Communications of the Association for Information Systems

Volume 48 Article 46

6-7-2021

How to Conduct Rigorous Supervised Machine Learning in Information Systems Research: The Supervised Machine Learning Report Card

Niklas Kühl
Karlsruhe Institute of Technology (KIT) / IBM, niklas.kuehl@kit.edu

Robin Hirt
Karlsruhe Institute of Technology (KIT) / Prenode

Lucas Baier
Karlsruhe Institute of Technology (KIT)

Björn Schmitz
Karlsruhe Institute of Technology (KIT) / IBM

Gerhard Satzger
Karlsruhe Institute of Technology (KIT) / IBM

Recommended Citation

Kühl, N., Hirt, R., Baier, L., Schmitz, B., & Satzger, G. (2021). How to Conduct Rigorous Supervised Machine Learning in Information Systems Research: The Supervised Machine Learning Report Card. Communications of the Association for Information Systems, 48, pp-pp. https://doi.org/10.17705/1CAIS.04845

Communications of the Association for Information Systems

Research Article DOI: 10.17705/1CAIS.04845 ISSN: 1529-3181

How to Conduct Rigorous Supervised Machine Learning in Information Systems Research: The Supervised Machine Learning Report Card

Niklas Kühl
Karlsruhe Institute of Technology (KIT) / IBM
niklas.kuehl@kit.edu

Robin Hirt
Karlsruhe Institute of Technology (KIT) / Prenode

Lucas Baier
Karlsruhe Institute of Technology (KIT)

Björn Schmitz
Karlsruhe Institute of Technology (KIT) / IBM

Gerhard Satzger
Karlsruhe Institute of Technology (KIT) / IBM

Abstract:

In the last decade, applying supervised machine learning (SML) has become increasingly popular in the information systems (IS) field. However, SML results rely on many different data-preprocessing techniques, algorithms, and ways to implement them, which has contributed to an inconsistency in the way researchers have documented their SML efforts and, thus, the degree to which others can reproduce their results. In one sense, we can understand this inconsistency given the goals and motivations for SML applications vary and the research area's rapid evolution. However, for the IS research community, the inconsistency poses a big challenge because, even with full access to the data, researchers can neither completely evaluate the SML approaches that previous research has adopted or replicate previous research results. Therefore, in this paper, we provide the IS community with guidelines for comprehensively and rigorously conducting and documenting SML research. First, we review the literature concerning steps and SML process frameworks to extract relevant problem characteristics that researchers should report and relevant choices that they should make in applying SML. Second, we integrate these characteristics and choices into a comprehensive "Supervised Machine Learning Report Card (SMLR)" that researchers can use in future SML endeavors. Third, we apply this report card to a set of 121 relevant papers published in renowned IS outlets between 2010 and 2018 and demonstrate how and where these papers' authors could have improved their documentation and, thus, how and where researchers can better document their SML approaches in the future. Thus, with this work, we help researchers more completely and rigorously apply and document SML approaches and, thereby, enable researchers to more deeply evaluate and reproduce/replicate results in the IS field.

Keywords: Supervised Machine Learning, Research Documentation, Research Replication, Methodological Framework.

This manuscript underwent editorial review. It was received 4/16/2020 and was with the authors for four months for one revision. Oliver Müller served as Associate Editor.

1 Introduction

Replicating published research represents an important endeavor in the academic world. Replication studies repeat previously conducted studies in order to investigate whether their findings' reliability and to what extent one can generalize them. Over the last decade, a lack of these methodologically important supplements have constituted the so-called "replication crisis", which reflects that one cannot easily or at all replicate many scientific studies and their results. So far, this replication crisis has particularly been proclaimed in the medicine and psychology fields (Schooler, 2014; Tackett, Brandes, King, & Markon, 2019).

While information systems (IS) research has started to actively incentivize replication studies (Olbrich, Frank, Gregor, Niederman, & Rowe, 2017; Weinhardt, van der Aalst, & Hinz**,** 2019), the rise of methods from machine learning in IS entail new challenges in replication (Coiera, Ammenwerth, Georgiou, & Magrabi, 2018; Hutson, 2018). In particular, supervised machine learning (SML) has gained increasing popularity in the field: between 2010 and 2018, 35 contributions published in *Management Information Systems Quarterly (MISQ*), *Information Systems Research* (*ISR*) and *Journal of Management Information Systems* (*JMIS*) applied SML in their research. In addition, the number of publications in typical IS conferences (European Conference on Information Systems (ECIS) and International Conference on Information Systems (ICIS)) that rely on SML as a key method has steadily grown over time.

While SML enjoys widespread popularity and promises considerable potential in IS research, researchers have room to improve when it comes to rigorously applying these technologies. Many IS research papers do not thoroughly document the SML process and their results, which makes it challenging or virtually impossible to reproduce or replicate their results. Naturally, researchers may prefer discussing SML results' implications rather than stringently documenting the SML process itself. This, however, such a focus will contribute to spread the replication crisis that we describe above also in the IS research community since researchers will neither be able to follow or replicate the precise choices that research makes nor judge their results' meaningfulness. In this paper, we address this problem by developing and testing a documentation standard that will ultimately enable researchers to frequently replicate SML studies in IS. To this end, we review the literature to identify the typical problem characteristics and choices that one should make in SML endeavors. On this basis, we develop a "Supervised Machine Learning Report Card (SMLR)" to provide guidelines for comprehensively and rigorously conducting and documenting SML research. We review the literature concerning extant steps and SML process frameworks and integrate them into a comprehensive report card. Finally, we review 121 relevant papers that renowned IS outlets such as *Management Information Systems Quarterly* (*MISQ*), *Information Systems Research* (*ISR*) and *Journal of Management Information Systems* (*JMIS*) and the proceedings of the International Conference on Information Systems (ICIS) and the European Conference on Information Systems (ECIS) published between 2010 and 2018. We use this broad sample to analyze how and where these papers' authors could have improved their documentation and, thus, how and where researchers can better document their SML approaches in the future. Thus, with this work, we help researchers more completely and rigorously apply and document SML approaches and, thereby, enable researchers to more deeply evaluate and reproduce/replicate results in the IS field.

This paper proceeds as follows: in Section 2, we introduce the fundamentals and positioning. In Section 3, we derive and describe the problem characteristics and key choices of each SML endeavor. We also introduce the supervised machine learning report card (SMLR) to address them. In Section 4, we apply this report card in an empirical study to relevant IS papers and analyze their precision when it comes to SML application and documentation. In Section 5, we discuss our study's limitations, recommendations for future work, and conclude the paper.

2 Fundamentals and Positioning

One can classify machine learning (ML) techniques according to whether they learn in a supervised or unsupervised manner[1] (Mohri, Rostamizadeh, & Talwalkar, 2013). Based on Mohri et al. (2013, p. 5), we define supervised machine learning as "learning a function mapping an input to an output based on labeled training data, i.e. a sample of input-output pairs". In contrast, unsupervised machine learning uses unlabeled data to discover information. Most ML applications adopt supervised learning (Jordan & Mitchell, 2015) whereby they predict an element's (discrete or continuous) value by using an observational

[1] Other sources, such as Fu (2003), also consider reinforcement learning as a third type. However, we lack academic consensus on this classification.

data set in which one already knows the element and labels it with the correct value (Rätsch, 2004). Researchers refer to problems that involve solving discrete target values as classification problems (e.g., determining product returns in e-commerce) (Heilig, Hofer, Lessman, & Voc, 2016). In contrast, they refer to problems that involve predicting continuous variables as regression problems (e.g., forecasting electricity prices). In the latter, the SML algorithm does not output a class but a numerical value that specifies the predicted attribute.

An SML endeavor (i.e., applying SML methods to a problem) may serve different purposes, and its specific design heavily depends on the particular problem. Shmueli and Koppius (2011) differentiate these purposes as either *explaining* or *predicting* a phenomenon. As for explaining a phenomenon, statistical models can support explanatory-oriented research for testing causal hypotheses. For instance, if a researcher wants to *explain* patterns in the data with a linear regression, individual model results (such as the loading of the regression coefficients, the coefficient of determination R^2, or p-values) might already fully warrant applying the model; the researcher would have no further need to evaluate its predictive power on an unseen test or validation set to possibly deploy it in IS artifacts (Gong, Abhishek, & Li, 2017; Li, Chen, & Nunamaker, 2016; Martens & Provost, 2014).

On the other hand, researchers can use *predictive* models to anticipate unseen or future observations. To do so, researchers need to analyze SML's potential to solve an empirical prediction problem. Thus, they need to show its effectiveness in their field studies by reporting on a trained model's predictive qualities. Researchers might compare an SML endeavor to different benchmarks and, consequently, not only show its basic functionality but also the efficiency of leveraging SML for a certain possibly productive task (Pant & Srinivasan, 2010). For instance, they may analyze whether a machine can perform a task better than a human (Han, Otto, Liu, & Jain, 2015). Depending on the scope, this step may even require researchers to implement a predictive model and embed it into a software tool (e.g., to continuously make predictions) (Oroszi & Ruhland, 2010). In this paper, we focus on SML applications for *predictive* purposes.

When we discuss whether one can replicate or reproduce SML studies for predictive purposes, we need to distinguish different possible documentation levels. The spectrum of reproducibility that Peng (2011) originally developed for the computer science field suits our IS SML endeavors. On that basis, we denote the range of options that increasingly allow one to reproduce results in Figure 1. While publications that merely report results do not support reproducibility, publications that expose method details, code, and/or data can support it. He argues for researchers to publish "linked and executable code and data" along with papers themselves as a gold standard to assure reproducibility.

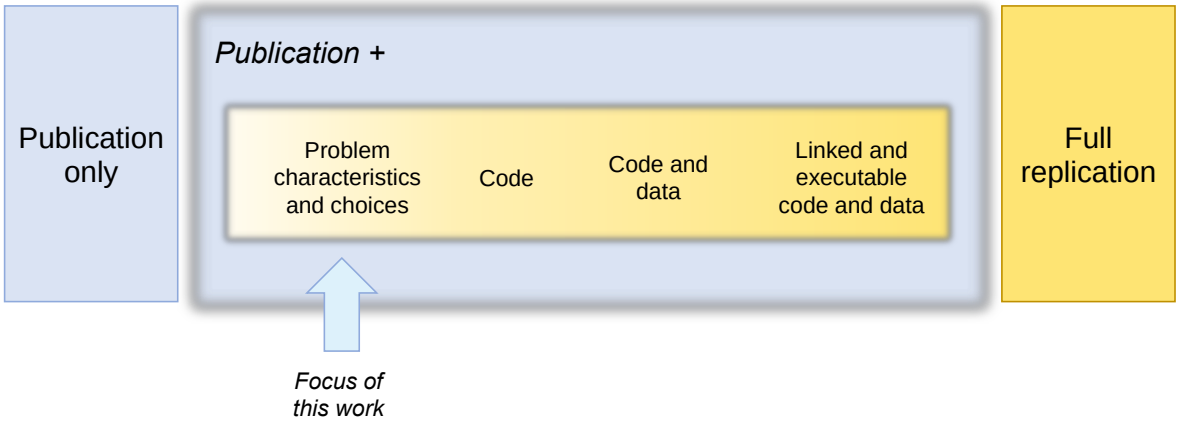

Figure 1. The Reproducibility Spectrum (Based on Peng, 2011)

However, typical IS studies cannot publish code and/or data due to confidentiality issues (Gimpel, Kleindienst, & Waldmann, 2018; Sharp & Babb, 2018; Timmerman & Bronselaer, 2019) at least if they use they do not use publicly available data sources. Therefore, we primarily focus on documenting the problem characteristics and choices that one should make in applying SML but still stress the need to provide code and data whenever possible

When it comes to process models that support SML for predictive tasks, various different possibilities exist. The most common include knowledge discovery in databases (KDD) (Fayyad, Piatetsky-Shapiro, & Smyth, 1996), cross-industry standard process for data mining (CRISP-DM) (Chapman et al., 2000) and

the team data science process (Microsoft, 2020). Although these process models have garnered extreme popularity, they have a broad nature and do not go deep enough to derive measurable criteria for SML endeavors. As their design focuses more on general data mining and machine learning purposes, they lack detail (by design), helpfulness, and transparency for our purposes. The same shortcoming (i.e., a high abstraction level) also applies to other, less popular process models (Anand & Büchner, 1998; Brodley & Smyth, 1995; Cabena, Hadjinian, Stadler, Verhees, & Zanasi, 1998; Cios, Teresinska, Konieczna, Potocka, & Sharma, 2000; Witten, Frank, & Hall, 2011). Since one can apply these process models to any kind of data analysis project and not SML endeavors exclusively given their highly generic nature, they only focus on a limited part of the overall choices and problem characteristics (Kurgan & Musilek, 2006). Furthermore, they do not include precise guidelines for deploying such an SML endeavor or estimating how well one performs, which have particular importance in IS (Shmueli & Koppius, 2011). A process model also does not suitably help one communicate results in a scientific publication.

