A Survey of Machine Learning in Credit Risk

Preprint · May 2020

DOI: 10.13140/RG.2.2.14520.37121

1 author:

Joseph Breeden
Prescient Models LLC
62 PUBLICATIONS **387** CITATIONS

SEE PROFILE

A Survey of Machine Learning in Credit Risk

Joseph L Breeden
Prescient Models LLC
breeden@prescientmodels.com

May 30, 2020

Abstract

Machine learning algorithms have come to dominate some industries. After decades of resistance from examiners and auditors, machine learning is now moving from the research desk to the application stack for credit scoring and a range of other applications in credit risk. This migration is not without novel risks and challenges. Much of the research is now shifting from how best to make the models to how best to use the models in a regulatory-compliant business context.

This article seeks to survey the impressively broad range of machine learning methods and application areas for credit risk. In the process of that survey, we create a taxonomy to think about how different machine learning components are matched to create specific algorithms. The reasons for where machine learning succeeds over simple linear methods is explored through a specific lending example. Throughout, we highlight open questions, ideas for improvements, and a framework for thinking about how to choose the best machine learning method for a specific problem.

Keywords: Machine learning, artificial intelligence, credit risk, credit scoring, stress testing

1 Introduction

The greatest difficulty in writing a survey of machine learning (ML) in credit risk is the extraordinary volume of published work. Just in the area of comparative analyses of machine learning applied to credit scoring, dozens of articles can be found. The goal of this survey cannot be to index all work on machine learning in credit risk. Even listing all of the worthy articles is beyond the attainable scope.

Rather, this survey seeks to identify the major methods being used and developed in credit risk and to document the breadth of application areas. Most importantly, this article seeks to provide some intuitive insights on why certain methods work in specific areas,. When does machine learning work better than linear methods only because it was a quicker path to an answer versus discovering something about the problem that was undiscoverable with traditional

methods? Further, as a result of this research, we hope to identify some areas of investigation that could be fruitful but have not yet been fully explored.

In attempting to provide a balanced view of the state of machine learning, some passages herein may take a tone that machine learning is "much ado about nothing". In other discussions, we are clearly singing the virtues of deep learning with discussions of ensemble methods for robustness, deep learning to analyze alternate data, and techniques for modeling the smallest data sets. Machine learning can be seen to be clearly successful in some cases and disturbingly overblown in others, bringing new innovations in important areas and painfully rediscovering old methods in some cases, and overall has made significant strides toward mainstream application while still having significant challenges to overcome.

The article begins with a definition of machine learning intended in part to limit the scope of this survey to a manageable breadth. The next section offers a modeling taxonomy based upon defining data structure, architectures, estimators, optimizers, and ensembles. From this perspective, much machine learning research is a human-based search of the meta-design space of what happens when you mix and match among those categories. Then, Section 4 provides a discussion of the many application areas within credit risk and some of the model approaches found within each. Section 5 reviews a specific example of testing many machine learning algorithms to illustrate the differences relative to traditional methods. Section 6 follows with a discussion of significant challenges in creating machine learning models and using them in business contexts. The conclusion pulls these thoughts together to highlight areas where future comparative studies could provide significant value to practitioners.

2 What is Machine Learning?

We tend to think of statistical models and linear methods as something other than machine learning, and yet simple linear regression can take on unbounded complexity through factor variables, spline approximations, interaction terms, and input massive numbers of descriptive variables through dimension reduction methods such as singular value decomposition. The heart of many machine learning algorithms is a search or optimization method that was pioneered decades or centuries ago in other contexts. Bagging, boosting, and random forests harken back to earlier work on ensemble methods [66, 213].

Harrell [126] proposes a distinction between statistical modeling and machine learning:

- Uncertainty: Statistical models explicitly take uncertainty into account by specifying a probabilistic model for the data.
- Structural: Statistical models typically start by assuming additivity of predictor effects when specifying the model.
- Empirical: Machine learning is more empirical including allowance for

high-order interactions that are not pre-specified, whereas statistical models have identified parameters of special interest.

The above items carry other implications. For example, search-based methods such as Monte Carlo simulation, genetic algorithms, and various forms of gradient descent usually do not provide confidence intervals for the parameters, and correspondingly are usually considered as machine learning. Ensemble methods where multiple models are combined are generally considered to be machine learning, even when the constituent models are statistical. One might also say that traditional statistical methods rely on analyst selection of input features and interaction terms whereas machine learning methods emphasize algorithmic selection of features, discovery of interaction terms, and even creation of features from raw data.

Drawing the line between machine learning and traditional modeling is challenging for the best scientific linguist. Practically speaking, machine learning seems like it should include models that emphasize nonlinearity, interactions, and data-driven structures and exclude simple additive linear methods with moderate numbers of inputs. The distinction may be more in the specific application than the method used. For example, an artificial neural network could be dumbed down to a nearly-linear adder, and common logistic regression can incorporate almost all the learnings from a sophisticated machine learning algorithm through artful use of binning, interaction terms, and segmentation.

Some methods might be viewed as intermediates, like transitional species in evolution. (The author recognizes that "transitional species" is a misnomer in evolutionary taxonomy, but the perspective is not inappropriate here.) Forward stepwise regression or backward stepwise regression automate feature selection while being statistically grounded. Principal Components Analysis (PCA) is an inherently linear, statistical method of dimensionality reduction via eigenvalue estimation, whereas other dimensionality reduction methods lean much more to machine learning. One of the greatest strengths of neural networks is as a nonlinear dimensionality reduction algorithm.

Within this attempted dichotomy, many machine learning techniques are rapidly taking on statistical rigor. This maturing process is what we see in any field where rapid advances are followed by a team of scientists filling in theoretical and technical details.

Many of the most public successes of machine learning are coupled with big data, massive data sets that allow equally massive parameterizations of the problem so that the optimal transformations of the inputs and dimensionality reduction are learned from the data rather than via human effort. However, machine learning should not be viewed as synonymous with big data. Some machine learning methods appear well suited to very thin data sets where even linear regression struggles. Eventually, as we truly move into human-style AI, the ability to learn from a single event in the context of a 'physical' model of the world would show the power of machine learning with the smallest of data.

In credit risk, we are often stuck with small data. This was observed in the credit scoring survey by Lessmann, et. al. [175] where only five of the

48 papers surveyed had 10,000 accounts or more to test, quite small samples compared to the big data headlines, but this is often the reality of credit risk modeling. For many actual portfolios, number of accounts * loss rate = very few training events. Even in subprime consumer lending where loss rates are higher, only the largest lenders have had the data sets needed to apply the most data-hungry techniques like deep learning, or so it seems. However, machine learning is succeeding in credit risk modeling even on smaller data sets, apparently by emphasizing robustness and simpler interactions as opposed to the extreme nonlinearities in big data contexts such as image processing [162], voice recognition [120], and natural language processing [69].

Machine learning is generating successes in credit risk, although less dramatically in well-worn domains like prime mortgages. The biggest wins appear to be in niche products, alternate channels, serving the underbanked [5], and alternate data sources. A well-trained machine learning algorithm may be preprocessing deposit histories [2], corporate financial statements, twitter posts [199], social media [97, 24, 10] or mobile phone use [38, 232] to create input factors that eventually feed into deceptively simple methods like logistic regression models.

Also, in looking at applications of machine learning to credit risk, we must look beyond predicting probability of default (PD). One of the great early success stories of ML was in fraud detection [110]. Anti-fraud [297], anti-money laundering [254, 273, 20, 217, 178], and target marketing applications [180, 22] make heavy use of machine learning, but are outside the boundaries we will draw here around credit risk applications. Still we must consider applications to predicting exposure at default, recovery modeling, collections queuing, and asset valuation, to name a few.

The following sections aim to provide an introduction to the literature on machine learning methods, applications in credit risk, what makes machine learning work, and what are the challenges with employing machine learning in credit risk.

3 Machine Learning Methods

Providing an exhaustive list of machine learning methods would not be possible, particularly when we look beyond credit scoring to the broader applications of machine learning across credit risk modeling. One of the greatest challenges of creating any list of models is the difficulty in defining a model. The name given to a model typically represents a combination of data structure, architecture, estimator, selection or ensemble process, and more. Authors may swap out one estimator for another or add ensembles on top and describe it as a new model. This abundant hybridization leads to exponential growth in the literature and model names. Finding the right combination is, of course, very valuable, but the human search through this model component space with publications as measurement points is more than can be cataloged here.

In this section we will identify key sets of available components behind the models and then categorize some of the most studied models according to the

components used. Of course, each of these lists can never be complete. They are intended only to be representative.

3.1 Data Structures

Choosing a data structure is the first step in either statistical modeling or machine learning. That model must be chosen to align with the data being modeled. A range of target variables are possible in credit risk and those variables can be observed with different frequencies and aggregation, depending upon the business application.

Table 1 lists some of the outputs one might wish to model in the domain of credit risk. Items like PPNR [183] and Prepayment [240] might not seem like credit risk tasks, but when they are modeled divorced from credit risk modeling, the result can be conflicting predictions leading to nonsensical financial projections. Taking a consistent, coordinated perspective of all account outcomes and performance as in competing risk architectures and models [168, 92] is the best hope of predicting pricing and profitability.

Even deposit modeling can leverage very similar methods and works best when a total customer view is taken. Deposit balances are a potentially valuable input to credit risk models, but are not always categorized as credit risk targets. Anti-fraud, anti-money laundering, and target marketing were considered as separate from credit risk, because they are not part of the analysis of an active customer relationship, although even here the boundaries are weak.

Target Variables
Loss Balance
PD, EAD, LGD [and PA (Probability of Attrition)]
Prepayment
Pre-provision net revenue (PPNR)
Asset Values
Deposit Balance*
Time Deposit Renewal Rate*

Table 1: List of target variables that can be modeled in credit risk applications. The items marked with * are likely candidates for using the same methods as the other items, but not strictly considered credit risk issues.

For any target to be modeled, a decision must be made on the aggregation level and performance to be predicted. Table 2 lists the most common answers. Each type of data usually has a corresponding literature. Econometric models [85, 277] focus on time series data, either for a portfolio or segments therein; Age-Period-Cohort models [113, 195, 106] are applied to vintage performance time series; survival models [256, 152] and panel data models [283, 140] are applied to account performance time series; and the large literature on credit scoring [257, 14] focuses mostly on account outcomes, using a single binary performance indicator for each account.

Data Types
Segment Time Series
Vintage Performance Time Series
Account Performance Time Series
Account Outcomes

Table 2: List of data types that can be modeled in credit risk applications.

By starting the discussion with target variables, what follows is immediately focused on supervised learning. The assumption is that unsupervised learning techniques might be used to create input factors. Many forms of dimensionality reduction and factor creation can be conducted using unsupervised methods. PCA and most segmentation methods can be considered unsupervised learning. However, a credit risk model will ultimately always finish with a supervised learning technique.

3.2 Architectures

Once the problem is stated as a target variable to be predicted and its data structure as in Tables 1 and 2, an architecture must be chosen for the problem, Table 3. This is the point where the distinction between traditional methods and machine learning can appear.

Additive effects refers to regression approaches [153, 131]. Additive fixed effects includes the use of fixed effects (dummy variables), again in a regression approach, panel model, etc.

State transition models, [206, 26] (also known as grade, rating, or score migration models depending upon whether they are applied to delinquency states, risk grades, agency ratings, or credit scores) are all variations on Markov chains [207]. Roll rate models [89] capture the net forward transition of a state transition model and are used throughout credit risk modeling. Generally, this architecture involves identifying a set of key intermediate states and modeling the transitions between those states and to the target state. Usually the target is a terminal state like charge-off or pay-off.

Going beyond the above architectures leads more into the realm of machine learning, although there are again few fixed boundaries. Convolutional networks [162], feed-forward networks [253, 15], and recurrent neural networks [182] are all kinds of artificial neural networks and are just a few of the many structures being tested.

Whenever the nonlinearity of a problem exceeds the flexibility of the underlying model, segmenting the analysis is a common solution. The more nonlinear the base model, the less segmentation is required. Traditional logistic regression models may actually be a collection of many separate regression models applied to different segments, whereas a neural network or decision tree may use a single model.

Some models are themselves segmentation engines. Methods such as support

vector machines (SVM) [266] use hyperplanes or other structures to segment the parameter space. Decision trees [226] can also be viewed as a high-dimensional segmentation technique and are employed in a variety of machine learning approaches. Nearest neighbor methods [71, 130] are difficult to classify in this architectural taxonomy, but seem closer to these than the rest.

Architecture
Additive Effects
Additive Fixed Effects
Convolutional Network
Clustering
Feed-forward Network
Fuzzy Rules and Rough Sets
Nearest Neighbors
Recurrent Neural Network
Segmentation
State Transitions
Trees

Table 3: List of internal architectures used in modeling.

Fuzzy rules are used to capture uncertainty directly in the forecasting process [202] and are often combined with other methods [220]. Rough sets [219] can be seen as having a similar objective of considering the vagueness and imprecision of available information, but using a different theoretical framework.

Recurrent neural networks (RNN) are used primarily to model time series data. By making the forecast from one period and input to the network for the next period, they are effectively a nonlinear version of vector ARMA models ((Multivariate Box-Jenkins) [197, 177]. Long short term memory (LSTM) networks apply a specific architecture to the recurrent neural network framework in order to scale and refine the use of memory in the forecasting.

Overall, many architectures can be used in time series forecasting. The same lagged inputs used in linear distributed lag models [11, 291] can be used as inputs to machine learning methods. To reduce the dimensionality of the problem and aid visualization, optimal state space reconstruction can be used, also known as the method of delays [216, 234, 52, 165].

Convolutional neural networks (CNN) being applied to consumer transaction data [167] seems far from the leading applications in image processing, but many more applications of CNNs are likely, particularly with recent advances incorporating rotational [78, 65] and other symmetry transformation to increase the generalization power of CNNs.

Not shown is the list of possible inputs, because this would be too extensive.

3.3 Estimators and Optimizers

The primary purpose of this modeling taxonomy is to illustrate that, for example, a genetic algorithm is not a model. Practitioners, both experienced and novice, often use sloppy terminology confusing data structures, architectures, and optimizers. Here we illustrate that many different estimators and optimizers can be applied in an almost mix-and-match fashion across the range of architectures. By clearly identifying the components of a model, researchers can find opportunities for creating useful hybrids.

The literature also attempts to carefully distinguish between estimators and optimizers. In simple terms, estimators all rely on a statistical principle to estimate values for the model's parameters, usually with corresponding confidence intervals and statistical tests in the traditional statistical framework. Optimizers generally follow an approach of specifying a fitness criteria to be optimized. As parameter values are changed, the fitness landscape can be mapped. Each optimizer follows a specific search strategy across that fitness landscape. Of course, here again it can be difficult to draw bright lines between these categories as estimators and optimizers can take on properties of each other.

Table 4 lists some of the many methods used to estimate parameters or even meta-parameters (architectures) of a model. Items such as back propagation are specific to a certain architecture, e.g. back prop as a way to revise the weights of a feed-forward neural network. Most, however, can be applied creatively across many architectures for a variety of problems.

Estimators
Least Squares
Maximum Likelihood [218]
Partial Likelihood [72]
Bayes Estimator [35]
Method of Moments [196, 121, 98]

Table 4: List of statistical estimators used in modeling.

Maximum likelihood estimation is the dominant statistical estimator, which is, for example, behind the logistic regression estimation that is ubiquitous in scoring and many other contexts. Least squares estimation predated maximum likelihood but can be derived from it. Partial likelihood estimation was a clever efficiency developed for estimating proportional hazards models without estimating the hazard function parameters needed in the full likelihood function.

Aside from some deep philosophical issues, Bayesian methods are particularly favored when a prior is available to guide the solution. Markov chain Monte Carlo (MCMC) starts with a Bayesian prior distribution for the parameters and uses a Markov chain to step toward the posterior distribution given the data, somewhat like a correlated random walk.

In data-poor settings, Bayesian methods provide a powerful mechanism for combining expert knowledge from the analyst with available observations to obtain a more robust answer. Computing a batting average in baseball is an

easy way to illustrate this. Someone who has never swung could be assumed to have a 50/50 chance of hitting the ball, a .500 average. After their first swing, a miss would take his batting average to .333 and a hit would take it to .667. With a maximum likelihood estimation, the best fit to the data would be .000 for a miss and 1.000 for a hit, which seems less helpful until more observations are acquired. This is Laplace's Rule of Succession. Not coincidentally, Laplace also formulated Bayes' Theorem.

