Proposal for Reporting Items in Machine Learning Evaluation (PRIME) Guidelines for Cardiovascular Imaging Endorsed by the American College of Cardiology Health Care Innovation Council

Writing Group Members:

Partho P Sengupta, MD, DM, FACC, FASE, Chair [1]
Sirish Shrestha, MSc, Vice Chair [1]
Béatrice Berthon, PhD [2]
Emmanuel Messas, MD [3]
Erwan Donal, MD [4]
Geoffrey Tison, MD, PhD [5]
James K Min, MD [6]
Jan D'hooge, MD [7]
Jens-Uwe Voigt, MD [7]
Joel Dudley, PhD [8,9]
Johan Verjans, MD, PhD [10, 11]
Khader Shameer, PhD [8,9]
Kipp Johnson, PhD [8,9]
Lasse Løvstakken, PhD [12]
Mahdi Tabassian, PhD [7]
Marco Piccirilli, PhD [1]
Mathieu Pernot, PhD [2]
Nicolas Duchateau, PhD [13]
Nobuyuki Kagiyama, MD, PhD [1]
Olivier Bernard, PhD [13]
Piotr Slomka, PhD [14]
Rahul Deo, MD, PhD [5]
Rima Arnaout, MD [5]

*Document reviewed and approved by Jai K. Nahar, MD, MBA and James E. Tcheng, MD on behalf of the American College of Cardiology Health Care Innovation Council

[1]West Virginia University Heart & Vascular Institute, Division of Cardiology, Morgantown, WV, USA; [2]Physique pour la Médecine Paris, Inserm U1273, CNRS FRE 2031, ESPCI Paris, PSL Research University, Paris, France; [3]Université Paris Descartes, Sorbonne Paris Cité, Paris, France; [4]Département de Cardiologie et Maladies Vasculaires, Service de Cardiologie et maladies vasculaires, CHU Rennes, Rennes, France; [5]Division of Cardiology, Department of Medicine, University of California San Francisco, San Francisco, CA, USA; [6]Cleerly, Inc. New York, New York; [7]Laboratory on Cardiovascular Imaging & Dynamics, Department of Cardiovascular Science, KU Leuven, Leuven, Belgium; [8]Department of Genetics & Genomic Sciences and Institute for Genomics and Multiscale Biology, Icahn School of Medicine at Mount

Sinai, New York, NY, USA; [9]Institute for Next Generation Healthcare, Department of Genetics and Genomic Sciences, Icahn School of Medicine at Mount Sinai, New York, NY, USA; [10]South Australian Health and Medical Research Institute, Adelaide, SA, Australia; [11]Australian Institute for Machine Learning, University of Adelaide, North Terrace, Adelaide, SA, Australia; [12]Department of Circulation and Medical Imaging, Faculty of Medicine and Health Sciences, Norwegian University of Science and Technology, Trondheim, Norway; [13]CREATIS, CNRS UMR 5220, INSERM U1206, Université Lyon 1, INSA-LYON, France; [14]Department of Imaging and Medicine, Cedars-Sinai Medical Center, Los Angeles, CA, USA

Disclosures:

Mr. Shrestha has nothing to disclose

Dr. Arnaout has nothing to disclose

Dr. Bernard has nothing to disclose

Dr. Berthon has nothing to disclose

Dr. D'hooge has nothing to disclose

Dr. Deo has nothing to disclose

Dr. Donal has nothing to disclose

Dr. Duchateau has nothing to disclose

Dr. Dudley has nothing to disclose

Dr. Johnson has nothing to disclose

Dr. Kagiyama has nothing to disclose

Dr. Lovstakken is a consultant for GE Vingmed Ultrasound AS

Dr. Messas has nothing to disclose

Dr. Min has nothing to disclose

Dr. Pernot has nothing to disclose

Dr. Piccirilli has nothing to disclose

Dr. Khader has nothing to disclose

Dr. Slomka has nothing to disclose

Dr. Tabassian has nothing to disclose

Dr. Tison has nothing to disclose

Dr. Verjans has nothing to disclose

Dr. Voigt has nothing to disclose

Dr. Sengupta is a consultant for Heartsciences., Hitachi Aloka Ltd.

Word count (Body): 7303; (Document): 11062

Address for correspondence:

Partho P. Sengupta, MD, DM

Heart & Vascular Institute,

West Virginia University,

1 Medical Center Drive,

Morgantown, WV 26506-8059.

E-mail: Partho.Sengupta@wvumedicine.org

Abstract

Machine learning (ML), a subset of artificial intelligence (AI) wherein computational algorithms acquire information by adapting their behavior to (large) databases, has been increasingly used within cardiology, particularly in the domain of cardiovascular imaging. Due to the inherent complexity and flexibility of ML algorithms, inconsistencies in the model validity and interpretation may occur. The goal of the PRIME guidelines is standardizing the application of AI and ML methods to allow the consistent and reproducible reporting of cardiovascular imaging study results. There have been several review articles recently published introducing basic principles of ML for general cardiologists. The PRIME guidelines is addressed towards investigators, data scientists, authors, editors, and reviewers involved in machine learning research with the intent of standardization. A multidisciplinary panel of ML experts, clinicians, and traditional statisticians were invited to contribute to a set of guidelines that consists of a list of pragmatic steps for reducing algorithmic errors and biases. Finally, the document provides a list of reporting items to be included to enable the correct application of ML models and the consistent reporting of model specifications and results in the field of cardiovascular imaging.

Keywords: Machine Learning, Artificial Intelligence, Guidelines, Cardiovascular imaging

In 2016, during a two day Think Tank meeting, The American College of Cardiology's Executive Committee and Cardiovascular Imaging Section Leadership Council initiated a discussion regarding the future of cardiovascular imaging among thought leaders in the field (1). One of the goals was focused on machine learning (ML) tools and methods that allow us to go beyond a mere description of the data. This document stresses the creation and adoption of standards, the development of registries, and the use of new techniques in bioinformatics. The imaging community's unfamiliarity with the approach was cited as a potential barrier. Since then, the field has seen a growing interest in the use of ML algorithms, demonstrating powerful and axiomatic algorithms that can analyze large and complex data; in contrast to conventional statistical methods, these algorithms deliver stellar performances and can be rapidly deployed. A standard protocol from modalities such as echocardiography, cardiac magnetic resonance imaging (MRI), cardiac computed tomography (CT), and nuclear imaging acquires large numbers of images and produces numerous multidimensional clinical and functional parameters, including quantitative variables, which have strengths and weaknesses (2). The availability of such rich information has the potential to advance cardiovascular imaging research and its translation into disease prediction (3). In conjunction with big data – that is robust, accurate, and clean – ML has the potential to mitigate missed diagnoses, reduce false-positives and other errors, and deliver precision medicine with individualized care (4–7).

