CrisisModeler:

A Tool for Exploring Crisis Predictions

Markus Holopainen
RiskLab Finland
Arcada University of Applied Sciences
Helsinki, Finland
markus@risklab.fi

Peter Sarlin
Department of Economics
Hanken School of Economics, Helsinki, Finland
RiskLab Finland
Arcada University of Applied Sciences
peter@risklab.fi

***Abstract*— The CrisisModeler tool presented in this paper allows for exploring financial crisis predictions. Despite wide interest in crisis prediction, little attention has been given to generalizable modeling solutions, real-time implementations, thorough comparisons among methods and interactive interfaces to explore models. This paper combines many approaches used in predicting financial crises within a fully-fledged framework for modeling and evaluation, and provides an implementation of a general-purpose tool with a web-based interactive interface to explore model output. We illustrate the use of the CrisisModeler with a case study on European banks, including a horse race of methods and investigations of different specifications. The case study illustrates the versatility and suitability of the tool for supporting exploration and communication of models for crisis prediction.**

Keywords—CrisisModeler, financial risk, bank stress, systemic risk, early-warning models

I. Introduction

This paper provides a general-purpose tool for exploring financial crisis predictions. The global financial crisis has stimulated a multitude of efforts into deriving new models and techniques for measuring systemic risk and predicting various types of adverse financial scenarios. Despite wide interest in models, little attention has been given to generalizable modeling solutions, real-time implementations, thorough comparisons among methods and interactive interfaces to explore models. The provided CrisisModeler tool enables interactive means for deriving and exploring predictive models.

While not being a new invention, the literature on predictive models, or so-called early-warning models, has exploded in the past years. The first early-warning models relied less on advanced statistical methods and computers, as financial ratio analysis was more of a handcraft (e.g., Ramser and Foster [28]; Fitzpatrick [18]). After contributions by Beaver [7] on univariate discriminant analysis (DA), next steps moved toward multivariate analysis in Altman [3]. After the early applications by Frank and Cline [20] and Taffler and Abassi [36] of DA for predicting sovereign debt crises, a large wave of currency crisis models were introduced in the mid-1990s, including Eichengreen and Rose [16] and Frankel and Rose [19]. Likewise, the recent wave of banking and systemic financial crises has triggered a large number of efforts targeted on these specific events, such as Alessi and Detken [1], Lo Duca and Peltonen [25] and Holopainen and Sarlin [23]. Despite abundant methods ranging from conventional statistics to more recent machine learning, as well as a multitude of other general advances, little work has targeted improvements in the general framework behind modeling.

This paper fills the gap of a general-purpose tool for crisis predictions. We provide a fully-fledged modeling and evaluation framework targeted for early-warning models, including a large number of different modeling and evaluation approaches. Further, we extend this with a visual interface that allows end-users to interactively manipulate parameters while observing how modeling output changes. The CrisisModeler tool described in this paper is illustrated with a unique European dataset on a large number of banks. Hence, we provide real-world evidence of CrisisModeler at work, which allows us to showcase how the tool can be applied as well as results for a large number of different modeling parameters. Due to a large number of tables, we provide most of the results tables in a web appendix. In the same vein, we accompany the paper with an online browser-based implementation of the CrisisModeler application.[1]

The paper is structured as follows. Section II presents the CrisisModeler tool, while Section III provides its application to European banks. Section IV concludes.

II. CrisisModeler

This section presents technical details and the general implementation of the CrisisModeler.

A. The decision problem in crisis prediction

Early-warning models are in need of evaluation criteria that account for the nature of the underlying problem, which relates to events with high impact, yet low probability. It is thus crucial that the evaluation framework, which optimizes the classifier threshold, accounts for the decision problem faced by a decisionmaker. The signal evaluation framework focuses on a decisionmaker with relative preferences between type I and II errors, and the usefulness that she derives by using a model, in relation to not using it. In the vein of the loss-function approach proposed by Alessi and Detken [1], the

[1] The browser-based implementation of CrisisModeler can be found at http://cm.infolytika.com/ and the web appendix at www.risklab.fi/cm.

framework applied here follows the updated version of Sarlin [32].

For the problem at hand, we need two types of data: historical distress events and indicators of distress. To mimic an ideal leading indicator, we build a binary state variable $C_n (h) \in \{0,1\}$ for observation n (where $n = 1,2,\ldots,N$) given a specified forecast horizon h. Let $C_n (h)$ be a binary indicator that is one during pre-crisis periods and zero otherwise. For detecting events C_n using information from indicators, we need to estimate the probability of being in a vulnerable state $p_n \in [0,1]$. Herein, we make use of a number of different methods m for estimating p_n^m, ranging from the standard logistic regression approach to more sophisticated techniques from machine learning. The probability p_n is turned into a binary prediction B_n, which takes the value one if p_n exceeds a specified threshold $\tau \in [0,1]$ and zero otherwise. The correspondence between the prediction B_n and the ideal leading indicator Cn can then be summarized into a so-called contingency matrix.

