Acta Neurochirurgica Supplement 134

Victor E. Staartjes
Luca Regli
Carlo Serra *Editors*

Machine Learning in Clinical Neuroscience

Foundations and Applications

MOREMEDIA

Springer

Acta Neurochirurgica Supplement 134

ACTA NEUROCHIRURGICA's Supplement Volumes provide a unique opportunity to publish the content of special meetings in the form of a Proceedings Volume. Proceedings of international meetings concerning a special topic of interest to a large group of the neuroscience community are suitable for publication in ACTA NEUROCHIRURGICA. Links to ACTA NEUROCHIRURGICA's distribution network guarantee wide dissemination at a comparably low cost. The individual volumes should comprise between 120 and max. 250 printed pages, corresponding to 20-50 papers. It is recommended that you get in contact with us as early as possible during the preparatory stage of a meeting. Please supply a preliminary program for the planned meeting. The papers of the volumes represent original publications. They pass a peer review process and are listed in PubMed and other scientific databases. Publication can be effected within 6 months. Hans-Jakob Steiger is the Editor of ACTA NEUROCHIRURGICA's Supplement Volumes. Springer Verlag International is responsible for the technical aspects and calculation of the costs. If you decide to publish your proceedings in the Supplements of ACTA NEUROCHIRURGICA, you can expect the following: • An editing process with editors both from the neurosurgical community and professional language editing. After your book is accepted, you will be assigned a developmental editor who will work with you as well as with the entire editing group to bring your book to the highest quality possible. • Effective text and illustration layout for your book. • Worldwide distribution through Springer-Verlag International's distribution channels.

More information about this series at http://www.springer.com/series/4

Victor E. Staartjes • Luca Regli • Carlo Serra
Editors

Machine Learning in Clinical Neuroscience

Foundations and Applications

Springer

Editors
Victor E. Staartjes
Machine Intelligence in Clinical Neuroscience (MICN) Laboratory, Department of Neurosurgery
Clinical Neuroscience Center, University Hospital Zurich, University of Zurich
Zurich
Switzerland

Luca Regli
Machine Intelligence in Clinical Neuroscience (MICN) Laboratory, Department of Neurosurgery
Clinical Neuroscience Center, University Hospital Zurich, University of Zurich
Zurich
Switzerland

Carlo Serra
Machine Intelligence in Clinical Neuroscience (MICN) Laboratory, Department of Neurosurgery
Clinical Neuroscience Center, University Hospital Zurich, University of Zurich
Zurich
Switzerland

ISSN 0065-1419 ISSN 2197-8395 (electronic)
Acta Neurochirurgica Supplement 134
ISBN 978-3-030-85291-7 ISBN 978-3-030-85292-4 (eBook)
https://doi.org/10.1007/978-3-030-85292-4

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Contents

Machine Intelligence in Clinical Neuroscience: Taming the Unchained Prometheus

1

Victor E. Staartjes, Luca Regli, and Carlo Serra

1.1 Preface

The question is not whether intelligent machines can have any emotions, but whether machines can be intelligent without any emotions — Marvin Minsky [1]

Advances in both statistical modeling techniques as well as in computing power over the last few decades have enabled the rapid rise of the field of data science, including artificial intelligence (AI) and machine learning (ML) [2]. While AI can be defined as a goal—the goal to emulate human, "wide" intelligence with the ability to solve a range of different complex tasks with one brain or algorithm—ML deals with learning problems by inductively and iteratively learning from experience (in the form of data) without being explicitly programmed, as a form of "narrow" AI (focused on just one specific task). Along with the broader application of epidemiological principles and larger sample sizes ("big data"), this has led to broad adoption of statistical prediction modeling in clinical practice and research. Clinical prediction models integrate a range of input variables to predict a specific outcome in the future and can aid in evidence-based decision-making and improved patient counseling [3–6].

Even in the field of clinical neuroscience—including neurosurgery, neurology, and neuroradiology—ML has been increasingly applied over the years, as evidenced by the sharp rise in publications on machine learning in clinical neuroscience indexed in PubMed/MEDLINE since the 2000s (Fig. 1.1). While the history of ML applications to the field of neurosurgery is rather compressed into the past two decades, some early efforts have been made as early as the late 1980s. Disregarding other uses of AI and ML—such as advanced histopathological or radiological diagnostics—and focusing on predictive analytics, in 1989 Mathew et al. [7] published a report in which they applied a fuzzy logic classifier to 150 patients, and were able to predict whether disc prolapse or bony nerve entrapment were present based on clinical findings. In 1998, Grigsby et al. [8] were able to predict seizure freedom after anterior temporal lobectomy using neural networks based on EEG and MRI features, using data from 87 patients. Similarly, in 1999, Arle et al. [9] applied neural networks to 80 patients to predict seizures after epilepsy surgery. Soon, and especially since 2010, a multitude of publications followed, applying ML to clinical outcome prediction in all subspecialties of the neurosurgical field [10–11]. The desire to model reality to better understand it and in this way predict its future behavior has always been a goal of scientific thought, and "machine learning," if not just for the evocative power of the term, may appear under this aspect at first sight as a resolutive tool. However, if on one hand there cannot be any doubt that predicting the future will always remain a chimera, on the other hand it is true that machine learning tools can improve our possibilities to analyze and thus understand reality. But while clinical prediction modeling has certainly been by far the most common application of ML in clinical neuroscience, other applications such as, e.g. in image recognition [12, 13], natural language processing [14], radiomic feature extraction [15, 16], EEG classification [17], and continuous data monitoring [18] should not be disregarded and probably constitute the most interesting fields of application.

Today, ML and other statistical learning techniques have become so easily accessible to anyone with a computer and internet access, that it has become of paramount importance to ensure correct methodology. Moreover, there has been a major "hype" around the terms ML and AI in recent years. Because of their present-day low threshold accessibility, these techniques can easily be misused and misinterpreted, without intent to do so. For example, it is still common to see highly complex and "data-hungry" algorithms such as deep

V. E. Staartjes (✉) · L. Regli · C. Serra
Machine Intelligence in Clinical Neuroscience (MICN) Laboratory, Department of Neurosurgery, Clinical Neuroscience Center, University Hospital Zurich, University of Zurich, Zurich, Switzerland
e-mail: victoregon.staartjes@usz.ch; https://micnlab.com/

V. E. Staartjes et al. (eds.), *Machine Learning in Clinical Neuroscience*, Acta Neurochirurgica Supplement 134,
https://doi.org/10.1007/978-3-030-85292-4_1

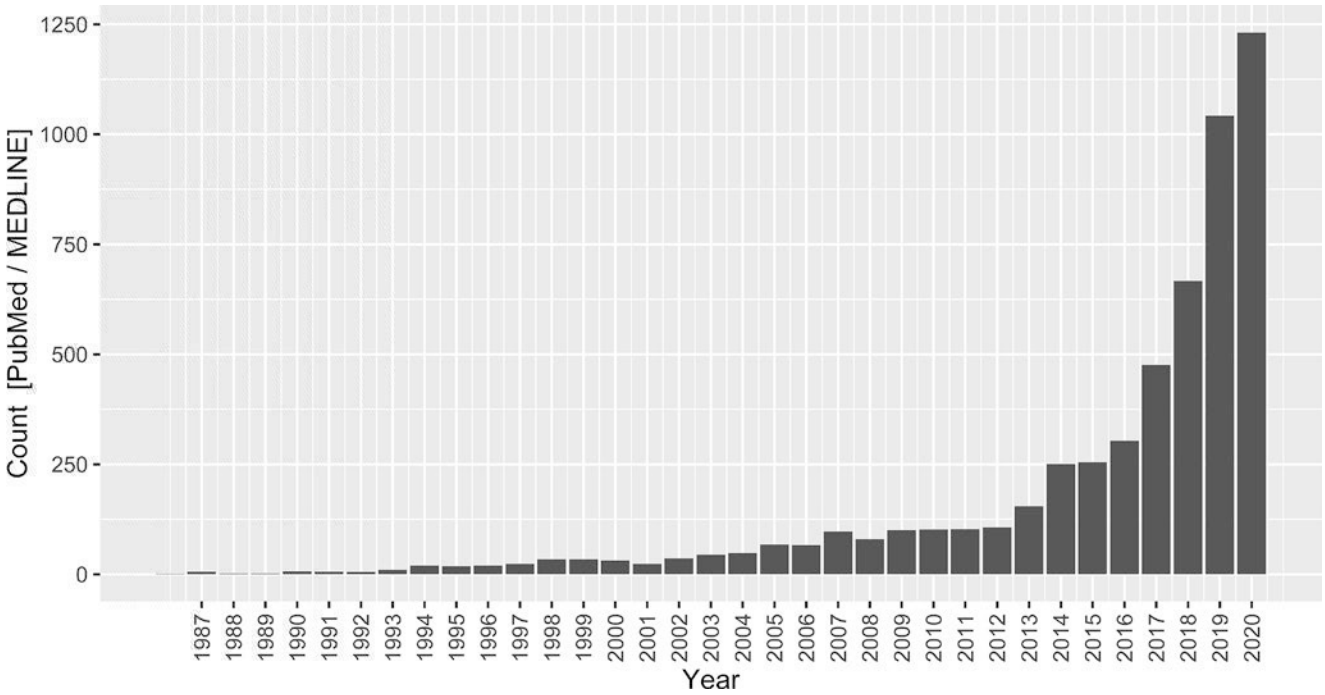

Fig. 1.1 Development of publication counts on machine learning in neurosurgery over the years. Counts were arrived at by searching "(*neurosurgery OR neurology OR neuroradiology) AND (machine learning OR artificial intelligence*)" on PubMed/MEDLINE

neural networks applied to very small datasets, to see overtly overfitted or imbalanced models, or models trained for prediction that are then used to "identify risk factors" (prediction vs. explanation/inference). Especially in clinical practice and in the medico-legal arena, it is vital that clinical prediction models intended to be implemented into clinical practice are developed with methodological rigor, and that they are well-validated and generalizable.

Some efforts such as, e.g. the EQUATOR Network's "Transparent reporting of a multivariable prediction model for individual prognosis or diagnosis" (TRIPOD) statement [19] have led to improved methodological quality and standardized reporting throughout the last 5 years [20]. Still, it is a fact that ML is often poorly understood, and that the methodology has to be well-appreciated to prevent publishing flawed models—standardized reporting is valuable, but not enough. Open-source ML libraries like Keras [21] and Caret [22] have truly democratized ML. While this fact—combined with the steadily increasing availability of large amounts of structured and unstructured data in the "big data" era—has certainly provided leverage to the whole field of ML, putting so much analytical power into anyone's hands without clear methodological foundations can be risky.

The immense technological progress during the past century has certainly sparked reflection on the responsibilities of humanity regarding limitations, safe use, fair distribution, and consequences of these advances.

Progress can be risky. As a matter of fact, ML tools are increasingly being used to aid in decision-making in several domains of human society. The lack of profound understanding of the capabilities and, most importantly, of the limitations of ML may lead to the erroneous assumption that ML may overtake, and not just aid, the decision-making capacity of the human mind. Needless to say, this attitude can have serious practical and most importantly ethical consequences. Today's greater power of humanity in controlling nature means that we must also realize their limitations and potential dangers, and to consequently limit our applications of those technologies to avoid potential disaster—this has become the most popular topic of modern philosophy on artificial intelligence [23].

Today, every scientific study is subject to ethical review and approval, but potential long-term sequelae of ML studies are seldomly considered. ML in medicine has great potential, but both doctors applying these technologies in clinical practice as well as those researchers developing tools based on these technologies must be acutely aware of their limitations and their ramifications. Further unsolved ethical issues regarding the use of ML and AI in clinical medicine pertain to protecting *data integrity*, ensuring *justice* in the distribution of ML-based resources, and maintaining *accountability*—Could algorithms learn to assign values and become independent moral agents? While some progress has been made in protecting data integrity, such as the use of *federated*

learning, developments in other ethical issues remain less predictable [24].