With method of moments, the moments of the distribution are expressed in terms of the model parameters. These parameters are then solved by setting the population moments equal to the sample moments.

Linear programming and quadratic programming are methods for incorporating constraints. Many other constrained optimization methods exist, such as Lagrange multipliers which provide a mechanism for adjusting the fitness function to incorporate penalty terms.

Optimizers
Gradient Descent
Simulated annealing [159] Back Propagation [128]
Reinforcement Learning
Genetic Algorithms [115]
Evolutionary Computation [151]
Genetic Programming [161]
Markov Chain Monte Carlo [108]
Kalman Filter [111]
Linear Programming [265]
Quadratic Programming [36]

Table 5: List of optimizers used in modeling.

Gradient descent can be accomplished via several specific algorithms, but it generally refers to computing the local gradient of the fitness landscape at a test point and stepping in the direction with the steepest slope, hopefully toward the desired minimum. Back propagation is gradient descent in the context of a neural network where the gradient is computed for each node's parameters. Reinforcement learning is the more general concept of adjusting parameters, usually in a neural network context, based upon new experiences. Kalman filters are an optimal update procedure for linear, normally distributed models, which could be thought of as a subset of reinforcement learning.

Genetic algorithms, evolutionary computation, and genetic programming are all modeled on evolutionary principles. In an optimization setting, mutation operations with survivor selection are equivalent to stochastic gradient descent. Including cross-over between candidates works if symmetries exist in the fitness landscape such that sets of parameters form a useful sub-solution within the model.

Also not shown are the many estimation methods developed to handle correlated input factors such as Lasso [258] and ridge regression [133].

Of course, many of these concepts can be combined. Stochastic back propagation and stochastic gradient descent [41] are widely used. Simulated annealing can be thought of as combining the stochastic gradient descent concept with the multiple candidate solutions approach of evolutionary methods. Bayesian methods can be combined with many other optimization approaches, such as Bayesian back propagation [56] or MCMC as described above.

3.4 Heterogenous Ensembles

Ensemble modeling is actually a general technique that can combine forecasts from different model types. "Triangulation" has been a common technique over several decades for portfolio managers to create loss forecasts by comparing the outputs of several different models, each with different confidence intervals and known strengths and weaknesses. Voting is largely a formalization of what managers have been doing intuitively, with several interesting variations [263, 166].

Ensemble modeling [73, 66, 213, 79, 222] has been in use well before the burst of activity in machine learning, but has quickly proven itself to be a valuable addition to most any machine learning technique, particularly in credit risk [271]. Most research into ensemble modeling can be split between homogenous methods, where multiple models of the same type are combined to create better overall forecasts and heterogenous methods where any types of models can be combined. We also consider a third category of hybrid ensembles where two complimentary model types are integrated via mechanisms more specific to the methods than in the generic heterogenous ensemble approaches.

For an ensemble to be more effective than the individual contributors, Hansen & Salamon [123] showed that the individual models must be more accurate than random and the models must not be perfectly correlated. In other words, we cannot create useful forecasts from a collection of random models, and the best ensembles have constituents that have complimentary strengths.

Ensemble modeling seems particularly well suited to credit risk, because of the typically limited data sets available. Although the underlying dynamics can be quite complex and explainable with a rich variety of observed and unobserved factors, the actual data available may support models of only limited complexity. Even though many factors can be important, issues of multicolinearity [205] can limit the modeler's ability to include more than a few factors and is often a deeper problem than is generally recognized [118, 49]. Dimensionality reduction methods such as singular value decomposition, principle components analysis, [150] and projection pursuit [103, 102, 146] are methods to address multicolinearity, but they do not address the sensitivity to outliers and overfitting questions as well as the full nonlinearity treatment available in machine learning.

The basic principle behind ensemble modeling is that different models can capture different aspects of the data. This can provide robustness to outliers and anomalies [281] as well as which factors are included in the modeling. Both theoretical [123, 163, 141] and empirical studies have shown that this diversity

when obtained for individually accurate predictors has significant out-of-sample advantages.

Heterogenous Ensemble Methods		
Binary	Categorical	Continuous
Plurality Voting	Plurality Voting	Average
Sum rule	Majority voting	Median
Product rule	Sum rule	Confidence weighted
Stacking	Product rule	Stacking
	Amendment vote	
	Runoff vote	
	Condorcet count	
	Pandemonium	
	Borda count	
	Single transferable vote	
	Stacking	

Table 6: List of methods used to combine forecasts in heterogenous ensemble modeling.

Table 6 lists some of the methods used for combining forecasts in ensemble modeling of potentially heterogenous models. Many of these methods were developed from the perspective of choosing from several possible categories [263]. In a broader credit risk context, we can have situations with binary outcomes, e.g. default or not; multiple (categorical) outcomes, e.g. transition to different states; or continuous outcomes, e.g. forecasting a default rate.

Combining forecasts for binary events can be performed with several methods. Voting methods are the most common, where each constituent model gets one vote. Plurality voting is the simplest of these, where the outcome with the most votes is chosen. If the constituent models produce probabilities or some kind of fractional forecast, then each constituent model can divide its vote proportionally between the two outcomes, which are then summed. Classification methods can be modified to produce probabilities to facilitate their use more broadly [221, 164]. In the product rule, these fractional votes are multiplied, which means extremely confident models can dominate an outcome.

When predicting multiple possible outcomes (categorical outputs), the above methods can be generalized easily. If addition, majority voting is different from plurality voting, where one outcome must have a majority of the votes. In no outcomes have a majority, the least favored outcome is removed and a majority is sought among the remaining outcomes. A run-off vote is a simple extension of the majority voting process until a single outcome remains.

Amendment voting starts with a majority vote between the first two candidate outcomes. The most favored is tested against the next candidate until one outcome remains. However, this procedure can be biased depending upon the sequence of comparisons.

The Condorcet count performs pairwise comparisons of all outcomes. The

favored outcome from each comparison receives one point and the outcome with the most points is chosen. Although complex, this has many favorable properties.

In Selfridges Pandemonium [241] method each model would choose one outcome, but that vote is stated with a confidence. Those weighted votes are summed to choose a winner, meaning that model confidence intervals become important.

If the constituent models cannot assign a probability to all possible outcomes, as needed for sum rule and product rule, but the models can rank the outcomes, then ranked voting can be used. The outcome can be chosen by mean rank [40], median rank, or a trimmed mean or median rank.

Single transferable vote also works from ranks, although not every model must rank every outcome. If one outcome has a majority of the top ranks, it is chosen. If not, the least preferred outcome is eliminated and the top ranks are re-aggregated. The procedure continues until one outcome receives a majority.

Beyond voting, one could imagine creating a model of models. In a linear regression context, this does not introduce any new information beyond the initial estimate. However, with stacking [282] the initial models are trained on a subset of the total data. Then a secondary, often linear regression, model is trained on the hold-out sample, considering model accuracies and correlations. Machine learning methods can also be used to create models of models [259].

One advantage of ensembles is the ability to create confidence measures for classification models, although direct, single-model approaches are also available [224].

For continuous-valued predictions, averages, medians, trimmed values, and stacking all apply. Continuous forecasts are often or in best practice should be accompanied by confidence intervals. Therefore, weighted averages or some method that incorporates those confidences would be preferable.

3.5 Homogeneous Ensembles

Any method for combining heterogenous model predictions can of course be applied to homogenous models, where multiple models of the same type are built to be combined. However, some methods have been specifically designed to work with homogenous ensembles.

3.5.1 Bagging

Bootstrap aggregation (bagging) [53, 173, 179] is a simple process of subsampling the available training data with replacement. Considering the typically limited size of the training samples in credit risk, the subsets can be 75% of the available data and upward. Bagging can be used with any model type and the resulting forecasts combined as described for heterogenous models, although the sum rule is used most often [160].

For random subspace modeling [132], a random sample of the available input factors is drawn for each model. This could also be done sequentially determin-

istic fashion, where the strongest explanatory variable from the first model is excluded from the next model in order to find structure among other variables, and so forth. The first application was for creating decision trees, leading to the literature on random forecasts, but the technique is generic to any model type.

Rotation forests [230] follow the random forests idea, but all of the data is used each time. Instead, a rotation of the axes in the data space for a subset of input factors is performed prior to building each model. This has the effect of testing many different projections for predictive ability.

Similar to the bagging concept is to use all of the training data each time, but different initial conditions for the parameter estimates. For model types such as neural networks [66] or decision trees [9] that employ some form of learning or gradient descent, this can also create a robust ensemble.

3.5.2 Boosting

Conceptually, one could say that boosting is a process of building subsequent models on the residuals of previous models, though for model types that have no explicit measure of residuals [236, 235]. AdaBoost [99] reweights the training data with each iteration to emphasize the points that were not predicted as well in the previous iterations. Gradient boosting [100] computes the gradient of a fitness function in order to provide weights to each model trained. Stochastic gradient boosting [101] combines bagging with gradient boosting, building an ensemble of ensembles where different gradient boosted ensembles are built for each data sample. These methods can also be applied to any model type. The popular XGBoost package (eXtreme Gradient Boosting) [63] is a highly optimized version of gradient boosting.

Many studies have been performed to compare ensemble methods [204, 271], but the winning approach probably depends upon the specific problem and data set. For example, gradient boosting has been reported to be more susceptible to outliers.

3.6 Hybrid Ensembles

A very large area of research involves creating hybrid models, where specific model types are chosen that are intended to be integrated in non-trivial ways, usually via an algorithm specifically tailored to the models chosen and to the application area. This is different from heterogenous ensembles where the forecasts are combined via one of the voting schemes in Table 6. Instead, hybrid ensembles create an architecture that leverages the specific traits of the models. The criterion for success is not about choosing which models are most orthogonal and accurate [123]. Rather, it involves combining models that may (1) use different data sources, (2) predict over different forecast horizons, or (3) identify different problem structures. So the models are inherently complimentary, often making measures like orthogonality or comparative accuracy undefined.

A classic example in credit risk is the use of roll rate models [89] for portfolio forecasting for the first six months combined with vintage models [45] for the

longer horizon forecasts. In this case, the analyst would usually switch from one model to the other at a certain forecast point or use a weighting between the models that is a function of forecast horizon. Some version of this approach has been in use for decades, because roll rates are known to be accurate for the short term and vintage models for the long term.

The list of hybrid ensembles (or hybrid models) in the literature is far too great, but these provide a few examples: decision trees and neural networks [171], support vector machines and neural networks [70, 6], naive Bayes and support vector machines [201], a classifier ensemble with genetic algorithms [294], and genetic algorithm and artificial neural networks [214]. Some authors provide surveys of collections of hybrid ensembles generally [17] or for specific application areas such as bankruptcy prediction [269]. Hybrids combining age-period-cohort (APC) models [195, 113, 136, 106] with origination scores, behavior scores, neural nets, or gradient boosted trees were created specifically to better solve the short economic cycle data described above [46, 50, 51]

4 Applications in Credit Risk

Machine learning methods received early attention from researchers, but adoption into operational contexts has been understandably cautious for reasons to be discussed in Section 6. The earliest experiments were primarily in fraud detection, credit scoring [188, 77, 129, 279, 288], corporate bankruptcy and default forecasting [209]. As machine learning methods have matured along the lines described above, parallel efforts occurred in the application of those techniques to areas of credit risk, resulting in a wide range of new applications.

4.1 Credit Scoring

Credit scores were created to predict the relative risk of default among borrowers [68, 176]. Their success as compared to human judgment was so great that they became part of the standard credit bureau offering and an essential part of the lending ecosystem. These bureau scores have been developed and refined over decades and are essentially the result of an optimization process where the disparate and complex consumer performance history has been linearized into factors what fit well into a logistic-regression model. This would seem to be the same kind of work done automatically by machine learning, but historically done through human intuition and experimentation.

Anecdotally, developers of modern bureau scores are said to use machine learning methods to search for additional interaction terms and nonlinearities. Those lessons are taken back to the original logistic regression-based model to create small improvements, but the advances available from machine learning appear to be small compared to the decades of human optimization already performed. However, Hand & Henley [129] showed that even small enhancements to credit score performance can have significant returns.

In principle any institution can purchase data from the bureaus similar to what goes into creating the bureau scores and do a head-to-head test of in-house machine learning model to bureau score. In any such test, the in-house model has a great advantage in that the target is known. When developing a bureau score, the model is attempting to predict default without knowing what product the consumer will be offered, or if default will come in the absence of new loans and based purely on existing loans. An in-house model is typically built to predict the outcome of offering a new loan of a specific type and perhaps even incorporating the terms of that loan. Fair comparisons are difficult, but perhaps unimportant. A developer creating an in-house model can jump straight to sophisticated modern methods, either taking the bureau score as an input or starting fresh, in each case bypassing the decades of labor put into the original bureau scores.

Machine learning in credit scoring is not new. Comparative surveys can be found as far back as 1994 [228]. New comparative analyses continue to appear as new methods are developed and more data becomes available. One of the most complete surveys was conducted by Lessmann, et. al. (2015) [175] in which they noted the irony that most published work on machine learning in credit scoring leveraged only very small data sets for comparing "big data" machine learning methods. Lessmann, et. al. sought to resolve that shortcoming by testing multiple methods on multiple, larger data sets. These surveys are useful both in bringing the readers up to date on the latest methods and in suggesting which methods could be best, but no single method wins in all studies [21]. The obvious conclusion is that not all data sets have similar structures, and the analyst can still expect to test several approaches to find which is most effective on a specific data set. Similarly, researchers need to be careful to avoid publishing conclusions that one method is better than another based only upon one data set over one time period.

4.1.1 Neural Networks

Neural networks are one of the most extensively tested methods for credit scoring and one of the first machine learning methods employed [149, 77, 268, 279, 190]. They can function like a nonlinear version of dimension reduction algorithms such as principal components analysis or as factor discovery methods in deep learning contexts. They offer additive and comparative interaction terms between variables. On the most basic level, neural networks provide a nonlinear response function between input and output. With enough training data, these attributes can be a powerful combination.

The first challenge with applying neural networks is in choosing an architecture. In theory, with enough data, a fully connected, feed-forward neural network should be able to learn its own architecture, but reality is more challenging. Some of the biggest success stories in using deep learning neural networks required vast amounts of training to determine the meta-parameters for the networks: number of inputs, number of hidden layers, number of nodes in each layers, activation functions for the nodes, etc.

Therefore, much of the work around neural networks is in how to choose or learn an optimal architecture. Genetic algorithms have been used to select the optimal set of inputs [288, 23]. Classic genetic algorithms performed cross-over and mutation on a binary encoding of the parameter space [116]. That binary encoding is rarely optimal for applications in credit risk [43]. A more general evolutionary approach [151] could operate on the full architecture of the neural network in order to share optimal subnets across candidate networks within a population.

Feed-forward networks are the most commonly used, largely because they are the easiest to train and comprehend. However, recurrent neural networks have been used to create memory within the network rather than have the analyst provide lagged inputs of dynamic variables in behavior scoring contexts [142]. When applied to massive amounts of input data, such as transactional information, convolutional neural networks have been applied [167].

Even with an optimal architecture, limiting overfitting [255, 172, 249] is a significant problem. Much work has been done in this area with some surprising findings that the number of parameters in deep learning networks may not be as much of a problem as we think [30]. One explanation may be that the initial random assignment of many small parameters might actually create robustness to input noise rather than the multicolinearity nightmare we would otherwise expect.

Even worse can be transient structures that are actually present in the data, but only for a short period of time. When we know that a certain structure will not persist in the future, such as an old account management policy of an expiring government program, how does one get the neural network to forget? One answer could be the 'given knowledge' approach suggested by Breeden and Leonova (2019) [51] where we could train a subnet on just the transient structure, embed this as a fixed component of a network trained to solve the larger problem on the full data set, and then remove the subnet when creating forecasts out of sample.

Neural networks are data hungry and time intensive to train, but can be successfully used. Many authors have studied these effects, comparing different neural network designs and comparing them to other methods [21, 190, 233, 170, 284, 3, 93]. When the available data is wide in the number of inputs but short in the number of observations, ensembles of small networks can also be effective [280].

4.1.2 Support Vector Machines

Support vector machines (SVMs) excel at creating segmentations of the input vector space for classification. The ability to segment the observation space with arbitrary hyperplanes provides an effective classification technique for an arbitrary number of end states and without assumptions about the distributions of the input factors or target categories. They are less well suited to continuous prediction problems, although techniques mentioned earlier can be applied to product continuous outputs. SVMs have been applied to credit scor-

ing by multiple authors and found to be an effective approach in many cases [264, 285, 21, 237].

