-validation during training using Out-Of-Bag (OOB) samples[7]. Our analysis compares OOB performance estimation with k-fold cross-validation, revealing reasonably good agreement.

3.2 Oversampling

Imbalanced classification has been a significant concern. To tackle class imbalance in our response variable Y (Table 2), particularly in the context of classification problems, we applied the oversampling method to all classes using the `oversample_classes` function. This ensured uniform representation of classes

(at levels 75 and 200 for restricted imputed data) across the dataset with our `oversample_classes` function.

Table 2: Class distribution of Y in our original and subset(restricted data)

Class	Original Data	Subset Data
0	16	5
1	154	51
3	46	15

This oversampling procedure involves considering Y as the categorical response variable with possible classes, where `possible_classes` is a vector containing the unique classes in Y, and N is the desired number of samples after oversampling. For each class `class_Y` in `possible_classes`, we performed the following steps:

1. Identify the indices `(ind)` of instances in Y belonging to the current class `class_Y`.

2. If the number of instances `length(ind)` for `class_Y` is greater than or equal to 2, randomly sample N instances with replacement from `(ind)` and store them in `s`.

3. If there is only one instance for `class_Y`, replicate the single index N times to create `s`.

4. If there are no instances for `class_Y`, set `s` to an empty vector(`s` is our sampling index).

5. Append `s` to the oversampled data `(ov)`.

Our function iterates through all possible classes, creating an oversampled dataset that addresses the imbalance by duplicating or replicating instances from minority classes. This samples with replacement from the dominating classes where for the

restricted(subset) data we have 75 and 200 for the imputed data. This ensures a more balanced distribution of classes in the restricted(subset data)and imputed data.

3.3 Imputation

Due to the unpredictable duration of the recordings in the experiments performed by the Manuel lab, the original dataset contains a large number of missing data. Missing data is a common problem when collecting data for motoneuron electric characteristics, thus posing significant challenges for data analysis and interpretation(Test of the Missing Data assumed MAR). This is a common issue in biological, clinical and social research [**?**]. Imputation is a powerful and versatile technique to address these issues, providing estimates for missing values based on observed data. Here, we outline the hot-deck imputation techniques, focusing on the [5]nearest-neighbor algorithm approach. Using our data, (1) we randomly select a sample of n out of the N units, (2) we identify r responding units among the n sampled units, where $r < n$, (3) label the first n units as sampled and the first r units as respondents, and (4) finally, standardize or normalize relevant covariates for all units (both responding and missing). Also, while using the nearest neighbor hot deck, one defines a distance metric $d(i, j)$ to measure the "closeness" between units i and j based on standardized covariates in the data[8]. Possible metrics include:

1. Equal Probability: $d(i, j) = 0$ if i and j are in the same adjustment cell(measures of dissimilarity or distance between two entities i and j), else 1.

2. Maximum Deviation: $d(i, j) = \max_k |x_{ik} - x_{jk}|$, where x_{ik} and x_{jk} are covariates of units i and j.

3. Predictive Mean: $d(i,j) = [\hat{y}(x_i) - \hat{y}(x_j)]^2$, where $\hat{y}(x)$ is the predicted value of Y based on covariates x.

By doing this, we consider that for each unit i with a missing value y_i, there is a set of k nearest neighbors j such that $d(i,j)$ is minimized, where k is a pre-defined number. The imputation works as follows:

1. Randomly select one value y_j from the observed k nearest neighbors.

2. Impute the missing value y_i with the randomly chosen y_j.(We Repeat steps 2 and 3 for all units with missing values).

Though the methodology assumes the availability of relevant covariates for both responding and missing units, the choice of k can significantly impact the accuracy of the imputation. Hence, we experiment with different values and choose the one that optimizes the chosen performance metric.

3.4 Model Selection

During our meticulous model selection process, our primary emphasis was on comparing the performance of the Multinomial Logistic Model (MLM) and Random Forest, with a focus on the dual criteria of accuracy and interpretability. We rigorously evaluated these models, both with and without oversampling, employing 5-fold cross-validation to ensure a robust estimation of performance on unseen data, surpassing the limitations of training data evaluation alone. Our evaluation criteria included metrics such as accuracy, precision for each class, and overall model performance, aligning with the methodology outlined by[9].

As our research advanced, interpretability emerged as a pivotal factor. MLM, with its interpretable coefficients, provided a lucid understanding of the intricate relationships between electrical properties and physiological types. Conversely,

Random Forest, being an ensemble method, lacked direct interpretability but contributed valuable insights into critical features for classification, as highlighted by Song et al. (2013) [10]. In the final analysis, we opted for MLM, balancing performance, interpretability, and complexity. This choice rendered MLM the most suitable model for accurately classifying physiological motoneuron types in mice [11].

List of References

[1] B. Ripley, W. Venables, and M. B. Ripley, "Package 'nnet'," *R package version*, vol. 7, no. 3-12, p. 700, 2016.

[2] J. C. Stoltzfus, "Logistic regression: a brief primer," *Academic emergency medicine*, vol. 18, no. 10, pp. 1099–1104, 2011.

[3] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.

[4] R. Diaz-Uriarte and S. Alvarez de Andres, "Gene selection and classification of microarray data using random forest," *BMC bioinformatics*, vol. 7, pp. 1–13, 2006.

[5] G. Biau and E. Scornet, "A random forest guided tour," *Test*, vol. 25, pp. 197–227, 2016.

[6] R. Xu, "Improvements to random forest methodology," Ph.D. dissertation, Iowa State University, 2013.

[7] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan, and B. P. Feuston, "Random forest: a classification and regression tool for compound classification and qsar modeling," *Journal of chemical information and computer sciences*, vol. 43, no. 6, pp. 1947–1958, 2003.

[8] R. J. Little and D. B. Rubin, *Statistical analysis with missing data*. John Wiley & Sons, 2019, vol. 793.

[9] T. Pranckevivcius and V. Marcinkevivcius, "Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification," *Baltic Journal of Modern Computing*, vol. 5, no. 2, p. 221, 2017.

[10] L. Song, P. Langfelder, and S. Horvath, "Random generalized linear model: a highly accurate and interpretable ensemble predictor," *BMC bioinformatics*, vol. 14, no. 1, pp. 1–22, 2013.

[11] J. A. Martin-Baos, R. Garcia-Rodenas, and L. Rodriguez-Benitez, "Revisiting kernel logistic regression under the random utility models perspective. an interpretable machine-learning approach," *Transportation Letters*, vol. 13, no. 3, pp. 151–162, 2021.

CHAPTER 4

Findings

4.1 Data Transformation

The Manuel lab provided the data used in this study, a subset of which was previously published [1]. Intracellular recordings of Triceps Surae motoneurons were collected from deeply anesthetized mice. Simultaneously, the force produced by each individual recorded motoneuron was measured at the tendon by a highly sensitive force transducer. Initially, the complete data had 37 variables with 216 observations. Upon analysis of this dataset, two issues became readily apparent. First, the distribution of the three class of motor units was highly unbalanced (**??**). In addition a large number of observations contained one or more missing data (3). The transformation of data from its original form to the imputed dataset involved several steps, addressing issues of class imbalance and missing data. A concise overview of the data is presented in Table 4.The subset data, derived using an R subset function, was the smallest due to a process involving variable importance testing, correlation matrix examination(Table 5), feature averaging (for dimensionality reduction), and addressing missing data percentages. Thirteen variables (rheobase, spikeWidth, ahpAmp, ahpT2P, ahpHalfRelax, RinPeak, RinPlat, SR, GinPeak, tauM, tau1, L, Ion) were considered for the subset data. Also,in the

Factors	Original Data	Subset Data	Imputed Data
Observations	216	71	216
Variables	37	13	21

Table 3: Table illustrating the number of observations and variables in each Data

imputed data, a LittleMCAR test revealed a p-value of 0 ¡ α (0.05), indicating not MCAR (Missing Completely at Random) data. Thus, assuming Missing at Ran-

Class	Original Data	Subset Data	Imputed Data
0	16	5	16
1	154	51	154
3	46	15	46
Total	216	71	216

Table 4: Class distribution of Y in our original, subset(restricted) and Imputed data

