**Predicting Endogenous Bank Health from FDIC Statistics  
on Depository Institutions Using Deep Learning**

David Jungreis,1 Noah Capp,1 Meysam Golmohammadi,1 and Joseph Picone1

1 Neural Engineering Data Consortium, Temple University, Philadelphia, PA, 19122, USA

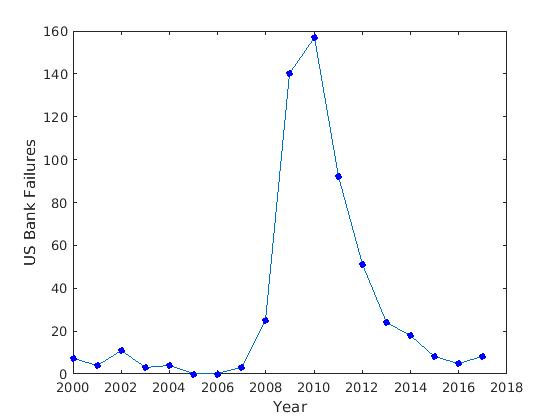
dbjungreis@gmail.com

**Abstract.** The Federal Deposit Insurance Corporation (FDIC) keeps records of banking data in its Statistics on Depository Institutions (SDI) going back to the fourth quarter of 1992. The data are reported quarterly on approximately 1,050 variables, such as total assets, liabilities, and deposits. We hypothesized that impending failure could be predicted from these data. We restricted the database to 60 quantitative variables that had no missing data for any bank in any quarter during the epoch under consideration: the first quarter of 2000 through the second quarter of 2017. Deep learning approaches based on multilayer perceptrons and convolutional neural networks were evaluated and failed to accurately predict failures better than guessing based on priors. These baseline experiments, particularly the inability to overfit to the training data, show the challenges of finding failure-predicting trends that are strictly intrinsic to a bank. Future deep learning work would have to include exogenous factors to link quarterly observations between banks.

**Keywords:** Data Analysis, Bank Health, Convolutional Neural Networks

1. **Introduction**

The frequent turbulence of financial institutions in recent years has generated significant interest in an automated tool to assess bank health. For example, an equity trader might be interested in identifying if any of the publicly traded banks are due to fail like Lehman Brothers and Bear Stearns did during the Global Financial Crisis (GFC) of 2007-2009 [1]. For the investor in more exotic instruments, bank health could be of interest for the trader of derivatives, particularly credit default and interest rate swaps. Lenders would want any advantage in determining risk and appropriate interest to charge on interbank loans. Regulators would want to know which banks are in danger of failing, as many did in the wake of the GFC [2].



**Fig. 1.**US Bank failures spiked in 2009, in the wake of the Global Financial Crisis, peaking in 2010 at 157.

In Fig. 1, we show the number of bank failures per year. Following a scarcity of failed banks in 2000-2007, bank failures began to rise as the effects of the GFC set in, causing a surge of bank failures starting in 2009, peaking at 157 in 2010, and only returning to pre-crisis levels in 2015. Bank failures during the GFC were driven by a combination of leverage, concentration of risk (particularly in mortgages and mortgage derivatives) and a focus on short-term gains at the expense of long-term risk [3].

Automated bank health analysis has been able to achieve strong performance using support vector machines [4-6], fully-connected neural networks [7], and logistic regression [8]. These studies, however, suffer from various shortcomings that we address in this paper. For example, many of these studies (e.g., Erdogan [4][5][6][9] and Boyacioglu [7]) used small data sets (fewer than 100 samples). Second, all six studies considered factors exogenous to the banks. Erdogan and Boyacioglu considered banks from roughly the same eras, limiting variability due to differences in the general health of the economy, while Zheng [8] explicitly included macroeconomic random effects. We address these issues in this work by considering 1,000 banks using variables strictly endogenous to the banks.

State-of-the-art machine learning is based on neural networks, particularly multi-layer networks known as deep learning [10]. These networks have the ability to discover hidden relationships within the data. This is a clear advantage over regression with interaction terms, where investigators must determine which interactions to explore. For instance, Lopez et al. [11][12] investigated EEG normal/abnormal binary classification using traditional forms of machine learning, such as random forests (RF) and k-nearest neighbors (kNN), yet she achieved superior performance when deep learning, in particular, Convolutional Neural Networks (CNN), was used.

