Developing machine learning solutions for healthcare problems

by

Mohammad Fili

A dissertation submitted to the graduate faculty

in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Major: Industrial Engineering

Program of Study Committee:
Guiping Hu, Major Professor
Kris De Brabanter
Qing Li
Sigurdur Olafsson
Cameron MacKenzie

The student author, whose presentation of the scholarship herein was approved by the program of study committee, is solely responsible for the content of this dissertation. The Graduate College will ensure this dissertation is globally accessible and will not permit alterations after a degree is conferred.

Iowa State University

Ames, Iowa

2022

DEDICATION

I would like to dedicate this dissertation to my mom and dad without whose support I would not have been able to complete this work.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

ACKNOWLEDGMENTS

I would like to take this opportunity to express my thanks to those who helped me with various aspects of conducting research and the writing of this dissertation. First and foremost, Dr. Guiping Hu for her guidance, patience and support throughout these years. Her insights and words of encouragement have often inspired me and renewed my hopes for completing my graduate education.

I would also like to thank my committee members for their efforts and contributions to this work: Dr. Kris De Brabanter, Dr. Qing Li, Dr. Sigurdur Olafsson, and Dr. Cameron MacKenzie.

I would additionally like to thank MD. Hillel Haim for his technical guidance and supports for the first study, Parvin Mohammadi in second study, Dr. Auriel Willette for the third study, and Dr. Gary Mirka for his mentorship and guidance.

At last, I would also like to thank my family, friends, colleagues, the department faculty, and staff for making my time at Iowa State University a wonderful experience

ABSTRACT

Machine learning (ML) is a multidisciplinary field that is concerned with designing and utilizing methods that automatically learn and improve the experience. ML footprints can be found in almost any field, including healthcare. In this dissertation, specific analytical solutions are developed for healthcare problems using a combination of ML and Optimization (Opt) tools. This dissertation includes three research questions in healthcare and our proposed solution to address them. In the first study, we explored how the changes in the amino acid sequences of the HIV-1 Env can be predicted using the neighboring variability. This study is a vital step toward predicting future changes in the amino acid sequences and personalized treatments for HIV patients. In the second study, we proposed an algorithm to accurately predict the ICU needs of COVID-19 patients so that the resources are managed more efficiently, and more lives could be saved. The last study focused on finding a mechanism to differentiate between Super-Agers and Cognitive Decliners using the resting-state functional MRI data. We utilized neural network data and cognitive information to optimally assign participants to one of the classes, Super-Agers or Cognitive Decliners, using a hybrid ML-Opt algorithm.

CHAPTER 1. GENERAL INTRODUCTION

Machine learning (ML) is a multidisciplinary field concerned with answering how we can construct computer programs that automatically improve the experience [1]. This field has roots in many disciplines such as algebra, statistics, data management, and computer science. Therefore, finding a clear boundary may not be possible. With the advance of ML, more of its applications are found that take advantage of techniques in this field to discover patterns in data. It is now inevitable to ignore the importance and the role of machine learning in new discoveries. It is now omnipresent and used widely in many applications [2]. ML is a broad area including so many tools and techniques and can be broken down into the following categories:

- **Supervised Learning**: This group of ML techniques utilizes the labels of the instances in the training set as a supervisor to instruct the model [3]. There are two main sets of problems that supervised learning techniques try to address: regression and classification. In regression, the response variable is of type numeric, while we have a categorical response variable in a classification problem.

- **Unsupervised Learning**: Unlike supervised learning, this group doesn't take advantage of the known labels or values for the response variable. Instead, the properties of the joint probability density of the input are inferred directly without having any supervisor [4]. Examples of this group are clustering, association rule mining, and outlier detection.

- **Semi-supervised Learning**: This group is a combination of the first two categories. Semi-supervised learning methods instruct the predictive model using both the labeled and unlabeled data [5].

- **Reinforcement Learning**: This type of learning is concerned with learning what to do to maximize a numeric reward signal [6]. It needs to interact with the environment and try different actions to learn which action yields the maximum reward.

ML footprints can be found in almost any field: agriculture [7], stock market [8], healthcare [9], manufacturing [10], transportation [11], geology [12], and psychology [13], to name a few. In this dissertation, we specifically focused on the application of ML in healthcare. The healthcare domain itself is a vast field in which ML tools can be used to address a specific research question of interest. Some of the examples are disease prediction [14], drug prescriptions [15], pandemic growth rate analysis [16], and medical image processing [17].

This dissertation includes three healthcare-related studies: the first study is pertinent to human immunodeficiency virus type 1 (HIV-1), second study focuses on the severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2), and the last study is related to cognitive performance and Alzheimer's Disease (AD). In the first study, we examined a hypothesis regarding the structural pattern of volatility in amino acid sequences of the envelope glycoproteins (Env) in HIV-1. We develop a novel classification method for this purpose which is explained in detail in Chapter 2. The second study deals with using patients' information within the first few hours after admission to predict the need for ICU beds which is explored in Chapter 3. In Chapter 4 and the last study, we explain the proposed algorithm to distinguish between Super-Agers and Cognitive Decliners using resting state functional MRI (rsfMRI) data. Finally, we conclude in Chapter 5 and discuss the findings in each study and how they contribute to the field of machine learning as well as the healthcare domain.

Between the years 1981 and 2019, 2.2 million new HIV infections are estimated among people aged at least 13 years in the United States [18], and 38 million individuals were estimated to live with HIV in 2019 [19]. To treat HIV-infected patients, multiple therapeutics are available; they bind to HIV-1 proteins and can effectively inhibit their function. However, the replication machinery of HIV-1 is highly prone to errors. As a result, new variants of its proteins are generated, some of which contain changes at the sites targeted by the therapeutics [20].

As a research question, we decided to investigate the possibility of personalized treatments for HIV-1 patients. For this purpose, we decided to explore how the changes in the amino acid sequences of the Env can be predicted for future time points. Since the future variability of amino acids sequences is dependent on the current variability, we started this project with a smaller step: we hypothesized that the absence or presence of sequence variance at any position of Env could be predicted based on the variance at adjacent positions on the three-dimensional structure of the protein.

For this purpose, we designed a novel dynamic ensemble selection technique based on bootstrapping. The intuition behind this method is to dynamically find and apply the best set of learners to predict the class label of an instance according to the performance of those learners in different regions of the sample space. In fact, as the complexity of any dataset increases, the ability of a single classifier to capture all patterns is reduced. Therefore, it is necessary to integrate multiple classifiers for improved classification accuracy. However, using the same set of classifiers statically over the whole feature space can affect the overall performance of an algorithm. In fact, a classifier may perform well in some subspaces of the data but exhibit poor performance in others. One solution to this problem is the selection of the best classifiers out of a pool of existing learners dynamically based on some competitiveness criteria and to use this subset of classifiers for predicting the class label of a new observation in a region.

KBC model showed considerable improvement in predicting variance at multi-position features. We tested two Env domains targeted by therapeutics; the CD4-binding site and the high-mannose patch of Env (composed of 23 and 10 positions, respectively). Both domains constitute targets for multiple HIV-1 therapeutics [21, 22, 23, 24, 25, 26, 27]. Chapter 2 will cover the details of this study.

Severe acute respiratory syndrome coronavirus 2 (Sars-CoV-2) emerged in December 2019 as a threat to public health, and in March 2020, the World Health Organization (WHO) declared it as a global pandemic [28]. Based on John Hopkins University Coronavirus Resource Center's estimations, about 48 million individuals were infected with Covid-19 [29], more than 27 million

people have been hospitalized, and about 7 million individuals have been admitted to ICU as of the end of November 2021 in the United States [30]. To find the patients in need of ICU, time is a critical factor since early identification of such patients will lead to life-saving results. In the second study, we developed an algorithm capable of identifying the patients in need of ICU in the first few hours after admission. We designed a new classification algorithm based on the stacked generalization that incorporates the set of regional and neighborhood evaluations to provide information to a second learner, called meta-learner. The proposed method improved the results compared to the baseline models and also the previous studies and reduced the waiting time significantly from 12 hours to only 2 hours. The detail of this study is provided in Chapter 3.

The third problem in this dissertation is related to cognitive performance and AD. Aging is frequently associated with a cognitive decline in several domains [31, 32]. Yet, there is significant variation in how cognitive function changes in mid- and late-life adults. Some adults aged 80 years or older, called Super-Agers, show cognitive performance that is similar to healthy middle-aged adults, particularly for declarative memory [33, 34, 35, 36]. In this study, we used neural network data to distinguish between Super-Agers and Cognitive Decliners. Since the problem did not have class labels, we had to find a mechanism to assign the class labels to participants. We proposed a hybrid algorithm of machine learning and optimization that finds optimal label assignment through a repetitive process using Bayesian Optimization. The proposed algorithm has an internal procedure for label assignment where it takes advantage of cognitive test information. We showed that this algorithm is capable of accurately identifying the Super-Agers and Cognitive-Decliners. The details of the algorithm is explained in Chapter 4.

1.1 References

[1] Tom Michael Mitchell. *Machine Learning textbook*. 1997.

[2] K. Shailaja, B. Seetharamulu, and M. A. Jabbar. Machine Learning in Healthcare: A Review. In *Proceedings of the 2nd International Conference on Electronics, Communication and Aerospace Technology, ICECA 2018*, 2018.

[3] Pádraig Cunningham, Matthieu Cord, and Sarah Jane Delany. Supervised learning. In *Cognitive Technologies*, 2008.

[4] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *Elements of Statistical Learning 2nd ed.* 2009.

[5] Yu Feng Li and De Ming Liang. Safe semi-supervised learning: a brief introduction, 2019.

[6] Richard S. Sutton and Andrew G. Barto. Reinforcement Learning, Second Edition: An Introduction - Complete Draft. *The MIT Press*, 2018.

[7] Saeed Khaki, Hieu Pham, and Lizhi Wang. Simultaneous corn and soybean yield prediction from remote sensing data using deep transfer learning. *Scientific Reports*, 2021.

[8] Ahmed Ibrahim, Rasha Kashef, and Liam Corrigan. Predicting market movement direction for bitcoin: A comparison of time series modeling methods. *Computers and Electrical Engineering*, 2021.

[9] Mohammad Fili, Guiping Hu, Changze Han, Alexa Kort, and Hillel Haim. A stacking-based classification approach: Case study in volatility prediction of HIV-1 viruses. In *Informs Conference on Service Science*.

[10] Thorsten Wuest, Daniel Weimer, Christopher Irgens, and Klaus Dieter Thoben. Machine learning in manufacturing: Advantages, challenges, and applications. *Production and Manufacturing Research*, 2016.

[11] M Fili and M Khedmati. Town trip forecasting based on data mining techniques. *Journal of Industrial Engineering, International*, 2020.

[12] Anuj Karpatne, Imme Ebert-Uphoff, Sai Ravela, Hassan Ali Babaie, and Vipin Kumar. Machine Learning for the Geosciences: Challenges and Opportunities. *IEEE Transactions on Knowledge and Data Engineering*, 2019.

[13] Dominic B. Dwyer, Peter Falkai, and Nikolaos Koutsouleris. Machine Learning Approaches for Clinical Psychology and Psychiatry, 2018.

[14] Mingyang Lu, Zhenjiang Fan, Bin Xu, Lujun Chen, Xiao Zheng, Jundong Li, Taieb Znati, Qi Mi, and Jingting Jiang. Using machine learning to predict ovarian cancer. *International Journal of Medical Informatics*, 2020.

[15] G. Segal, A. Segev, A. Brom, Y. Lifshitz, Y. Wasserstrum, and E. Zimlichman. Reducing drug prescription errors and adverse drug events by application of a probabilistic, machine-learning based clinical decision support system in an inpatient setting. *Journal of the American Medical Informatics Association*, 2019.

[16] Serkan Ballı. Data analysis of Covid-19 pandemic and short-term cumulative case forecasting using machine learning time series methods. *Chaos, Solitons and Fractals*, 2021.

[17] Muhammad Imran Razzak, Saeeda Naz, and Ahmad Zaib. Deep learning for medical image processing: Overview, challenges and the future. In *Lecture Notes in Computational Vision and Biomechanics*. 2018.

[18] Karin A. Bosh, H. Irene Hall, Laura Eastham, Demetre C. Daskalakis, and Jonathan H. Mermin. Estimated Annual Number of HIV Infections United States, 1981–2019. *MMWR. Morbidity and Mortality Weekly Report*, 2021.

[19] Global HIV & AIDS statistics - 2020 fact sheet, 2020.

[20] José M. Cuevas, Ron Geller, Raquel Garijo, José López-Aldeguer, and Rafael Sanjuán. Extremely High Mutation Rate of HIV-1 In Vivo. *PLoS Biology*, 2015.

[21] Leopold Kong, Jeong Hyun Lee, Katie J. Doores, Charles D. Murin, Jean Philippe Julien, Ryan McBride, Yan Liu, Andre Marozsan, Albert Cupo, Per Johan Klasse, Simon Hoffenberg, Michael Caulfield, C. Richter King, Yuanzi Hua, Khoa M. Le, Reza Khayat, Marc C. Deller, Thomas Clayton, Henry Tien, Ten Feizi, Rogier W. Sanders, James C. Paulson, John P. Moore, Robyn L. Stanfield, Dennis R. Burton, Andrew B. Ward, and Ian A. Wilson. Supersite of immune vulnerability on the glycosylated face of HIV-1 envelope glycoprotein gp120. *Nature Structural and Molecular Biology*, 2013.

[22] Laura M. Walker, Michael Huber, Katie J. Doores, Emilia Falkowska, Robert Pejchal, Jean Philippe Julien, Sheng Kai Wang, Alejandra Ramos, Po Ying Chan-Hui, Matthew Moyle, Jennifer L. Mitcham, Phillip W. Hammond, Ole A. Olsen, Pham Phung, Steven Fling, Chi Huey Wong, Sanjay Phogat, Terri Wrin, Melissa D. Simek, Wayne C. Koff, Ian A. Wilson, Dennis R. Burton, and Pascal Poignard. Broad neutralization coverage of HIV by multiple highly potent antibodies. *Nature*, 2011.

[23] Christine A. Bricault, Karina Yusim, Michael S. Seaman, Hyejin Yoon, James Theiler, Elena E. Giorgi, Kshitij Wagh, Maxwell Theiler, Peter Hraber, Jennifer P. Macke, Edward F. Kreider, Gerald H. Learn, Beatrice H. Hahn, Johannes F. Scheid, James M. Kovacs, Jennifer L. Shields, Christy L. Lavine, Fadi Ghantous, Michael Rist, Madeleine G. Bayne, George H. Neubauer, Katherine McMahan, Hanqin Peng, Coraline Chéneau, Jennifer J. Jones, Jie Zeng, Christina Ochsenbauer, Joseph P. Nkolola, Kathryn E. Stephenson, Bing Chen, S. Gnanakaran, Mattia Bonsignori, La Tonya D. Williams, Barton F. Haynes, Nicole Doria-Rose, John R. Mascola, David C. Montefiori, Dan H. Barouch, and Bette Korber. HIV-1 Neutralizing Antibody Signatures and Application to Epitope-Targeted Vaccine Design. *Cell Host and Microbe*, 2019.

[24] C. D. Murin, J.-P. Julien, D. Sok, R. L. Stanfield, R. Khayat, A. Cupo, J. P. Moore, D. R. Burton, I. A. Wilson, and A. B. Ward. Structure of 2G12 Fab2 in Complex with Soluble and Fully Glycosylated HIV-1 Env by Negative-Stain Single-Particle Electron Microscopy. *Journal of Virology*, 2014.

[25] Christopher O. Barnes, Harry B. Gristick, Natalia T. Freund, Amelia Escolano, Artem Y. Lyubimov, Harald Hartweger, Anthony P. West, Aina E. Cohen, Michel C. Nussenzweig, and Pamela J. Bjorkman. Structural characterization of a highly-potent V3-glycan broadly neutralizing antibody bound to natively-glycosylated HIV-1 envelope. *Nature Communications*, 2018.

[26] Marina Caskey, Florian Klein, Julio C.C. Lorenzi, Michael S. Seaman, Anthony P. West, Noreen Buckley, Gisela Kremer, Lilian Nogueira, Malte Braunschweig, Johannes F. Scheid, Joshua A. Horwitz, Irina Shimeliovich, Sivan Ben-Avraham, Maggi Witmer-Pack, Martin Platten, Clara Lehmann, Leah A. Burke, Thomas Hawthorne, Robert J. Gorelick, Bruce D. Walker, Tibor Keler, Roy M. Gulick, Gerd Fätkenheuer, Sarah J. Schlesinger, and Michel C. Nussenzweig. Viraemia suppressed in HIV-1-infected humans by broadly neutralizing antibody 3BNC117. *Nature*, 2015.

[27] J. E. Ledgerwood, E. E. Coates, G. Yamshchikov, J. G. Saunders, L. Holman, M. E. Enama, A. Dezure, R. M. Lynch, I. Gordon, S. Plummer, C. S. Hendel, A. Pegu, M. Conan-Cibotti, S. Sitar, R. T. Bailer, S. Narpala, A. McDermott, M. Louder, S. O'Dell, S. Mohan, J. P. Pandey, R. M. Schwartz, Z. Hu, R. A. Koup, E. Capparelli, J. R. Mascola, B. S. Graham, Floreliz Mendoza, Laura Novik, Kathy Zephir, William Whalen, Brenda Larkin, Olga Vasilenko, Nina Berkowitz, Brandon Wilson, Iris Pittman, Gretchen Schieber, Hope Decederfelt, Judith Starling, John Gilly, Srinivas Rao, Florence Kaltovich, Phyllis Renehan, Meghan Kunchai, Sarah Romano, Katie Menard, Ly Diep, Chuka Anude, and Mary Allen. Safety, pharmacokinetics and neutralization of the broadly neutralizing HIV-1 human monoclonal antibody VRC01 in healthy adults. *Clinical and Experimental Immunology*, 2015.

[28] Katarzyna Kotfis, Shawniqua Williams Roberson, Jo Ellen Wilson, Wojciech Dabrowski, Brenda T Pun, and E Wesley Ely. COVID-19: ICU delirium management during SARS-CoV-2 pandemic. 24(1), 2020.

[29] Home - Johns Hopkins Coronavirus Resource Center.

[30] Coronavirus Pandemic (COVID-19) - Statistics and Research - Our World in Data.

[31] Timothy A. Salthouse. When does age-related cognitive decline begin? *Neurobiology of aging*, 30:507–514, 2009.

[32] Archana Singh-Manoux, Mika Kivimaki, M. Maria Glymour, Alexis Elbaz, Claudine Berr, Klaus P. Ebmeier, Jane E. Ferrie, and Aline Dugravot. Timing of onset of cognitive decline: results from whitehall ii prospective cohort study. *BMJ (Clinical research ed.)*, 344, 1 2012.

[33] Tamar Gefen, Emily Shaw, Kristen Whitney, Adam Martersteck, John Stratton, Alfred Rademaker, Sandra Weintraub, M. Marsel Mesulam, and Emily Rogalski. Longitudinal neuropsychological performance of cognitive superagers. *Journal of the American Geriatrics Society*, 62:1598–1600, 2014.

[34] Tamar Gefen, Tamar Gefen, Melanie Peterson, Steven T. Papastefan, Adam Martersteck, Kristen Whitney, Alfred Rademaker, Alfred Rademaker, Eileen H. Bigio, Eileen H. Bigio, Sandra Weintraub, Sandra Weintraub, Emily Rogalski, M. Marsel Mesulam, M. Marsel Mesulam, and Changiz Geula. Morphometric and histologic substrates of cingulate integrity in elders with exceptional memory capacity. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 35:1781–1791, 1 2015.

[35] Theresa M. Harrison, Sandra Weintraub, M. Marsel Mesulam, and Emily Rogalski. Superior memory and higher cortical volumes in unusually successful cognitive aging. *Journal of the International Neuropsychological Society : JINS*, 18:1081–1085, 11 2012.

[36] Emily J. Rogalski, Tamar Gefen, Junzi Shi, Mehrnoosh Samimi, Eileen Bigio, Sandra Weintraub, Changiz Geula, and M. Marsel Mesulam. Youthful memory capacity in old brains: anatomic and genetic clues from the northwestern superaging project. *Journal of cognitive neuroscience*, 25:29–36, 2013.

CHAPTER 2. A NEW CLASSIFICATION METHOD BASED ON DYNAMIC ENSEMBLE SELECTION AND ITS APPLICATION TO PREDICT VARIANCE PATTERNS IN HIV-1 ENV

Mohammad Fili[1], Guiping Hu[1,2], Changze Han[3], Alexa Kort[3], John Trettin[3], Hillel Haim[3]

[1] Department of Industrial and Manufacturing Systems Engineering, Iowa State University, Ames, IA, USA.

[2] Department of Sustainability, Rochester Institute of Technology, Rochester, NY, USA.

[3] Department of Microbiology and Immunology, Carver College of Medicine, University of Iowa, Iowa City, IA, USA.

Modified from a manuscript under review in *Computer Methods and Programs in Biomedicine Update*

2.1 Abstract

Therapeutics that target the envelope glycoproteins (Envs) of human immunodeficiency virus type 1 (HIV-1) effectively reduce virus levels in patients. However, due to mutations, new Env variants are frequently generated, which may be resistant to the treatments. The appearance of such sequence variance at any Env position is seemingly random. A better understanding of the spatiotemporal patterns of variance across Env may lead to the development of new therapeutic strategies. We hypothesized that, at any time point in a patient, positions with sequence variance are clustered on the three-dimensional structure of Env. To test this hypothesis, we examined whether variance at any Env position can be predicted by the variance measured at adjacent positions. Sequences from 300 HIV-infected patients were applied to a new algorithm we developed. The k-best classifiers (KBC) method is a dynamic ensemble selection technique that identifies the best classifier(s) within the neighborhood of a new observation. It applies bootstrap

resampling to generate out-of-bag samples that are used with the resampled set to evaluate each classifier. For many positions of Env, primarily in the CD4-binding site, KBC accurately predicted variance based on the variance at their adjacent positions. KBC improved performance compared to the initial learners, static ensemble, and other baseline models. KBC also outperformed other algorithms for predicting variance at multi-position footprints of therapeutics on Env. These understandings can be applied to refine models that predict future changes in HIV-1 Env. More generally, we propose KBC as a new high-performance dynamic ensemble selection technique.

2.2 Introduction

Four decades after recognizing HIV-1 as the causative agent of acquired immune deficiency syndrome (AIDS), this virus is still a major health concern worldwide. In the year 2019, 38 million individuals were living with HIV, 690,000 died from AIDS-related disease, and 1.7 million were newly infected [1]. To treat HIV-infected individuals, multiple therapeutics are available; they bind to HIV-1 proteins and can effectively inhibit their function. However, the replication machinery of HIV-1 is highly prone to errors. As a result, new variants of its proteins are generated, some of which contain changes at the sites targeted by the therapeutics [2]. Subsequent expansion under the selective pressure of the therapeutic can lead to clinical resistance [3, 4]. Since the appearance of the mutations is random, the emergence of resistance by changes at any position of an HIV-1 protein is considered unpredictable. There is a critical need to better understand the patterns of change in HIV-1 within the host. Such knowledge can lead to the design of new strategies that tailor treatments to infected individuals based on the properties of the infecting virus and the changes expected to occur. Multiple tools have been developed over the past two decades to predict the evolution of other viruses, primarily influenza virus; they aim to forecast changes in structural properties of virus proteins that can inform the design of vaccines [5]. Unfortunately, the number of tools developed to model and predict the changes in HIV-1, and particularly within the host, is limited [6, 7, 8, 9]

2.2.1 Toward a Better Understanding of the Variance Patterns in HIV-1 Env Within the Infected Host

Of all HIV-1 proteins, the envelope glycoproteins (Envs) exhibit the highest level of diversity, both within and between hosts [10, 11]. Env adorns the surface of HIV-1 particles and allows the virus to enter cells [12]; it is thus a primary target in AIDS vaccine design [13]. Env is composed of approximately 850 amino acids (some diversity in length exists between different strains). In the infected host, new amino acid variants continuously appear at multiple positions of this protein. Consequently, at any time point during chronic infection, 10% or more of Env positions can exhibit variance in amino acid sequence between co-circulating strains [14, 15]. The random nature of the mutations, the extreme diversity of Env within and between hosts, and the structural complexity of this protein limit our ability to model the changes.

Whereas the amino acid sequence at any Env position can vary between strains in different hosts, the level of in-host variance in amino acid sequence at each Env position is similar in different individuals [6]. Variance at each Env position is specific for each subtype (clade) of HIV-1. Thus, patterns of variance in the host are not merely random "noise" but reflect the inherent properties of the virus. Variance describes the permissiveness of sites to contain amino acids with different chemical properties, which reflects the strength of the selective pressures applied to them. Such pressures are typically not applied on individual positions but on multi-position domains of Env. In this work, we investigate the spatial clustering of variance across the Env protein. Specifically, we test the hypothesis that the absence or presence of sequence variance at any position of Env can be predicted based on the variance at adjacent positions on the three-dimensional structure of the protein. If the tendency for co-variance of adjacent positions is stable over time, then the patterns of variance detected in a patient may provide insight into future changes. To test the above hypothesis, we developed a new algorithm that selects the best classifiers for a new observation using a dynamic mechanism.

2.2.2 Multiple Classifier Selection Algorithms

As the complexity of any dataset increases, the ability of any single classifier to capture all patterns is reduced. Therefore, it is necessary to integrate multiple classifiers for improved classification accuracy. However, the use of the same set of classifiers statically over the whole feature space can affect the overall performance of an algorithm. In fact, a classifier may perform well in some subspaces of the data but exhibit poor performance in others. One solution to this problem is the selection of the best classifiers out of a pool of existing learners dynamically based on some competitiveness criteria and to use this subset of classifiers for predicting the class label of a new observation in a region. Such a mechanism will help to select the best classifiers in each subspace.

A considerable amount of work has been conducted in the field of multiple classifier systems (MCS) during the past two decades. MCSs are divided into two basic categories: static and dynamic. Static classification techniques use the same classifiers over the whole range of the dataset, while dynamic classification methods use the best classifier(s) at local regions. The method we propose in this work is built upon the dynamic classification approach [16, 17]. Dynamic classification can be further divided into dynamic classifier selection (DCS) and dynamic ensemble selection (DES). The DCS approach uses one base classifier to predict class labels in a local region [18], while the DES methods use many base classifiers [19]. Here, we propose a DES-based classification technique.

DES methods are composed of three main steps: (i) classifier generation, (ii) ensemble selection, and (iii) classifier combination. For the first step, either heterogeneous classifiers [20, 21, 22] or homogeneous ones [23, 24, 25] can be used. The proposed method will use decision trees as the homogenous classifiers for its classifier generation phase. Since local accuracy is a key feature of the DES method, many algorithms use k-nearest neighbors as a framework [26, 19, 27]. Other methods for generating different homogeneous classifiers have been proposed, including random subspace [28], bagging [29], boosting [30], and clustering [31].

The second step in DES design is ensemble selection. Some selection mechanisms utilize probabilistic models that adopt the probability of correct classification as the measure of competence [32]. Alternatively, one can use a wide range of classifiers to increase diversity, which increases accuracy as a result [33]. Models based on accuracy are another type of selection method that considers the classifiers with the highest levels of accuracy in the local area [17].

The third step of a DES method is classifier combination, which aggregates the gathered information into a single class label. Dynamic classifier weighting is one aggregation technique [34, 35]. Artificial neural networks have also been adopted to aggregate the results of base learners and use a cost function to maximize accuracy [36]. The majority vote method has also been used, which applies predictions of the highest accuracy classifiers in a local region and then takes the majority of these predictions to determine its output [37].

In this study, we introduce a novel approach to use a dynamic mechanism to increase the flexibility of the algorithm. The best classifiers are chosen based on the performance of the initial learners in the neighborhood of a new observation. Bootstrapping has been adopted to increase randomness, thus introducing more diversity within the base learners. We also introduced a classifier scoring approach, upon which the selection decision of a classifier can be made. To define a neighborhood for a new observation, we used the k-nearest neighbors (KNN) algorithm. Each observation has a feature vector that is used for the neighborhood selection process. The novelty of this method is in the dynamic classifier selection approach, where we introduced a weighting mechanism to evaluate each classifier's performance within a neighborhood of an instance and decide if the classifier is good enough to be used for prediction. This approach is based on bootstrap resampling, which creates the out-of-bag sample that can be used along with the resampled data in the classifiers' evaluation process. We tested the KBC algorithm with a panel of sequences from HIV-1 infected individuals. These data describe for each patient the absence or presence of variance at each position on the three-dimensional structure of the Env protein. We examined whether the variance at each position (or at a group of positions) can be predicted based on variance at adjacent positions on the three-dimensional structure of the

protein. In many cases, the KBC method showed improvement in the classification metrics relative to other machine learning algorithms. Considerably higher performance was observed in the CD4-binding domain of Env, which is the target of multiple antibody therapeutics against HIV-1 [38, 39, 40, 41, 42]. The primary advantage of KBC is that it does not use the same set of classifiers for the entire problem space; instead, it identifies the subset of learners that are capable of better predicting class labels for the observations in a region. This flexibility helps to avoid the loss of helpful learners and to limit the retention of weak learners that occurs when a fixed number of classifiers is applied.

2.3 Materials and Method

2.3.1 Datasets

In this study, we developed a novel dynamic classification approach to predict the variance at any Env position based on variance at adjacent positions on the protein. Because of the folded structure of the protein, the physical distance between any two positions (in Ångstroms, Å) is used as the measure of proximity. These distances are based on the cryo-electron microscopy (cryo-EM) coordinates of the Env protein. A description of the datasets is provided in the following sections.

2.3.1.1 HIV-1 Env Sequence Data

Nucleotide sequences of the HIV-1 env gene were obtained from the National Center for Biotechnology Information (NCBI) database (`https://www.ncbi.nlm.nih.gov`) and from the Los Alamos National Lab (LANL) database (`https://www.hiv.lanl.gov`). Sequence data for HIV-1 clades B and C were downloaded and processed separately. The clade C dataset is composed of 1,960 sequences from 109 distinct patients. The clade B dataset is composed of 4,174 sequences from 191 distinct patients. For each patient sample, all Envs isolated were analyzed (6-30 sequences per sample). All env genes were cloned from the samples by the single genome amplification approach [43] and sequenced by the Sanger method. Sequences of non-functional

Envs were removed, as were all sequences with nucleotide ambiguities or large deletions in conserved regions [6, 44]. Nucleotide sequences were aligned using a Hidden Markov Model with the HMMER3 software [45] and then translated into the amino acid sequence, which was used for the analysis. All 856 Env positions described in the manuscript conform to the standard HXBc2 numbering of the Env protein [46]. Potential N-linked glycosylation sites (PNGSs) contain the sequence motif Asn-X-Ser/Thr, where X is any amino acid except Pro. To account for the presence of N-linked glycans on the Asn residues, the first position of all Asn-X-Ser/Thr triplets was assigned a unique identifier. All aligned sequences from each patient were compared to determine whether each of the 856 positions has variance in amino acid sequence (position is assigned a variance value of 1) or whether all sequences from that patient sample have the same amino acid at the position (assigned a variance value of 0).