Therefore, ML and AI must remain tools adjunctive to our own mind, tools that we should be able to master, control, and apply to our advantage—and that should not take over our minds. For example, it is inconceivable and even potentially dangerous to fully rely on predictions made by a ML algorithm in clinical practice, currently. The future cannot be easily predicted by machines, or by anyone for that matter—And even if near-perfect predictions were theoretically possible, our intuition would tell us that the mere knowledge of what is very likely going to happen in the future may lead us to change events, not dissimilar to *Heisenberg's uncertainty principle*. While our mind can recognize, abstract, and deal with the many uncertainties in clinical practice, algorithms cannot. Among many others, concepts such as *Turing's Test* [25] underline the importance of appreciating the limits of ML and AI: They are no alchemy, no magic. They do not make the impossible possible. They merely serve to assist and improve our performance on certain very specific tasks.

For these reasons, we embarked on a journey to compile a textbook for clinicians that demystifies the terms "machine learning" and "artificial intelligence" by illustrating their methodological foundations, as well as some specific applications throughout the different fields of clinical neuroscience, and its limitations. Of note, this book has been inspired and conceived by the group of machine learning specialists that also contributed to the *1st Zurich Machine Intelligence in Clinical Neuroscience Symposium* that took place on January 21st 2021 with presentations on their respective book chapters which we encourage readers to consider watching (the recorded contributions are available on: www.micnlab.com/symposium2021).

The book is structured in five major parts:

1. The first part deals with the methodological foundations of clinical prediction modeling as the most common clinical application of ML [4]. The basic workflow for developing and validating a clinical prediction model is discussed in detail in a five-part series, which is followed by spotlights on certain topics of relevance ranging from feature selection, dimensionality reduction techniques as well as Bayesian, deep learning, and clustering techniques, to how to deploy, update, and interpret clinical prediction models.
2. Part II consists of a brief *tour de force* through the domain of ML in neuroimaging and its foundational methods. First, the different applications and algorithms are laid out in detail, which is then followed by specific workflows including radiomic feature extraction, segmentation, and brain imaging classification.
3. The next part provides a glimpse into the world of natural language processing (NLP) and time series analysis (TSA), going through the algorithms used for such analyses, as well as workflows for both domains.
4. The fourth part of this book handles the various ethical implications of applying ML in clinical practice—From general ethical considerations on AI, to ways in which ML can assist doctors in daily practice and the limitations of predictive analytics. In addition, a brief history of ML in neurosurgery is provided, too.
5. The fifth and final part is targeted to demonstrating an overview over the various clinical applications that have already been implemented in clinical neuroscience, covering neuroimaging, neurosurgery, neurology, and ophthalmology.

Our hope is that this book may inspire and instruct a generation of physician-scientists to continue to grow and develop the seeds that have been planted for machine intelligence in clinical neuroscience, and to discover the limits of the clinical applications therein.

References

1. Minsky M. The Society of Mind. Simon and Schuster. 1986.
2. Kononenko I. Machine learning for medical diagnosis: history, state of the art and perspective. Artif Intell Med. 2001;23:89. https://doi.org/10.1016/S0933-3657(01)00077-X.
3. Senders JT, Staples PC, Karhade AV, Zaki MM, Gormley WB, Broekman MLD, Smith TR, Arnaout O. Machine learning and neurosurgical outcome prediction: a systematic review. World Neurosurg. 2018;109:476. https://doi.org/10.1016/j.wneu.2017.09.149.
4. Staartjes VE, Stumpo V, Kernbach JM, et al. Machine learning in neurosurgery: a global survey. Acta Neurochir. 2020;162(12):3081–91.
5. Saposnik G, Cote R, Mamdani M, Raptis S, Thorpe KE, Fang J, Redelmeier DA, Goldstein LB. JURaSSiC: accuracy of clinician vs risk score prediction of ischemic stroke outcomes. Neurology. 2013;81:448. https://doi.org/10.1212/WNL.0b013e31829d874e.
6. Steyerberg EW. Clinical prediction models: a practical approach to development, validation, and updating. New York, NY: Springer Science & Business Media; 2008.
7. Mathew B, Norris D, Mackintosh I, Waddell G. Artificial intelligence in the prediction of operative findings in low back surgery. Br J Neurosurg. 1989;3:161. https://doi.org/10.3109/02688698909002791.
8. Grigsby J, Kramer RE, Schneiders JL, Gates JR, Smith WB. Predicting outcome of anterior temporal lobectomy using simulated neural networks. Epilepsia. 1998;39:61. https://doi.org/10.1111/j.1528-1157.1998.tb01275.x.
9. Arle JE, Perrine K, Devinsky O, Doyle WK. Neural network analysis of preoperative variables and outcome in epilepsy surgery. J Neurosurg. 1999;90:998. https://doi.org/10.3171/jns.1999.90.6.0998.
10. Azimi P, Mohammadi HR, Benzel EC, Shahzadi S, Azhari S, Montazeri A. Artificial neural networks in neurosurgery. J Neurol

Neurosurg Psychiatry. 2015;86:251. https://doi.org/10.1136/jnnp-2014-307807.
11. Senders JT, Zaki MM, Karhade AV, Chang B, Gormley WB, Broekman ML, Smith TR, Arnaout O. An introduction and overview of machine learning in neurosurgical care. Acta Neurochir. 2018;160:29. https://doi.org/10.1007/s00701-017-3385-8.
12. Swinburne NC, Schefflein J, Sakai Y, Oermann EK, Titano JJ, Chen I, Tadayon S, Aggarwal A, Doshi A, Nael K. Machine learning for semi-automated classification of glioblastoma, brain metastasis and central nervous system lymphoma using magnetic resonance advanced imaging. Ann Transl Med. 2019;7(11):232.
13. Titano JJ, Badgeley M, Schefflein J, et al. Automated deep-neural-network surveillance of cranial images for acute neurologic events. Nat Med. 2018;24(9):1337–41.
14. Senders JT, Karhade AV, Cote DJ, et al. Natural language processing for automated quantification of brain metastases reported in free-text radiology reports. JCO Clin Cancer Inform. 2019;3:1–9.
15. Chang K, Bai HX, Zhou H, et al. Residual convolutional neural network for the determination of IDH status in low- and high-grade gliomas from MR imaging. Clin Cancer Res. 2018;24(5):1073–81.
16. Kernbach JM, Yeo BTT, Smallwood J, et al. Subspecialization within default mode nodes characterized in 10,000 UK Biobank participants. Proc Natl Acad Sci U S A. 2018;115(48):12295–300.
17. Varatharajah Y, Berry B, Cimbalnik J, Kremen V, Van Gompel J, Stead M, Brinkmann B, Iyer R, Worrell G. Integrating artificial intelligence with real-time intracranial EEG monitoring to automate interictal identification of seizure onset zones in focal epilepsy. J Neural Eng. 2018;15(4):046035.
18. Schwab P, Keller E, Muroi C, Mack DJ, Strässle C, Karlen W. Not to cry wolf: distantly supervised multitask learning in critical care. ArXiv. 2018:1802.05027. [cs, stat].
19. Collins GS, Reitsma JB, Altman DG, Moons KGM. Transparent reporting of a multivariable prediction model for individual prognosis or diagnosis (TRIPOD): the TRIPOD statement. BMJ. 2015;350:g7594.
20. Zamanipoor Najafabadi AH, Ramspek CL, Dekker FW, Heus P, Hooft L, Moons KGM, Peul WC, Collins GS, Steyerberg EW, van Diepen M. TRIPOD statement: a preliminary pre-post analysis of reporting and methods of prediction models. BMJ Open. 2020;10(9):e041537.
21. Chollet F. Keras: deep learning library for Theano and TensorFlow. 2015. https://keras.io/k.
22. Kuhn M, Wing J, Weston S, Williams A, et al. caret: classification and regression training. 2019.
23. Jonas H. Das Prinzip Verantwortung: Versuch einer Ethik für die technologische Zivilisation. Berlin: Suhrkamp; 2003.
24. Yang Q, Liu Y, Chen T, Tong Y. Federated machine learning: concept and applications. ArXiv. 2019:1902.04885. [cs].
25. Oppy G, Dowe D. The turing test. Stanford, CA: The Stanford Encyclopedia of Philosophy; 2020.

Part I

Clinical Prediction Modeling

2 Foundations of Machine Learning-Based Clinical Prediction Modeling: Part I—Introduction and General Principles

Julius M. Kernbach and Victor E. Staartjes

2.1 Introduction

Although there are many applications of machine learning (ML) in clinical neuroscience, including but not limited to applications in neuroimaging and natural language processing, classical predictive analytics still form the majority of the body of evidence that has been published on the topic.

When reviewing or working on research involving ML-based predictive analytics—which is becoming increasingly common—it is important to do so with a strong methodological basis. Especially considering the "democratization" of ML methods through libraries and the increasing computing power, as well as the exponentially increasing influx of ML publications in the clinical neurosciences, methodological rigor has become a major issue. This chapter and in fact the entire five-part series (cite Chaps. 3–6) is intended to convey that basic conceptual and programming knowledge to tackle ML tasks with some basic prerequisite R knowledge, with a particular focus on predictive analytics.

At this point, it is important to stress that the concepts and methods presented herein are intended as an entry-level guide to ML for clinical outcome prediction, presenting one of many valid approaches to clinical prediction modeling, and thus does not encompass all the details and intricacies of the field. Further reading is recommended, including but not limited to Max Kuhn's "Applied Predictive Modeling" [1] and Ewout W. Steyerberg's "Clinical Prediction Models" [2].

This first part focuses on defining the terms ML and AI in the context of predictive analytics, and clearly describing their applications in clinical medicine. In addition, some of the basic concepts of machine intelligence are discussed and explained. Part II goes into detail about common problems when developing clinical prediction models: What overfitting is and how to avoid it to arrive at generalizable models, how to select which input features are to be included in the final model (feature selection) or how to simplify highly dimensional data (feature reduction). We also discuss how data splits and resampling methods like cross-validation and the bootstrap can be applied to validate models before clinical use. Part III touches on several topics including how to prepare your data correctly (standardization, one-hot encoding) and evaluate models in terms of discrimination and calibration, and points out some recalibration methods. Some other points of significance and caveats that the reader may encounter while developing a clinical prediction model are discussed: sample size, class imbalance, missing data and how to impute it, extrapolation, as well as how to choose a cutoff for binary classification. Parts IV and V present a practical approach to classification and regression problems, respectively. They contain detailed instructions along with a downloadable code for the R statistical programming language, as well as a simulated database of Glioblastoma patients that allows the reader to code in parallel to the explanations. This section is intended as a scaffold upon which readers can build their own clinical prediction models, and that can easily be modified. Furthermore, we will not in detail explain the workings of specific ML algorithms such as generalized linear models, support vector machines, neural networks, or stochastic gradient boosting. While it is certainly important to have a basic understanding of the specific

J. M. Kernbach and V. E. Staartjes have contributed equally to this series, and share first authorship.

J. M. Kernbach
Neurosurgical Artificial Intelligence Laboratory Aachen (NAILA), Department of Neurosurgery, RWTH Aachen University Hospital, Aachen, Germany

V. E. Staartjes (✉)
Machine Intelligence in Clinical Neuroscience (MICN) Laboratory, Department of Neurosurgery, Clinical Neuroscience Center, University Hospital Zurich, University of Zurich, Zurich, Switzerland
e-mail: victoregon.staartjes@usz.ch; https://micnlab.com/

V. E. Staartjes et al. (eds.), *Machine Learning in Clinical Neuroscience*, Acta Neurochirurgica Supplement 134,
https://doi.org/10.1007/978-3-030-85292-4_2

algorithms one applies, these details can be looked up online [3] and detailed explanations of these algorithms would go beyond the scope of this guide. The goal is instead to convey the basic concepts of ML-based predictive modeling, and how to practically implement these.