Deep learning has been used in finance [13], but for bankruptcy prediction, less advanced techniques are emphasized in the literature [4]-[9]. Moreover, publications seem to address bankruptcy in general, not endogenous bank health for a specific bank. Addo [14] showed that deep learning could be used for determining credit risk (essentially the goal of this paper), but he did not demonstrate the technique on bank data. Tam [15][16], Le [17] and Abdou [18] developed neural network-based models to determine bank bankruptcy risk, but these were not based on modern deep learning architectures such as CNN.

The goal in this work is to assess baseline performance of deep learning systems on operational banking data provided by the U.S. Federal Government [2]. Operational data contain many forms of imperfection that make it extremely difficult for the application of traditional deep leaning systems. Given a consistent number of reporting periods, our goal was to predict if the bank will fail before the end of the next quarter, only considering the fixed effects of the data from the bank, not random effects of exogenous variables.

This is in contrast to previous work on bank failure and bankruptcy prediction, which has incorporated, either explicitly or implicitly [4-9], exogenous macroeconomic and time effects. That work has had some success in predicting bank failures but predicting within the same time period seems more in the spirit of descriptive rather than inferential statistics, and it certainly is not a form of forecasting. Using explicit macroeconomic conditions [8] presents the problem of having to predict the economy at large.

Further, our work does not develop an early warning system of any particular duration [9], only whether or not the bank will fail in the next quarter. The desired amount of warning depends on the user of the system. While an early warning system may be ideal for a regulator who would like to address bank mismanagement and avoid a failure, the trader of credit default swaps on banks might find this less helpful.

Indeed, to avoid excessively paying the spread on the swaps (a premium, in insurance terms), the ideal system for such a trader would give a tighter bound on the expected failure date, not a vague notion that a bank is likely to fail within the next two years. However, that does not account for liquidity and the availability of credit default swaps at the moment the trader wants them, and purchase and pricing of the swaps would be a judgment call by the particular trader. For these reasons, we developed a baseline system that performs a binary classification of failure or success.

1. **Description of the Data**

The Federal Deposit Insurance Corporation (FDIC) in the United States keeps records of banking data in its Statistics on Depository Institutions (SDI) [2]. This was selected for use in this study because this data was analogous to the official Turkish statistics used in several comparable studies (e.g., [4]-[7],[9]).

SDI record-keeping goes back to the 1992Q4 and continues through to the present. Data are reported for the end of each fiscal quarter. Data analysis began after the release of the 2017Q2 data, which was taken as the cutoff point for this work. The number of variables was dynamic: 1,040 in 1992Q4, and 1,076 in 2017Q2. The number of banks followed a strictly decreasing trend throughout the entire epoch, with fewer banks every quarter since 1992Q4.

It is interesting to note that data for the post-2000 epoch did capture numerous macroeconomic events. The epoch spanned the end of the Clinton years, the post-9/11 recession, the Global Financial Crisis, the post-Crisis bull market, and the “Trump run” bull market acceleration. Hence, the data was suitable for investigating the extent to which failures could be predicted without explicit knowledge of these events.

Organizing and preprocessing this data set to make it suitable for machine learning experiments proved to be quite a challenge. For example, tracking an institution by name does not always work. Many banks change names over time (e.g., Chase Manhattan acquired JPMorgan and renamed the company JPMorgan Chase). Acquisitions further muddy the waters (e.g., Chemical Bank purchased Chase Manhattan but continued to use the Chase Manhattan name, even though Chase was the smaller bank in terms of assets). Nonetheless, the merger and acquisition activity had to be included in the data.

An alternative to the bank name is to use the FDIC Certificate Number, which is labeled “cert” in the FDIC database. Each bank is assigned a unique cert by the FDIC for issuing insurance certificates [19]. Prior to the Chase acquisition, Chemical Bank had cert 628. After the acquisition, the combined company still used cert 628. Likewise, the entirety of JPMorgan Chase uses cert 628, even after the acquisition of JPMorgan by Chase Manhattan. Unfortunately, these kinds of transformations are extremely disruptive to machine learning algorithms attempting to model relationships between inputs and outputs since there are discontinuities in the input data. There are no input variables that describe or represent these changes explicitly, so external knowledge is required to fully interpret the data.

Perhaps the most challenging aspect of working with SDI data is the abundance of missing data. There are quarters where variables are completely missing for all banks (e.g., trading liabilities, labeled “tradel,” is completely missing for 1992Q4, though the variable is included in the header). For example, as summarized in Table 1, out of 1,076 variables that occurred in 70 quarters of data, there were only 60 variables that were complete.