2.3.1.2 Env Structural data

The Env protein is folded so that positions from different domains can be close in the three-dimensional structure of the protein. To determine the positions closest to each position of interest, we used the coordinates of the cryo-EM structure of Env. For clade B viruses, we used the coordinates of Env from HIV-1 clade B strain JRFL (Protein Data Bank, PDB ID 5FUU) [47]. For clade C viruses, we used the coordinates of Env from HIV-1 clade C strain 426c (PDB ID 6MZJ) [48]. The distance between any two positions was measured using the coordinates of the closest two atoms of the two amino acids. These data were applied to identify the 10 closest positions to each position of interest.

2.3.2 The KBC Method

In the KBC algorithm, available information is utilized regarding the base classifiers for the data points in the neighborhood of a new observation to predict the class labels. This allows us to identify the classifiers that perform well in the neighborhood of a new observation and apply only those learners for the prediction. Classifiers do not perform the same in different regions of the

problem space, especially when the data structure is complicated. This becomes even more tangible when more naïve learners are used. KBC applies the information from the training phase to identify those classifiers that are more helpful in finding the class label of a new instance based on their performance within a specific neighborhood. The foundation of this method, like other dynamic ensemble selection techniques, relies on three main steps: classifier generation, selection, and aggregation.

2.3.2.1 Classifier Generation

In the proposed method, we used the decision tree algorithm as the base learner. We selected this algorithm primarily because of its training speed. In general, for the first step, M base learners, $L_1, L_2, \ldots, L_M$, are generated. The number of base learners is a hyperparameter of the KBC model.

In the next step, to identify the best classifiers, we used bootstrap resampling. In this manner, we can reserve an untouched sample (i.e., the out-of-bag sample, OOB), which is used along with the resampled observations for the selection of the best classifiers. For the training set, X^{train}, we have $x_1, x_2, \ldots, x_N$ feature vectors (one for each observation), and $y_1, y_2, \ldots, y_N$ as their class labels. We denote the test set as X^{test}.

We use bootstrapping to train each base learner L_i. This approach creates two separate sets of observations for learner L_i: a resampled set (S_i^r) and an OOB set (S_i^{oob}). In other words, we make a partition:

$$X^{train} = (S_i^r \cup S_i^{oob}), \qquad \forall\ i = 1, 2, \ldots, M \tag{2.1}$$

$$S_i^r \cap S_i^{oob} = \varnothing, \qquad \forall\ i = 1, 2, \ldots, M \tag{2.2}$$

If we define an event A, as a data point x_j ($j = 1, 2, \ldots, N$) that belongs to the OOB sample:

$$A : x_j \in S_i^{oob}, \qquad \forall\ i = 1, 2, \ldots, M \tag{2.3}$$

then the probability of such event can be calculated as:

$$Pr(A) = (1 - \frac{1}{N})^N \approx e^{-1} \approx 0.368 \tag{2.4}$$

where, N is the total number of observations in the training set X^{train}. Later, we will show how to use this information in the algorithm as a starting point.

To introduce more randomness among the base learners, we used random sampling for feature selection. To accomplish this task, the algorithm randomly picks f features out of all available attributes for L_i. In other words, the learner L_i is trained over the subset of the features of S_i^r which is denoted by $S_i^{r,f}$. Knowing the set of f features for the learner L_i, one can also create $S_i^{oob,f}$ for the evaluation phase.

2.3.2.2 Classifier Selection

First, each base learner is used to predict all instances in X^{train}, including the resampled and OOB data. Then, the classification results are mapped onto a binary variable, z_{ij}, which is 1 or 0 based on whether the classifier L_i correctly classified the instance x_j or not, respectively:

$$z_{ij} = \begin{cases} 1 & \text{if } \hat{y}_{ij} = y_i \\ 0 & Otherwise \end{cases} \tag{2.5}$$

where, $i = 1, 2, .., M$ and $j = 1, 2, \ldots, N$ and $\hat{y}_{ij}$ is the class label that is predicted by the learner L_i for $x_j \in X^{train}$. The result of this phase is an $M \times N$ binary matrix Z, in which each row represents the mapped prediction result for one base learner, and each column corresponds to an observation in the training set, X^{train} :

$$Z = [z_{ij}]_{M \times N} \tag{2.6}$$

For efficiency, we perform this only once for all observations rather than during each iteration. In effect, not all observations will be used for selecting the best classifiers, but only the ones in the neighborhood of the new observation $x_q \in X^{test}$. To find the neighbors, different approaches can be applied, one of which is the KNN method. KNN finds the closest data points to the observation of interest. However, since it is based on the distance matrix, for very large datasets, the process can take much longer than a dataset with a small number of observations.

By defining Ψ_q^n as the neighborhood of a new data point x_q which includes n-closest observations, we can define:

$$\phi_q^n = \{j : \quad 1 \leq j \leq N,\ x_j \in \Psi_q^n\} \tag{2.7}$$

where, ϕ_q^n is the set of n indices for the data points within the neighborhood of the new instance x_q.

To emphasize the contribution of correctly predicting an observation in the OOB set relative to the resampled set, we can assign greater weights to the observations in the OOB set. In this manner, we can also define the level of emphasis by tweaking the weights. However, the optimal value can be found by hyperparameter tuning.

Weighting of the OOB and resampled sets can be described by:

$$W^{oob} + W^r = 1 \tag{2.8}$$

$$W^{oob}, W^r > 0 \tag{2.9}$$

where, W^{oob} and W^r are the weights for observations within the OOB (S_i^{oob}) and resampled sets (S_i^r) for learner L_i, respectively. From (2.4), we can conclude that the probability of a data point belonging to S_i^r is approximately 0.632. Therefore, we can use this value as the default W^{oob} ; however, the optimal value for this parameter can be obtained via hyperparameter tuning. In general, the higher the OOB weight, the greater the focus on the OOB observations rather than the resampled set.

Now, consider the matrix Π in which the type of data points (i.e., being from the OOB or resampled set) is stored:

$$\Pi = [\pi_{ij}]_{M \times N}, \qquad i = 1, 2, \ldots, M \ and \ j = 1, 2, \ldots, N. \tag{2.10}$$

where, π_{ij} is defined as:

$$z_{ij} = \begin{cases} W^{oob} & x_j \in s_i^{oob} \\ W^r & x_j \in s_i^r \end{cases} \tag{2.11}$$

where, $i = 1, 2, \ldots, M$ and $j = 1, 2, \ldots, N$. In the next step, the classifier's score, CS_i, can be calculated for base learner L_i:

$$CS_i = \sum_{j \in \phi_q^n} \pi_{ij} z_{ij} \qquad \forall\ i = 1, 2, \ldots, M \tag{2.12}$$

Then, we normalize the scores to the same scale:

$$CS_i' = \frac{CS_i - \min\limits_h CS_h + 1}{\max\limits_h CS_h - \min\limits_h CS_h + 1} \qquad \forall\ i = 1, 2, \ldots, M \tag{2.13}$$

where, $h = \{1, 2, \ldots, M\}$ is the set of all classifiers. This normalization rescales all scores into a range of $(0, 1]$, and facilitates the comparison:

$$0 < CS_i' \leq 1, \quad \forall\ i = 1, 2, \ldots, M \tag{2.14}$$

Here, CS_i' quantifies the relative importance of base learner L_i to the best classifier. For instance, if all learners perform the same in a region, then we expect a value of 1 for all (i.e., no distinction between the classifiers).

Figure 2.1 shows the relationship between the range of classifiers' scores with achievable normalized scores.

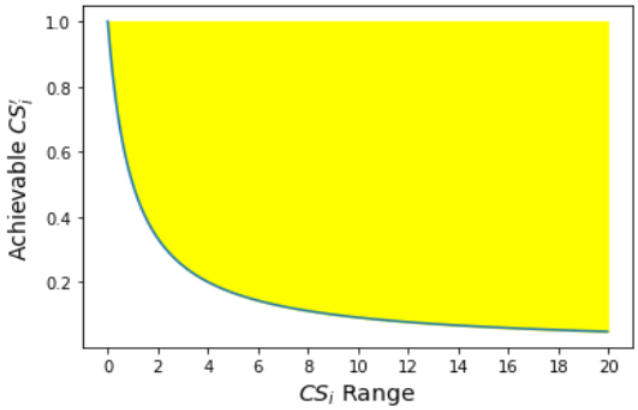

Figure 2.1 Achievable normalized scores (CS_i') for different values in the CS_i range (difference between the best and worst classifiers' scores). The yellow region shows all possible values.

For instance, if the difference between the maximum and minimum classifier's score is 1, then the normalized score of any classifier will be between 0.5 and 1. The yellow region in Figure 2.1 shows all possible normalized scores for any difference between the best and worst classifiers. The blue line on the left margin of the yellow region indicates the lower bound of the normalized score for any specific normalized score's range.

In the following equation, we can observe that, as the difference between the performance of the best and worst classifiers increases, there is greater confidence that classifiers with higher scores are performing significantly better relative to those with lower scores:

$$\lim_{Range \to \infty} \left(\max_i (CS_i^{'}) - \min_i (CS_i^{'}) \right) \to 1 \tag{2.15}$$

This occurs when the performance of the best and worst learners differs substantially. In such a case, we are more confident about the significance of the classifier with the high normalized score. As shown in Equation 2.15, for this extreme case, where the range approaches infinity, the limit of the difference between the best and worst normalized classifiers' score converges to the maximum value of 1. In other words, based on Equation 2.14, we can conclude that the best and worst classifiers are at the two ends of the interval. On the other hand, if the range of scores is 0 (i.e., all classifiers have the same performance), the normalized scores will be 1 for all of them (no distinction).

Finally, based on a minimum acceptance threshold δ, only those classifiers with scores higher than that value will be selected. Therefore, for different observations, we expect to have a different array of scores for base learners' performance within the neighborhoods. Thus, the algorithm will select the best classifiers based on the problem space, observations, and the base learners' capabilities to correctly classify similar instances each time. In Equation 2.16, the index corresponding to the k-best classifiers (out of M existing learners), for x_q, is defined.

$$K_q = \{i : \quad CS_i^{'} \geq \delta, \qquad 1 \leq i \leq M\} \tag{2.16}$$

The number of best classifiers can differ from observation to observation. However, for similar points (i.e., observations within a similar segment of the problem space), we expect to have a similar set of best classifiers for prediction.

2.3.2.3 Classifier Aggregation

Finally, once classifiers for the prediction are defined, we apply an aggregation method to obtain a single result for the new instance. Here we use the majority vote approach. For a general case in which we have P classes we can write:

$$\hat{y}_q = \arg\max_p \{c_p\}, \qquad p = 1, 2, \ldots, P \tag{2.17}$$

where, $\hat{y}_q$ is the predicted class for the new observation x_q, and c_p counts the number of base learners predicting class p. We can write it as:

$$c_p = \sum_{i \in K_q} 1_{\{\hat{y}_{iq} = p\}}, \qquad \forall\ p \in 1, 2, \ldots, P \tag{2.18}$$

where, $\hat{y}_{iq}$ is the class label predicted by learner L_i for x_q.

2.3.2.4 Hyperparameters

For the KBC algorithm, hyperparameters were designed to accommodate the variability in the dataset to ensure optimal performance. The hyperparameters of the algorithm are shown in Table 2.1. M is the number of base learners. If sufficient diversity exists within the base learners (i.e., among the decision trees generated), more learners typically lead to better results. We can also change the number of features (f) for each classifier. Using all available features can result in less randomness and reduces the level of diversity. On the other hand, using too few features, such as the extreme case of $f = 1$, can result in a naïve learner that may not be much better than the random guess. However, by using a suitable fraction of the available features for training the classifiers, we can add variation between the classifiers and also increase confidence that each classifier will perform well.

Table 2.1 Hyperparameters of the KBC algorithm

Parameter	Domain	Description
M	$\in \mathbb{N}$	Number of initial base learners
f	$\in \mathbb{N}$	Number of features to be selected randomly for each base learner
n	$\in \mathbb{N}$	Number of close neighbors to a new instance (similar instances)
W^{oob}	[0,1]	Weight of OOB instances. (Default=0.632)
δ	[0,1]	Minimum acceptance threshold for a base learner's score to be selected in a neighborhood of a new data point

The third hyperparameter is the number of neighbors for a new instance (n). Increasing the number of neighbors to all training observations (N) will lead to the majority vote for a fixed set of classifiers. In this case, we expect to have no variance but high bias. At the other extreme, if we use only one neighbor, the variance will be high. Therefore, it is a bias-variance trade-off, and selecting the optimal n is essential for performance. The effect of this parameter on the accuracy of the model is also explored in this study.

The weight of OOB instances (W^{oob}) plays an important role in emphasizing the unseen data for selecting the best classifiers. Since the data in the OOB set are not used during training of the base learners, predicting them correctly is more important than the observations in the resampled set. Choosing the weight as 1 will completely ignore the resampled data, whereas a weight of 0 will result in using only the resampled data in the training phase. The optimized value for the OOB weight can be obtained by hyperparameter tuning.

The last hyperparameter of the KBC model is the minimum acceptance threshold (δ), which determines the sensitivity in selecting the best classifiers. The higher the threshold, the smaller the number of learners we expect to have. In such a case, the variance may increase; however, at the same time, we are more confident about the set of selected classifiers in that region. On the other hand, using smaller values for δ may result in more learners, which reduces variance.

A summary of the KBC algorithm is shown in Figure 2.2. As explained, we start with the training set. By applying bootstrap resampling, we create OOB samples as well as a resampled

set for each learner. Then we train classifiers separately on their own resampled data. In the prediction phase, for each new data point, we first identify the closest neighbors. Then, based on whether a point belongs to the OOB or resampled set, the method assigns weights to the 0-1 mapping of the initial predictions (1 for correctly classifying an observation and 0 for misclassifying it). This approach introduces a scoring function that is used for evaluating the classifiers. Finally, according to a minimum threshold acceptance value, only those classifiers for which the normalized score exceeds the selected limit will be chosen for classifying the new instance. The individual predictions are then aggregated into a single result by the majority vote aggregation method. In Figure 2.3, we show the pseudo code for the entire KBC process.

2.3.3 Evaluation Metrics

In this study, 5 classification metrics are used: accuracy, precision, recall, F1 score, and balanced accuracy. Accuracy depicts the percent of predictions that are correct. Precision describes the percentage of correct classifications from the group of instances that are predicted as the positive group. Recall or sensitivity represents the correct classification rate from the group of true positive instances. The F1 score takes both the precision and recall metrics into consideration. Since the HIV-1 datasets are not balanced (i.e., for any position, the proportion of variance-positive and no-variance samples is not equal), we also used balanced accuracy, which is an average of sensitivity and specificity.

The formulae for these metrics are shown below:

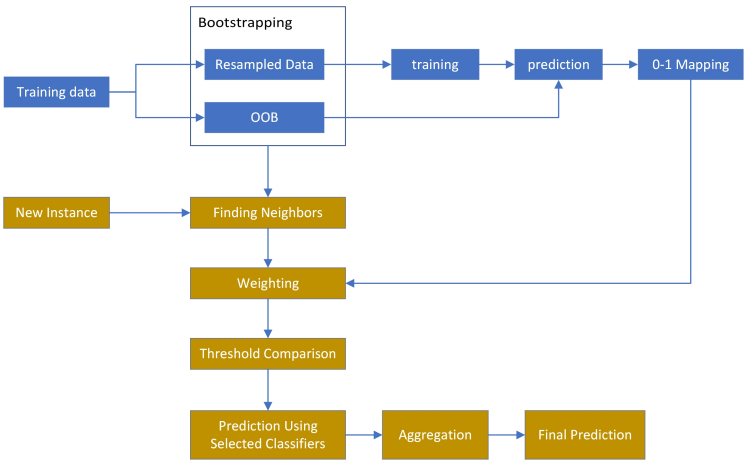

Figure 2.2 Flow chart of the KBC algorithm. It starts with the bootstrap resampling for each base learner. Then, for a neighborhood of a new data point, the weights are assigned to the 0-1 mapped values and then aggregated into a single score for each learner. Those classifiers surpassing the minimum threshold are selected for the classification of the new observation.

KBC Algorithm

1. Split the data randomly into X^{train}, X^{test}.
2. Select M base learners $L_1, L_2, \ldots, L_M$.

Training Stage:

3. For each L_i:
   - 3.1. Do bootstrap resampling from X^{train} and partition it into S_i^r, S_i^{oob}.
   - 3.2. Select f features randomly from available features and make $S_i^{r,f}$, $S_i^{oob,f}$.
   - 3.3. Train L_i on the $S_i^{r,f}$
4. Obtain the weight matrix Π
5. Obtain prediction mappings matrix Z

Prediction Stage:

6. For $x_q \in X^{test}$:
   - 6.1. Define Ψ_q^n as the neighborhood of x_q containing n similar (closest) neighbors.
   - 6.2. Find $\varphi_q^n = \{j: \quad 1 \le j \le N \ , \ x_j \in \Psi_q^n\}$
   - 6.3. For each L_i:
     - 6.3.1. Calculate $CS_i = \sum_{j \in \varphi_q^n} \pi_{ij} z_{ij}$
     - 6.3.2. Calculate $CS_i' = \frac{CS_i - \min_h CS_h + 1}{\max_h CS_h - \min_h CS_h + 1}$
   - 6.4. Obtain the indices for the best classifiers: $K_q = \{i: \ CS_i' \ge \delta \ , \ 1 \le i \le M\}$
   - 6.5. Predict: $\hat{y}_q = Argmax_p\{c_p\}$ where $c_p = \sum_{i \in K_q} 1_{\{\hat{y}_{iq} = p\}}$

Figure 2.3 Pseudo code for the KBC algorithm.

$$Accuracy = \frac{tp + tn}{tp + tn + fp + fn} \tag{2.19}$$

$$Precision = \frac{tp}{tp + fp} \tag{2.20}$$

$$Recall = \frac{tp}{tp + fn} \tag{2.21}$$

$$F1\ score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{2.22}$$

$$BalancedAccuracy = \frac{1}{2} \times \left(\frac{tp}{tp + fn} + \frac{tn}{tn + fp}\right) \tag{2.23}$$

For any position of Env, we distinguish between two states: **(i)** Variance-positive (variance in amino acid sequence at the position is greater than 0), and **(ii)** No-variance (all sequences from the patient have the same amino acid at the position). Accordingly, our definitions are:

- *tp*: Number of patients with a variance-positive position that are predicted as variance-positive.
- *fp*: Number of patients with a no-variance position that are predicted as variance-positive.
- *tn*: Number of patients with a no-variance position that are predicted as no-variance.
- *fn*: Number of patients with a variance-positive position that are predicted as no-variance.

2.4 Results and Discussion

2.4.1 Overview of the Approach

In this study, we tested the algorithm with patient-derived datasets from two HIV-1 subtypes. Specifically, we examined the ability of the algorithm to predict the level of variance at any position of Env based on the variance at the 10 closest positions on the three-dimensional structure of the protein. Initially, we tested performance for individual positions. We then proceeded to test performance for multi-position targets that represent the "footprint" of therapeutics on the Env protein.

We first compared the results of the KBC algorithm with those of the base learner (decision tree) and static ensemble (random forest). To increase the stringency of the comparison, we tuned both the decision tree and random forest algorithms for optimal performance. The shared hyperparameters between these models are the maximum depth of the trees and the minimum samples in the leaf nodes. Besides, we considered splitting criteria for the decision tree and the number of estimators for the random forest as the other hyperparameters for tuning. The grid search strategy was used to tune these models. Finally, we compared the performance of KBC with other classification techniques.

To estimate the evaluation metrics for each machine learning technique and each dataset, we used nested cross-validation to tune the hyperparameters and to obtain estimates of the classification metrics. We used k=5 for both the inner and outer folds.

2.4.2 Prediction of variance patterns in HIV-1 Env

New forms of the Env protein are continuously generated in HIV-infected individuals by the error-prone replication machinery of this virus. The appearance of new amino acid variants at Env positions targeted by therapeutics can lead to virus resistance to their effects. Such events appear to be random and thus considered unpredictable. There is a clinical need to understand the spatiotemporal patterns of variance in the HIV-infected host, which may lead to the development of new treatment strategies. We hypothesized that at any time in the infected host, positions that exhibit variance in amino acid sequence are spatially clustered on the Env protein. Such patterns are intuitive since the immune and fitness pressures mostly act on multi-position domains of Env rather than individual positions. Toward a better understanding of such patterns, we sought to determine whether the variance at any Env position can be accurately estimated based on the variance at adjacent positions on the protein.

To this end, we applied the KBC algorithm with sequences of Envs cloned from patient blood samples (6-30 Envs sequences were available and used for each sample). All sequences from each patient sample were aligned and compared to determine the absence or presence of in-host

variance at each of the 856 positions of Env. As detailed below, we focused our analyses on Env positions targeted by therapeutics. The response variable is thus the absence or presence of variance at specific key positions of Env. The explanatory variable is the variance at the 10 positions closest to the key position on the protein (determined by the physical distance between the closest atoms of the two positions, measured in Ångtroms). The goal is to correctly classify the variance of a position using the variance of the adjacent positions. We decided to use the 10 adjacent positions since this is approximately the maximal number of amino acids that can contact the position of interest on the three-dimensional structure of Env.

We note that the actual number of residues that are in contact or adjacent to each position may vary according to the location on the protein. For example, for any position buried within the core of the protein, its 10 nearest positions will be closer than for a position located on a loop that is exposed to the solvent. Nevertheless, we decided that as a first step, we will maintain this variable constant for all positions.

We first tested the ability of the KBC algorithm to predict the absence or presence of variance at individual positions in the high-mannose patch of Env (Figure 2.4). These N-linked glycans help to shield Env from recognition by host antibodies [49]; however, they also serve as targets for microbicidal agents such as lectins [50, 51] and therapeutic antibodies [52, 53]. We tested three positions in the high-mannose patch, namely positions 289, 332, and 339. These positions form part of the target sites for multiple agents that inhibit HIV-1, including antibodies 2G12, 10-1074, PGT135, PGT128, and DH270.5 [54, 55, 56, 57, 58], and the lectin microbicide griffithsin [59]. Data are composed of 1,960 amino acid sequences from 109 patients infected by HIV-1 subtype C, which is the most prevalent HIV-1 clade worldwide [60]. For position 289, the ratio of the variance-positive class to the no-variance class was 34:75. This ratio for positions 332 and 339 was 46:63 and 53:56, respectively.

For positions 289 and 339, the results of the KBC analyses showed improvement relative to the base learner (decision tree) and random forest (Figure 2.5). By contrast, the prediction of variance at position 332 by the KBC method was similar to that of the other methods.

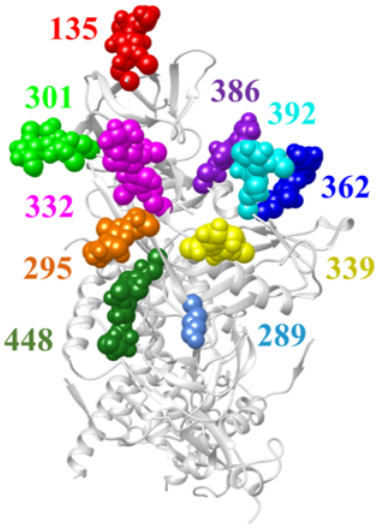

Figure 2.4 Cryo-EM structure of HIV-1 Env (side view, PDB ID 5FUU). Positions in the high-mannose patch are shown as spheres and labeled by Env position number. All positions shown contain N-linked glycans, except position 289, which contains Arg in the Env of HIV-1 isolate JRFL used to generate this structure.

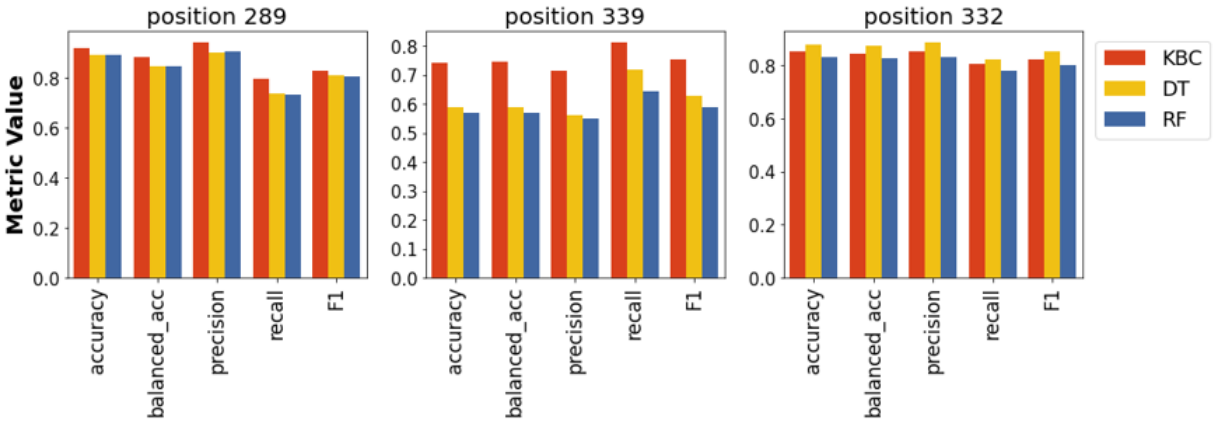

Figure 2.5 Predictions of variance at positions 289, 339, and 332 of Env using data from patients infected by HIV-1 clade C.

We also compared the performance of KBC with other machine learning algorithms (Table 2.2). Again, we observed modestly better performance of KBC for positions 289 and 339, whereas for position 332, the performance was similar to (or slightly worse than) other methods. We note that although KBC generally exhibited better point estimates than other methods, it also exhibited relatively high variability (see values in parentheses in Table 2.2).

Table 2.2 Prediction of variance at Env positions in the high-mannose patch by KBC and other algorithms.

Position	Method[a]	Balanced Accuracy[b]	Accuracy	Precision	Recall	F1 Score
289	KBC	0.88 (±0.13)	0.92 (±0.08)	0.94 (±0.07)	0.80 (±0.26)	0.83 (±0.20)
	QDA	0.83 (±0.01)	0.88 (±0.01)	0.90 (±0.07)	0.71 (±0.05)	0.79 (±0.01)
	LDA	0.87 (±0.02)	0.91 (±0.01)	0.93 (±0.05)	0.76 (±0.05)	0.84 (±0.03)
	NB	0.71 (±0.18)	0.69 (±0.26)	0.66 (±0.26)	0.76 (±0.05)	0.67 (±0.16)
	ADA	0.86 (±0.01)	0.90 (±0.01)	0.91 (±0.07)	0.76 (±0.05)	0.83 (±0.02)
	LogReg	0.86 (±0.01)	0.90 (±0.01)	0.91 (±0.07)	0.76 (±0.05)	0.83 (±0.02)
	SVM	0.83 (±0.01)	0.88 (±0.01)	0.90 (±0.07)	0.71 (±0.05)	0.79 (±0.01)
339	KBC	0.75 (±0.12)	0.74 (±0.12)	0.71 (±0.12)	0.81 (±0.13)	0.75 (±0.11)
	QDA	0.59 (±0.07)	0.59 (±0.07)	0.56 (±0.09)	0.72 (±0.09)	0.63 (±0.07)
	LDA	0.57 (±0.03)	0.57 (±0.03)	0.55 (±0.05)	0.64 (±0.13)	0.59 (±0.06)
	NB	0.65 (±0.08)	0.65 (±0.07)	0.64 (±0.00)	0.68 (±0.16)	0.66 (±0.19)
	ADA	0.60 (±0.02)	0.60 (±0.02)	0.59 (±0.03)	0.62 (±0.06)	0.59 (±0.02)
	LogReg	0.59 (±0.06)	0.58 (±0.07)	0.54 (±0.05)	0.94 (±0.05)	0.69 (±0.03)
	SVM	0.65 (±0.04)	0.66 (±0.05)	1.00 (±0.09)	0.30 (±0.04)	0.48 (±0.02)
332	KBC	0.85 (±0.07)	0.85 (±0.07)	0.86 (±0.12)	0.80 (±0.08)	0.83 (±0.08)
	QDA	0.84 (±0.05)	0.84 (±0.06)	0.84 (±0.12)	0.83 (±0.11)	0.82 (±0.06)
	LDA	0.87 (±0.03)	0.88 (±0.03)	0.89 (±0.05)	0.83 (±0.06)	0.85 (±0.04)
	NB	0.61 (±0.16)	0.57 (±0.21)	0.59 (±0.23)	0.91 (±0.13)	0.67 (±0.10)
	ADA	0.87 (±0.03)	0.88 (±0.03)	0.89 (±0.05)	0.83 (±0.06)	0.85 (±0.04)
	LogReg	0.82 (±0.09)	0.81 (±0.12)	0.80 (±0.18)	0.87 (±0.11)	0.81 (±0.08)
	SVM	0.88 (±0.02)	0.89 (±0.02)	0.89 (±0.05)	0.85 (±0.03)	0.87 (±0.03)

[a] Calculations were performed using data from 109 patients infected by HIV-1 clade C.
[b] Standard deviation values are indicated in parentheses.

This likely occurred due to the relatively small size of the dataset. Later, we show that increasing the size of the dataset drastically reduces the variability of the estimates.

Antiviral therapeutics bind to targets composed of multiple residues; their "footprint" on the viral protein can span a large surface that contains multiple amino acids [61, 62, 63, 64]. Changes at any of these contacts may reduce Env recognition by the therapeutic and cause resistance. We

examined the performance of the KBC algorithm to predict variance in a combined feature composed of 10 positions in the high-mannose patch shown in Figure 2.4. To this end, for each position A_i in the high-mannose patch ($i = 1, 2, \ldots, 10$), we relabeled its 10 adjacent positions as $v_{(1)}^{A_i}, v_{(2)}^{A_i}, \ldots, v_{(10)}^{A_i}$, where $v_{(j)}^{A_i}$ is the variance at the j-th adjacent position to A_i ($j = 1, 2, \ldots, 10$). For each j, we then combined the $v_{(j)}^{A_i}$ values of the 10 A_i positions.

We first used the dataset of sequences from 109 HIV-1 clade C infected individuals. Results were compared between the KBC method and the above machine learning methods. The ratio of positive-variance to the no-variance instances for this dataset was 450:640. Remarkably, KBC performed better than all models to predict sequence variance in the high-mannose patch (Table 2.3).

Table 2.3 Prediction of variance in the high-mannose patch of Env by KBC and other algorithms.