Several recent state-of-the-art review articles have focused on providing introductory concepts regarding ML algorithm applications for general cardiologists (4, 8–10). While ML is creating headlines in medical journals, congress, and on the web, considerable uncertainty and debate have arisen around topics such as problems with real-world data sources, the inconsistent

availability of labeled data and outcome information, bias injection, inaccurate measurements, reproducibility, lack of external validation, and insufficient reporting, which contribute to hindering the reliable assessment of prediction model studies and reliable interpretations of the results by clinicians. The Proposal for Reporting Items in Machine Learning Evaluation (PRIME) Guidelines for Cardiovascular Imaging aims to provide general framework as reference in guiding scientific work for investigators, data scientists, authors, editors, and reviewers involved in machine learning research in cardiovascular imaging. The goal of the PRIME guidelines document is to standardize the application of artificial intelligence (AI) and ML, including data preparation, model selection, performance assessment, to allow the reporting of consistent and reproducible results in cardiovascular imaging studies.

Designing the study plan

Determining the appropriateness of machine learning to the dataset

The first question researchers should address is whether the ML approach could be applicable and beneficial to their study. There is overlap between traditional statistics and ML, but they differ regarding the extent of the assumptions and the formulation of the methods to either predict or make inferences. Although their task is performed by learning the patterns in the data, the model created using ML closely represents the behavior of the data via one of the following learning methods: supervised, unsupervised, or reinforcement learning. If the dataset used to train a complex model is relatively small, then overfitting becomes much more difficult for ML to avoid (11). In such cases, simple models with limited features, including traditional statistical analyses, may provide better insight, performance, and interpretability(12). Similarly, if variables

that are important to modeling the data are missing, the ML model may underfit the data, thus producing less than optimal results (11). Furthermore, ML may benefit situations where the data are unstructured or exploratory analyses are preferred. As such, the learning algorithm may find patterns in the data to generate a homogenous faction and identify relationships in a data-driven manner beyond the a priori knowledge or existing hypotheses (4).

The abundant data may provide better performance using advance learning algorithms but at the cost of the interpretability, complexity, and the ability to draw a causal inference. Caution should be taken against causally interpreting results derived from models designed primarily for prediction. For tasks where the goal is to establish causality, the techniques that are commonly used in "traditional" biostatistics, including statistical analyses methods such as propensity score matching, or Bayesian inference, maybe better suited; however, newer methods involving ML algorithms are being developed for causal inferences (13, 14).

Understanding and Describing the Data

Irrespective of whether ML tools or biostatistical analysis methods are used, it is crucial to understand and describe the data available for analysis to draw appropriate conclusions, whether it is tabular, images, time-series data, or a combination. Important considerations about the data include the availability of data that is representative of the target population, the method used to obtain data, and the resultant biases that may influence the conclusions that can be drawn from the data. Describing the data can also help understand the relevance to the target population. The method of data collection, including the sampling method, is also important, as bias may be introduced from systematic error, coverage error, or selection bias. Various guidelines and associated checklists for medical research have been established to aid in the reporting of relevant details about the data, depending on the study design (15). Clearly describing the data

preprocessing or data cleaning methods used is essential to enable reproducibility and validation of the results.

It should be acknowledged that all ML or statistical algorithms are guided by basic data assumptions; an independent and identical distribution is an important assumption where the random variables are mutually independent and have the same statistical distribution and properties. Methods to check for the model assumptions, such as learning curve(16), diagnosing bias and variance(17) or error analyses, may be required. These assumptions may be violated in cardiovascular imaging research; however, some corrections and methods can be applied to confirm that the proper assumptions are applied (18–21).

Defining the process

When building ML models, it is crucial to specify the inputs (e.g., pixels in images, a set of parameter values, and patient information), labels (e.g., object categories and the presence or absence of disease), and desired outputs (an integer representing each category, the probability for each category, the prediction of a continuous outcome measurement, transformed pixel data) that are required. The optimal tuning of hyperparameters can often lead to higher accuracy in the appropriate model. While this is essential for supervised learning approaches, unsupervised learning approaches may also benefit from defining the output that is desired for the task to select an appropriate model. Some tasks, once well-defined, can only be achieved using certain types of algorithms. For example, image recognition tasks from raw pixel/image data may require the extraction of the optimal features from the data, which is intrinsically performed by deep neural networks and goes beyond the use of hand-crafted features as input. It is generally

preferable to start by selecting the simplest model necessary to accomplish the task of interest. Defining the problem or task as precisely as possible can also help to guide the data annotation strategy. Once the data analysis objective has been identified and the inputs/outputs have been defined for each task, it is easier to determine the appropriate models for the analysis pipeline (Figure 1).

Summary: Defining the goal of the analysis is a key first step that informs many downstream decisions as to whether to use machine learning at all or to incorporate data labelling and can alter the approach to model training, model selection, development, and tuning. Similarly, in order to train machine learning models, each task in the process should be narrowly defined; if necessary, the overall broader analysis goal should be divided into smaller tasks. It is important to thoroughly understand and describe the data that will be used for training to accurately represent the target population and to identify potential biases that may affect the conclusions or performance.

Recommendations:

- Identify and assess if machine learning could be appropriate.
- Define the pipeline necessary to achieve the overall goal.
- Understand and describe the data.
- Identify input and target variables.
- Describe the baseline data and understand biases that may exist.

Data standardization, feature engineering and learning

Data format

To analyze the data of *N* patients (also called 'observations'), each with *M* different measurements (also called 'variables' or 'dimensions'), e.g., ejection fraction, body mass index (BMI), and image pixels/voxels, by using an ML algorithm, a matrix *X* should be first constructed such that the rows of this matrix correspond to the observations and the columns correspond to the variables (Figure 2). Depending on the database and the problem at hand, *X* can be either a 'wide' (Figure 2a) or a 'tall' (Figure 2b) matrix. In the former case, the number of observations is much smaller than the number of variables ($N << M$), while in the latter, there is a large group of observations, but each observation has only a few variables ($N >> M$).

Generating a data matrix from cardiac images can be performed in two main ways, depending on the learning purpose. When the goal is to use a learning algorithm for modeling the global characteristics of the images, all the pixels of an image are considered to be the elements of one observation, which typically leads to a wide data matrix. To model regional image characteristics, however, a region of interest (ROI) or patch consisting of a small group of pixels is considered to be an observation, thus yielding a tall data matrix. Examples of wide and tall data matrices made by image pixels as variables are shown in Figures 2c and 2d, respectively. In Figure 2c, each image was considered an independent observation and its pixels formed the variables. Given that the number of variables is much larger than the number of observations, the resulting data matrix is wide. In Figure 2d, on the other hand, an ROI with 9 pixels was considered to be an observation resulting in a tall data matrix because of the many ROIs taken from the images and the small number of pixels (i.e., 9) that serve as the variables.

Data preparation

To analyze cardiac images in an ML framework, some preprocessing stages are usually carried out. The irrelevant areas of the images can be removed in a 'cropping' stage to focus on learning from useful regions and to prevent learning from extraneous regions (which can also contribute to leakage, as discussed below). If the images that are acquired from a group of subjects have different sizes, they typically should be first 'resized' (22) to a reference image size to construct a data matrix with the same number of variables. More advanced techniques from computational atlases are also necessary to align the anatomy-based data of each subject to a common geometry and temporal dynamics (23, 24). Another common preprocessing stage is 'noise removal', which helps a learning algorithm to better model the essential characteristics of the images. When the acquired images have poor contrast, a 'histogram equalization' (22) technique can be used to adjust the intensities of the pixels and to increase the contrast of a low contrast region, thus facilitating its interpretation and analysis. The pixel intensities can also be manually adjusted during image acquisition. An example is the changing of the dynamic range of echocardiographic images by an operator.