One of the biggest advantages in SVMs is the ability to use kernels to create optimally separating hyperplanes (OSHs). The "kernel trick" refers to the chosen maps the data to a higher-dimensional space, which can in some cases dramatically simplify the process of finding OSHs. The placement of the hyperplanes is a nonlinear problem requiring an optimizer.

As with neural networks, the challenge is optimizing the architecture. With SVMs, the input features and the kernel parameters must be optimized. The choice of whether to use a linear, polynomial, radial basis function, or other kernel is a matter of experimentation given a specific data set. No universal best answer exists, but the best advice is to start simple (linear) and move toward complex as required.

These choices across meta-parameters are interdependent. To optimize these meta-parameters, GAs have again been applied [104] and other hybrid approaches [143]. The lesson from studies into neural networks and SVMs is that optimizing the meta-parameters is essential to success.

4.1.3 Decision Trees

Decision trees are a simple concept that can be used to create sophisticated models. The concept is a recursive partitioning of the input space until enough confidence is achieved to make a prediction. They have been used for decades in credit risk [188, 75, 105] where the earliest decision trees were heuristically created. Modern algorithms can use a variety of partitioning criteria: misclassification error, Gini index, information gain, gain ratio, ANOVA, and others. The final forecast can be the state with the greatest representation in the final leaf, a probability based upon representation, or a small model as in regression trees [54, 91]. The meta-parameters are how to optimize the partitioning, the input factors, and when to stop partitioning. As usual, these need to be optimized.

A single tree can have the same overfitting concerns as previous methods, but the explosion in the use of decision trees has come with the introduction of ensembles. Bagged decision trees [293], boosted decision trees [27], random forests [164, 189, 109], rotation forests [204, 193], and stochastic gradient boosted trees [261, 60] are some of the most popular. Most authors agree that this list represents a steady improvement in methodology, currently with stochastic gradient boosted trees as the usual winner. Although ensemble methods are most popular in scoring when applied to decision trees, these methods are found combined with all credit scoring techniques [7].

One advantage of decision trees is the mapping between trees and rules. Trees can be compared to known rules and rules can be learned from trees [64].

In general, trees have an advantage in handling sparse data or data with outliers. Binning is a simple method to limit outlier sensitivity that is lacking in continuous methods like neural networks. In situations where the data is abundant, of good quality, and with clear nonlinearities, neural networks are

often the reported winners.

4.1.4 Nearest-neighbors and Case-based Reasoning

One category of models could be defined as those that learn from past examples. Case-Based Reasoning (CBR) [57] searches through past lending experiences to find a comparable loan. In commercial lending where examples are few and nearly unique, this can be an effective approach. Where most data is available, as with consumer lending, a kNN (k-nearest neighbors) [129, 191] approach is conceptually equivalent.

The challenge with both CBR and kNN lies with identifying comparables. This is not unlike the challenge for home appraisals. If the closest comparable home is at a distance, in a different kind of neighborhood, is it really comparable? This concept applies to both methods here. Any data set will be non-uniformly distributed along the explanatory factors. When optimizing the metric for identifying comparable loans or choosing "k" in kNN, the definition of a near neighbor that works well in one region of the space may be a poor choice in another.

Using geography as an example, finding 20 neighbors in an urban setting might provide a roughly homogenous set, whereas finding the same 20 neighbors in a sparse geography could span counties or even states. Of course, using CBR or kNN geographically could create a redlining risk, but the same concept applies, if more abstractly, to any set of explanatory factors. Therefore, the success of these methods appears to be tied to the uniformity of the distribution of the data set.

Where CBR and kNN may excel are in extremely sparse data situations. When tens of events or less are available, especially when the events are very heterogeneous in their properties, matching to prior experience without attempting to interpolate or extrapolate as in estimation-based approaches may be more effective.

4.1.5 Kernel Methods, Fuzzy Methods, and Rough Sets

Kernel methods, fuzzy methods, and rough sets are best viewed as a method to augment other modeling approaches. Decision trees, support vector machines, or any method that performs classification by drawing hard boundaries among the input factors will inevitably have uncertainty in the location of those boundaries. In general, one would assume that the greatest forecast errors should occur near the boundaries. Incorporating estimation kernels [287, 59] into these methods or treating the boundaries as fuzzy [134] can capture this uncertainty and potentially improve accuracy by reporting appropriate probabilities. Estimation kernels or fuzzy logic have been incorporated into many credit scoring methods [275, 289, 295, 119]. This may be particularly valuable in sparse data settings where the boundaries can only be approximations.

Rough sets have also seen application to credit scoring [198]. With an objective similar to kernel and fuzzy methods, rough sets have been combined with

other base modeling techniques to incorporate the imprecision of the available information. Along these lines, rough sets have been combined with decision trees [296] and with SVMs [62].

4.1.6 Genetic Programming

Genetic programming (GP) employs trees to perform computation. The leaves are input values or numerical constants. The branching nodes contain numerical operators or functions. In this way, nested algebraic operations can be performed to create predictions for credit scoring [212, 144, 4].

The genetic aspect refers to how the tree structure, constants, and input factors are chosen. As with genetic algorithms, concepts of mutation and crossover are employed. Mutation is a simple change in a constant, swapping an input factor, or swapping an operator or function. Crossover is the more interesting process of swapping subtrees between two trees. In genetic algorithms applied to binary representations, crossover rarely produces viable offspring because the fitness landscape lacks useful symmetries. In GP applications to credit scoring, such symmetries exist if subtrees can capture conceptual subsets of the problem, such as swapping the proper transformation of an input factor between candidate trees..

For credit scoring, the fitness function will be one or several measures of forecast accuracy in predicting the target variable. The optimization naturally occurs on an ensemble of candidate trees. The best tree at the end of the optimization process can be used as the model, but following the ensemble concept, one could also apply a voting algorithm across all qualifying trees. However, one challenge with genetically learned ensembles is that they tend to cluster around a single peak in the fitness landscape. A similarity penalty could be added to the fitness function to encourage diversity in the population, both to reduce the risk of being stuck in a local optimum and to increase the usefulness of the ensemble.

GP appears to be useful as a highly nonlinear method. To justify the slow search speed of genetic methods, one needs a problem that is equally complex. Simple credit scoring problems may not qualify, but the use of alternate data sources might make GP more interesting.

4.1.7 Alternate data sources

Some machine learning methods for credit scoring are specifically focused on how to incorporate new data sources into the scores. Cash flow analysis using data scraped from demand deposit accounts is a successful area of business application, particularly during the COVID-19 pandemic during which so much traditional scoring data is in doubt. Although the data source is new to scoring, the methods for analysis are more traditional. One seeks to determine the frequency and reliability of income by different sources. During the COVID-19 pandemic, someone with periodic, steady income could be a good credit risk

regardless of credit score, industry of employment, or many other underwriting criteria.

Mobile phone data is potentially an important data source in emerging markets and specifically for underbanked consumers. Research on credit risk for Chinese consumers using mobile phone calling records and billing information has been found to be effective for credit risk assessment [274]. Research in underdeveloped markets using smartphone metadata such as types of apps installed, text message history, etc. [274]. Both studies used well known credit scoring and machine learning methods, just with emphasis on sourcing and regularizing new data types.

Some novel data sources can require corresponding innovations in analysis. Social network data has proven to be quite interesting [278, 97], but incorporating data from networks into a credit score can be a challenge. Low-dimensional embeddings of network graphs [122] are the standard approach to creating a usable input factor for modeling. However, recent research [242] suggests that low dimensional embeddings lose much of the information in the network. The best approach for incorporating social network data will continue to be a topic of research for some time, as will the ethics and legality of incorporating such data within the underwriting process.

4.2 Corporate Defaults

Discussions of credit scoring usually carry an implication of consumer loans and large volumes of training data. Modeling corporate defaults and bankruptcies is a similar problem, but with fewer events in the training data and less standardized inputs. A panelist at a conference on machine learning in finance explained humbly that they used machine learning just to read the corporate financial statements. The scoring models were trivial. In fact, standardizing diverse and heterogenous inputs may be one of the best uses of machine learning in lending applications.

Even so, some large data sets on corporate defaults do exist, and a variety of papers have been published to apply ML to the problem [262, 247, 13] . Bankruptcy and default are not exactly the same thing, but bankruptcy filings are public so many works have focused there [200, 248, 198, 209, 67, 19, 267]. Across both applications, the methods tested cover the full range of machine learning techniques.

4.3 Other Scoring Applications

Published work often lags what is being done in-house at lenders around the world. For example, the author knows that prepayment and attrition models have been created using machine learning, with the short study in Section 5 as one such previously unpublished example. At this point, one can assume that machine learning is being tested everywhere models can be employed in lending.

4.3.1 Loss Given Default

The natural companion to credit risk forecasting is modeling loss severity, loss given default (LGD). LGD, or conversely, recovery modeling, has always been a challenging problem because of the inherent tri-modal distribution [239, 28]. Some percentage of borrowers will show no net loss in event of default because of the collateral value. Another significant percentage can be expected to have 100% LGD because of failure to recover the collateral (such as a totaled vehicle), and a distribution can exist between the two extremes. LGD has been modeled as a multi-stage problem where the first step is to predict 0, 1, or Intermediate and the second stage attempts to predict the specific value for the intermediates.

In addition to economic sensitivity [33], LGD can also depend on the age of the loan and the time since default. One approach is to use survival or age-period-cohort methods to predict monthly recoveries from the date of default with vintage defined by month of default.

Naturally, given the importance of LGD (the 2009 US mortgage crisis was as much an LGD crisis as a PD crisis because of the collapse in property values), work has also been done to apply machine learning to LGD [252, 246] or recovery rates [32]. Given the complexity of the problem, research into which approach is best for different asset classes could continue for some time.

4.3.2 Automated Valuation Models

The valuation of property in collateralized loans is part of the underwriting process providing a preview of what LGD could be in the event of default, much the way en primeur wine ratings are an early estimate of what the quality of a finished wine will be [8]. Automated valuation models (AVMs) replace the human property appraiser with a data driven model to speed the approval process and lower the origination costs. Machine learning methods are also being applied to AVMs [260, 31] where regression-based approaches have been previously deployed [86, 107].

4.4 Portfolio Forecasting and Stress testing

Time series applications of machine learning provide an interesting contrast to machine learning applications in credit scoring. The challenges and best methods are almost completely opposite. In credit scoring, large data sets are obtained by observing many accounts, transactions, or behaviors over a short period of time. Success comes largely through identifying nonlinearities and interactions. For time series modeling, the available data sets are very short relative to macroeconomic cycles [51] and credit cycles [48]. Some studies have reached questionable conclusions, because what looks like a linear response over a short time period may in fact be a cyclical response to a completely different factor when observed over a longer period.

The longest data sets in lending usually extend only as far back as the mid-1990s. For the US, that translates to only two clear recessions (2001 and 2009),

although some subcycles can also be observed [47]. At the time of this writing, the COVID-19 Global Recession is just beginning. This event may add more clarity about tail risk in our models and the kind of government responses we can expect in the event of extreme events.

When faced with short (in time) and wide (in variables) data sets, different approaches will be preferable to what was seen in credit scoring. In fact, most lenders struggle to obtain data back to 2006, which would qualify as one full economic cycle in the US. Modeling portfolio responses through a single economic cycle is akin to having four data points in a credit score: one good, one bad, one mediocre trending worse, and one mediocre trending better. The usual validation of testing on the last 12 months of data is no more than a continuity test, not a true out of sample test.

Regardless of the technique employed, creating time series forecasts and stress test models is about creating robustness much more than worrying about subtle interaction effects or subtle nonlinearities. As such, an ensemble of small regression models is more likely to succeed in the next recession than a deep learning neural network. Although decision trees are very successful in scoring, any binned method is less suitable to forecasting rates if it truncates the tail of the distribution. Continuous models only extrapolate to the tails of the distribution by making assumptions, but those can be explicitly expressed.

Therefore, forecasting and stress testing are applications where machine learning must be combined with intuition. Business and analyst experience serve as a human-powered smoothing and regularization technique for models that do not truly have enough examples to train upon. Some commentators have suggested that state-level or MSA-level modeling can solve the problem of not having enough economic cycles for training, but US states are highly correlated. Some lead-lag effects are present, but no state missed the 2009 recession. Oil shocks can create recessions in energy states like Alaska, North Dakota, West Virginia, and West Texas, but we are far from having 50 separate macroeconomic responses to model with 50 states.

This discussion should not be taken to imply that nonlinearities are unimportant. In fact, transformations of macroeconomic inputs must be carefully chosen. Percentage change in gross domestic product (GDP) is not a good fit to a linear regression model, because increases and decreases are not symmetric. Taking the logarithm of the ratio of the values would produce a roughly normally distributed distribution that is symmetric in changes, and thus better suited to use in a linear regression model. In short, either variables need to be transformed to scale linearly with the target, or the model needs to be flexible enough to learn the nonlinearities. However, in a limited data environment, there may not be enough information to learn the nonlinearities from the data, so human assistance through choosing the transforms is one path to success.

Using interest rates in loan default models provides an effective example. Over the last decade, analysts have commonly taken the natural log of interest rates or the log of the ratio of interest rates in order to control for the fact that a change from 4% to 3% is much more important than a change from 15% to 14%. Unfortunately, we have entered a realm where interest rates can go

negative and logarithm-based transformations are not suitable. Therefore, we need to find transformations that are more linear through zero but less sensitive for large values. Transformations such as $y = tanh(x)$ and sigmoid functions in general, as well as $y = sign(x)\sqrt{|x|}$ are reasonable candidates.

Conveniently, this observation about suitable transformations fits well with neural networks, leading to the thought that ensembles of short, wide neural networks could be an effective approach for portfolio forecasting and stress testing. Short because sufficient data is only available to support one to a few hidden layers. Wide because many macroeconomic factors can be taken as inputs and the neural network provides a nonlinear equivalent to dimensionality reduction like PCA. Ensemble because many such models trained on randomly selected data subsets when combined provide robustness relative to the limited economic cycles available.

Genetic methods like GP can and probably have been used to swap transformed macroeconomic factors between models, much as has been demonstrated in credit scoring [288]. However, the data sets in time series models are small enough that exhaustive search among input factors is often possible. Forward stepwise regression and backward stepwise regression are also common approaches.

Although the author has observed that ensembles of small time series models can be quite effective, they pose model risk management challenges under current practices, which is discussed in Section 6.

The target variables for time series modeling can be delinquency rates, default rates, charge-off rates, prepayment rates, and recovery rates. All of this can be combined to create time series forecasts of expected losses, payments, and revenue ultimately leading to cash flow modeling needed for estimating yield or loss reserves under CCAR, CECL or IFRS 9. No technical obstacle exists to creating the outputs with machine learning enhancements such as ensembles of nonlinear models, but in the author's experience, auditors are not yet ready to use them to produce numbers in financial statements.

4.4.1 Recurrent Neural Networks

Recurrent neural networks and LSTM were designed specifically to learn lag structures from data in time series problems, so one would expect that they be tested for loan loss stress testing. Although quite successful in speech recognition, challenges exist in application to credit risk time series modeling.

As mentioned in the previous section, the primary issue is with the number of events in the data. In a training data set for speech recognition, every vowel is another cycle in the data, unlike the extreme data sparsity in stress testing. However, they may yet find a niche.

We already know that each recession has unique aspects. Models of loan defaults need to focus on the direct drivers of borrower cash flows. However, when we build models across multiple recessions, in cases where we have data on multiple recessions, the lags and cross-correlations between economic factors change. In 2008, a collapse in house prices preceded a decline in GDP and a

subsequent drop in unemployment. In the 2020 COVID-19 recession, declines in GDP and unemployment rate are the leading effects and house price and commercial real estate declines follow. A simpler model will simply average across these structures, producing an unfocused model structure.

One alternative could be to use some form of regime switching [225] that detects the nature of the recession and switches between models corresponding to different types of crisis [26, 186]. Although plausible, the data is limited. The question not yet answered is whether some form of recurrent neural network could perform the equivalent of regime switching in a smoother, more continuous way and thereby adapt better to differing macroeconomic structures. This would only be conceivable with the longest data sets, possibly between 1995 and 2021, for example, to capture three or four recessions with at least three clearly different types of economic crises. That time is not so far in the future and it will be interesting to see what can be done to improve the state of the art in stress testing.