The frequencies of prediction-realization combinations in the contingency matrix can be used for computing measures of classification performance. A decisionmaker can be thought of to be primarily concerned with two types of errors: issuing a false alarm and missing a crisis. The evaluation framework described below is based upon that in Sarlin [30] for turning decisionmaker's preferences into a loss function, where the decisionmaker has relative preferences between type I and II errors. While type I errors represent the share of missed crises to the frequency of crises $T_1 \in [0,1] = FN/(TP+FN)$, type II errors represent the share of issued false alarms to the frequency of tranquil periods $T_2 \in [0,1] = FP/(FP+TN)$. Given probabilities p_n of a model, the decisionmaker then finds an optimal threshold τ^* such that her loss is minimized. The loss of a decisionmaker includes T_1 and T_2, weighted by relative preferences between missing crises (μ) and issuing false alarms ($1-\mu$). By accounting for unconditional probabilities of crises $P_1 = \Pr(C=1)$ and tranquil periods $P_2 = \Pr(C = 0) = 1-P_1$, as classes are not of equal size and errors are scaled with class size, the loss function can be written as follows:

$$L(\mu) = \mu T_1 P_1 + (1 - \mu) T_2 P_2, \qquad (1)$$

where $\mu \in [0, 1]$ represents the relative preferences of missing crises and $1-\mu$ of giving false alarms, T_1 the type I errors, and T_2 the type II errors. P_1 refers to the size of the crisis class and P_2 to the size of the tranquil class. Further, the Usefulness of a model can be defined in a more intuitive manner. First, the absolute Usefulness (U_a) is given by:

$$U_a(\mu) = \min(\mu P_1, (1 - \mu) P_2) - L(\mu), \qquad (2)$$

which computes the superiority of a model in relation to not using any model. As the unconditional probabilities are commonly unbalanced and the decisionmaker may be more concerned about the rare class, a decisionmaker could achieve a loss of $\min(\mu P_1, (1-\mu) P_2)$ by either always or never signalling a crisis. This predicament highlights the challenge in building a Useful early-warning model: With an imperfect model, it would otherwise easily pay off for the decisionmaker to always signal the high-frequency class. Second, we can compute the relative Usefulness U_r as follows:

$$U_r(\mu) = U_a(\mu) / \min(\mu P_1, (1 - \mu) P_2), \qquad (3)$$

where U_a of the model is compared with the maximum possible Usefulness of the model. That is, the loss of disregarding the model is the maximum available Usefulness. Hence, U_r reports U_a as a share of the Usefulness that a decisionmaker would gain with a perfectly-performing model, which supports interpretation of the measure. It is worth noting that U_a better lends to comparisons over different μ.

Beyond the above measures, the contingency matrix may be used for computing a wide range of other quantitative measures.[2] Receiver operating characteristics (ROC) curves and the area under the ROC curve (AUC) are also used for comparing performance of early-warning models. The ROC curve plots, for the complete range of $\tau \in [0, 1]$, the conditional probability of positives to the conditional probability of negatives:

$$ROC = \Pr(P = 1 \mid C = 1) / (1 - \Pr(P = 0 \mid C = 0)). \qquad (4)$$

B. *Modeling techniques*

This section presents the methods we use for the task of classification. Generally, classification is considered an instance of supervised learning, out of which we make use of a number of probabilistic classifiers, whose outputs are probabilities indicating membership to two qualitative classes (pre-crisis or tranquil periods). Starting from the benchmark method of logistic regression, we cover five more advanced machine learning approaches.

As is common in machine learning, we include four ensemble learning approaches mainly based on so-called bagging and boosting. Boosting [34] refers to computing output with several models and averaging results, whereas bagging [9] uses resampling from the original data and aggregates into one model output. We follow the ensemble approaches proposed in Holopainen and Sarlin [23], which were applied to banking crisis prediction. Two of the approaches aggregate the in-sample probabilities of all methods to a mean (arithmetic or weighted), after which these in-sample mean probabilities are treated as if they were outputs of a single method, and the classifier is then calibrated accordingly, as below explained in detail. The third ensemble simply uses the probabilities of the method which performs best in-sample, whereas the fourth and most simple ensemble disregards probabilities and computes its output based on a majority vote of the binary signals of all methods.

[2] Some of the commonly used evaluation measures include: Recall positives (or TP rate) = TP/(TP+FN), Recall negatives (or TN rate) = TN/(TN+FP), Precision positives = TP/(TP+FP), Precision negatives = TN/(TN+FN), Accuracy = (TP+TN)/(TP+TN+FP+FN), FP rate = FP/(FP+TN), and FN rate = FN/(FN+TP).

1) Logit analysis: Commonplace in early-warning literature, logit analysis (also known as logistic regression) uses the logistic function to describe the probability of an observation belonging to one of two classes, based on a regression of one or more continuous predictors. For the case with one predictor variable, the logistic function is $p(X) = e^{\beta_0+\beta_1 X} / (1+e^{\beta_0+\beta_1 X})$. This function is easily extended to the case of several predictors. Logit and probit models have frequently been applied to predicting financial crises, including Eichengreen and Rose [16], Frankel and Rose [19], Sachs et al. [29]), Barrell et al. [6] and Lo Duca and Peltonen [25]. Yet, the distributional (logistic/normal) assumption on the relationship between the indicators and the response as well as the absence of interactions between variables may often be violated. Lo Duca and Peltonen [25], for instance, show that the probability of a crisis increases nonlinearly as the number of fragilities increase.