2.2 Machine Learning: Definitions

As a field of study, ML in medicine is positioned between statistical learning and advanced computer science, and typically evolves around *learning problems*, which can be conceptually defined as optimizing a performance measure on a given task by learning through training experience on prior data. A ML algorithm inductively learns to automatically extract patterns from data to generate insights [4, 5] without being explicitly programmed. This makes ML an attractive option to predict even complex phenomena without prespecifying an a priori theoretical model. ML can be used to leverage the full granularity of the data richness enclosed in the *Big Data* trend. Both the complexity and dimensionality of modern medical data sets are constantly increasing and nowadays comprise many variables per observation, much so that we speak of "wide data" with generally more variables (in ML lingo called *features*) than observations (samples) [6, 7]. This has given rise to the so-called *omics* sciences including radiomics and genomics [8–10]. The sheer complexity and volume of data ranging from hundreds to thousands of variables at times exceeds human comprehension, but combined with increased computational power enables the full potential of ML [3, 11].

With the exponential demand of AI and ML in modern medicine, a lot of confusion was introduced regarding the separation of these two terms. AI and ML are frequently used interchangeably. We define ML as subset of AI—to quote Tom Mitchell—ML "is the study of computer algorithms that allow computer programs to automatically improve through experience" [12], involving the concept of "learning" discussed earlier. In contrast, AI is philosophically much vaster, and can be defined as an ambition to enable computer programs to behave in a human-like nature. That is, showing a certain human-like intelligence. In ML, we learn and optimize an algorithm from data for maximum performance on a certain learning task. In AI, we try to emulate natural intelligence, to not only learn but also apply the gained knowledge to make elaborate decisions and solve complex problems. In a way, ML can thus be considered a technique towards realizing (narrow) AI. Ethical considerations on the "AI doctor" are far-reaching [13, 14], while the concept of a clinician aided by ML-based tools is well accepted.

The most widely used ML methods are either supervised or unsupervised learning methods, with the exceptions of semi-supervised methods and reinforcement learning [6, 15]. In supervised learning, a set of input variables are used as training set, e.g. different meaningful variables such as age, gender, tumor grading, or functional neurological status to predict a known target variable ("label"), e.g. overall survival. The ML method can then learn the pattern linking input features to target variable, and based on that enable the prediction of new data points—hence, *generalize* patterns beyond the present data. We can train a ML model for survival prediction based on a retrospective cohort of brain tumor patients, since we know the individual length of survival for each patient of the cohort. Therefore, the target variable is *labeled*, and the machine learning-paradigm *supervised*. Again, the actually chosen methods can vary: Common models include support vector machines (SVMs), as example of a *parametric* approach, or *the k*-nearest neighbor (KNN) algorithm as a *non-parametric* method [16]. On the other hand, in *unsupervised* learning, we generally deal with *unlabeled* data with the assumption of the structural coherence. This can be leveraged in clustering, which is a subset of unsupervised learning encompassing many different methods, e.g. hierarchical clustering or *k*-means clustering [4, 17]. The observed data is partitioned into clusters based on a measure of similarity regarding the structural architecture of the data. Similarly, dimensionality reduction methods—including principal component analysis (PCA) or autoencoders—can be applied to derive a low-dimensional representation explicitly from the present data [4, 18].

A multitude of diverse ML algorithms exist, and sometimes choosing the "right" algorithm for a given application can be quite confusing. Moreover, based on the so-called *no free lunch theorem* [19] no single statistical algorithm or model can generally be considered superior for all circumstances. Nevertheless, ML algorithms can vary greatly based on the (a) representation of the candidate algorithm, (b) the selected performance metric, and (c) the applied optimization strategy [4, 5, 20]. Representation refers to the learner's hypothesis space of how they formally deal with the problem at hand. This includes but is not limited to instance-based learners, such as KNN, which instead of performing explicit generalization compares new observations with similar instances observed during training [21]. Other representation spaces include hyperplane-based models, such as logistic regression or naïve Bayes, as well as rule-based learners, decision trees or complex neural networks, all of which are frequently leveraged in various ML problems across the neurosurgical literature [22, 23]. The evaluated performance metrics can vary greatly, too. Performance evaluation and reporting play a pivotal role in predictive analytics (c.f. cite Chap. 4). Lastly, the applied ML algorithm is *optimized* by a so-called objective function such as greedy search or unconstrained continuous optimization options, including different choices of gradient descent [24, 25]. Gradient descent repre-

sents the most common optimization strategy for neural networks and can take different forms, e.g. batch- ("vanilla"), stochastic- or mini-batch gradient descent [25]. We delve deeper into optimization to illustrate how it is used in learning.

2.3 Optimization: The Central Dogma of Learning Techniques

At the heart of nearly all ML and statistical modeling techniques used in data science lies the concept of *optimization*. Even though optimization is the backbone of algorithms ranging from linear and logistic regression to neural networks, it is not often stressed in the non-academic data science space. Optimization describes the process of iteratively adjusting parameters to improve performance. Every optimization problem can be decomposed into three basic elements: First, every algorithm has *parameters* (sometimes called *weights*) that govern how the values of the input variables lead to a prediction. In linear and logistic regression, for example, these parameters include the coefficients that are multiplied with the input variable values, as well as the intercept. Second, there may be realistic *constraints* within which the parameters, or their combinations, must fall. While simple models such as linear and logistic regression often do not have such constraints, other ML algorithms such as support vector machines or *k*-means clustering do. Lastly and importantly, the optimization process is steered by evaluating a so-called *objective function* that assesses how well the current iteration of the algorithm is performing. Commonly, these objective functions are *error* (also called *loss*) functions, describing the deviation of the predicted values from the true values that are to be predicted. Thus, these error functions must be *minimized*. Sometimes, you may choose to use indicators of performance, such as accuracy, which conversely need to be *maximized* throughout the optimization process.

The optimization process starts by randomly *initializing* all model parameters—that is, assigning some initial value for each parameter. Then, predictions are made on the training data, and the error is calculated. Subsequently, the parameters are adjusted in a certain direction, and the error function is evaluated again. If the error increases, it is likely that the direction of adjustment of the parameters was awry and thus led to a higher error on the training data. In that case, the parameter values are adjusted in different directions, and the error function is evaluated again. Should the error decrease, the parameter values will be further modified in these specific directions, until a *minimum* of the error function is reached. The goal of the optimization process is to reach the *global minimum* of the error function, that is, the lowest error that can be achieved through the combination of parameter values within their constraints. However, the optimization algorithm must avoid getting stuck at *local minima* of the error function (see Fig. 2.1).

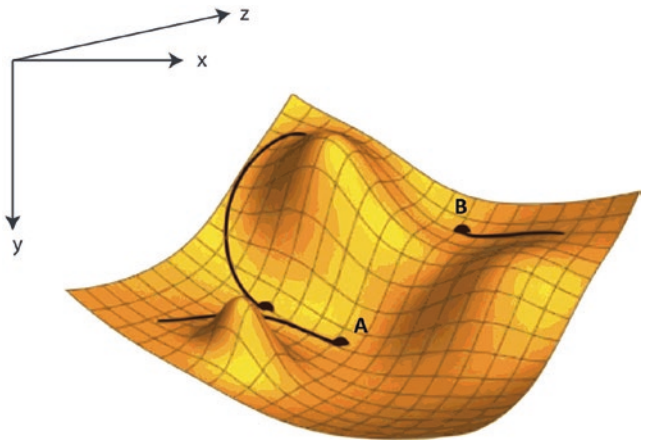

Fig. 2.1 Illustration of an optimization problem. In the *x* and *z* dimension, two parameters can take different values. In the *y* dimension, the error is displayed for different values of these two parameters. The goal of the optimization algorithm is to reach the *global minimum* (**A**) of the error through adjusting the parameter values, without getting stuck at a *local minimum* (**B**). In this example, three models are initialized with different parameter values. Two of the models converge at the global minimum (**A**), while one model gets stuck at a local minimum (**B**). Illustration by Jacopo Bertolotti. (This illustration has been made available under the Creative Commons CC0 1.0 Universal Public Domain Dedication)

The way in which the parameters are adjusted after each iteration is governed by an *optimization algorithm*, and approaches can differ greatly. For example, linear regression usually uses the ordinary least square (OLS) optimization method. In OLS, the parameters are estimated by solving an equation for the minimum of the sum of the square errors. On the other hand, *stochastic gradient descent*—which is a common optimization method for many ML algorithms—iteratively adjusts parameters as described above and as illustrated in Fig. 2.1. In stochastic gradient descent, the amount by which the parameters are changed after each iteration (also called *epoch*) is controlled by the calculated derivative (i.e. the slope or *gradient*) for each parameter with respect to the error function, and the *learning rate*. In many models, the learning rate is an important hyperparameter to set, as it controls how much parameters change in each iteration.

On the one hand, small learning rates can take many iterations to converge and make getting stuck at a local minimum more likely—on the other hand, a large learning rate can overshoot the global minimum. As a detailed discussion of the mathematical nature behind different algorithms remains beyond the scope of this introductory series, we refer to popular standard literature such as "Elements of Statistical Learning" by Hastie and Tibshirani [4], "Deep Learning" by Goodfellow et al. [26], and "Optimization for Machine Learning" by Sra et al. [27].

2.4 Explanatory Modeling Versus Predictive Modeling

The "booming" of applied ML has generated a methodological shift from *classical statistics* (experimental setting, hypothesis testing, group comparison, inference) to data-driven *statistical learning* (empirical setting, algorithmic modeling comprising ML, AI, pattern recognition) [28]. Unfortunately, the two statistical cultures have developed separately over the past decades [29] leading to incongruent evolved terminology and misunderstandings in the absence of an agreed-upon technical theorem (Table 2.1). This already becomes evident in the basic terminology describing model inputs and outputs: *predictors* or *independent variables* refer to model inputs in classical statistics, while *features* are the commonly used term in ML; outputs, known as *dependent variable* or *response*, are often labeled *target variable* or *label* in ML instead [30]. The duality of language has led to misconceptions regarding the fundamental difference between inference and prediction, as the term *prediction* has frequently been used incompatibly as in-sample correlation instead of out-of-sample generalization [31, 32]. The variation of one variable with a subsequent correlated variable later in time, such as the outcome, in the same group (in-sample correlation) does not imply prediction, and failure to account for this distinction can lead to false clinical decision-making [33, 34]. Strong associations between variables and outcome in a clinical study remain averaged estimates of the evaluated patient cohort, which does not necessarily enable predictions in unseen new patients. To shield clinicians from making wrong interpretations, we clarify the difference between explanatory modeling and predictive modeling, and highlight the potential of ML for strong predictive models.