Feature engineering and learning

The next stage after data preparation is extracting a set of 'features' from the data matrix to be later used as the input to the learning methods. Feature extraction helps to overcome the following two main problems that can limit the efficient performance of a learning framework:

(i) Curse of dimensionality: When the data matrix is wide, the variable/feature space of the data is called 'high-dimensional'. This feature may lead an algorithm failing to learning essential

characteristics of the data due to its complexity and poor generalization power when dealing with unseen data — a phenomenon that is referred to as the 'curse of dimensionality' (25, 26). To tackle these problems, the number of observations should increase significantly with the data dimensionality (27). However, a significant increase in the number of observations is not always possible, especially for medical data/studies, given that it necessitates the collection of data from a large group of patients. This curse of dimensionality is one of the main reason why having a tall database is desirable to build an efficient learning algorithm.

(ii) Correlated variables*:* When a database includes correlated variables, a subset of the variables that are mutually uncorrelated may be sufficient to learn the data characteristics effectively (25). Indeed, adding correlated variables to a database may only bring redundant information and would not help the learning algorithm to achieve a better understanding of the data. An example could be using BMI along with weight and height as variables. Given that BMI is comprised of weight and height, using BMI alone could lead to the same performance as using all three variables. For the image data, neighboring pixels typically have similar values and are highly correlated (28).

Considering the problems associated with the curse of dimensionality and given that increasing the number of observations could be a challenging task, using the 'feature extraction' technique to discard the variables (i.e., dimensions) that do not carry relevant information and employing learning techniques that also reduce the dimensionality of wide data are efficient solutions for handling the learning problem (25–27). The result of the feature extraction process should be a compact set of (potentially uncorrelated) features in the form of a tall matrix that encodes the essential characteristics of the data.

The available approaches for extracting features from the image data can be divided into the following three broad categories (Figure 3): (i) handcrafted methods (e.g., local binary patterns

(LBP) (29) and scale invariant feature transform (SIFT) (30)), (ii) classic ML methods for dimensionality reduction (e.g., principal component analysis (PCA) (31), independent component analysis (ICA) (32), and ISOMAP (33)) and (iii) deep learning methods (34). The methods of the first category are manually designed to extract specific types of features from the data, while in the second category of methods, the features are learned from the database itself. Nevertheless, the classical feature learning algorithms have some limitations in the data modeling approaches like linearity, sparsity, or lack of hierarchical representation. The deep learning techniques, on the other hand, can learn complex features from the data at multiple levels and do not have limitations of the classical algorithms. However, they need a large-scale database to achieve efficient learning of the data characteristics. To train a deep learning algorithm with a smaller database, the following two main strategies can be used: (i) data augmentation (e.g., by using different types of data/image transformations) (35) and (ii) transfer learning, which works by fine-tuning a deep network that has been pretrained with a different large database (e.g., natural images) (20, 35).

Variable normalization

For a database that is composed of several variables of different nature (e.g., anthropometric or imaging-derived measurements), the values of the variables lie in different ranges. Direct usage of these variables may bias the learning system towards the characteristics of the variables with larger values despite the usefulness of the variables with smaller values in solving a given problem. To deal with such challenges, a 'variable normalization' approach can be used to transform the variables such that they all lie in the same range prior to entering the learning phase (26, 27). Variable normalization is especially helpful for a deep learning algorithm, as it helps achieve faster convergence of a deep neural network (36).

Missing variable estimation

A common issue that a learning method can face is the absence of some of the variables for a subset of the observations. Although these observations can be simply excluded from the analyses, the performance of the learning method could be degraded due to having a smaller database. An alternative is to use a 'data imputation' technique to estimate the missing variables of an observation. With this technique, similarities are sought between the available values of an incomplete observation and those of complete observations in the dataset from which the missing values are estimated (20, 37, 38). In cardiovascular imaging, 2D images are normally collected from multiple views, e.g., for volumetric measurements, and 3D images are composed of multiple 2D slices. These images can also be acquired throughout the cardiac cycle. When some of the 2D views are not accessible or when a group of 2D images at some points during the cardiac cycle or in a 3D volume are artifactual/missing, an imputation technique can estimate these images or the parameters extracted from them (38). Thanks to the development of the new deep learning algorithms, such as generative adversarial networks (GAN) (39), missing images can often be estimated (40) based on the available data. However, it should be acknowledged that most of the imputation methods assume that the missing observations occur at random, are missing completely at random, or are missing not at random (41). Researchers should consider whether the missing observations carry any specific biases (e.g., selection bias or immortal time bias).

Feature selection

An important phase in designing a classic ML system is to determine the optimal number of preserved features. This determination can be performed by using a 'feature selection' technique

where a larger than required set of features is first extracted and then a subset with discriminative information is selected (27, 42). When a deep learning algorithm is used, the optimal features are automatically learned during the end-to-end training of the algorithm, and utilizing an independent feature selection method is not required (34).

Outliers

An observation is considered as an outlier if its values deviate significantly from the average values of a database, which may be attributed to measurement error, variability in the measurement, or abnormalities due to disease (26, 27). Given that the outliers can negatively influence and mislead the training process, they can result in longer training times and less accurate models. Therefore, the outliers should be carefully removed from the analyses using an outlier detection approach (43). However, as outliers may also carry relevant information about the disease, a learning algorithm that is robust to outliers should be used as an alternative. Examples of methods that are robustness to outliers are decision trees and *k*-nearest neighbor (KNN) (26).

Class imbalance

A significant imbalance in data classes (e.g., healthy vs diseased) is quite common in medical datasets because, on the one hand, the majority of subjects in a database are usually healthy and, on the other hand, because collecting patient data for some rare diseases is difficult and is not always possible. As a result, the performance of the learning algorithm might be skewed, as it only learns the characteristics of the larger sized categories. This problem is referred to as ‘class imbalance’ and can be dealt with in the following three established ways: (i) rebalancing the categories using ‘under-sampling’ or ‘over-sampling’ (i.e., making the different classes similarly

sized by omitting samples from the larger class or by up-sampling the data in the smaller class), (ii) giving more importance (i.e., weight) to the samples of smaller categories during the learning process (27), and iii) utilizing synthetic data generation methods, such as the synthetic minority over-sampling technique (SMOTE) (44), that generate synthetic examples that are similar rather than oversampling by using replacements. Recent advances in deep generative techniques, such as GAN or variational autoencoders (45), have made it possible to tackle complicated imbalanced data based on the learning strategies.