2) k-Nearest Neighbors (KNN): The method of *k*-nearest neighbors is a nonparametric method commonly used for classification (see, e.g. Altman [4]), where the method assigns an observation to the class most common among its *k* nearest neighbors in the training data. The method is considered to be among the simplest in the realm of machine learning. Given a positive integer k and an observation x_0, the algorithm first identifies the k points x_k in the training data closest to x_0, commonly by computing the Euclidean distance between x_0 and $x_1 ... x_k$. Then, the average output value of these k points is calculated. In this paper, we use the Minkowski distance to determine the nearest neighbors, as well as a kernel function (see e.g. Hechenbichler and Schliep [22]) which returns similarity measures of the neighbors based on proximity, allowing the output to be calculated as an average giving more weight to closer neighbors. We utilize the 'optimal' weighting kernel proposed in Samworth [30], and consider two free parameters, the integer k and a parameter p which determines the order of the Minkowski distance. The kNN method was shown to perform well in the horse race by Holopainen and Sarlin [23].

3) Classification trees: Classification trees, as discussed by Breiman et al. [10], implement a tree-type structure, which reach a decision by performing a sequence of tests on the values of the predictors. In a classification tree, the classes are represented by leaves, and the conjunctions of predictors are represented by the branches leading to the classes. These conjunction rules segment the predictor space into a number of simpler regions, allowing for decision boundaries of complex shapes. The method has proven successful in many areas of machine learning, and has the advantage of high interpretability. To reduce complexity and improve generalization ability, cross-validation is oftentimes used to prune sections of the tree until optimal out-of-sample performance is reached. One free parameter is considered, which relates to the desired complexity of the tree when it is being pruned. In the early-warning literature, the use of classification trees has been fairly common, including Kaminsky [24], Schimmelpfennig et al. [35], Chamon et al. [11] and Duttagupta and Cashin [15].

4) Random forest: The random forest method, introduced by Breiman [10], uses classification trees as building blocks to construct a more sophisticated ensemble-like method, which in general outperforms a single classification tree, although at the expense of interpretability. The method grows a pre-defined number of classification trees based on differently sampled subsets of the data. Additionally, at each split, a randomly selected sample of a pre-defined size is drawn from the full set of predictors. Only predictors from this sample are considered as candidates for the split, effectively forcing diversity in each tree. Lastly, the average of all trees is calculated. As the training data for each tree is sampled differently, this reduces correlation between the so-called bootstrapped trees, leading to a reduction in variance in the average. In this paper, two free parameters are considered: the number of trees, and the number of predictors sampled as candidates at each split. To the best of our knowledge, random forests have only been applied to early-warning exercises in Alessi and Detken [2] and Holopainen and Sarlin [23].

5) Artificial Neural Networks (ANN): Inspired by the functioning of neurons in the human brain, Artificial Neural Networks (ANNs) are composed of nodes or units connected by links (see, e.g., Venables and Ripley [38]). Each link between two units controls the flow of information and is associated with a weight, which serves to activate the receiving unit if certain conditions are met. These weights act as network parameters that are tuned iteratively by a learning algorithm. The simplest type of ANN is the single hidden layer feed-forward neural network (SLFN), which has one input, hidden and output layer. The input layer distributes the input values to the units in the hidden layer, whereas the units in the output layer compute the weighted sum of the inputs from the hidden layer, in order to yield a classifier probability. However, as neural networks grow in size, computation time increases exponentially and their interpretability diminishes. In this paper, a basic SLFN is used, with two free parameters: the number of units in the hidden layer and the weight decay. The first parameter controls the complexity of the network, while the second is used to control how the learning algorithm converges. ANNs have been applied to crisis prediction, including Nag and Mitra [26], Peltonen [27], Fioramanti [17] and Sarlin and Marghescu [33]. Further, Sarlin [31] used an ANN optimized with a genetic algorithm for predicting systemic financial crises.

6) Support Vector Machines (SVM): The Support Vector Machine, introduced by Cortes and Vapnik [14], is one of the most popular machine learning methods for supervised learning. It is a nonparametric method that, for classification, uses hyperplanes in a high-dimensional space to construct a decision boundary for a separation between classes. It comes with several desirable properties. First, a support vector machine constructs a maximum margin separator, i.e. the chosen decision boundary is the one with the largest possible distance to the training data points, enhancing generalization performance. Second, it relies on support vectors when constructing this separator, and not on all the data points, such as in logistic regression. These properties lead to the method having high flexibility, but still being somewhat resistant to overfitting. However, support vector machines generally lack interpretability. The free parameters considered are: the cost parameter, which affects the tolerance for misclassified observations when constructing the separator, and the gamma parameter, defining the area of influence for a support vector. In the horse race of Holopainen and Sarlin [23], SVMs were shown to be among the best-in-class approaches for predicting systemic banking crises.