Due to the abundance of missing data, we had to focus on deep learning techniques and training strategies that are robust to missing data. Alternatively, we could have focused on techniques that attempt to synthesize missing data points, but these techniques are much more experimental in nature and hence not suitable for a baseline study. Two approaches were taken to combat the issue of missing data. First, the time period considered was restricted to 2000Q1 to the present (2017Q2). Record-keeping improved over time, and this has an added effect of not modeling old data from the time series. Second, we defined a “complete variable” as a quantitative variable which had no missing data for the period under consideration. Only 60 variables for the post-2000 epoch qualified as complete variables out of over 1,000 possible variables. The others were discarded either for having missing data somewhere in post-2000 epoch, or because they were categorical variables like address.

**Table 1.** Aggregate statistics for a subset of the data extending from 2000Q1 to 2017Q2 are shown.

|  |  |
| --- | --- |
| **Variable** | **Quantity** |
| Banks | 11,576 |
| Quarters | 70 |
| Variables | 1,076 |
| Complete Variables | 60 |
| Bank Failures | 558 |

Formatting data to be amenable to processing by approaches such as CNN also proved challenging due to the way the SDI data were organized. The FDIC organized all data by quarter, rather than by bank, as is necessary for predicting classifications of the banks. Further, each quarter is divided into approximately 65 categories in separate spreadsheets organized by topics such as “assets & liabilities,” “derivatives,” or “securities.” Consequently, data had to be merged between files within each quarter. These files contained duplicated variables, such as finding derivatives (“obsdir”) in both the “assets & liabilities” spreadsheets and the “derivatives” spreadsheets. Further complicating the issue was the fact that variables did not correspond to the same column indices in every quarter.

The process of condensing these data so they were organized by bank was not trivial since the post-2000 epoch contains 4,278 files of data with 11,536 cert numbers. Each quarter had its spreadsheets condensed into one master spreadsheet for the entire quarter, containing no duplicated variables. There were 70 such files, each of which had to be opened 11,536 times for a total of 807,520 file openings. This still resulted in prohibitively long run times, even spread over 12 parallel jobs of 1,000 banks each. Since software was scripted in Python, this added an additional level of inefficiency.

Once the data were properly preprocessed, they still needed to be organized into training and testing data so that pilot experiments could be run. We also needed to integrate failure data. We did not consider blind evaluation data in our pilot experiments due to the limited amount of data and the fact that we were not going to do excessive parameter tweaking. In this study, we were mainly interested in assessing the suitability of the data for this type of research.

The FDIC provides a list of bank failures and the date of failure. Exactly 500 failed banks existed for 20 or more quarters during the post-2000 epoch, and thousands of successful banks existed for 20 quarters. Those 500 failed banks were partitioned into 400 for training and 100 for testing. The successful banks were allocated at random into a training set of 400 and a testing set of 100, giving both the training and testing sets an equal number of failed and healthy banks.