Data shift

Data shift is a common problem that afflicts the ML models in cardiovascular imaging in which the distribution of the database that is used for testing the performance of the learning models or systems may differ from the distribution of the training data. This may occur when the data acquisition conditions or the systems that are used for collecting the test data change from when the training dataset was acquired, and could induce i) a covariate shift – a shift in the distribution in the covariates, ii) a prior probability shift – a difference in the distribution of the target variable, or iii) a domain shift – a change in measurement systems or methods. It is imperative to assess and treat the shifts that may occur in the dataset prior to evaluating a model (46).

Data leakage

Data leakage is a major problem in ML, in which data outside of the training set seeps into the model while building the model. This event could lead to error-prone or invalid ML models. Data leakage could occur if the same patient's data is used in the training and testing sets and is generally a problem in complex datasets, such as time series, audio and images, or graph problems.

Summary

Data preparation and feature extraction is key to the success of model development. It ensures that the data format is appropriate for machine intelligence, the utilized variables carry relevant information for solving the problem at hand and the learning system is not biased towards a subset of the variables or categories in the database.

Recommendation

- The data format for training a machine learning algorithm should be tall and the ratio of the observations/measurements (i.e., N/M) should be at least three.
- When the data matrix is wide, a feature extraction/learning algorithm or dimensionality reduction techniques may be used.
- Multicollinearity should be removed, and variables should be normalized if applicable.
- Missing features and outliers should be addressed accordingly with relevant methods.
- Dataset shift, leakage and class imbalance are common pitfalls and should be assessed and treated as needed.

Selection of Machine Learning Models

Model selection is the process of identifying the model that yields the best resolution and generalizability for the project and can be defined in multiple levels, i.e., learning methods, algorithms, and tuning hyperparameters. Learning methods include supervised, unsupervised, and reinforcement learning. Supervised learning is a method that learns from labeled data, i.e., data with outcome information to develop a prediction model, while unsupervised learning aims to find patterns and rules in data that do not have labels. In contrast, reinforcement learning refers to the ML context, in which an agent learns the optimal action in the environment to gain a maximum reward (Figure 4).

Common algorithms, such as regression or instance-based learning, often handle high-dimensional data well and tend to perform better or equivalent to complex algorithms on small datasets while retaining the interpretability of the model. To achieve better performance, simple algorithms, or weak learners, may be combined in various ways using ensemble methods, such as boosting, bagging, and stacking, which sacrifice the interpretability. More complex algorithms that are also difficult to interpret, including neural networks, can outperform simpler models given an adequate amount of data. A subset of neural networks, known as deep convolutional neural networks, are particularly useful for finding patterns in image data without the need for feature extraction (47–50). The implementation of an algorithm can vary significantly in terms of the size and complexity (e.g., the size and number of features in a random forest decision tree, the number and complexity of kernels applied in an SVM, and the number and type of nodes and layers in a neural network) of the algorithms. Regardless of the choice of the algorithms, it may be imperative to perform hyperparameter tuning and model regularization to produce the optimal performance (51, 52).

These processes may be more important than selecting the types of algorithms that could impact the interpretability, simplicity, and accuracy.

The size and complexity of algorithms should be chosen carefully to minimize the *bias*, the model error on the training dataset, and the *variance*, the model error on the validation dataset. Simpler models may *underfit* the data; they may generalize better (lower variance) at the cost of lower accuracy (higher bias). Further, *overfitting* (high variance and low bias) may come from a too complex model or insufficient representative training samples. Several rules of thumb exist to guide the choice of the initial algorithm size/complexity based on the number of features in the dataset, but the final algorithm design is determined empirically.

Finally, an essential factor in algorithm selection is the need for the interpretability of the model's decisions, i.e., an understanding of which input features caused the model to make the decision it made. Interpretability may be extremely important for certain learning tasks and less important for others. Regression, decision trees, and instance-based learning methods are generally highly interpretable, while methods to interpret the function of deep neural networks are still evolving; saliency mapping, class activation, and attention mapping are some examples of methods for neural network interpretation(53, 54).

Summary:

Design of a model and selection of machine learning algorithm(s) flows directly from the experimental task at hand and the size, complexity, and available labeling of a given dataset. Bias and variance help guide choices of size and complexity of a machine learning algorithm.

Recommendations:

- For the initial model development, select the simplest algorithm that is appropriate for the available data.
- Complex algorithms must be benchmarked to the performance of the initial model across several metrics.
- Tune the hyperparameters to optimize the models and increase performance.

Model assessment

The next step after selecting a learning model is to evaluate the generalizability by applying it to new data, i.e., assessment of its performance on unseen data. Ideally, model assessment should be performed by randomly dividing the dataset into a 'training set' for learning the data characteristics, a 'validation set' for tuning the parameters of the learning model, and a 'test set' for estimating its generalization error, where all the three sets have the same probability distribution (i.e., the statistical characteristics of the data in these three sets are identical). However, in many domains, including cardiovascular imaging, having access to a large dataset is often difficult, thus preventing model assessment using three independent data subsets. As mentioned in the previous section, the ratio of the training samples to the number of measured variables should be at least *three* (27) to learn the data characteristics properly. If this criterion is not met, the data are called scarce. In this situation, the data may be divided into two subsets for

training and final validation of the learning algorithm. However, the results may depend on the random selection of the samples. Therefore, the training set can then be further partitioned into two subsets, but this process is repeated several times by selecting different training and testing subjects to obtain a good estimate of the generalization performance of the learning algorithm (26). This method of model assessment can be performed via 'cross-validation' or 'bootstrapping', as further explained below. These techniques ensure that (i) the learning model is trained properly given that the majority of the data samples can be used in the training process, (ii) the learning model is not biased towards the characteristics of a subset of the data and (iii) the optimal values of the hyperparameters of the learning model (e.g., the number of layers in a neural network and the neurons in each layer) can be determined (26).

Cross-validation

This technique works by dividing the data into multiple nonoverlapping training and testing subsets (also called folds) and using the majority of the folds for training a learning model and the remaining folds for evaluating its performance (25–27). The cross-validation process can be implemented in one of the following ways.

i. **k-fold cross-validation***:* The data is randomly partitioned into *k* folds of roughly equal sizes, and in each round of the cross-validation process, one of the folds is used for testing the learning algorithm and the rest of the folds are used for its training (Figure 5). This process is repeated *k* times such that all folds are used in the testing phase and the average performance on the testing folds is computed as an unbiased estimate of the overall performance of the algorithm (25, 26).

ii. **Leave-one-out cross-validation***:* In this technique, the number of the folds is equal to the number of the observations in the database, and in each round, only one observation is used for testing the learning algorithm.

iii. **Monte-Carlo cross-validation***:* In this method of cross-validation, there is no limit to the number of the folds, and a database can be randomly partitioned into multiple training and testing sets. The training samples are randomly selected 'without replacement', and the remaining samples are used for the testing group (Figure 6 I) (55).

Bootstrapping

This method works by randomly sampling observations from a database 'with replacement' to form a training set whose size is equal to the original database. As a result, some of the observations can appear several times in the training set, while some may never be selected. The latter observations are called 'out-of-bag' and are used to test the learning algorithm. This process is repeated multiple times to estimate the learning method's generalization performance (Figure 6 II) (26, 55).