Therefore, in this paper, we derive problem characteristics and key choices as part of the SMLR; every SML endeavor (and ML replication studies in the IS field) needs to consider and document them to enable readers and reviewers to fully grasp and judge the individual project (Hutson, 2018; Olorisade, Brereton, & Andras, 2017; Voets, Møllersen, & Bongo, 2018). Similarly to the report card that we propose for IS research, researchers have proposed related "checklists" in other fields with the idea to append them when submitting a manuscript to a conference or journal. Various papers from the medicine field focus on educating physicians in applying machine learning (Mongan, Moy, & Kahn, 2020; Pineau, 2020; Qiao, 2019; Winkler-Schwartz et al., 2019). However, while these papers share some problem characteristics and choices with IS research, they mainly focus on mapping them to a clinical audience's specific needs. The computer science field contains three important papers on SML. First, Pineau (2020) proposes a short checklist to foster reproducibility in general machine learning endeavors. He emphasizes precise descriptions for models, theory, data, code, and results (e.g., to include clear README files). Second Dodge, Gururangan, Card, Schwartz, and Smith (2019) stress the need to report results for natural language processing (NL)—especially for hyperparameter tuning. To allow for more realistic results, they propose that researchers use their novel technique called expected validation performance. Furthermore, they elaborate on how researchers should document the hardware they use. While hardware constitutes an important metric in computer science to estimate machine learning models' runtimes and complexities (Dodge et al., 2019; Pineau, 2020), these aspects play a minor role in reproducing more application-oriented IS; as such, we neglect them in this paper. Third, Mitchell et al. (2019) present a "model card" with a focus on the fairness and ethics of machine learning models as they conclude that data scientists have not yet integrated fairness and bias topics into their minds. Beyond the computer science field, which focuses on the industrial sector, Studer et al. (2020) propose an adapted version of CRISP-DM for researchers when applying machine learning in the automotive sector with a checklist on specific quality assessment measures. In contrast to these related checklists, our proposed SMLR 1) focuses on the holistic SML process from problem statement to productive deployment, 2) details the relevant problem characteristics that researchers should extract in applying SML (and not ML in general), and 3) presents the findings with an IS audience in mind. Where appropriate, we highlight where insights from other papers influenced our presented SMLR's design.

3 Towards Rigorous Supervised Machine Learning Documentation

From reviewing the literature, we confirmed that, thus far, no process model systematically captures all the problem characteristics that researchers should report and the choices that they should make in applying SML in the IS field. Thus, we decided to collect and merge the necessary problem characteristics and key choices from various sources. To do so, we gathered individual parts of the entire process from relevant literature and augmented other parts based on logical reasoning and best practices that researchers have gained from executing typical SML projects.

3.1 SML Problem Characteristics and Key Choices

For the subsequent analysis, we further divided an SML endeavor into the following three main steps: 1) model initiation, 2) model performance estimation, and, if applicable, 3) model deployment (Hirt, Kühl, & Satzger, 2017) (see Figure 2). In the model initiation step, researchers formulate the objectives for the endeavor and gather, prepare, and characterize the matching data set. Having initiated a model, they will then estimate its performance by training and testing models on a data set D in which they know the target that they will predict. First, models learn patterns in the data from a training subset $D_{Tr} \subseteq D$ and then apply

it towards a test set $D_{Te} = D / D_{Tr}$ of the data that researchers did not use for training. Researchers apply cross-validation approaches to perform this step with various alternative D_{Tr}/D_{Te} splits.

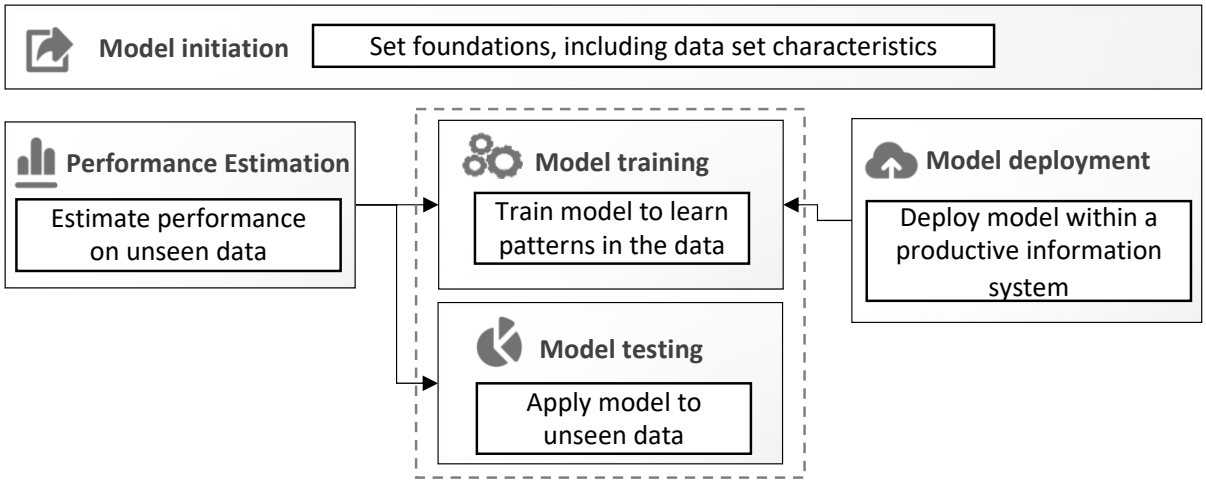

Figure 2. Overview of Supervised Machine Learning Steps

When conducting SML endeavors, researchers need to specify problem characteristics (e.g., class distribution) and elaborate on the choices they make (e.g., performance measure). Additionally, they need to state these key insights when publishing the results because readers can judge the endeavor's rigor and meaningfulness only with this contextual information. For instance, if researchers do not specify that they used hyperparameter optimization in the SML process, readers cannot easily verify whether they could further improve their model's performance or if the researchers simply accepted the performance of the first best tuple of hyperparameters (Dodge et al., 2019).

Researchers need to precisely define their endeavor's goals. They should note the purpose and the targeted application (Mongan et al., 2020). One needs the necessary initiation and performance estimation activities to estimate a model's performance on unseen data based on a set D_{Te} of data for which one knows the feature to be predicted. This step commonly appears in SML across all fields that leverage it, such as medicine (Shipp et al., 2002) and physics (Montavon et al., 2013). However, when conducting an SML endeavor in IS, both performance estimation and model deployment constitute inherent steps. This implementation in a productive software tool continuously exposes the model to new, incoming data (Shmueli & Koppius, 2011). While model performance estimation builds on both training and testing activities, model deployment leverages only the training to create a deployable model. For instance, in a model performance estimation, researchers cannot use all data to train the model since they need to save a certain share for validation and/or testing purposes. For model deployment, however, researchers should use as much data as they can because more data enables the model to perform better (Banko & Brill, 2001). Therefore, after estimating the model performance, researchers build the final model by using all available data *D* in the model deployment phase.

3.1.1 Model Initiation

When conducting SML, researchers need to define a model. We can consider a model as a tuple of parameters that describe which algorithm researchers use, how they initiate its parameters, and what the general process looks like. Researchers define these basic assumptions and surrounding conditions in the model initiation. They serve as the basis for the subsequent model building, for model evaluation (as part of performance estimation), and for model deployment.

First, researchers should state the problem that the SML endeavor addresses (Qiao, 2019). To do so, they need to specify a target value and the SML problem type (e.g., binary/multi-class classification or regression problems). It should be clear from the start what the problem type is ("what should be solved") (Chapman et al., 2000). Next, they need to consider the different aspects of the data that they use and its characteristics to estimate the task's complexity and also ensure that they can meaningfully judge the final results at a later point. Such efforts begin with data gathering and precise definitions for how they will perform it (Oquendo et al., 2012; Winkler-Schwartz et al., 2019). SML requires a target value, which researchers can either collect together with data or label separately (automatically or manually) after

collecting data. In any event, researchers need to explain if and how the labeling takes place. If they have much data to analyze, they can conduct sampling[2]; that is, to pull a representative subset of a larger data set (Dhar, Geva, Oestreicher-Singer, & Sundararajan, 2014). In recent years in particular, the sampling process has been not only relevant to retrieve a representative data set but also fair without any biases (Barocas, Hardt, & Narayanan, 2017). With a data set to analyze, researchers then need to specify additional problem characteristics and key choices. The data distribution has major importance since it ultimately determines how one interprets the results (He & Ma, 2013). For instance, in a binary classification on a data set with a minority class distribution of 10 percent, a dummy classifier that simply assigns all instances as the majority class can easily achieve an accuracy of 90 percent by simply predicting all observations as belonging to the majority class. Regardless of the performance metric, however, researchers need to specifically mention the number of classes and their shares for every classification problem (e.g., as a table). The same applies to regression problems (e.g., a representation as a boxplot) to enable the reader to understand the basic problem. Furthermore, researchers need to clarify if and which they applied data preprocessing methods—for any type of data. For instance, with natural language processing (NLP), manifold ways in which one can transform unstructured text data into structured, machine-digestible formats exist (Manning & Schütze, 2000). Therefore, researchers need to specify which transformation techniques they applied and why they applied them for a specific problem. Apart from the preprocessing, statements about the data quality can help readers to better understand the data the authors dealt with. Data quality covers many aspects, such as correctness ("is it true?"), accuracy ("how precise?"), completeness ("is it complete?"), and relevance ("is it related to the initial problem?") (Wang, Kon, & Madnick, 1993). Sparsity and noise represent two data quality characteristics, and researchers can use many different complexity measures to assess them (Ho & Basu, 2002).

3.1.2 Model Training and Testing

Training and testing represent essential parts of any machine learning endeavor. However, researchers need to clearly define why they conduct these activities. In particular, we distinguish between estimating the model's performance on unseen data (Section 3.3) and deploying a model in a software tool (Section 3.4).