Correlation	rheobase	spikeWidth	ahpAmp	ahpT2P	ahpHalfRelax	RinPeak	RinPlat	SR	GinPeak	tauM	tau1	L	Ion	twAmp	twCT	PhysioType3	PhysioType4
rheobase	1.00000000	-0.12153943	-0.258033287	-0.08713017	-0.23861033	-0.6530944	-0.61958571	-0.324572737	0.70721579	-0.41626045	-0.36975672	-0.03960072	0.76084868	0.61430546	-0.31634144	0.59667257	0.70698345
spikeWidth	-0.12153943	1.00000000	0.066695310	-0.11224230	0.08663882	0.2202368	0.20077968	0.064562705	-0.20562712	-0.02762926	-0.10175621	-0.13938448	-0.10145275	-0.20723782	-0.09767333	-0.16056764	-0.20907018
ahpAmp	-0.25803329	0.06669531	1.000000000	-0.34985492	0.42341743	0.4216640	0.41741395	0.236479895	-0.276768072	0.182705122	0.087955158	-0.086079197	-0.283368589	-0.002514237	0.377768349	-0.141160426	-0.205676201
ahpT2P	-0.08713017	-0.11224230	-0.349854919	1.00000000	0.20004254	0.2440232	0.19495725	-0.109327721	-0.17829806	0.38909454	0.23620380	-0.26728482	-0.10332410	-0.26518705	0.25058188	-0.16568480	-0.29686024
ahpHalfRelax	-0.23861033	0.08663882	0.423417428	0.20004254	1.00000000	0.3959124	0.37866902	0.145524300	-0.25704397	0.34844699	0.06963227	-0.37422333	-0.30596994	-0.12924786	0.52991892	-0.22663804	-0.33985856
RinPeak	-0.65309437	0.22023679	0.421664039	0.24402323	0.39591245	1.0000000	0.98309268	0.533482597	-0.8385821	0.6324045	0.51235380	-0.128045197	-0.7623673	-0.4999449	0.54936593	-0.48841957	-0.6784287
RinPlat	-0.61958571	0.20077968	0.417413946	0.19495725	0.37866902	0.9830927	1.00000000	0.645461735	-0.80642283	0.59801183	0.50184968	-0.09927849	-0.73610461	-0.46711003	0.52213568	-0.46338037	-0.63488926
SR	-0.32437274	0.06456271	0.236479895	-0.10932772	0.14552430	0.5334826	0.64546173	1.000000000	-0.580093229	0.248484570	0.259048792	-0.008853778	-0.529644273	-0.331237603	0.244590344	-0.403121420	-0.424302007
GinPeak	0.70721579	-0.20562712	-0.276768072	-0.17829806	-0.25704397	-0.8385821	-0.80642283	-0.580093229	1.0000000	-0.53309227	-0.43365873	0.117141599	0.8850692	0.651244348	-0.39862746	0.64966866	0.7774942
tauM	-0.41626045	-0.02762926	0.182705122	0.38909454	0.34844699	0.6324045	0.59801183	0.248484570	-0.53309227	1.00000000	0.81163001	-0.195045819	-0.50372074	-0.32890278	0.48510336	-0.31678702	-0.49178715
tau1	-0.36975672	-0.10175621	0.087955158	0.23620380	0.06963227	0.5123538	0.50184968	0.259048792	-0.43365873	0.81163001	1.00000000	0.379497132	-0.3777543	-0.23827541	0.35133400	-0.25745724	-0.3872452
L	-0.03960072	-0.13938448	-0.086079197	-0.26728482	-0.37422333	-0.1280452	-0.09927849	-0.008853778	0.158235652	0.100280956	-0.15165657	1.0000000	0.1582357	0.100280956	-0.15165657	0.03771466	0.1074721
Ion	0.76084868	-0.10145275	-0.283368589	-0.10332410	-0.30596994	-0.7623673	-0.73610461	-0.529644273	0.8850692	-0.50372074	-0.37775427	0.158235652	1.0000000	0.646964094	-0.40036505	0.59368380	0.7503886
twAmp	0.61430546	-0.20723782	-0.002514237	-0.26518705	-0.12924786	-0.4999449	-0.46711003	-0.331237603	0.651244348	-0.32890278	-0.23827541	0.100280956	0.6469641	1.000000000	-0.20676622	0.80934956	0.7806772
twCT	-0.31634144	-0.09767333	0.377768349	0.25058188	0.52991892	0.5493659	0.52213568	0.244590344	-0.39862746	0.48510336	0.35133400	-0.151656569	-0.4003650	-0.20676622	1.00000000	-0.36917959	-0.5365879
PhysioType3	0.59667257	-0.16056764	-0.141160426	-0.16568480	-0.22663804	-0.4884196	-0.46338037	-0.403121420	0.64966866	-0.31678702	-0.25745724	0.037714661	0.5936838	0.809349562	-0.36917959	1.00000000	0.8482251
PhysioType4	0.70698345	-0.20907018	-0.205676201	-0.29686024	-0.33985856	-0.6784287	-0.63488926	-0.424302007	0.7774942	-0.49178715	-0.38724522	0.107472100	0.7503886	0.780677174	-0.53658792	0.84822510	1.0000000

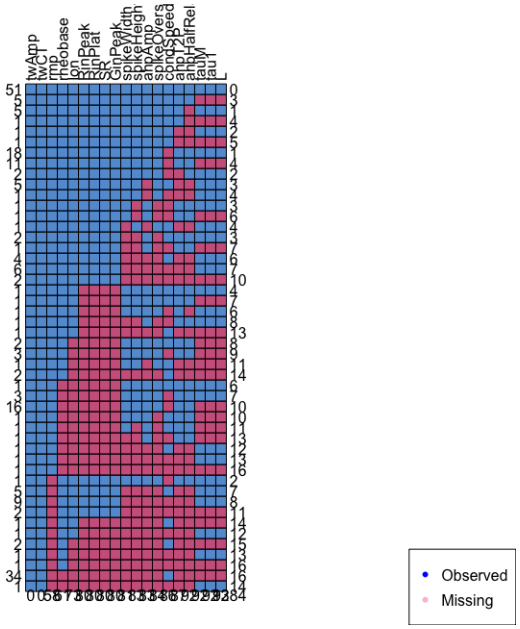
Table 5: Correlation Matrix for 13 variables (rheobase, spikeWidth, ahpAmp, ahpT2P, ahpHalfRelax, RinPeak, RinPlat, SR, GinPeak, tauM, tau1, L, Ion)

dom (MAR), oversampling was performed to equalize class counts (75 per class for restricted and 200 for imputed data). Subsequently, variables with missing data percentages exceeding 45% were removed, since this can result in bias imputation, create noise, and to preserve the data quality and reliability[2],thus imputation was carried out using the Hot Deck Imputation library's impute. NNHD package in R.

The missing data percentages in the original data are depicted in Figure 3A, while Figure 3B illustrates the missing data pattern after removing variables with missing observation percentages exceeding 45%. Sixteen out of 37 variables were removed due to high missing data percentages, minimizing noise, overfitting, and biased estimates. The variables were also averaged based on specified thresholds for CT (Contraction Time) and Amp (Amplitude) to extracting relevant information and enhancing the analysis of physiological data. By carefully choosing the thresholds and understanding the underlying assumptions, we effectively capture subtle events, reduce noise, and gain valuable insights into the phenomena. The

Variable	Percentage_Missing
condSpeed	39.81481
antiSpikeThreshold	87.03704
rheobase	28.24074
rmp	26.85185
spikeOvershoot	38.88889
spikeHeight	38.42593
spikeWidth	37.50000
ahpAmp	38.42593
ahpT2P	40.27778
ahpHalfRelax	42.59259
ahpDur	64.35185
ahpTau	87.03704
RinPeak	37.03704
RinPlat	37.03704
SR	37.03704
GinPeak	37.03704
tauM	42.59259
tau1	42.59259
L	42.59259
Ion	33.79630
Ioff	66.20370
deltaI	66.20370
deltaF	67.12963
ascSlope	71.75926
descSlope	72.22222
globSlope	72.22222
Vth	66.66667
RMP_CCRamp	68.51852
transSPRPR	76.38889
deltaSPR	85.64815
deltaSPRrel	87.96296
freqSPRPR	81.48148
ΔVth	97.22222
twAmp	0.00000
twCT	0.00000
PhysioType3	0.00000
PhysioType4	0.00000