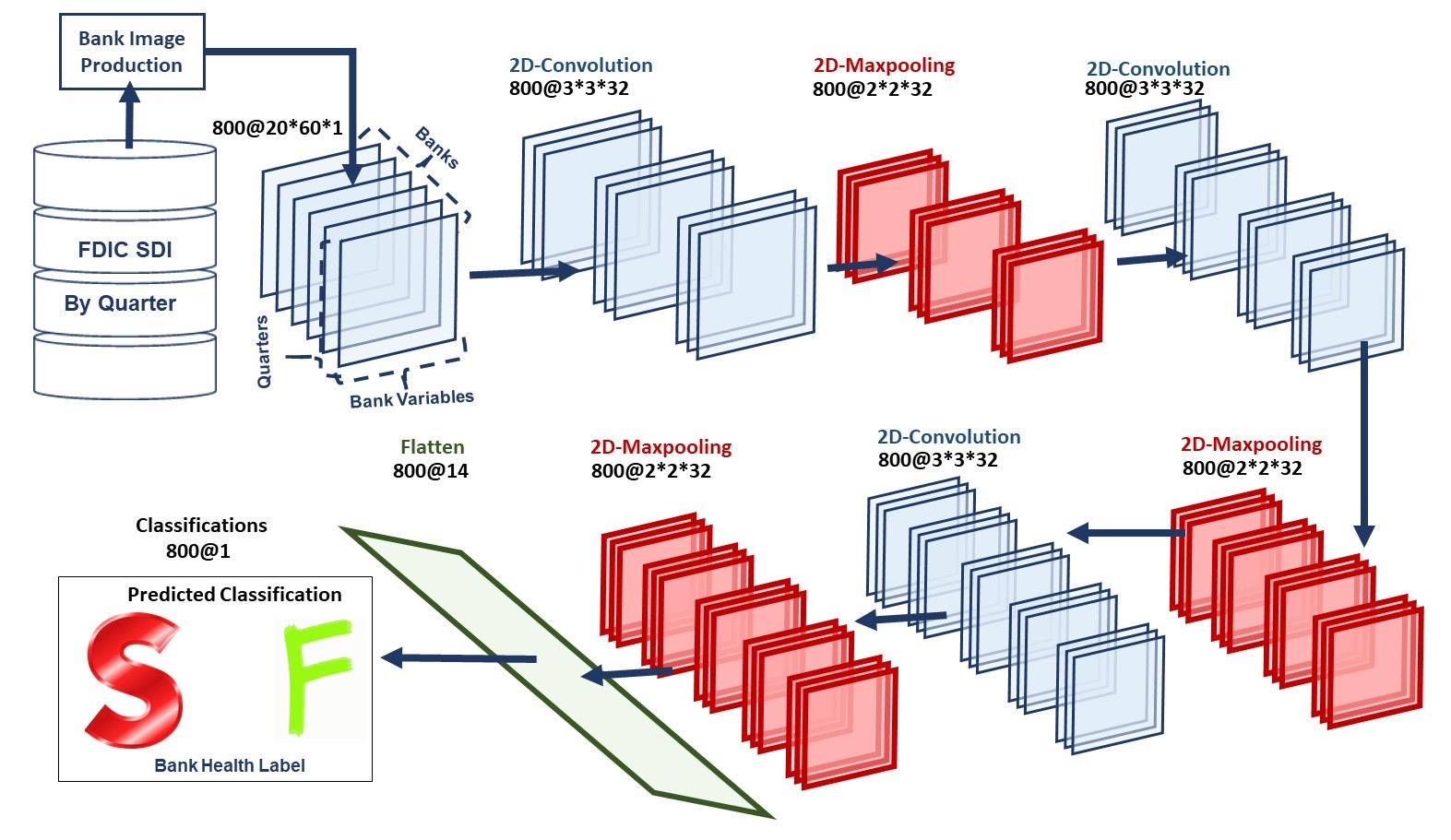
To run our deep learning algorithms, which generally prefer to see data of the same duration, the data for each bank were cropped to span equal lengths of time. For the failed banks, cropping was done at the end of the last quarter for which the bank reported data. For successful banks, the starting dates were selected at random. Sliding frames of bank data were designed to span 20 consecutive quarters. This duration was optimized by series of informal experiments using closed-loop training and evaluation.

1. **Approach**

For this study we leveraged a deep learning architecture that we have been developing for a variety of applications including speech processing, EEG analysis, and digital pathology [11][20][21]. The front end of these systems uses a recursive structure based on CNN and Long Short-Term Memory (LSTM) networks. Since the CNN approach, which captures temporal and spatial correlations, was pioneered for image processing applications, the time series for each bank will be referred to as its “bank image,” as each is a multichannel time series. The technique has been successful in the assessment of multichannel time series and produces results superior to more traditional techniques such as k-nearest neighbors and random forests [11][12]. We also include results for a pilot baseline system we developed based on logistic regression in which vectors were flattened to a length 1,200.

Baseline deep learning performance was evaluated using a fully-connected artificial neural network with one hidden layer with 512 nodes, 50% dropout, and “ReLU” activation. The batch size was 32, and 10 training epochs were used. The training data set consisted of 20x60 bank images. This approach was stepped up to a 20-layer multilayer perceptron neural network using steps of 2, 3, 4, 5, 6, 7, 8, 9, 10, 15, 20.

A CNN was implemented to inspect the data for correlations. Filtering layers used 3x3 filters. Three rounds of 3x3 filtering were performed, each followed by 2x2 maxpooling. This approach, depicted in Fig. 2, follows our work in other types of deep learning research [20][21]. Finally, a fully-connected layer of 512 nodes was used after all CNN steps, and the 512 nodes mapped to the outcomes of a successful or failed bank. A dropout of 25% was used after all three maxpooling steps.