In the model training phase, data sampling, which occurs prior to training a model, can have a significant impact on the model's performance (Chawla, 2005). Popular sampling techniques for dealing with uneven class sizes include undersampling, oversampling, or the synthetic minority over-sampling technique (SMOTE). Researchers apply undersampling when they limit the number of random sample instances that they take from the majority of observations to match the size of the minority data set that they use for training purposes (Rahman & Davis, 2013). In contrast, oversampling randomly duplicates instances from the minority class so that researchers can work with more instances than originally available (Rahman & Davis, 2013). SMOTE creates new additional synthetic instances to match the number of training set elements in the majority class (Chawla, Bowyer, Hall, & Kegelmeyer, 2002).

At its core, model training phase involves selecting an algorithm and its parameters, which creates another set of choices. For instance, popular ML frameworks such as the Python-based "scikit-learn" (Pedregosa et al., 2011) and the Java-based "WEKA" (Hall et al., 2009) feature more than 60 and 30 supervised learning algorithm implementations, respectively. One can classify SML algorithms in different ways (Caruana & Niculescu-Mizil, 2006; Hastie, Tibshirani, & Friedman, 2009; Kotsiantis, 2007). Aggarwal and Zhai (2012) divide supervised algorithms into the major classes of linear algorithms (e.g., support vector machines or regressions), decision trees, pattern (rule-)based algorithms, probabilistic and naive Bayes algorithms, and meta-algorithms. These classes each have their advantages and disadvantages in general and in relation to the specific data and problem that one applies them to. While we cannot go into the details about each class, Kotsiantis (2007) provides more details on the particular selection criteria.

When it comes to model testing, researchers need to early define one or multiple performance metrics, which serve as the central criteria to estimate alternative models' performance and to finally evaluate the SML endeavor's success. Common metrics used for classification tasks include accuracy, precision, sensitivity, specificity, recall, F-measure, and AUC (Powers, 2011). On the other hand, metrics for regression tasks include mean squared error (MSE), R^2, correlation coefficient (CC), normalized root mean squared error (NRMSE), signal-noise ratio (SNR), coefficient of determination (COD), and global

[2] Note that, when it comes to machine learning, one can use the term sampling in three different scenarios with different objectives. One can use it to pull representative data when gathering data (as we describe above), to distribute data for a fold as part of the cross-validation (stratified sampling), or to counterbalance a minority class as part of the model training set (e.g., oversampling).

deviation (GD) (Spuler, Sarasola-Sanz, Birbaumer, Rosenstiel, & Ramos-Murguialday, 2015). When it comes to choosing one or multiple metrics, researchers again need to consider the problem's and data set's nature. For instance, a valuable metric to present the fraction of relevant observations among retrieved observations, recalls lacks meaning on its own since simply predicting all observations as belonging to the positive class can easily bring it to 100 percent. F_β-metrics (e.g., F_1-score) feature an inherent tradeoff between precision and recall in their design (Goutte & Gaussier, 2005). As for regression, popular choices include R^2 and explained variance. Additionally, for both regression and classification, plotting a learning curve can be meaningful because it can show the training and test set errors for each fold of the cross-validation and the respective amounts of data, which helps researchers estimate the bias-variance tradeoff (Blanc, 2016).

3.1.3 Performance Estimation

Based on the performance estimation, researchers can draw conclusions on how the trained model performs on unseen data. In order to do so, they leverage the training and testing steps that we describe above. Thus, researchers need to split the data set to allow for these two activities. Researchers have two different options when it comes to data splitting: percentage split and cross-validation (Abdullah, Qasem, Mohammed, & Emad, 2011). A percentage split refers to a simple that researchers split into a (larger) training set and a (smaller) test set. Researchers train the machine learning model on the training set and then apply it to the test set for evaluation. In IS research, researchers often have access to a limited amount of available observations, which makes data precious. Therefore, evaluations of machine learning models evaluated with percentage split may vary significantly in performance depending on the instances in the training set because the data may or may not have trained it "well" (James, Witten, Hastie, & Tibshirani, 2013). Generally, researchers can divide the error resulting from this prediction into bias, variance, and irreducible error (Friedman, 1997). In order to counteract the random effect of choosing data for the sets, researchers can implement a k-fold cross-validation. To do so, they divide the original data into k folds of equal size. They train the model with (k-1) folds (training set) and apply it on the remaining fold, called the validation set or local test set. They repeat this process k times with each k fold. They average the aggregated performances from the individual iterations, which represent a more meaningful performance assessment than a single percentage split (Golub, Heath, & Wahba, 1979). For both cases, percentage split and cross-validation, stratified sampling allows researchers to maintain the original data set's distribution in the training and test set (Neyman, 1934), which reduces the randomness associated with allocating the two subsets.

If researchers simply want to demonstrate one machine learning model's capabilities, one-time splits, such as percentage or k-fold, can be sufficient. If, however, they want to try out different models, optimize parameters, and estimate the error of a model on unseen data, they should conduct additional steps. If researchers use any optimization, they need to test the model on completely unseen data (i.e., data that they have never used in any training or optimization iteration) (Cawley & Talbot, 2010). They should never use a so-called hold-out set or global test set to change or reconfigure models but preferably only to evaluate them once (Tušar, Gantar, Koblar, Ženko, & Filipič, 2017). In order to perform hyperparameter optimization without overfitting, the nested cross-validation first splits the data into training/validation set and a hold-out set. Then, researchers can apply cross-validation with parameter optimization in an inner cross-validation, which makes it possible to select and evaluate—but not again optimize—the best performing models in the outer cross-validation. To summarize, when it comes to model performance estimation, researchers need to consider whether to separate the data into multiple sets depending on the use case:

- Training set refers to the data set on which researchers train a model
- Validation set or local test set refers to the data set on which researchers optimize a model. However, they must not use it to evaluate the model's performance; otherwise, the model tends to overfit. Researchers require a validation set if they perform parameter optimization.
- Hold-out set or global test refers to the data set that researchers use to evaluate a model, and they should never use this set to optimize the model.

3.1.4 Model Deployment

In the final model deployment phase, researchers generate, implement, and distribute a previously built supervised machine learning model in a software tool. Data contains information and, thus, value; therefore, one should use the complete data set for the final machine learning deployment (see Figure 3) (Gama, Medas, Castillo, & Rodrigues, 2004). The model would incorporate parameters, which one would

typically have selected from the previous estimation. These parameters also help researchers to understand the model's robustness (i.e., its tendency to overfit). For instance, analyzing the optimal parameters of the cross-validation's inner folds might reveal that a specific parameter combination occurs multiple times or, if the model is stable, all the time. Researchers might then directly use this parameter combination for the final training. Alternatively, they can use an additional cross-validation with the complete data set to choose the parameters for final training.

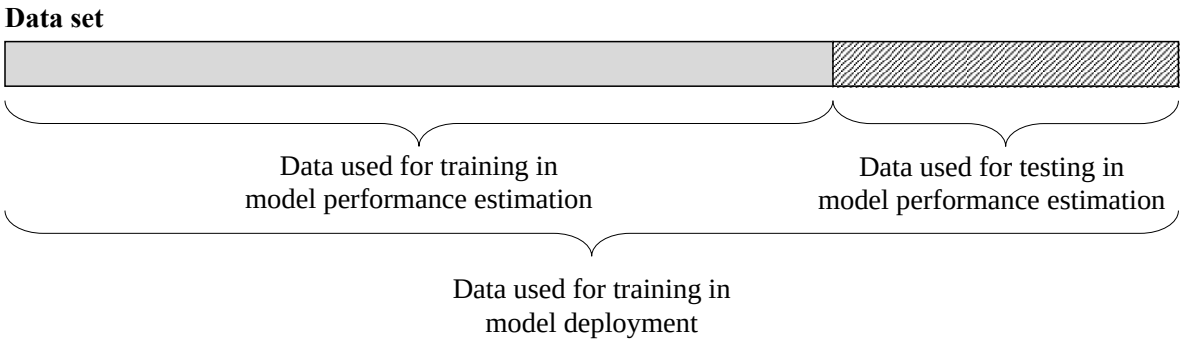

Figure 3. Data Sets for Training, Testing, and Final Deployment

Subsequently, researchers need to export the final model, also called serialization (Zaharia et al., 2018), to save its state and the preprocessing pipeline that one used for further usage. Having concluded the serialization phase, researchers can build the serialized object into a workflow, such as a connected Web service, to predict the target value of new, incoming data. To do so, one sends data to the serialized object for the model to preprocess and classify. They need to consider this final model's validity, such as how robust it remains to changes in the data (Gama et al., 2004) and/or whether it continuously maintains its performance (Feurer et al., 2015). They also need to address possible changes in the data in the future preferably directly by continuously updating the model automatically or, at least, by (qualitatively) estimating the performance for future changes (Baier, Kühl, & Satzger, 2019). For instance, a predictive model for sensor data in a production line might still be valid for a long time as long as the produced goods remain the same. However, if the production line changes or produces new goods, one would need to update the model. In sum, researchers need to address how the model copes with new, incoming data and, consequently, whether or not the model they need to continuously improve the model (and, if not, why).

3.2 The Supervised Machine Learning Report Card (SMLR)

For each step in an SML endeavor that we discuss in Section 3.1, we identify key choices and problem characteristics to systematically capture and document them. In Figure 4, we present the Supervised Machine Learning Report Card (SMLR), which allocates the identified problem characteristics and key choices alongside these steps. Researchers should address and define them when conducting and describing a supervised machine learning endeavor.

During the model initiation phase, the problem statement itself constitutes a key characteristic. This statement classifies the supervised machine learning problem as being either a binary, a multiclass, or a regression problem. Since every supervised approach requires data, researchers should describe the data-gathering process and how they constructed a ground truth data set in detail. In order to better understand the data itself, researchers should describe the data's distribution and overall quality (e.g., its sparsity and noise). Depending on the distribution of classes, researchers might need to sample data points and, thus, describe that process (e.g., type of sampling). Lastly, data preprocessing (Kotsiantis, 2007), feature engineering, and vectorizing (Domingos, 2012) not only have a major influence on the trained model's overall performance but also can cause major methodological mistakes, such as data leakage. Thus, researchers need to consider different methods and reasons for using them.

In the error estimation phase, researchers should estimate the model's performance on unseen data. Thus, they need to specify information about the algorithm, the parameter search space, the search algorithm (e.g., grid search, random search), and the data-splitting method (e.g., percentage split, cross validation). In the proposed report card, we list model training and testing as two separate units that require thorough description. During the model training phase, researchers can sample data to train a

better prediction model. Furthermore, they should describe the algorithm that they used and how they implemented it. To do so, they need to go beyond simply reporting the name of the approach that they used. For neural networks in particular, researchers need to rigorously document their model's architecture, which includes the type of network layers (e.g., convolutional, recurrent, or fully connected layers) they applied and the number of neurons per layer. Researchers need to choose a suitable performance metric for a given problem to ensure a supervised machine learning endeavor succeeds. Whereas accuracy might represent a model's performance well in a class-balanced scenario, its descriptive capability typically decreases when it comes to highly imbalanced data. Each performance metric has its advantages and disadvantages. Researchers should use either multiple (e.g., accuracy + precision + recall + AUC) or composed (e.g., F-score) metrics as single metrics can be easily tuned and do not holistically overview a predictive model's qualities. Furthermore, researchers need to contextualize the results according to a performance evaluation/benchmark. For instance, if other papers or even data science challenges such as Kaggle have used the data set that researchers use, the performance results from these works should serve as a benchmark for direct comparison. If researchers cannot access such results or they do not exist, they should refer to obvious benchmarks, such as naïve models (e.g., a random guess or the prediction of the majority class/mean from the training set) or simpler models (e.g., a basic linear or logistic regression). By providing this context, readers can better understand the obtained performance's quality.