(a) Missing Percentage



(b) Missing Data Pattern

Figure 3: A display of the Missing Data Percentages of the Original Data in **A**, while after considering the deletion of percentages $> 45\%$, **B** displays its missing pattern

averaging of features for the restricted and imputed data is presented in Figure 4, showcasing the impact of the applied thresholds (CT 20mNs and Amp 8mNs) on the dataset. Applying these techniques allowed for capturing more pertinent

thrCT	thrAmp	feature	class	mean	std
20	8	rheobase	0	0.4714555	1.49344608
20	8	ahpAmp	0	4.0272000	2.43341355
20	8	ahpHalfRelax	0	14.9318667	3.73967685
20	8	RinPeak	0	5.1966667	0.81077429
20	8	Ion	0	2.0699368	0.74007545
20	8	SR	0	0.8191575	0.07531488
20	8	spikeWidth	0	0.2933333	0.04742942
20	8	L	0	1.3637333	0.19578514
20	8	rheobase	1	6.9736168	3.93044229
20	8	ahpAmp	1	1.8437333	1.00215818
20	8	ahpHalfRelax	1	9.4726667	2.71399144
20	8	RinPeak	1	2.9872000	1.93188716
20	8	Ion	1	7.7469537	4.43559879
20	8	SR	1	0.7579019	0.13061795
20	8	spikeWidth	1	0.2782667	0.05962435
20	8	L	1	1.4909333	0.18953837
20	8	rheobase	3	14.1940549	5.22547304
20	8	ahpAmp	3	1.8854667	1.06293486
20	8	ahpHalfRelax	3	9.5730667	2.58811210
20	8	RinPeak	3	1.3737333	0.32704150
20	8	Ion	3	13.4420184	3.72783529
20	8	SR	3	0.6509879	0.13910493
20	8	spikeWidth	3	0.2514667	0.03501711
20	8	L	3	1.4308000	0.17447806

(a) Restricted

thrCT	thrAmp	feature	class	mean	std
20	8	rheobase	0	2.8062935	2.75762943
20	8	ahpAmp	0	4.0869000	2.60791458
20	8	ahpHalfRelax	0	14.9155500	3.47125869
20	8	RinPeak	0	5.0640500	1.19941782
20	8	Ion	0	2.7837788	1.70850380
20	8	SR	0	0.7869069	0.11062452
20	8	spikeWidth	0	0.3332000	0.07487182
20	8	L	0	1.3331500	0.16028109
20	8	rheobase	1	6.8853619	3.90194750
20	8	ahpAmp	1	1.8371500	1.05077161
20	8	ahpHalfRelax	1	9.7079000	2.61917007
20	8	RinPeak	1	2.8001500	1.40868144
20	8	Ion	1	7.0748103	3.95731535
20	8	SR	1	0.7337700	0.12060473
20	8	spikeWidth	1	0.2879500	0.06260105
20	8	L	1	1.4490500	0.20285053
20	8	rheobase	3	11.8004193	4.81389757
20	8	ahpAmp	3	2.1147000	1.21021486
20	8	ahpHalfRelax	3	9.0753500	1.77729026
20	8	RinPeak	3	1.6239500	0.39012548
20	8	Ion	3	11.0325221	3.12500273
20	8	SR	3	0.6861493	0.11623235
20	8	spikeWidth	3	0.2834000	0.05697959
20	8	L	3	1.4225000	0.14142402

(b) Imputed

Figure 4: Averaging the Features of Restricted Data in **(a)** and **(b)** Imputed Data for the set threshold of CT 2OmNs and Amp 8mNs

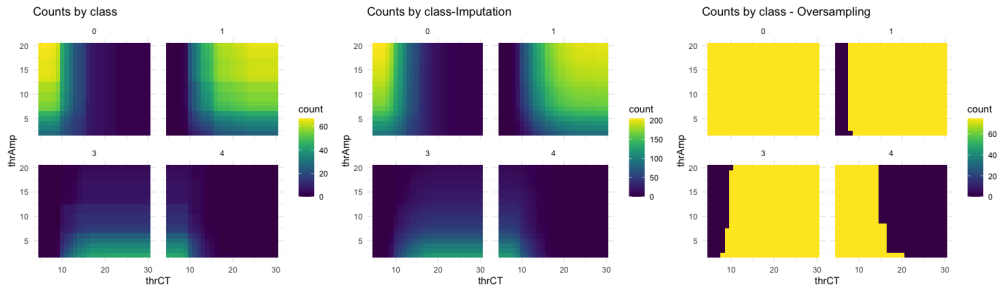
data points related to specific physiological phenomena, simplifying subsequent analyses. For example, the variable "rheobase" exhibited a lower mean in the restricted data compared to the imputed data and also in both data, we realized that its standard deviation gets higher in most classes, thus indicating that the data points spread further away from the mean.

4.2 Outcome

Addressing the challenge of imbalanced classes through oversampling appeared to balance the class distribution, with counts generated independently of the chosen model. However, a closer examination of the baseline revealed the potential existence of a fourth class within the specified range. This raised whether this potential class significantly influenced overall accuracy or accuracy by class in the model.

Counts Restricted Data & Imputed Data

This shows how the counts are distributed on the respective dataset and the methods (oversampling) employed. Though there is some similarity on the heatmap for the Restricted and Imputed Data Figure 5, the sizes are different.



(a) Restricted at 71 (b) Imputation at 216 (c) Oversampled at 75

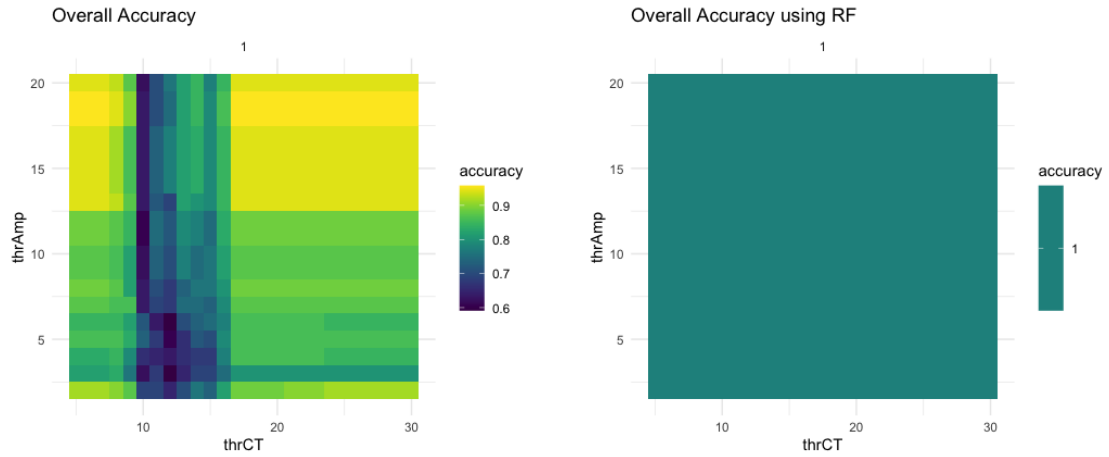
Figure 5: This heatmap show how the class by counts are distributed by Restricted on the Subset, Oversampling, Imputation.The analysis involves a sequential threshold range for Contraction Time (CT) from 5mNs to 30mNs and Amplitude (Amp) from 2mNs to 20mNs, each incremented by 1mNs.

4.2.1 Restricted Data - MLM vs Random Forests

In Figure 6, a heatmap illustrates the overall accuracy for Multinomial Logistic Model (MLM) and Random Forest (RF) at various threshold values for Contraction Time (CT) and Amplitude (Amp). RF exhibited a constant accuracy of 1.0 with O0B rate of 0.0278, which is low, indicating that the model predicts well on unseen data. Due to its sensitivity to model performance during cross-validation. Figure 6 depicts class-specific accuracy, revealing instances where certain classes, especially 3 and 4, had no predictions for specific thresholds (CT $>$ 20mNs and $<$ 8mNs). However, adjusting the thresholds, specifically for CT 15mNs & Amp 5mNs, revealed the possibility of predicting a fourth class with a probability of 0.73.