Clade	Method	Balanced Accuracy[a]	Accuracy	Precision	Recall	F1 Score
Clade C	**KBC**	0.65 (±0.05)	0.69 (±0.05)	0.67 (±0.08)	0.47 (±0.05)	0.55 (±0.08)
	DT	0.59 (±0.05)	0.63 (±0.07)	0.54 (±0.07)	0.39 (±0.21)	0.43 (±0.17)
	RF	0.59 (±0.02)	0.63 (±0.03)	0.60 (±0.05)	0.34 (±0.13)	0.42 (±0.10)
	QDA	0.60 (±0.02)	0.63 (±0.04)	0.57 (±0.02)	0.39 (±0.16)	0.45 (±0.12)
	LDA	0.58 (±0.04)	0.62 (±0.06)	0.54 (±0.07)	0.39 (±0.17)	0.44 (±0.13)
	NB	0.61 (±0.06)	0.63 (±0.08)	0.54 (±0.08)	0.52 (±0.21)	0.52 (±0.14)
	ADA	0.50 (±0.05)	0.44 (±0.04)	0.42 (±0.02)	0.88 (±0.09)	0.56 (±0.02)
	LogReg	0.57 (±0.06)	0.61 (±0.04)	0.55 (±0.07)	0.33 (±0.20)	0.38 (±0.15)
	SVM	0.58 (±0.04)	0.62 (±0.06)	0.56 (±0.06)	0.35 (±0.23)	0.40 (±0.17)
Clade B	**KBC**	0.65 (±0.02)	0.74 (±0.02)	0.68 (±0.06)	0.39 (±0.03)	0.50 (±0.04)
	DT	0.60 (±0.05)	0.70 (±0.02)	0.61 (±0.03)	0.29 (±0.16)	0.36 (±0.16)
	RF	0.62 (±0.00)	0.72 (±0.00)	0.61 (±0.02)	0.35 (±0.02)	0.45 (±0.01)
	QDA	0.63 (±0.01)	0.73 (±0.01)	0.64 (±0.03)	0.36 (±0.02)	0.46 (±0.02)
	LDA	0.64 (±0.01)	0.72 (±0.01)	0.61 (±0.01)	0.40 (±0.04)	0.48 (±0.03)
	NB	0.67 (±0.04)	0.70 (±0.03)	0.53 (±0.05)	0.59 (±0.07)	0.56 (±0.06)
	ADA	0.41 (±0.05)	0.29 (±0.02)	0.28 (±0.03)	0.74 (±0.14)	0.40 (±0.05)
	LogReg	0.64 (±0.02)	0.72 (±0.01)	0.61 (±0.02)	0.39 (±0.05)	0.47 (±0.03)
	SVM	0.63 (±0.02)	0.70 (±0.02)	0.55 (±0.03)	0.41 (±0.01)	0.47 (±0.02)

[a] Standard deviation values are indicated in parentheses

To validate these results, we examined the ability of KBC to predict variance in a second independent panel of sequences derived from individuals infected by HIV-1 clade B. This clade is the most prevalent in the United States and Europe [60]. Sequences from 191 patients were tested

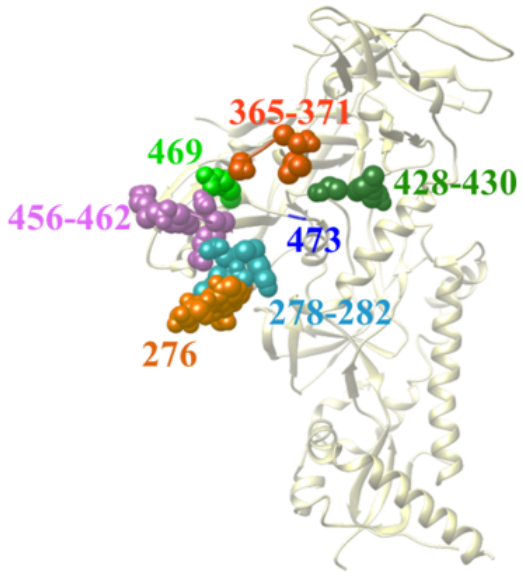

Figure 2.6 Side view of the cryo-EM structure of HIV-1 Env (PDB ID 5FUU). Positions in the CD4-binding site contacted by antibodies 3BNC117 and VRC01 are shown as spheres and labeled.

to predict variance at the multi-position high-mannose patch using the different algorithms. Consistent with the data shown for clade C, the performance of KBC was superior, albeit modestly, to that of the other algorithms (Table 2.3). The ratio of the positive-variance class to the no-variance class for the clade B dataset was 621:1289.

We further expanded our studies to test a second clinically significant domain of the Env protein, namely the CD4-binding site. This domain interacts with the receptor for the virus, which allows entry of the viral genome into the cell [65]. Since this site is conserved among diverse HIV-1 strains, it also serves as a target for multiple therapeutics, including the small molecule Fostemsavir [39] and antibody therapeutics VRC01 and 3BNC117 [38, 41]. We tested a combination of the 23 positions that serve as the contact sites for both antibodies VRC01 and 3BNC117 (Figure 2.6). We applied the same procedure explained for the high-mannose patch positions to combine the positions of the CD4-binding site.

Table 2.4 Prediction of variance in the CD4-binding site of Env using KBC and other algorithms

Clade	Methods	Balanced Accuracy[a]	Accuracy	Precision	Recall	F1 Score
Clade C	KBC	0.71 (±0.01)	0.85 (±0.01)	0.81 (±0.07)	0.45 (±0.03)	0.58 (±0.02)
	DT	0.61 (±0.13)	0.76 (±0.03)	0.32 (±0.23)	0.34 (±0.42)	0.25 (±0.25)
	RF	0.56 (±0.04)	0.76 (±0.03)	0.49 (±0.08)	0.19 (±0.18)	0.22 (±0.14)
	QDA	0.59 (±0.07)	0.75 (±0.04)	0.50 (±0.08)	0.29 (±0.28)	0.28 (±0.15)
	LDA	0.69 (±0.11)	0.79 (±0.03)	0.59 (±0.09)	0.50 (±0.31)	0.46 (±0.19)
	NB	0.67 (±0.09)	0.73 (±0.09)	0.45 (±0.11)	0.58 (±0.31)	0.45 (±0.16)
	ADA	0.46 (±0.18)	0.62 (±0.24)	0.44 (±0.29)	0.15 (±0.10)	0.20 (±0.14)
	LogReg	0.65 (±0.12)	0.79 (±0.01)	0.56 (±0.04)	0.39 (±0.33)	0.38 (±0.21)
	SVM	0.64 (±0.11)	0.76 (±0.03)	0.50 (±0.06)	0.43 (±0.35)	0.37 (±0.16)
Clade B	KBC	0.69 (±0.02)	0.89 (±0.01)	0.78 (±0.02)	0.40 (±0.04)	0.53 (±0.04)
	DT	0.54 (±0.02)	0.82 (±0.02)	0.33 (±0.02)	0.13 (±0.08)	0.17 (±0.10)
	RF	0.53 (±0.02)	0.82 (±0.03)	0.35 (±0.18)	0.11 (±0.05)	0.15 (±0.06)
	QDA	0.56 (±0.03)	0.82 (±0.04)	0.42 (±0.16)	0.17 (±0.11)	0.21 (±0.09)
	LDA	0.58 (±0.05)	0.82 (±0.01)	0.38 (±0.05)	0.24 (±0.13)	0.28 (±0.10)
	NB	0.64 (±0.08)	0.75 (±0.10)	0.34 (±0.14)	0.47 (±0.21)	0.37 (±0.13)
	ADA	0.48 (±0.03)	0.17 (±0.02)	0.15 (±0.01)	0.93 (±0.10)	0.26 (±0.02)
	LogReg	0.56 (±0.04)	0.84 (±0.00)	0.42 (±0.05)	0.17 (±0.10)	0.23 (±0.11)
	SVM	0.55 (±0.04)	0.82 (±0.03)	0.35 (±0.15)	0.17 (±0.11)	0.21 (±0.10)

[a] Standard deviation values are indicated in parentheses

The ratio of positive-variance to the no-variance classes for the CD4-binding site dataset was 685:3708 and 557:1950 for clades B and C, respectively. The performance of KBC was compared with all other algorithms tested above. Interestingly, the performance of the KBC algorithm was considerably higher than that of other algorithms (Table 2.4). For positions in the CD4-binding site, the increase in performance was better than that observed for positions in the high-mannose patch (Table 2.3). Comparison of the results in Table 2.3 and Table 2.4 shows that the variability of the estimates was considerably lower when we analyzed a group of positions rather than individual positions. Thus, for the CD4-binding site, the standard deviation in accuracy, balanced accuracy, recall, and F1 score obtained by KBC is the smallest among all other models for clade C. Indeed, KBC shows higher point estimates as well as smaller standard deviation values for the estimates.

Taken together, these findings show that when Env positions are tested individually, KBC outperforms other algorithms for most (but not all) positions. Nevertheless, this algorithm shines

in its performance when tested with a combination of positions that describe the complex (multi-position) target sites of the therapeutics on the Env protein.

2.4.3 KBC Hyperparameters Analysis

We examined the effects of two important hyperparameters of the KBC model on its performance. Data that describe variance patterns in the high-mannose patch were used. To evaluate the performance, we used the balanced accuracy metric. We explored the effect of one hyperparameter while maintaining the rest at a constant level. We used 20 decision trees (M=20); for each, we picked 4 features randomly (f=4), and the maximum depth was set to be 4. The OOB weight was fixed for both experiments at its default value, 0.632.

First, we explored the effect of the minimum acceptance threshold (δ). For this experiment, the number of neighbors was set to 10. The experiment was conducted with a variety of thresholds from 0 to 1, and the balanced accuracy was calculated. We observed that for the clade B dataset, increasing the minimum acceptance threshold improved the performance of the KBC model (Figure 2.7A). For the clade C dataset, the performance also increased gradually; however, it peaked at a threshold of 0.65, followed by a modest reduction (Figure 2.7B). These findings suggest that increasing the value for δ results in an overall increase in performance due to the higher confidence in the set of selected classifiers. However, in some cases, further increases in δ may result in loss of useful learners that can reduce overall performance.

We also explored the effect of the neighborhood size on the performance of the KBC algorithm. Here we used δ=0.8 as the minimum acceptance threshold. Different numbers of neighbors (ranging between 1 and 100) were tested to examine the effect of neighborhood size on the performance of the model. We observed that for both clades B and C, increasing the number of neighbors up to approximately 15 or 20 increased the performance (Figure 2.7C, and D). Further increases in the neighborhood size decreased the performance in clade B, whereas it did not impact clade C. These findings suggested that a neighborhood size of approximately 15 is

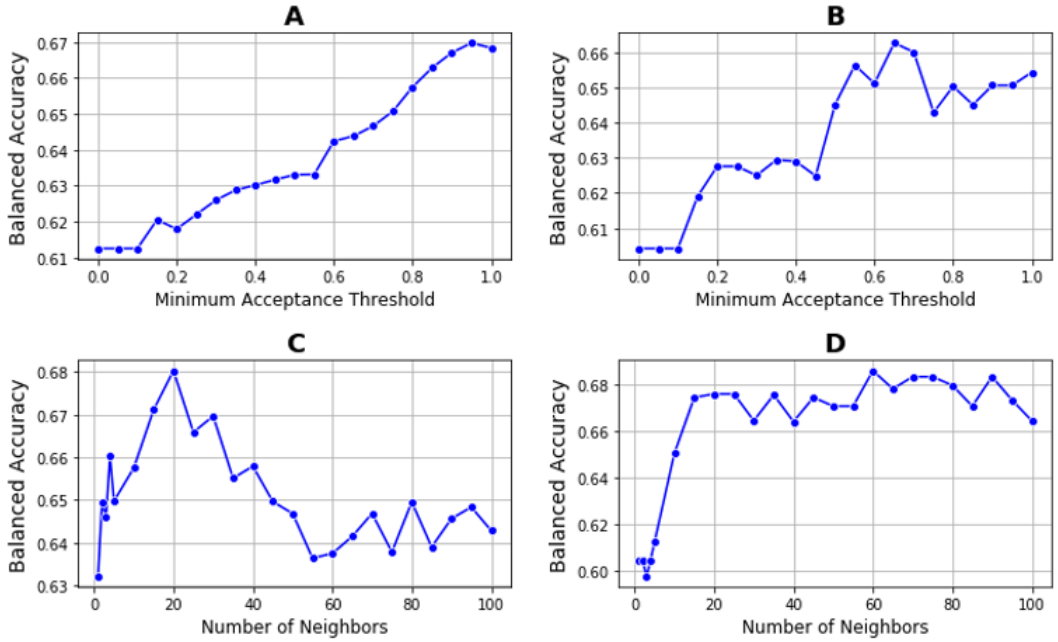

Figure 2.7 (A) Effect of the minimum acceptance threshold on the balanced accuracy using data that describe variance patterns in the high-mannose patch in clade B (B) Effect of the minimum acceptance threshold on the balanced accuracy using data that describe variance patterns in the high-mannose patch in clade C, (C) Effect of the neighborhood size on the balanced accuracy using data that describe variance patterns in the CD4-binding site in clade B, and (D) Effect of the neighborhood size on the balanced accuracy using data that describe variance patterns in the CD4-binding site in clade C.

optimal for the data that describe variance patterns in high-mannose patch given the above hyperparameters.

2.4.4 Effect of Base Learners in the KBC Algorithm

As a further analysis, we examined whether the choice of base learner in the KBC algorithm affects the overall performance of the method. To this end, we used logistic regression and Naïve Bayes (separately) as the base learners. We evaluated the KBC method using data from HIV-1 clade C that describe variance patterns in the high-mannose patch and the CD4-binding site. These results were compared with the results obtained using decision tree as the base learner. In

this experiment, we kept the structure of the KBC algorithm as before, with the exception that for each trial a homogenous set of base learners from one type was utilized (i.e., decision tree, logistic regression, or Naïve Bayes). For the tuning process and for the KBC with logistic regression as the base learner, we incorporated the hyperparameter C, which is the inverse of the regularization strength. For the KBC method with Naïve Bayes as the base learner, no hyperparameter was added to the KBC's hyperparameter list.

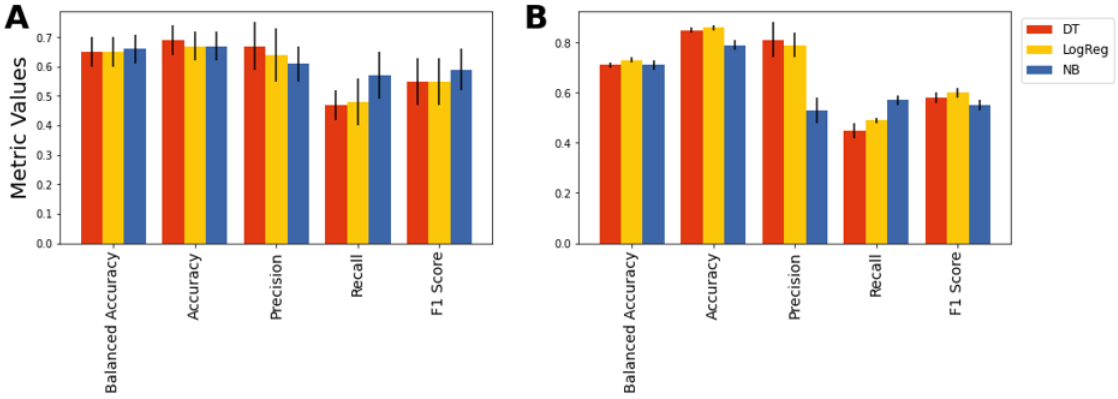

Figure 2.8 Performance of the KBC algorithm using decision tree, logistic regression, and Naïve Bayes as the base learners with data from HIV-1 clade C that describe variance patterns in the high-mannose patch (A) and the CD4-binding site (B). Error bars indicate standard deviation.

Results of the above tests are shown in Figure 2.8. For the high-mannose patch (Figure 2.8A), decision tree yielded modestly higher point estimates for accuracy and precision, whereas Naïve Bayes showed modestly better recall and F1 score. However, these differences were not statistically significant (see error bars in Figure 2.8). Therefore, for this dataset, the choice of base learner did not impact the performance of the KBC method. For the CD4-binding site (Figure 2.8B), decision tree and logistic regression performed equally well as the base learners and were both better than Naïve Bayes in accuracy and precision metrics. Similar to the high-mannose-patch data, the recall was better for Naïve Bayes; however, this improvement was not sufficient to counterbalance the considerably lower precision, resulting in an F1 score for Naïve Bayes that was modestly smaller than that of decision tree and logistic regression.

In summary, KBC is a general framework in which any choice of base learners can be used. As shown in Figure 2.8, the choice of base learner may affect the performance of the KBC algorithm. These effects are likely specific for each problem. In this study, we utilized decision tree as the base learner due to its speed and performance that was at least as good as other options.

2.5 Conclusions

Many viruses, including HIV-1, exhibit a high error rate during their replication [66, 67]. New variants of their proteins are continuously generated in the host. The ability to create diversity allows viruses to rapidly adapt to selective pressures, including antiviral therapeutics. The first step in the emergence of resistance is the appearance of sequence variance at a position of the viral protein targeted by the therapeutic. Variance patterns across the Env protein seem random and are thus considered unpredictable. In these studies, we examined whether positions that exhibit sequence variance are spatially clustered on the three-dimensional structure of the HIV-1 Env protein. Specifically, we tested whether the absence or presence of sequence variance at any position of Env in a patient can be predicted by variance at adjacent positions on the protein. To address this question, we developed a new dynamic ensemble selection algorithm based on bootstrap resampling.

We used the KBC method, which defines the neighborhood of a new data point using the KNN algorithm. Specifically, for each position of interest, KBC defines the neighborhood by identifying observations that have similar feature vectors (i.e., a similar variance profile of the 10 adjacent positions). We then selected the k-best classifier(s) within that neighborhood based on a weighted score procedure. By comparing each classifier's score with a minimum acceptance threshold, we obtain the set of best classifiers to predict the class label for each new instance. The dynamism, along with the specific design, resulted in a flexible approach that is not constrained to select a constant number of learners every time it wants to classify a new observation. Therefore, based on the performance of the learners, only those classifiers surpassing an explicit expectation are chosen; this results in an improvement in the overall performance. The novelty of

this algorithm is in the dynamic classifier selection mechanism, in which we designed a weighting procedure to evaluate each classifier's performance within a neighborhood of an instance and decide if the classifier is good enough to classify the observation. This approach is based on bootstrap resampling, which creates out-of-bag samples that can be used along with the resampled data in the classifiers' evaluation process.

We applied the algorithms to predict the level of variance at individual positions of Env based on variance at adjacent positions on the molecule. Results were compared with a variety of state-of-art methods, such as the Adaboost, Naïve Bayes, logistic regression, linear and quadratic discriminant analysis methods, and SVM. Overall, the KBC algorithm predicted the absence or presence of variance better than the above machine learning tools. Interestingly, performance varied with the domain of Env tested. Only modest enhancement of performance by the KBC method was observed for the high-mannose patch of Env, whereas dramatic enhancement was observed for the CD4-binding site. Improvement in all classification metrics was observed.

Importantly, KBC showed considerable improvement for predicting variance at multi-position features. We tested two Env domains targeted by therapeutics; the CD4-binding site and the high-mannose patch of Env (composed of 23 and 10 positions, respectively). Both domains constitute targets for multiple HIV-1 therapeutics [54, 55, 38, 56, 41, 57, 58]. These antibody footprints on Env were analyzed using sequence data from patients infected by HIV-1 clades B and C, which were analyzed separately. For both domains and in both clades, the absence or presence of variance was predicted better using KBC than other algorithms. These results are highly encouraging since therapeutics do not recognize single positions but rather multi-position footprints on the protein; a change at any position can reduce binding of the agents and increase clinical resistance [62, 63, 68]. The ability to predict the variance in a given domain based on the adjacent sites suggests that if these associations are stable over time, they may provide insight into future changes that may occur based on the current patterns of variance in the patient. Interestingly, for small datasets (e.g., analysis of single Env positions), KBC exhibited high point estimates but also high variability. By contrast, using larger datasets (e.g., multi-position

targets), KBC exhibited both higher estimates and also smaller variability compared to the other algorithms. This finding suggested that the KBC model works better with large datasets.

We observed that despite using homogenous and simple learners, KBC competes well with even sophisticated algorithms such as SVM, Adaboost, and discriminant analysis techniques. We also evaluated the effects of using logistic regression and Naïve Bayes as the base learners. Differences in the performance of KBC with each of these base learners were explored. Our results suggested that the choice of base learner may impact the overall performance; however, the effects are likely specific for each problem. We selected to focus most of our studies on decision tree as the base learner because of its relative speed and its performance, which was at least as high as that of the other options. Nevertheless, we note that by using more advanced methods as the base learner and by increasing diversity using a pool of different methods, KBC may exhibit even higher performance, which can be explored in future studies.

It should be noted that this study is subjected to a few limitations which suggest future research directions. First, the entire training dataset needs to be scanned for every new instance to find the neighbors with KNN. This may lead to computational intractability for very large datasets. Innovative methods for defining the neighborhood can be studied to improve efficiency. An example of such methods could be clustering algorithms that group similar instances into the same clusters [31]. Then, for each new observation, one can find the most similar cluster to the new data point and evaluate the classifiers within that neighborhood. Second, although the algorithm is capable of accommodating various configurations for the base learners, adding a high number of sophisticated methods can result in an increase in the training time for the model. Therefore, optimizing the choices for base learners would be necessary to balance running time with classification performance. Finally, we observed that for small datasets, KBC shows higher variability for the estimates compared to the other machine learning techniques. To address this issue, one may incorporate variance reduction techniques in the algorithm to decrease the variability for an estimate of interest such that it can be used for datasets with few observations.

2.6 References

[1] Global HIV & AIDS statistics - 2020 fact sheet, 2020.

[2] José M. Cuevas, Ron Geller, Raquel Garijo, José López-Aldeguer, and Rafael Sanjuán. Extremely High Mutation Rate of HIV-1 In Vivo. *PLoS Biology*, 2015.

[3] R Kantor, R W Shafer, S Follansbee, J Taylor, D Shilane, L Hurley, D P Nguyen, D Katzenstein, and W J Fessel. Evolution of resistance to drugs in HIV-1-infected patients failing antiretroviral therapy. *AIDS*, 18(11):1503–1511, 2004.

[4] R M Novak, L Chen, R D MacArthur, J D Baxter, K Huppler Hullsiek, G Peng, Y Xiang, C Henely, B Schmetter, J Uy, M van den Berg-Wolf, M Kozal, and Terry Beirn Community Programs for Clinical Research on AIDS 058 Study Team. Prevalence of antiretroviral drug resistance mutations in chronically HIV-infected, treatment-naive patients: implications for routine resistance screening before initiation of antiretroviral therapy. *Clin Infect Dis*, 40(3):468–474, 2005.

[5] Joseph K. Agor and Osman Y. Özaltın. Models for predicting the evolution of influenza to inform vaccine strain selection, 3 2018.

[6] Orlando DeLeon, Hagit Hodis, Yunxia O'Malley, Jacklyn Johnson, Hamid Salimi, Yinjie Zhai, Elizabeth Winter, Claire Remec, Noah Eichelberger, Brandon Van Cleave, Ramya Puliadi, Robert D. Harrington, Jack T. Stapleton, and Hillel Haim. Accurate predictions of population-level changes in sequence and structural properties of HIV-1 Env using a volatility-controlled diffusion model. *PLoS Biology*, 2017.

[7] Matthijs Meijers, Kanika Vanshylla, Henning Gruell, Florian Klein, and Michael Lässig. Predicting in vivo escape dynamics of HIV-1 from a broadly neutralizing antibody, 8 2020.

[8] Monique Nijhuis, Charles A.B. Boucher, Pauline Schipper, Thomas Leitner, Rob Schuurman, and Jan Albert. Stochastic processes strongly influence HIV-1 evolution during suboptimal protease-inhibitor therapy. *Proceedings of the National Academy of Sciences of the United States of America*, 95(24):14441–14446, 11 1998.

[9] Fabio Zanini, Johanna Brodin, Lina Thebo, Christa Lanz, Göran Bratt, Jan Albert, and Richard A. Neher. Population genomics of intrapatient HIV-1 evolution. *eLife*, 4(DECEMBER2015), 12 2015.

[10] John Archer and David L. Robertson. Understanding the diversification of HIV-1 groups M and O. *AIDS*, 2007.

[11] Brian Gaschen, Jesse Taylor, Karina Yusim, Brian Foley, Feng Gao, Dorothy Lang, Vladimir Novitsky, Barton Haynes, Beatrice H. Hahn, Tanmoy Bhattacharya, and Bette Korber. Diversity considerations in HIV-1 vaccine selection, 2002.

[12] B Chen. Molecular Mechanism of HIV-1 Entry. *Trends Microbiol*, 27(10):878–891, 2019.

[13] Steven W. de Taeye, John P. Moore, and Rogier W. Sanders. HIV-1 Envelope Trimer Design and Immunization Strategies To Induce Broadly Neutralizing Antibodies, 2016.

[14] Phillippe Lemey, Andrew Rambaut, and Oliver G. Pybus. HIV evolutionary dynamics within and among hosts, 2006.

[15] Raj Shankarappa, Joseph B. Margolick, Stephen J. Gange, Allen G. Rodrigo, David Upchurch, Homayoon Farzadegan, Phalguni Gupta, Charles R. Rinaldo, Gerald H. Learn, Xi He, Xiao-Li Huang, and James I. Mullins. Consistent Viral Evolutionary Changes Associated with the Progression of Human Immunodeficiency Virus Type 1 Infection. *Journal of Virology*, 1999.

[16] Michael Sabourin, Amar Mitiche, Danny Thomas, and George Nagy. Classifier combination for hand-printed digit recognition. In *Proceedings of 2nd International Conference on Document Analysis and Recognition (ICDAR'93)*, pages 163–166. IEEE, 1993.

[17] Kevin Woods, W. Philip Kegelmeyer, and Kevin Bowyer. Combination of multiple classifiers using local accuracy estimates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1997.

[18] Giorgio Giacinto and Fabio Roli. Dynamic classifier selection. In *International Workshop on Multiple Classifier Systems*, pages 177–189. Springer, 2000.

[19] Albert H.R. Ko, Robert Sabourin, and Alceu Souza Britto. From dynamic classifier selection to dynamic ensemble selection. *Pattern Recognition*, 2008.

[20] Azizi Nabiha and Farah Nadir. New dynamic ensemble of classifiers selection approach based on confusion matrix for arabic handwritten recognition. In *2012 International Conference on Multimedia Computing and Systems*, pages 308–313. IEEE, 2012.

[21] Piotr Porwik, Rafal Doroz, and Krzysztof Wrobel. An ensemble learning approach to lip-based biometric verification, with a dynamic selection of classifiers. *Expert Systems with Applications*, 2019.

[22] Yufei Xia, Junhao Zhao, Lingyun He, Yinguo Li, and Mengyi Niu. A novel tree-based dynamic heterogeneous ensemble method for credit scoring. *Expert Systems with Applications*, 2020.

[23] Robert Burduk and Paulina Heda. Homogeneous ensemble selection-experimental studies. In *International Multi-Conference on Advanced Computer Systems*, pages 58–67. Springer, 2016.

[24] Manh Truong Dang, Anh Vu Luong, Tuyet-Trinh Vu, Quoc Viet Hung Nguyen, Tien Thanh Nguyen, and Bela Stantic. An ensemble system with random projection and dynamic ensemble selection. In *Asian Conference on Intelligent Information and Database Systems*, pages 576–586. Springer, 2018.

[25] Anil Narassiguin, Haytham Elghazel, and Alex Aussem. Similarity Tree Pruning: a novel dynamic ensemble selection approach. In *2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW)*, pages 1243–1250. IEEE, 2016.

[26] Giorgio Giacinto and Fabio Roli. Dynamic classifier selection based on multiple classifier behaviour. *Pattern Recognition*, 2001.

[27] Yue Zhao, Zain Nasrullah, Maciej K. Hryniewicki, and Zheng Li. LSCP: Locally selective combination in parallel outlier ensembles. In *SIAM International Conference on Data Mining, SDM 2019*, 2019.

[28] Chris Ballard and Wenjia Wang. Dynamic ensemble selection methods for heterogeneous data mining. In *2016 12th World Congress on Intelligent Control and Automation (WCICA)*, pages 1021–1026. IEEE, 2016.

[29] X Fan, S Hu, and J He. Target Recognition Algorithm for Maritime Surveillance Radars Based on Clustering and Random Reference Classifier. *Moshi Shibie yu Rengong Zhineng/Pattern Recognition and Artificial Intelligence*, 30:983–994, 2017.

[30] Jufu Feng, Liwei Wang, Masashi Sugiyama, Cheng Yang, Zhi-Hua Zhou, and Chicheng Zhang. Boosting and margin theory. *Frontiers of Electrical and Electronic Engineering*, 7(1):127–133, 2012.

[31] Ashfaqur Rahman and Brijesh Verma. Novel layered clustering-based approach for generating ensemble of classifiers. *IEEE Transactions on Neural Networks*, 2011.

[32] Marek Kurzynski, Tomasz Woloszynski, and Rafal Lysiak. On two measures of classifier competence for dynamic ensemble selection-experimental comparative analysis. In *2010 10th International Symposium on Communications and Information Technologies*, pages 1108–1113. IEEE, 2010.

[33] Rafal Lysiak, Marek Kurzynski, and Tomasz Woloszynski. Probabilistic approach to the dynamic ensemble selection using measures of competence and diversity of base classifiers. In *International Conference on Hybrid Artificial Intelligence Systems*, pages 229–236. Springer, 2011.

[34] Aiman Qadeer and Usman Qamar. A Dynamic Ensemble Selection Framework Using Dynamic Weighting Approach. In Yaxin Bi, Rahul Bhatia, and Supriya Kapoor, editors, *Intelligent Systems and Applications*, pages 330–339, Cham, 2020. Springer International Publishing.

[35] Shiliang Sun. Local within-class accuracies for weighting individual outputs in multiple classifier systems. *Pattern Recognition Letters*, 31(2):119–124, 2010.

[36] Ashfaqur Rahman and Brijesh Verma. Effect of ensemble classifier composition on offline cursive character recognition. *Information processing & management*, 49(4):852–864, 2013.

[37] Dario Di Nucci, Fabio Palomba, Rocco Oliveto, and Andrea De Lucia. Dynamic Selection of Classifiers in Bug Prediction: An Adaptive Method. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 2017.

[38] Marina Caskey, Florian Klein, Julio C.C. Lorenzi, Michael S. Seaman, Anthony P. West, Noreen Buckley, Gisela Kremer, Lilian Nogueira, Malte Braunschweig, Johannes F. Scheid, Joshua A. Horwitz, Irina Shimeliovich, Sivan Ben-Avraham, Maggi Witmer-Pack, Martin Platten, Clara Lehmann, Leah A. Burke, Thomas Hawthorne, Robert J. Gorelick, Bruce D. Walker, Tibor Keler, Roy M. Gulick, Gerd Fätkenheuer, Sarah J. Schlesinger, and Michel C. Nussenzweig. Viraemia suppressed in HIV-1-infected humans by broadly neutralizing antibody 3BNC117. *Nature*, 2015.

[39] Max Lataillade, Jacob P. Lalezari, Michael Kozal, Judith A. Aberg, Gilles Pialoux, Pedro Cahn, Melanie Thompson, Jean Michel Molina, Santiago Moreno, Beatriz Grinsztejn, Ricardo S. Diaz, Antonella Castagna, Princy N. Kumar, Gulam H. Latiff, Edwin De Jesus, Marcia Wang, Shiven Chabria, Margaret Gartland, Amy Pierce, Peter Ackerman, and Cyril Llamoso. Safety and efficacy of the HIV-1 attachment inhibitor prodrug fostemsavir in heavily treatment-experienced individuals: week 96 results of the phase 3 BRIGHTE study. *The Lancet HIV*, 2020.

[40] Annemarie Laumaea, Amos B.Smith Iii, Joseph Sodroski, and Andrés Finzi. Opening the HIV envelope: Potential of CD4 mimics as multifunctional HIV entry inhibitors, 2020.