7) Ensemble learning: Our first two ensembles are based on averages of the methods. Ensemble I is simply implemented as an observation-wise arithmetic mean of the probabilistic outputs of all methods above. Ensemble II is calculated as a weighted mean of the probabilistic outputs of all methods above. For each observation j, the weight of method i out of n models is calculated as

$$w_{i,j} = U_{i,j} \,/\, \Sigma_{k=1}^{n} U_{k,j} \,, \tag{5}$$

where $U_{i,j}$ is the in-sample Usefulness for observation i of method j. In the event of one or more methods having negative Usefulness, the following changes are made to the weights. If one or more methods have negative Usefulness, their weights are set to zero, removing them from the ensemble. If the Usefulness-values of all methods are negative, only the best method is used (as equal to Ensemble III). If the Usefulness-values of all methods are missing, all methods are given identical weights (as equal to Ensemble I). Ensemble III makes use of the single best method for each recursion, as determined by the highest in-sample Usefulness. The final Ensemble IV is based on the principle of voting, and simply utilizes the signals of all six methods in order to signal or not, based on a majority vote. In this paper, we signal if at least three methods signal, which results in a more cautious ensemble.

C. *Modeling strategies*

With the objective of deriving models for out-of-sample prediction, we outline herein the strategies used in this paper. We tackle the problem in two separate parts: the model selection procedure and the evaluation exercise.

As five of the methods presented in Section II.B include free parameters which in different ways control the complexity of each of these methods, they need to be optimized based on the data set at hand. For this task we make use of a so-called grid search. A set of values to be tested are selected based on common rules of thumb for each parameter (i.e., usually minimum and maximum values and regular steps in between), after which a grid search is performed on the discrete parameter space of the Cartesian product of the parameter sets. To avoid overfitting as the result of parameters of too high complexity, we employ 10-fold cross-validation and rank the parameter choices of each method based on out-of-sample Usefulness. After this the single parameter (or the parameter combinations, for methods with several free parameters) yielding the highest out-of-sample Usefulness are chosen and used to calibrate the model.

As pooled models with panel data are common in the literature, data generally include a cross-sectional and time dimension. Thus, we ought to consider that data is likely to exhibit temporal dependencies. Although the cross-validation literature has put forward advanced techniques to decrease the impact of dependence (see e.g. Arlot and Celisse [5]), the most prominent approach is to limit estimation samples to historical data for each prediction. In order to test our models from the viewpoint of real-time analysis, we implement the recursive exercise as in Holopainen and Sarlin [23], which derives a new model at each quarter using only data available up to that point in time. The exercise enables testing whether the use of classification models would have provided means for predicting future events, and how different techniques rank in terms of performance for the task. The recursive algorithm proceeds as follows. For each quarter q (or other chosen frequency), we estimate a model based on all available information up to that point $t = 0, 1, \ldots, q-1$ and predict the out-of-sample values for $t = q$. The in-sample probabilities for $t = 0, 1, \ldots, q-1$ are used to find an optimal threshold τ^*, which is used for $t = q$ to generate out-of-sample signals. Thus both the optimal threshold and the models themselves are time-varying. At the end, we collect all predictions and evaluate how well the model has performed in out-of-sample analysis. The same exercise is then performed for all separate methods as well as for the ensembles, as outlined in Section II.B.

D. *Interaction with CrisisModeler*

The CrisisModeler tool is implemented in R with a web-browser interface, as is shown in Fig. 1. Thus, a web-server implementation requires no further installation of software or in-depth technical knowledge, as the end-user needs only to interact with the browser-based user interface. The main view of the application consists of a left-side panel with settings and parameters, and a main page for the output of the exercise. As changes in exercise or method parameters are made in the left-hand panel, the main page calculates a new output based on the chosen parameters. The user is given the possibility to load their own data with specific indicators and distress events. Based on the data, CrisisModeler constructs an ideal leading indicator (with a specific forecast horizon), trains the selected probabilistic classifiers, computes their optimal thresholds and returns binary classifications. The models account for the preference between type I and II errors, as specified by the user.

The left panel includes manual input of data, a number of parameters relating to the exercise output (such as the pre-crisis and post-crisis horizon intervals, and the preference between type I and II errors), as well as checkboxes enabling the choice of single methods along with their corresponding parameters. The main page output is, by default, the performance results of the recursive exercise with the chosen exercise parameters and methods in a table format. The performance measures include Usefulness, the area under the ROC curve (denoted AUC), as well as a large number of other common measures. This table, as well as the out-of-sample data, including output probabilities, thresholds and predictions, is downloadable in csv format for reference and further analysis. Additional views include various other model details, including visualizations of model output of individual methods, summaries of modeling parameters for chosen methods, and model descriptions for more details of each method.