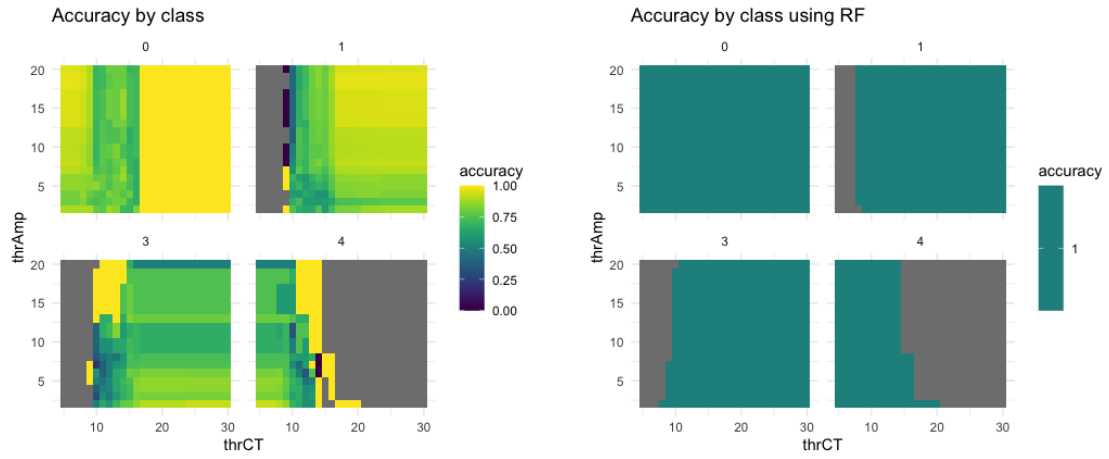
4.2.2 Imputed Data - MLM vs Random Forests

With an equal number of observations (216) as presented in Table 3, oversampling of 200 observations was applied to balance class distribution, and a 5-Fold Cross



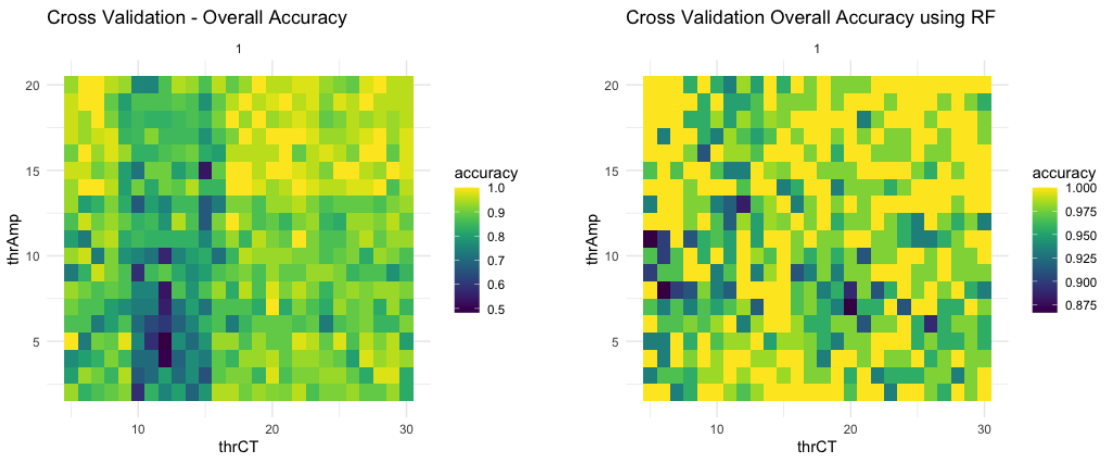
(a) MLM (b) RF

Figure 6: A heatmap illustrating the Overall Accuracy of Multinomial Logistic Model (MLM) and Random Forest (RF) on the Restricted Data. The analysis involves a sequential threshold range for Contraction Time (CT) from 5mNs to 30mNs and Amplitude (Amp) from 2mNs to 20mNs, each incremented by 1mNs.



(a) MLM (b) RF

Figure 7: A heatmap illustrating the Class Specific Accuracy of Multinomial Logistic Model (MLM) and Random Forest (RF) on the Restricted Data. The analysis involves a sequential threshold range for Contraction Time (CT) from 5mNs to 30mNs and Amplitude (Amp) from 2mNs to 20mNs, each incremented by 1mNs.



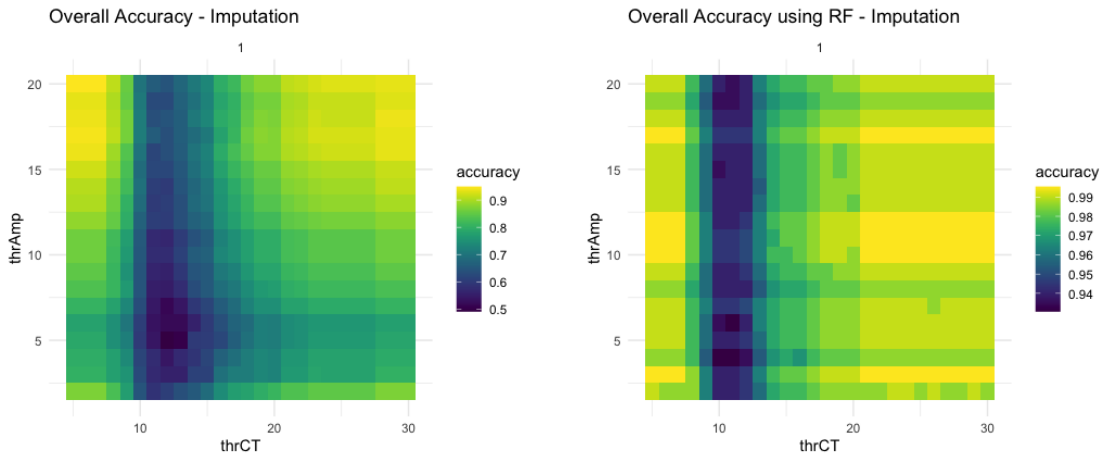
(a) MLM (b) RF

Figure 8: A heatmap depicting the Overall Accuracy of Multinomial Logistic Model (MLM) and Random Forest (RF) with Cross-Validation on the Restricted Data. The analysis involves a sequential threshold range for Contraction Time (CT) from 5mNs to 30mNs and Amplitude (Amp) from 2mNs to 20mNs, each incremented by 1mNs.

(a) MLM (b) RF

Figure 9: A heatmap depicting the Class-specific Accuracy of Multinomial Logistic Model (MLM) and Random Forest (RF) with Cross-Validation on the Restricted Data. The analysis involves a sequential threshold range for Contraction Time (CT) from 5mNs to 30mNs and Amplitude (Amp) from 2mNs to 20mNs, each incremented by 1mNs.

Validation was utilized. Figures 12 to 17 depict well-dispersed overall accuracy and accuracy by classes across various approaches and thresholds. Focusing on the CT 20mNs and Amp 8mNs threshold: The overall accuracy of 0.94623



(a) MLM (b) RF

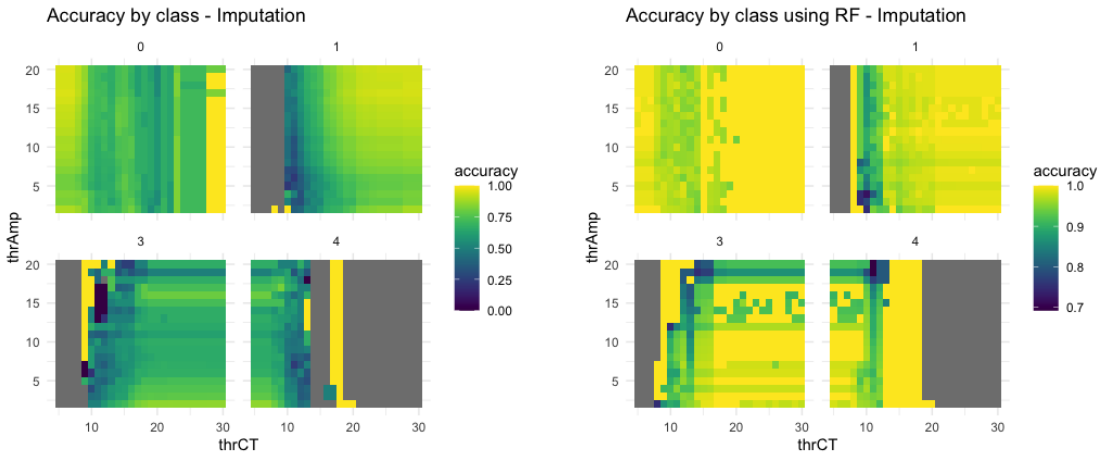
Figure 10: A heatmap illustrating the Overall Accuracy of Multinomial Logistic Model (MLM) and Random Forest (RF) on the Imputed Data. The analysis involves a sequential threshold range for Contraction Time (CT) from 5mNs to 30mNs and Amplitude (Amp) from 2mNs to 20mNs, each incremented by 1mNs.

under the cross-validation approach, with class-specific accuracies of 0.9687500, 0.9642857, and 0.9090909, outperformed the MLM with oversampling (overall accuracy: 0.7683; class-specific accuracies: 0.8764, 0.6393, and 0.8128). These differences in results may be influenced by the considerations outlined in Table 6.