4.4.2 Survival and Vintage Models

Survival and vintage models occupy a middle ground between scoring methods and top-down time series models. Vintage models, such as Age-Period-Cohort (APC) models, operate simultaneously on multiple time series segmented by vintage (origination date) cohort so that dynamics versus age of the loan, credit risk by vintage, and environmental impacts may be quantified and used in forecasting [113, 286, 106]. Survival models operate on individual account performance data, but with the important addition of when an event occurred, rather than simply if something occurred, as with traditional credit scores [95]. Both methods can produce the periodic forecasts required for forecasting, stress testing, cash flow modeling, and pricing.

Both survival and vintage models include a function of risk by age of the account known as the hazard function or lifecycle function, respectively. The estimation of this function is inherently nonlinear, so the now-standard methods developed decades ago should fairly be considered machine learning methods. Nonparameteric estimation [82, 157], parametric and spline estimation [231] as in APC, and Bayesian methods [238] are all standard approaches. Neural networks or even decisions trees will probably be tested for estimating hazard functions, although the necessity is not clear given available nonparametric methods.

Account-level Cox proportional hazards models [256, 250, 80] and other survival scoring techniques [50] importantly include a scoring aspect that can be performed with a version of regression or more generally with machine learning techniques as well [87, 37]. Wang, Li, and Reddy (2019) [272] provide a thorough survey of machine learning survival methods to date.

The environmental or econometric modeling aspect of survival and vintage models can be addressed via the time series methods discussed in the previous section. Therefore, although survival and vintage models were machine learning methods from the start, they are being aggressively hybridized with the latest

techniques [51]. One of the great advantages of these methods is the proven separability of nonlinear effects in age of the account, vintage, and environment by calendar date [135]. That separability creates a semi-structured approach where each of those pieces can be estimated by the methods and data set most suitable to the problem while the underlying mathematical structure guarantees a consistent framework for combining the pieces.

Naturally, ensemble methods for survival and vintage models have also appeared. Random survival forecasts and other ensembles [147, 139, 138] have been reported to be quite successful for credit risk modeling.

4.5 Portfolio Optimization

Modern portfolio theory [192, 84] is based upon stable linear expected returns and covariances for a set of possible instruments. Experience has shown that expected returns and covariances are rarely stable or linear, so this area is also being explored for enhancement with machine learning. In a lending context, optimization is constrained to the limits of how much certain asset types can be grown and whether or how much holdings can be reduced.

Optimization under modern portfolio theory can be viewed as optimizing the Sharpe ratio [245], defined as the ratio of expected return to expected volatility. Even without machine learning, many enhancements have been offered to this view, such as the Sortino ratio [88] which looks only at the volatility arising from negative returns.

In the context of optimizing a lending portfolio or any investment portfolio that includes loans, one needs to consider unique aspects of loan losses. As seen more dramatically for retail portfolio, increases in losses can occur because of lifecycle (loss timing) effects or intended changes in credit quality. Similar to the way the Sortino ratio computes volatlity without penalizing for positive increases, loan loss volatility and correlations should not include structure that is a feature of the product or intended management. Simulation-based methods exist for recreating historic loss time series to remove such expected variances [44].

Once we understand the true covariance structure with loan products, portfolio optimization methods based upon machine learning should apply equally well here as with general investment portfolios where they were originally developed. Early work focused on capturing nonlinearities in the covariance structure, tail risk, and boundaries. Copulas have seen significant application in this area [42, 154], but neural networks [58], genetic methods [243, 169], fuzzy optimization [184] and others have also been used. Ban (2018) [25] provides a review of available methods.

5 What Makes ML Work

In a 2018, Casey Foltz at Oregon Community Credit Union (OCCU) used 23 different machine learning algorithms readily available in R to compare checking

account attrition models. More than just a comparison of AUC values, the project's goal was also to understand the reasons for the winners and losers. Experiments were conducted on how to modify the inputs and model meta-parameters in order to explore what made one method work better than another and how to improve the weaker performers.

The explanatory factors included a number of measures of fees incurred, transaction errors by the financial institution, complaints and denied credit applications. As would become important later, most of these variables were not normally or even lognormally distributed. The outcome variable was binary, attrite or not attrite during the two year observation period.

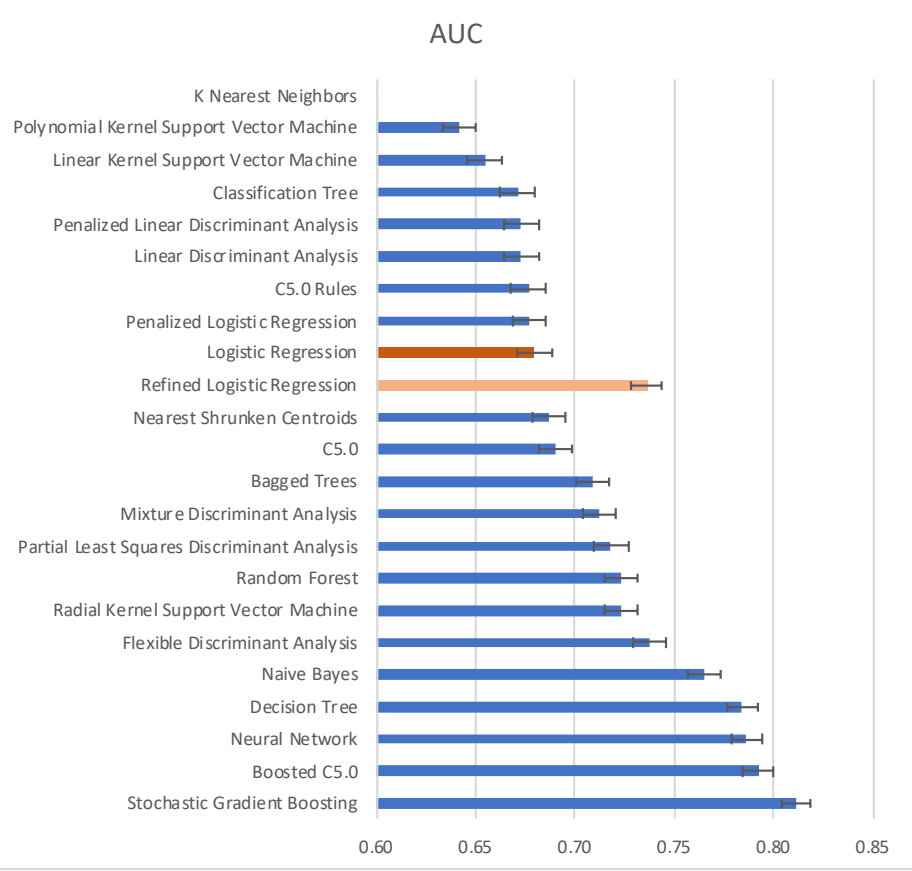

Figure 1: A comparison of AUC values for models of checking account attrition probability.

The explanatory factors and target variable were fed unmodified into each of the available algorithms with default parameters. Figure 1 shows the initial comparison between the methods. In reviewing these results, the first step was

to review the meta-parameters of each method. For example, the neural network was implemented with `nnet` in R, which only allows for a single hidden layer. Packages like `tensorflow` allow for significantly more flexibility, but that would involve quite a bit more exploration. However, even unoptimized, the neural net was the third best approach.

Other methods could also benefit from optimization. For example, the linear discriminant analysis performs best with normally distributed variables. Applying logarithmic transformations to the variables with roughly lognormal distributions added 5% to the AUC value.

Stochastic gradient boosted trees was the winning method in this study of checking account attrition. However, the deeper question was why. How far could we push the logistic regression model toward the stochastic gradient boosted tree's performance?

To investigate this question, three simple things were done. Lognormally distributed variables were transformed with a logarithmic function. All other variables were binned so that graphs of test factor versus probability of attrition were created. For a large number of variables, those graphs showed the data to exhibit two regimes with linear relationships to attrition on either side of a break point. Therefore, those variables were split with an interaction term allowing for two different linear responses. Finally, the odd variables were just manually binned. With the couple dozen available input variables, this exercise took about an hour of manual work. The result is also shown in Figure 1. Confidence intervals for the AUC values were computed according to DeLong, et. al. [76].

The logistic regression model moved from the bottom third of the methods to the upper third. Decision trees and neural networks still performed better than the refined logistic regression, implying that more could be done to linearize the input factors and identify needed interaction terms. Capturing nonlinearities has previously been shown to be important for credit risk modeling [185], so this result is not surprising.

The improvements come from the boosted methods, employing multiple models rather than a single model. Figure 2 shows the ROC curves for the original logistic regression model, the refined logistic regression model, and the stochastic gradient boosted trees.

This is just one of numerous examples in the literature, but it illustrates the progression of predictability gained through adapting to nonlinear responses, interaction terms, and ensemble models.

6 Challenges of Employing Machine Learning

For all its promise, machine learning presents some unique challenges to application in credit risk. Unlike applications in speech recognition or image processing, accuracy alone is not sufficient in lending. FCRA guidelines require that lenders not discriminate against protected classes and that consumers are offered explanations for denial of credit. Such concerns have dramatically slowed the

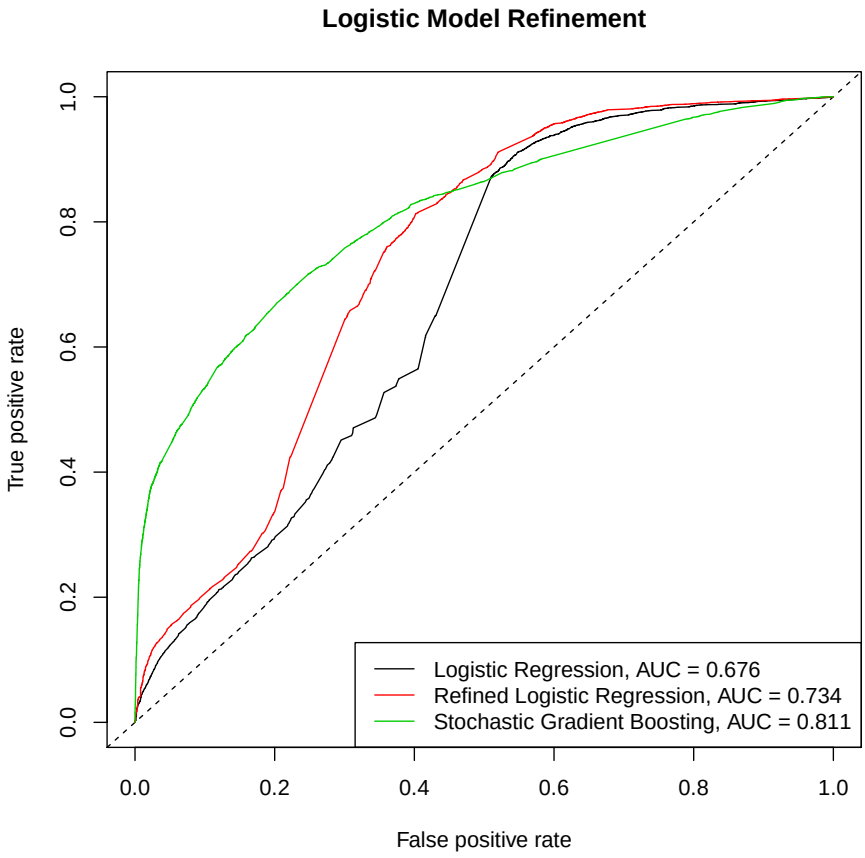

Figure 2: ROC curve comparison for the original logistic regression model, a refined logistic regression, and stochastic gradient boosting.

adoption of machine learning, and with good reason. These and other valid concerns in model risk management must be addressed before the models can be widely adopted.

Note that Sections 6.4 Unintended Bias, 6.5 Adverse Action Notices, 6.6 Predicting without Understanding, and 6.7 Adapting to Sudden Behavioral Shifts are all facets of explainability. Collectively, model explainability is the most critical challenge to widespread adoption of machine learning in credit risk. Financial institutions and regulatory bodies [39] cannot rely on models that they do not understand, for all of the reasons listed in these sections.

6.1 Large data needs

The promise of machine learning comes largely from the ability to incorporate nonlinearities in the input variables and interdependence between variables. That promise is fulfilled only in the presence of very large data sets both for identifying the structure and for testing to make sure the structure is not spurious. While those data sets are appearing in some contexts, some machine learning models are being built without the requisite data.

Conversely, the many studies that compare machine learning methods in hopes of identifying which is best often fail to note that the answer is strongly tied to how much data is available. In sparse data environments, k-nearest neighbor models may beat neural networks. With large, complex data sets, deep learning neural networks are likely to win. With intermediate data sets that produce many spurious to transient correlations, boosted trees might come out ahead. The simple answer is that we are unlikely ever to crown a single winning method, because the data sets and output requirements vary so widely, even in specific contexts like credit risk modeling.

Interestingly, ensembles of models can be used to identify where more data is needed [163]. This has been raised as an aid in the reject inference problem. Testing the model in regions where unlabeled data (rejection applications) predominates can highlight where the model is most in need of additional data.

6.2 Imbalanced data sets

Another problem that is more prevalent in credit risk than generic machine learning applications is the extreme imbalance between outcomes [148, 29]. For example, in a commercial loan portfolio, defaults might occur for only 0.1% of accounts. This imbalance means that many machine learning algorithms will be happy to classify the non-defaults while largely ignoring the defaults, resulting in ever poorer performance where it is needed most [270, 158].

Two main approaches have been explored to address the data imbalance problem. Brown and Mues (2012) [55] tested a range of machine learning methods across data sets with varying levels of imbalance in defaults to identify those methods best suited to modeling data sets with different default rates. One notable result was that traditional methods like logistic regression and linear discriminant analysis are robust to the degree of imbalance in the data, so

this is largely a machine learning question.

Others have pursued various strategies of modifying the training data to create more balance [145, 194]: over-sample the smaller class, under-sample the larger class, apply weights to the training data, or generate synthetic data to augment the lesser class, as with SMOTE [61].

Overall, the results appear to show that adding to the under-represented class is most effective, with SMOTE being a commonly used approach. SMOTE is basically a random sampling along hyperplanes connecting pairs of points in the smaller class, a linear interpolation. Since this is using a simple model to generate data to feed into a more sophisticated model, it is no surprise that other methods have been proposed.

With any data manipulation approach, the analyst must remember that the underlying probabilities are being modified. The resulting model may be used for scoring, but will require work in order to reintroduce predictions of probabilities. The simplistic approach of introducing a scalar to adjust for the over-sampling is risky, as the sampling will not be perfectly uniform across the feature space, so the probabilities likely will not be accurately recreated locally.

6.3 Overfitting the overfitting tests

Many machine learning methods use performance on an out-of-sample data set to determine when to stop training the model. This approach to preventing overfitting is generally effective, but it carries a caveat. As a rule, the more often a data point is tested the less it can be considered out-of-sample. This was recognized several decades ago. In the case of hypothesis testing, the significance of the result should be adjusted based upon the number of tests conducted, as in the HolmBonferroni method [137].

When repeatedly testing scoring metrics or goodness-of-fit measures on an out-of-sample data set rather than hypothesis testing as above, the author is not aware of an equivalent adjustment, but the same principles apply. Simply stated, a good result with fewer out-of-sample tests is better than a slightly better result after many more tests. This needs to be considered when creating machine learning models and when reviewing work done.

These principles apply to both scoring models tested across hold out samples and time series models tested on an out-of-time sample. Rerunning an out-of-time test repeatedly can result in "look-ahead bias" where the meta-parameter decisions are based upon the analyst's judgement of accuracy on data that was supposed to be out-of-sample. This problem is particularly acute when modeling a short time series relative to the cycle being studied.

6.4 Unintended bias

Machine learning has been in production for fraud detection longer than any other application in lending. Conversations with those involved at the beginning suggest that the earliest efforts did not have zip code as an input, but were

essentially zip code detection tools. Using or inferring zip codes in loan underwriting or pricing is called redlining and is prohibited [1]. In fraud detection, no such prohibition exists, and one wonders why they didn't just give it zip code to start with.

This story is useful only in the notion that given many other inputs, a sophisticated machine learning algorithm recreated the data that it needed most. That is the greatest danger for using machine learning in credit risk. With linear methods, we generally feel safe in saying that no information on protected class status was given to the model, so the results are unbiased. The same cannot be said of machine learning [90, 223], especially when given alternate inputs. Big Data and sophisticated modeling approaches create significant unobserved risks of inequality and unfair treatment [211].

Consider the case of Amazon's AI-based attempt to find the best job applicants [74]. It was apparently shut down because it was identifying female applicants based upon association with women's groups, and Amazon didn't hire many female engineers, so following the pattern meant that women were rejected. That tale could easily be replayed in credit risk, where a machine learning algorithm infers protected class status using social media data, credit card transactions, branch transactions, etc. One such example showed that the digital footprint of an online borrower was as predictive as FICO score, yet all of those digital footprint data elements probably correlate to protected class status [34]. Excluding protected data is insufficient to assert that the final model's forecasts do not correlate to protected status. Simple linear correlation is the standard for discrimination.