[41] J. E. Ledgerwood, E. E. Coates, G. Yamshchikov, J. G. Saunders, L. Holman, M. E. Enama, A. Dezure, R. M. Lynch, I. Gordon, S. Plummer, C. S. Hendel, A. Pegu, M. Conan-Cibotti, S. Sitar, R. T. Bailer, S. Narpala, A. McDermott, M. Louder, S. O'Dell, S. Mohan, J. P. Pandey, R. M. Schwartz, Z. Hu, R. A. Koup, E. Capparelli, J. R. Mascola, B. S. Graham, Floreliz Mendoza, Laura Novik, Kathy Zephir, William Whalen, Brenda Larkin, Olga Vasilenko, Nina Berkowitz, Brandon Wilson, Iris Pittman, Gretchen Schieber, Hope Decederfelt, Judith Starling, John Gilly, Srinivas Rao, Florence Kaltovich, Phyllis Renehan, Meghan Kunchai, Sarah Romano, Katie Menard, Ly Diep, Chuka Anude, and Mary Allen. Safety, pharmacokinetics and neutralization of the broadly neutralizing HIV-1 human monoclonal antibody VRC01 in healthy adults. *Clinical and Experimental Immunology*, 2015.

[42] Marie Pancera, Yen Ting Lai, Tatsiana Bylund, Aliaksandr Druz, Sandeep Narpala, Sijy O'Dell, Arne Schön, Robert T. Bailer, Gwo Yu Chuang, Hui Geng, Mark K. Louder, Reda Rawi, Djade I. Soumana, Andrés Finzi, Alon Herschhorn, Navid Madani, Joseph Sodroski, Ernesto Freire, David R. Langley, John R. Mascola, Adrian B. McDermott, and Peter D. Kwong. Crystal structures of trimeric HIV envelope with entry inhibitors BMS-378806 and BMS-626529. *Nature Chemical Biology*, 2017.

[43] Jesus F. Salazar-Gonzalez, Elizabeth Bailes, Kimmy T. Pham, Maria G. Salazar, M. Brad Guffey, Brandon F. Keele, Cynthia A. Derdeyn, Paul Farmer, Eric Hunter, Susan Allen, Olivier Manigart, Joseph Mulenga, Jeffrey A. Anderson, Ronald Swanstrom, Barton F. Haynes, Gayathri S. Athreya, Bette T. M. Korber, Paul M. Sharp, George M. Shaw, and Beatrice H. Hahn. Deciphering Human Immunodeficiency Virus Type 1 Transmission and Early Envelope Diversification by Single-Genome Amplification and Sequencing. *Journal of Virology*, 2008.

[44] Changze Han, Jacklyn Johnson, Rentian Dong, Raghavendranath Kandula, Alexa Kort, Maria Wong, Tianbao Yang, Patrick J. Breheny, Grant D. Brown, and Hillel Haim. Key positions of HIV-1 ENV and signatures of vaccine efficacy show gradual reduction of population founder effects at the clade and regional levels. *mBio*, 2020.

[45] Brian Gaschen, Carla Kuiken, Bette Korber, and Brian Foley. Retrieval and on-the-fly alignment of sequence fragments from the HIV database. *Bioinformatics*, 2001.

[46] B.T. Korber, B.T. Foley, C.L. Kuiken, S.K. Pillai, and J.G. Sodroski. Numbering positions in HIV relative to HXB2CG. *Human retroviruses and AIDS*, 1998.

[47] Jeong Hyun Lee, Gabriel Ozorowski, and Andrew B. Ward. Cryo-EM structure of a native, fully glycosylated, cleaved HIV-1 envelope trimer. *Science*, 2016.

[48] Andrew J. Borst, Connor E. Weidle, Matthew D. Gray, Brandon Frenz, Joost Snijder, M. Gordon Joyce, Ivelin S. Georgiev, Guillaume B.E. Stewart-Jones, Peter D. Kwong, Andrew T. McGuire, Frank Dimaio, Leonidas Stamatatos, Marie Pancera, and David Veesler. Germline VRC01 antibody recognition of a modified clade C HIV-1 envelope trimer and a glycosylated HIV-1 gp120 core. *eLife*, 2018.

[49] Xiping Wei, Julie M. Decker, Shuyi Wang, Huxiong Hui, John C. Kappes, Xiaoyun Wu, Jesus F. Salazar-Gonzalez, Maria G. Salazar, J. Michael Kilby, Michael S. Saag, Natalia L. Komarova, Martin A. Nowak, Beatrice H. Hahn, Peter D. Kwong, and George M. Shaw. Antibody neutralization and escape by HIV-1. *Nature*, 2003.

[50] Jacklyn Johnson, Manuel G. Flores, John Rosa, Changze Han, Alicia M. Salvi, Kris A. DeMali, Jennifer R. Jagnow, Amy Sparks, and Hillel Haim. The High Content of Fructose in Human Semen Competitively Inhibits Broad and Potent Antivirals That Target High-Mannose Glycans. *Journal of Virology*, 2020.

[51] Manjari Lal, Manshun Lai, Shweta Ugaonkar, Asa Wesenberg, Larisa Kizima, Aixa Rodriguez, Keith Levendosky, Olga Mizenina, José Fernández-Romero, and Thomas Zydowsky. Development of a Vaginal Fast-Dissolving Insert Combining Griffithsin and Carrageenan for Potential Use Against Sexually Transmitted Infections. *Journal of Pharmaceutical Sciences*, 2018.

[52] Marina Caskey, Till Schoofs, Henning Gruell, Allison Settler, Theodora Karagounis, Edward F. Kreider, Ben Murrell, Nico Pfeifer, Lilian Nogueira, Thiago Y. Oliveira, Gerald H. Learn, Yehuda Z. Cohen, Clara Lehmann, Daniel Gillor, Irina Shimeliovich, Cecilia Unson-O'Brien, Daniela Weiland, Alexander Robles, Tim Kümmerle, Christoph Wyen, Rebeka Levin, Maggi Witmer-Pack, Kemal Eren, Caroline Ignacio, Szilard Kiss, Anthony P. West, Hugo Mouquet, Barry S. Zingman, Roy M. Gulick, Tibor Keler, Pamela J. Bjorkman, Michael S. Seaman, Beatrice H. Hahn, Gerd Fätkenheuer, Sarah J. Schlesinger, Michel C. Nussenzweig, and Florian Klein. Antibody 10-1074 suppresses viremia in HIV-1-infected individuals. *Nature Medicine*, 2017.

[53] Julian K.C. Ma, Jürgen Drossard, David Lewis, Friedrich Altmann, Julia Boyle, Paul Christou, Tom Cole, Philip Dale, Craig J. van Dolleweerd, Valerie Isitt, Dietmar Katinger, Martin Lobedan, Hubert Mertens, Mathew J. Paul, Thomas Rademacher, Markus Sack, Penelope A.C. Hundleby, Gabriela Stiegler, Eva Stoger, Richard M. Twyman, Brigitta Vcelar, and Rainer Fischer. Regulatory approval and a first-in-human phase I clinical trial of a monoclonal antibody produced in transgenic tobacco plants. *Plant Biotechnology Journal*, 2015.

[54] Christopher O. Barnes, Harry B. Gristick, Natalia T. Freund, Amelia Escolano, Artem Y. Lyubimov, Harald Hartweger, Anthony P. West, Aina E. Cohen, Michel C. Nussenzweig, and Pamela J. Bjorkman. Structural characterization of a highly-potent V3-glycan broadly neutralizing antibody bound to natively-glycosylated HIV-1 envelope. *Nature Communications*, 2018.

[55] Christine A. Bricault, Karina Yusim, Michael S. Seaman, Hyejin Yoon, James Theiler, Elena E. Giorgi, Kshitij Wagh, Maxwell Theiler, Peter Hraber, Jennifer P. Macke, Edward F. Kreider, Gerald H. Learn, Beatrice H. Hahn, Johannes F. Scheid, James M. Kovacs, Jennifer L. Shields, Christy L. Lavine, Fadi Ghantous, Michael Rist, Madeleine G. Bayne, George H. Neubauer, Katherine McMahan, Hanqin Peng, Coraline Chéneau, Jennifer J. Jones, Jie Zeng, Christina Ochsenbauer, Joseph P. Nkolola, Kathryn E. Stephenson, Bing Chen, S. Gnanakaran, Mattia Bonsignori, La Tonya D. Williams, Barton F. Haynes, Nicole Doria-Rose, John R. Mascola, David C. Montefiori, Dan H. Barouch, and Bette Korber. HIV-1 Neutralizing Antibody Signatures and Application to Epitope-Targeted Vaccine Design. *Cell Host and Microbe*, 2019.

[56] Leopold Kong, Jeong Hyun Lee, Katie J. Doores, Charles D. Murin, Jean Philippe Julien, Ryan McBride, Yan Liu, Andre Marozsan, Albert Cupo, Per Johan Klasse, Simon Hoffenberg, Michael Caulfield, C. Richter King, Yuanzi Hua, Khoa M. Le, Reza Khayat, Marc C. Deller, Thomas Clayton, Henry Tien, Ten Feizi, Rogier W. Sanders, James C. Paulson, John P. Moore, Robyn L. Stanfield, Dennis R. Burton, Andrew B. Ward, and Ian A. Wilson. Supersite of immune vulnerability on the glycosylated face of HIV-1 envelope glycoprotein gp120. *Nature Structural and Molecular Biology*, 2013.

[57] C. D. Murin, J.-P. Julien, D. Sok, R. L. Stanfield, R. Khayat, A. Cupo, J. P. Moore, D. R. Burton, I. A. Wilson, and A. B. Ward. Structure of 2G12 Fab2 in Complex with Soluble and Fully Glycosylated HIV-1 Env by Negative-Stain Single-Particle Electron Microscopy. *Journal of Virology*, 2014.

[58] Laura M. Walker, Michael Huber, Katie J. Doores, Emilia Falkowska, Robert Pejchal, Jean Philippe Julien, Sheng Kai Wang, Alejandra Ramos, Po Ying Chan-Hui, Matthew Moyle, Jennifer L. Mitcham, Phillip W. Hammond, Ole A. Olsen, Pham Phung, Steven Fling, Chi Huey Wong, Sanjay Phogat, Terri Wrin, Melissa D. Simek, Wayne C. Koff, Ian A. Wilson, Dennis R. Burton, and Pascal Poignard. Broad neutralization coverage of HIV by multiple highly potent antibodies. *Nature*, 2011.

[59] Kathryn Fischer, Kimberly Nguyen, and Patricia J. LiWang. Griffithsin Retains Anti-HIV-1 Potency with Changes in gp120 Glycosylation and Complements Broadly Neutralizing Antibodies PGT121 and PGT126. *Antimicrobial Agents and Chemotherapy*, 2020.

[60] Matthew J. Gartner, Michael Roche, Melissa J. Churchill, Paul R. Gorry, and Jacqueline K. Flynn. Understanding the mechanisms driving the spread of subtype C HIV-1, 2020.

[61] Ronald Derking, Gabriel Ozorowski, Kwinten Sliepen, Anila Yasmeen, Albert Cupo, Jonathan L. Torres, Jean Philippe Julien, Jeong Hyun Lee, Thijs van Montfort, Steven W. de Taeye, Mark Connors, Dennis R. Burton, Ian A. Wilson, Per Johan Klasse, Andrew B. Ward, John P. Moore, and Rogier W. Sanders. Comprehensive Antigenic Map of a Cleaved Soluble HIV-1 Envelope Trimer. *PLoS Pathogens*, 2015.

[62] Shilei Ding, Melissa C. Grenier, William D. Tolbert, Dani Vézina, Rebekah Sherburn, Jonathan Richard, Jérémie Prévost, Jean-Philippe Chapleau, Gabrielle Gendron-Lepage, Halima Medjahed, Cameron Abrams, Joseph Sodroski, Marzena Pazgier, Amos B. Smith, and Andrés Finzi. A New Family of Small-Molecule CD4-Mimetic Compounds Contacts Highly Conserved Aspartic Acid 368 of HIV-1 gp120 and Mediates Antibody-Dependent Cellular Cytotoxicity. *Journal of Virology*, 2019.

[63] Yen Ting Lai, Tao Wang, Sijy O'Dell, Mark K. Louder, Arne Schön, Crystal S.F. Cheung, Gwo Yu Chuang, Aliaksandr Druz, Bob Lin, Krisha McKee, Dongjun Peng, Yongping Yang, Baoshan Zhang, Alon Herschhorn, Joseph Sodroski, Robert T. Bailer, Nicole A. Doria-Rose, John R. Mascola, David R. Langley, and Peter D. Kwong. Lattice engineering enables definition of molecular features allowing for potent small-molecule inhibition of HIV-1 entry. *Nature Communications*, 2019.

[64] Raymond H.Y. Louie, Kevin J. Kaczorowski, John P. Barton, Arup K. Chakraborty, and Matthew R. McKay. Fitness landscape of the human immunodeficiency virus envelope protein that is targeted by antibodies. *Proceedings of the National Academy of Sciences of the United States of America*, 2018.

[65] Paul R. Clapham and Áine McKnight. Cell surface receptors, virus entry and tropism of primate lentiviruses, 2002.

[66] Bradley D. Preston, Bernard J. Poiesz, and Lawrence A. Loeb. Fidelity of HIV-1 reverse transcriptase. *Science*, 1988.

[67] Everett Clinton Smith, Nicole R. Sexton, and Mark R. Denison. Thinking outside the triangle: Replication fidelity of the largest RNA viruses. *Annual Review of Virology*, 2014.

[68] Thiruvarangan Ramaraj, Thomas Angel, Edward A. Dratz, Algirdas J. Jesaitis, and Brendan Mumey. Antigen-antibody interface properties: Composition, residue interactions, and features of 53 non-redundant structures. *Biochimica et Biophysica Acta - Proteins and Proteomics*, 2012.

CHAPTER 3. A STACKING-BASED CLASSIFICATION METHOD TO PREDICT ICU ADMISSION IN HOSPITALIZED COVID-19 PATIENTS

Mohammad Fili[1], Parvin Mohammadiarvejeh[1], Guiping Hu[1,2]

[1] Department of Industrial and Manufacturing Systems Engineering, Iowa State University, Ames, IA, USA.

[2] Department of Sustainability, Golisano Institute for Sustainability, Rochester Institute of Technology, Rochester, New York, USA.

Modified from a manuscript under review in *Artificial Intelligence in Medicine*

3.1 Abstract

As of November 2021, more than 27 million people have been hospitalized, and about 7 million individuals have been admitted to intensive care units (ICU) due to Covid-19 in the United States. This leads to a long-stay increase for ICU beds that are near or at capacity. In these circumstances, predicting ICU needs for patients as early as possible is critical since hospitals can better manage the resources and ultimately save more lives. In this study, we propose a novel stacking-based classification method named Stacked Ensemble using Regional and Neighborhood Assessment (SERNA) that can predict the ICU need using the data for the first 2 hours of admission. The proposed method is a four-stage algorithm: in the first three stages, new sets of features are created to generate connections between a data point and the performance of base learners in the vicinity of that data point. In the last stage, the generated features are fed into a meta-learner so that it can strategically give higher weights to the strong learners while discarding or giving lower weights to the weaker ones. We implemented the proposed model based on the Covid-19 ICU admission data. The performance of the proposed model was compared with that of the base learners employed within the model. SERNA model improved accuracy by 5%,

precision by 3%, recall by 19%, F1 score by 9%, and AUC by 5%. Compared to the previous studies, SERNA demonstrated a 13% improvement in recall while using only the data for the first 2 hours after admission instead of 12 or 24 hours utilized in other studies, saving at least 10 hours of reaction time.

3.2 Introduction

Severe acute respiratory syndrome coronavirus 2 (Sars-CoV-2) emerged in December 2019 as a threat to public health, and in March 2020, the World Health Organization (WHO) declared it as a global pandemic [1]. Based on John Hopkins University Coronavirus Resource Center's estimations, about 48 million individuals were infected with Covid-19 [2], more than 27 million people have been hospitalized, and about 7 million individuals have been admitted to ICU as of the end of November 2021 in the United States [3]. Between May and August 2020, 1,754 out of 8,013 (about 22%) Covid-19 cases visited emergency departments [4]. In such critical periods, the need for ICU increases significantly, and hospitals are near or at capacity. Therefore, it is critical to identify patients in need of ICU using the patient's information early in the initial hours of reception. This is essential for hospitals to prioritize individuals based on the likelihood of ICU admission and better utilize the resources to save lives.

In this paper, we propose a staking-based classification method that can predict whether a patient needs ICU or not based on the patient's information collected within the first 2 hours of admission. Multiple studies have been conducted to predict ICU admission. They used different machine learning tools to predict the ICU need such as decision tree [5], Random Forest (RF) [6, 7, 8, 9], logistic regression [10, 11, 12, 13], boosting methods [7, 9, 12], support vector machines (SVM) [9], and artificial neural networks [7, 9]. In this study, a novel classification algorithm has been proposed that incorporates state-of-art methods as the base learners to generate new and informative features. It then feeds the generated features into a meta-learner to improve the classification results.

It should be noted that the data used in different studies are collected within different time intervals of admission. Cheng et al. worked on the prediction of ICU transfer on the first day of admission [6]. In another study, data from the first 24 hours after admission was used to make the prediction [12]. Jimenez-Solem improved this interval by using the first 12 hours to predict the ICU admission [8]. To improve the timeliness of prediction, in our study, we only used the first 2 hours after admission to predict the need for ICU. This significant time saving can help to manage the resources better and eventually save more lives.

In the literature, there are variations in terms of the objective and the set of features used in different studies. Research objectives besides the ICU admission have been considered, such as predicting mortality [10, 7, 8, 12, 13], the long-term stay [10], and ventilation use [7, 8, 12]. In addition, the set of features applied in each study was different. Cheng et al. used socio-demographic characteristics, time-series data of vital signs, laboratory results, electrocardiograms, and condition information about patients in nursing notes [6], while Paranjape et al. only incorporated socio-demographic characteristics, comorbidities, and clinical variables [11]. Eskes et al. used respiratory conditions in addition to the socio-demographics, comorbidities, vital signs, and laboratory tests results [10]. The dataset used for this study includes features such as demographic information, vital signs, blood biomarkers, and history of previous diseases.

The foundation of the proposed method is the stacked generalization, and the novelty lies in the design of the procedures that extract the new and informative set of features quantifying the performance of each base learner in the vicinity of a data point. In the following, we describe resemblances and differences between the proposed method and other relevant studies.

The incorporation of multiple base learners is one of the common aspects of stacking-based models. The pool of base learners can be homogenous [14] or heterogeneous [15]; large as in [16], where authors used SVM, K-nearest neighbors (KNN), random forest, gradient boosting decision tree, decision tree, logistic regression, and MLP, or small as in [17] where only random forest, logistic regression, and KNN were considered. In our study, we incorporate a set of heterogeneous learners, including extra tree classifier (ETC), Adaboost (Ada), random forest (RF), linear

discriminant analysis (LDA), and logistic regression (LogReg). We selected these models based on our preliminary analysis.

There have been studies using clustering techniques to divide the space into partitions and training several models for each partition [14, 18]. However, we incorporated multiple clustering techniques to find the performance of the base learners in different subspaces. The reason to use multiple clustering techniques in our model is to obtain different groupings based on different similarity definitions. The other key difference is that the base learners in our study are trained on the whole training set rather than being trained locally. It is because we are not looking for learners that are performing well locally, but understanding in which parts of the sample space a base learner is performing well so that the meta-learner can emphasize (i.e., give higher weights to its prediction) when the observation is located in that region.

One of the other mutual aspects of stacking-based models is applying a meta-learner to aggregate the generated results obtained from the base learners. The choice of meta-learner can differ based on the needs and application. The meta-learner can be MLP [14, 19], boosting methods [15, 16], or SVM [17]. In this study, we designed the algorithm with the choice of MLP as the meta-learner since it can learn intricate interactions between the generated features.

The contributions of this study can be summarized as follows:

- We developed a novel classification model capable of predicting the need for ICU for a patient using only the data for the first two hours of admission
- We introduced a procedure to automatically generate new sets of informative features for the meta-learner that quantifies the performance of each base learner within a region;
- We designed a procedure that enables the meta-learner to make a connection between the location of a data point and the performance of each base learner in that area through the regional and neighborhood assessment procedures.

This paper is organized as follows: In Section 3.3, first, we explain the dataset used in this study and the data preparations applied. We then describe the method in detail and show how

the model generates different sets of features for the meta-learner. The hyperparameter tuning procedure is explained at the end of this section. In Section 3.4, we show the numerical results and discuss how these findings show improvement compared to the base learners and also previous studies. Finally, Section 3.5 concludes with a summary of findings and the future directions.

3.3 Materials and Method

In this section, first, we describe the dataset and the different groups of features used in this study. Then, we explain the preprocessing steps applied to prepare the data for the training phase. Finally, we explore the method and different stages of training the model and also the prediction procedure.

3.3.1 Data

This study uses the public anonymized data from Hospital Sírio-Libanês, São Paulo and Brasilia, which is available on the Kaggle website [20]. The dataset has 235 features, including patient demographics, previous diseases, blood biomarkers, and vital signs. As discussed, this study aims to predict the need for ICU for hospitalized COVID-19 patients. Hence, the target variable (i.e., ICU admission) is binary. Regarding the patient demographic characteristics, we have the age and gender information for each patient. In this dataset, there are two features corresponding to age: a binary feature indicating whether a patient is over 65 or not and another variable corresponding to the patient's age percentile. Figure 3.1. shows the bar plot of individuals admitted to ICU based on their age percentile. As we can see, there was an increasing trend in ICU admission with respect to the age percentile.

There are nine attributes that correspond to different diseases in this dataset, two of which are hypertension and immunocompromised diseases. In addition, there are 36 blood biomarkers in this dataset, such as Calcium, Potassium, Glucose, Sodium, Hemoglobin, Lactate, Hematocrit, and free fatty acids (FFA). Figure 3.2 visualizes the boxplot for those admitted and not admitted

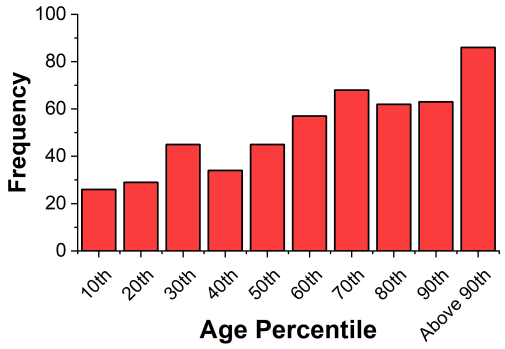

Figure 3.1 Bar plot of individuals admitted to ICU based on age percentile

to ICU for eight well-known blood biomarkers. A two-sample Kolmogorov-Smirnov test is applied for each to examine whether the distribution of two groups differs significantly or not.

Another group of critical attributes in our data is the vital signs. We have six variables corresponding to the vital signs: systolic and diastolic blood pressure, heart rate, respiratory rate, body temperature, and oxygen saturation. The mean, median, maximum, minimum, range, and relative range for each variable are available in the dataset. It is worth mentioning that all variables in the dataset are pre-standardized.

Figure 3.3 shows violin plots for the maximum values of 6 vital signs. We performed a two-sample Kolmogorov-Smirnov test to examine the difference in the distribution of those admitted and not admitted to ICU.

The blood biomarkers and vital signs are recorded at the beginning of the patients' admission and for some intervals afterward. In this regard, several time windows are defined, which can be one of the [0, 2], [2, 4], [4, 6], [6, 12], or 12+. Each interval refers to the number of hours after admission. We only used the data for the interval between 0 to 2 hours after admission in our study.

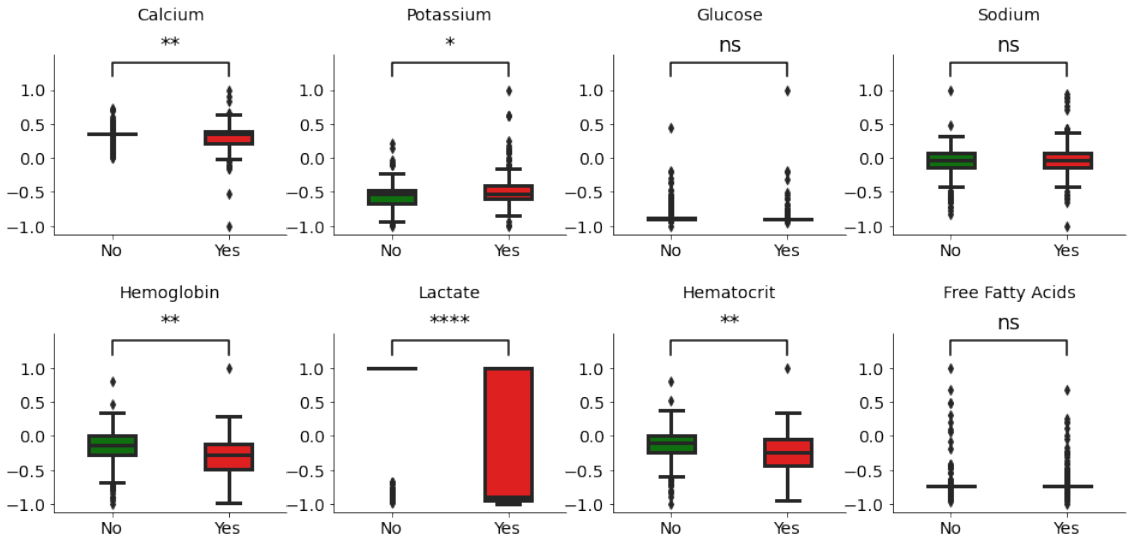

Figure 3.2 Comparison of the mean observed values of the blood biomarkers for the patients admitted or not admitted to ICU, labeled as Yes or No, respectively. (ns, non-significant; *,$p \leq 0.05$; **,$p \leq 0.01$; ***,$p \leq 0.001$; ****,$p \leq 0.0001$).

3.3.2 Data Preprocessing

To prepare the data for the modeling and analyses, we applied multiple data preprocessing steps, which are detailed in this section. One of the variables in this study is the age percentile which is a categorical variable. Therefore, we converted this variable to binary attributes using dummy variables. According to the dataset description, some variables may not be recorded when the patient's condition is stable. Therefore, a forward-fill approach was used to impute the missing values for blood biomarkers and vital signs.

As mentioned in the previous part, patients' information was recorded in specific time intervals. The ICU admission attribute shows if the patient has been transferred to ICU in that window. We defined a new variable to specify whether the patient has been transferred to the ICU or not regardless of the transmission time. Then, since our goal was to predict the ICU need, we removed any record with the initial ICU admission variable showing the patient is in the ICU. We kept only the records within the interval [0, 2] since our objective is to identify patients who need ICU, in the shortest time possible.

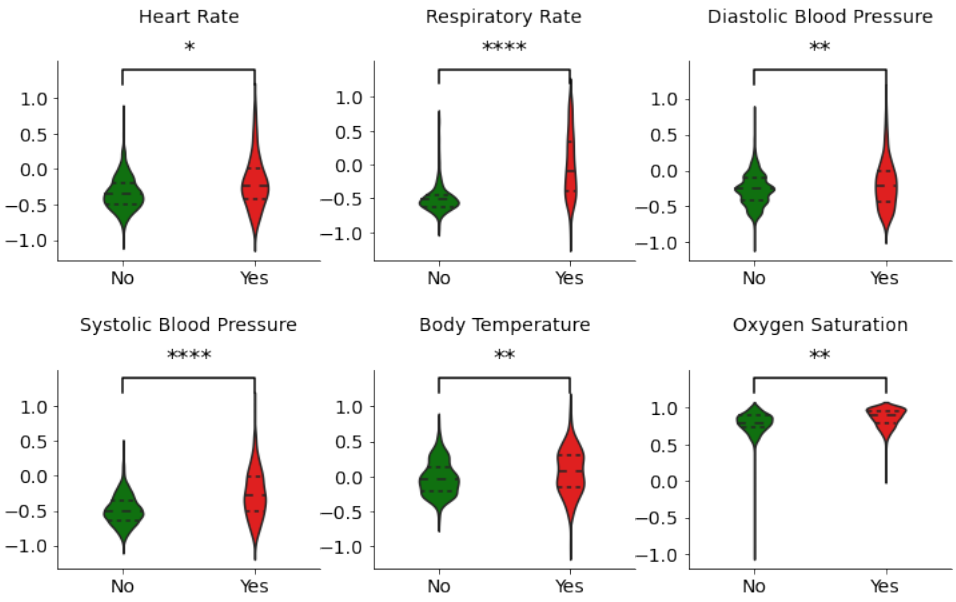

Figure 3.3 Comparison of the maximum observed value of the vital signs for the patients admitted or not admitted to ICU, labeled as Yes or No, respectively. (ns, non-significant; *,$p \leq 0.05$; **,$p \leq 0.01$; ***,$p \leq 0.001$; ****,$p \leq 0.0001$).

3.3.3 SERNA Algorithm

In this section, we explain the details of the SERNA algorithm. SERNA's foundation is based on the stacked generalization technique. First, the model utilizes a set of learners for initial prediction and then feeds their results to a new learner called meta-learner for final prediction. In the SERNA algorithm, not only we use the prediction results obtained from the base learners, but also, we create a procedure by which the performance of base learners in different parts of the sample space is evaluated. The output of this procedure is then fed into the meta-learner along with the base learners' prediction results.

Training the SERNA algorithm requires four stages: 1) Training base learners, 2) Neighborhood assessment, 3) Regional assessment, 4) Meta-learner training. Figure 3.4 is a flowchart that illustrates the process of training the SERNA model. First, we use PCA to make the variables uncorrelated and then train the base learners with the new features. We apply Bayesian optimization to tune the hyperparameters of each base learner. In the next step, for

each data point in the training set, we find a neighborhood using KNN and evaluate each base learner within that neighborhood. The other set of features is generated using regional assessment. To do so, first, we divide the sample space into subspaces or clusters of similar data points using different clustering methods. Then, within the region of each clustering method, we evaluate the base learners. In the last step, the predicted probabilities, neighborhood evaluations, and regional assessments are aggregated using an MLP model, and final predictions are made.