**Fig. 2.** A convolutional neural network (CNN) of three layers of 3x3 filtering, each followed by a layer of 2x2 maxpooling, was used as a baseline system.

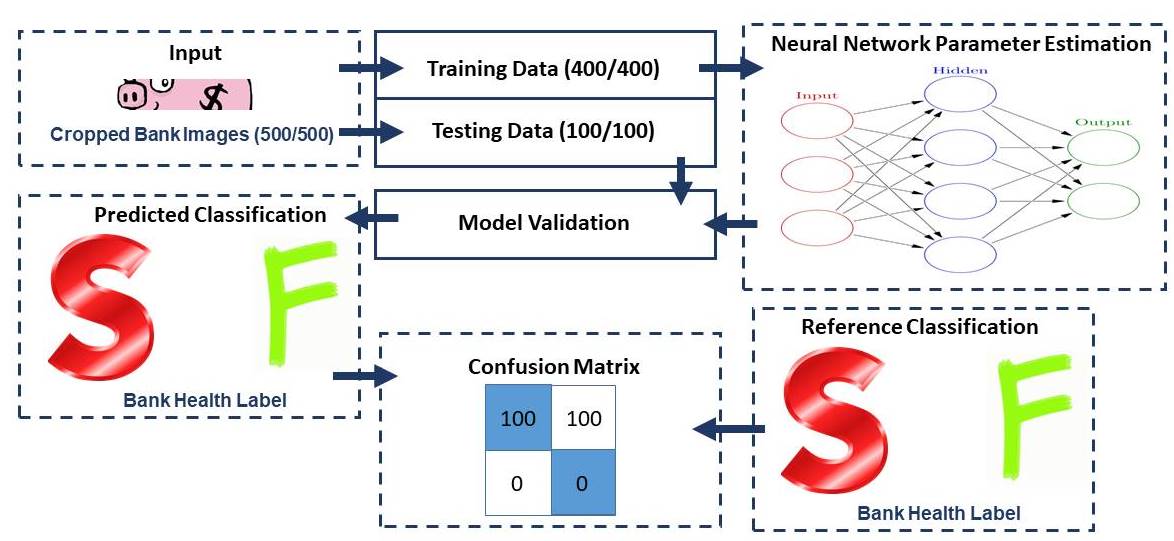
Marginal time series differences and second-order differences were applied to the bank images, with the differenced data run through the 3x3-filtering convolutional and fully-connected neural networks. Throughout all experiments, a binary cross-entropy loss function was used along with the Adam optimizer [20].

1. **Results**

Logistic regression failed to separate the distributions to any significant (α < 0.05) extent, even on the training data. In-sample results were 56% accuracy, 65% sensitivity, and 47% specificity, while out-of-sample results were 51% accuracy, 57% sensitivity, and 44% specificity. Because of the poor performance on the training data, these results suggested that the logistic regression was an underfit, high-bias model. The opportunity to fit hundreds of thousands or even millions of parameters in neural networks motivated us to take a closer look at deep learning approaches, as we searched for bias reduction to correct this underfitting.

Baseline deep learning performance with a one hidden layer neural network was unable to beat random guessing based on the prior distribution of 50% successful and 50% failed banks (α < 0.05), even on closed-set experiments on the training data. The system was able to find two distributions, but seemingly only because the input data gave the two groups of successful and failed banks. Performance did not significantly improve as the neural network was extended to 20 hidden layers. This network had over 24 million parameters when dropout was set to 0% to induce overfitting. However, the network could not accurately classify the training data under this overfitting condition.

Most surprisingly, the multilayer perceptron (MLP) networks were unable to produce two distributions. As larger and more parameter-heavy networks were explored, results tended to give zero values along an entire row of the confusion matrix: 100% sensitivity with 0% specificity, or vice versa, as shown in Fig. 3. Further, the same model setup could be run multiple times and not give the same result. Some runs gave 100% sensitivity and 0% specificity, while the next run of the same model would give the reverse. CNN gave this same lack of performance. This indicates that the models found nothing in the data onto which they could latch and separate the data.