Table 2.1 A comparison of central concepts in classical/inferential statistics versus in statistical/machine learning

Classical/inferential statistics	Statistical/machine learning
Explanatory modeling	**Predictive modeling**
An a priori chosen theoretical model is applied to data in order to test for causal hypotheses.	The process of applying a statistical model or data mining algorithm to data for the purpose of predicting new or future observations.
Focus on in-sample estimates	**Focus on out-of-sample estimates**
Goal: to confirm the existence of an effect in the entire data sample. Often using significance testing.	Goal: Use the best performing model to make new prediction for single new observations. Often using resampling techniques.
Focus on model interpretability	**Focus on model performance**
The model is chosen a priori, while models with intrinsic means of interpretability are preferred, e.g. a GLM, often parametric with a few fixed parameters.	Different models are applied and the best performing one is selected. Models tend to be more flexible and expressive, often non-parametric with many parameters adapting to the present data.
Experimental data	**Empirical data**
Long data (n samples > p variables)	**Wide data (n samples $\ll p$ variables)**
Independent variables	**Features**
Dependent variable	**Target variable**
Learn deductively by model testing	**Learn a model from data inductively**

Knowledge generation in clinical research has nearly exclusively been dominated by classical statistics with the focus on *explanatory modeling* (EM) [32]. In carefully designed experiments or clinical studies, a constructed theoretical model, e.g. a regression model, is applied to data in order to test for causal hypotheses. Based on theory, a model is chosen a priori, combining a fixed number of experimental variables, which are under the control of the investigator. Explicit model assumptions such as the Gaussian distribution assumption are made, and the model, which is believed to represent the true *data generating process*, is evaluated for the entire present data sample based on hypothesis and significance testing ("inference"). In such association-based modeling, a set of independent variables (X) are assumed to behave according to a certain mechanism ("theory") and ultimately cause an effect measured by the dependent variable (Y). Indeed, the role of *theory* in explanatory modeling is strong and is always reflected in the applied model, with the aim to obtain the most accurate representation of the underlying theory (technically speaking, classical statistics seeks to minimize *bias*). Whether *theory* holds true and the effect actually exists is then confirmed in the data, hence the overall analytical goal is *inference*.

Machine learning-based *predictive modeling* (PM) is defined as the process of applying a statistical model or data mining algorithm to data for the purpose of predicting future observations. In a heuristic approach, ML or PM is applied to *empirical data* as opposed to experimentally controlled data.

As the name implies, the primary focus lays on optimizing the prediction of a target variable (Y) for new observations given their set of features (X). As opposed to explanatory modeling, PM is *forward looking* [32] with the intention of predicting new observations, and hence *generalization beyond the present data* is the fundamental goal of the analysis. In contrast to EM, PM seeks to minimize both *variance* and *bias* [35, 36], occasionally sacrificing the theoretical interpretability for enhanced predictive power. Any underlying method can constitute a predictive model ranging from parametric and rigid models to highly flexible non-parametric and complex models. With a minimum of a priori specifications, a model is then heuristically derived from the data [37, 38]. The true data generating process lays in the data, and is inductively learned and approximated by ML models.

2.5 Workflow for Predictive Modeling

In clinical predictive analytics, *generalization* is our ultimate goal. To answer different research objectives, we develop, test, and evaluate different models for the purpose of clinical application (for an overview see https://topepo.github.io/caret/available-models.html). Many research objectives in PM can be framed either as the prediction of a continuous endpoint (regression) such as progression-free survival measured in months or alternatively as the prediction of a binary endpoint (classification), e.g. survival after 12 months as a dichotomized binary. Most continuous variables can easily be reduced and dichotomized into binary variables, but as a result data granularity is lost. Both regression and classification share a common analytical workflow with difference in regard to model evaluation and reporting (c.f. *cite* Chap. 5 *Classification problems* and *cite* Chap. 6 *Regression problems* for a detailed discussion). An adaptable pipeline for both regression and classification problems is demonstrated in Parts IV and V. Both sections contain detailed instructions along with a simulated dataset of 10,000 patients with glioblastoma and the code based on the statistical programming language R, which is available as open-source software.

For a general overview, a four-step approach to PM is proposed (Fig. 2.2): First and most important (1) all data needs to be pre-processed. ML is often thought of as *letting data do the heavy lifting*, which in part is correct, however, the raw data is often not suited to learning well in its current form. A lot of work needs to be allocated to preparing the input data including data cleaning and pre-processing (imputation, scaling, normalization, encoding) as well as *feature engineering* and *selection*. This is followed by using (2) resampling techniques such as *k*-fold cross-validation (c.f. cite Chap. 3 *generalization and overfitting*) to train different models and perform hyperparameter tuning. In a third step (3), the different models are compared and evaluated for generalizability based on a chosen out-of-sample performance measure in an independent testing set. The best performing model is ultimately selected, the model's out-of-sample calibration assessed (c.f. cite Chap. 4 *Evaluation and points of significance*), and, in a fourth step (4) the model is externally validated—or at least prospectively internally validated—to ensure clinical usage is safe and generalizable across locations, different populations and end users (c.f. *cite* Chap. 3 *Generalization and overfitting*). The European Union (EU) and the Food and Drug Administration (FDA) have both set standards for classifying machine learning and other software for use in healthcare, upon which the extensiveness of validation that is required before approved introduction into clinical practice is based. For example, to receive the CE mark for a clinical decision support (CDS) algorithm—depending on classification—the EU requires compliance with ISO 13485 standards, as well as a clinical evaluation report (CER) that includes a literature review and clinical testing (validation) [39].

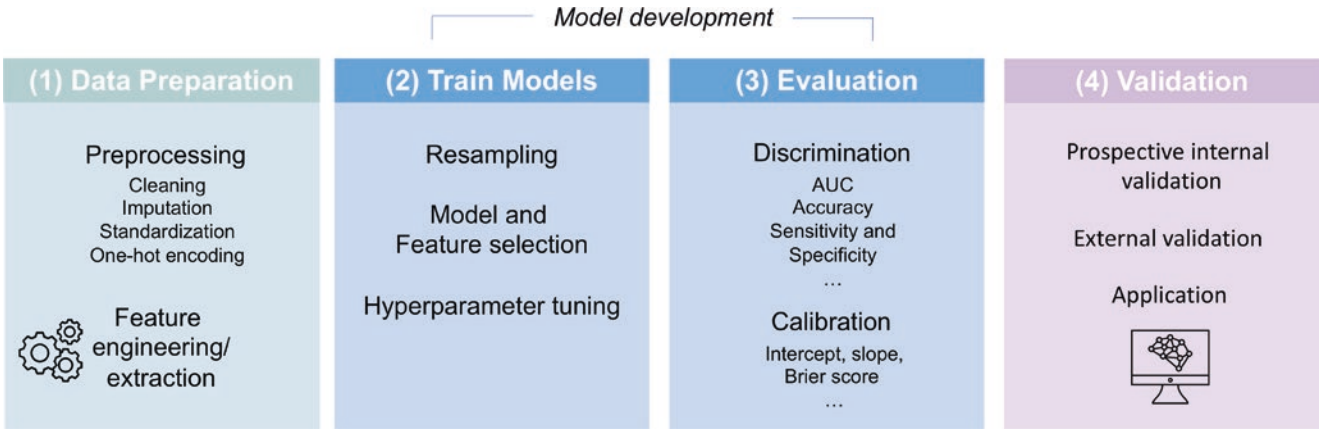

Fig. 2.2 A four-step predictive modeling workflow. (1) Data preparation includes cleaning and featurization of the given raw data. Data preprocessing combines cleaning and outlier detection, missing data imputation, the use of standardization methods, and correct feature encoding. The pre-processed data is further formed into features—manually in a process called *feature engineering* or automatically deduced by a process called *feature extraction*. In the training process (2) resampling techniques such as *k*-fold cross-validation are used to train and tune different models. Most predictive features are identified in a *feature selection* process. (3) Models are compared and evaluated for generalizability in an independent testing set. The best performing model is selected, and out-of-sample discrimination and calibration are assessed. (4) The generalizing model is prospectively internally and externally validated to ensure safe clinical usage across locations and users

2.6 Conclusion

We appear to be at the beginning of an accelerated trend towards data-driven decision-making in biomedicine enabled by a transformative technology—machine learning [5]. Given the ever-growing and highly complex "big data" biomedical datasets and increases in computational power, machine learning approaches prove to be highly successful analytical strategies towards a patient-tailored approach regarding diagnosis, treatment choice, and outcome prediction. Going forward, we expect that training neuroscientists and clinicians in the concepts of machine learning will undoubtably be a corner stone for the advancement of individualized medicine in the realm of precision medicine. With the series "*Machine learning-based clinical prediction modeling*," we aim to provide both a conceptual and practical guideline for predictive analytics in the clinical routine to strengthen every clinician's competence in modern machine learning techniques.

Disclosures

Funding No funding was received for this research.

Conflict of Interest All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

Ethical Approval All procedures performed in studies involving human participants were in accordance with the ethical standards of the 1964 Helsinki declaration and its later amendments or comparable ethical standards.

Informed Consent No human or animal participants were included in this study.

References

1. Kuhn M, Johnson K. Applied predictive modeling. New York, NY: Springer Science & Business Media; 2013.
2. Steyerberg EW. Clinical prediction models: a practical approach to development, validation, and updating. New York, NY: Springer Science & Business Media; 2008.
3. Senders JT, Zaki MM, Karhade AV, Chang B, Gormley WB, Broekman ML, Smith TR, Arnaout O. An introduction and overview of machine learning in neurosurgical care. Acta Neurochir. 2018;160:29. https://doi.org/10.1007/s00701-017-3385-8.
4. Hastie T, Tibshirani R, Friedman J. The elements of statistical learning: data mining, inference, and prediction. New York, NY: Springer Science & Business Media; 2013.
5. Jordan MI, Mitchell TM. Machine learning: trends, perspectives, and prospects. Science. 2015;349:255. https://doi.org/10.1126/science.aaa8415.
6. Hastie T, Tibshirani R, Wainwright M. Statistical learning with sparsity: the lasso and generalizations. New York, NY: Chapman and Hall; 2015. https://doi.org/10.1201/b18401.
7. Tibshirani R. Regression shrinkage and selection via the Lasso. J R Stat Soc Ser B Methodol. 1996;58:267. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x.
8. Aerts HJWL, Velazquez ER, Leijenaar RTH, et al. Decoding tumour phenotype by noninvasive imaging using a quantitative radiomics approach. Nat Commun. 2014;5:4006. https://doi.org/10.1038/ncomms5006.
9. Li H, Zhu Y, Burnside ES, et al. Quantitative MRI radiomics in the prediction of molecular classifications of breast cancer subtypes in the TCGA/TCIA data set. NPJ Breast Cancer. 2016;2:16012. https://doi.org/10.1038/npjbcancer.2016.12.
10. Thawani R, McLane M, Beig N, Ghose S, Prasanna P, Velcheti V, Madabhushi A. Radiomics and radiogenomics in lung cancer: a review for the clinician. Lung Cancer. 2018;115:34. https://doi.org/10.1016/j.lungcan.2017.10.015.
11. Weng SF, Vaz L, Qureshi N, Kai J. Prediction of premature all-cause mortality: a prospective general population cohort study comparing machine-learning and standard epidemiological approaches. PLoS One. 2019;14(3):e0214365.
12. Mitchell TM. The discipline of machine learning. Mach Learn. 2006;17:1. https://doi.org/10.1080/026404199365326.
13. Keskinbora KH. Medical ethics considerations on artificial intelligence. J Clin Neurosci. 2019;64:277. https://doi.org/10.1016/j.jocn.2019.03.001.
14. Shortliffe EH, Sepúlveda MJ. Clinical decision support in the era of artificial intelligence. JAMA. 2018;320:2199. https://doi.org/10.1001/jama.2018.17163.
15. Grigsby J, Kramer RE, Schneiders JL, Gates JR, Smith WB. Predicting outcome of anterior temporal lobectomy using simulated neural networks. Epilepsia. 1998;39:61. https://doi.org/10.1111/j.1528-1157.1998.tb01275.x.
16. Bzdok D, Krzywinski M, Altman N. Points of significance: machine learning: supervised methods. Nat Methods. 2018;15:5. https://doi.org/10.1038/nmeth.4551.
17. Altman N, Krzywinski M. Points of significance: clustering. Nat Methods. 2017;14:545. https://doi.org/10.1038/nmeth.4299.
18. Murphy KP. Machine learning: a probabilistic perspective. Cambridge, MA: MIT Press; 2012.
19. Wolpert DH. The lack of a priori distinctions between learning algorithms. Neural Comput. 1996;8:1341. https://doi.org/10.1162/neco.1996.8.7.1341.
20. Domingos P. A few useful things to know about machine learning. Commun ACM. 2012;55(10):78.
21. Armañanzas R, Alonso-Nanclares L, DeFelipe-Oroquieta J, Kastanauskaite A, de Sola RG, DeFelipe J, Bielza C, Larrañaga P. Machine learning approach for the outcome prediction of temporal lobe epilepsy surgery. PLoS One. 2013;8:e62819. https://doi.org/10.1371/journal.pone.0062819.
22. Bydon M, Schirmer CM, Oermann EK, Kitagawa RS, Pouratian N, Davies J, Sharan A, Chambless LB. Big data defined: a practical review for neurosurgeons. World Neurosurg. 2020;133:e842. https://doi.org/10.1016/j.wneu.2019.09.092.
23. Senders JT, Staples PC, Karhade AV, Zaki MM, Gormley WB, Broekman MLD, Smith TR, Arnaout O. Machine learning and neurosurgical outcome prediction: a systematic review.