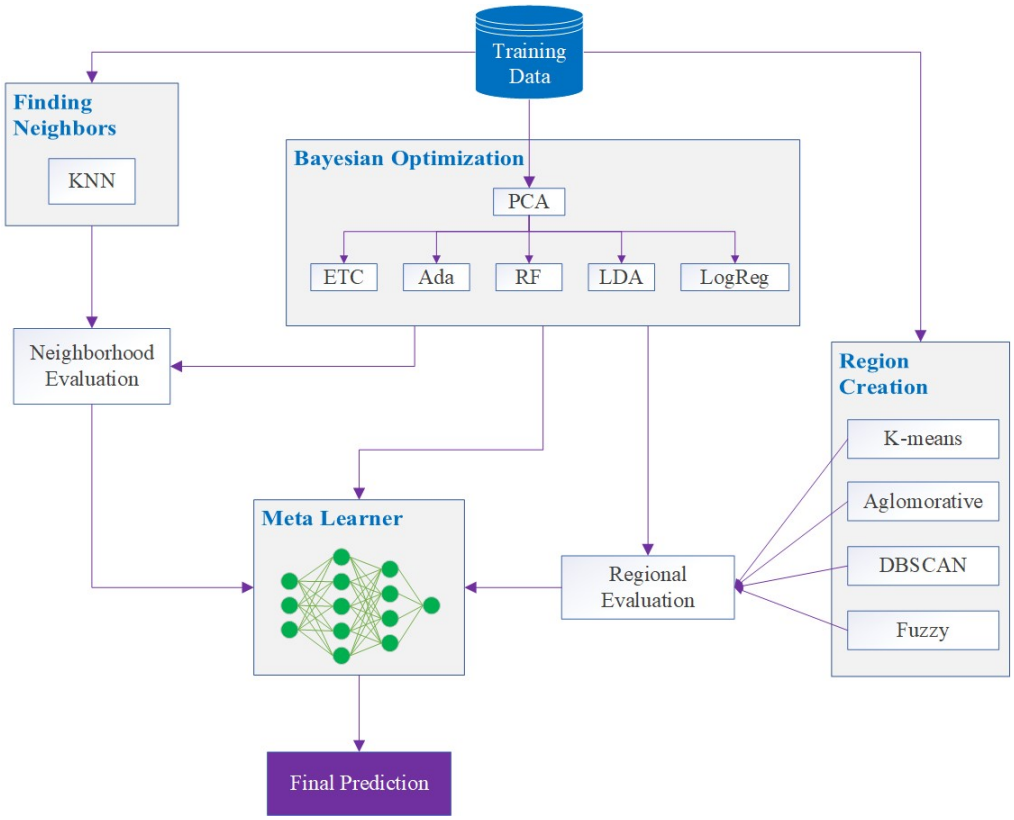

Figure 3.4 Flowchart of the training procedure for the SERNA model

Figure 3.5 shows the prediction flowchart, which is slightly different from the training process. We create a performance database called training information (TI) during the training phase. The TI includes information regarding the regions created by different clustering methods, the performance of base learners within each region, and the predicted probabilities of the training observations obtained from each base learner. For a new data point in the test set, first, we use

the trained base learners to predict the probability of ICU admission. Then, we find the neighbors from the training set and evaluate the learners within that neighborhood according to the metric(s) of interest. In the next step, we find the most similar region or cluster of observations to that data point and retrieve the performance of each base learner in that region from the TI database. This set of procedures provides the newly created features to the trained meta-learner for prediction. In the following, we discuss each step of the SERNA algorithm in more detail.

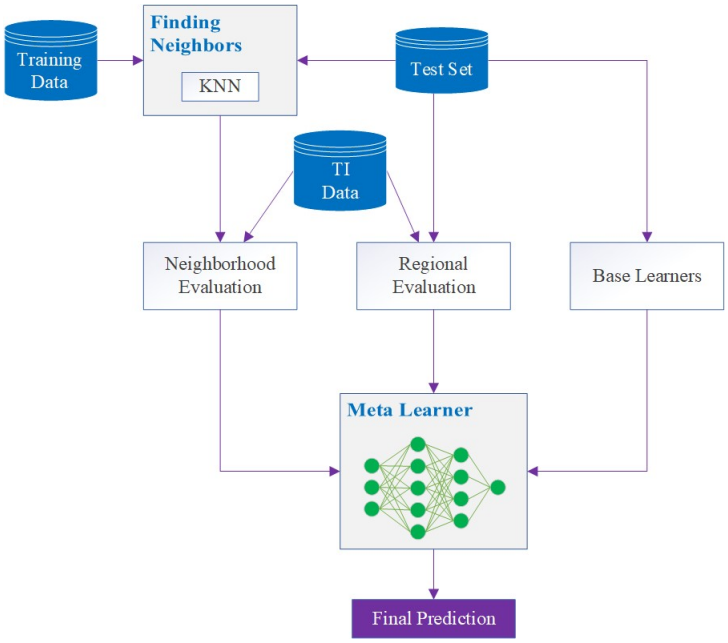

Figure 3.5 Flowchart of the predictive procedure for the SERNA model

Stage I: Base Learners Training

Similar to any stacked ensemble classification method, a group of base learners is incorporated at this step to produce the initial set of predictions. In this study, we used five heterogeneous base learners: extra tree classifier (ETC), Adaboost (Ada), random forest (RF), linear discriminant analysis (LDA), and logistic regression (LogReg). Before training the models, we applied principal component analysis (PCA), which helps us remove the correlation between the variables.

Instead of using the crisp 0 and 1 class label predictions, we utilized the probability of ICU admission since it gives more flexibility to the meta-learner for final prediction later on. We then

use these probabilities as the input to the meta-learner and also for the regional and neighborhood assessments. In general, if we have a training set X^{train} with N observations, and a pool of base learners $\mathscr{B}$ with $|\mathscr{B}|$ base learners in it, then the output of this stage in the training phase will be the matrix $\Pi_{N\times|\mathscr{B}|}$ with elements π_{ij} representing the predicted probability for observation x_i using learner j $(i = 1, 2, \ldots, N \quad and \quad j \in \mathscr{B})$.

To tune the hyperparameters of the base learners, we used Bayesian optimization, which is explained in more detail in the hyperparameters section.

Stage II: Neighborhood Assessment

In this part of the algorithm, we compute the base learners' performances in the neighborhood of an observation. This way, we provide new information regarding each base learner to the meta-learner so that it can adjust its emphasis on each learner accordingly. In ensemble learning using majority vote, we treat all learners equally, while here, the meta-learner tries to distinguish between the base learners according to their performances.

To define the neighborhood, we need to know how similar each observation is to the data point of interest. A common similarity metric is Euclidean distance. The smaller the distance, the more similar the two observations are. In this study, we used KNN to identify the neighborhood of an instance. The number of observations from the training set that we want to include as the neighbors, denoted as $\aleph$, is a hyperparameter that can be tuned later.

After specifying the neighbors for a data point, we evaluate the performance of each learner in the neighborhood defined. To distinguish between the metric(s) used in neighborhood assessment and the final metrics used for evaluating the overall performance of the model, we denote the former as $\mathscr{M}_n$ representing the set of neighborhood assessment metric(s). Here, $|\mathscr{M}_n|$ shows the number of metrics used for neighborhood assessment. In this study, we only used precision as the metric of interest for neighborhood assessment.

Newly created features from this procedure are denoted as p_{ijk} which represents the performance of learner $j \in \mathscr{B}$, in the neighborhood of the data point x_i $(i = 1, 2, \ldots, N)$, according to the metric $k \in \mathscr{M}_n$.

Stage III: Regional Assessment

In this section of the algorithm, we divide the sample space into groups of similar observations. Then by evaluating each base learner in different regions, we can understand how well each performs in different subspaces. For a new observation, based on the partition to which the data point most likely belongs, we can decide which learner was performing better in that area based on the performances obtained during the training phase.

One can use clustering techniques to define the regions, which put similar observations into the same cluster. In that sense, data points within a group are supposed to be very similar, while those in different groups share the least similarity. The way regions are defined plays a critical role in the performance evaluation results.

$\mathscr{R}$ denotes the set of region-definers or the clustering methods used to create the partitions. In this study, we applied various clustering techniques to include many region definitions. We used k-means [20, 20], agglomerative clustering [21], Density-Based Spatial Clustering of Applications with Noise (DBSCAN) [22], and fuzzy C-means [23]. Each clustering method tries to create the regions such that similar data points are grouped into the same clusters; however, the definition of similarity differs from one method to another. That's why we utilized various techniques to extract as much information as possible.

For the k-means algorithm, we used the elbow method to find the best number of clusters where the decrease in inertia begins to slow.

Agglomerative clustering is a bottom-up clustering procedure. In the beginning, this method considers all data points as individual clusters; then, in sequential steps, the model finds the two closest clusters to merge till all observations are in one cluster [21]. In agglomerative clustering, the linkage criterion specifies how the dissimilarity between clusters is measured, and the model merges the pairs of clusters accordingly. Usually, there are four types of linkages: ward, complete, single, and average. The ward strategy minimizes the variance between clusters. The complete

strategy minimizes the maximum distances between the observations within two clusters, while the single linkage minimizes the minimum of the distances between those observations. Finally, the average linkage method minimizes the average distances between the observations of two clusters. The Euclidean distance is often used as the metric to calculate the linkage [21, 24]. The number of clusters can be determined using a dendrogram that shows the merging process step by step [25].

DBSCAN is capable of clustering data with arbitrary shapes in the presence of noise and outlier [26]. This method has two main parameters: the minimum number of samples and epsilon. The minimum number of samples is the number of samples in a neighborhood of a data point, and epsilon is the maximum distance between two samples for one to be considered as in the neighborhood of the other. The matrix of k-nearest neighbor distances is calculated to find the optimum number of clusters. This matrix includes the average distance of each point to the k-nearest neighbors. Then, the k distances should be plotted in ascending order. The knee in the plot can be used as the optimum value for the epsilon.

Fuzzy C-means is a clustering technique that assigns a membership degree, between 0 and 1, to each data point in a cluster. A higher membership degree represents a better match between the point and the cluster. In this regard, the Fuzzy Partition Coefficient (FPC) is defined as a metric to evaluate how well the model describes the data. The FPC is between 0 and 1, and higher values are preferable. So, to find the best number of clusters using the fuzzy C-means method, we plotted the FPC against the number of centers. Table 3.1 summarizes the number of clusters used for each clustering technique.

Table 3.1 Number of clusters selected for each clustering method used in the SERNA model

Method	# of Clusters
k-means	3
Aglomorative Clustering	2
DBSCAN	2
Fuzzy C-means	2

Milligan et al. evaluated different indices regarding the agreement between two clustering results [27], and the adjusted Rand index (ARI) [28] is a metric that can be used in this regard. If we have two clustering methods U with partitions $u_1, u_2, \ldots, u_r$, and V with partitions $v_1, v_2, \ldots, v_s$, then, the ARI can be defined as Equation 3.1:

$$ARI = \frac{\sum_{i,j}\binom{n_ij}{2} - \sum_i \binom{n_{i.}}{2}\sum_j \binom{n_{.j}}{2}/\binom{n}{2}}{\frac{1}{2}[\sum_i \binom{n_{i.}}{2} + \sum_j \binom{n_{.j}}{2}] - \sum_i \binom{n_{i.}}{2}\sum_j \binom{n_{.j}}{2}/\binom{n}{2}} \tag{3.1}$$

where, n is the total number of data points, n_ij is the number of observations in partition u_i (according to method U) while they are assigned to the partition v_j by method V, $n_(i.)$ is the total number of observations in partition u_i, and $n_(.j)$ is the number of observations in $v_j (i = 1, 2, \ldots, r \quad and \quad j = 1, 2, \ldots, s)$.

This value is equal to 1 only if a partition is identical to the intrinsic structure and close to 0 if it is a random partition. Here, we don't have access to the true clustering structure. Therefore, in our case, ARI measures the consensus between the partitions created from different clustering methods. The result of this analysis is shown in the results section.

After defining the regions, we evaluate the performance of base learners in each region. $\mathscr{M}_r$ is the set of metric(s) used for regional assessment, with $|\mathscr{M}_r|$ as the size of this set. In this study, we only used precision.

The output of this procedure is the newly created features denoted as R^c_{ijkl} representing the regional performance of base learner $j \in \mathscr{B}$, in region l $(l = 1, 2, \ldots, |c|, \forall c \in \mathscr{R})$, using clustering method $c \in \mathscr{R}$, for a data point x_i, and according to the metric $k \in \mathscr{M}_r$. Here, $|c|$ is the size (number of clusters or regions) for each clustering method $c \in \mathscr{R}$.

Stage IV: Meta-learner Training

In this step, we feed all generated features into the meta-learner for final prediction. For the choice of meta-learner, we used the multi-layer perceptron (MLP) model. The meta-learner uses regional assessment, neighborhood assessment, and the predicted probabilities to find a relationship between the performance of the learners and the position of data points in the sample

space. This way, for classifying a new observation, the meta-learner can emphasize each base learner accordingly. This phase can be considered the aggregation part since all information extracted will be aggregated to predict the class labels.

An important point to consider in the SERNA algorithm is the number of features that we want to generate for the meta-leaner. Table 3.2 shows the output and number of features generated in each stage of the SERNA algorithm. For a general case with a pool of base learners $\mathscr{B}$, a set of region-definers $\mathscr{R}$, neighborhood assessment metric(s) $\mathscr{M}_n$, and regional assessment metric(s) $\mathscr{M}_r$, we can find the number of features for any data point x_i $(i = 1, 2, \ldots, N)$ as shown in Table 3.2.

Table 3.2 Output and number of generated features at the first three stages of the SERNA algorithm

Stage	Outputs	# of Generated Features
I	π_{ij}	$\lvert\mathscr{B}\rvert$
II	p_{ijk}	$\lvert\mathscr{B}\rvert \cdot \lvert\mathscr{M}_n\rvert$
III	R^c_{ijkl}	$\lvert\mathscr{B}\rvert \cdot \lvert\mathscr{M}_r\rvert \cdot \sum_{c\in\mathscr{R}} \lvert c\rvert$

The total number of features can be calculated based on Equation 3.2.

$$\lvert\mathscr{F}\rvert = \lvert\mathscr{B}\rvert \cdot \left(1 + \lvert\mathscr{M}_n\rvert + \lvert\mathscr{M}_r\rvert \cdot \sum_{c\in\mathscr{R}} \lvert c\rvert\right) \tag{3.2}$$

where, $\lvert\mathscr{F}\rvert$ is the total number of generated features, and $\lvert c\rvert$ is the size (number of clusters) for each clustering method $c \in \mathscr{R}$. Adding more base learners or increasing the number of clustering methods can rapidly increase the $\lvert\mathscr{F}\rvert$. Therefore, choosing the proper structure for the SERNA algorithm is critical.

3.3.4 Hyperparameters

SERNA has a group of hyperparameters that can be tuned for the best results. The first set of hyperparameters is related to the base learners. Depending on the choice of base learners, the group of hyperparameters differs. Table 3.3 summarizes the hyperparameters considered in our study based on the selection of base learners. In order to tune the hyperparameters of the base

learners, we used Bayesian optimization in the training phase. The Bayesian function uses the train set to find the best hyperparameters for each base learner. We maximized recall in the objective function to find the best set of hyperparameters. The number of iterations was set to 250 for each base learner, and we used 5-fold cross-validation to estimate learners' performance.

Table 3.3 Hyperparameters of the base learners used in SERNA model

Method	Hyperparameters	Range/Choices
ETC	Number of estimators	[5, 200]
	Max features to split	{square root, log}
	Min fraction of samples required to split	(0, 1)
Ada	Number of estimators	[5, 200]
	Learning rate	(0, 0.5)
RF	Max depth of tree	[1, 40]
	Max features to split	{square root, log}
	Min fraction of samples required to be at a leaf node	(0, 1)
	Number of estimators	[5, 200]
LDA	Solver	{Singular value decomposition, Least-squares solution, Eigenvalue decomposition}
	Shrinkage	(0, 1)
LogReg	Solver	{Newton-CG, Limited-memory Broyden–Fletcher–Goldfarb–Shanno (LBFGS), Coordinate descent algorithm, Stochastic Average Gradient descent (SAG), SAGA (extension of SAG)}
	Penalty	{L-1, L-2, elastic net, no penalty}
	C	(0, 10)

There is another set of hyperparameters that is related to the meta-learner. In Table 3.4, the hyperparameters that are used to tune the MLP model are shown.

Table 3.4 Hyperparameters of the meta-learner (MLP model)

Hyperparameter	Range/Choices
Number of hidden layers	{1, 2, 3}
Number of nodes in hidden layers	[10, 30]
Learning Rate	[0.0001, 0.15]
Optimizer	{Adam, SGD}

3.4 Results and Discussions

In this section, we discuss the results and analyses of the proposed model. First, we describe the clustering outputs and assess the dissimilarity between different methods used for clustering. Then we delve into the main results, including the set of comparisons between the SERNA model and the base learners. We compare our study against other related studies in the final section.

3.4.1 Region Creation

We start with an overview of the region creation results. As mentioned earlier, we used four clustering methods to create the regions so that the algorithm can evaluate the performance of the base learners within different regions and use this information to emphasize better learners for predicting the class label of unseen data. We used ARI to examine dissimilarity in our clustering outputs. Figure 3.6 shows the ARI values for different pairs of methods with darker colors indicating less similarity between the clustering outputs for the two methods.

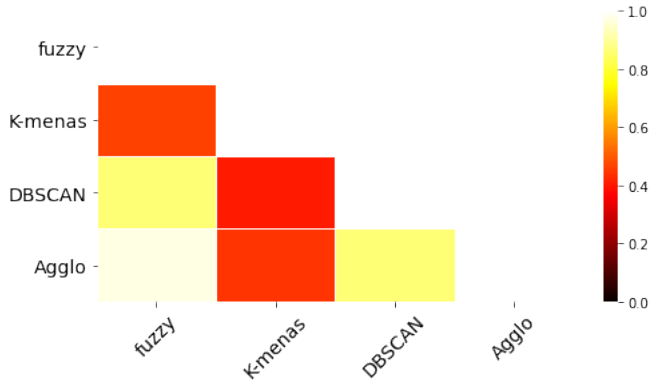

Figure 3.6 Calculated ARI values for different clustering methods

We can observe that k-means generates the most dissimilar outputs compared to other methods. The least dissimilarity is observed between the agglomerative clustering and fuzzy C-means methods with the ARI value greater than 0.9. However, each method alone generates dissimilar clustering results when it is compared with the rest. This difference in how each method creates the regions helps feed additional information regarding the location of a data point and base learner's performance in that region to the meta-learner for final prediction. Optimizing this set can improve the overall performance, which is suggested in the conclusion section as a future research direction.

To show the clusters in 2 dimensions, we used T-distributed Stochastic Neighbor Embedding (T-SNE) [29]. Figure 3.7 shows the T-SNE output for different clustering methods. Here, we can see how data points are grouped differently based on each method.

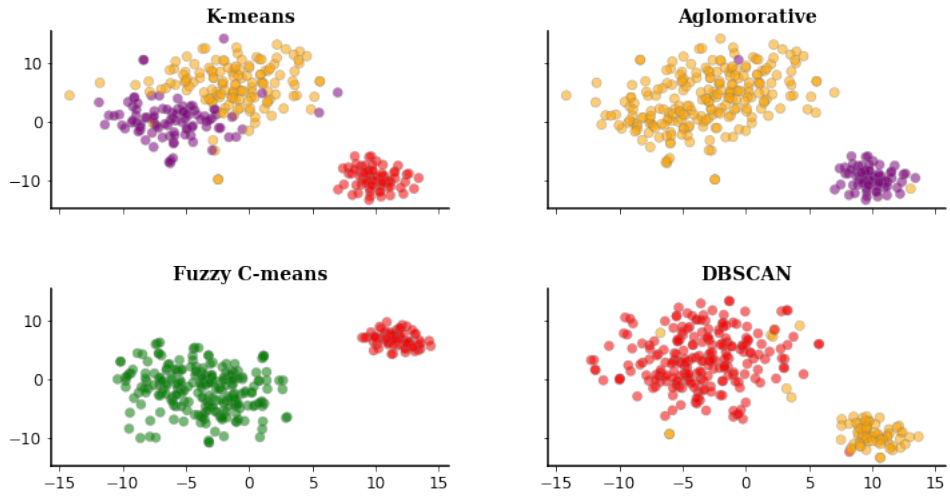

Figure 3.7 Visualization of the clusters defined by each method in 2 dimensions using T-SNE; k-means (A), agglomerative clustering (B), fuzzy c-means (C), and DBSCAN (D).

3.4.2 Model Performance

In this section, we compare the performance of the SERNA algorithm with that of the individual base learners. For our comparisons, we used the metrics shown in Equations 3.3, 3.4, 3.5, and 3.6.

$$Accuracy = \frac{tp + tn}{tp + tn + fp + fn} \tag{3.3}$$

$$Precision = \frac{tp}{tp + fp} \tag{3.4}$$

$$Recall = \frac{tp}{tp + fn} \tag{3.5}$$

$$F1\ score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{3.6}$$

$$\tag{3.7}$$

- tp: Number of patients admitted to ICU that are predicted as individuals in need of ICU.
- fp: Number of patients not admitted to ICU that are predicted as individuals in need of ICU.

- *tn*: Number of patients not admitted to ICU that are predicted as individuals not in need of ICU.
- *fn*: Number of patients admitted to ICU that are predicted as individuals not in need of ICU.

Table 3.5 shows the evaluation results. We can observe that SERNA improved accuracy by 5%, precision by 3%, recall by 19%, F1 score by 9%, and AUC by 5%. The recall, which is a crucial metric, drastically increased from 71% to 90%. It is clear that SERNA drastically improved the results compared to the individual base learners.

Table 3.5 Comparison between the performances of SERNA and the base learners

Base learner	Accuracy	Precision	Recall	F1	AUC
RF	0.62	0.52	0.67	0.59	0.64
ETC	0.62	0.52	0.71	0.60	0.69
Ada	0.57	0.45	0.48	0.46	0.55
LogReg	0.66	0.56	0.71	0.63	0.73
LDA	0.70	0.60	0.71	0.65	0.75
SERNA	0.75	0.63	0.90	0.74	0.80

3.4.3 Comparison with other Studies

We compared our proposed method with respect to the time interval used to make the predictions against other studies. Table 3.6 summarizes the results. Despite using a significantly smaller period to predict the ICU need, we achieved higher performance compared to other studies. In fact, not only we saved 10 to 22 hours, but also we improved the recall by 13%. Saving this amount of time can help emergency units plan way in advance and manage resources such as ICU beds more efficiently, and eventually saving more lives. On the other hand, the improvement in recall helps better identify the patients in need of ICU.

Table 3.6 Comparison between studies (metrics in %)

Study	Best Model	Period	Recall	AUC
Cheng et al, (2020) [6]	RF	24 hrs	72.8	79.9
Patel et al, (2021) [9]	RF	24 hrs	73.0	**80.0**
Wang et al, (2020) [12]	RF	24 hrs	77.0	77.0
Jimenez-Solem et al, (2021) [8]	RF	12 hrs	-	72.1
Our Study	SERNA	**2 hrs**	**90.0**	**80.0**

3.5 Conclusion

Since its emergence, Covid-19 has impacted the entire population of people around the world. As of the end of November 2021, more than 27 million individuals were hospitalized, and about 7 million people were admitted to ICU due to Covid-19 in the United States. With the increase in the number of hospitalization, the need for ICU beds also increases. This enforces the necessity to have a reliable approach for identifying patients in need of ICU soon enough so that hospitals and emergency units can better manage their resources to serve people in need of ICU and save more lives.

In this study, we proposed a stacking-based classification technique, named Stacked Ensemble using Regional and Neighborhood Assessment (SERNA), capable of identifying the ICU need for hospitalized COVID-19 patients using only the data from the first two hours of patients' admission. SERNA is a four-stage algorithm that automatically generates features for a meta-learner, including the predicted probabilities, the evaluation of each base learner within the neighborhood of a new data point and within the region to which the data point most likely belongs. In fact, SERNA tries to create a connection between the location of observation in the sample space with the base learners' performances in the vicinity of that data point. Then it uses the learned relationship to predict the class label of a new observation.

In the first stage, SERNA utilizes a pool of base learners to generate the probabilities for ICU admission. Then, in the second stage, for a new data point, it finds the most similar instances using KNN and evaluates the performance of each base learner within that neighborhood. In the third stage, the model uses one or more region-definers (clustering methods) to create partitions

and evaluate each base learner within that partition. At the final stage, it feeds the predicted probabilities, neighborhood assessments, and regional assessments to a meta-learner for final predictions.

Using SERNA for predicting the ICU, we observed superior performance compared to the base learners: extra tree classifier, Adaboost, logistic regression, random forest, linear discriminant analysis. SERNA improved accuracy, precision, recall, F1 score, and AUC by 5%, 3%, 19%, 9%, and 5% respectively. Recall which is an important metric, reached 90%, which shows the capability of the model to identify the patients in need of ICU. We also compared the proposed method with other studies and showed that SERNA outperforms methods explored in previous studies. As a result, we observed that recall improved by 13% while achieving the same AUC as the best method in previous studies.

The proposed method has some limitations, which suggest future research directions. First, the number of generated features with SERNA can grow very rapidly. Therefore, we need to wisely choose the set of region-definers, assessment metric(s), and the pool of base learners. Second, the choice of KNN for finding the neighborhood may slow down the training process, especially for large datasets. As a future research direction, one can examine other methods to find the neighborhood to increase the speed and improve the performance of the algorithm. Third, we utilized a group of clustering methods for the regional assessment; however, this process has not been optimized in our study. One can search for the smallest set of clustering methods giving the highest amount of information to the meta-learner and hence, optimize the model. At last, the idea of bootstrapping can be explored to obtain a better assessment of the learners in the training phase.

3.6 References

[1] Katarzyna Kotfis, Shawniqua Williams Roberson, Jo Ellen Wilson, Wojciech Dabrowski, Brenda T Pun, and E Wesley Ely. COVID-19: ICU delirium management during SARS-CoV-2 pandemic. 24(1), 2020.

[2] Home - Johns Hopkins Coronavirus Resource Center.

[3] Coronavirus Pandemic (COVID-19) - Statistics and Research - Our World in Data.

[4] Sonu Subudhi, Ashish Verma, Ankit B Patel, C Corey Hardin, Melin J Khandekar, Hang Lee, Dustin McEvoy, Triantafyllos Stylianopoulos, Lance L Munn, Sayon Dutta, and Rakesh K Jain. Comparing machine learning algorithms for predicting ICU admission and mortality in COVID-19. *npj Digital Medicine*, 4(1), 2021.

[5] Jose Luis Izquierdo, Julio Ancochea, and Joan B Soriano. Clinical Characteristics and Prognostic Factors for Intensive Care Unit Admission of Patients with COVID-19: Retrospective Study Using Machine Learning and Natural Language Processing. *Journal of Medical Internet Research*, 22(10), 2020.

[6] Fu Yuan Cheng, Himanshu Joshi, Pranai Tandon, Robert Freeman, David L. Reich, Madhu Mazumdar, Roopa Kohli-seth, Matthew Levin, Prem Timsina, and Arash Kia. Using machine learning to predict ICU transfer in hospitalized COVID-19 patients. *Journal of Clinical Medicine*, 2020.

[7] Fernando Timoteo Fernandes, Tiago Almeida de Oliveira, Cristiane Esteves Teixeira, Andre Filipe de Moraes Batista, Gabriel Dalla Costa, and Alexandre Dias Porto Chiavegatto Filho. A multipurpose machine learning approach to predict COVID-19 negative prognosis in São Paulo, Brazil. *Scientific Reports*, 11(1), 2021.

[8] Espen Jimenez-Solem, Tonny S Petersen, Casper Hansen, Christian Hansen, Christina Lioma, Christian Igel, Wouter Boomsma, Oswin Krause, Stephan Lorenzen, Raghavendra Selvan, Janne Petersen, Martin Erik Nyeland, Mikkel Zöllner Ankarfeldt, Gert Mehl Virenfeldt, Matilde Winther-Jensen, Allan Linneberg, Mostafa Mehdipour Ghazi, Nicki Detlefsen, Andreas David Lauritzen, Abraham George Smith, Marleen de Bruijne, Bulat Ibragimov, Jens Petersen, Martin Lillholm, Jon Middleton, Stine Hasling Mogensen, Hans Christian Thorsen-Meyer, Anders Perner, Marie Helleberg, Benjamin Skov Kaas-Hansen, Mikkel Bonde, Alexander Bonde, Akshay Pai, Mads Nielsen, and Martin Sillesen. Developing and validating COVID-19 adverse outcome risk prediction models from a bi-national European cohort of 5594 patients. *Scientific Reports*, 11(1), 2021.

[9] Dhruv Patel, Vikram Kher, Bhushan Desai, Xiaomeng Lei, Steven Cen, Neha Nanda, Ali Gholamrezanezhad, Vinay Duddalwar, Bino Varghese, and Assad A Oberai. Machine learning based predictors for COVID-19 disease severity. *Scientific Reports*, 11(1), 2021.

[10] Authors H Eskes and J L M Schilderink. Risk factors for ICU admission, long-term stay and mortality in hospitalized COVID-19 patients.

[11] Neha Paranjape, Lauren L Staples, Christina Y Stradwick, Herman Gene Ray, and Ian J Saldanha. Development and validation of a predictive model for critical illness in adult patients requiring hospitalization for COVID-19. *PLoS ONE*, 16(3 March), 2021.

[12] Joshua M Wang, Wenke Liu, Xiaoshan Chen, Michael P Mcrae, John T Mcdevitt, and David Fenyö. Title: Predictive modeling of morbidity and mortality in COVID-19 hospitalized patients and its clinical implications.

[13] Zirun Zhao, Anne Chen, Wei Hou, James M. Graham, Haifang Li, Paul S. Richman, Henry C. Thode, Adam J. Singer, and Tim Q. Duong. Prediction model and risk scores of ICU admission and mortality in COVID-19. *PLoS ONE*, 2020.

[14] Mohammad Amini, Jalal Rezaeenour, and Esmaeil Hadavandi. Effective intrusion detection with a neural network ensemble using fuzzy clustering and stacking combination method. *Journal of Computing and Security*, 1(4):293–305, 2014.

[15] Maryam AlJame, Imtiaz Ahmad, Ayyub Imtiaz, and Ameer Mohammed. Ensemble learning model for diagnosing COVID-19 from routine blood tests. *Informatics in Medicine Unlocked*, 21, 2020.

[16] Shaokang Hou, Yaoru Liu, and Qiang Yang. Real-time prediction of rock mass classification based on TBM operation big data and stacking technique of ensemble learning. *Journal of Rock Mechanics and Geotechnical Engineering*, 2021.

[17] Smitha Rajagopal, Poornima Panduranga Kundapur, and Katiganere Siddaramappa Hareesha. A Stacking Ensemble for Network Intrusion Detection Using Heterogeneous Datasets. *Security and Communication Networks*, 2020, 2020.

[18] Shaoze Cui, Yanzhang Wang, Yunqiang Yin, T C E Cheng, Dujuan Wang, and Mingyu Zhai. A cluster-based intelligence ensemble learning method for classification problems. *Information Sciences*, 560:386–409, 2021.

[19] Ghasem Shakourian Ghalejoogh, Hussain Montazery Kordy, and Farideh Ebrahimi. A hierarchical structure based on Stacking approach for skin lesion classification. *Expert Systems with Applications*, 145, 2020.

[20] Sanjay ; Khaled; Ranka and Vineet Singh. An efficient k-means clustering algorithm, 1997.

[21] Marcel R Ackermann, Johannes Blömer, Daniel Kuntze, and Christian Sohler. Analysis of agglomerative clustering. In *Algorithmica*, volume 69, pages 184–215, 2014.

[22] Martin Ester, Hans-Peter Kriegel, Jiirg Sander, and Xiaowei Xu. A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, 1996.

[23] James C Bezdek. FCM: THE FUZZY c-MEANS CLUSTERING ALGORITHM, 1984.

[24] K Sasirekha and P Baby. Agglomerative Hierarchical Clustering Algorithm-A Review. *International Journal of Scientific and Research Publications*, 3(3), 2013.

[25] Alena Lukasov. HIERARCHICAL AGGLOMERATIVE CLUSTERING PROCEDURE.