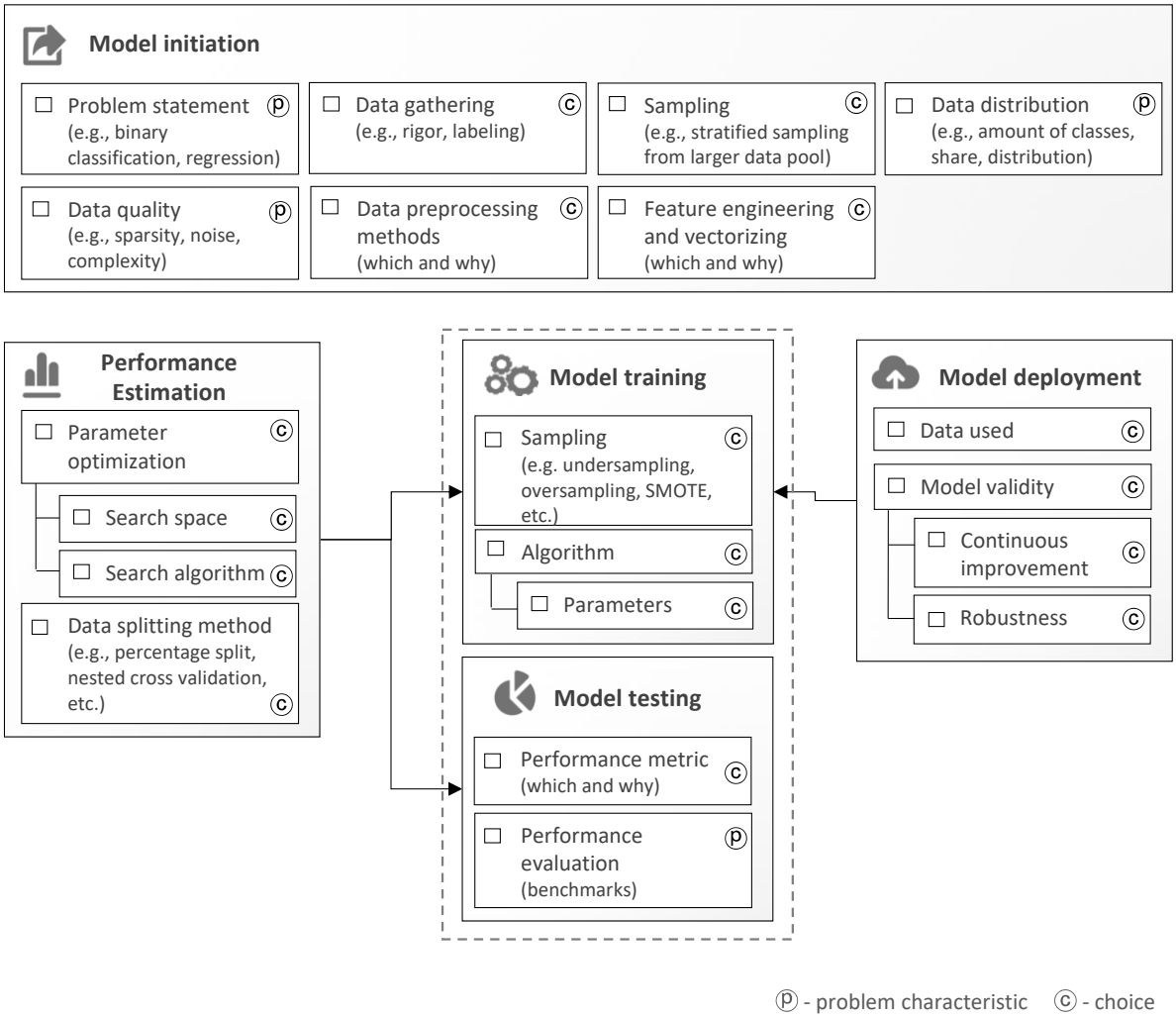

Figure 4. Overview of Supervised Machine Learning Steps and Corresponding Problem Characteristics and Key Choices

Researchers can use an estimated model's performance to show a model's effectiveness. If researchers need to implement a model for predictive modeling in the model deployment phase, they put the model into practice to solve an initial problem. In this scenario, researchers should use the algorithm and the

parameters and sampling method they identified previously to train the model. Furthermore, they should describe the data that they used to train the final model. Since models can only represent a hypothesis based on training data, their validity decreases as the corresponding real-world situation changes. In order to address these changes, researchers should address mode validity, possible continuous improvement techniques, and the model's application to unseen data (robustness).

To ensure we provided a complete approach, we compared the characteristics and choices that we included in the report card with two widely used process models for data science projects: CRISP-DM (Chapman et al., 2000) and the team data science process (Microsoft, 2020). The analysis revealed that the report card covered all important aspects of a machine learning endeavor. We determined a gap between the report card and the two process models only in documenting requirements from the field and describing the business assessment. However, those two aspects usually do not apply to the academic context. We compare our report card with CRISP-DM and the team data science process in Tables C1 and C2 in the Appendix in detail.

We primarily wrote this paper and developed the SMLR to generate awareness about the identified problem characteristics and key choices when researchers conduct SML. However, if applicable, researchers can also use it as a framework to document these precise choices. To demonstrate a possible application, we depict a typical machine learning challenge using the Iris data set (Fisher, 1936) and report on the results in Table 1.

Table 1. Exemplary Report Card based on the Iris Data Set

Problem statement	Predict iris flower class based on four attributes: petal length, petal width, sepal length, sepal width		
Data gathering	Pre-defined data set by scikit-learn package for Python (Pedregosa et al., 2011) from Fisher (1936)		
Data distribution	Three flower classes setosa, versicolor, virginica with 50 instances each; 150 instances in total		
Sampling	No sampling		
Data quality	No missing values		
Data preprocessing methods	No preprocessing		
Feature engineering and vectorizing	No additional features apart from *Petal Length, Petal Width, Sepal Length, Sepal width*, no		
Performance estimation			
Parameter optimization	Yes		
	Search space	RBF kernel	$\gamma \in \{0.001;0.0001\}$ $C \in \{1;10;100;1000\}$
		Linear kernel	$C \in \{1;10;100;1000\}$
	Search algorithm	Grid search	
Data split	Nested cross-validation, 3 outer folds, 5 inner folds		
Algorithm	Support vector classifier		
Sampling	No sampling		
Performance metric	F_1-score as a compromise between precision and recall		
Performance evaluation	Average F_1-score performance on outer folds: 0.9778, which is a nearly perfect score		
Model deployment			
Data used	Full data set (150 instances)		
Model validity	**Continuous improvement**	No continuous improvement	
	Robustness	No statement about the suitability possible	
Sampling	No sampling		
Algorithm	Support vector classifier		
	Parameters	RBF kernel	$\gamma = 0.001$ $C = 1000$

Note: bold writing indicates a problem characteristic or choice from the report card.

4 Empirical Study

With the SMLR at hand, we review renowned papers from the IS literature to identify their strengths and possible ways to improve them based on the presented key choices and problem characteristics.

4.1 Methodology and Data Set

We focused on covering a broad range of high-quality publications in IS. We used the JOURQUAL3 rating, which considers 64,113 journal and conference evaluations from 1,100 professors (VHB, 2012, 2019) in total, as our basis. We focused on the top three journals and top two conference proceedings in the IS community (Hennig-Thurau, Walsh, & Schrader, 2004): *Management Information Systems Quarterly* (*MISQ*), *Information Systems Research* (*ISR*), *Journal of Management Information Systems (JMIS)*, and the proceedings of the International Conference on Information Systems (ICIS) and the European Conference on Information Systems (ECIS).

Table 2. Number of Screened and Relevant Papers for each Outlet from 2010 to 2018

	MISQ	ISR	JMIS	ICIS	ECIS	Total
Screened papers	288	463	390	3,118	2,257	6,516
Relevant papers	7	13	15	43	43	121
Binary classification	1	8	8	19	24	60
Multi-class classification	5	2	7	15	10	39
Regression	1	3	0	9	9	22

In order to obtain enough papers for our study, we considered papers that the above outlets published from 2010 to 2018. In total, we downloaded and screen 6,516 papers. Among them, we identified papers in which SML played a major role. Naturally, we identified "borderline cases" in which papers SML applied as a side note and documented it in a few sentences or a small paragraph. To name a few examples, Huang, Boh, and Goh (2017) applied SML for automated sentiment labeling, Walden, Cogo, Lucus, Moradiabadi, and Safi (2018) used SML for an aspect of their experiment analysis, and Ivanov and Sharman (2018) merely applied SML in the appendix for a robustness check. We excluded these cases as they did not use SML as their *main* method. However, we stress that our proposed SMLR would meaningfully help researchers document SML in such small applications too. While researchers would not need to go into detail in the text body, they could just append the filled report card in an appendix for interested readers and readers who wanted to replicate the results (see Table C1 in the Appendix).

Based on this process, we identified 121 research papers (both completed and in progress) that described applying SML as we show in Table 2. One can also see how SML in IS has grown in use over the years. In 2010, only six papers applied SML in their research; in 2018, 30 research papers did. We provide more details on the chronological development in the distinct outlets in Figure 5.

Next, we thoroughly examined all 121 papers across the entire time frame regarding the report card steps with their problem characteristics and key choices (see Section 3). We distinguished between binary classification, multi-class classification, and regression problems (Chollet, 2018). Most SML-based papers (60) solved binary classification problems (e.g., Oh & Sheng 2011; Pant & Srinivasan, 2010; Amrit, Wijnhoven, & Beckers, 2015) followed by 39 papers that solved multi-class problems (e.g., Dorner & Alpers, 2017; Wang et al., 2013; Geva & Oestreicher-Singer, 2013) and 22 papers that solved regression problems (e.g., Riekert et al., 2017; Feuerriegel, Riedlinger, & Neumann, 2014; Ding, Li, & Chatterjee, 2015).

Next, we describe our findings with regard to the model initiation, performance estimation, and model deployment steps (see Section 3). We summarize these findings in Figure 6 and discuss them in Sections 4.2 to 4.4. We show the analyses for the individual journals and conferences in Figures A1 and B1 in the Appendix. Note that we assessed all publications according to the same objective criteria. We did not consider whether each indicator contributed meaning to the individual publication; for example, a study on SML's feasibility for a certain business challenge might not need to deal with the necessary steps for deployment.