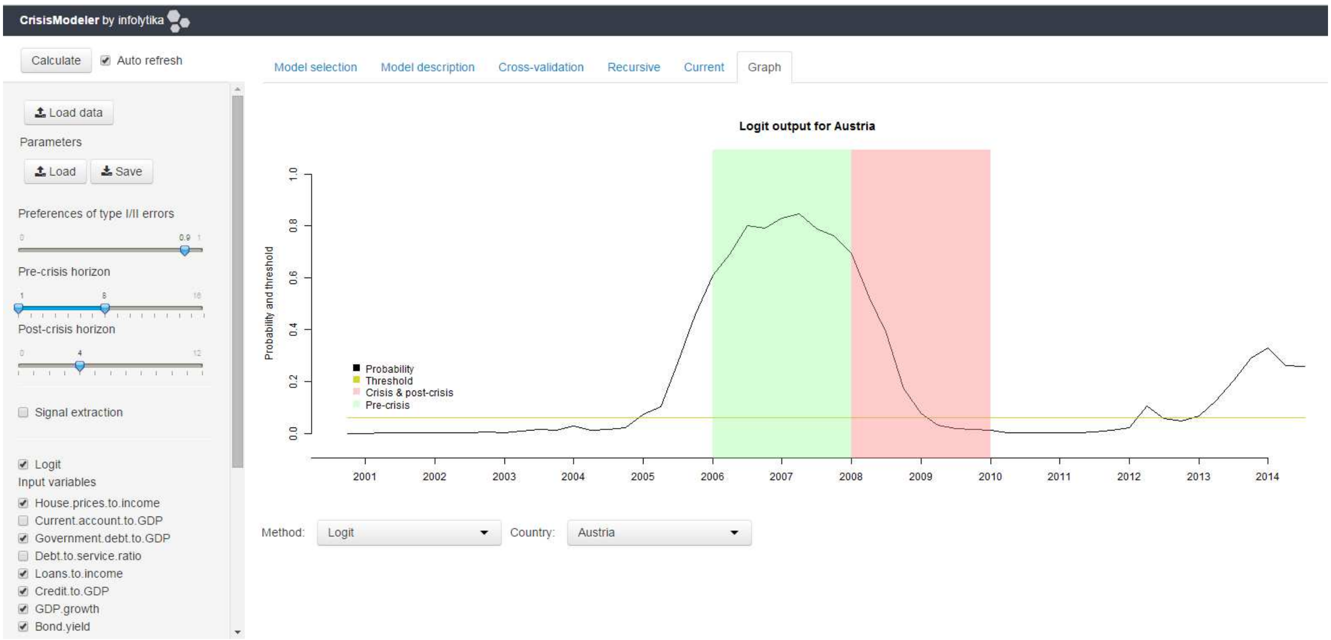

FIGURE 1. A SCREENSHOT OF THE CRISISMODELER

III. CRISISMODELER AT WORK

This section presents an application of CrisisModeler to predict bank distress in Europe. In the following, we describe the used data and prediction results for a number of model specifications.

A. *Crises and indicators*

In order to derive early-warning models for European banks, we use a data set based on a large number of different sources of publicly available data (following Betz et al. [8]). The data is collected, for an observation period from 2000Q1 to 2014Q3, on 546 banks with a minimum of EUR 1bn in total assets, resulting in a total of 29547 quarterly observations. Thus, the data set covers large banks relevant to systemic risk due to interlinkages and interconnectedness. We utilize information which would have been accessible at each specific point in time. Data reported annually are used in the data set for the subsequent four quarters, and publication lags are accounted for.

As European bank failures have been extremely rare, little data is available on bank defaults in the strict sense, which causes challenges in identification of crisis events. To tackle this, the data set also accounts for state aid and forced mergers, in addition to bankruptcies, liquidations and defaults.

Direct bank failures are captured as follows. A bankruptcy is defined to occur if the net worth of a bank falls below the guidelines of the country in question, and a liquidation is defined to occur if a bank is sold according to the guidelines of the liquidator, in which case shareholders may not be compensated in full. Defaults are defined as either when a bank has failed to pay interest or principal on at least one financial obligation outside any grace period specified in the terms, or when a bank finalizes a distressed exchange, in which at least one financial obligation is repurchased or replaced by other instruments with a diminished total value. The data source for the bankruptcies and liquidations is Bankscope, whilst annual default data is retrieved from Moody's and Fitch. A distress event is defined to start when distress is announced, and ends when the actual event occurs.

Next, we include data on state intervention to identify banks in distress. A bank is defined to be in distress if it receives a capital injection by the state or participates in an asset relief program. The events are based on data from the European Commission with accompaniments market sources (Bloomberg and Reuters). As above, the events are defined to start from the time of announcement to the execution of the state support program.

Finally, merged entities are defined to be in distress if either a parent receives state support within 12 months after the merger, or if a merged entity has a coverage ratio below zero within 12 months before the merger. The reasoning behind only including this rule for mergers is that a single bank may still survive with a negative coverage ratio, whereas merged entities may have been forced to do so due to distress. The coverage ratio is calculated as the ratio of capital equity and loan reserves minus nonperforming loans to total assets. Merger data is obtained from Bankscope, and data for the coverage ratio is retrieved from Bloomberg. The events obtained are cross-checked using market sources (Reuters and Bloomberg) to avoid possible mismatches. The events are defined as to start when a merger occurs and to end when the parent receives state support, and to start when the coverage ratio falls below zero and to end when the merger occurs.

We use a forecast horizon of 8 quarters, meaning that the binary pre-distress variable is defined as the value 1 in 1-8 quarters prior to the actual distress event as defined above, and 0 otherwise.

The explanatory variables used are chosen from three separate classes following a micro-macro perspective with the aim to capture underlying vulnerabilities. In addition to bank-specific balance-sheet and income-statement indicators, we complement the data with country-specific indicators for the banking sector, as well as with country-specific measures of macroeconomic and financial imbalances. The bank-specific indicators chosen account for all dimensions in the CAMELS rating system and are constructed using Bloomberg data. The banking-sector-specific indicators proxy for imbalances at the banking system level and are calculated using statistics from the Balance Sheet Items of the Monetary, Financial Institutions and Markets as obtained from the ECB. The third and final category of variables consists of selected internal and external indicators of the EU Macroeconomic Imbalance Procedure (MIP) which identify country-specific macro-economic imbalances. They are obtained from Eurostat and Bloomberg, complemented with house price indicators from the ECB.