[26] Kamran Khan, Saif Ur Rehman, Kamran Aziz, Simon Fong, S Sarasvady, and Amrita Vishwa. DBSCAN: Past, present and future. In *5th International Conference on the Applications of Digital Information and Web Technologies, ICADIWT 2014*, pages 232–238. IEEE Computer Society, 2014.

[27] Glenn W Milligan and Martha C Cooper. A Study of the Comparability of External Criteria for Hierarchical Cluster Analysis. *Multivariate Behavioral Research*, 21(4):441–458, 1986.

[28] Lawrence Hubert and Phipps Arabie. Comparing partitions. *Journal of Classification 1985 2:1*, 2(1):193–218, 1985.

[29] Laurens Van Der Maaten and Geoffrey Hinton. Visualizing Data using t-SNE, 2008.

CHAPTER 4. A HYBRID MACHINE LEARNING-OPTIMIZATION ALGORITHM TO DISTINGUISH SUPER-AGERS FROM COGNITIVE DECLINERS USING NEURAL NETWORKS DATA

Mohammad Fili[1], Parvin Mohammadiarvejeh[1], Brandon S. Klinedinst[2], Qian Wang[3], Shannin Moody[4], Neil Barnett[4], Amy Pollpeter[5], Brittany Larsen[6], Tianqi Li[7], Sara A. Willette[8], Jonathan P. Mochel[9], Karin Allenspach[10], Guiping Hu[1], Auriel A. Willette[3,5,6,7,8,11]

[1] Department of Industrial and Manufacturing Systems Engineering, Iowa State University, Ames, IA, USA

[2] Department of Medicine, University of Washington, Seattle, WA, USA

[3] Department of Food Science and Human Nutrition, Iowa State University, Ames, IA, USA

[4] Department of Human Development and Family Studies, Iowa State University, Ames, IA, USA

[5] Bioinformatics and Computational Biology Graduate Program, Iowa State University, Ames, IA, USA

[6] Neuroscience Graduate Program, Iowa State University, Ames, IA, USA

[7] Genetics and Genomics Graduate Program, Iowa State University, Ames, IA, USA

[8] IAC Tracker Inc., Ames, IA, USA

[9] Department of Biological Sciences, Iowa State University, Ames, IA, USA

[10] Department of Veterinary Clinical Sciences, Iowa State University, Ames, IA, USA

[11] Department of Neurology, University of Iowa, Iowa City, USA

Modified from a manuscript to be submitted to *Alzheimer's & Dementia*

4.1 Abstract

Aging is often associated with a higher risk of dementia and progressive decline in different cognitive domains. However, there exists a group of adults, called Super-Agers, with at least 80

years of age, who have a comparable cognitive function to young adults. Studying Super-Agers, and exploring neural differences between this group and normal Cognitive Decliners may contribute to cognitive aging and mitigate Alzheimer's Disease (AD). In this study, we proposed a hybrid algorithm of machine learning and optimization that successfully distinguishes between Super-Agers and Cognitive Decliners using rsfMRI data and demographics information. The proposed algorithm, named Optimal Labeling using Bayesian Optimization (OLBO), uses an iterative process to improve the labeling where the final goal is to maximize a classifier's performance. OLBO has an internal procedure for label assignment where it takes advantage of cognitive test information. The parameters in the labeling procedure are tuned via Bayesian optimization (BO). We compared the OLBO algorithm against two sets of baseline models: univariate and multivariate random baseline models. In each univariate model, one cognitive test pertinent to a specific time point is used to find the labels, whereas, in the multivariate model, all cognitive tests are applied without any optimization. We showed that OLBO outperforms all baseline models and reaches an AUC of 85% with a precision of 81%, accuracy of 78%, sensitivity of 77%, and specificity of 79%.

4.2 Introduction

Aging is frequently characterized by a cognitive decline in several domains [1, 2]. The negative association between age and performance of all cognitive areas like executive function, processing speed, visuospatial skills, and particularly memory are well known [3, 4]. Furthermore, a significant decline in episodic memory is shown in adults with Mild Cognitive Impairment (MCI) and Alzheimer's Disease (AD) [5, 6, 7]. Yet, there is substantial variation in how cognitive function evolves in mid- and late-life adults. Some adults aged 80 years or older, called Super-Agers, show cognitive performance that is similar to healthy middle-aged adults, particularly for declarative memory [8, 9, 10, 11]. Similarly, we have recently found that roughly 20% of middle-aged to early-aged adults in UK Biobank show gains over time in fluid intelligence, a form of executive function strongly impacted by aging [12]. Cognitive health is considered an

important factor in successful aging [13]. Therefore, it is necessary to investigate the neural correlates of younger Super-Agers, to determine how brain structure and function differ compared to cognitively unimpaired adults who nonetheless show age-related cognitive decline.

Some neuroimaging studies focused on Super-Agers have examined regional brain volumes in areas mostly related to memory. Super-Agers aged 80 years or older, based on superior episodic memory, showed minimal differences in cortical thickness compared to middle-aged adults. Moreover, compared to age-matched controls, Super-Agers had more regional cortical thickness in areas critical for memory consolidation and retrieval, including the precuneus, the posterior cingulate, and the prefrontal cortices [10]. Another study found that Super-Agers had the thickest anterior cingulate cortex and the least amount of neurofibrillary degeneration compared to healthy people of the same age, as well as participants with MCI and AD [9]. Although studies have largely converged on differences in brain structure between adults with various levels of cognitive abilities, these methods are typically limited to group comparisons. For precision medicine, however, developing techniques to distinguish individuals is essential; machine learning and prediction models could play a great role in this matter [14].

Beyond structural imaging, resting state functional Magnetic Resonance Imaging (rsfMRI) provides the opportunity to understand how neural network functional connectivity contributes to cognitive function [15]. For example, rsfMRI has frequently been used in classification models for neurodegenerative diseases [16]. For late-life Super-Agers, a recent study [17] found that 32 Super-Agers compared to 58 age-matched controls, had better functional connectivity among brain areas critical for declarative memory. Ensemble classification using machine learning achieved very high discrimination between these groups. However, larger sample sizes are warranted to avoid potential model overfitting and to achieve robust solutions. It is also important to establish how intrinsic functional connectivity [18] in neural networks explains longitudinal changes in cognitive performance between adults who show cognitive gains versus decline over time.

The following study leveraged brain rsfMRI and multi-visit cognitive assessments from the UK Biobank. The goals of this study were to: 1) introduce a PCA-based labeling approach for the initial characterization of latent groups with different cognitive trajectories; 2) develop a novel labeling mechanism; and 3) create a hybrid algorithm of machine learning and optimization to optimally distinguish between largely middle-aged participants who showed cognitive gains (i.e., "Super-Agers") versus cognitive decline.

4.3 Methods

In this study, 15,939 participants were a part of the UK Biobank cohort, a long-term prospective biomedical research study including comprehensive questionnaires, physical measures, cognitive function, imaging, and biological sampling on a cohort of more than half of a million UK adults [19]. Participants were aged 40 to 70 years at the baseline visit in 2006-2010 (t_1). This visit and the first follow-up visit in 2012-2013 (t_2) consisted of: 1) informed consent; 2) touchscreen questionnaire; 3) verbal interview; 4) eye measures; 5) anthropometric measures; and 6) blood/urine sample collection. In the subsequent two follow-up visits in 2014 (t_3) and 2019 (t_4), additional imaging was done of the brain, heart, and body [20, 21]. Socio-demographic characteristics, occupation, lifestyle, and cognitive function were gathered by questionnaires using touchscreens or laptops. The UK Biobank protocol was approved by the Northwest Multi-Centre Research Ethics Committee.

UK Biobank brain imaging has been done in three centers with identical scanners (3T Siemens Skyra) with the same standards and protocol [22]. Briefly, UK Biobank functional modalities included task-based functional MRI (tfMRI) and resting state functional MRI (rsfMRI). In the next section, the definition of rsfMRI and data acquisition is provided.

4.3.1 Demographic Data

The demographics data include information on the age, gender, body mass index (BMI), socioeconomic class, handedness, education level, skin color, waist circumference, tobacco use, and

tobacco type of 15,939 participants. Age is defined as the age in years of each participant at their baseline visit. Socioeconomic class is defined by the participant's average total household income. There were five options provided based on UK Biobank Data-Coding to be selected by participants in British pounds, which included "Less than 18000", "18000 to 30999", "31000 to 51999", "52000 to 100000", and "Greater than 100000". These groups were labeled as Under, Lower, Middle, Upper-Middle, and Upper-class. For education level, a categorical variable is used with the following levels: college or other higher-level status; post-secondary or vocational; secondary; and none of the previous ones. Finally, tobacco smoking status was defined as a categorical variable with levels: "Never smoked," "Previously smoked," and "Currently smoking." The pair plots for the continuous variables in this dataset are shown in Supplementary Figure S1. The bar plots for the frequency of the levels of each categorical variable in the demographic dataset are shown in Supplementary Figure S2.

4.3.1.1 rsfMRI Data

rsfMRI is an indirect measure of neural activity based on changes in blood oxygenation and neural metabolic demand [23] in the absence of a particular task. In this manner, 'network nodes' have been defined as voxel clusters that fluctuate most strongly in corresponding brain regions, while 'network edges' have been defined as nodes that have weaker effects [24]. In this study, we focused on a low-dimensional decomposition of the brain into 21 independent components corresponding to functional neural networks [25].

Since raw imaging data are not useful directly, particularly for non-imaging experts, UK Biobank used a processing pipeline to generate both processed images and image-deprived phenotypes (IDPs) [22]. In rsfMRI, IDPs represent edge connectivity strengths and node fluctuation amplitudes. Key acquisition parameters for rsfMRI were spatial resolution = 2.4mm, TR = 0.735s, factor = 8 multiband accelerator. The rsfMRI data has 21 non-noise components for time point t_3. Figure 4.1 shows the Pearson correlation between different components.

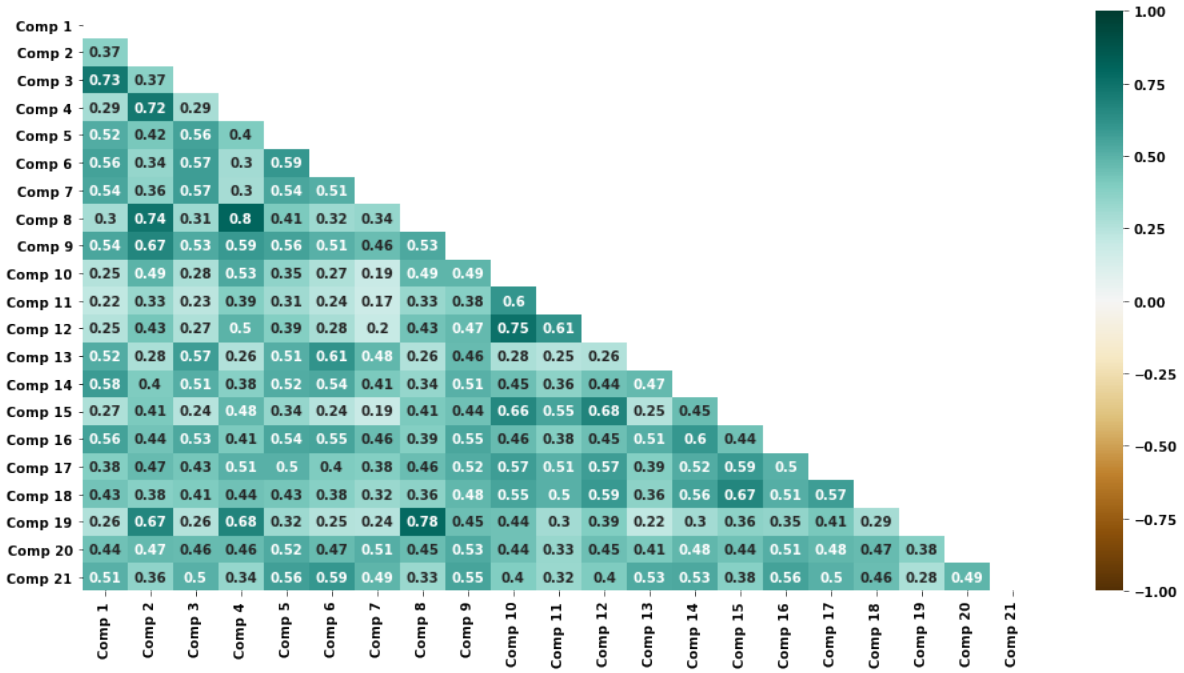

Figure 4.1 Pearson correlation values for rsfMRI components.

4.3.2 Cognitive Testing

In UK Biobank, cognitive testing corresponded to an early phase of "quick" cognitive tests and a late phase with standardized neuropsychological tests. Specifically, for the early phase, timepoints t_1, t_2, and t_3 had a roughly 5-minute series of tests for fluid intelligence (FI), prospective memory (PM), pairs matching memory (PMM), and reaction time (RT). Despite their idiosyncratic nature, the general cognitive ability tapped by these tests highly corresponded to standardized neuropsychological tests in a sub-sample of UK Biobank participants [26]. For the late phase at timepoints t_3 and t_4, standardized tests included Trails Making Test parts A and B (TMT-A, TMT-B), as well as Symbol Digit Substitution (SDS). Appendix B. SUPPLEMENTARY TEXT describes these tests in detail. Due to skewness, the following tests were transformed: pairs matching memory ($log(x+1)$), reaction time (log), and Trails Making Tests parts A and B (log).

Given these two different sets of cognitive tests, labeling methods to distinguish cognitive trajectory types were conducted separately for early-phase and late-phase tests. This corresponds to cognitive performance largely before rsfMRI conducted at t_3 versus performance conducted during the same visit as rsfMRI and later at t_4.

We denoted the dataset with the cognitive test information as G, where G has 15,003 participants. The time points of interest in this study are t_1 and t_3. Therefore, overall, we have eight features in dataset G:

$$G = (FI_1\ FI_3\ PM_1\ PM_3\ PMM_1\ PMM_3\ RT_1\ RT_3) \tag{4.1}$$

The histograms of the change in cognitive test values between time points t_1 and t_3 is shown in Supplementary Figure S3.

4.3.3 Data Preparation

In order to prepare the data for the modeling phase, we applied multiple preprocessing steps. We denoted the dataset of integrated MRI and demographics as X. To bring continuous variables into the same scale, we used standardization as below:

$$x_{ij}^{new} = \frac{x_{ij} - \overline{x_j}}{s_{x_j}} \tag{4.2}$$

Where x_{ij} is the i-th observation in the j-th feature in X, and $\overline{x_j}$ and s_{x_j} are the training mean and standard deviation for the j-th feature, respectively. The final preprocessed dataset is denoted as $\mathbb{X}$.

Next, in dataset G, we removed observations for which cognition test results were not available for any of the time points t_1 or t_3. Then, we randomly split the observations into training (G_{train}), validation (G_{val}), and testing (G_{test}) sets. We then standardized them, using Equation (4.2), and created $\mathbb{G}_{train}$, $\mathbb{G}_{val}$, and $\mathbb{G}_{test}$, respectively. The same split was applied to the dataset $\mathbb{X}$.

4.3.4 Classification

The predictability of the class labels in a problem can be attributed to two sources: 1) the power of the classification model to capture the underlying pattern and distinguish between the classes, and 2) the degree to which the classes are well separated in essence. In our study, we did not have the class labels for the participants. In other words, the classes were latent. Therefore, a supervised learning algorithm could not be used. Thus, we needed to find a mechanism for assigning labels to the participants based on some criteria and then predicting those classes using rsfMRI data.

To achieve this goal, we propose a hybrid ML-Opt algorithm, named Optimal Labeling using Bayesian Optimization (OLBO) algorithm, that automatically assigns labels to the participants using the information from the cognitive tests, such that it maximizes the predictive capability of the classification model. We utilized the early-phase cognitive tests to find the optimal labeling for distinguishing latent cognitive trajectory groups. The proposed algorithm enabled superior predictions via an unsupervised labeling procedure. The algorithm consisted of two components: machine learning and optimization. Logistic regression was used for the predictive side of the algorithm. Despite its simplicity and interpretability, it can be equipped with the feature selection mechanism through regularization. For the other side of the algorithm, to optimize the labeling procedure, we incorporated Bayesian Optimization (BO). See Appendix B. SUPPLEMENTARY TEXT for detailed descriptions of the logistic regression, Bayesian Optimization, OLBO algorithm, and its internal procedures, including labeling and initialization.

4.3.5 Baseline Models

In this study, we used two different sets of baseline models to compare against our proposed algorithm. The first category of baseline models is a univariate approach that utilizes individual cognitive tests to distinguish between Super-Agers and Cognitive Decliners. For this purpose, we incorporated the cognition data, one test at a time, to categorize the participants. We defined two

percentiles $\mathcal{P}_1$ and $\mathcal{P}_2$ that determine the Super-Agers and Cognitive Decliners, where:

$$\mathcal{P}_2 = 1 - \mathcal{P}_1 \tag{4.3}$$

Since higher FI and PM scores reflect better cognitive performance, participants with test values greater than $\mathcal{P}_2$ were labeled as Super-Agers, and those below $\mathcal{P}_1$ were categorized as Cognitive Decliners. For PMM and RT where higher values translate to slower or worse cognitive performance, we used the opposite labeling. Different values of $\mathcal{P}_1$, from the set of values 0.2, 0.3, and 0.4, were tested in each baseline model. Logistic regression was used as the classifier and was tuned using a grid search approach.

The second baseline model defined is a multivariate model that utilizes random samples from the variable space ($\mathcal{VS}$) to label the participants. The labeling procedure used is the same as in OLBO (see Labeling Procedure), except that it does not employ any optimization tools. Instead, it takes a random procedure to choose the samples from the variable space. In this baseline model, a random sampling scheme was defined in which 2600 sets of random values for the variables are selected according to the distributions described in Supplementary Text, OLBO Algorithm. Then, a logistic regression classifier was trained to predict the class labels.

4.3.6 Evaluation Metrics

Different metrics were used to evaluate our algorithm. Terms were defined as:

- True Positives, or tp: Number of Super-Agers that are predicted as Super-Agers
- False Positives, or fp: Number of Cognitive Decliners that are predicted as Super-Agers
- True Negatives,or tn: Number of Cognitive Decliners that are predicted as Cognitive Decliners
- False Negatives, or fn: Number of Super-Agers that are predicted as Cognitive Decliners

The evaluation metrics and their equations are:

$$Accuracy = \frac{tp + tn}{tp + tn + fp + fn} \tag{4.4}$$

$$Precision = \frac{tp}{tp + fp} \tag{4.5}$$

$$Recall(Sensitivity) = \frac{tp}{tp + fn} \tag{4.6}$$

$$Specificity = \frac{tn}{tn + fp} \tag{4.7}$$

$$F1\ Score = \frac{2 * precision * recall}{precision + recall} \tag{4.8}$$

4.4 Results

In this section, first, we explore the demographics of the participants in the final optimal grouping. We also investigate the difference between the Super-Agers and Cognitive Decliners across different neural components. Then, we show the numeric results obtained from the set of baseline models and the OLBO algorithm. Finally, we describe the model's performance with respect to the set of parameters selected in each trial and the improvement of loss function over time.

4.4.1 Demographic Information of Cognitive Groups

We will show in Section 4.4.3 that the OLBO algorithm is the optimal labeling solution. In this section, first, we explore the diversity of participants in terms of their demographics within each of the groups: Super-Agers and Cognitive Decliners. Table 4.1 shows the descriptive statistics, including the number of participants and the corresponding percentages in each class for categorical features and the mean and standard deviation for continuous variables. We also conducted a univariate significance test for each of the variables in the demographic dataset. We employed a Chi-square independence test for the categorical variables and a two-sample two-sided Kolmogorov-Smirnov test for the continuous features. The P-value for each of these tests is reported in Table 4.1.

Table 4.1 Prediction of variance at Env positions in the high-mannose patch by KBC and other algorithms.

Variable	Level	Unit	Decliners	Super-Agers	P-value
Sex	Male	No. (%)	344 (40.2)	512 (59.8)	**0.018** [a]
	Female		356 (46.1)	416 (53.9)	
Household Income	Under Class	No. (%)	118 (60.5)	77 (39.5)	**<0.001**
	Lower Class		200 (58.7)	141 (41.3)	
	Middle Class		187 (39.0)	292 (61.0)	
	Upper-middle Class		158 (31.9)	338 (68.1)	
	Upper Class		37 (31.6)	80 (68.4)	
Handedness	Ambidextrous	No. (%)	9 (64.2)	5 (35.7)	0.13
	Left-Handed		60 (38.2)	97 (61.8)	
	Right-Handed		631 (43.3)	826 (56.7)	
Education	Secondary Education	No. (%)	93 (53.8)	80 (46.2)	**<0.001**
	Post-Secondary or Vocational		117 (40.0)	176 (60.0)	
	College or Similar		429 (39.7)	651 (60.3)	
	Other Education		61 (74.3)	21 (25.7)	
Skin Color	Brown-Skinned	No. (%)	4 (80.0)	1 (20.0)	0.16
	Olive-Skinned		134 (40.6)	196 (59.4)	
	Pale-Skinned		562 (43.4)	731 (56.5)	
Tobacco Use	Non-Smoker	No. (%)	441 (41.8)	613 (58.2)	**0.03**
	Prior Smoker		232 (47.1)	261 (52.9)	
	Smoker		27 (33.3)	54 (66.7)	
Tobacco Type	Cigars Pipes	No. (%)	3 (37.5)	5 (62.5)	0.36
	Hand-Rolled Cigs		4 (33.3)	8 (66.7)	
	Manufactured Cigs		10 (29.4)	24 (70.6)	
	Non-Smoker		683 (43.4)	891 (56.6)	
Age	-	Years	68.53 ± 6.7 [b]	58.56 ± 7.1	**<0.001**
BMI	-	Kg/m^2	18.11 ± 3.0	18.29 ± 3.1	0.95
Waist Circumference	-	cm	88.83 ± 11.7	88.34 ± 12.1	0.59

[a] P-value in bold font shows significance (P-value < 0.05)

[b] $mean \pm standard\ deviation$

4.4.2 Neural Network Comparisons

We examined the distribution of rsfMRI for different components with respect to the two classes, Super-Agers, and Cognitive Decliners (see Supplementary Figure S4). We also investigated the difference in the cognition tests among the Super-Agers and Cognitive Decliners, as shown in Supplementary Figure S5. Briefly, Cognitive Decliners showed the worst test performance or slower reaction time except for fluid intelligence. Furthermore, except for FI at t_1, all cognitive tests showed statistical significance between Super-Agers and Cognitive Decliners.

4.4.3 Numerical Results

In this section, we gathered the numeric output obtained from the OLBO algorithm and baseline models. As explained in the previous sections, the ultimate goal of this study is to find a labeling procedure that results in the best separation between Super-Agers and Cognitive Decliners groups. In other words, we addressed this question "how the classification boundary should be defined between the Super-Agers and Cognitive Decliners such that the subsequent labeling yields the best predictability?" For this purpose, we compared our proposed algorithms against two sets of baseline models, i.e., univariate and random multivariate models. The former utilizes one cognitive test at a time, and the latter employs all cognitive tests but without any optimization and through a random sampling procedure. We gathered the numerical results in Table 4.2. It shows that OLBO outperformed all baseline models. The total sample size used by the OLBO algorithm was 1,628 participants.

The AUC for the proposed algorithm reached 85%, with a precision of 81% and a recall value of 77%. Figure 4.2 shows the receiver operating characteristic (ROC) curves for the OLBO algorithm and baseline models.

Table 4.2 Prediction of variance at Env positions in the high-mannose patch by KBC and other algorithms.

Model	Accuracy	Precision	Sensitivity	Specificity	F1-Score	AUC
$FI_1(\mathcal{P}_1 = 0.2)$	0.65	0.67	0.74	0.52	0.7	0.7
$FI_1(\mathcal{P}_1 = 0.3)$	0.6	0.55	0.73	0.5	0.6	0.67
$FI_1(\mathcal{P}_1 = 0.4)$	0.61	0.65	0.71	0.48	0.68	0.64
$FI_3(\mathcal{P}_1 = 0.2)$	0.66	0.67	0.75	0.55	0.71	0.72
$FI_3(\mathcal{P}_1 = 0.3)$	0.62	0.55	0.72	0.53	0.62	0.69
$FI_3(\mathcal{P}_1 = 0.4)$	0.61	0.62	0.71	0.5	0.67	0.67
$PM_1(\mathcal{P}_1 = 0.2)$	0.59	0.91	0.6	0.52	0.72	0.59
$PM_1(\mathcal{P}_1 = 0.3)$	0.59	0.91	0.6	0.52	0.72	0.59
$PM_1(\mathcal{P}_1 = 0.4)$	0.59	0.91	0.6	0.52	0.72	0.59
$PM_3(\mathcal{P}_1 = 0.2)$	0.62	**0.92**	0.61	0.63	0.74	0.66
$PM_3(\mathcal{P}_1 = 0.3)$	0.62	**0.92**	0.61	0.63	0.74	0.66
$PM_3(\mathcal{P}_1 = 0.4)$	0.62	**0.92**	0.61	0.63	0.74	0.66
$PMM_1(\mathcal{P}_1 = 0.2)$	0.58	0.58	0.56	0.61	0.57	0.61
$PMM_1(\mathcal{P}_1 = 0.3)$	0.57	0.64	0.52	0.62	0.58	0.58
$PMM_1(\mathcal{P}_1 = 0.4)$	0.57	0.55	0.52	0.61	0.53	0.58
$PMM_3(\mathcal{P}_1 = 0.2)$	0.58	0.5	0.51	0.63	0.5	0.59
$PMM_3(\mathcal{P}_1 = 0.3)$	0.58	0.55	0.53	0.62	0.54	0.6
$PMM_3(\mathcal{P}_1 = 0.4)$	0.56	0.63	0.51	0.62	0.57	0.59
$RT_1(\mathcal{P}_1 = 0.2)$	0.69	0.71	0.64	0.74	0.67	0.76
$RT_1(\mathcal{P}_1 = 0.3)$	0.65	0.67	0.6	0.7	0.63	0.71
$RT_1(\mathcal{P}_1 = 0.4)$	0.62	0.62	0.58	0.65	0.6	0.67
$RT_3(\mathcal{P}_1 = 0.2)$	0.71	0.73	0.67	0.75	0.7	0.78
$RT_3(\mathcal{P}_1 = 0.3)$	0.68	0.71	0.63	0.73	0.67	0.75
$RT_3(\mathcal{P}_1 = 0.4)$	0.66	0.68	0.61	0.72	0.64	0.72
Multivariate Random	0.72	0.68	0.73	0.72	0.7	0.8
OLBO	**0.78** [a]	0.81	**0.77**	**0.79**	**0.79**	**0.85**

[a] Values in bold font show the highest value for each metric.

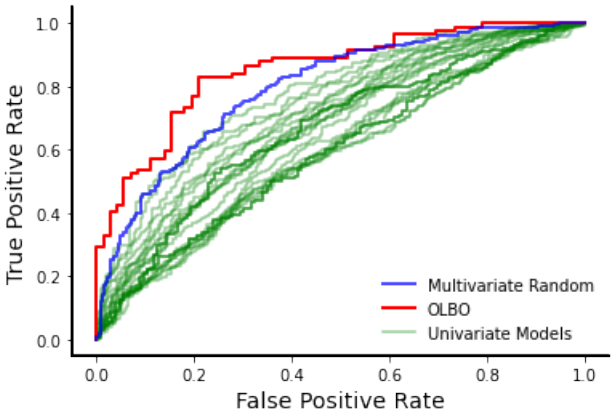

Figure 4.2 ROC curves for OLBO algorithm and baseline models.

Among univariate baseline models, *RT* showed better performance according to AUC for both time points t_1 and t_3. Moreover, the random model achieved higher AUC than the univariate baseline models. This suggests that despite sampling randomly from the variable space in the labeling procedure, the multivariate approach led to better results compared to any of the univariate baseline models. Therefore, here, using a combination of all available cognitive information is valuable.

In the next step, we analyzed the importance of each feature using the coefficients of each variable in the logistic regression model (see the coefficients in Supplementary Figure S6). The feature importance corresponding to each variable is shown in Figure 4.3. The procedure for calculating the feature importance is explained in Supplementary Text, Feature Importance Calculation. There are three colors in this plot: green color shows positive influence, variables with yellow have no effects, and red color depicts a negative impact on the odds ratio when other variables are fixed.

As shown in Figure 4.3, from the set of neural components, component 17 has the highest impact on the odds ratio: a unit increase to the value of this component will increase the odds of Super-Agers vs. Cognitive Decliners by a factor of 1.335. On the other hand, a unit increase in the value of component 13 will result in a drop in the odds ratio by a factor of 0.77 when all other features are held fixed.

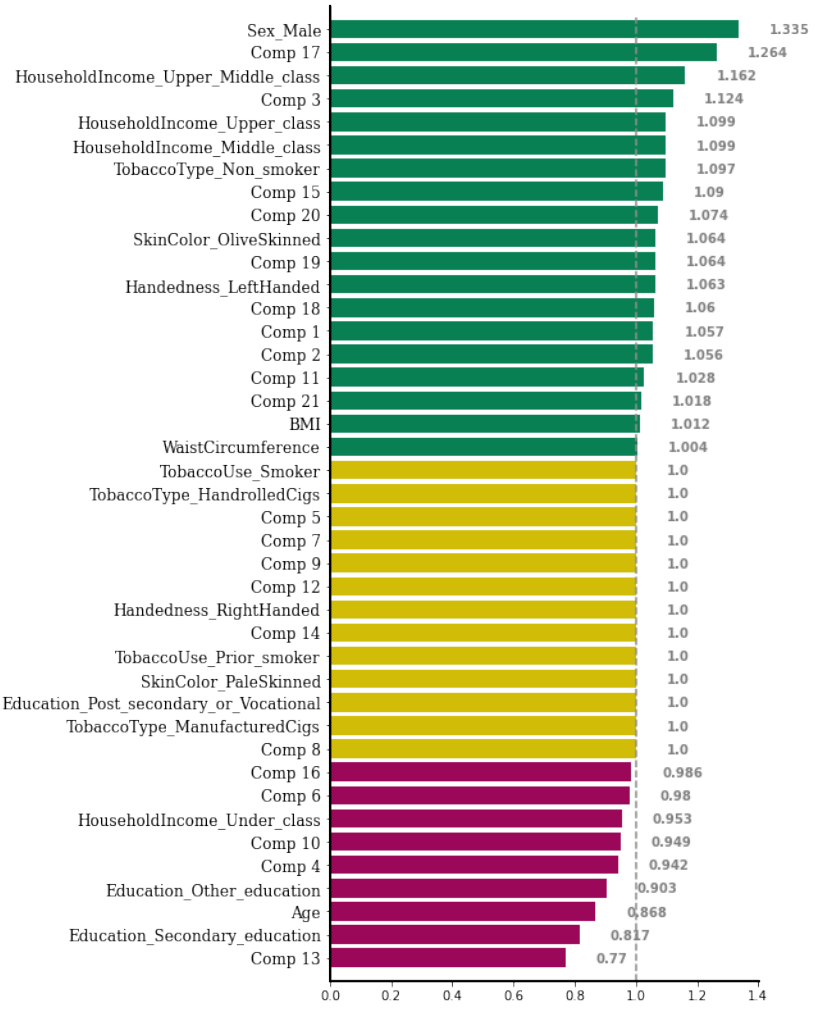

Figure 4.3 Feature Importance obtained from the OLBO algorithm.