A significant amount of research is being conducted on how to identify and mitigate disparate impacts from machine learning. Current methods can largely be grouped into two approaches. One group is modifying the input data to prevent models from finding biases [155, 298, 292, 90, 124]. The second group modifies the learning algorithm to add constraints that would enforce fairness conditions. [229, 94, 114, 156, 290].

The challenge with both approaches is the need to tag the data with information about protected class status. If we knew the demographic data for each account in the training data, one could trivially run correlations to prove that no bias exists after applying one of the above methods or others. Unfortunately, a linear mindset underlies the regulations. US lenders are not allowed to save data about race, gender, and such for anything except mortgages, so they lack the data necessary to prove that the models are performing fairly. Something will need to change here.

The risk of unintended bias is one of the greatest obstacles to widespread adoption of machine learning models. The solutions will be legal as much as statistical [174].

6.5 Adverse Action Notices

The Equal Credit Opportunity Act (ECOA) [12], as implemented by Regulation B, and the Fair Credit Reporting Act (FCRA), require lenders to provide

Adverse Action Notices when a consumer is denied credit. These notices are specifically intended to be both understandable by the consumer and actionable in the sense that the consumer can make improvements in their financial position in order to qualify in the future. Machine learning has many applications in credit risk, but when it is the primary underwriting tool, it must have good answers for consumers.

Unlike the previous discussion about global interpretability, providing reasons for specific decisions is an inherently local problem. Several methods exist for this, but it remains an important area of research, referred to as the quest for explainable AI (XAI) [81, 203, 181, 112].

The first widely adopted method was Local Interpretable Model-agnostic Explanations (LIME) [227]. LIME samples the space around the decision point to generate a small data set. These points are weighted by distance from the original point, and a small local linear model is built. In fact, the original idea was that any model could work, but the standard implementation is linear. So, it's making a local linear model of a potentially highly nonlinear model overall and using the smaller model to explain the decision just as one would with a linear model.

Shapley values [244] use game theory to allocate significance across input factors. The focus here is local rate of change of the forecast relative to a specific input The approach leverages the original model rather than a locally created simplified model as in LIME. This concept has been enhanced for application to XAI by several authors [251, 187], including integrating elements of LIME [18].

Unfortunately, LIME can be unstable, and both Shapley values and LIME can suffer when a forecast point is at an inflection point in the input variables. In such cases, important dependencies will be missed. Significant research into XAI is currently happening in image processing. Recent work there has developed an approach of explaining an answer relative to a reference image [81]. Work in credit risk has shown that the same reference approach can be effective. Moreover, using a distribution of reference points can provide both explainability and robustness [125].

Certainly more methods will follow. For linear methods, explainability is inherent. Hopefully in the near future, XAI will be an integral part of all machine learning methods.

6.6 Predicting without Understanding

Henley and Hand's work is often cited [129] showing that even small gain in a credit score adds business value. This is taken as proof that prediction is important above all, presumably including explanation. However, those in business know that understanding gained from the modeling process can be used in intangible ways during the underwriting process to add value. One of the greatest risks with machine learning is that analysts can create effective models without learning about the problem they are modeling. For both the analyst and the business, learning matters.

One of our deepest insights from the checking attrition project in Section 5 was the realization that readily available machine learning packages allow analysts to create highly predictive models without understanding what is driving those models. Even when used with default settings, many of these algorithms performed quite well, but is it a good thing to be able to create such models without seizing the opportunity to learn more about the business? In our attempt to understand the relative performance, we actually did learn more about the underlying dynamic between customer and lender, but this was not necessary for the model's success. The importance of explanatory methods for machine learning is not just about educating customers and regulators, but also so that analysts learn about the business.

Some of the understanding gained from a detailed explanation of the model can be more about the data itself. Several machine learning methods are robust to outliers, but if those outliers are data errors, this robustness can lead to a false extrapolation. Robust machine learning models put a greater burden on the analyst to validate the data to assure the model does not just learn an entry error.

Part of the solution also comes back to the disparate impact analysis. We need to recognize that explainable AI is valuable and necessary not just for consumers but also for analysts. Model risk managers need to start asking for a deeper inspection of what makes a machine learning model work, what are the key structures being leveraged, and what can we do with this knowledge to improve the input data and model development process.

Some have gone so far as to say that a bad model that can be understood is better than a good model that cannot be. Let's be clear. That is also a bad answer. The correct answer is to work harder to explain the good models.

6.7 Adapting to Sudden Behavioral Shifts

This article is being written during the depths of the COVID-19 recession. As soon as shelter-in-place orders were issued in the US, we knew that the models would have a problem. All of the algorithms discussed here are data-driven pattern recognition engines. When past patterns are not predictive of future behavior, the models will fail.

Asking for forbearance on a mortgage was no longer a risk indicator, just sensible cash flow management. Job loss and filing for unemployment might be a joint strategy of employer and employee to maximize government benefits until the business reopens. "Strategic delinquency" will probably appear in the research literature in a year or two, exploring the behavioral dynamics leading consumers to go delinquent even when they have money just in order to hoard cash. Sudden increases in deposits, drops in spending, and increases in forbearance reinforce this perspective. Early monitoring of machine learning models in the crisis suggest that exactly these kinds of failures are occurring [127].

In a model driven world, we cannot just wait months or years for new data to arrive to allow us to retrain the models. Model triage becomes an immediate top priority. Human judgment is required to create intuitive models of how

behavior is shifting and what adjustments or overlays should be deployed to compensate. In such crises, linear methods or models with separable pieces have the advantage, because then their human masters can understand more easily where model weaknesses might lie, the presumed sensitivities that are no longer true, and what adjustments might compensate for the new situation.

A complex machine learning model could essentially be picking up on the same structures as a linear model, yet lack of interpretability will be a major obstacle to use. The best way to make the machine learning methods robust through such behavior shifts is to make them explainable globally [203] so that the managers can understand enough to compensate.

That is precisely the objective with global interpretability methods in machine learning. Permutation tests [210, 96] randomize either the outcome labels or the input values to measure the significance of the model and specific inputs. Partial dependence plots [100] create a graph of the average forecast versus values for a given input , where the test value is substituted into each input data element. This idea spawned several others including accumulated local effects (ALE) [16], where the change in the model forecast for small changes about a test value is averaged across all corresponding values of the other inputs. Individual Conditional Expectation is essentially a disaggregation of the partial dependence plots, showing the forecasts for each input at the test values rather than simply aggregating to an average value. That disaggregated visualization can provide additional insights into what drives the model [117]. These are a few of the methods available for visualizing the dynamics of machine learning models.

This section could have been called "Run global interpretability tests". However, in the rush to finish a model and start the next task, analysts usually leave them for "later". In a crisis, the tools that get used are the ones that are already in place and are understandable. Therefore, measures to provide insight into a model must be part of model development and validation. Such insights are obvious with linear models. Machine learning must adopt such measures as standard practice before models are deployed.

6.8 p-Value arbitrage

In comparing machine learning with traditional methods, the worst reason to choose a winner would be if they were being judged by different standards. For the most part, machine learning models are considered acceptable if they test well out-of-sample, provide a reasonable disparate impact analysis, and do not appear to be biased. For logistic regression, the list is a bit longer.

The most notable difference is the use of p-values to screen for insignificant factors in logistic regression models. Standard practice among model validators and auditors is to make sure that all coefficients in the model are statistically significant according to the p-value, given a reasonably chosen threshold. The p-value is essentially measuring the distance from zero considering the estimation uncertainty. For binned variables where each bin has a corresponding coefficient, the appropriate interpretation is that "some" of the bins should have statistically

significant coefficients. For example, if month-of-year were an input with one coefficient for each month, you would not delete June from the model if its coefficient were zero so long as other months were significantly non-zero. Given that, let's focus on coefficients for continuous variables.

Consider Figure 3. The figure compares coefficients estimated for three different input variables. The first coefficient fails the p-value test, because with it's 95% confidence interval touching zero, it would not meet the 5% p-value threshold, i.e. its coefficient is not provably non-zero. The second coefficient also fails the p-value test for the same reason. However, assuming the inputs are standardized, the first variable is potentially much more important than the second, just equally uncertain. The third coefficient passes and would be allowed into the model, even though it is only weakly useful.

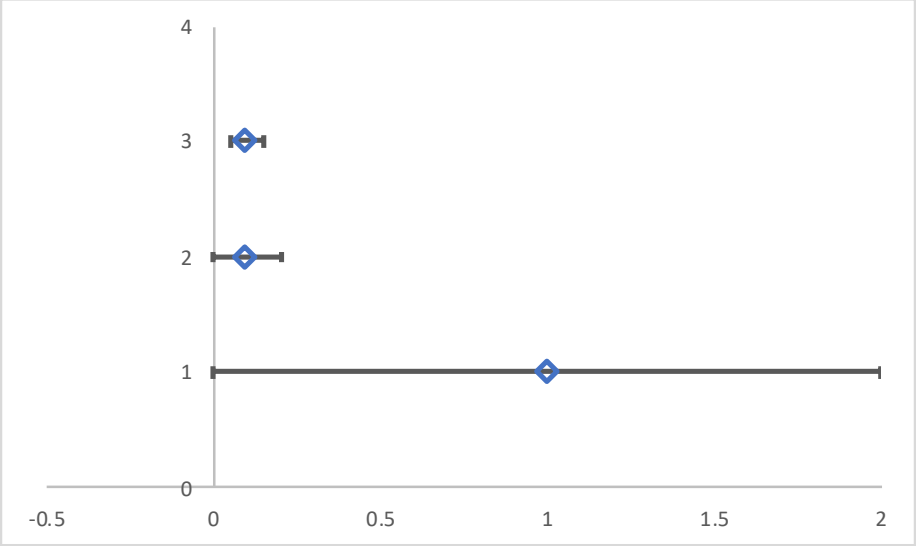

Figure 3: Coefficients with confidence intervals are shown for three hypothetical input variables. The x-axis shows the estimated value and confidence interval. The y-axis just lists three different events.

The American Statistical Society says this is not a correct use or interpretation of p-values [276, 208], and yet it is standard practice in credit risk modeling. By using a p-value criterion for screening variables in regression models but not in machine learning models, we are creating a p-value arbitrage situation. In one of the model comparison studies, we should test the significance of the input factors to see if machine learning models are including factors deleted from the regression models.

It is important that we avoid creating a situation where analysts inadvertently choose machine learning methods over regression methods just because of inconsistent evaluation standards by those in model risk management.

7 Conclusions

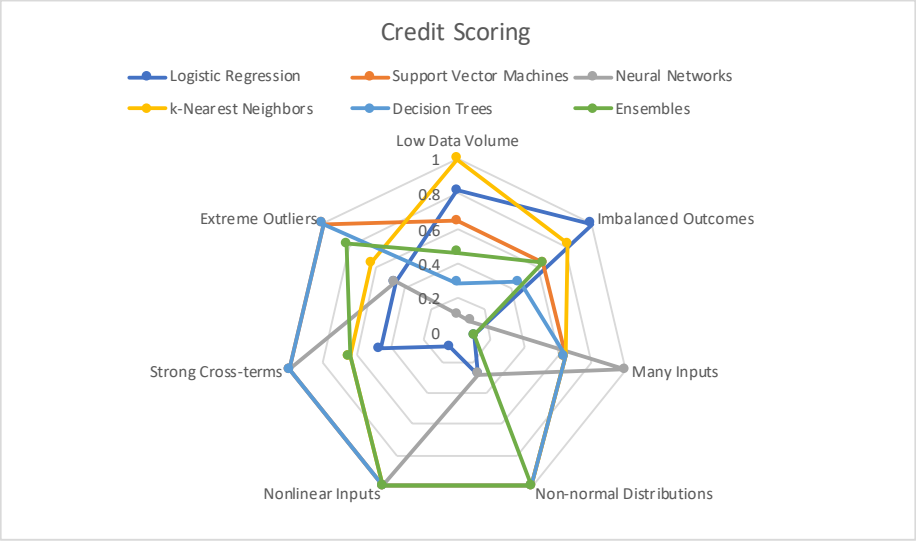

Figure 4: An intuitive comparison of potential strengths and weaknesses of various models for credit scoring. 0 is the weakest and 1 is the strongest under a given challenge.

In reviewing the many machine learning methods available and the equally numerous applications, it becomes clear that declaring a single best method is impossible. Methods have specific strengths and weaknesses that align to different applications. In a specific application, the best method often involves the combination of elements of several methods, both statistical and machine learning.

For academics and researchers, the goal should be to develop a problem space map showing the optimal domains for the methods. Figures 4 and 5 give the author's rough intuitive assessment relative to common modeling challenges for credit scoring and credit risk time series modeling. Each vertex gives a modeling challenge and each modeling technique is rated from 0 (worst) to 1 (best) at addressing that challenge. In the course of creating this survey, the author did not find any method that would be best against all challenges.

If these model rankings could truly be quantified, we could create a recommendation engine that would assess a modeling task and recommend a subset of methods that are likely candidates. Of course, these maps are only guesses, do not include all variants of all methods, and do not consider all modeling challenges. Adding those details would be worthy additions to the literature.

As far as where the field goes from here, several trends are apparent. Research continues into how best to model image-like data sets. We noted that some researchers used image processing techniques to analyze credit card trans-

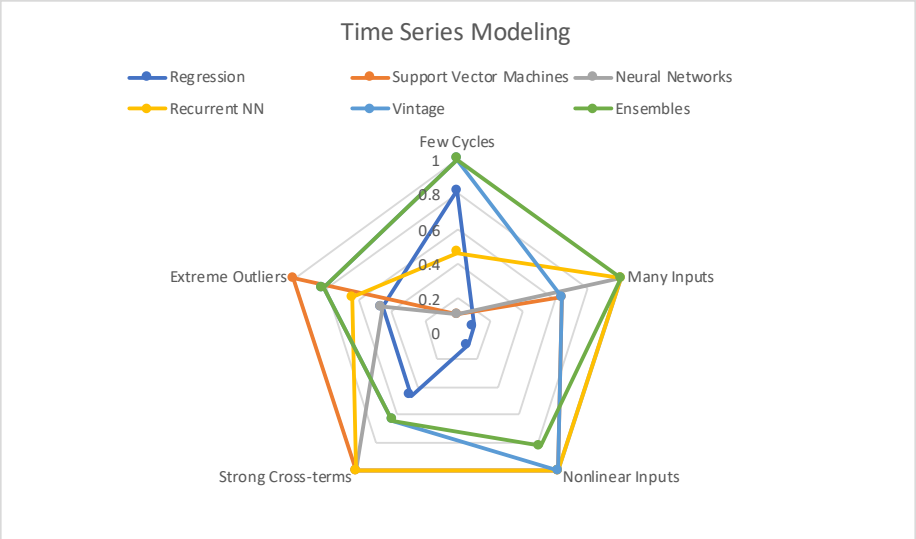

Figure 5: An intuitive comparison of potential strengths and weaknesses of various models for time series modeling in credit risk. 0 is the weakest and 1 is the strongest under a given challenge.

action data, so those could find application in credit risk. Memory-dependent methods such as time series modeling is still seeing rapid development. For all methods reviewed, methods for selecting meta-parameters could be dramatically improved.

Overall, however, we must note that good machine learning methods exist across a range of scoring and time series modeling applications. The greatest advances from here are likely to be more in addressing the challenges of Section 6. All of those challenges involve looking past model accuracy to issues of how to make the models function productively in the real world.

One thing missing from all of these methods is that they produce expectation values. Distributions of possible outcomes are obtained only by running multiple input scenarios, as in stress testing, or looking across many models, as with ensemble distributions. Could we move beyond these forecasts of expectation values to performing calculations upon entire distributions so that the final output of any model is immediately a distribution?

Perhaps, this is where quantum computing [215, 83] could revolutionize credit risk modeling (and many other industries as well). With quantum calculations could we incorporate the full uncertainty of the non-normal distributions of our problems through each step to the final answer? Clearly the greatest failing in using credit risk models is the infrequent generation of confidence intervals and the even rarer use of those in decision making. If all forecasts had accurate measures of uncertainty attached expressing their full non-normal distributions, we would find a great deal of false precision being employed.