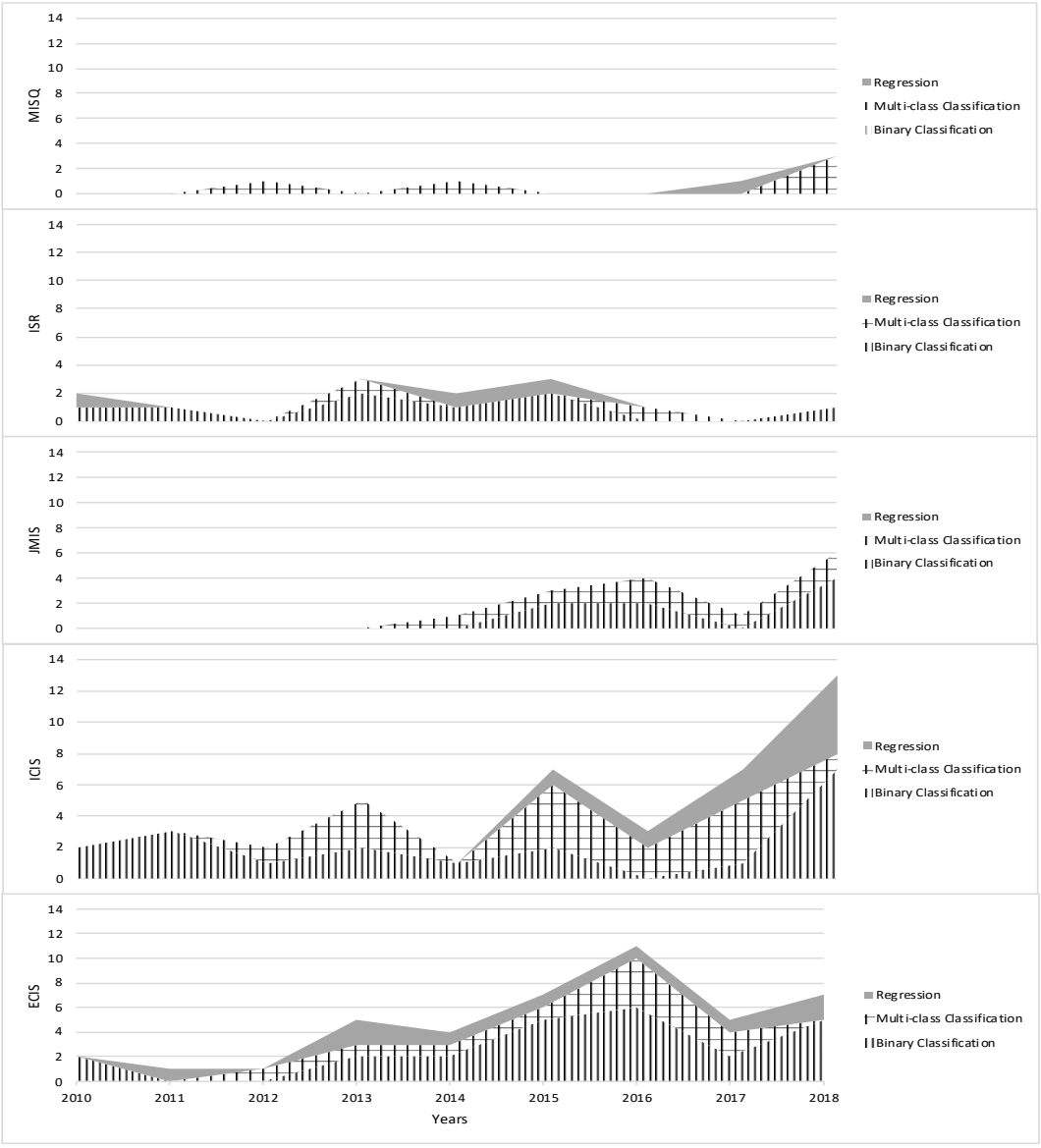

Figure 5. Amount of Supervised Machine Learning Papers in *MISQ*, *ISR*, and *JMIS* and the proceedings of ICIS and ECIS from 2010 to 2018

Step	Indicator		Described in papers			Positive example
Model initiation	Problem statement		100.00%	(121/121)		Abbasi, Albrecht, Vance, & Hansen (2012)
	Data gathering		87.60%	(106/121)		Lin, Chen, Brown, Li, & Yang (2017)
	Data distribution		74.38%	(90/121)		Stange & Funk (2016)
	Sampling		14.88%	(18/121)		Dhar et al. (2014)
	Data quality		67.77%	(82/121)		Hopf, Sodenkamp, Riechel, & Staake (2017)
	Data preprocessing methods		76.03%	(92/121)		Johnson, Safadi, & Faraj (2015)
	Feature engineering and vectorizing		74.38%	(90/121)		Martens & Provost (2014)
Performance estimation	Parameter Optimization	Search Space	12.40%	(15/121)		Riekert et al. (2017)
		Search Algorithm	13.22%	(16/121)		Zhou (2017)
	Data split		95.87%	(116/121)		Urbanke et al. (2017)
	Algorithm		100.00%	(121/121)		Oh & Sheng (2011)
	Sampling		6.61%	(8/121)		Wang et al. (2013)
	Performance metric (reasoned)		48.76%	(59/121)		Fang, Hu, Li, & Tsai (2013)
	Performance evaluation		48.76%	(59/121)		Han et al. (2017)
Model deployment	Data used		1.65%	(2/121)		Abbas, Zhou, Deng, & Zhang (2018)
	Model validity	Continuous Improvement	1.65%	(2/121)		Seebach, Pahlke, & Beck (2011)
		Robustness	15.70%	(19/121)		Koroleva & José Bolufé Röhler (2012)

Figure 6. Overview of Supervised Machine Learning Report Card Steps and their Documentation

4.2 Model Initiation

Researchers need to describe data's characteristics to understand the model that they build on top of it. At first, researchers need to describe the data source and/or the data-collection process. We found that, among the 121 papers in our sample, 15 papers (12%) neither clearly stated the data's origin nor where their authors gathered it from. Data's quality determines a model's quality; however, 39 papers (32%) did not provide any information on data quality. Furthermore, 31 papers (26%) did not describe the applied data set's statistical distribution. Statistic distribution relates to both the target variable's distribution and to the information about the attributes that one uses for prediction. When authors do not provide this information, readers cannot judge the final model's performance for a given metric. Furthermore, if readers do not know the distribution, the performance values can be meaningless (e.g., one can achieve a reported accuracy of 99% with a 1% minority class by simply assigning all instances to the majority class). In referring to the total number of messages and the number of harassment messages (target variable) that their dataset included, Bretschneider and Peters (2016) exemplify a sound data description. Data preprocessing and feature engineering also represent essential choices in an SML endeavor. However, 16 papers (13%) did not include any information about the preprocessing or feature engineering activities that their authors chose. Yet, any researcher or practitioner who wants to build a predictive model in the same domain would find this information very valuable. For instance, if we do not know how researchers have handled quality issues, such as incomplete data, their results may be flawed. Furthermore, researchers would not be able to re-create results if a paper omits the data's preprocessing techniques because various different possibilities for preprocessing exist. Stange and Funk (2015) thoroughly explain how they transformed real-time advertising data before feeding this data into the model training phase and, thus, enable others to benefit from their knowledge.

Assessing a model's performance depends highly on the performance metric that researchers choose, which makes the choice a critically important one for every SML endeavor. Thus, researchers need to specify the performance metric and the reason why they chose it. Nevertheless, only 59 papers (49%) actually explained why they chose the evaluation metric that they did. For instance, Riekert, Leukel, and Klein (2016) stated that they applied accuracy as evaluation metric; however, they did not explain why it suited or what meaning it added to their underlying problem.

4.3 Performance Estimation

We found only 23 papers (19%) that mentioned the parameters that their authors used in the model training phase. A model's performance can vary significantly depending on the chosen parameters; as such, researchers have to thoroughly define and describe the parameter space[3]. In fact, 116 papers (96%) included information about how their authors split the dataset into a training set and a test set (e.g., Lash & Zhao, 2016; Urbanke, Uhlig, & Kranz, 2017; Chatterjee, Saeedfar, Tofangchi, & Kolbe, 2018). If authors do not disclose this information, the reader cannot judge results' rigor because it might even imply that they did not split their data at all. If researchers perform model training and model testing on the same data set, the measured performance is misleading and unrealistically high (James et al., 2013).

In order to comprehensively understand a trained model's performance, researchers need to compare it to previously built models or other approaches that strive to solve the same problem. Thus, if any previous research or algorithm deals with the same problem or data set, researchers should always compare the developed model to the previous model. If no previous research exists, researchers should compare its performance to other metrics, such as random guesses (Li, Goethals, Giangreco, & Baesens, 2013) or standard SML algorithms. Kozlovskiy, Sodenkamp, Hopf, and Staake (2016) provide a good example by comparing their model's performance to a random guess. Only 59 papers (49%) actually introduced a performance comparison (e.g., Cui, Wong, & Wan, 2012; Geva & Oestreicher-Singer, 2013; Han, Wang, & Huang, 2017). The remaining papers merely introduced the results of the predictive models without any comparison in which case a reader can hardly judge the presented model's actual quality.

4.4 Model Deployment

We found the least report card compliance in our sample when it comes to the model deployment phase. As we note in Section 3X, SML endeavors in IS research do not necessarily need to conduct the deployment phase. In certain cases, authors may only want to prove an approach's feasibility, which includes applying SML. When conducting projects that focus on such a goal, researchers do not need to build and/or describe a deployable solution. Nevertheless, only 31 papers (26%) at least described thoughts about a possible model deployment and the corresponding implications even though the IS field should have a strong focus on producing final, implementable results and implications for practice (Gholami, Watson, Molla, Hasan, & Bjørn-Andersen, 2016). On the other hand, we also found some evidence for solutions that researchers deployed (e.g., Schwaiger, Lang, Johannsen, & Leist, 2017), which included explanations about how they built their tool and which choices they needed to make to deploy it in an industry setting. However, even the examples that discussed the model deployment phase did not emphasize which data they could use for the final deployable model.

Model validity in general and model updates in particular constitute another consideration (Baier et al., 2019; Studer et al., 2020). One builds an SML model on data. One assumes that a machine learning model extracts the underlying concepts in this data to fulfill a given task. If one then deploys a SML model, these concepts should not change over time; otherwise the model has to adapt to such "concept drifts" (Gama et al., 2004). If, for example, researchers do not update a model that classifies user-written texts on a social media platform according their authors' age from time to time, its prediction quality will decrease—language (i.e., phrases used by certain age groups) will change over time. Thus, we claim that researchers need to properly ensure they preserve model validity.

5 Conclusion

Supervised machine learning (SML) has become a popular method to solve problems in the IS and other fields. Although SML offers many possibilities for proving effectiveness, efficiency, and application in the predictive modeling space, researchers need to conduct this research in a rigorous and comprehensive manner. Only by doing so can IS researchers enable their peers to understand and reproduce the research they conduct. In this paper, we develop a supervised machine learning report card (SMLR) that captures important key choices and problem characteristics that researchers need to consider in every SML endeavor. We elaborate on them and their importance. We used the report card to analyze whether recent papers in renowned IS outlets have already applied the necessary scrutiny in SML descriptions,

[3] However, researchers do not have to describe such parameters in, for example, linear regression, since they do not need to make any parameter choice.

and we identified several shortcomings in how researchers have documented SML. For instance, not all the papers we reviewed justified their chosen performance metrics, and only a minority used benchmarks to help the reader understand how they evaluated the models.