World Neurosurg. 2018;109:476. https://doi.org/10.1016/j.wneu.2017.09.149.
24. Bottou L. Large-scale machine learning with stochastic gradient descent. In: Proc COMPSTAT2010; 2010. https://doi.org/10.1007/978-3-7908-2604-3_16.
25. Ruder S. An overview of gradient descent optimization algorithms. ArXiv. 2017:160904747. Cs.
26. Goodfellow I, Bengio Y, Courville A. Deep learning. Cambridge, MA: MIT Press; 2016.
27. Sra S, Nowozin S, Wright SJ. Optimization for machine learning. Cambridge, MA: MIT Press; 2012.
28. Gravesteijn BY, Nieboer D, Ercole A, Lingsma HF, Nelson D, van Calster B, Steyerberg EW, CENTER-TBI Collaborators. Machine learning algorithms performed no better than regression models for prognostication in traumatic brain injury. J Clin Epidemiol. 2020;122:95–107.
29. Breiman L. Statistical modeling: the two cultures (with comments and a rejoinder by the author). Stat Sci. 2001;16(3):199–231.
30. Bzdok D. Classical statistics and statistical learning in imaging neuroscience. Front Neurosci. 2017;11:543. https://doi.org/10.3389/fnins.2017.00543.
31. Gabrieli JDE, Ghosh SS, Whitfield-Gabrieli S. Prediction as a humanitarian and pragmatic contribution from human cognitive neuroscience. Neuron. 2015;85:11. https://doi.org/10.1016/j.neuron.2014.10.047.
32. Shmueli G. To explain or to predict? Stat Sci. 2011;25(3):289–310.
33. Whelan R, Garavan H. When optimism hurts: inflated predictions in psychiatric neuroimaging. Biol Psychiatry. 2014;75:746. https://doi.org/10.1016/j.biopsych.2013.05.014.
34. Yarkoni T, Westfall J. Choosing prediction over explanation in psychology: lessons from machine learning. Perspect Psychol Sci. 2017;12:1100. https://doi.org/10.1177/1745691617693393.
35. Domingos P. A unified bias-variance decomposition and its applications. In: Proc 17th Int. Conf Mach. Learn. San Francisco, CA: Morgan Kaufmann; 2000. p. 231–8.
36. James G, Hastie T. Generalizations of the bias/variance decomposition for prediction error. Stanford, CA: Department of Statistics, Stanford University; 1997.
37. Abu-Mostafa YS, Malik M-I, Lin HT. Learning from data: a short course. Chicago, IL: AMLBook; 2012. https://doi.org/10.1108/17538271011063889.
38. Van der Laan M, Hubbard AE, Jewell N. Learning FROM DATA. Epidemiology. 2010;21:479. https://doi.org/10.1097/ede.0b013e3181e13328.
39. Harvey H. How to get clinical AI tech approved by regulators. Medium; 2019. https://towardsdatascience.com/how-to-get-clinical-ai-tech-approved-by-regulators-fa16dfa1983b. Accessed 3 May 2020.

3 Foundations of Machine Learning-Based Clinical Prediction Modeling: Part II—Generalization and Overfitting

Julius M. Kernbach and Victor E. Staartjes

3.1 Introduction

In the first part of this review series, we have discussed general and important concepts of machine learning (ML) and presented a four-step workflow for machine learning-based predictive pipelines. However, many regularly faced challenges, which are well-known within the ML community, are less established in the clinical community. One common source of trouble is *overfitting*. It is a common pitfall in predictive modeling, whereby the model not only fits the true underlying relationship of the data but also fits the individual biological or procedural noise associated with each observation. Dealing with overfitting remains challenging in both regression and classification problems. Erroneous pipelines or ill-suited applied models may lead to drastically inflated model performance, and ultimately cause unreliable and potentially harmful clinical conclusions. We discuss and illustrate different strategies to address overfitting in our analyses including *resampling methods*, regularization and penalization of model complexity [1]. In addition, we discuss *feature selection* and *feature reduction*. In this section, we review overfitting as potential danger in predictive analytic strategies with the goal of providing useful recommendations for clinicians to avoid flawed methodologies and conclusions (Table 3.1).

3.2 Overfitting

Overfitting occurs when a given model adjusts too closely to the training data, and subsequently demonstrates poor performance on the testing data (Fig. 3.1). While the model's goodness of fit to the present data sample seems impressive, the model will be unable to make accurate predictions on new observations. This scenario represents a major pitfall in ML. At first, the performance within the training data seems excellent, but when the model's performance is evaluated on the hold-out data ("out-of-sample error") it generalizes poorly. There are various causes of overfitting, some of which are intuitive and easily mitigated. Conceptually, the easiest way to overfit is simply by memorizing observations [2–4].

We simply remember all data patterns, important patterns as well as unimportant ones. For our training data, we will get an exceptional model fit, and minimal training error by recalling the known observations from memory—implying the illusion of success. However, once we test the model's performance on independent test data, we will observe predictive performance that is no better than random. By overtraining on the present data, we end up with a too close fit to the training observations. This fit only partially reflects the underlying true data-generating process, but also includes random noise specific to the training data. This can either be sample-specific noise, both procedural as well as biological, but also the hallucination of unimportant patterns [5]. Appling the overfitted model to new observations will out itself as an out-of-sample performance that is massively worse than the training performance. In this way, the amount of overfitting can be defined as the difference among discriminatory training and testing performance—while it is

J. M. Kernbach and V. E. Staartjes have contributed equally to this series, and share first authorship.

J. M. Kernbach
Neurosurgical Artificial Intelligence Laboratory Aachen (NAILA), Department of Neurosurgery, RWTH Aachen University Hospital, Aachen, Germany

V. E. Staartjes (✉)
Machine Intelligence in Clinical Neuroscience (MICN) Laboratory, Department of Neurosurgery, Clinical Neuroscience Center, University Hospital Zurich, University of Zurich, Zurich, Switzerland
e-mail: victoregon.staartjes@usz.ch; https://micnlab.com/

V. E. Staartjes et al. (eds.), *Machine Learning in Clinical Neuroscience*, Acta Neurochirurgica Supplement 134,
https://doi.org/10.1007/978-3-030-85292-4_3

Table 3.1 Concept summaries

Concept	Explanation
Noise	Noise is unexplained and random variation inherent to the data (biological noise) or introduced by variables of no interest (procedural noise, including measurement errors, site variation).
Overfitting	Over-learning of random patterns associated with noise or memorization in the training data. Overfitting leads to a drastically decreased ability to generalize to new observations.
Bias	Bias quantifies the error term introduced by approximating highly complicated real-life problems by a much simpler statistical model. Models with high bias tend to underfit.
Variance	Variance refers to learning random structure irresponsible of the underlying true signal. Models with high variance tend to overfit.
Data Leakage/ Contamination	Or the concept of "looking at data twice". Overfitting is introduced when observations used for testing also re-occur in the training process. The model then "remembers" instead of learning the underlying association.
Model Selection	Iterative process using resampling such as *k*-fold cross-validation to fit different models in the training set.
Model Assessment	Evaluation of a model's out-of-sample performance. This should be conducted on a test set of data that was set aside and not used in training or model selection. The use of multiple measures of performance (AUC, F1, etc.) is recommended.
Resampling	Resampling methods fit a model multiple times on different subsets of the training data. Popular methods are *k*-fold cross-validation and the bootstrap.
k-Fold Cross-Validation	Data is divided in k equally sized folds/sets. Iteratively, $k-1$ data is used for training and evaluated on the remaining unseen fold. Each fold is used for testing once.
LOOCV	LOOCV (leave-one-out cross-validation) is a variation of cross-validation. Each observation is left out once, the model is trained on the remaining data, and then evaluated on the held-out observation.
Bootstrap	The bootstrap allows to estimate the uncertainty associated with any given model. Typically, in 1000–10,000 iterations bootstrapped samples are repetitively drawn with replacement from the original data, the predictive model is iteratively fit and evaluated.
Hyperparameter Tuning	Hyperparameters define how a statistical model learns and need to be specified before training. They are model specific and might include regularization parameters penalizing model's complexity (ridge, lasso), number of trees and their depth (random forest), and many more. Hyperparameters can be tuned, that is, iteratively improved to find the model that performs best given the complexity of the available data.

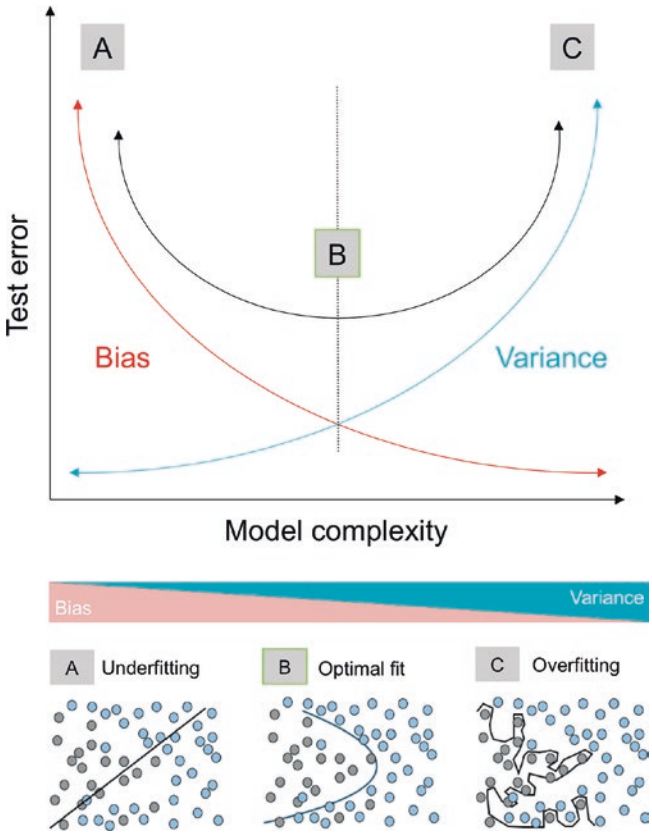

Fig. 3.1 Conceptual visualization of the bias-variance trade-off. A predictive model with *high bias* and *low variance* (**A**), consistently approximates the underlying data-generating process with a much simpler model (here a hyperplane), and hence result in an underfit solution. (**B**) A U-shaped decision boundary represents the optimal solution in this scenario, here, both bias and variance are low, resulting in the lowest test error. (**C**) Applying an overly flexible model results in overfitting. Data quirks and random non-predictive structures that are unrelated to the underlying signal are learned

normal that out-of-sample performance is equal to or ever so slightly worse than training performance for any adequately fitted model, a massive difference suggests relevant overfitting. This is one reason why in-sample model performance should never be reported as evidence for predictive performance. Instead model training and selection should always be performed on a separate train set, and only in the final step should the final model be evaluated on an independent test set to judge true out-of-sample performance.