Influential Change of the Threshold

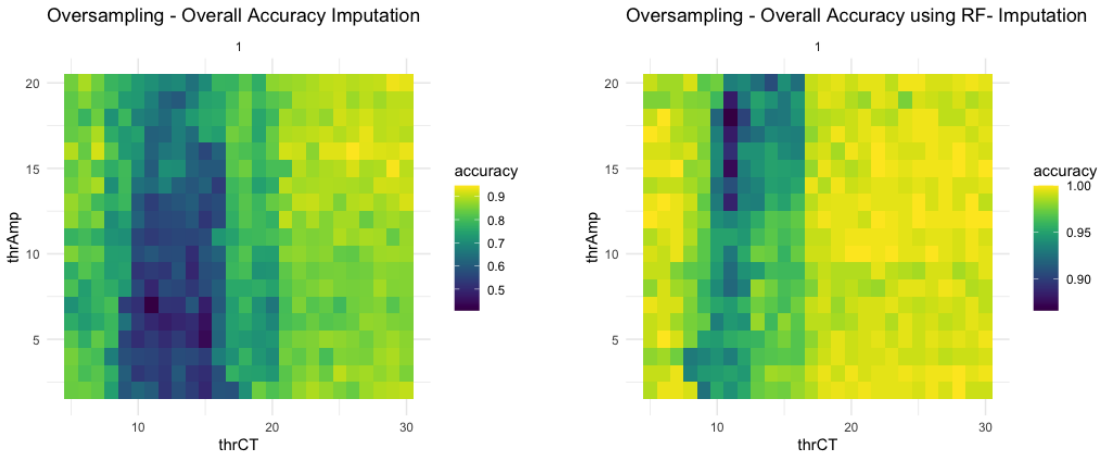
Changing the threshold values had a notable impact on the overall accuracy and class-specific accuracy of the models, as presented in Tables 4 and 5. For example, at CT 25mNs & Amp 10mNs, MLM with oversampling achieved an overall accuracy of 1.000, while MLM without oversampling and RF with cross-validation scored 0.8732 and 0.9785, respectively

The analysis of model performance across different threshold settings under-



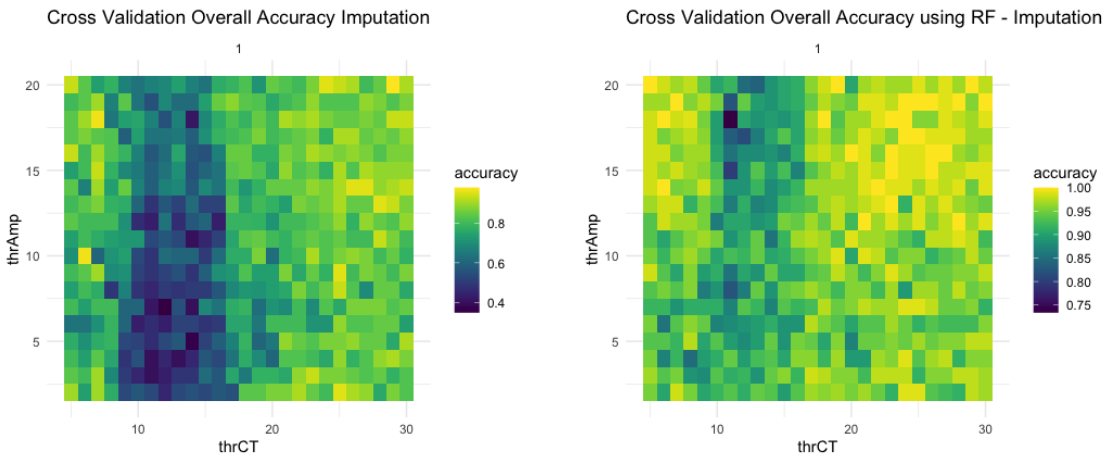
(a) MLM (b) RF

Figure 11: A heatmap illustrating the Class Specific Accuracy of Multinomial Logistic Model (MLM) and Random Forest (RF) on the Imputed Data. The analysis involves a sequential threshold range for Contraction Time (CT) from 5mNs to 30mNs and Amplitude (Amp) from 2mNs to 20mNs, each incremented by 1mNs.



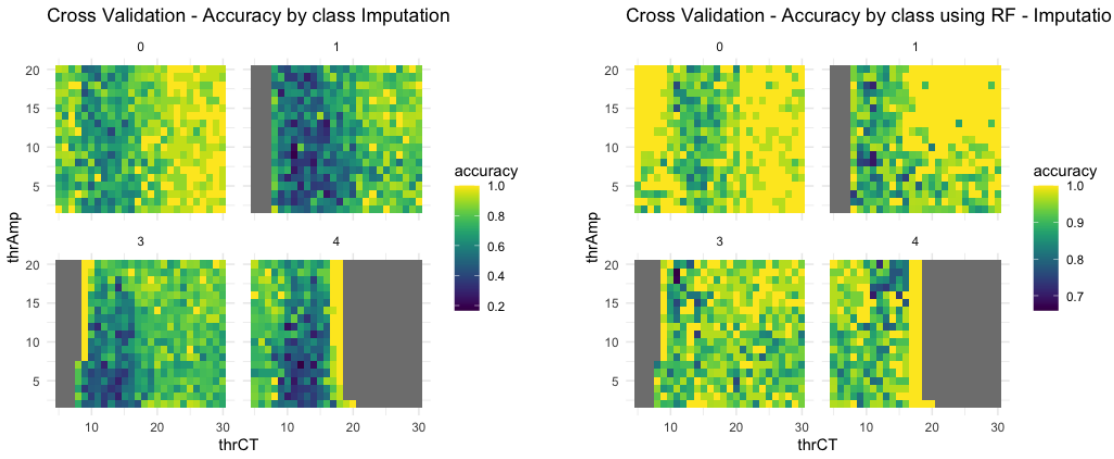
(a) MLM (b) RF

Figure 12: A heatmap illustrating the Overall Accuracy of Multinomial Logistic Model (MLM) and Random Forest (RF) with Oversampling on the Imputed Data. The analysis involves a sequential threshold range for Contraction Time (CT) from 5mNs to 30mNs and Amplitude (Amp) from 2mNs to 20mNs, each incremented by 1mNs.



(a) MLM (b) RF

Figure 13: A heatmap illustrating the Overall Accuracy of Multinomial Logistic Model (MLM) and Random Forest (RF) with Cross Validation on the Imputed Data. The analysis involves a sequential threshold range for Contraction Time (CT) from 5mNs to 30mNs and Amplitude (Amp) from 2mNs to 20mNs, each incremented by 1mNs.



(a) MLM (b) RF

Figure 14: heatmap illustrating the Overall Accuracy of Multinomial Logistic Model (MLM) and Random Forest (RF) with Cross Validation on the Imputed Data. The analysis involves a sequential threshold range for Contraction Time (CT) from 5mNs to 30mNs and Amplitude (Amp) from 2mNs to 20mNs, each incremented by 1mNs.