Our paper has two major limitations. First, we reviewed papers only from five journals/proceedings and considered instances only from 2010 to 2018. While we based the selection on an acknowledged ranking (VHB, 2019), other rankings on important outlets obviously exist. We treated journal and conference publications alike according to the ranking, although journal publications are typically more mature and have longer revision histories. On the other hand, conference publications are timelier and a good indicator for upcoming topics and methods in the community. For the interested reader, however, we append differentiated analyses in the appendix. Regardless of rankings and precise outlets, the general message still remains that we can observe a lack of documentation. Two reasons may explain why: either researchers did not consider the key choices and problem characteristics that they identified, or they fell victim to shortening (e.g., due to the review process or submission guidelines). Therefore, we can analyze only whether research has addressed important key steps; we cannot draw conclusions on the actual conducted research.

The proposed SMLR may prove helpful in future SML endeavors and serve as a guideline to more rigorous, comprehensive research in this area. Once implemented, the report card will enable a more transparent view on SML articles and their reproducibility in the future.

References

Abbasi, A., Albrecht, C., Vance, A., & Hansen, J. (2012). Metafraud: A meta-learning framework for detecting financial fraud. *MIS Quarterly*, *36*(4), 1293-1327.

Abbasi, A., Zhou, Y., Deng, S., & Zhang, P. (2018). Text analytics to support sense-making in social media: A language-action perspective. *MIS Quarterly*, *42*(2), 427-464.

Abdullah, H., Qasem, A., Mohammed, N., & Emad, M. (2011). A comparison study between data mining tools over some classification methods. *International Journal of Advanced Computer Science and Applications*, 18-26.

Aggarwal, C. C., & Zhai, C. (2012). A survey of text classification algorithms. In C. Aggarwal & C. Zhai (Eds.), *Mining text data* (pp. 163-222). Boston, MA: Springer.

Amrit, C., Wijnhoven, F., & Beckers, D. (2015). Information waste on the World Wide Web and combating the clutter. In *Proceedings of the 23rd European Conference on Information Systems.*

Anand, S. S., & Büchner, A. G. (1998). *Decision support using data mining.* London, UK: Financial Times Pitman Publishers.

Baier, L., Kühl, N., & Satzger, G. (2019). How to cope with change? Preserving validity of predictive services over time. In Proceedings of the 52nd *Hawaii International Conference on System Sciences.*

Banko, M., & Brill, E. (2001). Scaling to very very large corpora for natural language disambiguation. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*.

Barocas, S., Hardt, M., & Narayanan, A. (2019). *Fairness in machine learning*. Retrieved from https://fairmlbook.org/

Baumann, A., Lessmann, S., Coussement, K., & De Bock, K. W. (2015). Maximize what matters: Predicting customer churn with decision-centric ensemble selection. In *Proceedings of the 23rd European Conference on Information Systems.*

Blanc, S. M. (2016). *Bias-variance aware integration of judgmental forecasts and statistical models* (dissertation). Faculty Faculty of Economics, Institute Institute for Information Economics and Marketing, Köln.

Blanc, S. M., & Setzer, T. (2015). Improving forecast accuracy by guided manual overwrite in forecast debiasing. In *Proceedings of the 23rd European Conference on Information Systems.*

Bretschneider, U., & Peters, R. (2016). Detecting cyberbullying in online communities. In *Proceedings of the 24th European Conference on Information Systems.*

Brodley, C. E., & Smyth, P. (1995). The process of applying machine learning algorithms. In *Proceedings of the ICML-95 Workshop on Applying Machine Learning in Practice*.

Cabena, P., Hadjinian, P., Stadler, R., Verhees, J., & Zanasi, A. (1998). *Discovering data mining: frOm concept to implementation*. Upper Saddle River, NJ: Prentice-Hall.

Caruana, R., & Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd International Conference on Machine Learning*.

Cawley, G. C., & Talbot, N. L. C. (2010). On over-fitting in model selection and subsequent selection bias in performance evaluation. *Journal of Machine Learning Research*, *11*, 2079-2107.

Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C., & Wirth, R. (2000). *CRISP-DM 1.0.* Retrieved from https://the-modeling-agency.com/crisp-dm.pdf

Chatterjee, S., Saeedfar, P., Tofangchi, S., & Kolbe, L. (2018). Intelligent road maintenance: A machine learning approach for surface defect detection. *In Proceedings of the 26th European Conference on Information Systems.*

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321-357.

Chawla, N. V. (2005). Data mining for imbalanced datasets: An overview. In O. Maimon & L. Rokach (Eds.), *Data mining and knowledge discovery handbook*. Boston, MA: Springer.

Chollet, F. (2018). *Deep learning with Python*. Shelter Island, NY: Manning Publications.

Cios, K. J., Teresinska, A., Konieczna, S., Potocka, J., & Sharma, S. (2000). Diagnosing myocardial perfusion from PECT bull's-eye maps—a knowledge discovery approach. *IEEE Engineering in Medicine and Biology Magazine*, *19*(4), 17-25.

Coiera, E., Ammenwerth, E., Georgiou, A., & Magrabi, F. (2018). Does health informatics have a replication crisis? *Journal of the American Medical Informatics Association*, *25*(8), 963-968.

Cui, G., Wong, M. L., & Wan, X. (2012). Cost-sensitive learning via priority sampling to improve the return on marketing and CRM investment. *Journal of Management Information Systems*, *29*(1), 341-374.

Dhar, V., Geva, T., Oestreicher-Singer, G., & Sundararajan, A. (2014). Prediction in economic networks. *Information Systems Research*, *25*(2), 264-284.

Ding, A. W., Li, S., & Chatterjee, P. (2015). Learning user real-time intent for optimal dynamic webpage transformation. *Information Systems Research*, *26*(2), 339-359.

Dodge, J., Gururangan, S., Card, D., Schwartz, R., & Smith, N. A. (2019). Show your work: Improved reporting of experimental results. *ArXiv*. Retrieved from https://arxiv.org/abs/1909.03004

Domingos, P. (2012). A few useful things to know about machine learning. *Communications of the ACM*, *55*(10), 78-87.

Dong, W., Liao, S., & Zhang, Z. (2018). Leveraging financial social media data for corporate fraud detection. *Journal of Management Information Systems*, *35*(2), 461-487.

Dorner, V., & Alpers, G. W. (2017). Detecting panic potential in social media tweets. In Proceedings of the *25th European Conference on Information Systems*.

Fang, X., Hu, P. J. H., Li, Z. L., & Tsai, W. (2013). Predicting adoption probabilities in social networks. *Information Systems Research*, *24*(1), 128-145.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. (1996). The KDD process for extracting useful knowledge from volumes of data. *Communications of the ACM*, *39*(11), 27-34.

Feuerriegel, S., & Fehrer, R. (2016). Improving decision analytics with deep learning: The case of financial disclosures. In *Proceedings of the 24th European Conference on Information Systems.*

Feuerriegel, S., Riedlinger, S., & Neumann, D. (2014). Predictive analytics for electricity prices using feed-ins from renewables. In *Proceedings of the 22nd European Conference on Information Systems.*

Feurer, M., Klein, A., Eggensperger, K., Springenberg, J., Blum, M., & Hutter, F. (2015). Efficient and robust automated machine learning. In *Proceedings of the Conference on Advances in Neural Information Processing Systems.*

Fisher, R. A. (1936). The use of multiple measures in taxonomic problems. *Annals of Eugenics*, *7*(2), 179-188.

Friedman, J. H. (1997). On bias, variance, 0/1—loss, and the curse-of-dimensionality. *Data Mining and Knowledge Discovery*, *1*(1), 55-77.

Fu, L.-M. (2003). *Neural networks in computer intelligence*. New York, NY: McGraw-Hill College.

Gama, J., Medas, P., Castillo, G., & Rodrigues, P. (2004). Learning with drift detection. In *Proceedings of the Brazilian Symposium on Artificial Intelligence*.

Geva, T., & Oestreicher-Singer, G. (2013). Do customers speak their minds? Using forums and search for predicting sales. *In* Proceedings of the International Conference on Information Systems.

Gholami, R., Watson, R. T., Molla, A., Hasan, H., & Bjørn-Andersen, N. (2016). Information systems solutions for environmental sustainability: How can we do more? *Journal of the Association for Information Systems*, *17*(8), 521-536.

Gimpel, H., Kleindienst, D., & Waldmann, D. (2018). The disclosure of private data: Measuring the privacy paradox in digital services. *Electronic Markets*, *28*(4), 475-490.

Goby, N., Brandt, T., Feuerriegel, S., & Neumann, D. (2016). Business intelligence for business processes: The case of IT incident management. In *Proceedings of the 24th European Conference on Information Systems.*

Golub, G. H., Heath, M., & Wahba, G. (1979). Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, *21*(2), 215-223.

Gong, J., Abhishek, V., & Li, B. (2017). Examining the impact of keyword ambiguity on search advertising performance: A topic model approach. *MIS Quarterly*, *42*(3), 1-26.

Goutte, C., & Gaussier, E. (2005). A probabilistic interpretation of precision, recall and F-score, with implication for evaluation. In *Proceedings of the European Conference on Information Retrieval*.

Hall, M., National, H., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: An Update. *SIGKDD Explorations*, *11*(1), 10-18.

Han, H., Otto, C., Liu, X., & Jain, A. K. (2015). Demographic estimation from face images: Human vs. machine performance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *37*(6), 1148-1161.

Han, X., Wang, L., & Huang, H. (2017). Deep investment behavior profiling by recurrent neural network in P2P lending. In *Proceedings of the International Conference on Information Systems.*

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The elements of statistical learning* (pp. 9-41). New York, NY: Springer.

He, H., & Ma, Y. (2013). *Imbalanced learning: Foundations, algorithms, and applications*. New York, NY: John Wiley & Sons.

Heilig, L., Hofer, J., Lessmann, S., & Voc, S. (2016). Data-driven product returns prediction: A cloud-based ensemble selection approach. In *Proceedings of the 24th European Conference on Information Systems.*

Hennig-Thurau, T., Walsh, G., & Schrader, U. (2004). VHB-JOURQUAL: Ein ranking von betriebswirtschaftlich-relevanten zeitschriften auf der grundlage von expertenurteilen. *Schmalenbachs Zeitschrift Für Betriebswirtschaftliche Forschung*, *56*(6), 520-545.

Hirt, R., Kühl, N., & Satzger, G. (2017). An end-to-end process model for supervised machine learning classification: From problem to deployment in information systems. In *Proceedings of the 12th International Conference on Design Science Research in Information Systems and Technology.*

Ho, T. K., & Basu, M. (2002). Complexity measures of supervised classification problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(3), 289-300.

Hopf, K., Sodenkamp, M., Riechel, S., & Staake, T. (2017). Predictive customer data analytics—the value of public statistical data and the geographic model transferability. In *Proceedings of the International Conference on Information Systems.*

Huang, J., Boh, W. F., & Goh, K. H. (2017). A Temporal study of the effects of online opinions: Information sources matter. *Journal of Management Information Systems*, *34*(4), 1169-1202.

Huang, K.-Y., Nambisan, P., & Uzuner, Ö. (2010). Informational support or emotional support: Preliminary study of an automated approach to analyze online support community contents. In *Proceedings of the International Conference on Information Systems.*

Hutson, M. (2018). Missing data hinder replication of artificial intelligence studies. *Science*. Retrieved from https://www.sciencemag.org/news/2018/02/missing-data-hinder-replication-artificial-intelligence-studies

Ivanov, A., & Sharman, R. (2018). Impact of user-generated internet content on hospital reputational dynamics. *Journal of Management Information Systems*, *35*(4), 1277-1300.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning*. New York, NY: Springer.