The Bias-Variance Trade-Off

In ML we opt to make accurate and generalizable predictions. When the test error is significantly higher than the training error, we can diagnose overfitting. To understand what is going on we can decompose the predictive error into

its essential parts *bias* and *variance* [6, 7]. Their competing nature, commonly known under the term *bias-variance trade-off*, is very important and notoriously famous in the machine learning community. Despite its fame and importance, the concept is less prominent within the clinical community. *Bias* quantifies the error term introduced by approximating highly complicated real-life problems by a much simpler statistical model, that is underfitting the complexity of the data-generating process. In other words, a model with high bias tends to consistently learn the wrong response. That by itself does not necessarily need to be a problem, as simple models were often found to perform very well sometimes even better than more sophisticated ones [8]. However, for maximal predictive compacity we need to find the perfect balance between bias and variance. The term *variance* refers to learning random structure irresponsible of the underlying true signal. That is, models with high variance can hallucinate patterns that are not given by the reality of the data. Figure 3.1 illustrates this in a classification problem. A linear model (Fig. 3.1a, high bias and low variance) applied to class data, in which the frontier between the two classes is not a hyperplane, is unable to induce the underlying true boundary. It will consistently learn the wrong response, that is a hyperplane, despite the more complex true decision boundary and result into "underfitting" the true data-generating process. On the other extreme, an excessively flexible model with high variance and low bias (Fig. 3.1c) will learn random non-predictive structure that is unrelated to the underlying signal. Given minimally different observations, the overly flexible model fit could drastically change in an instance. The latter complex model would adapt well to all training observations but would ultimately fail to generalize and predict new observations in an independent test set. Neither the extremely flexible nor the insufficiently flexible model is capable of generalizing to new observations.

Combatting Overfitting: Resampling

We could potentially collect more data for an independent cohort to test our model, but this would be highly time-consuming and expensive. In rich data situations, we can alternatively split our sample into a data set for training and a second set for testing (or hold-out set) to evaluate the model's performance in new data (i.e., the model's out-of-sample performance) more honestly. We would typically use a random 80%/20% split for training and testing (while remaining class balance within the training set, see Chap. 4). Because we often lack a sufficiently large cohort of patients to simply evaluate generalization performance using data splits, we need to use a less data-hungry but equally efficient alternatives. The gold standard and popular approach in machine learning to address overfitting is to evaluate the model's generalization ability via *resampling methods* [9]. Some of these resampling methods—particularly the bootstrap—have already long been used in inferential statistical analysis to generate measures of variance [10]. Resampling methods are an indispensable tool in today's modern data science and include various forms of *cross-validation* [3, 11]. All forms have a common ground: they involve splitting the available data iteratively into a non-overlapping train and test set. Our statistical model is then refitted and tested for each subset of the train and test data to obtain an estimate of generalization performance. Most modern resampling methods have been derived from the jackknife—a resampling technique developed by Maurice Quenouille in 1949 [12]. The simplest modern variation of cross-validation—also based on the jackknife—is known as leave-one-out cross-validation (LOOCV). In LOOCV, the data (n) is iteratively divided into two unequal subsets with the train set of $n - 1$ observations and the test set containing the remaining one observation. The model is refitted and evaluated on the excluded held-out observation. The procedure is then repeated n times and the test error is then averaged over all iterations. A more popular alternative to LOOCV and generally considered the gold standard is k-fold cross-validation (Fig. 3.2). The k-fold approach randomly divides the available data into a k amount of non-overlapping groups, or folds, of approximately equal size. Empirically, $k = 5$ or $k = 10$ are preferred and commonly used [13]. Each fold is selected as test set once, and the model is fitted on the remaining $k - 1$ folds. The average over all fold-wise performances estimates the generalizability of a given statistical model. Within this procedure, importantly, no observation is selected for both training and testing. This is essential, because, as discussed earlier, predicting an observation that was already learned during training equals memorization, which in turn leads to overfitted conclusions.

Cross-validation is routinely used in both model selection and model assessment. Yet another extremely powerful and popular resampling strategy is the *bootstrap* [14, 15], which allows for the estimation of the accuracy's uncertainty applicable to nearly any statistical method. Here, we obtain new bootstrapped sets of data by repeatedly sampling observations from the original data set *with replacement*, which means any observation can occur more than once in the bootstrapped data sample. Thus, when applying the bootstrap, we repeatedly randomly select n patients from an n-sized training dataset, and model performance is evaluated after every iteration. This process is repeated many times—usually with 25–1000 repetitions.

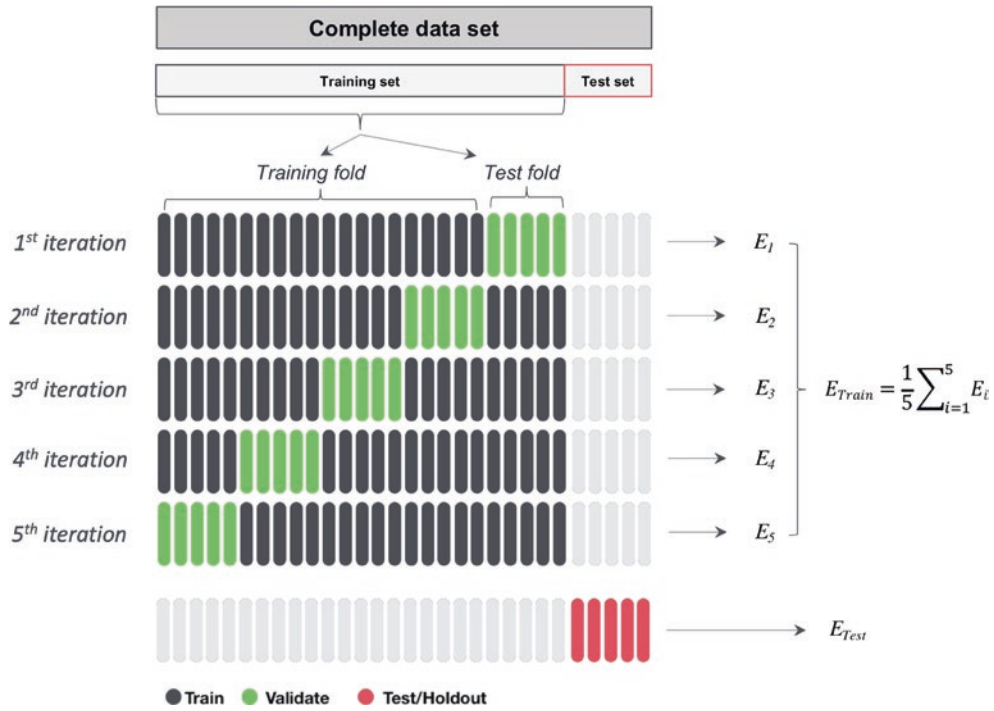

Fig. 3.2 *k*-fold cross-validation with an independent hold-out set. The complete dataset is portioned into training data (~80%) and testing data (~20%) before any resampling is applied. Within the training set, *k*-fold cross-validation is used to randomly divide the available data into $k = 5$ equally sized folds. Iteratively, $k - 1$ folds are used to train a chosen model, and the fold-wise performance (E_k) is evaluated on the remaining unseen validation fold. These fold-wise performances are averaged, and together, the out-of-sample performance is estimated as E_{Train}. When different models are trained, the best performing one is selected and tuned (model selection, hyperparameter tuning) and evaluated on the independent hold-out set (or "test set"). The resulting performance E_{Test} is reported and estimates the predictive performance beyond the present data

Considerations on Algorithm Complexity

To avoid over- or underfitting, an appropriate level of model complexity is required [11, 16]. Modulating complexity can be achieved by adding a regularization term, which can be used with any type of predictive model. In that instance, the regularization term is added to favor less-complex models with less room to overfit. As complexity is intrinsically related to the number and magnitude of parameters, we can add a regularization or penalty term to control the magnitude of the model parameters, or even constrain the number of parameters used. There are many different penalties specific to selected models. In a regression setting, we could add either a *L1 penalty* (LASSO, least absolute shrinkage and selection operator), which selectively removes variables form the model, a *L2 penalty* (Ridge or Tikhonov regularization), which shrinks the magnitude of parameters but never fully removes them from the model or an *elastic net* (combination of L1 and L2) [13, 17, 18]. For neural networks, *dropout* is a very efficient regularization method [19]. Finding the right balance based on regularization, that is, to define how complex a model can be, is controlled by the model's hyperparameters (L1 or L2 penalty term in regression, and many more). Restraining model complexity by adding a regularization term is an example of a model hyperparameter. Typically, hyperparameters are *tuned*, which means that the optimal level is evaluated during model training. Again, it is important to respect the distinction of train and test data. As a simple guideline, we recommend to automate all necessary pre-processing steps including hyperparameter tuning within the chosen resampling approach to ensure none of the above are performed on the complete data set before cross-validation [20]. Otherwise, this would result in circularity and inflate the overall predictive performance [21].

Data Leakage

Whenever resampling techniques are applied, the investigator has to ensure that *data leakage* or *data contamination* is not accidently introduced. From the standpoint of ML, data contamination—part of the test data leaking into the model-fitting procedure—can have severe consequences, and lead to drastically inflated predictive performance. Therefore, caution needs to be allocated to the clean isolation of train and test data. As a general rule-of-thumb, no feature

selection or dimensionality reduction method that involves the outcome measure should be performed on the complete data set before cross-validation or splitting. This would open doors for procedural bias, and raise concerns regarding model validity. Additionally, nested cross-validation should be used in model selection and hyperparameter tuning. The nestedness adds an additional internal cross-validation loop to guarantee clean distinction between the "test data" for model selection and tuning and ultimately the "test data" used for model performance assessment.

Usually the data splits are then named "train"—"test"—"(external) validation," however, different nomenclatures are frequently used.

While resampling techniques can mitigate overfitting, they can also lead to manual overfitting when too many hyperparameter choices are made in the process [22]. Another consideration to keep in mind is that whenever a random data split is selected, it is with the assumption that each split is representative of the full data set. This can become problematic in two cases: (1) When data is dependent, data leakage occurs when train and test data share non-independent observations, such as the inclusion of both the index and revision surgery of patients. Both observations are systematically similar, induce overfitting and ultimately undermine the validity of the resulting model performance. (2) When data is not identically distributed: this is a serious problem in small sample scenarios, where splits are drawn out of a highly variable set of observations. Depending on which of the patients end up in the train or test data, the model performance can greatly fluctuate, and can be an overly optimistic estimate of predictive performance. Generally, less inflated predictive performance can be observed as the sample size increases [23]. As studies based on small sample sizes can generate highly variable estimates, conclusions may often be exaggerated or even invalid. Hence, predictive modeling should be restricted or used with caution when only small amounts of data are available. Considerations regarding sample size are discussed in *Part III*.