From the large number of indicators covering the above-mentioned classes, the twelve most relevant are determined by the LASSO (Least Absolute Shrinkage and Selection Operator, see Tibshirani [37]) procedure to be used in modeling. These are presented ordered by class in Table I. This choice of variables leads to a data set with 9776 quarterly observations with no missing values, containing 292 distress observations and 1052 pre-distress observations.

TABLE I. VARIABLES

Class	Variable
Bank	Tangible capital to assets Interest expenses to liabilities Reserves to assets
Sector	Financial assets to GDP Mortgages to loans, 1-year change Securities to liabilities, 1-year change
Macro	Total credit to GDP Total credit to GDP, 3-year change House price deviation from trend International investment position to GDP Private sector debt to GDP 10-year bond yield, 1-year change

TABLE II. MODEL SELECTION

Method	Parameters	
Trees	Complexity parameter = 0.01	
KNN	k = 17	Distance = 4
Random forest	No. of trees = 20	No. of predictors sampled = 9
NN	No. of units = 100	Weight decay = 0.01
SVM	Gamma = 0.05	Cost = 15

B. *A horse race of modeling techniques*

A horse race of methods involves choosing a number of specifications. Following the reasoning in Bussière and Fratzscher [12], we account for post-crisis and crisis bias by not including the period when an actual crisis occurs or one year thereafter. These periods of time are not considered useful data for training, as they represent neither a vulnerable pre-crisis period nor a period of tranquility. These observations are thus dropped from all in-sample data used, whereas testing data for each recursion is kept intact. For comparability reasons, the out-of-sample probabilities are transformed to reflect the distribution of the in-sample data, utilizing the empirical cumulative distribution function. Using this function, both the in-sample and the out-of-sample probabilities are converted to percentiles of the in-sample probabilities for each recursion.

As above discussed, we perform model selection with a grid search. This provides a set of optimal model parameters, as summarized in Table II. To issue binary signals based on probabilities p_n, an optimal threshold τ^* needs to be specified for each method and for each recursion. The threshold is set to optimize the in-sample Usefulness in regard to a specified preference μ of the decisionmaker. Following the reasoning that a positive signal is only a call for internal investigation and that the negative repercussions of false alarms are low, we assume the benchmark preference μ to be 0.9. The recursive exercise is performed as follows. The algorithm is repeated for all quarters from 2007Q1 to 2013Q1, where the first is defined as the starting quarter for our exercise, and the second quarter is the ending quarter.

Table III shows the results of the recursive horse race with all six methods. It may firstly be noted that the more complex machine learning methods outperform methods designed for interpretability, namely classification trees and the conventional logit method, when ranked by descending Usefulness. The AUC for the top two methods, KNN and Random forest, is significantly higher than the rest, suggesting robust performance.

C. *Aggregating model output*

In addition to using a single technique or many techniques alongside each other, the logical next step is to aggregate the methods into one output. As outlined in Section II.B, we derive four ensembles, which consist of two averages (arithmetic mean and weighted mean) of all probabilities, as well as an in-sample best-of-method and finally an ensemble based on a majority vote. The results of the ensemble horse race are shown in Table IV. In general, the ensembles perform well across the board. The voting ensemble has a higher bias towards false positives due to its construction as it signals distress based on positive signals from at least three methods. However its' Usefulness is still the highest out of all ensembles. The best-of ensemble is identical to the Random forest method, which has experienced the best in-sample performance over all recursive quarters. For all probability-based ensembles, the AUC is also high (AUC cannot be calculated for the voting ensemble as it is not probabilistic).

D. *Varying model specifications*

Next, we study the robustness of our results over different model specifications. This is accomplished by varying one model specification component from the benchmark at a time and studying its effects on the horse race. We look at several aspects of model specification; country-specific versus pooled models, and large banks versus small banks. As in the benchmark results of Tables III and IV, the decisionmaker's preference μ is 0.9.[3]

1) Country-specific vs. pooled models: The notion of preferring pooled models (see e.g. Fuertes and Kalotychou [21]) originates from the desire to model a wide variety of crises, as well as the common shortage of distress events in individual countries. We compare the effects of our pooled models with their country-specific equivalents by first separating the signals of the recursive horse race by individual countries, and then recalculating their out-of-sample performance. For comparison, we then set up new country-specific recursive horse races using only data from one country at a time. These are evaluated out-of-sample, and compared country-wise to the models which have been trained using the entire pooled data set.

Our data set comprises banks from 27 European countries, however, only nine countries are usable for performing the country-specific recursive horse races due to lack of data. Of these nine countries, we present the horse races of German and French banks in Tables A.I and A.II, and Tables A.III and A.IV, respectively. These countries each represent an adequately large sample, with 764 and 796 observations. For both countries, the un-pooled models outperform the pooled models. However, performance of the pooled models is generally good and country-specific models can only be computed for a few countries.

2) Large vs. small banks: Using the same notion as above, we want to investigate the effects of models trained using data only from large or from small banks, compared to models trained using pooled data. The banks are split according to the median of a variable related to their size, resulting in two samples, with 155 small banks and 163 large banks. The out-of-sample signals of the benchmark pooled models are separated into categories "large banks" and "small banks" and their performance is re-evaluated. These results are then compared to models trained using only data of large banks and data of small banks, respectively.