Johnson, S. L., Safadi, H., & Faraj, S. (2015). The emergence of online community leadership. *Information Systems Research*, *26*(1), 165-187.

Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*, *349*(6245), 255-260.

Kitchens, B., Dobolyi, D., Li, J., & Abbasi, A. (2018). Advanced customer analytics: Strategic value through integration of relationship-oriented big data. *Journal of Management Information Systems*, *35*(2), 540-574.

Koroleva, K., & José Bolufé Röhler, A. (2012). Reducing information overload: Design and evaluation of filtering and ranking algorithms for social networking sites. In *Proceedings of the 20th European Conference on Information Systems.*

Kotsiantis, S. B. (2007). Supervised machine learning: A review of classification techniques. *Informatica*, *31*, 249-268.

Kowatsch, T., & Maass, W. (2018). A data-analytical system to predict therapy success for obese children. In *Proceedings of the International Conference on Information Systems*.

Kozlovskiy, I., Sodenkamp, M. A., Hopf, K., & Staake, T. (2016). Energy informatics for environmental, economic and societal sustainability: A case of the large-scale detection of households with old heating systems. In *Proceedings of the 24th European Conference on Information Systems.*

Kurgan, L., & Musilek, P. (2006). A survey of knowledge discovery and data mining process models. *The Knowledge Engineering Review*, *21*(1), 1-24.

Laing, S., & Kühl, N. (2018). Comfort-as-a-service: Designing a user-oriented thermal comfort artifact for office buildings. In *Proceedings of the International Conference on Information Systems.*

Lash, M. T., & Zhao, K. (2016). Early predictions of movie success: The who, what, and when of profitability. *Journal of Management Information Systems*, *33*(3), 874-903.

Li, L., Goethals, F., Giangreco, A., & Baesens, B. (2013). Using social network data to predict technology acceptance. In *Proceedings of the International Conference on Information Systems.*

Li, W., Chen, H., & Nunamaker, J. F., Jr. (2016). Identifying and profiling key sellers in cyber carding community: AZSecure text mining system. *Journal of Management Information Systems*, *33*(4), 1059-1086.

Lin, Y.-K., Chen, H., Brown, R. A., Li, S.-H., & Yang, H.-J. (2017). Healthcare predictive analytics for risk profiling in chronic care: A Bayesian multitask learning approach. *MIS Quarterly*, *41*(2), 473-495.

Lüttenberg, H., Bartelheimer, C., & Beverungen, D. (2018). designing predictive maintenance for agricultural machines. In *Proceedings of the 26th European Conference on Information Systems.*

Manning, C. D., & Schütze, H. (2000). *Foundations of natural language processing*. Cambridge, MA: MIT Press.

Martens, D., & Provost, F. (2014). Explaining data-driven document classifications. *MIS Quarterly*, *38*(1), 73-99.

Martens, D., Provost, F., Clark, J., & Junqué de Fortuny, E. (2016). Mining massive fine-grained behavior data to improve predictive analytics. *MIS Quarterly*, *40*(4), 869-888.

Microsoft. (2020). *Microsoft team data science process documentation*. Retrieved from https://docs.microsoft.com/en-us/azure/machine-learning/team-data-science-process/overview

Mitchell, M., Wu, S., Zaldivar, A., Barnes, P., Vasserman, L., Hutchinson, B., Spitzer, E., Raji, I. D., & Gebru, T. (2019). Model cards for model reporting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*.

Mo, J., Sarkar, S., & Menon, S. (2018). Know when to run: Recommendations in crowdsourcing contests. *MIS Quarterly*, *42*(3), 919-944.

Mohri, M., Rostamizadeh, A., & Talwalkar, A. (2013). Foundations of machine learning. *Journal of Chemical Information and Modeling*, *53*(9), 1689-1699.

Mongan, J., Moy, L., & Kahn, C. E., Jr. (2020). Checklist for artificial intelligence in medical imaging (CLAIM): A guide for authors and reviewers. *Radiology: Artificial Intelligence*, *2*(2).

Neyman, J. (1934). On the two different aspects of the representative method: The method of stratified sampling and the method of purposive selection. *Journal of the Royal Statistical Society*, *97*(4), 558-625.

Oh, C., & Sheng, O. (2011). Investigating predictive power of stock micro blog sentiment in forecasting future stock price directional movement. In *Proceedings of the International Conference on Information Systems.*

Olbrich, S., Frank, U., Gregor, S., Niederman, F., & Rowe, F. (2017). On the merits and limits of replication and negation for IS research. *AIS Transactions on Replication Research*, *3*(1), 1-19.

Olorisade, B. K., Brereton, P., & Andras, P. (2017). *Reproducibility in machine learning-based studies: An example of text mining*. In *Proceedings of the 34th International Conference on Machine Learning.*

Oquendo, M. A., Baca-Garcia, E., Artes-Rodriguez, A., Perez-Cruz, F., Galfalvy, H. C., Blasco-Fontecilla, H., Madigan, D., & Duan, N. (2012). Machine learning and data mining: Strategies for hypothesis generation. *Molecular Psychiatry*, *17*(10), 956-959.

Oroszi, F., & Ruhland, J. (2010). An early warning system for hospital acquired pneumonia. In *Proceedings of the European Conference on Information Systems.*

Pant, G., & Srinivasan, P. (2010). Predicting web page status. *Information Systems Research*, *21*(2), 345-364.

Pant, G., & Srinivasan, P. (2013). Status locality on the Web: Implications for building focused collections. *Information Systems Research*, *24*(3), 802-821.

Pedregosa, F., & Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, *12,* 2825-2830

Peng, R. D. (2011). Reproducible research in computational science. *Science*, *334*(6060), 1226-1227.

Pineau, J. (2020). *The machine learning reproducibility checklist*. Retrieved from http://www.cs.mcgill.ca/~jpineau/ReproducibilityChecklist-v2.0.pdf

Powers, D. M. (2011). Evaluation: From precision, recall and f-measure to roc, informedness, markedness & correlation. *Journal of Machine Learning Technologies*, *2*(1), 37-63.

Pröllochs, N., Feuerriegel, S., & Neumann, D. (2015). Generating domain-specific dictionaries using Bayesian learning. In *Proceedings of the 23rd European Conference on Information Systems.*

Qiao, N. (2019). A systematic review on machine learning in sellar region diseases: quality and reporting items. *Endocrine Connections*, *8*(7), 952-960.

Rahman, M. M., & Davis, D. N. (2013). Addressing the class imbalance problem in medical datasets. *International Journal of Machine Learning and Computing*, *3*(2), 224-228.

Ram, S., Wang, Y., Currim, F., & Currim, S. (2015). Using big data for predicting freshmen retention. In *Proceedings of the International Conference on Information Systems*.

Rätsch, G. (2004). *A brief introduction into machine learning*. Retrieved from http://www.csc.villanova.edu/~tway/courses/mse2400/s2016/handouts/Ratsch%20-%20Brief%20Intro%20into%20Machine%20Learning.pdf

Riekert, M., Leukel, J., & Klein, A. (2016). Online media sentiment: Understanding machine learning-based classifiers. In *Proceedings of the 24th European Conference on Information Systems.*

Riekert, M., Premm, M., Klein, A., Lyubomir Kirilov, Kenngott, H., Apitz, M., Wagner, M., & Ternes, L. (2017). Predicting the duration of surgeries to improve process efficiency in hospitals. In *Proceedings of the 25th European Conference on Information Systems.*

Montavon, G., Rupp, M., Gobre, V., Vazquez-Mayagoitia, A., Hansen, K., Tkatchenko, A., Müller, K.-R., & Von Lilienfeld, O. A. (2013). Machine learning of molecular electronic properties in chemical compound space. *New Journal of Physics*, *15*(9), 1-9.

Samtani, S., Chinn, R., Chen, H., & Nunamaker, J. F., Jr. (2017). Exploring emerging hacker assets and key hackers for proactive cyber threat intelligence. *Journal of Management Information Systems*, *34*(4), 1023-1053.

Schooler, J. W. (2014). Metascience could rescue the "replication crisis". *Nature*, *515*(7525), 9.

Schwaiger, J., Lang, M., Johannsen, F., & Leist, S. (2017). "What does the customer want to tell us?" An automated classification approach for social media posts at small and medium-sized enterprises. In *Proceedings of the 25th European Conference on Information Systems*.

Seebach, C., Pahlke, I., & Beck, R. (2011). Tracking the digital footprints of customers: How firms can improve their sensing abilities to achieve business agility. In *Proceedings of the 19th European Conference on Information Systems.*

Sharp, J., & Babb, J. (2018). Is information systems late to the party? The current state of DevOps research in the Association for Information Systems eLibrary. In *Proceedings of the Americas Conference on Information Systems.*

Shi, D., Guan, J., Zurada, J., & Manikas, A. (2017). A data-mining approach to identification of risk factors in safety management systems. *Journal of Management Information Systems*, *34*(4), 1054-1081.

Shipp, M. A., Ross, K. N., Tamayo, P., Weng, A. P., Kutok, J. L., Aguiar, R. C. T., Gaasenbeek, M., Angelo, M., Reich, M., Pinkus, G. S., Ray, T. S., Koval, M. A., Last, K. W., Norton, A., Lister, T. A., Mesirov, J., Neuberg, D. S., Lander, E. S., Aster, J. C., & Golub, T. R. (2002). Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. *Nature Medicine*, *8*(1), 68-74.

Shmueli, G., & Koppius, O. R. (2011). Predictive analytics in information systems research. *MIS Quarterly*, *35*(3), 553-572.

Spuler, M., Sarasola-Sanz, A., Birbaumer, N., Rosenstiel, W., & Ramos-Murguialday, A. (2015). Comparing metrics to evaluate performance of regression methods for decoding of neural signals. In *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society.*

Stange, M., & Funk, B. (2015). How much tracking is necessary—the learning curve in Bayesian user journey analysis. In *Proceedings of the 23rd European Conference on Information Systems.*

Stange, M., & Funk, B. (2016). Predicting online user behavior based on real-time advertising data. In *Proceedings of the 24th European Conference on Information Systems.*

Staudt, P., Rausch, B., & Weinhardt, C. (2018). Predicting redispatch in the German electricity market using information systems based on machine learning. In *Proceedings of the International Conference on Information Systems.*

Studer, S., Bui, T. B., Drescher, C., Hanuschkin, A., Winkler, L., Peters, S., & Mueller, K.-R. (2020). Towards CRISP-ML(Q): A machine learning process model with quality assurance methodology. *ArXiv.* Retrieved from https://arxiv.org/abs/2003.05155

Tackett, J. L., Brandes, C. M., King, K. M., & Markon, K. E. (2019). Psychology's replication crisis and clinical psychological science. *Annual Review of Clinical Psychology*, *15*, 579-604.

Tafti, A., & Gal, D. (2018). Predicting complainers on social media: A machine learning approach. In *Proceedings of the International Conference on Information Systems*.

Timmerman, Y., & Bronselaer, A. (2019). Measuring data quality in information systems research. *Decision Support Systems*, *126*.

Tripathi, M., & Kaur, I. (2018). Oil prices forecasting: A comparative analysis. In *Proceedings of the International Conference on Information Systems.*

Tušar, T., Gantar, K., Koblar, V., Ženko, B., & Filipič, B. (2017). A study of overfitting in optimization of a manufacturing quality control procedure. *Applied Soft Computing*, *59*, 77-87.