Summary

The basic concept of training and evaluating an ML model is to split the data into the subset where model is trained (training set), the subset where it is evaluated and tuned (validation set), and the holdout set, i.e., where the performance of the established model is tested (test set). When data is scarce, the model assessment should be performed by creating multiple training and testing sets from the database to obtain a good estimate of the generalization performance of the learning algorithm. The two main techniques that can be used for this purpose are cross-validation and bootstrapping.

Recommendations

- The size of the dataset and the complexity of the employed learning algorithm should be considered to achieve a good compromise between 'bias' and 'variance' in the estimations.
- Bootstrapping yields a lower variance in the performance estimation than cross-validation but at the expense of a higher bias for small databases.
- Typical numbers for *k* in a *k*-fold cross-validation are 5 and 10.
- Leave-one-out cross validation is an appropriate choice when the data is small.

Model Evaluation

The reporting of accuracy in ML is closely linked to the reporting of summary statistics, and the same background and assumptions apply. While a review of statistical theory is out of scope for the PRIME guidelines, we encourage the readers to obtain a clear understanding of the statistics for classification and prediction (56–62). Most of the following section applies to supervised learning algorithms, for which labels are used in the definition of the performance measures. Unsupervised learning is more difficult to evaluate but should also evaluate the relevance of the output data representation and the stability of the results against the data and model parameters.

For classification tasks, the accuracy is the percentage of data that is correctly classified by the model, which could be influenced by the quality of the expert annotations. The balance of classes in the training data is also a known source of bias. As such, a prerequisite for reporting accuracy measures is to provide a clear description of the data material used for training and validation.

We further recommend balancing the class data according to prevalence when possible, or that balanced accuracy measures are reported (63).

The model parameters (e.g., initialization scheme, number of feature maps, and loss function), regularization strategies (e.g., smoothness and dropouts), and hyperparameters (e.g., optimizer, learning rate, and stopping criterion) also play a part in the model performance. A second prerequisite is, therefore, to provide a clear description of how the ML model was generated. We further recommend that the certainty of the accuracy measure is reported where applicable, for instance, by estimating the ensemble average and variance from several models generated with random initialization. Additionally, cross-validation analysis should be added to underline the robustness of the model, especially for limited training and test data (see the previous section). Furthermore, to assess the generalizability of the algorithm, it is recommended to report the accuracy of the model by testing the data from different geographical locations with similar statistical properties and distributions (64).

A report of the accuracy for ML algorithms in cardiovascular imaging will depend on the method and problem. For instance, the classification of disease from image features differs from the classification of image pixels in semantic segmentation, both in terms of the measures reported and of the risk in use.

For multiclass/label classification, we recommend using a statistical language close to the clinical standard. For instance, the report sensitivity, specificity, and odds ratio should be used instead of the precision, recall, and F1 score. This will also ensure that true negative outcomes are considered (65). Nonetheless, for classification tasks, the confusion matrix should normally be included but could be supplementary material. For image segmentation problems, we recommend reporting several measures to summarize both the global and local deviations, such

as the mean absolute error (MAE), the Dice score to summarize the average performance and the Hausdorff distance metric to capture local outliers.

When the output of the regression or segmentation algorithms are linked to clinical measurements (e.g., ejection fraction), we recommend Bland-Altman plots as for conventional evaluation of the image measurements, and we stress the importance of comparing the performance with several expert observers for both intra- and inter- expert variability.

For the classification of disease from image features, the cost of misclassification should be clearly conveyed, e.g., rare diseases may not be properly represented in the dataset. The balance of classes should reflect the prevalence of the disease of interest, and scoring rules based on estimated probability distributions should be used for the accuracy reporting when possible, instead of direct classification. The choice of the scoring rule used for the decision, e.g., mean squared error, Brier score, and log-loss, should be rationalized. The common classification scores (sensitivity, specificity) should include a full ROC analysis to provide a more in-depth evaluation of the detection performance. It is also relevant to include benchmark results from alternative ML methods as well as more traditional techniques, such as logistic regression.

Summary:

For classification problems in cardiovascular imaging, we require a clear description of the data and the machine learning setup. The certainty of the accuracy measure (e.g., variance) should be estimated and reported, and ten-fold cross-validation should be used to validate the model. Use a statistical language close to the standard in medicine and add ROC analysis where applicable. For the measurements, include Bland-Altmann plots and inter-/intra-observer reference values. Use relevant scoring rules instead of direct classification where possible.

Recommendations:

- Provide a clear description of the data used for training, validation and testing and a summary of the model parameters and training setup.
- Use a statistical language close to the clinical standard and introduce new measures only when needed.
- Balance the classes according to prevalence where available or report balanced accuracy measures.
- Estimate the accuracy certainty, e.g., from an ensemble of models, to strengthen the confidence in the values reported.
- Include Bland-Altman plots when machine learning is linked to clinical measurements.
- Include an inter-/ intra-observer variability measures as a reference where possible.
- The risk of misclassification should be conveyed, and appropriate scoring rules for decisions may be needed for the classification of a disease.

Software Engineering Best Practices and Data Availability for Reproducibility

The reproducibility of scientific results is essential to make progress in cardiac medicine. The ability to reproduce findings helps to ensure the validity and correctness, as well as enabling others to translate the results into clinically actionable scenarios. However, there are several complementary definitions of reproducibility. We focus here primarily upon *computational reproducibility*; i.e., the ability to independently confirm published results by inspecting and executing data and code. Computational reproducibility is especially important in ML projects, which often involve custom software scripts, the use of external libraries, and intensive or expensive computation. Actions taken at any point in an ML workflow, from quality control and data preparation into suitable data structures to algorithm development to the visualization of results, are often based upon heuristic judgments, and there are potentially numerous justifiable analytic options. Ultimately, these selections may significantly alter the results and conclusions.

The first step for making ML projects reproducible could be the release of all the original code written for a project. Academic research code may be released to the academic community under a permissive open-source license, which allows reviewers and other scientists to utilize and build upon the code in their own projects. Although there are numerous open source licenses available, in most cases, either of two licenses will suffice: the Massachusetts Institute of Technology/Berkeley Software Distribution (MIT/BSD) licenses (https://opensource.org/licenses/MIT) (the MIT and BSD licenses are essentially equivalent) and the GNU General Public License (GNU GPLv3). The MIT/BSD licenses allow published code to

be distributed, modified, and executed freely without liability or warranty; the GNU GPLv3 license allows the same with the additional restriction that all software-based upon the original code must also be freely available under the GNU GPLv3 license, meaning others cannot reuse the original code in a closed-source product. Researchers who wish to commercialize their code or software projects are free to issue their code under any license to commercial, non-academic entities.

There are several options for the publication of code. When possible, we recommend uploading source code with software and packages' version information as supplementary material alongside the manuscript. Other options include permanent archival on a lab website, or per-project archival on commercial or open source and public source code repositories, such as GitHub, Bitbucket, or protocols.io. The use of version control software, such as Git, allows easy inspection and auditing of the progress of algorithm development. Manuscripts should explicitly state where and how the code may be downloaded and under what license. All software must come with a license.