**Fig. 3.** The results of closed-loop analysis using an MLP network are shown.

1. **Discussion**

The inability even to overfit to the training data was particularly surprising, especially since previous work [4]-[9] was able to produce strong accuracy in predicting bank failures, both in terms of sensitivity and specificity. These results raise a number of questions. The first is a criticism of the techniques used, particularly given the somewhat small amount of data used to build the fully-connected and convolutional neural networks. Why use such complicated modeling techniques? That the neural networks were unable even to overfit indicates that an overly complicated modeling technique for the data size is not the culprit of poor performance. Moreover, logistic regression was unable to separate the successful and failure distributions. Since all classifiers failed on training data, we conclude that the FDIC data [2] are inadequate to solve this bank failure question.

The second question concerns the time sensitivity of bank health. In particular, do exogenous factors contribute to bank failures? Intuitively, this seems like it would be the case. Before and after the Global Financial Crisis, bank failures were rare. During the GFC, however, bank failures spiked. It is hard to dismiss this as coincidence. This brings up the frightening prospect that the generally good economy since the end of the GFC has masked problems at banks that simply have yet to experience another exogenous trigger of failure.

The third question is if the variables in this study are adequate to capture the health of a bank. Indeed, the variables were chosen due to limitations of the SDI and the requirements of CNNs that preclude the use of variables with missing data. Further work could allow for some missing data but not having such a hard restriction of excluding any variables that had even one missing datum for any bank across the entire time span. For variables with the occasional missing datum, various techniques could be used to fill in the gaps, ranging from simple interpolation to more complex techniques such as autoencoders [22].

Once the 20-layer fully-connected neural network showed no ability to separate the training data, the prospects for the other techniques looked rather grim. Indeed, high parameter counts for models of the data produced no separation of the data. The lack of even producing two classifications suggests that the data for both failed and successful banks are generated by the same process, the opposite of the expectation that the successful banks have their processes of good management while the failed banks have their processes of mismanagement.

1. **Conclusion**

It appears that the information necessary to classify a bank as healthy or unhealthy is not endogenous to the signals used in this study. High-parameter neural networks were unable to give performance significantly (α < 0.05) better than random guessing based on the 50/50 prior distribution, even on the training data.

Endogenous bank health predicted solely from the SDI data seems to be an intractable problem. Introduction of data capturing exogenous factors seems critical. The data used in this study come from the reputable source of the American Federal Government, yet existing state-of-the-art classification techniques did not work to make the ever-critical prediction of banks in danger of failure.

Regardless of data set and the problems present (such as missing data), perhaps the biggest challenge of a neural network approach to bank health is that failed banks necessarily have shorter time series than successful banks. The limited amount of data makes it difficult to build complex models. This is further complicated by the fact that the data spans different periods in time and the durations vary. Signals of uneven length are routine in some disciplines like speaker and language recognition. Techniques have been developed to normalize the inputs to neural network systems so that each input has the same number of elements [23][24]. Such normalization techniques are worth exploring in future studies on bank health. Similarly, we have yet to explore data augmentation procedures for bank health prediction though such procedures have been extremely effective in disciplines like speech and image processing.

Deep learning has shown great promise in many fields, making this a very exciting time in the history of machine learning and artificial intelligence. However, even the best algorithms cannot overcome problems created by poor data. Future research needs to address improvements in the quantity and quality of the data, particularly including quantitative measures for exogenous factors.