3.3 Importance of External Validation in Clinical Prediction Modeling

External validation of clinical prediction models represents an important part in their development and rollout [24, 25]. In order to generalize, the input data, i.e. the training sample, needs to be *representative*. However, without external validation, the *site bias* or *center bias*, which includes variations in treatment protocols, surgical techniques, level of experience between departments and clinical users, as well as the so-called *sampling/selection bias*, which refers to systematically different data collection in regard to the patient cohort, cannot be detected. For these reasons, an empirical assessment of model performance on an unrelated, "external" dataset is required before an application can publicly be released. Erroneous or biased predictions can have severe sequelae for patients and clinicians alike, if misjudgments are made based upon such predictions. As a gold standard, external validation enables *unbiased testing* of model performance in a new cohort with different demographics. If a clinical prediction model shows comparable discrimination and calibration performance at external validation, generalizability may be confirmed. Then, it may be safe to release the model into the clinical decision-making progress. As an alternative to external validation—certainly the gold standard to ensure generalizability of a clinical prediction model—one might consider prospective internal validation (i.e. validation on a totally new sample of patients who are, however, derived from the same center with the same demographics, surgeons, and treatment protocols as the originally developed model). While prospective internal validation will also identify any overfitting that might be present, and will enable safe use of the prediction model at that specific center, this method does not allow ruling out center bias, i.e. does not ensure the safe use of the model in other populations.

3.4 Feature Reduction and Selection

In overtly complex and high-dimensional data with too many parameters, we find ourselves in an over-parameterized analytical setting. However, due to 'the curse of dimensionality'—a term famously coined by Richard Bellmann in 1961—generalization becomes increasingly more difficult in high dimensions. The approach to avoid "the curse" has been to find lower representation of the given feature space [26]. If there were too many features or variables present, *feature reduction* or *feature selection* methods can be applied. In *feature reduction*, methods are applied to simplify the complexity of the given high-dimensional data while retaining important and innate patterns of the data. Principal component analysis (PCA) is a popular illustration [27]. As an unsupervised ML method PCA is conceptually similar to clustering, and learns from data without any reference or a priori knowledge of the predicted outcome. Analytically, PCA reduces high-dimensional data by projecting them onto the so-called principal components, which represent summaries of the data in fewer dimensions. PCA can hence be used as a strong statistical tool to reduce the main axis of variance within a given feature space. *Feature selection* refers to a similar procedure, which is also applied to initially too large feature spaces to reduce the number of input features. The key in feature selection is not to summarize data into lower dimensions as in feature reduction, but to actually reduce the number of included features to end up with only the "most useful" ones—and eliminate all non-informative ones. Naturally, if certain domain knowl-

edge is present, vast sets of features can be constructed to a better set of informative features. For instance, in brain imaging, voxels of an MRI scan can either be considered individually or can be summarized into functionally or anatomically homogenous areas—a concept of topographical segregation that dates back to Brodmann [28, 29]. The problem of feature selection is well-known in the ML community and has generated a vast body of literature early on [30, 31]. A common pruning technique to select features that together maximize, e.g. classification performance is *recursive feature elimination* (RFE) [32, 33]. In RFE, a given classifier or regressor is iteratively trained, and a ranking criterion for all features is estimated. The feature with the smallest respective ranking criterion is then eliminated. Introduced by Guyon and colleagues [32], RFE was initially used to extract small subsets of highly discriminant genes in DNA arrays and build reliable cancer classifiers. As an instance of backward elimination—that is, we start with the complete set of variables and progressively eliminate the least informative features—RFE can be used both in classification and regression settings with any given learner, but remains computationally greedy ("brute force"), as many different, e.g. classifiers on feature subsets of decreasing size are revisited. As an important consideration, RFE selects *subsets* of variables based on an optimal *subset* ranking criterion. Consequently, a group of features combined may lead to optimal predictive performance, while the individual features included do not necessarily have to be the most important. Embedded in the process of model training, variable selection procedures such as RFE can improve performance by selecting subsets of variables that together maximize predictive power. Importantly, resampling methods should be applied when using RFE to factor in the variability caused by feature selection when calculating performance.

3.5 Conclusion

Overfitting is a multifactorial problem, and there are just as many possible approaches to reduce its negative impact. We encourage the use of resampling methods such as cross-validation in every predictive modeling pipeline. While there are various options to choose from, we recommend the usage of *k*-fold cross-validation or the bootstrap. Nested loops may be used for hyperparameter tuning and model selection. While the use of resampling does not solve overfitting, it helps to gain a more representative understanding of the predictive performance, especially of out-of-sample error. Feature reduction and selection methods, such as PCA and RFE are introduced for handling high-dimensional data. A potential pitfall along the way is *data contamination*, which occurs when data leaks from the resampled test to train set and hence leads to overconfident model performance. We encourage the use of standardized pipelines (see Chaps. 5 and 6 here for examples), which include feature engineering, hyperparameter tuning and model selection within one loop to minimize the risk of unintentionally leaking test data. Finally, we recommend including a regularization term as hyperparameter and to restrict extensive model complexity, which will avoid overfitted predictive performance.

Disclosures

Funding No funding was received for this research.

Conflict of Interest All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers' bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

Ethical Approval All procedures performed in studies involving human participants were in accordance with the ethical standards of the 1964 Helsinki declaration and its later amendments or comparable ethical standards.

Informed Consent No human or animal participants were included in this study.

References

1. Domingos P. Process-oriented estimation of generalization error. In: IJCAI Int. Jt. Conf. Artif. Intell; 1999. p. 714–9.
2. Arplt D, Jastrzebskl S, Bailas N, et al. A closer look at memorization in deep networks. In: 34th Int. Conf. Mach. Learn. ICML 2017; 2017.
3. Goodfellow I, Yoshua Bengio AC. Deep learning book. In: Deep learn. Cambridge, MA: MIT Press; 2015. https://doi.org/10.1016/B978-0-12-391420-0.09987-X.
4. Zhang C, Vinyals O, Munos R, Bengio S. A study on overfitting in deep reinforcement learning. ArXiv. 2018:180406893.
5. Domingos P. A few useful things to know about machine learning. Commun ACM. 2012;55(10):78.
6. Domingos P. A unified bias-variance decomposition and its applications. In: Proc 17th Int. Conf Mach. Learn. San Francisco, CA: Morgan Kaufmann; 2000. p. 231–8.
7. James G, Hastie T. Generalizations of the bias/variance decomposition for prediction error. Stanford, CA: Department of Statistics, Stanford University; 1997.
8. Holte RC. Very simple classification rules perform well on most commonly used datasets. Mach Learn. 1993;11:63. https://doi.org/10.1023/A:1022631118932.
9. Staartjes VE, Kernbach JM. Letter to the editor regarding "Investigating risk factors and predicting complications in deep brain stimulation surgery with machine learning algorithms". World Neurosurg. 2020;137:496.
10. Davison AC, Hinkley DV. Bootstrap methods and their application. Cambridge: Cambridge University Press; 1997.
11. Gravesteijn BY, Nieboer D, Ercole A, Lingsma HF, Nelson D, van Calster B, Steyerberg EW, CENTER-TBI Collaborators. Machine

learning algorithms performed no better than regression models for prognostication in traumatic brain injury. J Clin Epidemiol. 2020;122:95–107.
12. Quenouille MH. Notes on bias in estimation. Biometrika. 1956;43(3–4):353–60.
13. Hastie T, Tibshirani R, Friedman J. The elements of statistical learning: data mining, inference, and prediction. New York, NY: Springer Science & Business Media; 2013.
14. Efron B, Tibshirani RJ. An introduction to the bootstrap. New York, NY: Chapman and Hall; 1993. https://doi.org/10.1007/978-1-4899-4541-9.
15. Hastie T, Tibshirani R, James G, Witten D. An introduction to statistical learning. New York, NY: Springer; 2006. https://doi.org/10.1016/j.peva.2007.06.006.
16. Staartjes VE, Kernbach JM. Letter to the editor. Importance of calibration assessment in machine learning-based predictive analytics. J Neurosurg Spine. 2020;32:985–7.
17. Lever J, Krzywinski M, Altman N. Points of significance: regularization. Nat Methods. 2016;13:803. https://doi.org/10.1038/nmeth.4014.
18. Zou H, Hastie T. Regularization and variable selection via the elastic net. J R Stat Soc Ser B Stat Methodol. 2005;67:301. https://doi.org/10.1111/j.1467-9868.2005.00503.x.
19. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res. 2014;15(1):1929–58.
20. Poldrack RA, Huckins G, Varoquaux G. Establishment of best practices for evidence for prediction: a review. JAMA Psychiatry. 2019;77:534. https://doi.org/10.1001/jamapsychiatry.2019.3671.
21. Kriegeskorte N, Simmons WK, Bellgowan PS, Baker CI. Circular analysis in systems neuroscience: the dangers of double dipping. Nat Neurosci. 2009;12:535. https://doi.org/10.1038/nn.2303.
22. Ng AY. Preventing "overfitting" of cross-validation data. CEUR Workshop Proc. 2015;1542:33. https://doi.org/10.1017/CBO9781107415324.004.
23. Varoquaux G. Cross-validation failure: small sample sizes lead to large error bars. NeuroImage. 2018;180:68. https://doi.org/10.1016/j.neuroimage.2017.06.061.
24. Collins GS, Ogundimu EO, Le Manach Y. Assessing calibration in an external validation study. Spine J. 2015;15:2446. https://doi.org/10.1016/j.spinee.2015.06.043.
25. Staartjes VE, Schröder ML. Class imbalance in machine learning for neurosurgical outcome prediction: are our models valid? J Neurosurg Spine. 2018;26:736. https://doi.org/10.3171/2018.5.SPINE18543.
26. Breiman L. Statistical modeling: the two cultures (with comments and a rejoinder by the author). Stat Sci. 2001;16(3):199–231.
27. Lever J, Krzywinski M, Altman N. Points of significance: principal component analysis. Nat Methods. 2017;14:641. https://doi.org/10.1038/nmeth.4346.
28. Amunts K, Zilles K. Architectonic mapping of the human brain beyond brodmann. Neuron. 2015;88:1086. https://doi.org/10.1016/j.neuron.2015.12.001.
29. Glasser MF, Coalson TS, Robinson EC, et al. A multi-modal parcellation of human cerebral cortex. Nature. 2016;536:171. https://doi.org/10.1038/nature18933.
30. Blum AL, Langley P. Selection of relevant features and examples in machine learning. Artif Intell. 1997;97:245. https://doi.org/10.1016/s0004-3702(97)00063-5.
31. Kohavi R, John GH. Wrappers for feature subset selection. Artif Intell. 1997;97:273. https://doi.org/10.1016/s0004-3702(97)00043-x.
32. Guyon I, Weston J, Barnhill S, Vapnik V. Gene selection for cancer classification using support vector machines. Mach Learn. 2002;46:389. https://doi.org/10.1023/A:1012487302797.
33. Iguyon I, Elisseeff A. An introduction to variable and feature selection. J Mach Learn Res. 2003;3:1157. https://doi.org/10.1162/153244303322753616.

4 Foundations of Machine Learning-Based Clinical Prediction Modeling: Part III—Model Evaluation and Other Points of Significance

Victor E. Staartjes and Julius M. Kernbach

4.1 Introduction

Once a dataset has been adequately prepared and a training structure (e.g. with a resampling method such as *k*-fold cross validation, see Chap. 3) has been set up, a model is ready to be trained. Already during training and the subsequent model tuning and selection, metrics to evaluate model performance become of central importance, as the hyperparameters and parameters of the models are tuned according to one or multiple of these performance metrics. In addition, after a final model has been selected based on these metrics, internal or external validation should be carried out to assess whether the same performance metrics can be achieved as during training. This section walks the reader through some of the common performance metrics to evaluate the discrimination and calibration of clinical prediction models based on machine learning (ML). We focus on clinical prediction models for continuous and binary endpoints, as these are by far the most common clinical applications of ML in neurosurgery. Multiclass classification—thus, the prediction of a categorical endpoint with more than two levels—may require other performance metrics.