Although the availability of code is required for computational reproducibility, equally important is the availability of the data used in the project. Clinical data should be anonymized, or if anonymization is not possible (as in the case of some genetic data), then data should be made available to other researchers with appropriate IRB approval. Other options include the generation of synthetic datasets with the same statistical properties as the original dataset, a field of study called differential privacy. Manuscripts must state where both the raw and

manipulated/transformed data may be obtained and justify any restrictions to data availability. All data should also be accompanied by a codebook (also known as a data dictionary) containing clear and succinct explanations of all variables.

Additionally, it is highly recommended that the original methods and pipelines used in a clinical publication are previously detailed in a technical publication that better justifies the originality and soundness of the technical contribution, while the clinical reviewing process focuses more on the application. Depending on the application, a variety of open challenges with publicly available data are increasingly available, as recently demonstrated for supervised tasks, such as the automatic estimation of ventricular volumes (https://www.kaggle.com/c/second-annual-data-science-bowl) and for left- and right-ventricular automatic segmentation (66). We strongly encourage researchers to benchmark their algorithms on these types of challenges when relevant and available.

Finally, we note that even in the case of freely available data and open-source code, it can be difficult to reproduce the results of published work due to the complexities of software package versioning and interactions between different computing environments. We, thus, recommend that authors make the entire analyses automatically reproducible through the use of software environments (e.g., Docker containers, https://www.docker.com/). Analyses in software containers may be freely downloaded and run from beginning to end by other scientists, greatly improving the computational reproducibility.

Summary

Computational reproducibility of machine learning efforts is essential. Scientists must release their code under an open source license to other scientists. The code should be archived with the journal, on personal websites, or in commercial repositories. The data used in a project should also be made available to the scientific community with minimal restrictions. When possible, the entire analyses should be made reproducible via scripts or containers.

Recommendations

- Use the MIT/BSD or GPLv3 license to release open-source code.
- Upload the code and data as supplementary information alongside the manuscript when possible; otherwise, make the code and data available via an academic website or in commercial repositories.
- Release a codebook (data dictionary) with clear and succinct explanations of all variables.
- Document the exact version of all external libraries and software environments.
- Consider the use of Docker containers or similar fully automated analysis for straightforward computational reproducibility.

Reporting limitations, bias, alternative/additional analyses

"All models are wrong, but some are useful" is a well-known statistical aphorism attributed to George Box. Accurate reporting and acknowledgement of limitations are required for manuscripts incorporating ML (ML). Any statistical model or ML algorithm incorporates some assumptions regarding the data. All model assumptions should be affirmatively identified and checked with the dataset utilized in the manuscript, and the results should be reported in the manuscript or supplementary material. The algorithms used in computational research efforts span a large spectrum of complexity. Generally, more basic models and algorithms should first be investigated before additional complexity is incorporated into models or different algorithms are selected. Deep learning models should be benchmarked against more simplistic models whenever possible, especially when applied to tabular data. Statistical or ML models incorporating large numbers of variables (e.g., polygenic risk score models) should be benchmarked against standard clinical risk prediction models using more traditional clinical variables.

Concordant findings from multiple, independent datasets dramatically increase the scientific value of manuscripts, since it decreases the likelihood that the algorithms have been erroneously overfit to the idiosyncratic features of a certain dataset. Deep learning models are especially notorious for harnessing spurious or confounding features of the dataset to perform well. For example, Zech and Badgeley et al. reported a case where a convolutional neural network trained on a health system's chest X-rays used the presence of a "PORTABLE" label on X-ray images to predict cardiomegaly with high accuracy (67). Furthermore, in the case of supervised ML

involving human-annotated variables or outcomes, it should be noted that ML algorithms will recapitulate the underlying biases of the humans who constructed the dataset.

Summary

Accurate reporting and acknowledgement of limitations are required for manuscripts incorporating machine learning. All models make some assumptions of their data, and these assumptions should be identified and checked. Begin with simpler models over more complex models and justify the use of more complex models. Benchmark models against alternative data sources and obtain external validation in independent datasets.

Recommendations

- Affirmatively identify and check relevant model assumptions and report the findings.
- Benchmark complex algorithms against simpler algorithms.
- Benchmark algorithms incorporating high-dimensional data or novel data sources against other data sources.
- Obtain external validation in independent datasets using the same algorithm.

Summary and Future Directions

As artificial intelligence and ML technologies continue to grow, three specific areas of opportunities will need further consideration for future standardization. First, there has been growing enthusiasm in the use of automated machine learning (auto-ML) platforms that democratize machine-learning strategies. Together with the use of affordable computational resources and cloud computing-based platforms, such auto-ML strategies will reduce the encumbrance on researchers to execute learning algorithms. Second, the competition between and improvement in mobile devices have presented pathways for numerous sensors and physiological biomarkers. Using the 'multiomics-approach', such a data set would need to be integrated with imaging variables and can potentially provide more algorithmic sophistication and objectivity to the existing taxonomy of risk factors and cardiac diseases (68–70). Finally, sophisticated algorithms and variations of GAN will be increasingly used to synthesize data that closely resemble the distribution of the input data (71–73). This approach may be particularly fruitful for the field of simulation and in-silico clinical trials, which were recently recognized by the Food and Drug Administration (FDA) as key new directions to validate novel devices and therapies (74). In this context, recent studies have combined computational modeling with ML for synthetic data generation or tracking a disease course (75). The ML research presents disparate idiosyncratic methods that may be challenging to apply in medicine and software as medical devices, but the integration of intelligent software using ML algorithms is on the verge of restructuring the industry, research, and medical alliance. This fact is well recognized by the FDA, which has produced guidelines that mandate the standardization and applications of software as medical devices (76). With the advancement of organizations and cardiac medicine and imaging towards the actualization of precision medicine, the recommendations provided in

the PRIME guidelines may need to be updated continuously as ML algorithms continue to transform cardiovascular imaging practice over the next decade.

.

References:

1. Douglas PS, Cerqueira MD, Berman DS, et al. The Future of Cardiac Imaging: Report of a Think Tank Convened by the American College of Cardiology. JACC Cardiovasc. Imaging 2016;9:1211–1223.

2. Lang RM, Badano LP, Mor-Avi V, et al. Recommendations for cardiac chamber quantification by echocardiography in adults: An update from the American society of echocardiography and the European association of cardiovascular imaging. Eur. Heart J. Cardiovasc. Imaging 2015;16:1-39.e14. Available at: https://orbi.uliege.be/bitstream/2268/221545/1/jase lang.pdf. Accessed December 17, 2018.

3. Sengupta PP, Shrestha S. Machine Learning for Data-Driven Discovery. JACC Cardiovasc. Imaging 2018:0–2.

4. Dey D, Slomka PJ, Leeson P, et al. Artificial Intelligence in Cardiovascular Imaging: JACC State-of-the-Art Review. J. Am. Coll. Cardiol. 2019;73:1317–1335.

5. Krittanawong C, Zhang HJ, Wang Z, Aydar M, Kitai T. Artificial Intelligence in Precision Cardiovascular Medicine. J. Am. Coll. Cardiol. 2017;69:2657–2664.