Machine learning in some form is clearly the future, but that does not mean it is the present. The challenges listed are not insignificant. The institutional knowledge required for the proper development, validation, monitoring, and overriding of machine learning models currently exists only in pockets. Recognizing those challenges is the best way to speed wider adoption with the fewest possible number of newsworthy blow-ups.

References

[1] Federal Trade Commision, September 2012. 15 U.S.C. S 1681.

[2] New credit score unveiled drawing on bank account data. ABA Banking Journal, October 2018. Newsbytes, Retail and Marketing, Technology.

[3] Hussein Abdou, John Pointon, and Ahmed El-Masry. Neural nets versus conventional techniques in credit scoring in egyptian banking. *Expert Systems with Applications*, 35(3):1275–1292, 2008.

[4] Hussein A Abdou. Genetic programming for credit scoring: The case of egyptian public sector banks. *Expert systems with applications*, 36(9):11402–11417, 2009.

[5] Umar Farouk Ibn Abdulrahman, Joseph Kobina Panford, and James Ben Hayfron-acquah. Fuzzy logic approach to credit scoring for micro finance in Ghana: a case study of kwiqplus money lending. *International Journal of Computer Applications*, 94(8), 2014.

[6] Mohammad Zoynul Abedin, Guotai Chi, Sisira Colombage, and Fahmida-E Moula. Credit default prediction using a support vector machine and a probabilistic neural network. *Journal of Credit Risk*, 14(2).

[7] Joaquín Abellán and Carlos J Mantas. Improving experimental studies about ensembles of classifiers for bankruptcy prediction and credit scoring. *Expert Systems with Applications*, 41(8):3825–3830, 2014.

[8] Héla Hadj Ali, Sébastien Lecocq, and Michael Visser. The impact of gurus: Parker grades and en primeur wine prices. *The Economic Journal*, 118(529):F158–F173, 2008.

[9] K. Ali and M. Pazzani. Error reduction through learning multiple descriptions. *Machine Learning*, 24:172–202, 1996.

[10] Linda Allen, Lin Peng, and Yu Shan. Social networks and credit allocation on fintech lending platforms. Technical report.

[11] Shirley Almon. The distributed lag between capital appropriations and expenditures. *Econometrica: Journal of the Econometric Society*, pages 178–196, 1965.

[12] Sarah Ammermann. Adverse action notice requirements under the ECOA and the FCRA. Consumer Compliance Outlook, 2013.

[13] Ioannis Anagnostou, Javier Sánchez Rivero, Sumit Sourabh, and Drona Kandhai. Contagious defaults in a credit portfolio: A bayesian network approach. *Journal of Credit Risk*, 16(1):1–26, 2019.

[14] Raymond Anderson. *Credit Intelligence & Modelling: Many Paths through the Forest*. Independently Published, 2019.

[15] Eliana Angelini, Giacomo di Tollo, and Andrea Roli. A neural network approach for credit risk evaluation. *The quarterly review of economics and finance*, 48(4):733–755, 2008.

[16] Daniel W Apley and Jingyu Zhu. Visualizing the effects of predictor variables in black box supervised learning models. *arXiv preprint arXiv:1612.08468*, 2016.

[17] Sina Ardabili, Amir Mosavi, and Annamária R Várkonyi-Kóczy. Advances in machine learning modeling reviewing hybrid and ensemble methods. In *International Conference on Global Research and Education*, pages 215–227. Springer, 2019.

[18] Miller Janny Ariza-Garzón, Javier Arroyo, Antonio Caparrini, and Maria-Jesus Segovia-Vargas. Explainability of a machine learning granting scoring model in peer-to-peer lending. *IEEE Access*, 8:64873–64890, 2020.

[19] Amir F Atiya. Bankruptcy prediction for credit risk using neural networks: A survey and new results. *IEEE Transactions on neural networks*, 12(4):929–935, 2001.

[20] Abhishek Awasthi. Clustering algorithms for anti-money laundering using graph theory and social network analysis. 2012.

[21] Bart Baesens, Tony Van Gestel, Stijn Viaene, Maria Stepanova, Johan Suykens, and Jan Vanthienen. Benchmarking state-of-the-art classification algorithms for credit scoring. *Journal of the operational research society*, 54(6):627–635, 2003.

[22] Bart Baesens, Stijn Viaene, Dirk Van den Poel, Jan Vanthienen, and Guido Dedene. Bayesian neural network learning for repeat purchase modelling in direct marketing. *European Journal of Operational Research*, 138(1):191–211, 2002.

[23] A. C. Bahnsen and A. M. Gonzalez. Evolutionary algorithms for selecting the architecture of a MLP neural network: A credit scoring case. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 725–732, 2011.

[24] Michael Bailey, Rachel Cao, Theresa Kuchler, Johannes Stroebel, and Arlene Wong. Social connectedness: Measurement, determinants, and effects. *Journal of Economic Perspectives*, 32(3):259–80, 2018.

[25] Gah-Yi Ban, Noureddine El Karoui, and Andrew EB Lim. Machine learning and portfolio optimization. *Management Science*, 64(3):1136–1154, 2018.

[26] Anil Bangia, Francis X Diebold, André Kronimus, Christian Schagen, and Til Schuermann. Ratings migration and the business cycle, with application to credit portfolio stress testing. *Journal of Banking & Finance*, 26(2-3):445–474, 2002.

[27] Joao Bastos. Credit scoring with boosted decision trees. Technical Report MPRA Paper No. 8034, CEMAPRE, School of Economics and Management (ISEG), Technical University of Lisbon, April 2007.

[28] João A Bastos. Forecasting bank loans loss-given-default. *Journal of Banking & Finance*, 34(10):2510–2517, 2010.

[29] Gustavo EAPA Batista, Ronaldo C Prati, and Maria Carolina Monard. A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD explorations newsletter*, 6(1):20–29, 2004.

[30] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine learning practice and the bias-variance trade-off. *arXiv preprint arXiv:1812.11118*, 2018.

[31] Anthony Bellotti. Reliable region predictions for automated valuation models. *Annals of Mathematics and Artificial Intelligence*, 81(1-2):71–84, 2017.

[32] Anthony Bellotti, Damiano Brigo, Paolo Gambetti, and Frédéric D Vrins. Forecasting recovery rates on non-performing loans with machine learning. In *Credit Scoring and Credit Control XVI Conference*, Edinburgh, August 2019.

[33] Tony Bellotti and Jonathan Crook. Loss given default models incorporating macroeconomic variables for credit cards. *International Journal of Forecasting*, 28(1):171–182, 2012.

[34] Tobias Berg, Valentin Burg, Ana Gombović, and Manju Puri. On the rise of fintechs–credit scoring using digital footprints. Technical report, National Bureau of Economic Research, 2018.

[35] James O Berger. *Statistical decision theory and Bayesian analysis*. Springer Science & Business Media, 2013.

[36] Michael J Best. *Quadratic programming with computer programs*. Chapman and Hall/CRC, 2017.

[37] Elia Biganzoli, Patrizia Boracchi, Luigi Mariani, and Ettore Marubini. Feed forward neural networks for the analysis of censored survival data: a partial logistic regression approach. *Statistics in medicine*, 17(10):1169–1186, 1998.

[38] Daniel Björkegren and Darrell Grissen. Behavior revealed in mobile phone usage predicts loan repayment. *arXiv preprint arXiv:1712.05840*, 2017.

[39] Financial Stability Board. Artificial intelligence and machine learning in financial services. *Market developments and financial stability implications*, 1, 2017.

[40] Jean-Charles de Borda. Mémoire sur les élections au scrutin: Histoire de lacadémie royale des sciences. *Paris, France*, 12, 1781.

[41] Léon Bottou. Stochastic gradient descent tricks. In *Neural networks: Tricks of the trade*, pages 421–436. Springer, 2012.

[42] Heni Boubaker and Nadia Sghaier. Portfolio optimization in the presence of dependent financial returns with long memory: A copula based approach. *Journal of Banking & Finance*, 37(2):361–377, 2013.

[43] Joseph L Breeden. GA-optimal fitness functions. In *International Conference on Evolutionary Programming*, pages 95–102. Springer, 1998.

[44] Joseph L. Breeden. Portfolio optimisation. In *Reinventing Retail Lending Analytics: Forecasting, Stress Testing, Capital and Scoring for a World of Crises*, pages 299–321. Risk Books, London, 2010.

[45] Joseph L. Breeden. *Reinventing Retail Lending Analytics: Forecasting, Stress Testing, Capital and Scoring for a World of Crises, 2nd Impression.* Risk Books, London, 2014.

[46] Joseph L. Breeden. Incorporating lifecycle and environment in loan-level forecasts and stress tests. *European Journal of Operational Research*, 255(2):649 – 658, 2016.

[47] Joseph L. Breeden. Measuring economic cycles in data. *Journal of Risk Model Validation*, 14(1):1–17, March 2020.

[48] Joseph L. Breeden and Jose J. Canals-Cerdá. Consumer risk appetite, the credit cycle, and the housing bubble. *Journal of Credit Risk*, 14(2):1–30, 2018.

[49] Joseph L Breeden, Anthony Bellotti, E Leonova, and A Yablonski. Instabilities using cox ph for forecasting or stress testing loan portfolios. In *Credit Scoring and Credit Control Conference XIV*, 2015.

[50] Joseph L. Breeden and Jonathan Crook. Multihorizon survival models. In *Credit Scoring and Credit Control XVI Conference*, Edinburgh, August 2019.

[51] Joseph L Breeden and Eugenia Leonova. When big data isnt enough: Solving the long-range forecasting problem in supervised learning. In *2019 International Conference on Modeling, Simulation, Optimization and Numerical Techniques (SMONT 2019)*. Atlantis Press, 2019.

[52] Joseph L. Breeden and Norman Packard. A learning algorithm for optimal representation of experimental data. *International Journal of Bifurcation and Chaos*, 4(2):311–326, April 1994.

[53] Leo Breiman. Bagging predictors. *Machine Learning*, pages 123–140, 1996.

[54] Leo Breiman, Jerome Friedman, Charles J Stone, and Richard A Olshen. *Classification and regression trees*. CRC press, 1984.

[55] Iain Brown and Christophe Mues. An experimental comparison of classification algorithms for imbalanced credit scoring data sets. *Expert Systems with Applications*, 39(3):3446–3453, 2012.

[56] Wray L Buntine and Andreas S Weigend. Bayesian back-propagation. *Complex systems*, 5(6):603–643, 1991.

[57] Paul Buta. Mining for financial knowledge with cbr. *Ai Expert*, 9(2):34–41, 1994.

[58] C Casas. Reducing portfolio volatility with artificial neural networks. In *Artificial Intelligence and Applications: IASTED International Conference Proceedings*, 2005.

[59] Stephan K Chalup and Andreas Mitschele. Kernel methods in finance. In *Handbook on information technology in finance*, pages 655–687. Springer, 2008.

[60] Yung-Chia Chang, Kuei-Hu Chang, and Guan-Jhih Wu. Application of extreme gradient boosting trees in the construction of credit risk assessment models for financial institutions. *Applied Soft Computing*, 73:914–920, 2018.

[61] Nitesh V Chawla, Nathalie Japkowicz, and Aleksander Kotcz. Special issue on learning from imbalanced data sets. *ACM SIGKDD explorations newsletter*, 6(1):1–6, 2004.

[62] Fei-Long Chen and Feng-Chia Li. Combination of feature selection approaches with svm in credit scoring. *Expert systems with applications*, 37(7):4902–4909, 2010.

[63] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[64] Yen-Liang Chen and Lucas Tzu-Hsuan Hung. Using decision trees to summarize associative classification rules. *Expert Systems with Applications*, 36(2):2338–2351, 2009.

[65] Gong Cheng, Peicheng Zhou, and Junwei Han. Learning rotation-invariant convolutional neural networks for object detection in vhr optical remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, 54(12):7405–7415, 2016.

[66] Robert T. Clemen. Combining forecasts: A review and annotated bibliography. *International Journal of Forecasting*, 5:559–583, 1989.

[67] Pamela K Coats and L Franklin Fant. Recognizing financial distress patterns using a neural network tool. *Financial management*, pages 142–155, 1993.

[68] John Y. Coffman and Gary G. Chandler. Applications of performance scoring to accounts receivable management in consumer credit. Technical report, Credit Research center, Krannert Graduate School of Management, Purdue University, 1983.

[69] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th International Conference on Machine Learning*, ICML 08, page 160167, New York, NY, USA, 2008. Association for Computing Machinery.

[70] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[71] Thomas Cover and Peter Hart. Nearest neighbor pattern classification. *IEEE transactions on information theory*, 13(1):21–27, 1967.

[72] D. R. Cox. Regression models and life-tables. *Journal of the Royal Statistical Society. Series B (Methodological)*, 34(2):187–220, 1972.

[73] Belur V Dasarathy and Belur V Sheela. A composite classifier system design: Concepts and methodology. *Proceedings of the IEEE*, 67(5):708–713, 1979.

[74] Jeffrey Dastin. Amazon scraps secret AI recruiting tool that showed bias against women. *Technology News*, 2018.

[75] Rober Hunter DAVIS, DB Edelman, and AJ Gammerman. Machine-learning algorithms for credit-card applications. *IMA Journal of Management Mathematics*, 4(1):43–51, 1992.

[76] Elizabeth R DeLong, David M DeLong, and Daniel L Clarke-Pearson. Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. *Biometrics*, pages 837–845, 1988.

[77] Vijay S Desai, Jonathan N Crook, and George A Overstreet Jr. A comparison of neural networks and linear scoring models in the credit union environment. *European Journal of Operational Research*, 95(1):24–37, 1996.

[78] Sander Dieleman, Kyle W Willett, and Joni Dambre. Rotation-invariant convolutional neural networks for galaxy morphology prediction. *Monthly notices of the royal astronomical society*, 450(2):1441–1459, 2015.

[79] Thomas G Dietterich. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer, 2000.

[80] Viani Biatat Djeundje and Jonathan Crook. Dynamic survival models with varying coefficients for credit risks. *European Journal of Operational Research*, 275(1):319–333, 2019.

[81] Ann-Kathrin Dombrowski, Maximillian Alber, Christopher Anders, Marcel Ackermann, Klaus-Robert Müller, and Pan Kessel. Explanations can be manipulated and geometry is to blame. In *Advances in Neural Information Processing Systems*, pages 13567–13578, 2019.

[82] Bradley Efron. The two-way proportional hazards model. *Journal of the Royal Statistical Society B*, 64:899 – 909, 2002.

[83] Daniel J Egger, Ricardo Gacía Gutiérrez, Jordi Cahué Mestre, and Stefan Woerner. Credit risk analysis using quantum computers. *arXiv preprint arXiv:1907.03044*, 2019.

[84] Edwin J Elton, Martin J Gruber, Stephen J Brown, and William N Goetzmann. *Modern portfolio theory and investment analysis*. John Wiley & Sons, 2009.

[85] Walter Enders. *Applied Econometric Time Series, Fourth Edition*. John Wiley & Sons, 2014.

[86] Muhammad Faishal Ibrahim, Fook Jam Cheng, and Kheng How Eng. Automated valuation model: an application to the public housing resale market in singapore. *Property Management*, 23(5):357–373, 2005.

[87] David Faraggi and Richard Simon. A neural network model for survival data. *Statistics in medicine*, 14(1):73–82, 1995.

[88] Simone Farinelli, Manuel Ferreira, Damiano Rossello, Markus Thoeny, and Luisa Tibiletti. Beyond sharpe ratio: Optimal asset allocation using different performance ratios. *Journal of Banking & Finance*, 32(10):2057–2063, 2008.

[89] FDIC. Credit card activities manual. https://www.fdic.gov/regulations/examinations/credit_card/ch12.html, 2007.

[90] Michael Feldman, Sorelle A Friedler, John Moeller, Carlos Scheidegger, and Suresh Venkatasubramanian. Certifying and removing disparate impact. In *proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 259–268, 2015.

[91] Holmes Finch and Mercedes K Schneider. Classification accuracy of neural networks vs. discriminant analysis, logistic regression, and classification and regression trees. *Methodology*, 3(2):47–57, 2007.

[92] Jason P. Fine and Robert J. Gray. A proportional hazards model for the subdistribution of a competing risk. *Journal of the American Statistical Association*, 94(446):496–509, 1999.

[93] Steven Finlay. Multiple classifier architectures and their application to credit risk assessment. *European Journal of Operational Research*, 210(2):368–378, 2011.

[94] Benjamin Fish, Jeremy Kun, and Adám D Lelkes. Fair boosting: a case study. In *Workshop on Fairness, Accountability, and Transparency in Machine Learning*. Citeseer, 2015.