Second, when developing a new clinical prediction model, there are several caveats and other points of significance that the readers should be aware of. These include what sample size is necessary for a robust model, how to pre-process data correctly, how to handle missing data and class imbalance, how to choose a cutoff for binary classification, and why extrapolation is problematic. In the second part of this section, these topics are sequentially discussed.

4.2 Evaluation of Classification Models

The Importance of Discrimination and Calibration

The performance of classification models can roughly be judged along two dimensions: Model discrimination and calibration [1]. The term *discrimination* denotes the ability of a prediction model to correctly classify whether a certain patient is going to or is not going to experience a certain outcome. Thus, discrimination described the accuracy of a binary prediction—yes or no. *Calibration*, however, describes the degree to which a model's predicted probabilities (ranging from 0% to 100%) correspond to the actually observed incidence of the binary endpoint (true posterior). Many publications do not report calibration metrics, although these are of central importance, as a well-calibrated predicted probability (e.g. your predicted probability of experiencing a complication is 18%) is often much more valuable to clinicians—and patients!—than a binary prediction (e.g. you are likely not going to experience a complication) [1].

There are other factors that should be considered when selecting models, such as complexity and interpretability of the algorithm, how well a model calibrates out-of-the-box, as well as e.g. the computing power necessary [2]. For instance, choosing an overly complex algorithm for relatively simple data (i.e. a deep neural network for tabulated medical data) will vastly increase the likelihood of overfitting with only negligible benefits in performance. Similarly, even though discrimination performance may be ever so slightly better with a more complex model such as a neural network, this

J. M. Kernbach and V. E. Staartjes have contributed equally to this series, and share first authorship.

V. E. Staartjes (✉)
Machine Intelligence in Clinical Neuroscience (MICN) Laboratory, Department of Neurosurgery, Clinical Neuroscience Center, University Hospital Zurich, University of Zurich, Zurich, Switzerland
e-mail: victoregon.staartjes@usz.ch; https://micnlab.com/

J. M. Kernbach
Neurosurgical Artificial Intelligence Laboratory Aachen (NAILA), Department of Neurosurgery, RWTH Aachen University Hospital, Aachen, Germany

V. E. Staartjes et al. (eds.), *Machine Learning in Clinical Neuroscience*, Acta Neurochirurgica Supplement 134,
https://doi.org/10.1007/978-3-030-85292-4_4

comes at the cost of reduced interpretability ("black box" models) [3]. The term "black box" model denotes a model for which we may know the input variables are fed into it and the predicted outcome, although there is no information on the inner workings of the model, i.e. why a certain prediction was made for an individual patient and which variables were most impactful. This is often the case for highly complex models such as deep neural networks or gradient boosting machines. For these models, usually only a broad "variable importance" metric that described a ranking of the input variables in order of importance can be calculated and should in fact be reported. However, how exactly the model integrated these inputs and arrived at the prediction cannot be comprehended in highly complex models [3]. In contrast, simpler ML algorithms, such as generalized linear models (GLMs) or generalized additive models (GAMs), which often suffice for clinical prediction modeling, provide interpretability in the form of odds ratios or partial dependence metrics, respectively. Lastly, highly complex models often exhibit poorer calibration out-of-the-box [2].

Consequently, the single final model to be internally or externally validated, published, and readied for clinical use should not only be chosen based on resampled training performance [4]. Instead, the complexity of the dataset (i.e. tabulated patient data versus a set of DICOM images) should be taken into account. Whenever suitable, highly interpretable models such as generalized linear models or generalized additive models should be used. Overly complex models such as deep neural networks should generally be avoided for basic clinical prediction modeling.

Model Discrimination

For a comprehensive assessment of model discrimination, the following data are necessary for each patient in the sample: A true outcome (also called "label" or "true posterior"), the predicted probabilities produced by the model, and the classification result based on that predicted probability (predicted outcome). To compare the predicted outcomes and the true outcomes, a confusion matrix (Table 4.1) can be generated. Nearly all discrimination metrics can then be derived from the confusion matrix.

Area Under the Curve (AUC)

The only common discrimination metric that cannot be derived directly from the confusion matrix is the area under the receiver operating characteristic curve (AUROC, commonly abbreviated to AUC or ROC, also called *c*-statistic). For AUC, the predicted probabilities are instead contrasted with the true outcomes. The curve (Fig. 4.1) shows the performance of a classification model at all binary classification cutoffs, plotting the true positive rate (Sensitivity) against the false positive rate (1—Specificity). Lowering the binary classification cutoff classifies more patients as positive, thus increasing both false positives and true positives. It follows that AUC is the only common discrimination metric that is uniquely not contingent upon the chosen binary classification cutoff. The binary classification cutoff at the top left point of the curve, known as the "closest-to-(0,1)-criterion," can even be used to derive an optimal binary classification cutoff, which is explained in more detail further on [5]. Models are often trained and selected for AUC, as AUC can give a relatively broad view of a model's discriminative ability. An AUC value of 1.0 indicates perfect discrimination, while an AUC of 0.5 indicates a discriminative performance not superior to random prediction. Usually, a model is considered to perform well if an AUC of 0.7 or 0.8 is achieved. An AUC above 0.9 indicated excellent performance.

Table 4.1 A confusion matrix

	Negative label	Positive label
Predicted Negative	800 (*True Negative*)	174 (*False Negative*)
Predicted Positive	157 (*False Positive*)	869 (*True Positive*)

Accuracy

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{P + N}$$

Based on the confusion matrix, a model's accuracy equals the total proportion of patients who were correctly classified as either positive or negative cases. While accuracy can give

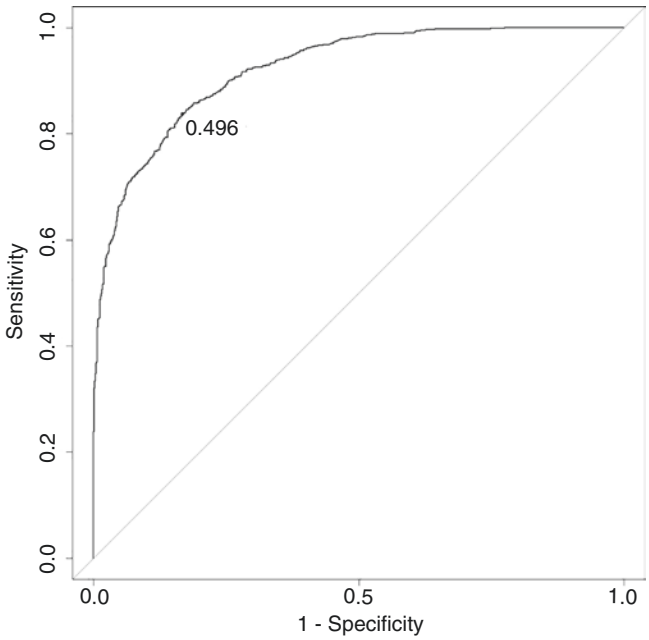

Fig. 4.1 Area under the receiver operating characteristic curve (AUC) plot demonstrating an AUC of 0.922. The plot also indicated that, according to the "closest-to-(0,1)-criterion", 0.496 is the optimal binary classification cutoff that balances sensitivity and specificity perfectly

a broad overview of model performance, it is important to also consider sensitivity and specificity, as accuracy can be easily skewed by several factors including class imbalance (a caveat discussed in detail later on). An accuracy of 100% is optimal, while an accuracy of 50% indicates a performance that is equal to random predictions. The confusion matrix in Table 4.1 gives an accuracy of 83.5%.

Sensitivity and Specificity

$$\text{Sensitivity} = \frac{\text{TP}}{P}$$

$$\text{Specificity} = \frac{\text{TN}}{N}$$

Sensitivity denotes the proportion of patients who are positive cases and who were indeed correctly predicted to be positive. Conversely, specificity measures the proportion of patients who are negative cases, and who were correctly predicted to be negative. Thus, a prediction model with high sensitivity generates only few false negatives, and the model can be used to "rule-out" patients if the prediction is negative. A model with high specificity, however, can be used to "rule-in" patients if positive, because it produces only few false positives. In data science, sensitivity is sometimes called "recall." The confusion matrix in Table 4.1 gives a sensitivity of 83.3% and a specificity of 83.6%.

Positive Predictive Value (PPV) and Negative Predictive Value (NPV)

$$\text{PPV} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{NPV} = \frac{\text{TN}}{\text{TN} + \text{FN}}$$

PPV is defined as the proportion of positively predicted patients who are indeed true positive cases. Conversely, NPV is defined as the proportion of negatively predicted patients who turn out to be true negatives. PPV and NPV are often said to be more easily clinically interpretably in the context of clinical prediction modeling than sensitivity and specificity, as they relate more directly to the prediction itself: For a model with a high PPV, a positive prediction is very likely to be correct, and for a model with a high NPV, a negative prediction is very likely to be a true negative. In data science, PPV is sometimes called "precision." The confusion matrix in Table 4.1 gives a PPV of 84.7% and a NPV of 82.1%.

F1 Score

$$\text{F1} = 2 \times \frac{\text{PPV} \times \text{Sensitivity}}{\text{PPV} + \text{Sensitivity}}$$

The F1 score is a composite metric popular in the ML community, which is mathematically defined as the harmonic mean of PPV and sensitivity. Higher values represent better performance, with a maximum of 1.0. The F1 score is also commonly used to train and select models during training. The confusion matrix in Table 4.1 gives a F1 score of 0.840.

Model Calibration

Calibration Intercept and Slope

As stated above, calibration describes the degree to which a model's predicted probabilities (ranging from 0% to 100%) correspond to the actually observed incidence of the binary endpoint (true posterior). Especially for clinically applied models, a well-calibrated predicted probability (e.g. your predicted probability of experiencing a complication is 18%) is often much more valuable to clinicians and patients alike than a binary prediction (e.g. you are likely not going to experience a complication) [1]. A quick overview of a model's calibration can be gained from generating a calibration plot (Fig. 4.2), which we recommend to include for every published clinical prediction model. In a calibration plot, the patients of a certain cohort are stratified into g equally-sized groups ranked according to their predicted probabilities. If you have a large cohort available, opt for $g = 10$; if you have only few patients you may opt for $g = 5$ to smooth the calibration curve to a certain degree. On the y axis, for each of the g groups, the observed proportion of positive cases is

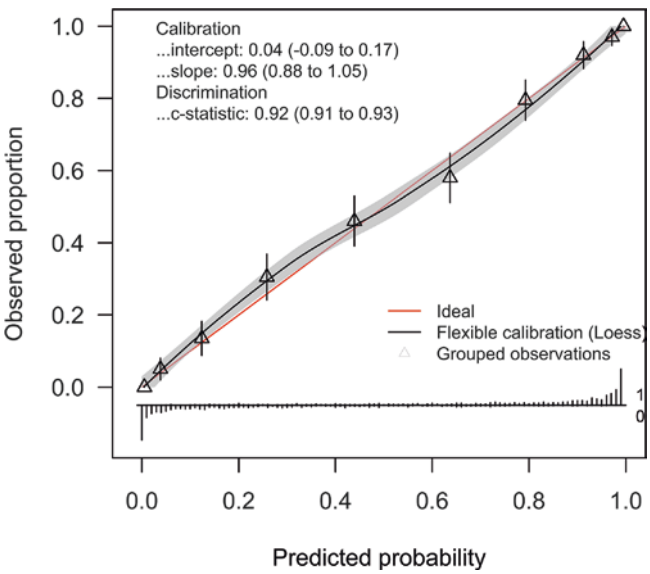

Fig. 4.2 Calibration plot comparing the predicted probabilities—divided into ten bins—of a binary classification model to the true observed outcome proportions. The diagonal line represents the ideal calibration curve. A smoother has been fit over the ten bins. This model achieved an excellent calibration intercept of 0.04, with a slope of 0.96