6. Zheng T, Xie W, Xu L, et al. A machine learning-based framework to identify type 2 diabetes through electronic health records HHS Public Access. Int J Med Inf. 2017;97:120–127. Available at: http://europepmc.org/backend/ptpmcrender.fcgi?accid=PMC5144921&blobtype=pdf.

7. Dawes TJW, de Marvao A, Shi W, et al. Machine Learning of Three-dimensional Right Ventricular Motion Enables Outcome Prediction in Pulmonary Hypertension: A Cardiac MR

Imaging Study. Radiology 2017;283:381–390. Available at: http://pubs.rsna.org/doi/10.1148/radiol.2016161315.

8. Johnson KW, Soto JT, Glicksberg BS, et al. Arti fi cial Intelligence in Cardiology. 2018;71.

9. Krittanawong C, Zhang HJ, Wang Z, Aydar M, Kitai T. Artificial Intelligence in Precision Cardiovascular Medicine. J. Am. Coll. Cardiol. 2017;69:2657–2664.

10. Al'Aref SJ, Anchouche K, Singh G, et al. Clinical applications of machine learning in cardiovascular disease and its relevance to cardiac imaging. Eur. Heart J. 2019.

11. Ghojogh B, Ca B, Crowley M, Ca M. The Theory Behind Overfitting, Cross Validation, Regularization, Bagging, and Boosting: Tutorial.

12. Dhurandhar A, Luss R, Shanmugam K, Olsen P. Improving simple models with confidence profiles. Adv. Neural Inf. Process. Syst. 2018;2018-Decem:10296–10306.

13. Blakely T, Lynch J, Simons K, Bentley R, Rose S. Reflection on modern methods : when worlds collide — prediction , machine learning and causal inference. 2019:1–7.

14. Leng S, Xu Z. Reconstructing directional causal networks with random forest : Causality meeting machine learning Reconstructing directional causal networks with random forest : Causality meeting machine learning. 2019;093130.

15. Vandenbroucke JP, von Elm E, Altman DG, et al. Strengthening the Reporting of Observational Studies in Epidemiology (STROBE): Explanation and elaboration. Int. J. Surg. 2014.

16. Cohen O, Malka O, Ringel Z. Learning Curves for Deep Neural Networks : A Gaussian Field Theory Perspective. 2019.

17. Mehta P, Bukov M, Wang CH, et al. A high-bias, low-variance introduction to Machine Learning for physicists. Phys. Rep. 2019;810:1–124.

18. Wachinger C, Reuter M. Domain adaptation for Alzheimer's disease diagnostics. Neuroimage 2016.

19. Daumé H, Marcu D. Domain adaptation for statistical classifiers. J. Artif. Intell. Res. 2006.

20. Bengio Y. Deep Learning of Representations for Unsupervised and Transfer Learning. In: Workshop on Unsupervised and Transfer Learning.Vol 27., 2012:17–37.

21. Ganin Y, Lempitsky V. Unsupervised domain adaptation by backpropagation. In: 32nd International Conference on Machine Learning, ICML 2015., 2015.

22. Gonzalez RC, Woods RE. Digital Image Processing (3rd Edition). 2007.

23. Duchateau N, De Craene M, Pennec X, Merino B, Sitges M, Bijnens B. Which reorientation framework for the atlas-based comparison of motion from cardiac image sequences? In: Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics).Vol 7570 LNCS., 2012:25–37.

24. Duchateau N, Craene M De, Piella G, et al. A spatiotemporal statistical atlas of motion for the quantification of abnormal myocardial tissue velocities To cite this version : HAL Id : hal-02282430 Preprint version accepted to appear in Medical Image Analysis . Final version of this paper will be a. 2019.

25. Bishop CM. Pattern recognition and machine learning. Springer; 2006.

26. Hastie T, Tibshirani R, Friedman J. The Elements of Statistical Learning. Springer New York; 2009.

27. Theodoridis S, Koutroumbas K. Pattern recognition. Academic Press; 2009.

28. Hyvärinen A, Hurri J, Hoyer PO. Natural Image Statistics. London: Springer London; 2009.

29. Ojala T, Pietikainen M, Maenpaa T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Mach. Intell. 2002;24:971–987.

30. Lowe DG. Object recognition from local scale-invariant features. In: Proceedings of the Seventh IEEE International Conference on Computer Vision. IEEE, 1999:1150–1157 vol.2.

31. Jolliffe I. Principal Component Analysis. John Wiley & Sons, Ltd; 2002.

32. Hyvärinen A, Karhunen J, Oja E. Independent component analysis. 2001.

33. Tenenbaum JB. A global geometric framework for nonlinear dimensionality reduction. Science (80-.). 2000;290:2319–2323.

34. LeCun YA, Bengio Y, Hinton GE. Deep learning. Nature 2015;521:436–444.

35. Shin H-C, Roth HR, Gao M, et al. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. IEEE Trans. Med. Imaging 2016;35:1285–98.

36. Ioffe S, Szegedy C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In: International Conference on Machine Learning., 2015:1–9.

37. Troyanskaya O, Cantor M, Sherlock G, et al. Missing value estimation methods for DNA microarrays. Bioinformatics 2001;17:520–525.

38. Tabassian M, Alessandrini M, Jasaityte R, De Marchi L, Masetti G, D'hooge J. Handling

missing strain (rate) curves using K-nearest neighbor imputation. In: IEEE International Ultrasonics Symposium (IUS)., 2016:1–4.

39. Goodfellow I, Pouget-Abadie J, Mirza M, et al. Generative Adversarial Nets. Adv. Neural Inf. Process. Syst. 27 2014:2672–2680.

40. Shang C, Palmer A, Sun J, Chen K-S, Lu J, Bi J. VIGAN: Missing view imputation with generative adversarial networks. In: IEEE International Conference on Big Data., 2017:766–775.

41. Trust E. Improving Disparity Research by Imputing Missing Data in Health Care. 2015:939–945.

42. Inza I, Larrañaga P, Blanco R, Cerrolaza AJ. Filter versus wrapper gene selection approaches in DNA microarray domains. Artif. Intell. Med. 2004;31:91–103.

43. Stromberg AJ. Robust Diagnostic Regression Analysis. J. Am. Stat. Assoc. 2002.

44. Chawla N V, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE : Synthetic Minority Over-sampling Technique. 2002;16:321–357.

45. Rezende DJ, Mohamed S. Variational Inference with Normalizing Flows. In: International Conference on Machine Learning., 2015:1–9.

46. Schulam P. Preventing Failures Due to Dataset Shift: Learning Predictive Models That Transport. 2019;89.

47. Wang J, Ding H, Bidgoli FA, et al. Detecting Cardiovascular Disease from Mammograms with Deep Learning. IEEE Trans. Med. Imaging 2017.

48. Litjens G, Ciompi F, Wolterink JM, et al. State-of-the-Art Deep Learning in Cardiovascular Image Analysis. JACC Cardiovasc. Imaging 2019;12:1549–1565.