Thresholds	MLM_w/o	MLM_w	MLM_cv	RF_w/o	RF_w	RF_cv
CT 25mNs & Amp 10mNs	0.8732	1.000	0.9333	1.0000	1.0000	0.9785
CT 15mNs & Amp 5mNs	0.7042	0.7333	0.6333	1.0000	1.0000	0.9113
CT 20mNs & Amp 10mNs	0.8732	0.9289	0.8889	1.0000	1.0000	0.9247

Table 6: Distribution of Overall Accuracy of the Models: w (with oversampling), w/o (without oversampling), and cv (cross-validation)

Thresholds	Class	MLM_w/o	MLM_w	MLM_cv	RF_w/o	RF_w	RF_cv
CT 25mNs & Amp 10mNs	0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	1	0.9123	1.0000	1.0000	1.0000	1.0000	0.9677
	3	0.6667	1.0000	0.8333	1.0000	1.0000	0.9677
CT 15mNs & Amp 5mNs	0	0.7273	0.7907	0.6667	1.0000	1.0000	1.0000
	1	0.6563	0.7500	0.7778	1.0000	1.0000	0.7949
	3	0.7500	0.6563	0.5455	1.0000	1.0000	1.0000
CT 20mNs & Amp 10mNs	0	1.0000	1.0000	1.0000	1.0000	1.0000	0.9118
	1	0.9074	0.9403	0.7778	1.0000	1.0000	0.9286
	3	0.6667	0.8554	0.9167	1.0000	1.0000	0.9355

Table 7: Accuracy by Class Influence with Changes in Threshold: w (with oversampling), w/o (without oversampling), and cv (cross-validation)

scores the impact of oversampling on accuracy enhancement, showcasing its effectiveness in addressing class imbalances. Models with oversampling consistently outperform counterparts without oversampling, as evident in various threshold combinations, such as CT 25 ms Amp 10 mN, where both MLM and RF models achieve perfect accuracy for Class 0. This stability persists across different thresholds, emphasizing oversampling's reliability. Cross-validation results reinforce the models' robustness, particularly in scenarios like CT 15 ms Amp 5 mN, where MLM models, especially the oversampled version, excel in handling imbalanced classes. The RF algorithm, even without oversampling, maintains high accuracy across various thresholds, highlighting its inherent ability to handle imbalanced data. At CT 20 ms Amp 10 mN, oversampling contributes to improved performance in Class 1 and Class 3 for both MLM and RF models. The consistent accuracy enhancement with oversampling, coupled with the stability of the RF al-

Data Type	MLM_w/o	MLM_w	MLM_cv	RF_w/o	RF_w	RF_cv
Restricted	0.7685	0.9333	0.9778	1.0000	1.0000	1.0000
Imputed	0.7685	0.9333	0.8000	0.9814	0.9935	0.9462

Table 8: Restricted and Imputed Data: Distribution of Overall Accuracy of the Models: w (with oversampling), w/o (without oversampling), and cv (cross-validation) with threshold of 20mNs and 8mNs

Data Type	Class	MLM_w/o	MLM_w	MLM_cv	RF_w/o	RF_w	RF_cv
Restricted	0	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
	1	0.9123	1.0000	0.9375	1.0000	1.0000	1.0000
	3	0.7333	0.9259	0.8333	1.0000	1.0000	1.0000
Imputed	0	0.5833	0.8367	0.9285	1.0000	1.0000	0.9687
	1	0.8023	0.5894	0.8000	0.9746	1.0000	0.9643
	3	0.6563	0.7383	0.7143	1.0000	0.9810	0.9091

Table 9: Restricted and Imputed Data: Accuracy by Class Influence: w (with oversampling), w/o (without oversampling), and cv (cross-validation) with threshold of 20mNs and 8mNs

gorithm, positions it as a reliable choice for motoneuron classification tasks across diverse scenarios, instilling confidence in generalization capabilities.

Analyzing results from Tables 6 and 7 provides additional insights into the stability and performance of the Random Forests (RF) algorithm. While consistently achieving high accuracy overall and by class, the threshold setting of 15mNs and 5mNs introduces an intriguing observation. At this threshold, the RF model demonstrates the potential to predict a fourth class with 73% accuracy, showcasing its effective handling of diverse instances. Low variability in out-of-bag (OOB) estimates supports the RF model's reliability, indicating consistent prediction errors across scenarios and robust performance on unseen data. In contrast, the Multilevel Model (MLM) experiences accuracy variations with changes in thresholds, underscoring sensitivity to adjustments. This observation emphasizes the need for careful consideration and selection of threshold values to opti-

mize MLM performance in motoneuron classification. Nuanced insights from the RF model, combined with considerations for the MLM, provide a comprehensive understanding of strengths and potential areas for refinement. Further examina-

Considerations	Selection & Reason
Nature of Data	MLM
Handling Imbalanced Data	Depends
Interpretability	MLM
Computational Efficiency	MLM

Table 10: Description of Considered Factors and Selected Model

tion of class-specific accuracies at specific thresholds, such as CT 15mNs Amp 5mNs and CT 20mNs Amp 10mNs, offers nuanced perspectives. At CT 15mNs Amp 5mNs, variations in class-specific accuracies highlight potential trade-offs. For Class 0, MLM without oversampling scored 0.7273, MLM with oversampling achieved 0.7907, and RF with cross-validation reached a perfect accuracy of 1.000. For Class 1, MLM without oversampling scored 0.6563, MLM with oversampling achieved 0.7500, and RF with cross-validation scored 0.7949. For Class 3, MLM without oversampling achieved 0.7500, MLM with oversampling scored 0.6563, and RF with cross-validation reached a perfect accuracy of 1.000. Similar variations were observed at CT 20mNs Amp 10mNs. Considering these results, the choice between MLM and RF depends on factors outlined in Table 6, including data nature, interpretability, and computational efficiency, with the selection influenced by specific requirements for handling imbalanced data.

In summary, the findings indicate that imputation proves beneficial as it preserves data size, leading to more accurate and stable classification models . The increased sample size from 71 to 216 contributes to improved model performance, crucial for tasks with diverse class distributions. Imputation also helps reduce bias introduced by missing data, enhancing the representativeness of each class and facilitating reliable class statistics. This, in turn, supports more robust downstream

analyses and generalization of the MLM model. The choice between MLM and RF should consider the specific nature of the data, interpretability requirements, and computational efficiency. Therefore, MLM is advantageous due to its ease in understanding the impact of each predictor on the logit response variable of 0, 1, 3, which is crucial for comprehending the relationship between electrical properties and the response variable, especially considering our relatively small dataset.

List of References

[1] M. d. L. Martinez-Silva, R. D. Imhoff-Manuel, A. Sharma, C. Heckman, N. A. Shneider, F. Roselli, D. Zytnicki, and M. Manuel, "Hypoexcitability precedes denervation in the large fast-contracting motor units in two unrelated mouse models of als," *Elife*, vol. 7, p. e30955, 2018.

[2] R. J. Little and D. B. Rubin, *Statistical analysis with missing data*. John Wiley & Sons, 2019, vol. 793.

CHAPTER 5

Conclusions

This study delves into classification methods for predicting classes of motoneurons based on their electrical properties, drawing insights from empirical data obtained in Dr. Manuel's laboratory experiments. The proposed classification models, multinomial logistic regression, and random forests, showcase remarkable accuracy in classifying Physiological Types. Notably, the proposed analysis combines these models and effectively addresses class imbalances through the application of oversampling techniques and handles missing data using hot deck imputation. In the context of real-world data, the study highlights the non-standardized nature of threshold choices, emphasizing the impact of opting for specific thresholds such as CT 20mNs and Amp 8mNs, which resulted in enhanced overall accuracy and accuracy by class, as illustrated in Tables 8 and 9. A similar observation is made concerning overall accuracy, particularly in the transition from MLM_w/o to MLM_w and RF_w/o to RF_w (Table 6 and 7). For instance, at CT 25mNs Amp 10mNs, both models exhibit a rise in overall accuracy from 0.8732 to 1.000, with RF maintaining its accuracy at 1.0. Notably, fixing thresholds within the ranges of 15mNs and 5mNs led to improved accuracy, potential inclusion of new variables for imputation, and the intriguing prediction of a fourth class with 73% accuracy, showcasing effective handling of diverse instances. Additionally, Table 10 provides precision insights, with lower standard errors (e.g., SR, ahpHalfRelax, and RinPeak), indicating reliable estimates. The exponential values in Table 11 further underscore specific variables' substantial influence on the response.