The results of the horse races for small banks are shown in Tables A.V and A.VI in the Appendix, where the former is based on the benchmark pooled models, and the latter is the corresponding where data of small banks has been used for training of the models. The differences in Usefulness overall are minor between the two approaches, notably the SVM does not perform well in the models trained with small bank data (Table A.VI) and consequently affects the performance of the arithmetic mean ensemble. In general, the ensembles perform well.

The corresponding results for large banks are shown in Tables A.VII and A.VIII in the Appendix, where the former is based on the pooled models and the latter trained using data from large banks only. The SVM does not perform well, probably due to an overfit, but when comparing the two tables over all other techniques there are only minor differences. It may be noted that the Usefulness values are slightly lower than both the benchmark results and the results for the small banks, suggesting that distress prediction of larger banks is more challenging for this sample.

IV. Conclusion

Despite a multitude of recent approaches for modeling financial crises, little or no efforts have targeted improvements in general frameworks and infrastructure behind modeling. This paper combines a large number of approaches used in predicting financial crises, as well as provides an implementation of a general-purpose tool with an interactive interface to explore model output.

The CrisisModeler tool presented in this paper allows exploring crisis predictions in an early-warning fashion. The CrisisModeler tool enables calibration of classifiers in regards to a loss function with preferences between type I and II errors, thus mimicking the real-world decision problem faced by the decisionmaker. The tool includes a palette of classification methods, including four derived ensembles, as well as implementations of recursive modeling exercises for the methods to mimic a real-time setup. The paper presents essential features of the interface to the web-based CrisisModeler application, including a large number of method and exercise parameters that allow for end-user interaction with the modeling task.

This paper has also applied the CrisisModeler to a European bank-level data set with the goal of predicting bank distress. The used data set features low probability, high impact distress events, thus underlining the relevance of accounting for decisionmaker preferences when optimizing

TABLE III. RECURSIVE HORSE RACE

					Positives		Negatives							
Method	**TP**	**FP**	**TN**	**FN**	**Precision**	**Recall**	**Precision**	**Recall**	**Accuracy**	**FP rate**	**FN rate**	**$U_a(\mu)$**	**$U_r(\mu)$**	**AUC**
KNN	573	1108	4892	165	0.34	0.78	0.97	0.82	0.81	0.18	0.22	0.05	57 %	0.858
Random forest	472	491	5509	266	0.49	0.64	0.95	0.92	0.89	0.08	0.36	0.05	52 %	0.870
Neural network	452	1127	4873	286	0.29	0.61	0.94	0.81	0.79	0.19	0.39	0.03	38 %	0.788
SVM	497	1639	4361	241	0.23	0.67	0.95	0.73	0.72	0.27	0.33	0.03	37 %	0.781
Logit	471	1518	4482	267	0.24	0.64	0.94	0.75	0.74	0.25	0.36	0.03	35 %	0.788
Trees	352	956	5044	386	0.27	0.48	0.93	0.84	0.80	0.16	0.52	0.02	26 %	0.636

TABLE IV. RECURSIVE HORSE RACE, ENSEMBLES

					Positives		Negatives							
Method	**TP**	**FP**	**TN**	**FN**	**Precision**	**Recall**	**Precision**	**Recall**	**Accuracy**	**FP rate**	**FN rate**	**$U_a(\mu)$**	**$U_r(\mu)$**	**AUC**
Voting	547	1063	4937	191	0.34	0.74	0.96	0.82	0.81	0.18	0.26	0.05	54 %	NA
Best-of	472	491	5509	266	0.49	0.64	0.95	0.92	0.89	0.08	0.36	0.05	52 %	0.870
Weighted	499	942	5058	239	0.35	0.68	0.95	0.84	0.82	0.16	0.32	0.04	48 %	0.857
Non-weighted	486	882	5118	252	0.36	0.66	0.95	0.85	0.83	0.15	0.34	0.04	48 %	0.851

classifier thresholds. In addition to the horse race - allowing for direct comparison between different methods - the tool was used to investigate a few different setups, such as country-specific versus pooled models and the use of only small versus large banks for model training. The data set, spanning a relatively short period of time due to data availability, causes an ambiguous scenario where general performance is not optimal. This highlights the importance of interactive means for exploring model performance over different modeling specifications. Some methods were shown to perform slightly better than others, but notably the ensembles were consistently among best-in-class methods. For a selected sample of countries, we have shown that country-specific models outperform pooled models in a bank-level setting, yet this is far from a generalizable feature. As an application of the CrisisModeler tool, this case study has illustrated the versatility and suitability of the tool for supporting exploration and communication of models for crisis predictions.

REFERENCES

[1] L. Alessi and C. Detken. Quasi real time early warning indicators for costly asset price boom/bust cycles: A role for global liquidity. European Journal of Political Economy, 27(3):520-533, 2011.

[2] L. Alessi and C. Detken. Identifying excessive credit growth and leverage. ECB Working Paper No. 1723, 2014.

[3] E. Altman. Financial Ratios, Discriminant Analysis and the Prediction of Corporate Bankruptcy, Journal of Finance 23: 589–609, 1968.