49. Retson TA, Besser AH, Sall S, Golden D, Hsiao A. Machine learning and deep neural networks in thoracic and cardiovascular imaging. J. Thorac. Imaging 2019;34:192–201.

50. Avendi MR, Kheradvar A, Jafarkhani H. A combined deep-learning and deformable-model approach to fully automatic segmentation of the left ventricle in cardiac MRI. Med. Image Anal. 2016;30:108–119.

51. Kostoglou K, Robertson AD, MacIntosh B, Mitsis GD. A novel framework for estimating time-varying multivariate autoregressive models and application to cardiovascular responses to acute exercise. IEEE Trans. Biomed. Eng. 2019.

52. Al'Aref SJ, Singh G, van Rosendael AR, et al. Determinants of In-Hospital Mortality After Percutaneous Coronary Intervention: A Machine Learning Approach. J. Am. Heart Assoc. 2019;8.

53. Gilpin LH, Bau D, Yuan BZ, Bajwa A, Specter M, Kagal L. Explaining explanations: An overview of interpretability of machine learning. In: Proceedings - 2018 IEEE 5th International Conference on Data Science and Advanced Analytics, DSAA 2018., 2019.

54. Fong RC, Vedaldi A. Interpretable Explanations of Black Boxes by Meaningful Perturbation. In: Proceedings of the IEEE International Conference on Computer Vision., 2017.

55. Kuhn M, Johnson K. Applied predictive modeling. Spriger; 2013.

56. Wheelan CJ. Naked statistics : stripping the dread from the data.

57. Mlodinow L. The Drunkard's walk : how randomness rules our lives. Pantheon Books; 2009.

58. Wasserman L. All of statistics : a concise course in statistical inference. Springer; 2004.

59. Urdan TC. Statistics in plain English.

60. Cohen PR. Empirical methods for artificial intelligence. MIT Press; 1995.

61. Box GEP, Hunter JS, Hunter WG. Statistics for experimenters : design, innovation, and discovery. Wiley-Interscience; 2005.

62. Sabo R, Boone E. Statistical research methods : a guide for non-statisticians. Springer; 2013.

63. Wainer J, Franceschinell RA. An empirical evaluation of imbalanced data strategies from a practitioner's point of view.

64. The Lancet Digital Health. Walking the tightrope of artificial intelligence guidelines in clinical practice. Lancet Digit. Heal. 2019;1:e100. Available at: http://dx.doi.org/10.1016/S2589-7500(19)30063-9.

65. Tharwat A. Applied Computing and Informatics Classification assessment methods. Appl. Comput. Informatics 2018. Available at: https://doi.org/10.1016/j.aci.2018.08.003.

66. Bernard O, Lalande A, Zotti C, et al. Deep Learning Techniques for Automatic MRI Cardiac Multi-Structures Segmentation and Diagnosis: Is the Problem Solved? IEEE Trans. Med. Imaging 2018;37:2514–2525.

67. Zech JR, Badgeley MA, Liu M, Costa AB, Titano JJ, Oermann EK. Variable generalization performance of a deep learning model to detect pneumonia in chest radiographs: A cross-sectional study. PLoS Med. 2018;15:1–17.

68. Natarajan S, Jain A, Krishnan R, Rogye A, Sivaprasad S. Diagnostic Accuracy of Community-Based Diabetic Retinopathy Screening With an Offline Artificial Intelligence System on a Smartphone. JAMA Ophthalmol. 2019.

69. Rajalakshmi R, Subashini R, Mohan R, Viswanathan A. Automated diabetic retinopathy

detection in smartphone-based fundus photography using arti fi cial intelligence. Eye 2018:1138–1144. Available at: http://dx.doi.org/10.1038/s41433-018-0064-9.

70. Valys A, Albert D. Smartwatch Performance for the Detection and Quantification of Atrial Fibrillation. 2019:1–9.

71. You GAN, Gan GAN. GAN You Do the GAN GAN?

72. Wang X, He K, Hopcroft JE. AT-GAN : A Generative Attack Model for Adversarial Transferring on Generative Adversarial Net arXiv : 1904 . 07793v3 [cs . CV] 21 May 2019.

73. Chang C, Yu C, Chen S, Chang EY. KG-GAN : Knowledge-Guided Generative Adversarial Networks. :1–11.

74. Morrison T. How Simulation Can Transform Regulatory Pathways | FDA. Available at: https://www.fda.gov/science-research/about-science-research-fda/how-simulation-can-transform-regulatory-pathways. Accessed October 14, 2019.

75. Kagiyama N, Shrestha S, Farjo PD, Sengupta PP. Arti fi cial Intelligence : Practical Primer for Clinical Research in. 2019:1–12.

76. Anon. Software as a Medical Device (SaMD) | FDA. Available at: https://www.fda.gov/medical-devices/digital-health/software-medical-device-samd. Accessed October 14, 2019.

Figure 1 – Machine learning pipeline

Schematic diagram of a general ML pipeline. The data section consists of project planning, data collection, cleaning, and exploration. The modelling section describes the model building, in which hyperparameter tuning and the dimensionality reduction process, such as feature selection and engineering, model optimization and selection, and evaluation, are included. Finally, the reporting segment consists of the reporting mechanisms of the analysis, including reproducibility and maintenance, and a description of the limitations and alternatives.

Figure 2: Schematic demonstration of wide (a) and tall (b) data matrices and the way that they can be created from the image data. In a wide data matrix, the number of observations is much smaller than the number of variables ($N<<M$). Considering the whole image as one observation and all its pixels as variables may lead to a wide data matrix, as the number of images is typically smaller than the number of pixels (c). To make a tall data matrix from the image data, an image can be divided to many (overlapping) ROIs or patches, each with a small number of pixels (d).

Figure 3: The main approaches for feature engineering and learning. The hand-engineering approaches are manually designed to extract certain types of features from the data. The classic learning techniques use data samples to learn their characteristics, but they have limitations in their data modeling techniques, such as linearity, sparsity or lack of hierarchical representation. Deep learning methods, however, can learn complex features from the data at multiple levels.

Figure 4: Model selection process

Illustration of the model selection process, which consists of identifying the three classes of ML. A) Supervised learning method, in which the data are used for classification or regression. B) Unsupervised learning method, in which the data are either utilized for clustering, topical modeling, or representing the data distribution while reducing the dimensionality of the data according to the problem to be solved. Finally, C) the reinforcement learning technique, in which an agent receives feedback from the environment to adjust the policy with which it learns.

Figure 5: Schematic illustration of the *k*-fold cross-validation process. Data are randomly partitioned into *k* distinct folds, and in each round, (*k*-1) folds are used for training the learning algorithm, and the *k*th fold is used for testing its performance. This process is repeated *k* times such that all folds are used in the testing phase.

Figure 6: I) The process of Monte-Carlo cross-validation, which can be performed in B rounds. In each round, the training and testing samples are randomly selected without replacement from the original data. II) The bootstrapping process, which can be performed in B rounds. In each round, the training data is generated by randomly sampling from the original data with replacement. The samples that are not included in the training set (i.e., out-of-bag samples) form the testing group.