Twyman, N. W., Proudfoot, J. G., Schuetzler, R. M., Elkins, A. C., & Derrick, D. C. (2015). Robustness of multiple indicators in automated screening systems for deception detection. *Journal of Management Information Systems*, *32*(4), 215-245.

Urbanke, P., Uhlig, A., & Kranz, J. (2017). A customized and interpretable deep neural network for high-dimensional business data—evidence from an e-commerce application. In *Proceedings of the International Conference on Information Systems.*

VHB. (2012). VHB-JOURQUAL3. Retrieved from https://vhbonline.org/en/vhb4you/vhb-jourqual/vhb-jourqual-3

VHB. (2019). *Complete list of the journals in VHB-JOURQUAL3 in alphabetical order*. Retrieved from https://vhbonline.org/en/vhb4you/vhb-jourqual/vhb-jourqual-3/complete-list

Voets, M., Møllersen, K., & Bongo, L. A. (2018). Replication study: Development and validation of deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. *ArXiv.* Retrieved from https://arxiv.org/abs/1803.04337

Walden, E., Cogo, G. S., Lucus, D. J., Moradiabadi, E., & Safi, R. (2018). Neural correlates of multidimensional visualizations: An fMRI comparison of bubble and three-dimensional surface graphs using evolutionary theory. *MIS Quarterly*, *42*(4), 1097-1116.

Wang, R. Y., Kon, H. B., & Madnick, S. E. (1993). Data quality requirements analysis and modeling. In *Proceedings of IEEE 9th International Conference on Data Engineering*.

Wang, T., Kannan, K. N., Ulmer, J. R., Wang, T., Kannan, K. N., & Ulmer, J. R. (2013). The association between the disclosure and the realization of information security risk factors. *Information Systems Research*, *24*(2), 201-218.

Weinhardt, C., van der Aalst, W. M. P., & Hinz, O. (2019). Introducing registered reports to the information systems community. *Business and Information Systems Engineering*, *61*, 381-384.

Winkler-Schwartz, A., Bissonnette, V., Mirchi, N., Ponnudurai, N., Yilmaz, R., Ledwos, N., Siyar, S., Azarnoush, H., Karlik, B., & Del Maestro, R. F. (2019). Artificial intelligence in medical education: Best practices using machine learning to assess surgical expertise in virtual reality simulation. *Journal of Surgical Education*, *76*(6), 1681-1690.

Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data Mining: Practical machine learning tools and techniques.* Burlington, MA: Morgan Kaufmann.

Zaharia, M., Chen, A., Davidson, A., Ghodsi, A., Hong, S. A., Konwinski, A., Murching, S., Nykodym, T., Ogilvie, P., Parkhe, M., Xie, F., & Zumar, C. (2018). Accelerating the machine learning lifecycle with MLflow. *IEEE Data Engineering Bulletin*, *41*(4), 39-45.

Zhou, J. (2017). Data mining for individual consumer credit default prediction under e-commence context: A comparative study. In *Proceedings of the International Conference on Information Systems.*

Appendix A: Results of SMLR Study for Journals

Step	Indicator		Described in papers			Positive example
Model initiation	Problem statement		100.00%	(35/35)		Abbasi et al. (2012)
	Data gathering		88.57%	(31/35)		Lin et al. (2017)
	Data distribution		82.86%	(29/35)		Wang et al. (2013)
	Sampling		37.14%	(13/35)		Samtani et al. (2017)
	Data quality		37.14%	(13/35)		Dong, Liao, & Zhang (2018)
	Data preprocessing methods		71.43%	(25/35)		Pant & Srinivasan (2013)
	Feature engineering and vectorizing		62.86%	(22/35)		Twyman et al. 2015
Performance estimation	Parameter Optimization	Search Space	8.57%	(3/35)		Martens, Provost, Clark, & Junqué de Fortuny (2016)
		Search Algorithm	11.43%	(4/35)		Martens et al. (2016)
	Data split		94.29%	(33/35)		Lash & Zhao (2016)
	Algorithm		100.00%	(35/35)		Li et al. (2016)
	Sampling		5.71%	(2/35)		Kitchens, Dobolyi, Li, & Abbasi (2018)
	Performance metric (reasoned)		42.86%	(15/35)		Shi, Guan, Zurada, & Manikas (2017)
	Performance evaluation		68.57%	(24/35)		Cui et al. (2012)
Model deployment	Data used		2.86%	(1/35)		Abbasi et al. (2018)
	Model validity	Continuous Improvement	2.86%	(1/35)		Abbasi et al. (2018)
		Robustness	8.57%	(3/35)		Mo, Sarkar, & Menon (2018)

Figure A1. Overview of Supervised Machine Learning Report Card Steps, their Problem Characteristics and Choices, and their Documentation in the Analyzed Journal Publications

Appendix B: Results of SMLR Study for Conferences

Step	Indicator		Described in papers			Positive example
Model initiation	Problem statement		100.00%	(86/86)		Kowatsch & Maass (2018)
	Data gathering		87.21%	(75/86)		Ram, Wang, Currim, & Currim (2015)
	Data distribution		70.93%	(61/86)		Huang, Nambisan, & Uzuner (2010)
	Sampling		5.81%	(5/86)		Stange & Funk (2015)
	Data quality		80.23%	(69/86)		Riekert et al. (2017)
	Data preprocessing methods		77.91%	(67/86)		Pröllochs, Feuerriegel, & Neumann (2015)
	Feature engineering and vectorizing		79.07%	(68/86)		Baumann, Lessmann, Coussement, & De Bock (2015)
Performance estimation	Parameter Optimization	Search Space	13.95%	(12/86)		Tafti & Gal (2018)
		Search Algorithm	13.95%	(12/86)		Staudt, Rausch, & Weinhardt (2018)
	Data split		96.51%	(83/86)		Chatterjee et al. (2018)
	Algorithm		100.00%	(86/86)		Tripathi & Kaur (2018)
	Sampling		6.98%	(6/86)		Lüttenberg, Bartelheimer, & Beverungen (2018)
	Performance metric (reasoned)		51.16%	(44/86)		Blanc & Setzer (2015)
	Performance evaluation		40.70%	(35/86)		Geva and Oestreicher-Singer (2013)
Model deployment	Data used		1.16%	(1/86)		Laing & Kühl (2018)
	Model validity	Continuous Improvement	1.16%	(1/86)		Seebach et al. (2011)
		Robustness	18.60%	(16/86)		Goby, Brandt, Feuerriegel, & Neumann (2016)

Figure B1. Overview of Supervised Machine Learning Report Card Steps, their Problem Characteristics and Choices, and their Documentation in the Analyzed Conference Publications

Appendix C: Comparison to Data Science Processes

Table C1. CRISP-DM and the Report Card

CRISP DM phases and tasks	Related report card choices / characteristics
Business understanding	
Determine business objectives	Model initiation—problem statement
Assess situation	N/A
Determine data mining goals	Model initiation—problem statement
Produce project plan	N/A
Data understanding	
Collect initial data	Model initiation—data gathering
Describe data	Model initiation—data distribution
Explore data	Model initiation—data distribution
Verify data quality	Model initiation—data quality
Data preparation	
Select data	Model initiation—sampling
Clean data	Model initiation—data quality
Construct data	Model initiation—data preprocessing methods
Integrate data	Model initiation—data gathering
Format data	Model initiation—feature engineering and vectorizing
Modeling	
Select modeling technique	Model training—algorithm
Generate test design	Performance estimation—data Splitting method
Build model	Model training—algorithm/performance estimation—parameter optimization
Assess model	Model testing—performance metric
Evaluation	
Evaluate results	Model testing—performance evaluation (benchmarks)
Review process	N/A
Determine next steps	N/A
Deployment	
Plan deployment	Model deployment—data used
Plan monitoring and maintenance	Model deployment—model validity (continuous improvement / robustness)
Produce final report	N/A
Review project	N/A

Table C2. Team Data Science Process and the Report Card

MTDSP stages	Related report card choices / characteristics
Business understanding	
Define objectives	Model initiation—problem statement
Identify data sources	Model initiation—data gathering
Data acquisition and understanding	
Ingest the data	Model initiation—data gathering
Explore the data	Model initiation—data distribution, model initiation—data quality
Set up a data pipeline	N/A
Modeling	
Feature engineering	Model initiation—data preprocessing methods, model initiation—feature engineering and vectorizing
Model training	Model training—algorithm, performance estimation—data splitting method, model testing—performance metric
Suitability for production	Model testing—performance evaluation (benchmarks)
Deployment	
Operationalize a model	Model deployment
Customer acceptance	
System validation	N/A
Project hand-off	N/A

About the Authors

Niklas Kühl is leading the Applied AI in Services Lab at the Karlsruhe Service Research Institute (KSRI) at the Karlsruhe Institute of Technology (KIT). He also works as a Data Scientist for IBM Global Business Services in diverse industry projects. His goal is to continuously facilitate the exchange between academia and industry by publishing highly relevant work for both worlds. He has been researching applications in machine learning for the past five years in different domains. He did his PhD in designing and implementing a machine learning based tool capable to automatically identify customer needs in social media data, such as for e-mobility needs expressed via Twitter. Currently, he and his team of nine researchers are working on different AI solutions within industrial services, sales forecasting, production lines and many other examples.

Robin Hirt is the Co-Founder and Co-CEO of prenode GmbH, a German-based company that provides solutions for decentralized machine learning and has been acknowledged by Gartner as one of the few companies that enable privacy-preserving cross-organizational machine learning with its unique technologies. He is also an associated researcher at the Applied AI in Services Lab at the Karlsruhe Service Research Institute at the Karlsruhe Institute of Technology (KIT). He did his PhD in designing and developing algorithms and systems for realizing privacy-preserving horizontal and vertical federated machine learning and collaborated with the MIT-IBM AI Lab as a visiting researcher.

Lucas Baier is a Research Associate in the Applied AI in Services Lab at the Karlsruhe Service Research Institute (KSRI) which is located at the Karlsruhe Institute of Technology (KIT). He holds a bachelor and a master's degree in Industrial Engineering from KIT. His research is concerned with solving the challenges of deployed machine learning services in real-world settings with a special focus on concept drift. He has developed various concept drift handling algorithms for domains such as mobility or e-commerce and his ideas have been implemented in various research projects funded by both industry and government.

Björn Schmitz is a Data Science Manager and Senior Data Scientist in the cognitive and analytics practice of IBM Services. In his role, he supports clients in diverse industries and markets in designing, developing and operating solutions that utilize Machine Learning and Artificial Intelligence. He acquired a PhD in Information Systems at the Karlsruhe Institute of Technology (KIT) focusing on estimating cost uncertainties in industrial full-service contracts. Since then, he has collaborated closely with the scientific community in various research projects and is currently working as a lecturer for computer vision at KIT.

Gerhard Satzger is Director of the Karlsruhe Service Research Institute, an "industry-on-campus" initiative focused on innovation in IT-based services, and professor at the Karlsruhe Institute of Technology (KIT) in Germany. His research is concerned with conceiving and developing digital services and corresponding business models with a particular focus on human-centered design for services as well as on creating value from data via AI. Drawing both on academic qualifications and experience in IS as well as on a multi-year industry track record in various national and international roles at IBM, he strives to drive effective innovation by collaboration between industry and academia. His work is published in IS and service journals and finds its application in projects with various industry partners.