[95] T.R. Fleming and D.P. Harrington. *Counting Processes and Survival Analysis*. Wiley, New York, 1991.

[96] Eibe Frank and Ian H Witten. Using a permutation test for attribute selection in decision trees. 1998.

[97] Seth Freedman and Ginger Zhe Jin. The information value of online social networks: lessons from peer-to-peer lending. *International Journal of Industrial Organization*, 51:185–222, 2017.

[98] Christoph Frei and Marcus Wunsch. Moment estimators for autocorrelated time series and their application to default correlations. *Journal of Credit Risk*, 14(1).

[99] Yoav Freund and Robert E Schapire. Experiments with a new boosting algorithm. In *icml*, volume 96, pages 148–156. Citeseer, 1996.

[100] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[101] Jerome H Friedman. Stochastic gradient boosting. *Computational statistics & data analysis*, 38(4):367–378, 2002.

[102] Jerome H Friedman and Werner Stuetzle. Projection pursuit regression. *Journal of the American statistical Association*, 76(376):817–823, 1981.

[103] Jerome H Friedman and John W Tukey. A projection pursuit algorithm for exploratory data analysis. *IEEE Transactions on computers*, 100(9):881–890, 1974.

[104] Holger Frohlich, Olivier Chapelle, and Bernhard Scholkopf. Feature selection for support vector machines by means of genetic algorithm. In *Proceedings. 15th IEEE International Conference on Tools with Artificial Intelligence*, pages 142–148. IEEE, 2003.

[105] Halina Frydman, Edward I Altman, and Duen-Li Kao. Introducing recursive partitioning for financial classification: the case of financial distress. *The Journal of Finance*, 40(1):269–291, 1985.

[106] Wenjiang Fu. *A Practical Guide to Age-Period-Cohort Analysis: The Identification Problem and Beyond*. Chapman and Hall/CRC, 2018.

[107] Joshua Gallin, Raven Molloy, Eric Reed Nielsen, Paul A Smith, and Kamila Sommer. Measuring aggregate housing wealth: New insights from an automated valuation model. 2018.

[108] Dani Gamerman and Hedibert F Lopes. *Markov chain Monte Carlo: stochastic simulation for Bayesian inference*. Chapman and Hall/CRC, 2006.

[109] Nazeeh Ghatasheh. Business analytics using random forest trees for credit risk prediction: A comparison study. *International Journal of Advanced Science and Technology*, 72(2014):19–30, 2014.

[110] Sushmito Ghosh and Douglas L Reilly. Credit card fraud detection with a neural-network. In *System Sciences, 1994. Proceedings of the Twenty-Seventh Hawaii International Conference on*, volume 3, pages 621–630. IEEE, 1994.

[111] Bruce P Gibbs. *Advanced Kalman filtering, least-squares and modeling: a practical handbook*. John Wiley & Sons, 2011.

[112] Leilani H Gilpin, David Bau, Ben Z Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, pages 80–89. IEEE, 2018.

[113] Norval D. Glenn. *Cohort Analysis, 2nd Edition*. Sage, London, 2005.

[114] Gabriel Goh, Andrew Cotter, Maya Gupta, and Michael P Friedlander. Satisfying real-world goals with dataset constraints. In *Advances in Neural Information Processing Systems*, pages 2415–2423, 2016.

[115] David E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional, 1989.

[116] DE Goldberg. *Genetic algorithms in search, optimization, and machine learning*. Addison-Wesley, Reading, MA, 1989.

[117] Alex Goldstein, Adam Kapelner, Justin Bleich, and Emil Pitkin. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, 24(1):44–65, 2015.

[118] Dale L Goodhue, William Lewis, and Ronald L Thompson. A dangerous blind spot in is research: False positives due to multicollinearity combined with measurement error. In *AMCIS*, 2011.

[119] Asogbon Mojisola Grace and Samuel Oluwarotimi Williams. Comparative analysis of neural network and fuzzy logic techniques in credit risk evaluation. *International Journal of Intelligent Information Technologies (IJIIT)*, 12(1):47–62, 2016.

[120] Alex Graves, Santiago Fernndez, Faustino Gomez, and Jrgen Schmidhuber. Connectionist temporal classification: Labelling unsegmented sequence data with recurrent neural 'networks. volume 2006, pages 369–376, 01 2006.

[121] Alastair R Hall. *Generalized method of moments*. Oxford university press, 2005.

[122] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in neural information processing systems*, pages 1024–1034, 2017.

[123] L. Hansen and P. Salamon. Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:993–1001, 1990.

[124] Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323, 2016.

[125] Ian Hardy. Robust explainability in AI models, 2020.

[126] Frank Harrell. Road map for choosing between statistical modeling and machine learning. https://www.fharrell.com/post/stat-ml, 2019.

[127] Will Douglas Heaven. Our weird behavior during the pandemic is messing with AI models. MIT Technology Review, May 2020.

[128] Robert Hecht-Nielsen. Theory of the backpropagation neural network. In *Neural networks for perception*, pages 65–93. Elsevier, 1992.

[129] WE Henley and D Hand. Construction of a k-nearest-neighbour credit-scoring system. *IMA Journal of Management Mathematics*, 8(4):305–321, 1997.

[130] WE Henley and David J Hand. Ak-nearest-neighbour classifier for assessing consumer credit risk. *Journal of the Royal Statistical Society: Series D (The Statistician)*, 45(1):77–95, 1996.

[131] Joseph M. Hilbe. *Logistic Regression Models*. Taylor & Francis, 5 2009.

[132] T. K. Ho. The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832–844, 1998.

[133] Arthur E Hoerl and Robert W Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[134] F Hoffmann, Bart Baesens, Christophe Mues, Tony Van Gestel, and Jan Vanthienen. Inferring descriptive and approximate fuzzy rules for credit scoring using evolutionary algorithms. *European journal of operational research*, 177(1):540–555, 2007.

[135] T R Holford. The estimation of age, period and cohort effects for vital rates. *Biometrics*, 39(2):311–324, 1983.

[136] Theodore Holford. *Encyclopedia of Statistics in Behavioral Science*, chapter Age-Period-Cohort Analysis. Wiley, 2005.

[137] Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979.

[138] Torsten Hothorn, Peter Bühlmann, Sandrine Dudoit, Annette Molinaro, and Mark J Van Der Laan. Survival ensembles. *Biostatistics*, 7(3):355–373, 2006.

[139] Torsten Hothorn, Berthold Lausen, Axel Benner, and Martin Radespiel-Tröger. Bagging survival trees. *Statistics in medicine*, 23(1):77–91, 2004.

[140] Cheng Hsiao. *Analysis of Panel Data*. Cambridge University Press, 2014.

[141] Kuo-Wei Hsu. A theoretical analysis of why hybrid ensembles work. *Computational intelligence and neuroscience*, 2017, 2017.

[142] T. Hsu, S. Liou, Y. Wang, Y. Huang, and Che-Lin. Enhanced recurrent neural network for combining static and dynamic features for credit card default prediction. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1572–1576, 2019.

[143] Cheng-Lung Huang, Mu-Chen Chen, and Chieh-Jen Wang. Credit scoring with a data mining approach based on support vector machines. *Expert systems with applications*, 33(4):847–856, 2007.

[144] Jih-Jeng Huang, Gwo-Hshiung Tzeng, and Chorng-Shyong Ong. Two-stage genetic programming (2sgp) for the credit scoring model. *Applied Mathematics and Computation*, 174(2):1039–1053, 2006.

[145] Yueh-Min Huang, Chun-Min Hung, and Hewijin Christine Jiau. Evaluation of neural networks and data mining methods on a credit assessment task for class imbalance problem. *Nonlinear Analysis: Real World Applications*, 7(4):720–747, 2006.

[146] Peter J Huber. Projection pursuit. *The annals of Statistics*, pages 435–475, 1985.

[147] Hemant Ishwaran, Udaya B Kogalur, Eugene H Blackstone, and Michael S Lauer. Random survival forests. *The annals of applied statistics*, 2(3):841–860, 2008.

[148] Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study. *Intelligent data analysis*, 6(5):429–449, 2002.

[149] Herbert L Jensen. Using neural networks for credit scoring. *Managerial finance*, 1992.

[150] I.T. Jolliffe. *Principal Component Analysis, second edition.* Springer, 2002.

[151] Kenneth A. De Jong. *Evolutionary Computation: A Unified Approach.* The MIT Press, 2016.

[152] Jr., David W. Hosmer, Stanley Lemeshow, and Susanne May. *Applied Survival Analysis: Regression Modeling of Time to Event Data, Second Edition.* Wiley Series in Probability and Statistics, New York, 2008.

[153] Frank E. Harrell Jr. *Regression Modeling Strategies: With Applications to Linear Models, Logistic and Ordinal Regression, and Survival Analysis.* Springer, 2015.

[154] Iakovos Kakouris and Berç Rustem. Robust portfolio optimization with copulas. *European Journal of Operational Research*, 235(1):28–37, 2014.

[155] Faisal Kamiran and Toon Calders. Classification with no discrimination by preferential sampling. In *Proc. 19th Machine Learning Conf. Belgium and The Netherlands*, pages 1–6, 2010.

[156] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 35–50. Springer, 2012.

[157] E. L. Kaplan and P. Meier. Nonparametric estimation from incomplete observations. *Journal of the American Statistical Association*, 53(282):457–481, 1958.

[158] Kenneth Kennedy, Brian Mac Namee, and Sarah Jane Delany. Learning without default: A study of one-class classification and the low-default portfolio problem. In *Irish Conference on Artificial Intelligence and Cognitive Science*, pages 174–187. Springer, 2009.

[159] Scott Kirkpatrick, C Daniel Gelatt, and Mario P Vecchi. Optimization by simulated annealing. *science*, 220(4598):671–680, 1983.

[160] J. Kittler, M. Hatef, R. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, 1998.

[161] John R Koza et al. *Genetic programming*. MIT press Cambridge, 1994.

[162] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS12, page 10971105, Red Hook, NY, USA, 2012. Curran Associates Inc.

[163] A. Krogh and J. Vedelsby. *Neural network ensembles, cross validation, and active learning*, pages 231–238. MIT Press, Cambridge, 1995.

[164] Jochen Kruppa, Alexandra Schwarz, Gerhard Arminger, and Andreas Ziegler. Consumer credit risk: Individual probability estimates using machine learning. *Expert Systems with Applications*, 40(13):5125–5131, 2013.

[165] Dimitris Kugiumtzis. State space reconstruction parameters in the analysis of chaotic time seriesthe role of the time window length. *Physica D: Nonlinear Phenomena*, 95(1):13–28, 1996.

[166] Ludmila I Kuncheva. A theoretical study on six classifier fusion strategies. *IEEE Transactions on pattern analysis and machine intelligence*, 24(2):281–286, 2002.

[167] Hvard Kvamme, Nikolai Sellereite, Kjersti Aas, and Steffen Sjursen. Predicting mortgage default using convolutional neural networks. *Expert Systems with Applications*, 102:207 – 217, 2018.

[168] Prentice R. L. the analysis of failure times in the presence of competing risks. *Biometrics*, 34:541, 1978.

[169] Kin Keung Lai, Lean Yu, Shouyang Wang, and Chengxiong Zhou. A double-stage genetic optimization algorithm for portfolio selection. In *International Conference on Neural Information Processing*, pages 928–937. Springer, 2006.

[170] Kin Keung Lai, Lean Yu, Shouyang Wang, and Ligang Zhou. Neural network metalearning for credit scoring. In *International Conference on Intelligent Computing*, pages 403–408. Springer, 2006.

[171] William B Langdon, SJ Barrett, and Bernard F Buxton. Combining decision trees and neural networks for drug discovery. In *European Conference on Genetic Programming*, pages 60–70. Springer, 2002.

[172] Steve Lawrence, C Lee Giles, and Ah Chung Tsoi. Lessons in neural network training: Overfitting may be harder than expected. In *AAAI/IAAI*, pages 540–545. Citeseer, 1997.

[173] Tae-Hwy Lee and Yang Yang. Bagging binary and quantile predictors for time series. *Journal of econometrics*, 135(1-2):465–497, 2006.

[174] David Lehr and Paul Ohm. Playing with the data: what legal scholars should learn about machine learning. *UCDL Rev.*, 51:653, 2017.

[175] Stefan Lessmann, Bart Baesens, Hsin-Vonn Seow, and Lyn C. Thomas. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research*, 247(1):124 – 136, 2015.

[176] Edward M. Lewis. *An Introduction to Credit Scoring*. The Athena Press, San Rafael, California, 1994.

[177] WK Li and Al McLeod. Distribution of the residual autocorrelations in multivariate arma time series models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 43(2):231–239, 1981.

[178] Xiangfeng Li, Shenghua Liu, Zifeng Li, Xiaotian Han, Chuan Shi, Bryan Hooi, He Huang, and Xueqi Cheng. Flowscope: Spotting money laundering based on graphs.

[179] Guohua Liang, Xingquan Zhu, and Chengqi Zhang. An empirical study of bagging predictors for different learning algorithms. In *Twenty-Fifth AAAI Conference on Artificial Intelligence*, 2011.

[180] Charles X Ling and Chenghui Li. Data mining for direct marketing: Problems and solutions. In *Kdd*, volume 98, pages 73–79, 1998.

[181] Zachary C Lipton. The mythos of model interpretability. *Queue*, 16(3):31–57, 2018.

[182] Zachary C Lipton, John Berkowitz, and Charles Elkan. A critical review of recurrent neural networks for sequence learning. *arXiv preprint arXiv:1506.00019*, 2015.

[183] Laura Liu, Hyungsik Roger Moon, and Frank Schorfheide. Forecasting with dynamic panel data models. Working Paper 25102, National Bureau of Economic Research, September 2018.

[184] Yan-Chun Liu, Tie Wang, Li-Qun Gao, Ping Ren, and Bao-Zheng Liu. Fuzzy portfolio optimization model based on worst-case var. In *2005 International Conference on Machine Learning and Cybernetics*, volume 6, pages 3512–3516. IEEE, 2005.

[185] Christian Lohmann and Thorsten Ohliger. Nonlinear relationships in a logistic model of default for a high-default installment portfolio. *Journal of Credit Risk*, 14(1).

[186] André Lucas and Pieter Klaassen. Discrete versus continuous state switching models for portfolio credit risk. *Journal of Banking & Finance*, 30(1):23–35, 2006.

[187] Scott Lundberg and Su-In Lee. An unexpected unity among methods for interpreting model predictions. *arXiv preprint arXiv:1611.07478*, 2016.

[188] Paul Makowski. Credit scoring branches out. *Credit World*, 75(1):30–37, 1985.

[189] Milad Malekipirbazari and Vural Aksakalli. Risk assessment in social lending via random forests. *Expert Systems with Applications*, 42(10):4621–4631, 2015.

[190] Rashmi Malhotra and Davinder K Malhotra. Evaluating consumer loans using neural networks. *Omega*, 31(2):83–96, 2003.

[191] Yannis Marinakis, Magdalene Marinaki, Michael Doumpos, Nikolaos Matsatsinis, and Constantin Zopounidis. Optimization of nearest neighbor classifiers via metaheuristic algorithms for credit risk assessment. *Journal of Global Optimization*, 42(2):279–293, 2008.

[192] Harry M Markowitz and G Peter Todd. *Mean-variance analysis in portfolio choice and capital markets*, volume 66. John Wiley & Sons, 2000.

[193] Ana I Marqués, Vicente García, and José Salvador Sánchez. Exploring the behaviour of base classifiers in credit scoring ensembles. *Expert Systems with Applications*, 39(11):10244–10250, 2012.

[194] Ana Isabel Marqués, Vicente García, and José Salvador Sánchez. On the suitability of resampling techniques for the class imbalance problem in credit scoring. *Journal of the Operational Research Society*, 64(7):1060–1070, 2013.

[195] W.M. Mason and S. Fienberg. *Cohort Analysis in Social Research: Beyond the Identification Problem*. Springer, 1985.

[196] László Mátyás, Christian Gourieroux, Peter CB Phillips, et al. *Generalized method of moments estimation*, volume 5. Cambridge University Press, 1999.

[197] Jose Alberto Mauricio. Exact maximum likelihood estimation of stationary vector arma models. *Journal of the American Statistical Association*, 90(429):282–291, 1995.

[198] Thomas E Mckee. Developing a bankruptcy prediction model via rough sets theory. *Intelligent Systems in Accounting, Finance & Management*, 9(3):159–173, 2000.