References

1. Greenspan, A.: Never Saw It Coming: Why the Financial Crisis Took Economists by Surprise. Foreign Aff. 92, 88–96 (2013).
2. Federal Deposit Insurance Corporation: Failures and Assistance Transactions - Historical Statistics on Banking, https://www5.fdic.gov/hsob/SelectRpt.asp?EntryTyp=30& Header=1&tab=bankFailures. last accessed 2018/12/08.
3. Financial Crisis Inquiry Commission: The Financial Crisis Inquiry Report: Final Report of the National Commission on the Causes of the Financial and Economic Crisis in the United States. U.S. Government Printing Office, Washington, DC, USA (2011).
4. Erdogan, B.E.: Prediction of Bankruptcy Using Support Vector Machines: An Application to Bank Bankruptcy. J. Stat. Comput. Simul. 83, 1543–1555 (2013).
5. Erdogan, B.E., Akyüz, S.Ö.: A Weighted Ensemble Learning by SVM for Longitudinal Data: Turkish Bank Bankruptcy. In: Tez, M. and von Rosen, D. (eds.) Trends and Perspectives in Linear Statistical Inference. pp. 89–103. Springer International Publishing, Istanbul, Turkey (2018).
6. Erdogan, B.E., Egrioglu, E.A.: Support Vector Machines vs Multiplicative Neuron Model Neural Network in Prediction of Bank Failures. Am. J. Intell. Syst. 7, 125–131 (2017).
7. Boyacioglu, M.A., Kara, Y., Baykan, Ö.K.: Predicting Bank Financial Failures Using Neural Networks, Support Vector Machines and Multivariate Statistical Methods: A Comparative Analysis in the Sample of Savings Deposit Insurance Fund (SDIF) Transferred Banks in Turkey. Expert Syst. Appl. 36, 3355–3366 (2009).
8. Zheng, C., Cheung, A., Cronje, T.: Bank Liquidity, Bank Failure Risk and Bank Size. In: The 2016 ECU Business Doctoral and Emerging Scholars Colloquium. pp. 1–32, Edith Cowan University, Perth, Western Australia (2016).
9. Erdogan, B.E.: Bankruptcy Prediction of Turkish Commercial Banks Using Financial Ratios. Appl. Math. Sci. 2, 2973–2982 (2008).
10. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature. 521, 436–444 (2015).
11. Lopez, S.: Automated Identification of Abnormal Adult Electroencephalograms, Department of Electrical and Computer Engineering, Temple University, Philadelphia, Pennsylvania, USA (2017).
12. Lopez, S., Suarez, G., Jungreis, D., Obeid, I., Picone, J.: Automated Identification of Abnormal EEGs. In: IEEE Signal Processing in Medicine and Biology Symposium. pp. 1–4., Philadelphia, Pennsylvania, USA (2015).
13. Bao, W., Yue, J., Rao, Y.: A Deep Learning Framework for Financial Time Series Using Stacked Autoencoders and Long Short-Term Memory. PLoS One. 12, 25 (2017).
14. Addo, P.M., Guegan, D., Hassani, B.: Credit Risk Analysis Using Machine and Deep Learning Models. Risks. 6, 38 (2018).
15. Tam, K.: Neural Network Models and the Prediction of Bank Bankruptcy. Omega Int. J. of Mgmt. Sci. 19(5), 429-445 (1991).
16. Tam, K., Kiang, M.: Managerial Applications of Neural Networks: The Case of Bank Failure Predictions. Manage. Sci. 38, 926–947 (1992).
17. Le, H.H., Viviani, J.-L.: Predicting bank failure: An improvement by implementing a machine-learning approach to classical financial ratios. Res. Int. Bus. Financ. 44, 16–25 (2018).
18. Abdou, H., Abdallah, W., Mulkeen, J., Ntim, C., Wang, Y.: Prediction of Financial Strength Ratings Using Machine Learning and Conventional Techniques. Invest. Manag. Financ. Innov. 14, 194–211 (2017).
19. FDIC: Bankfind Glossary, https://research.fdic.gov/bankfind/glossary.html, last accessed 2017/10/01.
20. Shah, V., Golmohammadi, M., Ziyabari, S., von Weltin, E., Obeid, I., Picone, J.: Optimizing Channel Selection for Seizure Detection. In: Obeid, I. and Picone, J. (eds.) Proceedings of the IEEE Signal Processing in Medicine and Biology Symposium. pp. 1–5. IEEE, Philadelphia, Pennsylvania, USA (2017).
21. Golmohammadi, M., Ziyabari, S., Shah, V., Obeid, I., Picone, J.: Gated Recurrent Networks for Seizure Detection. In: Obeid, I. and Picone, J. (eds.) Proceedings of the IEEE Signal Processing in Medicine and Biology Symposium. pp. 1–5. IEEE, Philadelphia, Pennsylvania, USA (2017).
22. Beaulieu-Jones, B.K., Moore, J.H.: Missing Data Imputation in the Electronic Health Record Using Deeply Learned Autoencoders. In: Pacific Symposium on Biocomputing. pp. 207–219, Puako, Hawaii (2017).
23. Ghahremani, P., Sankar Nidadavolu, P., Chen, N., Villalba, J., Povey, D., Khudanpur, S., Dehak, N.: End-to-end Deep Neural Network Age Estimation. In: INTERSPEECH. pp. 277–281, Hyderabad, India (2018).
24. Snyder, D., Garcia-Romero, D., Sell, G., Povey, D., Khudanpur, S.: X-Vectors: Robust DNN Embeddings for Speaker Recognition. In: 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 5329–5333, Alberta, Canada (2018).