4.4.4 Model's Details

In this section, we explored the results pertaining to the proposed algorithm. In each iteration of the OLBO algorithm, a combination of parameters sampled from the feasible domain was used, and the loss function corresponding to that trial was recorded (see Supplementary Figure S7). To examine the effect of each variable in our algorithm, we plotted the values sampled for each variable against the loss function that resulted in that trial (see Supplementary Figure S8). This figure shows how different values of weights, biases, and thresholds affect the loss function in this problem.

We also examined the effect of hyperparameters used in the logistic regression on the loss function, which can be seen in Supplementary Figure S9. We also explored the baseline models: Supplementary Figure S10 shows the loss function values obtained for different individual cognition tests (univariate baseline model) in combination with time points t_1 and t_3 and different percentiles $\mathcal{P}_1$. Supplementary Figure S11 shows the loss function values obtained for different random samples (the multivariate random baseline model).

4.5 Discussion

Although aging-related changes in different cognitive domains and higher risk of dementia are considered normal processes in humans [1, 2], the recent discovery of Supe-Agers who have been specified as adults with at least 80 years of age with comparable cognitive function to young adults [8], strengthen the viewpoint of successful cognitive aging. Studying Super-Agers, and investigating neural differences between Super-Agers and normal cognitive decliners may contribute to cognitive aging and mitigate AD.

In this matter, this study used rsfMRI data, which is an indirect measure of neural activity, to predict cognitive trajectory type for a sample of UK biobank adults with longitudinal cognitive function exams. To distinguish Super-Agers from Cognitive Decliners, cognitive function tests including *FI*, *PM*, *PMM*, and *RT* individually and in an aggregation form have been used to initialize the cognitive trajectory type labeling, and a hybrid ML-Opt algorithm designed to find the optimal labeling solution. The proposed novel OLBO algorithm outperformed all the models corresponding to individual cognitive tests with an AUC of 85%.

Furthermore, this study provided comprehensive information about how different cognitive domains in mid- and late-life are capable of recognizing cognitive trajectory type by rsfMRI data. For clarity, the role of different cognitive function exams in classification models is discussed in the following. *FI* score and *RT* cognitive tests showed better performance among the baseline models. *FI* score, which corresponds to verbal and numeric reasoning area, correlates to the default mode network (DMN) of resting state connectivity [27]. RT is related to the processing

speed domain. The association between processing speed and rsfMRI has been investigated in research papers [28, 29, 30]. With higher RT, dorsal anterior cingulate cortex shows weaker functional dissociation to a cluster in the precuneus [31]. PM and PMM cognitive tests, which measure visual, declarative memory, and prospective memory, respectively, performed relatively similar to the labeling criteria in baseline models. Executive control network (ECN) and default DMN regions are associated with poor episodic memory [28]. Furthermore, stronger connectivity in DMN is related to better working and episodic memory [28, 32]. Also, spatial working memory may show an increase after hyperbaric oxygen administration [33].

One of the analyses conducted in this study was feature importance evaluation. We described the metric used as the feature importance in Supplementary Text, Feature Importance Calculation. We observed that components 1, 2, 3, 11, 15, 17, 18, 19, 20, and 21 are influencing the odds ratio of being Super-Agers vs. Cognitive Decliners in a positive way, meaning that a unit increase in the value of any of these components, is followed by an increase in the odds ratio when the other variables are kept the same. On the other hand, components 4, 6, 8, 10, 13, and 16 turned out to be negatively influential on the odds ratio if other variables remained the same.

Among demographic features, being male increases the odds ratio the most by a factor of 1.335, and having secondary education drops the odds ratio the most by a factor of 0.817, given all other variables stay the same. For the household income, being in upper-, middle-, and upper-middle-class increases the odds ratio, while being in the under-class group negatively influences the odds ratio. As expected, age negatively affects the odds ratio, and this was found to be by a factor of 0.868 when other variables are kept constant.

4.6 Conclusion

In this study, we proposed a hybrid algorithm that incorporates machine learning and optimization to distinguish between Super-Agers and Cognitive Decliners using rsfMRI data. The proposed algorithm, OLBO, employs cognitive test information to find the optimal labeling for the participants such that it maximizes the performance of a classifier. The classifier then learns

to predict the class labels using rsfMRI and demographics information. OLBO uses Bayesian optimization as the main optimization tool for label assignment. For the classification part of the problem, a logistic regression model was used. We introduced an initialization procedure to let the model start off with a good initial set of values for the decision variable to save time and help faster convergence. We compared the performance of the OLBO algorithm against two sets of baseline models: univariate and multivariate random models. We showed that OLBO outperformed the baseline models and achieved an AUC of 85%, accuracy of 78%, precision of 81%, sensitivity of 77%, and specificity of 79%, which were all higher than any of the baseline models.

4.7 Appendix A. Supplementary Figures

4.7.1 Supplementary Figure S1

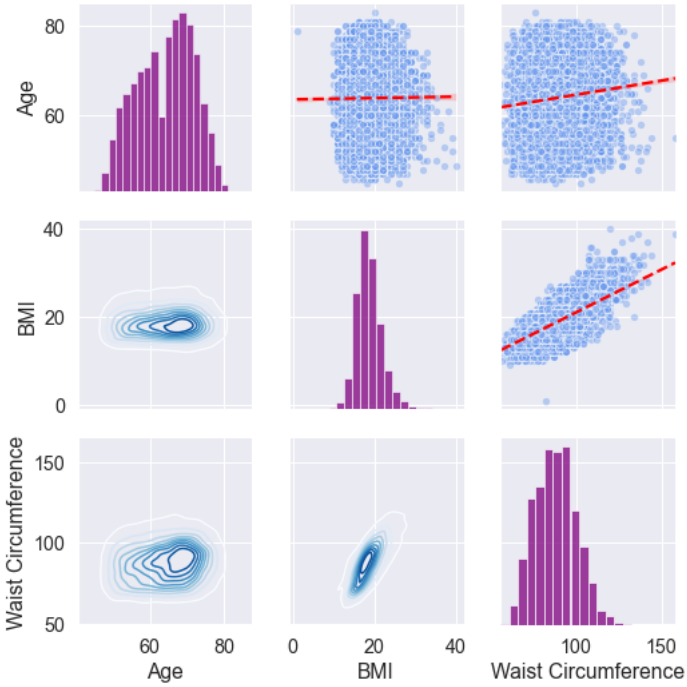

Figure 4.4 Pair plots for the continuous variables in the demographic dataset. The 2D–density plots and scatter plots are on the lower and upper triangles of the figure. A histogram of each feature is on the diagonal.

4.7.2 Supplementary Figure S2

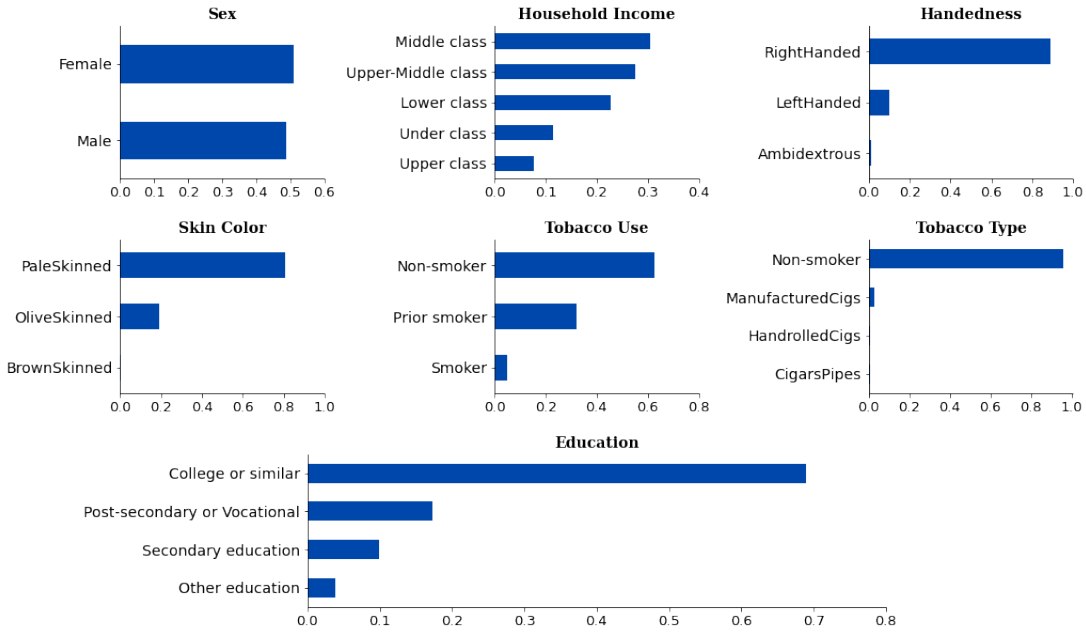

Figure 4.5 Bar plot of categorical variables for the demographic data.

4.7.3 Supplementary Figure S3

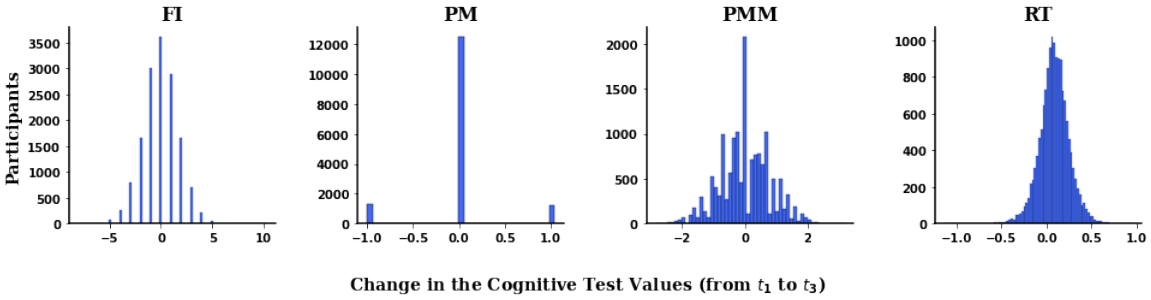

Figure 4.6 Histogram of the change in cognitive tests (from t_1 to t_3). The bandwidth for each histogram is calculated using the rule-of-thumb bandwidth selection method.

4.7.4 Supplementary Figure S4

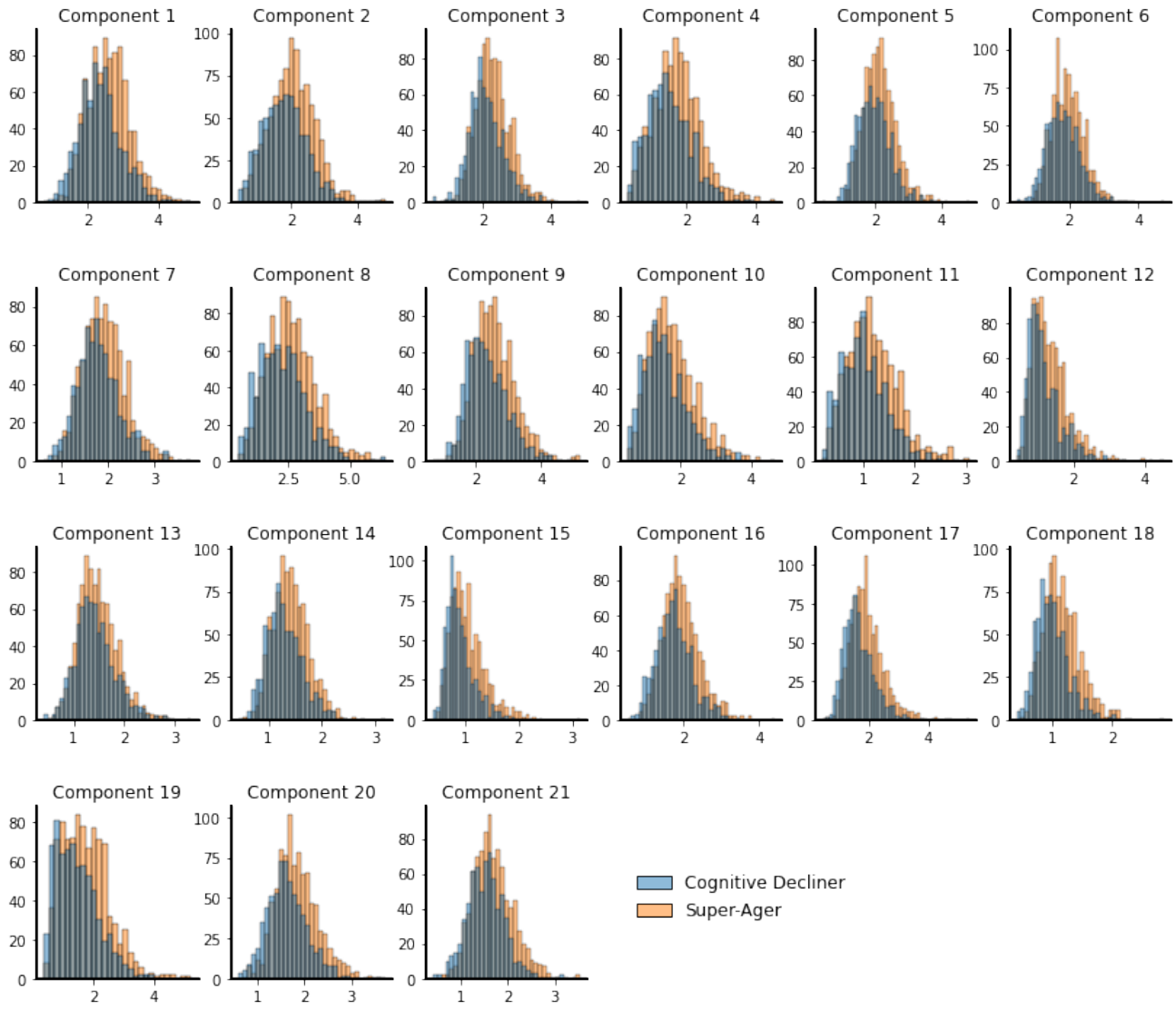

Figure 4.7 Histogram of rsfMRI values in different neural components for Super-Agers and Cognitive Decliners.

4.7.5 Supplementary Figure S5

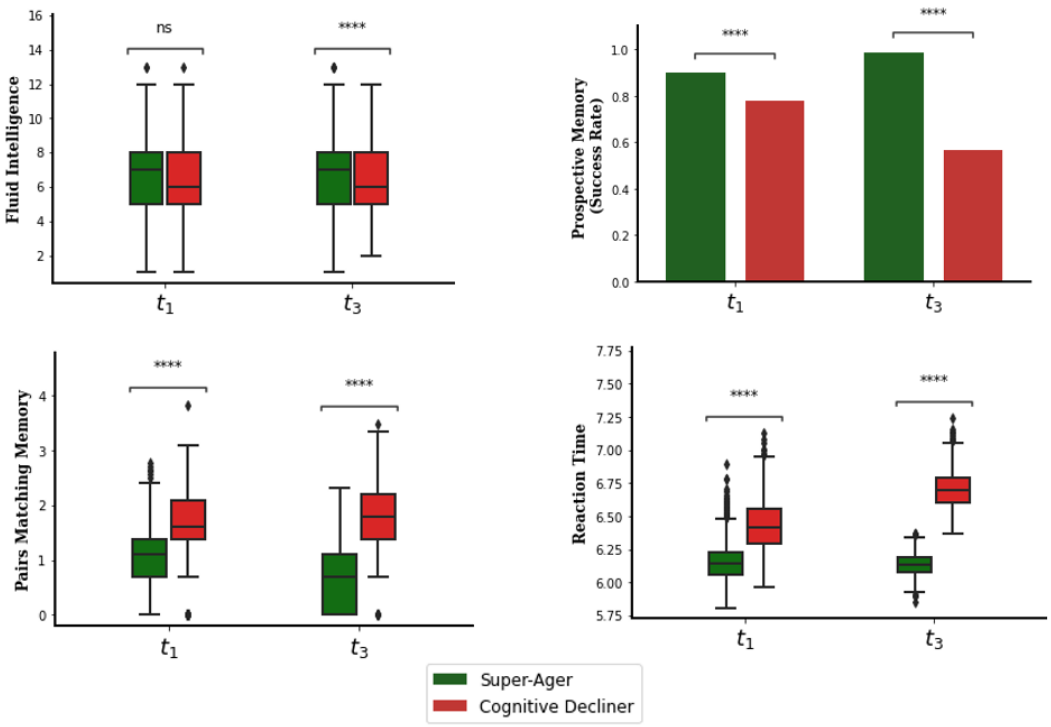

Figure 4.8 Comparison of cognitive tests between Super-Agers and Cognitive Decliners. An independent t-test was conducted for *FI*, *PMM*, an *RT* cognitive tests separately at each time point. We used a proportion z-test for PM cognitive test where we wanted to test whether the success rate (percentage of participants with correct initial answer) is the same between Super-Agers and Cognitive Decliners or not.(ns: $p > 0.05$; *: $0.01 < p \leq 0.05$; **: $0.001 < p \leq 0.01$; ***: $0.0001 < p \leq 0.001$; ****: $p \leq 0.0001$).

4.7.6 Supplementary Figure S6

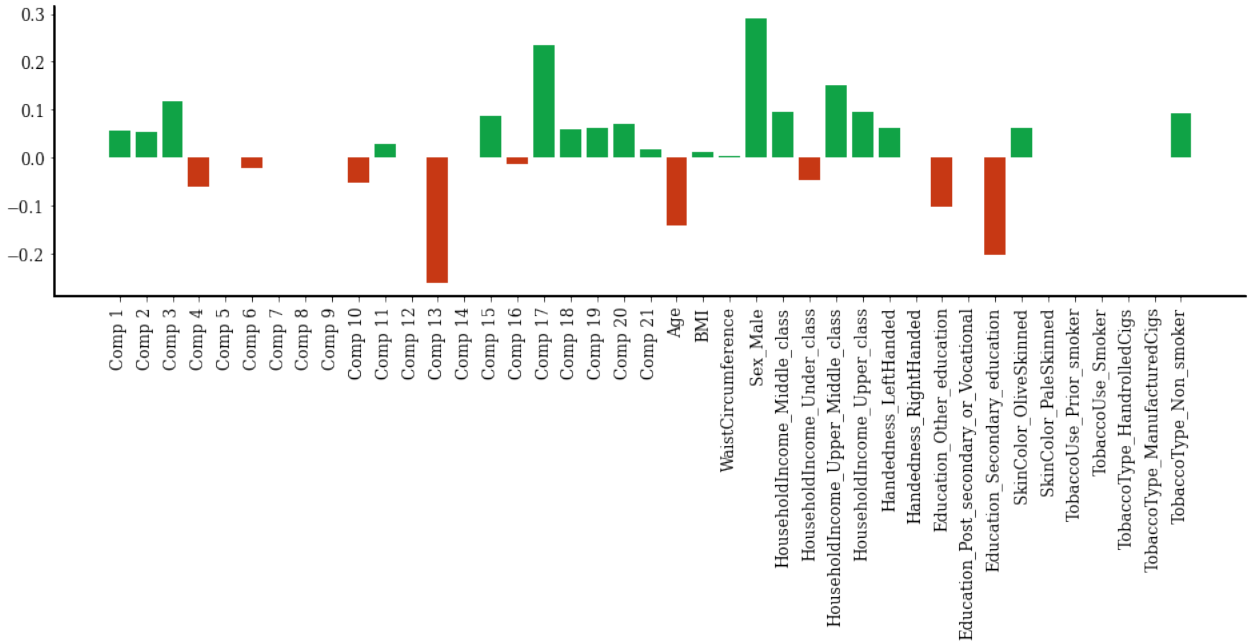

Figure 4.9 The coefficients of the tuned logistic regression model in the OLBO algorithm.

4.7.7 Supplementary Figure S7

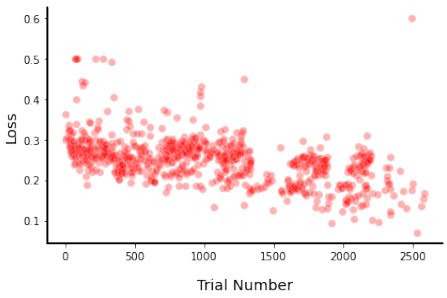

Figure 4.10 Loss function improvement in the OLBO algorithm over different iterations.

4.7.8 Supplementary Figure S8

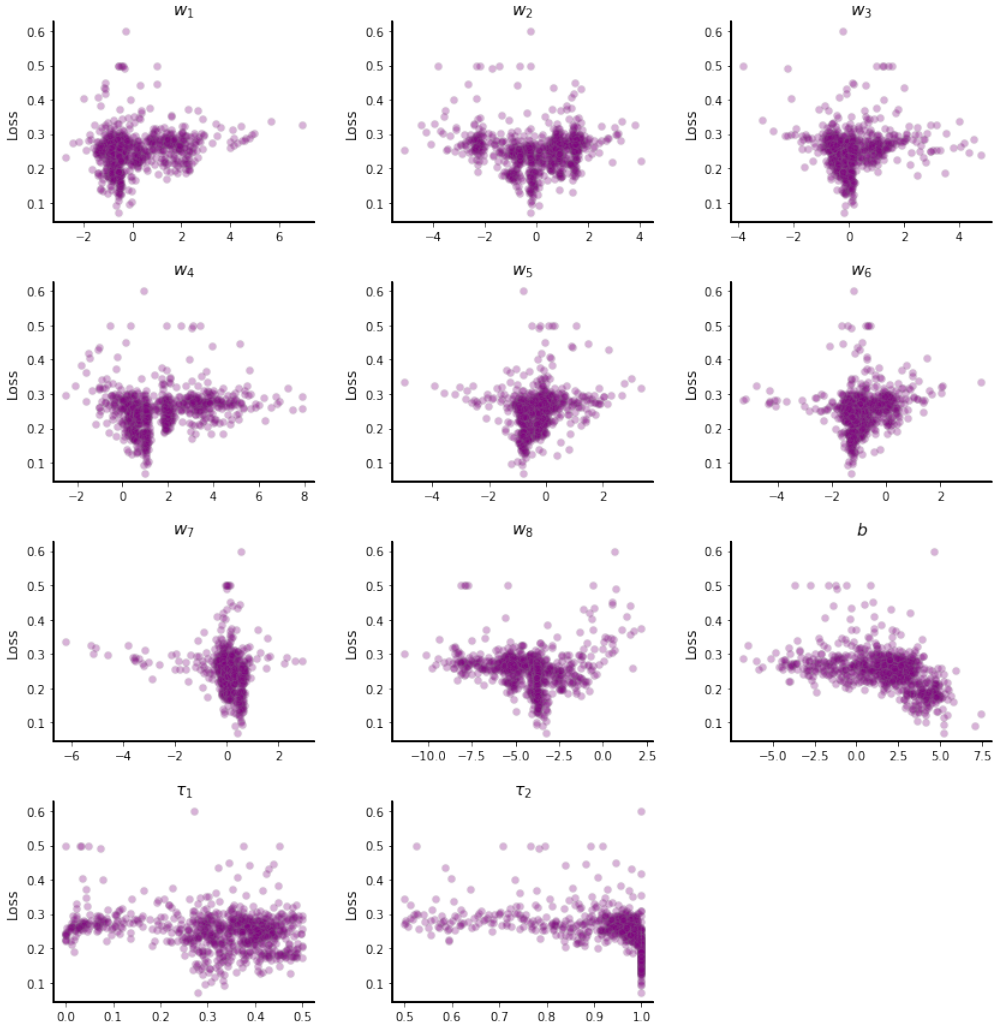

Figure 4.11 Relation between each of the OLBO's parameters with the loss function over all trials

4.7.9 Supplementary Figure S9

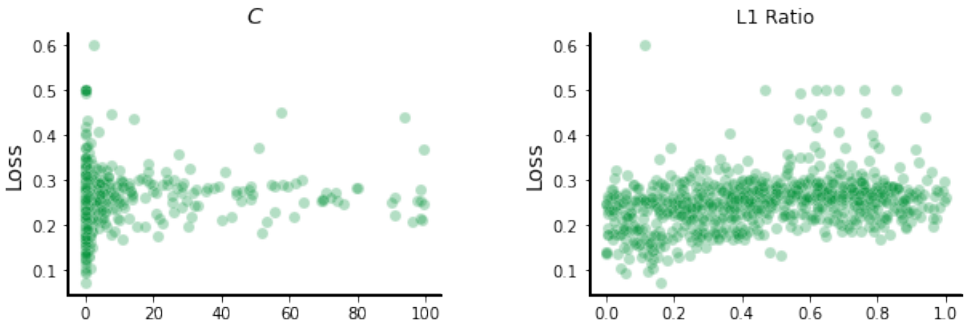

Figure 4.12 Effect of logistic regression hyperparameters on the loss function in the OLBO algorithm.

4.7.10 Supplementary Figure S10

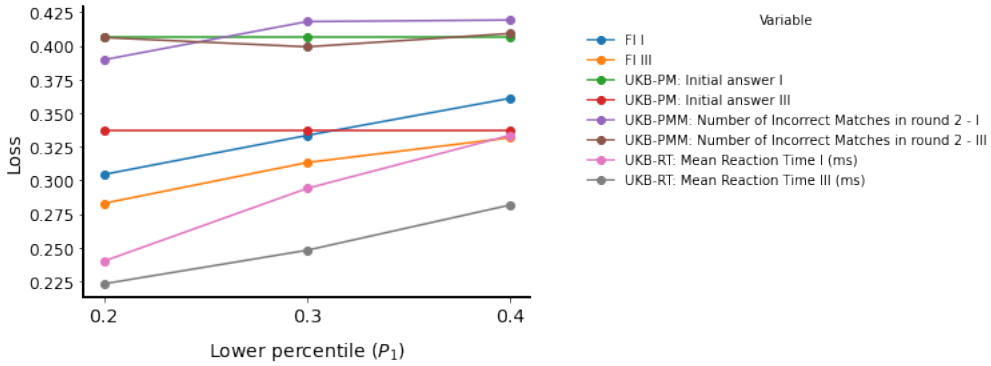

Figure 4.13 Loss function values obtained for univariate baseline models using individual cognition tests (different values of lower percentile, P_1, was tested).

4.7.11 Supplementary Figure S11

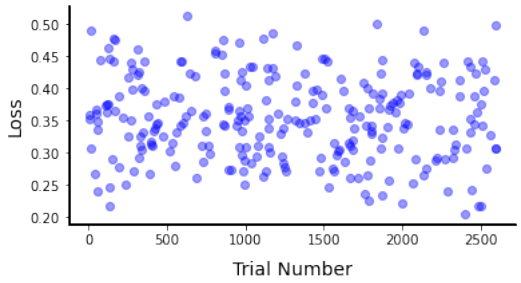

Figure 4.14 Loss function values over different trials (in each trial, a $\Theta \in \mathcal{VS}$ is randomly sampled according to the variables' distributions)

4.8 Appendix B. SUPPLEMENTARY TEXT

4.8.1 Description of Cognitive Tests

Fluid intelligence is defined as the ability to solve problems that require logic and reasoning without any prior knowledge or experience. In this test, participants answered 13 multi-choice questions, including logical, numerical, and combinational types. The score was the number of questions with the correct answer in 2 minutes (https://biobank.ndph.ox.ac.uk/showcase/label.cgi?id=100027).

For prospective memory, at the beginning of the cognitive test battery, participants were shown a message on the screen "At the end of the games, we will show you four colored shapes and ask you to touch the Blue Square. However, to test your memory, we want you to actually touch the Orange Circle instead." In the end, four shapes, including a blue square, pink star, grey cross, and orange circle, were seen by instruction of clicking on the blue square. If the participant touched the blue square, the prompt was repeated. The score was 1 if the participants correctly touched the orange circle and 0 if they touched other shapes (https://biobank.ndph.ox.ac.uk/showcase/label.cgi?id=100031).

The pairs matching memory task assessed visual memory. In this test, participants were shown a screen with several pairs of matching cards, and they were requested to memorize the matched cards as many pairs as possible. Then, they were asked to match the cards that turned face down in the fewest tries. The score was the number of errors made on the second trial of this test (https://biobank.ndph.ox.ac.uk/showcase/label.cgi?id=100030).

The reaction time test was designed to assess the reaction time based on 12 rounds of the card game. In this test, participants were shown two cards with symbols. If the symbols were matched, they were requested to touch the button-box on the desk as quickly as possible. If cards were different, they were supposed to do nothing. Five rounds were practice trials, and they were not counted. The score was the mean reaction to pressing the button in four rounds with matching cards (https://biobank.ndph.ox.ac.uk/showcase/label.cgi?id=100032).

The Trail Making Test is the computerized version of a test called the Halstead-Reitan Trail Making Test [34]. This exam evaluated executive function. In part A, participants were shown a set of random numbers (1-25), and they were asked to touch the numbers in numerical order. In part B, the numbers (1-13) and the letters (A-L) were arranged randomly on the screen. Participants were supposed to switch between touching the numbers in numeric order and letters in alphabetical order. The score was the duration to finish the test in each part (https://biobank.ndph.ox.ac.uk/showcase/label.cgi?id=505).

Symbol Digit Substitution is designed to assess working memory and processing speed. This test is similar to other substitution tests like the Wechsler Adult Intelligence Scale IV (WAIS-IV) Coding and the Symbol Digit Modalities tests [35, 36]. In this test, participants were shown a key that paired symbols (top row) with single-digit numbers (bottom row). There was a second row of symbols under the key, and the participants were supposed to enter the integer numbers which were paired with the symbols. The score was the number of correct symbol-digit matches made in 60 seconds (https://biobank.ndph.ox.ac.uk/showcase/label.cgi?id=502).

4.8.2 Logistic Regression

The logistic regression model is an extension of linear regression [37, 38], which is used for the prediction of a binary response variable. The logistic regression model [39] can be written as follows:

$$p_i = p(y_i = 1) = \frac{1}{1 + exp\big(-(\beta_0 + \beta_1 x_i)\big)} \tag{4.9}$$

We can extend this model and incorporate more predictors into the model [40]:

$$ln(\frac{p_i}{1 - p_i}) = z_i \tag{4.10}$$

Where $ln(\frac{p_i}{1-p_i})$ is the log odds ratio, and z_i is defined as:

$$z_i = \beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p \tag{4.11}$$

The log-likelihood loss function for the above model can be written as:

$$\mathcal{L}(\beta) = -\sum_{i=1}^{n} log\Big(y_i log p_i + (1 - y_i) log(1 - p_i)\Big) \tag{4.12}$$

We utilized elastic net regularization, which leverages a combination of L1-norm and L2-norm simultaneously. The corresponding loss function [41] can be described below:

$$\mathcal{L}_{elastic}(\beta) = -\sum_{i=1}^{n} log\Big(y_i log p_i + (1 - y_i) log(1 - p_i)\Big) + \lambda_1 \sum_{j=1}^{p} |\beta_j| + \lambda_2 \sum_{j=1}^{p} \beta_j^2 \tag{4.13}$$

4.8.3 Bayesian Optimization

Bayesian Optimization (BO) [42, 43] is an algorithm that develops a population of candidate solutions through the creation and sampling of Bayesian networks [44]. In BO, gradient information is not required. It is a promising tool for black-box optimization [45]. Suppose a problem of minimizing the unknown function f exists as:

$$X^* = \arg\min_{X \in \mathcal{X}} f(X) \tag{4.14}$$

The algorithm creates a surrogate for the objective function f and utilizes a Bayesian machine-learning technique to quantify the uncertainty in the surrogate function [46]. First, a prior belief is prescribed over the unknown objective function. Data are then observed, and the belief is refined in each iteration of the model via Bayesian posterior updating [47]. The Gaussian Process (GP) is a convenient prior distribution that has been used a lot [48, 49, 50, 51]. Besides the GP-based approach, there are other mechanisms to approximate the unknown function. One of the popular methods is Tree Parzen Estimator (TPE), which was employed in this study. While GP-based approach models $p(y|x)$ directly, the TPE strategy models $p(x|y)$ and $p(y)$ [52].