In the realm of Random Forest analysis, the identification of the 8 most crucial variables, especially RinPeak, demonstrates their significance in both restricted and

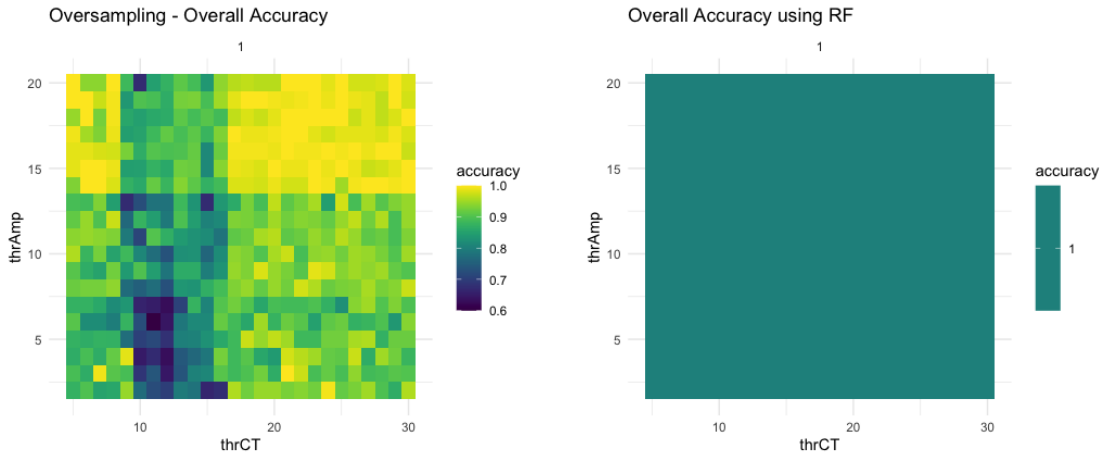
imputed data scenarios. Looking ahead, future research should explore advanced machine learning techniques, including ensemble and deep learning algorithms such as Gradient Boosting Machines (GBM), XGBoost, Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN)[1]. Emphasizing transfer learning, feature engineering, and extensive experimentation on diverse datasets is crucial for validating and enhancing model performance. Additionally, addressing the absence of significant interaction terms in reported tests should be a focal point for future investigations, ensuring a comprehensive understanding of the underlying dynamics.

List of References

[1] P. Ladosz, L. Weng, M. Kim, and H. Oh, "Exploration in deep reinforcement learning: A survey," *Information Fusion*, vol. 85, pp. 1–22, 2022.

.1 Appendix A

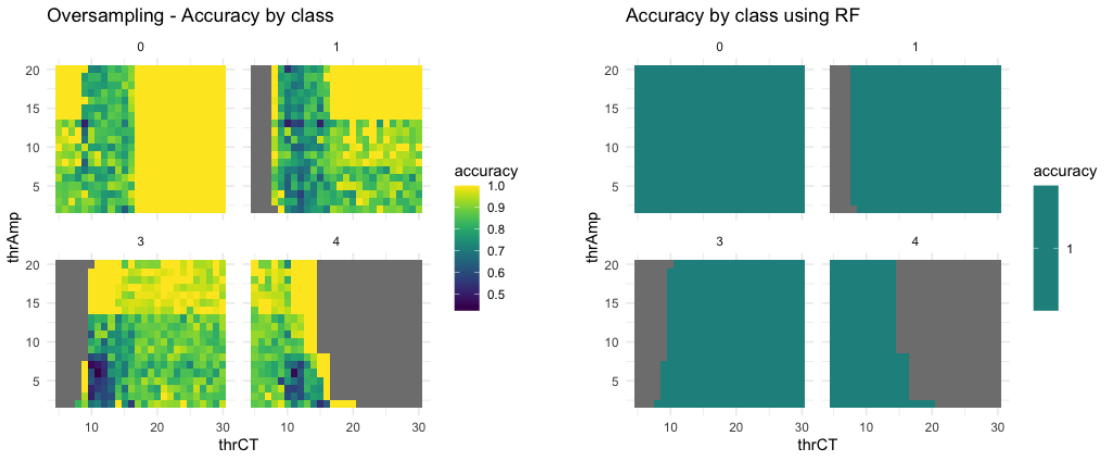
This shows the other heatmaps or plots of our findings



(a) MLM (b) RF

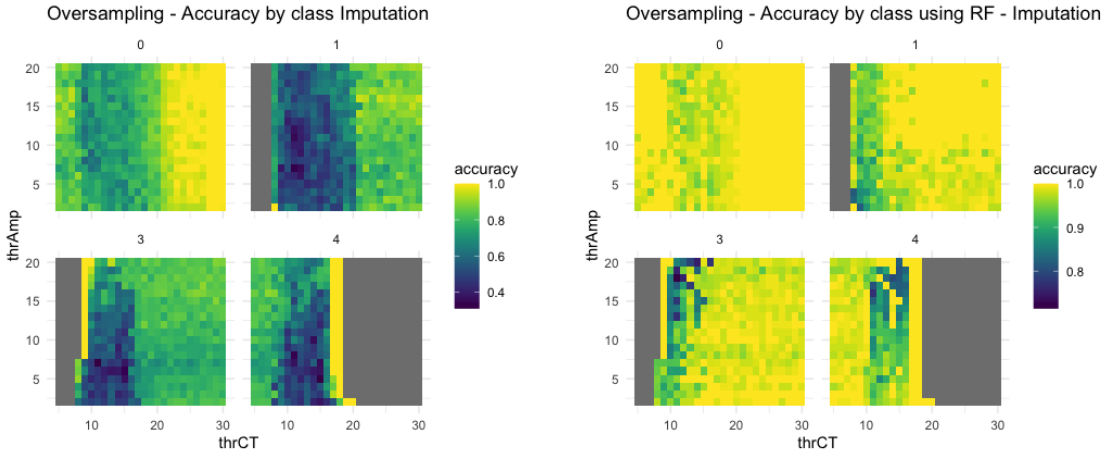
Figure .15: A heatmap illustrating the Overall Accuracy of Multinomial Logistic Model (MLM) and Random Forest (RF) with Oversampling on the Restricted Data. The analysis involves a sequential threshold range for Contraction Time (CT) from 5 ms to 30 ms and Amplitude (Amp) from 2 mN to 20 mN, each incremented by 1

List of References



(a) MLM (b) RF(Both baseline and oversampling)

Figure .16: A heatmap illustrating the Class-specific Accuracy of Multinomial Logistic Model (MLM) and Random Forest (RF) with Oversampling on the Restricted Data. The analysis involves a sequential threshold range for Contraction Time (CT) from 5 ms to 30 ms and Amplitude (Amp) from 2 mN to 20 mN, each incremented by 1



(a) MLM (b) RF

Figure .17: A heatmap illustrating the Class Specific Accuracy of Multinomial Logistic Model (MLM) and Random Forest (RF) with Oversampling on the Imputed Data. The analysis involves a sequential threshold range for Contraction Time (CT) from 5 ms to 30 ms and Amplitude (Amp) from 2 mN to 20 mN, each incremented by 1.

.2 Appendix B

This shows the other tables of our findings:

	(Intercept)	rheobase	ahpAmp	ahpHalfRelax	RinPeak	Ion	SR	spikeWidth	L
1	61.67243	27.76538	1.906413	-13.45122	26.5574	14.06556	-3.596994	74.55216	-73.20303
3	74.46532	28.07269	3.354719	-13.26260	22.7855	14.01311	-15.577276	66.00908	-76.39380

Table .11: Coefficients of MLM

List of References

[1] B. Ripley, W. Venables, and M. B. Ripley, "Package 'nnet'," *R package version*, vol. 7, no. 3-12, p. 700, 2016.