[199] Aaron Mengelkamp, Sebastian Hobert, and Matthias Schumann. Corporate credit risk analysis utilizing textual user generated content-a twitter based feasibility study. In *PACIS*, page 236, 2015.

[200] Jae H Min and Young-Chan Lee. Bankruptcy prediction using support vector machine with optimal choice of kernel function parameters. *Expert systems with applications*, 28(4):603–614, 2005.

[201] Jun-Ki Min and Sung-Bae Cho. Activity recognition based on wearable sensors using selection/fusion hybrid ensemble. In *2011 IEEE International Conference on Systems, Man, and Cybernetics*, pages 1319–1324. IEEE, 2011.

[202] Asunción Mochón, David Quintana, Yago Sáez, and Pedro Isasi. Soft computing techniques applied to finance. *Applied Intelligence*, 29(2):111–115, 2008.

[203] Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2019.

[204] Loris Nanni and Alessandra Lumini. An experimental comparison of ensemble of classifiers for bankruptcy prediction and credit scoring. *Expert systems with applications*, 36(2):3028–3033, 2009.

[205] J Neeter, W Wasserman, and MH Kutner. Applied linear statistics. *Irwin, Homeland, IL*, 1985.

[206] Pamela Nickell, William Perraudin, and Simone Varotto. Stability of rating transitions. *Journal of Banking & Finance*, 24(1-2):203–227, 2000.

[207] J. R. Norris. *Markov Chains*. Cambridge University Press, 1998.

[208] Regina Nuzzo. Scientific method: statistical errors. *Nature News*, 506(7487):150, 2014.

[209] Marcus D Odom and Ramesh Sharda. A neural network model for bankruptcy prediction. In *1990 IJCNN International Joint Conference on neural networks*, pages 163–168. IEEE, 1990.

[210] Markus Ojala and Gemma C Garriga. Permutation tests for studying classifier performance. *Journal of Machine Learning Research*, 11(Jun):1833–1863, 2010.

[211] Cathy O'neil. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Broadway Books, 2016.

[212] Chorng-Shyong Ong, Jih-Jeng Huang, and Gwo-Hshiung Tzeng. Building credit scoring models using genetic programming. *Expert Systems with Applications*, 29(1):41–47, 2005.

[213] David Opitz and Richard Maclin. Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169–198, 1999.

[214] Stjepan Oreski, Dijana Oreski, and Goran Oreski. Hybrid system with genetic algorithm and artificial neural networks and its application to retail credit risk assessment. *Expert systems with applications*, 39(16):12605–12617, 2012.

[215] Roman Orus, Samuel Mugel, and Enrique Lizaso. Quantum computing for finance: overview and prospects. *Reviews in Physics*, page 100028, 2019.

[216] Norman H Packard, James P Crutchfield, J Doyne Farmer, and Robert S Shaw. Geometry from a time series. *Physical review letters*, 45(9):712, 1980.

[217] Ebberth L Paula, Marcelo Ladeira, Rommel N Carvalho, and Thiago Marzagão. Deep learning anomaly detection as support fraud investigation in brazilian exports and anti-money laundering. In *2016 15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 954–960. IEEE, 2016.

[218] Yudi Pawitan. *In all likelihood: statistical modelling and inference using likelihood*. Oxford University Press, 2001.

[219] Zdzisław Pawlak. Rough sets. *International journal of computer & information sciences*, 11(5):341–356, 1982.

[220] Selwyn Piramuthu. Financial credit-risk evaluation with neural and neurofuzzy systems. *European Journal of Operational Research*, 112(2):310–321, 1999.

[221] John Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

[222] R. Polikar. Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine*, 6(3):21–45, 2006.

[223] Anya ER Prince and Daniel Schwarcz. Proxy discrimination in the age of artificial intelligence and big data. *Iowa L. Rev.*, 105:1257, 2019.

[224] Foster Provost and Pedro Domingos. Tree induction for probability-based ranking. *Machine learning*, 52(3):199–215, 2003.

[225] Richard E Quandt. A new approach to estimating switching regressions. *Journal of the American statistical association*, 67(338):306–310, 1972.

[226] J. Ross Quinlan. Induction of decision trees. *Machine learning*, 1(1):81–106, 1986.

[227] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[228] Leslie Richeson, Raymond A Zimmermann, and Kevin Gregory Barnett. Predicting consumer credit performance: Can neural networks outperform traditional statistical methods? *International Journal of Applied Expert Systems*, 2(2):116–130, 1994.

[229] Goce Ristanoski, Wei Liu, and James Bailey. Discrimination aware classification for imbalanced datasets. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 1529–1532, 2013.

[230] J. J. Rodriguez, L. I. Kuncheva, and C. J. Alonso. Rotation forest: A new classifier ensemble method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1619–1630, 2006.

[231] Philip S Rosenberg. Hazard function estimation using b-splines. *Biometrics*, pages 874–887, 1995.

[232] Jose San Pedro, Davide Proserpio, and Nuria Oliver. Mobiscore: towards universal credit scoring from mobile phone data. In *international conference on user modeling, adaptation, and personalization*, pages 195–207. Springer, 2015.

[233] Natasa Sarlija, Mirta Bensic, and Marijana Zekic-Susac. A neural network classification of credit applicants in consumer credit scoring. In *Artificial Intelligence and Applications*, pages 205–210, 2006.

[234] Tim Sauer, James A Yorke, and Martin Casdagli. Embedology. *Journal of statistical Physics*, 65(3-4):579–616, 1991.

[235] Robert E Schapire. The boosting approach to machine learning: An overview. In *Nonlinear estimation and classification*, pages 149–171. Springer, 2003.

[236] Robert E Schapire and Yoav Freund. Boosting: Foundations and algorithms. *Kybernetes*, 2013.

[237] Klaus B Schebesch and Ralf Stecking. Support vector machines for classifying and describing credit applicants: detecting typical and critical regions. *Journal of the Operational Research Society*, 56(9):1082–1088, 2005.

[238] Volker Schmid and Leonhard Held. Bayesian age-period-cohort modeling and prediction - bamp. *Journal of Statistical Software, Articles*, 21(8):1–15, 2007.

[239] Til Schuermann. What do we know about loss given default? 2004.

[240] Eduardo S. Schwartz and Walter N. Torous. Prepayment and the valuation of mortgage-backed securities. *The Journal of Finance*, 44(2):375–392, 1989.

[241] OG Selfridge. Pandemonium: a paradigm for learning, mechanism of thought processes: Proceedings of a symposium held at the national physical laboratory, 1958.

[242] C. Seshadhri, Aneesh Sharma, Andrew Stolman, and Ashish Goel. The impossibility of low-rank representations for triangle-rich complex networks. *Proceedings of the National Academy of Sciences*, 117(11):5631–5637, 2020.

[243] John Shapcott. Index tracking: genetic algorithms for investment portfolio selection. *Edinburgh Parallel Computing Centre*, 1992.

[244] Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games*, 2(28):307–317, 1953.

[245] William F Sharpe. The sharpe ratio. *Journal of portfolio management*, 21(1):49–58, 1994.

[246] Han Sheng Sun and Zi Jin. Estimating credit risk parameters using ensemble learning methods: An empirical study on loss given default. *Journal of Credit Risk, Forthcoming*, 2016.

[247] Kyung-shik Shin and Ingoo Han. A case-based approach using inductive indexing for corporate bond rating. *Decision Support Systems*, 32(1):41–52, 2001.

[248] Kyung-Shik Shin and Yong-Joo Lee. A genetic algorithm application in bankruptcy prediction modeling. *Expert systems with applications*, 23(3):321–328, 2002.

[249] Nitish Srivastava. Improving neural networks with dropout. *University of Toronto*, 182(566):7, 2013.

[250] Maria Stepanova and Lyn Thomas. Survival analysis methods for personal loan data. *Operations Research*, 50(2):277–289, 2002.

[251] Erik Štrumbelj and Igor Kononenko. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41(3):647–665, 2014.

[252] Han Sheng Sun and Zi Jin. Estimating credit risk parameters using ensemble learning methods: an empirical study on loss given default. *Journal of Credit Risk*, 12(3):43–69, 2016.

[253] Kar Yan Tam and Melody Y Kiang. Managerial applications of neural networks: the case of bank failure predictions. *Management science*, 38(7):926–947, 1992.

[254] Jun Tang and Jian Yin. Developing an intelligent data discriminating system of anti-money laundering based on svm. In *2005 International conference on machine learning and cybernetics*, volume 6, pages 3453–3457. IEEE, 2005.

[255] Igor V Tetko, David J Livingstone, and Alexander I Luik. Neural network studies. 1. comparison of overfitting and overtraining. *Journal of chemical information and computer sciences*, 35(5):826–833, 1995.

[256] Terry M. Therneau and Patricia M. Grambsch. *Modeling Survival Data: Extending the Cox Model*. Springer-Verlag, New York, 2000.

[257] Lyn C. Thomas, Jonathan N. Crook, and David B. Edelman. *Credit Scoring and Its Applications, Second Edition*. Society for Industrial and Applied Mathematics, Philadelphia, 2017.

[258] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288, 1996.

[259] Ljupčo Todorovski and Sašo Džeroski. Combining classifiers with meta decision trees. *Machine learning*, 50(3):223–249, 2003.

[260] Bogdan Trawiński, Zbigniew Telec, Jacek Krasnoborski, Mateusz Piwowarczyk, Michał Talaga, Tedeusz Lasota, and Edward Sawiłow. Comparison of expert algorithms with machine learning models for real estate appraisal. In *2017 IEEE International Conference on INnovations in Intelligent SysTems and Applications (INISTA)*, pages 51–54. IEEE, 2017.

[261] Bhekisipho Twala. Multiple classifier application to credit risk assessment. *Expert Systems with Applications*, 37(4):3326–3336, 2010.

[262] Parastoo Rafiee Vahid and Abbas Ahmadi. Modeling corporate customers credit risk considering the ensemble approaches in multiclass classification: evidence from iranian corporate credits. *Journal of Credit Risk*, 12(3):71–95, 2016.

[263] Merijn Van Erp, Louis Vuurpijl, and Lambert Schomaker. An overview and comparison of voting methods for pattern recognition. In *Proceedings Eighth International Workshop on Frontiers in Handwriting Recognition*, pages 195–200. IEEE, 2002.

[264] Tony Van Gestel, Johan AK Suykens, Bart Baesens, Stijn Viaene, Jan Vanthienen, Guido Dedene, Bart De Moor, and Joos Vandewalle. Benchmarking least squares support vector machine classifiers. *Machine learning*, 54(1):5–32, 2004.

[265] Robert J Vanderbei. *Linear Programming: Foundations and Extensions*. International Series in Operations Research & Management Science, 2013.

[266] Valdimir N Vapnik. T he nature of statistical l earning t heory. *New York: Springer2V erlag*, 1995.

[267] P-CG Vassiliou. Fuzzy semi-markov migration process in credit risk. *Fuzzy Sets and Systems*, 223:39–58, 2013.

[268] Alfredo Vellido, Paulo JG Lisboa, and J Vaughan. Neural networks in business: a survey of applications (1992–1998). *Expert Systems with applications*, 17(1):51–70, 1999.

[269] Antanas Verikas, Zivile Kalsyte, Marija Bacauskiene, and Adas Gelzinis. Hybrid and ensemble-based soft computing techniques in bankruptcy prediction: a survey. *Soft Computing*, 14(9):995–1010, 2010.

[270] Veronica Vinciotti and David J Hand. Scorecard construction with unbalanced class sizes. 2003.

[271] Gang Wang, Jinxing Hao, Jian Ma, and Hongbing Jiang. A comparative assessment of ensemble learning for credit scoring. *Expert Systems with Applications*, 38:223–230, 2011.

[272] Ping Wang, Yan Li, and Chandan K Reddy. Machine learning for survival analysis: A survey. *ACM Computing Surveys (CSUR)*, 51(6):1–36, 2019.

[273] Su-Nan Wang and Jian-Gang Yang. A money laundering risk evaluation method based on decision tree. In *2007 International Conference on Machine Learning and Cybernetics*, volume 1, pages 283–286. IEEE, 2007.

[274] Xiaofei Wang. Machine learning-driven credit risk modelling using smartphone metadata. In *Proceedings of the 2019 Credit Scoring and Credit Control Conference, Edinburgh, UK*, volume 31.

[275] Yongqiao Wang, Shouyang Wang, and Kin Keung Lai. A new fuzzy support vector machine to evaluate credit risk. *IEEE Transactions on Fuzzy Systems*, 13(6):820–831, 2005.

[276] Ronald L Wasserstein and Nicole A Lazar. The asa statement on p-values: context, process, and purpose, 2016.

[277] William W. S. Wei. *Time Series Analysis: Univariate and Multivariate Models, 2nd Edition*. Addison-Wesley Pub Co, 1990.

[278] Yanhao Wei, Pinar Yildirim, Christophe Van den Bulte, and Chrysanthos Dellarocas. Credit scoring with social network data. *Marketing Science*, 35(2):234–258, 2016.

[279] D West. Neural network credit scoring models. *Comput Opns Res*, 27:1131, 2000.

[280] David West, Scott Dellana, and Jingxia Qian. Neural network ensemble strategies for financial decision applications. *Computers & operations research*, 32(10):2543–2559, 2005.

[281] T. Windeatt and G. Ardeshir. Decision tree simplification for classifier ensembles. *International Journal of Pattern Recognition*, 18(5):749–776, 2004.

[282] D. H. Wolpert. Stacked generalization. *Neural Networks*, 5(2):241–259, 1992.

[283] Jeffrey Wooldridge. *Econometric Analysis of Cross Section and Panel Data, Second Edition*. The MIT Press, 2010.

[284] Wenbing Xiao, Qian Zhao, and Qi Fei. A comparative study of data mining methods in consumer loans credit scoring management. *Journal of Systems Science and Systems Engineering*, 15(4):419–435, 2006.

[285] Xiujuan Xu, Chunguang Zhou, and Zhe Wang. Credit scoring algorithm based on link analysis ranking with support vector machine. *Expert Systems with Applications*, 36(2):2625–2632, 2009.

[286] Y. Yang and K.C. Land. *Age-Period-Cohort Analysis*. Taylor and Francis, Boca Raton, 2014.

[287] Yingxu Yang. Adaptive credit scoring with kernel learning methods. *European Journal of Operational Research*, 183(3):1521–1536, 2007.

[288] Mumine B Yobas, Jonathan N Crook, and Peter Ross. Credit scoring using neural and evolutionary techniques. *IMA Journal of Management Mathematics*, 11(2):111–125, 2000.

[289] Lean Yu, Shouyang Wang, and Kin Keung Lai. An intelligent-agent-based fuzzy group decision making model for financial multicriteria decision support: The case of credit scoring. *European journal of operational research*, 195(3):942–959, 2009.

[290] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness constraints: Mechanisms for fair classification. *arXiv preprint arXiv:1507.05259*, 2015.

[291] Antonella Zanobetti, MP Wand, J Schwartz, and LM Ryan. Generalized additive distributed lag models: quantifying mortality displacement. *Biostatistics*, 1(3):279–292, 2000.

[292] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International Conference on Machine Learning*, pages 325–333, 2013.

[293] Defu Zhang, Xiyue Zhou, Stephen CH Leung, and Jiemin Zheng. Vertical bagging decision trees model for credit scoring. *Expert Systems with Applications*, 37(12):7838–7843, 2010.

[294] Wenyu Zhang, Hongliang He, and Shuai Zhang. A novel multi-stage hybrid model with enhanced multi-population niche genetic algorithm: An application in credit scoring. *Expert Systems with Applications*, 121:221–232, 2019.

[295] Xiaofei Zhou, Wenhan Jiang, Yong Shi, and Yingjie Tian. Credit risk evaluation with kernel-based affine subspace nearest points learning method. *Expert Systems with Applications*, 38(4):4272–4279, 2011.

[296] XiYue Zhou, DeFu Zhang, and Yi Jiang. A new credit scoring method based on rough sets and decision tree. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 1081–1089. Springer, 2008.

[297] Xun Zhou, Sicong Cheng, Meng Zhu, Chengkun Guo, Sida Zhou, Peng Xu, Zhenghua Xue, and Weishi Zhang. A state of the art survey of data mining-based fraud detection and credit scoring. In *MATEC Web of Conferences*, volume 189, page 03002. EDP Sciences, 2018.

[298] Indre Zliobaite, Faisal Kamiran, and Toon Calders. Handling conditional discrimination. In *11th IEEE International Conference on Data Mining, ICDM 2011, Vancouver, BC, Canada, December 11-14*, pages 992–1001, 2011.