Another aspect of BO is the approach for sampling the next observation. The algorithm utilizes a function called acquisition function denoted as $\alpha : X \rightarrow R^{+}$, which constructs a utility function from the posterior to specify the next observation to evaluate [53]. This approach incorporates the uncertainty in the posterior to guide the exploration process [47]. There are various functions that can be used as the acquisition function, one of which, used in our study, is the expected improvement (EI) [54]. Suppose:

$$f_{min} = min\big(f^{(1)}, f^{(2)}, \ldots, f^{(n)}\big) \tag{4.15}$$

This creates the best current value for the function. Then, let us consider the uncertainty at $f(X)$ as the realization of a random variable Y which is normally distributed. Then, improvement at point X can be written as:

$$I = max\big(f_{min} - Y, 0\big) \tag{4.16}$$

Since Y is a random variable, then the I is also a random variable. Therefore, we can obtain the expected improvement by:

$$E[I(X)] = E\Big[max\big(f_{min} - Y, 0\big)\Big] \tag{4.17}$$

4.8.4 OLBO Algorithm

The OLBO Algorithm consists of three main procedures, including initialization, labeling, and optimization. OLBO starts with the cognition training set, $\mathbb{G}_t rain$. In each iteration, given a set

of values for the decision variables, denoted as the vector Θ, it assigns one of the labels "Super-Ager" or "Cognitive Decliner" to each observation. A detailed description of the labeling procedure can be found in Supplementary Text: Labeling Procedure.

In the next step, using the labels assigned, a logistic regression model is trained on the training set and then evaluated on the validation set. The area under the receiver operating characteristic (ROC) curve, AUC, is used as the primary metric for optimization. The loss function for each iteration in the BO is calculated using the following:

$$\mathcal{L}^{(j)} = 1 - AUC^{(j)} \quad \forall\ j = 0, 1, \ldots, J \tag{4.18}$$

Based on the value of the loss function and the uncertainty in the posterior, BO draws a sample from the variable space defined, which specifies the value of for the next iteration. The algorithm repeats the process J times. To make the convergence faster and help the algorithm find the optimal solution in fewer iterations, an initialization procedure was introduced that finds a starting point for the algorithm, $\Theta^{(0)}$. The initialization procedure is explained in detail in Supplementary Text: Initialization Process. The pseudocode for the algorithm is provided below:

Inputs: J, $\mathbb{X}_{train}$, $\mathbb{X}_{val}$, $\mathbb{G}_{train}$, and $\mathbb{G}_{val}$
Output: Θ^* (optimal values for decision variables)

For j from 0 to J **Do**:
  If $j = 0$ **Do**:
    Initialization and finding $\Theta^{(0)}$.
  End If
  Given $\theta^{(j)}$, apply labeling procedure, and obtain $\mathcal{y}_{train}^{(j)}$, $\mathcal{y}_{val}^{(j)}$.
  Fit logistic regression on $(\mathbb{X}_{train}, \mathcal{y}_{train}^{(j)})$.
  Evaluate the model on $(\mathbb{X}_{val}, \mathcal{y}_{val}^{(j)})$ and calculate $\mathcal{L}^{(j)}$.
  Obtain $\Theta^{(j+1)} \in \mathcal{VS}$ using the EI acquisition function
End For

Return: $\Theta^* = \max_j \Theta^{(j)}$

Figure 4.15 Pseudocode for OLBO algorithm.

The flowchart for the OLBO algorithm is as follows:

OLBO algorithm has certain variables (Θ) that can be optimized for the maximal performance of the predictive model. The vector Θ can be written as:

$$\Theta = (W^\top, b, \tau_1, \tau_2, c, \phi) \qquad \Theta \in \mathcal{VS} \tag{4.19}$$

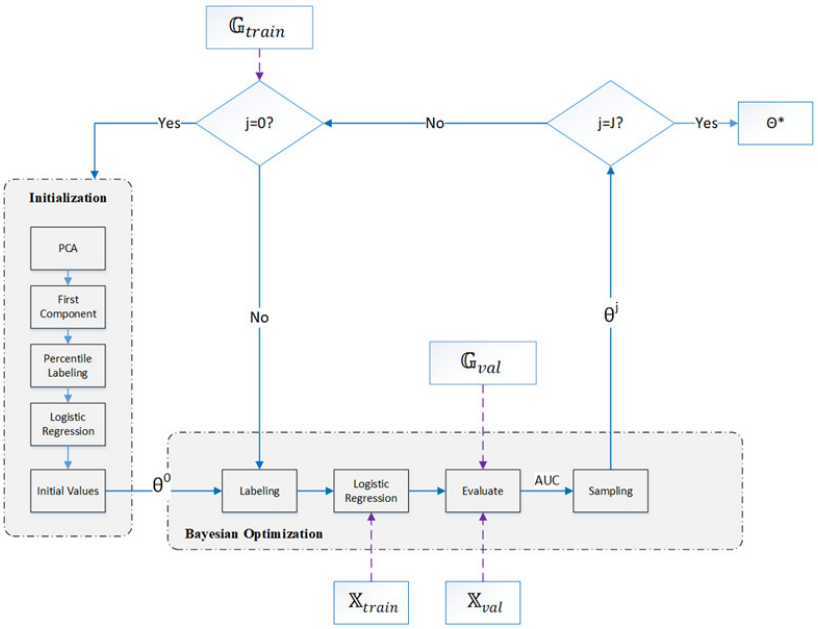

Figure 4.16 Flowchart of the OLBO algorithm.

Where $\mathcal{VS}$ is the variable space from which the sampling is drawn.

The most vital set of variables is the weight vector (W) and bias (b). The weight vector for iteration j of the algorithm can be written as:

$$W^{(j)} = [w_{FI_1}^{(j)}, w_{FI_3}^{(j)}, w_{PM_1}^{(j)}, w_{PM_3}^{(j)}, w_{PMM_1}^{(j)}, w_{PMM_3}^{(j)}, w_{RT_1}^{(j)}, w_{RT_3}^{(j)}]^\top \quad j = 0, 1, \ldots, J \tag{4.20}$$

Where J is the total number of iterations specified by the analyst. The values of these variables determine the z-value and hence the final labels for the observations, as described in detail in Supplementary Text: Labeling Procedure. In this study, the weights were sampled from a standard normal distribution as shown below:

$$\begin{gathered} w_k \sim N(0,1) \\ k \in \{FI_1, FI_3, PM_1, PM_3, PMM_1, PMM_3, RT_1, RT_3\} \end{gathered} \tag{4.21}$$

The bias term can be practically any real number. To restrict the feasible space, b is sampled from a uniform distribution between -10 and 10, shown as:

$$b \sim uniform(-10, 10) \tag{4.22}$$

To label the observations, two more variables are needed: a lower threshold, τ_1, and an upper threshold, τ_2, where:

$$\{\tau_1, \tau_2 \in (0,1) |\ \tau_1 < \tau_2\} \tag{4.23}$$

These two variables determine which participants should be Super-Agers and which should be Cognitive Decliners. The description of the mechanism can be found in Supplementary Text: Labeling Procedure. The two thresholds are sampled from a uniform distribution between 0 and 1, as shown below:

$$\begin{aligned} \tau_1 &\sim uniform(0,1) \\ \tau_2 &\sim uniform(0,1) \end{aligned} \tag{4.24}$$

In addition to the variables described above, we have some specific variables pertinent to the predictive model that we want to use for classification. In other words, we embedded the hyperparameter tuning procedure for the predictive model into the algorithm. Since we are using the logistic regression model as the classifier, we included the cost, c, and the ratio of L1- and L2-norms, ϕ, for the elastic regularization. We used log-uniform and uniform distributions for c and ϕ, respectively:

$$\begin{aligned} c &\sim loguniform(exp(-10), 100) \\ \phi &\sim uniform(0,1) \end{aligned} \tag{4.25}$$

4.8.5 Labeling Procedure

For a given matrix W and bias b, the algorithm calculates the linear combination of features weighted by W and adjusted by b:

$$z = GW + b \tag{4.26}$$

It then brings the z-values obtained into a scale between 0 and 1 using a sigmoid function:

$$\Pi(z) = \frac{1}{1 + exp(-z)} \tag{4.27}$$

To label the observations at iteration j, we use the following:

$$\mathcal{C}_i = \begin{cases} 1, & \text{if } \Pi(z_i) \geq \tau_2 \\ 0, & \text{if } \Pi(z_i) \leq \tau_1 \end{cases} \tag{4.28}$$

Where $\mathcal{C}_i$ is the class label for $x_i \in X$. Class label 0 was used for Cognitive Decliners and class label 1 was for Super-Agers. The initial values of the thresholds ($\tau_1^{(0)}$ and $\tau_2^{(0)}$) are set to 0.3 and 0.7, respectively. However, these variables are optimized throughout the optimization process later. The labels that are assigned at each iteration for training, validation, and testing sets are denoted as $y_{train}^{(j)}$, $y_{val}^{(j)}$, and $y_{test}^{(j)}$, respectively.

4.8.6 Initialization Process

OLBO requires initial values, $\Theta^{(0)}$ to start the procedure. Although arbitrary values can be used from the feasible domain, $\mathcal{VS}$, it is wiser to use an initialization procedure to make the model converge faster. In the first step, we transform the G using principal component analysis (PCA) and retain the first component with the largest variance explained, denoted as C_1. Then, we opt for the top and bottom 30th percentiles of C_1. Depending on the relation between the $\mathbb{G}$ and C_1, we label the selected groups as Super-Agers or Cognitive Decliners. The expected initial labeling procedure using the correlation coefficient sign of C_1 and any cognition tests at t_3 is summarized in the table below:

Table 4.3 Expected relation between the cognition tests and C_1

Correlation Coefficient Sign between C_1 and any of the variables below				Label Assignment	
FI_3	PM_3	PMM_3	RT_3	Bottom 30th	Top 30th
+	+	-	-	Cognitive decliners	Super-Agers
-	-	+	+	Super-Agers	Cognitive decliners

For instance, if we have a positive correlation between C_1 and FI_3, and between C_1 and PM_3, and a negative correlation between C_1 and PMM_3 and between C_1 and RT_3, then it suggests that the top 30th percentile should be labeled as Super-Agers and the bottom 30th percentile as Cognitive Decliners (first row of the table above). But if the correlation coefficient is negative for FI_3 and PM_3, and is positive for PMM_3 and RT_3 then we need to use the opposite labeling, meaning that the top 30th percentiles of C_1 are Cognitive Decliners, and the bottom 30th percentiles are Super-Agers. Now that candidate labeling $y_{train}^{(0)}$ is suggested, one can find the

corresponding values for W and b using a logistic regression model. To train the model, we use the $\mathbb{G}_{train}$ as predictors and the suggested labeling $y^{(0)}_{train}$ as the response variable. This procedure gives the initial values to start with, i.e., $W^{(0)}$ and $b^{(0)}$.

4.8.7 Constraints

Since the sampling for each variable is done independently of the other variables, we might end up with unacceptable sets of parameters in some trials. This occurs due to 1) combinations that are not feasible (e.g., when $\tau_1 > \tau_2$), or 2) when the resulting set of sampled values leads to a severe imbalance ratio in the class labels. To avoid such scenarios, we assigned a loss function value of 4, which is much worse than the loss function of a completely random model (i.e., $\mathcal{L} = 0.5$). This large penalty punishes the algorithm for sampling unacceptable sets of variables and helps the model to learn through iterations what samples are more likely to result in better loss function values. We set the maximum acceptable imbalance ratio to 0.7, meaning that if the majority class has more than 70% of the instances, the following labeling is not desirable.

4.8.8 Feature Importance Calculation

In this study, we used logistic regression as the predictive model in the OLBO algorithm. The choice of logistic regression model gives us the power of interpretability through the exploration of the coefficient values. For this purpose, we utilized a procedure introduced in [55]. let's find the odds ratio, i.e., the ratio of the probability of being a Super-Ager vs. Cognitive Decliner:

$$odds = \frac{p(y=1)}{p(y=0)} = \frac{p(y=1)}{1-p(y=1)} = exp\Big(\beta_0 + \beta_1 x_1 + \cdots + \beta_p x_p\Big) \tag{4.29}$$

Now, we can compare what will happen if we increase the value of one of the covariates x_j by 1 unit, using the following equation:

$$\frac{odds(x_j+1)}{odds(x_j)} = exp(\beta_j) \tag{4.30}$$

4.9 References

[1] Timothy A. Salthouse. When does age-related cognitive decline begin? *Neurobiology of aging*, 30:507–514, 2009.

[2] Archana Singh-Manoux, Mika Kivimaki, M. Maria Glymour, Alexis Elbaz, Claudine Berr, Klaus P. Ebmeier, Jane E. Ferrie, and Aline Dugravot. Timing of onset of cognitive decline: results from whitehall ii prospective cohort study. *BMJ (Clinical research ed.)*, 344, 1 2012.

[3] Cheryl L. Grady and Fergus Im Craik. Changes in memory processing with age. *Current opinion in neurobiology*, 10:224–231, 4 2000.

[4] Trey Hedden and John D.E. Gabrieli. Insights into the ageing mind: a view from cognitive neuroscience. *Nature reviews. Neuroscience*, 5:87–96, 2004.

[5] Salvador Algarabel, Joaquín Escudero, José Francisco Mazón, Alfonso Pitarque, Manuel Fuentes, Vicente Peset, and Laura Lacruz. Familiarity-based recognition in the young, healthy elderly, mild cognitive impaired and alzheimer's patients. *Neuropsychologia*, 47:2056–2064, 8 2009.

[6] Salvador Algarabel, Manuel Fuentes, Joaqun Escudero, Alfonso Pitarque, Vicente Peset, José Francisco Mazón, and Juan Carlos Meléndez. Recognition memory deficits in mild cognitive impairment. *Neuropsychology, development, and cognition. Section B, Aging, neuropsychology and cognition*, 19:608–619, 9 2012.

[7] Joshua D. Koen and Andrew P. Yonelinas. The effects of healthy aging, amnestic mild cognitive impairment, and alzheimer's disease on recollection and familiarity: a meta-analytic review. *Neuropsychology review*, 24:332–354, 9 2014.

[8] Tamar Gefen, Emily Shaw, Kristen Whitney, Adam Martersteck, John Stratton, Alfred Rademaker, Sandra Weintraub, M. Marsel Mesulam, and Emily Rogalski. Longitudinal neuropsychological performance of cognitive superagers. *Journal of the American Geriatrics Society*, 62:1598–1600, 2014.

[9] Tamar Gefen, Tamar Gefen, Melanie Peterson, Steven T. Papastefan, Adam Martersteck, Kristen Whitney, Alfred Rademaker, Alfred Rademaker, Eileen H. Bigio, Eileen H. Bigio, Sandra Weintraub, Sandra Weintraub, Emily Rogalski, M. Marsel Mesulam, M. Marsel Mesulam, and Changiz Geula. Morphometric and histologic substrates of cingulate integrity in elders with exceptional memory capacity. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 35:1781–1791, 1 2015.

[10] Theresa M. Harrison, Sandra Weintraub, M. Marsel Mesulam, and Emily Rogalski. Superior memory and higher cortical volumes in unusually successful cognitive aging. *Journal of the International Neuropsychological Society : JINS*, 18:1081–1085, 11 2012.

[11] Emily J. Rogalski, Tamar Gefen, Junzi Shi, Mehrnoosh Samimi, Eileen Bigio, Sandra Weintraub, Changiz Geula, and M. Marsel Mesulam. Youthful memory capacity in old brains: anatomic and genetic clues from the northwestern superaging project. *Journal of cognitive neuroscience*, 25:29–36, 2013.

[12] Parvin Mohammadiarvejeh, Brandon S. Klinedinst, Qian Wang, Tianqi Li, Brittany Larsen, Amy Pollpeter, Shannin N. Moody, Sara A. Willette, Jon P. Mochel, Karin Allenspach, Guiping Hu, and Auriel A. Willette. Bioenergetic and vascular predictors of potential super-ager and cognitive decline trajectories-a uk biobank random forest classification study. *GeroScience*, 2022.

[13] Jennifer Reichstadt, Colin A. Depp, Lawrence A. Palinkas, David P. Folsom, and Dilip V. Jeste. Building blocks of successful aging: a focus group study of older adults' perceived contributors to successful aging. *The American journal of geriatric psychiatry : official journal of the American Association for Geriatric Psychiatry*, 15:194–201, 2007.

[14] Christos Davatzikos. Machine learning in neuroimaging: Progress and challenges. *NeuroImage*, 197:652–656, 8 2019.

[15] Emily S. Finn, Xilin Shen, Dustin Scheinost, Monica D. Rosenberg, Jessica Huang, Marvin M. Chun, Xenophon Papademetris, and R. Todd Constable. Functional connectome fingerprinting: identifying individuals using patterns of brain connectivity. *Nature neuroscience*, 18:1664–1671, 11 2015.

[16] Xilin Shen, Emily S. Finn, Dustin Scheinost, Monica D. Rosenberg, Marvin M. Chun, Xenophon Papademetris, and R. Todd Constable. Using connectome-based predictive modeling to predict individual behavior from brain connectivity. *Nature protocols*, 12:506–518, 3 2017.

[17] Chang hyun Park, Bori R Kim, Hee Kyung Park, Soo Mee Lim, Eunhee Kim, Jee Hyang Jeong, and Geon Ha Kim. Predicting superagers by machine learning classification based on the functional brain connectome using resting-state functional magnetic resonance imaging. *Cerebral cortex (New York, N.Y. : 1991)*, 32, 9 2022.

[18] Randy L. Buckner, Fenna M. Krienen, and B. T.Thomas Yeo. Opportunities and limitations of intrinsic functional connectivity mri. *Nature neuroscience*, 16:832–837, 7 2013.

[19] Cathie Sudlow, John Gallacher, Naomi Allen, Valerie Beral, Paul Burton, John Danesh, Paul Downey, Paul Elliott, Jane Green, Martin Landray, Bette Liu, Paul Matthews, Giok Ong, Jill Pell, Alan Silman, Alan Young, Tim Sprosen, Tim Peakman, and Rory Collins. Uk biobank: an open access resource for identifying the causes of a wide range of complex diseases of middle and old age. *PLoS medicine*, 12, 3 2015.

[20] Biobank first public imaging release.

[21] Biobanknew data from brain imaging and on heart attacks and strokes available.

[22] Karla L. Miller, Fidel Alfaro-Almagro, Neal K. Bangerter, David L. Thomas, Essa Yacoub, Junqian Xu, Andreas J. Bartsch, Saad Jbabdi, Stamatios N. Sotiropoulos, Jesper L.R. Andersson, Ludovica Griffanti, Gwenaëlle Douaud, Thomas W. Okell, Peter Weale, Iulius Dragonu, Steve Garratt, Sarah Hudson, Rory Collins, Mark Jenkinson, Paul M. Matthews, and Stephen M. Smith. Multimodal population brain imaging in the uk biobank prospective epidemiological study. *Nature neuroscience*, 19:1523–1536, 10 2016.

[23] Nikos K. Logothetis. What we can do and what we cannot do with fmri. *Nature*, 453:869–878, 6 2008.

[24] Stephen M. Smith, Diego Vidaurre, Christian F. Beckmann, Matthew F. Glasser, Mark Jenkinson, Karla L. Miller, Thomas E. Nichols, Emma C. Robinson, Gholamreza Salimi-Khorshidi, Mark W. Woolrich, Deanna M. Barch, Kamil Uğurbil, and David C. Van Essen. Functional connectomics from resting-state fmri. *Trends in cognitive sciences*, 17:666–682, 12 2013.

[25] Tianqi Li, Colleen Pappas, Scott T. Le, Qian Wang, Brandon S. Klinedinst, Brittany A. Larsen, Amy Pollpeter, Ling Yi Lee, Mike W. Lutz, William K. Gottschalk, Russell H. Swerdlow, Kwangsik Nho, and Auriel A. Willette. Apoe, tomm40, and sex interactions on neural network connectivity. *Neurobiology of aging*, 109:158–165, 1 2022.

[26] Chloe Fawns-Ritchie and Ian J. Deary. Reliability and validity of the uk biobank cognitive tests. *PloS one*, 15, 4 2020.

[27] Matthew Evan Magnuson, Garth John Thompson, Hillary Schwarb, Wen Ju Pan, Andy McKinley, Eric H. Schumacher, and Shella Dawn Keilholz. Errors on interrupter tasks presented during spatial and verbal working memory performance are linearly linked to large-scale functional network connectivity in high temporal resolution resting state fmri. *Brain imaging and behavior*, 9:854–867, 12 2015.

[28] Kimberly M. Albert, Guy G. Potter, Brian D. Boyd, Hakmook Kang, and Warren D. Taylor. Brain network functional connectivity and cognitive performance in major depressive disorder. *Journal of psychiatric research*, 110:51–56, 3 2019.

[29] Matteo De Marco, Leandro Beltrachini, Alberto Biancardi, Alejandro F. Frangi, and Annalena Venneri. Machine-learning support to individual diagnosis of mild cognitive impairment using multimodal mri and cognitive assessments. *Alzheimer disease and associated disorders*, 31:278–286, 2017.

[30] Chandra Sripada, Saige Rutherford, Mike Angstadt, Wesley K. Thompson, Monica Luciana, Alexander Weigard, Luke H. Hyde, and Mary Heitzeg. Prediction of neurocognition in youth from resting state fmri. *Molecular psychiatry*, 25:3413–3421, 12 2020.

[31] Florian N. Götting, Viola Borchardt, Liliana R. Demenescu, Vanessa Teckentrup, Katharina Dinica, Anton R. Lord, Tim Rohe, Dorothea I. Hausdörfer, Meng Li, Coraline D. Metzger, and Martin Walter. Higher interference susceptibility in reaction time task is accompanied by weakened functional dissociation between salience and default mode network. *Neuroscience letters*, 649:34–40, 5 2017.

[32] Thorsten Bartsch and Christopher Butler. Transient amnesic syndromes. *Nature reviews. Neurology*, 9:86–97, 2 2013.

[33] Ronghao Yu, Bin Wang, Shumei Li, Junjing Wang, Feng Zhou, Shufang Chu, Xianyou He, Xue Wen, Xiaoxiao Ni, Liqing Liu, Qiuyou Xie, and Ruiwang Huang. Cognitive enhancement of healthy young adults with hyperbaric oxygen: A preliminary resting-state fmri study. *Clinical neurophysiology : official journal of the International Federation of Clinical Neurophysiology*, 126:2058–2067, 2015.

[34] Ralph M. Reitan and Deborah Wolfson. *The Halstead-Reitan Neuropsychological Test Battery: Theory and Interpretattion*. 1985.

[35] Aaron Smith. *Symbol Digit Modalities Test: Manual. Torrance*. 1973.

[36] David Wechsler. *WAIS-IV Administration and Scoring Manual*. 2010.

[37] Patrick Schober and Thomas R. Vetter. Linear regression in medical research. *Anesthesia and Analgesia*, 132:108–109, 1 2021.

[38] Thomas R. Vetter and Patrick Schober. Regression: The apple does not fall far from the tree. *Anesthesia and analgesia*, 127:277–283, 7 2018.

[39] David W. Hosmer, Stanley Lemeshow, and Rodney X. Sturdivant. Applied logistic regression: Third edition. *Applied Logistic Regression: Third Edition*, pages 1–510, 8 2013.

[40] Taghreed M. Jawa. Logistic regression analysis for studying the impact of home quarantine on psychological health during covid-19 in saudi arabia. *Alexandria Engineering Journal*, 61:7995–8005, 10 2022.

[41] Zakariya Yahya Algamal and Muhammad Hisyam Lee. Regularized logistic regression with adjusted adaptive elastic net for gene selection in high dimensional cancer classification. *Computers in biology and medicine*, 67:136–145, 12 2015.

[42] Martin Pelikan, David E Goldberg, and Erick Cantú-Paz. Boa: The bayesian optimization algorithm. pages 525–532. Morgan Kaufmann Publishers Inc., 1999.

[43] https://www.kaggle.com/S

[44] Martin Pelikan. Bayesian optimization algorithm. pages 31–48, 2 2005.

[45] Hakki Mert Torun, Madhavan Swaminathan, Anto Kavungal Davis, and Mohamed Lamine Faycal Bellaredj. A global bayesian optimization algorithm and its application to integrated system design. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 26:792–802, 4 2018.

[46] Peter I Frazier. A tutorial on bayesian optimization. 2018.

[47] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P. Adams, and Nando De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104:148–175, 1 2016.

[48] Joan Gonzalvez, Edmond Lezmi, Thierry Roncalli, and Jiali Xu. Financial applications of gaussian processes and bayesian optimization. *Bayesian Reasoning and Gaussian Processes for Machine Learning Applications*, pages 111–122, 3 2019.

[49] Ali Hebbal, Loïc Brevault, Mathieu Balesdent, El Ghazali Talbi, and Nouredine Melab. Bayesian optimization using deep gaussian processes with applications to aerospace system design. *Optimization and Engineering*, 22:321–361, 3 2021.

[50] Y. Morita, S. Rezaeiravesh, N. Tabatabaei, R. Vinuesa, K. Fukagata, and P. Schlatter. Applying bayesian optimization with gaussian process regression to computational fluid dynamics problems. *Journal of Computational Physics*, 449:110788, 1 2022.

[51] Yanan Sui, Vincent Zhuang, Joel W. Burdick, and Yisong Yue. Stagewise safe bayesian optimization with gaussian processes. *35th International Conference on Machine Learning, ICML 2018*, 11:7602–7613, 6 2018.

[52] James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. *Advances in Neural Information Processing Systems*, 24, 2011.

[53] Jasper Snoek, Hugo Larochelle, and Ryan P. Adams. Practical bayesian optimization of machine learning algorithms. *Advances in Neural Information Processing Systems*, 25, 2012.

[54] Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13:455–492, 1998.

[55] Christoph Molnar. *Interpretable Machine Learning*. 1985.

CHAPTER 5. GENERAL CONCLUSIONS

Machine learning (ML) is a multi-disciplinary field which provides analytical solutions to many problems in various domains. Healthcare is one of the areas that ML has been widely utilized to provide insight about underlying relationship between different variables or reveal hidden patterns. Even in a specific domain like healthcare, there exist so many application that ML can help.

In this dissertation, three challenging problems in healthcare are addressed and the appropriate solutions are proposed using ML and Optimization tools. The focus of each study is different: first study on HIV-1, second study on Severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2), and the last one on cognitive performance and Alzheimer's Disease (AD). In the following, the findings and results for each study and the contributions made by each one are summarized.

5.1 Study 1

The first study targets a critical component toward having personalized healthcare for HIV-1 patients. To be able to develop personalized therapeutics, predicting the future changes in the amino acid sequences of the envelope glycoproteins (Envs) of HIV-1 is vital. To have great results, understanding the relation between the current variability of an Env position with the neighboring position is important. Therefore, We hypothesized that, at any time point in a patient, positions with sequence variance are clustered on the three-dimensional structure of the Env. To verify this hypothesis, we examined whether variance at any position can be predicted using the variance of adjacent positions. We then, developed a dynamic ensemble selection technique to predict such variances on the Env.

We tested two Env domains targeted by therapeutics; the CD4-binding site and the high-mannose patch of Env. For both domains and in both clades B and C, the absence or presence of variance was predicted better using KBC than other classification models.

- Contributions in ML:
  - Developing a novel dynamic ensemble selection technique.
  - Introducing a new selection procedure for the best classifiers at each region.
- Contributions in the HIV field:
  - Predicting the amino acid variances of individual or group of positions using the variance of adjacent positions on the molecule.
  - Verifying the hypothesis that "positions with sequence variance are clustered on the three-dimensional structure of the Env"

5.2 Study 2

Between May and August 2020, 1,754 out of 8,013 (about 22%) Covid-19 cases visited emergency departments [1]. In such critical periods, the need for ICU increases significantly, and hospitals are near or at capacity. Therefore, it is critical to identify patients in need of ICU using the patient's information early in the initial hours of reception. This is essential for hospitals to prioritize individuals based on the likelihood of ICU admission and better utilize the resources to save lives.

In the second study, we found a solution for predicting the ICU need for COVID-19 patients, using only the data for the first few hours after admission. For this purpose, we proposed a stacking-based classification algorithm capable of identifying the ICU need for hospitalized COVID-19 patients using only the data from the first two hours of patients' admission. The proposed algorithm outperformed baseline models as well as those suggested in previous studies. In addition, the algorithm saved 10 hours of reaction time by reducing the 12 or 24 hours of time interval used in other studies down to only 2 hours.

- Contributions in ML:
  - Developing a novel stacking-based classification algorithm
  - Introducing a procedure to automatically generate new sets of informative features for the meta-learner that quantifies the performance of each base learner within a region
  - designing a procedure that enables the meta-learner to make a connection between the location of a data point and the performance of each base learner in that area through the regional and neighborhood assessment procedures.
- Contributions in the SARS-CoV-2 field:
  - Predicting ICU need more accurately than other models
  - Reducing reaction time from 24 and 12 hours to only 2 hours.

5.3 Study 3

Aging is frequently associated with a higher risk of dementia and progressive decline in cognitive performance. However, there exists a group of adults, called Super-Agers, with at least 80 years of age, who have a comparable cognitive function to young adults. The goal of the last study was to distinguish between Super-Agers and Cognitive Decliners using neural network data. Since the problem did not have any information about the class labels of participants, we had to assign the labels first. We then pushed the boundaries further by looking for the optimal labeling, where the objective function was defined based on the performance of a classifier. We proposed a hybrid algorithm, utilizing machine learning and optimization, as a solution for this problem. The proposed algorithm showed superior performance compared to all other baseline models.

- Contributions in ML:
  - Developing a hybrid algorithm to optimally assign class labels to instances using Bayesian optimization
  - Developing a novel labeling mechanism

  - Introducing a PCA-based labeling approach for the initial characterization of latent groups with different cognitive trajectories

- Contributions in the cognitive performance field:

  - Distinguishing between largely middle-aged participants who showed cognitive gains (i.e., "Super-Agers") versus cognitive decline
  - Identify the association between different neural network components and Super-Agers (or Cognitive Decliners)

5.4 References

[1] Sonu Subudhi, Ashish Verma, Ankit B. Patel, C. Corey Hardin, Melin J. Khandekar, Hang Lee, Dustin McEvoy, Triantafyllos Stylianopoulos, Lance L. Munn, Sayon Dutta, and Rakesh K. Jain. Comparing machine learning algorithms for predicting icu admission and mortality in covid-19. *npj Digital Medicine*, 4, 12 2021.