	(Intercept)	rheobase	ahpAmp	ahpHalfRelax	RinPeak	Ion	SR	spikeWidth	L
1	18.89031	14.82227	14.23583	3.258721	8.618745	37.86559	3.723497	14.13919	12.43165
3	19.00807	14.82507	14.24809	3.261127	8.742742	37.86801	3.807966	14.29871	12.42715

Table .12: Standard Errors of the Coefficient of the MLM

	(Intercept)	rheobase	ahpAmp	ahpHalfRelax	RinPeak	Ion	SR	spikeWidth	L
1	6.081313e+26	1.143801e+12	6.728911	1.439489e-06	341769128001	1284089	2.740598e-02	2.385578e+32	1.615572e-32
3	2.187144e+32	1.555293e+12	28.637548	1.738301e-06	7863489662	1218478	1.717414e-07	4.649222e+28	6.646491e-34

Table .13: Exponentials of Coefficients of the MLM[1]

Variable	Importance (%)
RinPeak	100.000
Ion	48.883
ahpHalfRelax	10.896
ahpAmp	9.309
SR	9.045
rheobase	2.416
L	1.752
spikeWidth	0.000

Table .14: Random Forest Variable Importance (Restricted)

Variable	Importance (%)
RinPeak	100.00
Ion	68.54
ahpHalfRelax	46.12
rheobase	37.38
ahpAmp	31.47
L	28.94
SR	10.82
spikeWidth	0.00

Table .15: Random Forest Variable Importance (Imputed)

BIBLIOGRAPHY

Allison, P. D., *Missing data*. Sage publications, 2001.

Bennett, D. A., "How can i deal with missing data in my study?" *Australian and New Zealand journal of public health*, vol. 25, no. 5, pp. 464–469, 2001.

Biau, G. and Scornet, E., "A random forest guided tour," *Test*, vol. 25, pp. 197–227, 2016.

Bkaczyk, M., Manuel, M., Roselli, F., and Zytnicki, D., "Diversity of mammalian motoneurons and motor units," in *Vertebrate Motoneurons*. Springer, 2022, pp. 131–150.

Breiman, L., "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.

Buda, M., Maki, A., and Mazurowski, M. A., "A systematic study of the class imbalance problem in convolutional neural networks," *Neural networks*, vol. 106, pp. 249–259, 2018.

Burke, R., Levine, D., Tsairis, P., and Zajac Iii, F., "Physiological types and histochemical profiles in motor units of the cat gastrocnemius," *The Journal of physiology*, vol. 234, no. 3, pp. 723–748, 1973.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P., "Smote: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

Chen, X.-W. and Wasikowski, M., "Fast: a roc-based feature selection metric for small samples and imbalanced data classification problems," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2008, pp. 124–132.

Dev, V. A. and Eden, M. R., "Formation lithology classification using scalable gradient boosted decision trees," *Computers & chemical engineering*, vol. 128, pp. 392–404, 2019.

Diaz-Uriarte, R. and Alvarez de Andres, S., "Gene selection and classification of microarray data using random forest," *BMC bioinformatics*, vol. 7, pp. 1–13, 2006.

Drummond, C., Holte, R. C., *et al.*, "C4. 5, class imbalance, and cost sensitivity: why under-sampling beats over-sampling," in *Workshop on learning from imbalanced datasets II*, vol. 11, 2003, pp. 1–8.

Enders, C., "Applied missing data analysis: Guilford press," *New York*, 2010.

Galar, M., Fernandez, A., Barrenechea, E., Bustince, H., and Herrera, F., "A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp. 463–484, 2011.

He, H. and Garcia, E. A., "Learning from imbalanced data," *IEEE Transactions on knowledge and data engineering*, vol. 21, no. 9, pp. 1263–1284, 2009.

Heckman, C. and Enoka, R. M., "Motor unit," *Comprehensive physiology*, vol. 2, no. 4, pp. 2629–2682, 2012.

James, G., Witten, D., Hastie, T., Tibshirani, R., *et al.*, *An introduction to statistical learning*. Springer, 2013, vol. 112.

Ladosz, P., Weng, L., Kim, M., and Oh, H., "Exploration in deep reinforcement learning: A survey," *Information Fusion*, vol. 85, pp. 1–22, 2022.

Little, R. J., "A test of missing completely at random for multivariate data with missing values," *Journal of the American statistical Association*, vol. 83, no. 404, pp. 1198–1202, 1988.

Little, R. J. and Rubin, D. B., *Statistical analysis with missing data*. John Wiley & Sons, 2019, vol. 793.

Manuel, M., Chardon, M., Tysseling, V., and Heckman, C., "Scaling of motor output, from mouse to humans," *Physiology*, vol. 34, no. 1, pp. 5–13, 2019.

Manuel, M., Iglesias, C., Donnet, M., Leroy, F., Heckman, C., and Zytnicki, D., "Fast kinetics, high-frequency oscillations, and subprimary firing range in adult mouse spinal motoneurons," *Journal of Neuroscience*, vol. 29, no. 36, pp. 11 246–11 256, 2009.

Manuel, M. and Zytnicki, D., "Alpha, beta and gamma motoneurons: functional diversity in the motor system's final pathway," *Journal of integrative neuroscience*, vol. 10, no. 03, pp. 243–276, 2011.

Manuel, M. and Zytnicki, D., "Molecular and electrophysiological properties of mouse motoneuron and motor unit subtypes," *Current opinion in physiology*, vol. 8, pp. 23–29, 2019.

Martin-Baos, J. A., Garcia-Rodenas, R., and Rodriguez-Benitez, L., "Revisiting kernel logistic regression under the random utility models perspective. an interpretable machine-learning approach," *Transportation Letters*, vol. 13, no. 3, pp. 151–162, 2021.

Martinez-Silva, M. d. L., Imhoff-Manuel, R. D., Sharma, A., Heckman, C., Shneider, N. A., Roselli, F., Zytnicki, D., and Manuel, M., "Hypoexcitability precedes denervation in the large fast-contracting motor units in two unrelated mouse models of als," *Elife*, vol. 7, p. e30955, 2018.

Pranckevivcius, T. and Marcinkevivcius, V., "Comparison of naive bayes, random forest, decision tree, support vector machines, and logistic regression classifiers for text reviews classification," *Baltic Journal of Modern Computing*, vol. 5, no. 2, p. 221, 2017.

Ripley, B., Venables, W., and Ripley, M. B., "Package 'nnet'," *R package version*, vol. 7, no. 3-12, p. 700, 2016.

Rubin, D. B., "Inference and missing data," *Biometrika*, vol. 63, no. 3, pp. 581–592, 1976.

Schafer, J. L. and Graham, J. W., "Missing data: our view of the state of the art." *Psychological methods*, vol. 7, no. 2, p. 147, 2002.

Scheffer, J., "Dealing with missing data," 2002.

Schiaffino, S. and Reggiani, C., "Fiber types in mammalian skeletal muscles," *Physiological reviews*, vol. 91, no. 4, pp. 1447–1531, 2011.

Song, L., Langfelder, P., and Horvath, S., "Random generalized linear model: a highly accurate and interpretable ensemble predictor," *BMC bioinformatics*, vol. 14, no. 1, pp. 1–22, 2013.

Stoltzfus, J. C., "Logistic regression: a brief primer," *Academic emergency medicine*, vol. 18, no. 10, pp. 1099–1104, 2011.

Svetnik, V., Liaw, A., Tong, C., Culberson, J. C., Sheridan, R. P., and Feuston, B. P., "Random forest: a classification and regression tool for compound classification and qsar modeling," *Journal of chemical information and computer sciences*, vol. 43, no. 6, pp. 1947–1958, 2003.

Toolan, T. M. and Tufts, D. W., "Detection and estimation in non-stationary environments," in *Proceedings IEEE Asilomar Conference on Signals, Systems & Computers*, Nov. 2003, pp. 797–801.

Van Buuren, S. and Groothuis-Oudshoorn, K., "mice: Multivariate imputation by chained equations in r," *Journal of statistical software*, vol. 45, pp. 1–67, 2011.

Witten, D. and James, G., *An introduction to statistical learning with applications in R.* springer publication, 2013.

Xu, R., "Improvements to random forest methodology," Ph.D. dissertation, Iowa State University, 2013.

Zengel, J. E., Reid, S. A., Sypert, G. W., and Munson, J. B., "Membrane electrical properties and prediction of motor-unit type of medial gastrocnemius motoneurons in the cat," *Journal of neurophysiology*, vol. 53, no. 5, pp. 1323–1344, 1985.