[4] N. S. Altman. An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician, 46(3):175-185, 1992.

[5] S. Arlot and A. Celisse. A survey of cross-validation procedures for model selection. Statistical Surveys, 4:40-79, 2010.

[6] R. Barrell, P. E. Davis, D. Karim, and I. Liadze. Bank regulation, property prices and early warning systems for banking crises in OECD countries. Journal of Banking & Finance, 34(9):2255-2264, 2010.

[7] W. Beaver. Financial ratios as predictors of failure. Empirical research in accounting: selected studies, Journal of Accounting Research 4: 71–111, 1966.

[8] F. Betz, S. Oprică, T. A. Peltonen, and P. Sarlin. Predicting Distress in European Banks. Journal of Banking & Finance, 45, 225-241, 2014.

[9] L. Breiman. Bagging predictors. Machine Learning, 24(2): 123-140, 1996.

[10] L. Breiman. Random forests. Machine Learning, 45(1):5-32, 2001.

[11] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. Classification and Regression Trees. Wadsworth, Belmont, CA, 1984.

[12] M. Bussière and M. Fratzscher. Towards a new early warning system of financial crises. Journal of International Money and Finance, 25(6):953-973, 2006.

[13] M. Chamon, P. Manasse, and A. Prati. Can we predict the next capital account crisis? IMF Staff Papers, 54:270-305, 2007.

[14] C. Cortes and V. Vapnik. Support-vector networks. Machine Learning, 20(3):273-297, 1995.

[15] R. Duttagupta and P. Cashin. Anatomy of banking crises in developing and emerging market countries. Journal of International Money and Finance, 30(2):354-376, 2011.

[16] B. Eichengreen and A. K. Rose. Staying afloat when the wind shifts: External factors and emerging-market banking crises. NBER Working Paper, No. 6370, 1998.

[17] M. Fioramanti. Predicting sovereign debt crises using artificial neural networks: A comparative approach. Journal of Financial Stability, 4(2):149-164, 2008.

[18] P. Fitzpatrick. A Comparison of the Ratios of Successful Industrial Enterprises with Those of Failed Companies, The Accountants Publishing Company, 1932.

[19] J. A. Frankel and A. K. Rose. Currency crashes in emerging markets: An empirical treatment. Journal of International Economics, 41(3-3):351-366, 1996.

[20] C. Frank and W. Cline. Measurement of Debt Servicing Capacity: An Application of Discriminant Analysis, Journal of International Economics 1: 327–344, 1971.

[21] A.-M. Fuertes and E. Kalotychou. Early warning system for sovereign debt crisis: The role of heterogeneity. Computational Statistics and Data Analysis, 5:1420-1441, 2006.

[22] K. Hechenbichler and K. Schliep. Weighted k-nearest-neighbor techniques and ordinal classification. Discussion Paper 399, SFB 386, Ludwig-Maximilians University Munich, 2004.

[23] M. Holopainen and P. Sarlin. Toward robust early-warning models: A horse race, ensembles and model uncertainty. Bank of Finland Discussion Paper 6, 2015.

[24] G. Kaminsky. Varieties of currency crises. NBER Working Papers, No. 10193, National Bureau of Economic Research, 2003.

[25] M. Lo Duca and T. A. Peltonen. Assessing systemic risk and predicting systemic events. Journal of Banking & Finance, 37(7):2183-2195, 2013.

[26] A. Nag and A. Mitra. Neural networks and early warning indicators of currency crisis. Reserve Bank of India Occasional Papers 20(2), 183-222, 1999.

[27] T. A. Peltonen. Are emerging market currency crises predictable? A test. ECB Working Paper No. 571, 2006.

[28] J. Ramser and Foster, L. A demonstration of ratio analysis. 1931, Bureau of Business Research, University of Illinois,Urbana, IL, Bulletin 40, 1931.

[29] J. Sachs, A. Tornell, and A. Velasco. Financial crises in emerging markets: The lessons from 1995. Brookings Papers on Economic Activity, No. 1:147-218, 1996.

[30] R. J. Samworth. Optimal weighted nearest neighbor classifiers. The Annals of Statistics, 40(5):2733-2763, 2012.

[31] P. Sarlin. On biologically inspired predictions of the global financial crisis. Neural Computing and Applications, 24(3-4):663-673, 2013.

[32] P. Sarlin. On policymakers' loss functions and the evaluation of early warning systems. Economics Letters, 119(1):1-7, 2013.

[33] P. Sarlin and D. Marghescu. Neuro-genetic predictions of currency crises. Intelligent Systems in Accounting, Finance and Management, 18(4):145-160, 2011.

[34] R. E. Schapire. The strength of weak learnability. Machine Learning, 5(2):197{227, 1990.

[35] A. Schimmelpfennig, N. Roubini, and P. Manasse. Predicting sovereign debt crises. IMF Working Papers 03/221, International Monetary Fund, 2003.

[36] R. Taffler and B. Abassi. Country risk: A model for predicting debt-servicing problems in developing countries. Journal of the Royal Statistical Society 147: 541−568, 1984.

[37] R. Tibshirani. Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society, Series B, 58:267–288, 1996.

[38] W. N. Venables and B. D. Ripley. Modern Applied Statistics with S. Fourth edition. Springer, 2002.